

Artigo

# Linguagem Python e Jupyter Notebook: uma proposta didática a fim de determinar o preço adequado à venda utilizando conceitos de Álgebra Linear, Cálculo Diferencial e Métodos Numéricos

Souza, L. F. S.<sup>1\*</sup>, Rodriguez, B. D. A.<sup>2</sup> and Poffal, C. A.<sup>3</sup>

<sup>1</sup> Universidade Federal do Rio Grande; Instituto de Matemática, Estatística e Física; [luisfernandessaucedosouza4@gmail.com](mailto:luisfernandessaucedosouza4@gmail.com)

<sup>2</sup> Universidade Federal do Rio Grande; Instituto de Matemática, Estatística e Física; [barbararodriguez@furg.br](mailto:barbararodriguez@furg.br)

<sup>3</sup> Universidade Federal do Rio Grande; Instituto de Matemática, Estatística e Física; [cristianaandrade@furg.br](mailto:cristianaandrade@furg.br)

\* Correspondência: [luisfernandessaucedosouza4@gmail.com](mailto:luisfernandessaucedosouza4@gmail.com)

Received: 18/01/2024; Accepted: 25/01/2024; Published: 31/01/2024

**Resumo:** Este trabalho tem como objetivo apresentar uma proposta didática, prevista no projeto de pesquisa e ensino intitulado “Laboratório de Matemática Aplicada e Computacional”, onde quer-se determinar o preço adequado de venda de um produto, a fim de maximizar o lucro de uma empresa fictícia utilizando a linguagem Python, a plataforma Github e a ferramenta computacional Jupyter Notebook. Para atingir os objetivos propostos, obtêm-se as funções que melhor descrevem a previsão de demanda e do lucro através de uma interpolação polinomial. Os métodos de Newton, Secante e Bissecção são aplicados na solução do problema. Os resultados e um estudo comparativo, bem como os *notebooks* gerados, são apresentados na plataforma GitHub. A presente proposta pretende, através de uma atividade didática, desenvolver a capacidade dos acadêmicos de organizar ideias e promover a apreensão de conceitos de Cálculo Diferencial, Álgebra Linear e Métodos Numéricos.

**Palavras-chave:** Python; Métodos Numéricos; Álgebra Linear; Cálculo Diferencial; Ensino.

Article

# Python Language and Jupyter Notebook: a didactic proposal to determine the appropriate selling price using concepts from Linear Algebra, Differential Calculus, and Numerical Methods

Souza, L. F. S.<sup>1\*</sup>, Rodriguez, B.D.A.<sup>2</sup> and Poffal, C.A.<sup>3</sup>

<sup>1</sup> Federal University of Rio Grande; Institute of Mathematics, Statistics, and Physics; [luisfernandessaucedosouza4@gmail.com](mailto:luisfernandessaucedosouza4@gmail.com)

<sup>2</sup> Federal University of Rio Grande; Institute of Mathematics, Statistics, and Physics; [barbararodriguez@furg.br](mailto:barbararodriguez@furg.br)

<sup>3</sup> Federal University of Rio Grande; Institute of Mathematics, Statistics, and Physics; [cristianaandrade@furg.br](mailto:cristianaandrade@furg.br)

\* Correspondence: [luisfernandessaucedosouza4@gmail.com](mailto:luisfernandessaucedosouza4@gmail.com)

Received: 18/01/2024; Accepted: 25/01/2024; Published: 31/01/2024

**Abstract:** This work aims to present a didactic proposal foreseen in the research and teaching project titled "Laboratory of Applied and Computational Mathematics" where the goal is to determine the appropriate selling price of a product to maximize the profit of a fictional company using the Python language, the GitHub platform, and the computational tool Jupyter Notebook. To achieve the proposed objectives, functions that best describe the forecast of demand and profit are obtained through polynomial interpolation. The Newton, Secant, and Bisection methods are applied to solve

the problem. The results, along with a comparative study and the generated notebooks, are presented on the GitHub platform. This proposal aims to, through a didactic activity, develop the ability of students to organize ideas and promote the understanding of concepts in Differential Calculus, Linear Algebra, and Numerical Methods.

**Keywords:** Python; Numerical Methods; Linear Algebra; Differential Calculus; Teaching.

---

## 1. Introdução

O avanço tecnológico e a transformação digital vêm promovendo mudanças significativas nas relações interpessoais, no mercado de trabalho e também na área da educação, particularmente, no ensino superior. Cada vez mais percebe-se nos acadêmicos ingressantes um grande interesse na tecnologia e suas aplicações. A partir deste interesse e da “exigência da atualidade” de uma formação de indivíduos que pensem de forma flexível, crítica e criativa (BARBOSA; RODRIGUEZ; POFFAL, 2020), iniciou-se o projeto de pesquisa e ensino intitulado “Laboratório de Matemática Aplicada e Computacional: solução de problemas utilizando a linguagem Python e os ambientes Anaconda e Jupyter Notebook”. O projeto apresenta aos acadêmicos dos cursos de Ciências Exatas e Engenharias ferramentas e metodologias de matemática aplicada, inserindo-os em atividades de pesquisa, e tem como objetivo investigar e propor métodos de solução computacional e interpretar resultados de problemas. Segundo Oliveira (2021), a matemática é uma maneira de estruturar raciocínios sobre problemas ou situações ao redor. Para Carolina da Costa, sócia da Mauá Capital para Finanças Sustentáveis e Impacto Oliveira (2021): “todas as profissões visam, em algum aspecto, resolver um problema; determinar quais variáveis o afetam e como estão relacionadas e de que forma é possível, ao longo do tempo, avaliar se as soluções propostas estão ainda resolvendo o problema”. Particularmente na área de Economia não é diferente e conceitos de Cálculo Diferencial, Álgebra Linear e Estatística, aliados a ferramentas computacionais, são aplicados na solução de problemas, tais como: determinar custo e receita total, obter curvas de maximização de lucro e prever modelos de crescimento. Perante este cenário e com os objetivos de desenvolver o pensamento estratégico e aproximar os acadêmicos do mundo de trabalho, este artigo descreve uma proposta didática na qual quer se obter a solução de um problema clássico da Economia: determinar o preço adequado de venda de um produto, a fim de maximizar o lucro de uma empresa aplicando conceitos de Álgebra Linear, Cálculo Diferencial e Métodos Numéricos, a linguagem Python, a plataforma GitHub (BATISTA et al., 2017) e o Jupyter Notebook (JUPYTER, 2023) como ferramentas computacionais. Destaca-se que a proposta apresentada tem a finalidade de expor aos discentes situações motivadoras permitindo construir um aprendizado conceitual, procedimental e atitudinal (aprendizagem significativa) (FRASSON; LABURÚ; ZOMPERO, 2019). A fim de atingir os objetivos propostos, o texto foi organizado da seguinte forma: na seção 2 é apresentado o referencial teórico que justifica a utilização da linguagem de programação para o ensino de conceitos matemáticos. Na seção 3 são descritos os problemas a serem solucionados e a metodologia utilizada para sua resolução. Finalmente, na seção 4, as considerações finais são apresentadas.

## 2. Breve referencial teórico

Atualmente, professores do ensino superior buscam estratégias, a fim de dinamizar o processo de aprendizagem, desenvolver nos alunos uma autonomia intelectual e a capacidade de investigar, propor soluções e interpretar resultados. Para Frasson, Luburú e Zompero (2019), a formação de competências nos alunos será mais bem-sucedida se as diferentes naturezas dos conteúdos forem lecionadas de forma complementar e concomitante, pois dessa forma os aprendizes tornam-se aptos a relacionar os conhecimentos científicos ao uso social dos saberes acadêmicos. A Teoria de Duval (DUVAL, 2003) destaca que há uma pluralidade de registros de representação de um mesmo objeto e que a articulação desses diferentes registros é a condição para a compreensão em matemática. Segundo Fassarella (2020), a descrição de um algoritmo em uma linguagem de programação constitui um registro de representação semiótica, pois cumpre as três características fundamentais: formação de uma representação identificável, tratamento e conversão (DUVAL, 2003). Dentro deste contexto e sob a perspectiva de que a atualidade exige a formação de indivíduos que pensem de forma flexível, crítica e criativa, a programação torna-se uma ferramenta fundamental, pois desenvolve a capacidade de organizar as ideias, memorizá-las e pode promover a apreensão de conceitos, principalmente quando ocorrem em uma conversão envolvendo outros registros (FASSARELA, 2020). Para Santos e Fassarella (2023), a resolução de um problema matemático por intermédio da programação (elaboração e codificação de um algoritmo) envolve uma conversão da linguagem natural (enunciado do problema) para o pseudocódigo ou linguagem de programação. Atualmente, em meio a tantas automatizações, saber programar torna-se um elemento-chave, tanto para compreender o funcionamento de um equipamento lógico programável, quanto para desenvolver seus próprios algoritmos. Para Menezes (2010), aprender uma linguagem de programação é treinar a mente a pensar de novas formas, o que contribui tanto para usar, quanto para desenvolver outros programas, uma vez que o pensamento adquirido desenvolve a capacidade de resolver problemas. Existem inúmeras linguagens de

programação com características diferentes entre si, desde a forma como as palavras e os símbolos são definidos, até mesmo a forma como as “instruções” dadas ao computador são traduzidas. No âmbito deste trabalho, optou-se por recorrer a uma aplicação interativa com crescente relevância no ensino e na investigação em todos os domínios: o Python. Esta linguagem vem se destacando como uma alternativa gratuita e que possui uma sintaxe simples e intuitiva o que torna o aprendizado mais natural e fluido. Destaca-se ainda, sua utilização nos cursos da área tecnológica como a primeira disciplina de programação para computadores (BARBOSA et al., 2019).

### 3. Materiais e métodos

O projeto consiste na seleção de um problema pelos professores responsáveis e o aluno, considerando o grau de conhecimento do acadêmico e qual área da matemática aplicada deseja estudar. Durante o ano de 2023, foram trabalhadas metodologias de solução mais simples, sob o ponto de vista matemático, envolvendo conceitos da área da Economia, tais como, lucro, receita e preço de equilíbrio de mercado. O enunciado do problema, bem como a teoria matemática empregada, são disponibilizados na plataforma GitHub (<https://github.com/LaboratorioPythonFURG/Preco-Venda-Metodos>) como mostra a Figura 1, escolhida para o gerenciamento e apresentação do material produzido à comunidade científica.

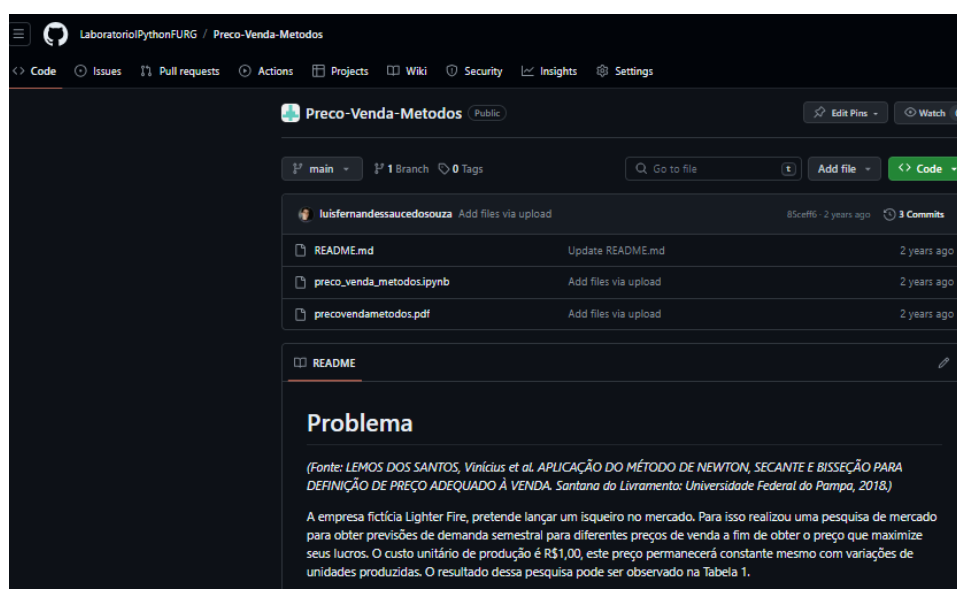


Figura 1. Plataforma GitHub.

Fonte: Os Autores.

Para a solução do problema proposto utilizou-se a linguagem Python no ambiente Jupyter Notebook, interface que possibilita a edição de *notebooks* (documento que permite código e texto) em um navegador web, possibilitando uma maior didática e compreensão dos códigos apresentados, focando no aprendizado do aluno. Os documentos gerados são do tipo *notebook* (.ipynb). Os textos apresentados, junto ao código, incluem uma linguagem simples de marcação chamada de *Markdown*. Além disso, o ambiente possui suporte de saída dos arquivos LaTeX e PDF, facilitando o compartilhamento do material produzido, permitindo maior clareza na resolução do problema e melhor organização dos arquivos. Dentro do Python foram utilizadas as bibliotecas NumPy (<https://numpy.org>) responsável por funções para manipulações de objetos do tipo array, SymPy (<https://www.sympy.org/pt/>) que permite trabalhar com matemática simbólica e a Matplotlib (<https://matplotlib.org>) para visualização através de gráficos e imagens. Particularmente, a solução do problema em determinar o preço adequado de venda de um produto fictício foi obtida a partir do emprego de conceitos de Álgebra Linear (ANTON, 2019), Cálculo Diferencial (SILVA, 2010) e os Métodos Numéricos da Secante, de Newton e da Bisseção (RUGGIERO e GOMES, 2019) implementados em linguagem Python. Nas subseções seguintes, apresentam-se a questão norteadora e sua solução computacional.

#### 3.1. Questão norteadora

A empresa fictícia *Lighter Fire* pretende lançar um novo produto no mercado: um isqueiro. Para isso, realizou uma pesquisa de mercado com o objetivo de obter previsões de demanda semestral para diferentes preços de venda, a fim de obter o preço que maximize seus lucros. O custo unitário de produção é R\$1,00, este preço permanecerá

constante mesmo com variações de unidades produzidas. O resultado dessa pesquisa pode ser observado na Tabela 1. Destaca-se que a questão norteadora foi extraída do trabalho de Santos et al. (2018).

**Tabela 1.** Pesquisa de Lucro e Demanda por Preço.

Preço (R\$)	Previsão de demanda (R\$)	Previsão de lucro (R\$)
2,0	500000	500000
2,5	450000	675000
3,0	350000	700000
3,5	200000	500000
4,0	40000	120000

Fonte: (SANTOS et al., 2018).

### 3.2. Determinação dos polinômios de interpolação

Para resolver o problema, primeiramente obtêm-se as funções que descrevem a previsão de demanda e de lucro através da definição de um polinômio interpolador. Particularmente, utiliza-se a matriz de Vandermonde (1), que descreve o sistema de equações matricialmente e gera um polinômio na forma (2).

$$\begin{bmatrix} x_1^{n-1} & x_1^{n-2} & \dots & x_1^1 & 1 \\ x_2^{n-1} & x_2^{n-2} & \dots & x_2^1 & 1 \\ \vdots & \vdots & \ddots & \vdots & 1 \\ x_n^{n-1} & x_n^{n-2} & \dots & x_n^1 & 1 \end{bmatrix} \begin{bmatrix} a_{n-1} \\ a_{n-2} \\ \vdots \\ a_0 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad (1)$$

$$p(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \dots + a_0x^0. \quad (2)$$

Os polinômios são obtidos a partir dos dados dispostos na Tabela 1 e com o emprego da biblioteca NumPy da linguagem de programação Python, como demonstrado nas Figuras 2 e 3.

```
##Importar biblioteca numpy
import numpy as np

##Definir variáveis demanda prevista
x = [2.0, 2.5, 3.0, 3.5, 4.0]
y = [500000, 450000, 350000, 200000, 40000]

##Resolver matriz 5x5 dos coeficientes por numpy
x = np.array([[x[0]**4, x[0]**3, x[0]**2, x[0], x[0]**0],
              [x[1]**4, x[1]**3, x[1]**2, x[1], x[1]**0],
              [x[2]**4, x[2]**3, x[2]**2, x[2], x[2]**0],
              [x[3]**4, x[3]**3, x[3]**2, x[3], x[3]**0],
              [x[4]**4, x[4]**3, x[4]**2, x[4], x[4]**0]])

y = np.array(y)

c = np.linalg.solve(x, y)

print(f"a4: {c[0]:.2f} a3: {c[1]:.2f} a2: {c[2]:.2f} a1: {c[3]:.2f} a0: {c[4]:.2f}")

a4: 26666.67 a3: -293333.33 a2: 1093333.33 a1: -1776666.67 a0: 1600000.00
```

**Figura 2.** Determinando os coeficientes polinômio demanda.

Fonte: Os Autores.

Destaca-se a utilização de funções próprias da linguagem Python e da biblioteca Numpy para a declaração de matrizes, forma escalonada de matrizes e solução de sistemas lineares. Os passos, apresentados na Figura 2, determinam os coeficientes do polinômio interpolador considerando preço e demanda. O polinômio obtido é dado por:

$$p_D(x) = 26666,67x^4 - 293333,33x^3 + 1093333,33x^2 - 1776666,67x + 1600000,00. \quad (3)$$

```
##Definir variáveis Lucro previsto
x = [2.0, 2.5, 3.0, 3.5, 4.0]
y = [500000, 675000, 700000, 500000, 120000]

##Resolver matriz 5x5 dos coeficientes por numpy
x = np.array([[x[0]**4, x[0]**3, x[0]**2, x[0], x[0]**0],
              [x[1]**4, x[1]**3, x[1]**2, x[1], x[1]**0],
              [x[2]**4, x[2]**3, x[2]**2, x[2], x[2]**0],
              [x[3]**4, x[3]**3, x[3]**2, x[3], x[3]**0],
              [x[4]**4, x[4]**3, x[4]**2, x[4], x[4]**0]])

y = np.array(y)

c = np.linalg.solve(x, y)

print(f"a4: {c[0]:.2f} a3: {c[1]:.2f} a2: {c[2]:.2f} a1: {c[3]:.2f} a0: {c[4]:.2f}")

a4: 80000.00 a3: -980000.00 a2: 4030000.00 a1: -6530000.00 a0: 4000000.00
```

Figura 3. Determinando os coeficientes polinômio lucro.

Fonte: Os Autores.

Na Figura 3 é apresentado o cálculo para os coeficientes do polinômio interpolador considerando preço e lucro. Obtendo-se o polinômio representado por:

$$p_L(x) = 80000,00x^4 - 980000,00x^3 + 4030000,00x^2 - 6530000,00x + 4000000,00. \tag{4}$$

Através da biblioteca Matplotlib, foi possível obter a representação gráfica, Figuras 4 e 5, para cada um dos polinômios interpoladores.

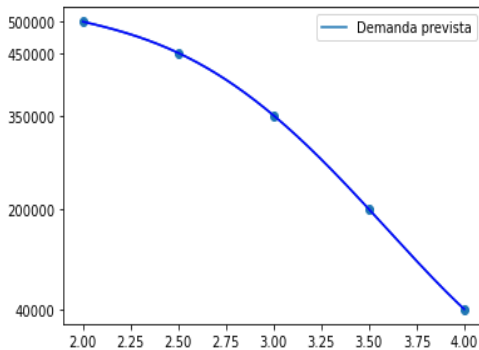


Figura 4. Gráfico do Polinômio de Demanda.

Fonte: Os Autores.

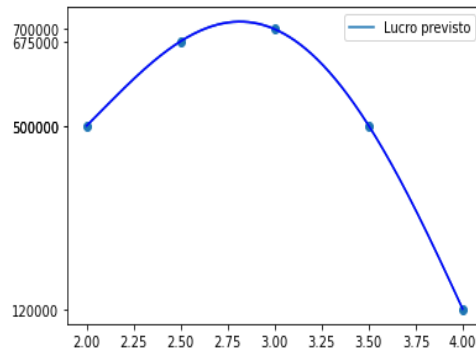


Figura 5. Gráfico do Polinômio de Lucro.

Fonte: Os Autores.

### 3.3. Maximizando o lucro da empresa Lighter Fire

Para maximizar o lucro da empresa, ou seja, determinar o valor máximo da função lucro representada pela equação (4), primeiramente, calcula-se a primeira derivada da função lucro em relação ao preço, utilizando-se a biblioteca Sympy e o comando diff (Figura 6).

```
import sympy as sp
x = sp.Symbol('x')
#definir função Lucro
flucro = 80000*x**4 - 980000*x**3 + 4030000*x**2 - 6530000*x + 4000000.00
#definir derivadas
#definir primeira derivada da função Lucro prevista
f = flucro.diff(x)
#definir segunda derivada da função Lucro prevista
df = f.diff(x)
print(f"f' = {f}\nf'' = {df}")
#definir derivada função Lucro
def f(x):
    return 320000*x**3 - 2940000*x**2 + 8060000*x - 6530000
#testar alguns preços e estudar os sinais da função
print(f" f'(0) | f'(1) | f'(2) | f'(2.5) | f'(3) | f'(4) | f'(5) | \n{f(0)} | {f(1)} | {f(2)} |
f' = 320000*x**3 - 2940000*x**2 + 8060000*x - 6530000
f'' = 960000*x**2 - 5880000*x + 8060000
f'(0) | f'(1) | f'(2) | f'(2.5) | f'(3) | f'(4) | f'(5) |
-6530000 | -1099000 | 399000 | 245000.0 | -170000 | -850000 | 270000 |
```

Figura 6. Definindo derivadas e estudo de sinal.

Fonte: Os Autores.

Para determinar o preço que maximiza o lucro, iguala-se a primeira derivada da função lucro a zero, obtendo-se uma equação polinomial de grau 3. Neste momento, aplicam-se os métodos numéricos de Newton (Figura 7), da Secante (Figura 8) e da Bissecção (Figura 9), a fim de obter o preço. Os métodos consistem, basicamente, em, a partir de uma aproximação inicial para a raiz, refinar essa aproximação através de um processo iterativo (RUGGIERO e GOMES, 2019). Tal aproximação inicial pode ser obtida por meio da análise gráfica ou a partir do estudo do sinal da primeira derivada da função lucro, determinando-se o intervalo (de preços) no qual o lucro atinge valor máximo (Figura 6). O critério de parada para os métodos de Newton e da Secante é o número de iterações e na Bissecção, um k ótimo em função da precisão desejada.

```
inicio = time.time()
#importar bibliotecas
from sympy import *
#definir derivada da função Lucro prevista
def f(x):
    return 320000*x**3 - 2940000*x**2 + 8060000*x - 6530000
#definir segunda derivada da função Lucro prevista
def df(x):
    return 960000*x**2 - 5880000*x + 8060000
#definir função Lucro
def flucro(x):
    return 80000*x**4 - 980000*x**3 + 4030000*x**2 - 6530000*x + 4000000.00
#definir função demanda
def fdemanda(x):
    return 26666.67*x**4 - 293333.33*x**3 + 1093333.33*x**2 - 1776666.67*x + 1600000.0
#definir parâmetros
x = symbols('x')
x0 = 2.5 #x inicial
k = 3 #iteracao
e = 0.001
#metodo newton
for i in range(k):
    sn = x0 - (f(x0)/df(x0))
    x0 = sn
fim = time.time()
tempo = fim - inicio
print(f"O preço que maximiza o lucro é R${sn}")
print(f"O lucro máximo será de R${flucro(sn)}")
print(f"A demanda será de {round(fdemanda(sn))} unidades")
print(f"Tempo de execução: {tempo}")
O preço que maximiza o lucro é R$2.8147972753348047
O lucro máximo será de R$715584.6087796651
A demanda será de 393739 unidades
Tempo de execução: 0.0019991397857666016
```

Figura 7. Método de Newton.

Fonte: Os Autores.

```

início = time.time()
#metodo secante
#definir parâmetros do intervalo da precisão e das iterações

xs0 = 2.5 #início intervalo
xs1 = 2.7 #ponto proximo início intervalo
e = 0.001 #precisao
k = 3 #iteracao
i = 0

while True:
    xs2 = xs1 - f(xs1)*(xs1-xs0)/(f(xs1)-f(xs0))

    if i == k:
        break;

    else:
        xs0 = xs1
        xs1 = xs2

    i = i + 1

fim = time.time()
tempo = fim - início

print(f"O preço pelo método da secante é R${xs2}")
print(f"O lucro máximo será de R${flucro(xs2)}")
print(f"A demanda será de {round(fdemanda(xs2))} unidades")
print(f"Tempo de execução: {tempo}")

O preço pelo método da secante é R$2.8147970162000047
O lucro máximo será de R$715584.6087796949
A demanda será de 393739 unidades
Tempo de execução: 0.001001596450005664

```

Figura 8. Método de Secante.

Fonte: Os Autores.

```

início = time.time()
import numpy as np

#definir parâmetros do intervalo da precisão e das iterações

a = 2.5 #início intervalo
b = 3.0 #fim intervalo
e = 0.001 #precisao
k = floor((np.log10(b-a)-np.log10(e))/np.log10(2)) #iteracao

for i in range(k):
    if (b - a) < e:
        sb = (a + b)/2

    m = f(a)
    sb = (a + b)/2

    if m*f(sb) > 0:
        a = sb
    else:
        b = sb

fim = time.time()
tempo = fim - início

print(f"O preço pelo método da bissecção é R${sb}")
print(f"O lucro máximo será de R${flucro(sb)}")
print(f"A demanda será de {round(fdemanda(sb))} unidades")
print(f"Tempo de execução: {tempo}")

O preço pelo método da bissecção é R$2.814453125
O lucro máximo será de R$715584.5564615428
A demanda será de 393813 unidades
Tempo de execução: 0.22519159317016602

```

Figura 9. Método de Bissecção.

Fonte: Os Autores.

### 3.4. Análise dos resultados

Uma vez implementados os algoritmos para os métodos de Newton e da Secante, obteve-se como resultado que o preço que maximiza o lucro da empresa é de R\$2,81 com um lucro de aproximadamente R\$715.584,61 para uma demanda de 393.739 unidades. Foram necessárias 3 iterações (Tabela 2). Para o método da Bissecção, o preço que maximiza o lucro da empresa é de R\$2,81 e o lucro é aproximadamente R\$715.584,56 para uma demanda de 393.813

unidades, após o total de 8 iterações (Tabela 2). De um modo geral, os três métodos apresentados resolveram o problema resultando no mesmo preço e com tempos abaixo de 0,01 segundos. Destaca-se que, como esperado, o número de iterações no método da Bisseção foi maior do que nos métodos de Newton e da Secante (RUGGIERO e GOMES, 2019). Porém, o método da Secante é mais rápido (tempo de execução) e mais simples, uma vez que não é necessário definir as derivadas como no de Newton.

**Tabela 2.** Resumo dos Métodos.

	Newton	Secante	Bisseção
Dados iniciais	$x_0 = 2,5$	$x_0 = 2,5, x_1 = 2,7$	[2,5, 3,0]
Lucro (R\$)	715.584,61	715.584,61	715.584,61
Iterações	3	3	8
Tempo (s)	0,001893	0,000379	0,001068

Fonte: Os Autores.

#### 4. Considerações Finais

Neste artigo apresentou-se uma proposta didática a fim de determinar o preço adequado de venda de um produto, objetivando maximizar o lucro de uma empresa fictícia utilizando a linguagem Python, as bibliotecas Numpy, Sympy e Matplotlib, a plataforma GitHub e a ferramenta computacional Jupyter Notebook. Para atingir os objetivos propostos, foram obtidas funções para descrever a previsão de demanda e do lucro, através de uma interpolação polinomial. Os métodos numéricos de Newton, Secante e Bisseção foram aplicados para a obtenção da solução do problema.

Acredita-se que a modelagem de um problema prático em Economia, com dados tabelados, a obtenção das funções de previsão de demanda e de lucro, a partir da solução de um sistema de equações lineares e a maximização dessas funções, relacionam conceitos de diferentes disciplinas e auxiliam o aluno a visualizar a aplicação social dos conteúdos estudados. Além disso, a manipulação dos diferentes tipos de registros, utilizados para obtenção da solução do problema (tabela, equação, gráfico), facilitam a compreensão matemática e conduzem o discente a uma aprendizagem significativa. O desenvolvimento de algoritmos e o emprego da linguagem de programação potencializam o processo de ensino e aprendizagem, permitindo um maior protagonismo. Quando o estudante escreve o código de um algoritmo ele relaciona os conceitos envolvidos e analisa a coerência dos resultados obtidos, indo além do conteúdo estudado em sala de aula.

Destaca-se, ainda, a aplicabilidade do ambiente Jupyter Notebook, uma ferramenta de aprendizado muito útil na modelagem de problemas e para a interpretação de seus resultados. Ela suporta tanto a linguagem de marcação *Markdown*, importante para a escrita de fórmulas e tabelas, como a linguagem de programação Python, que vem sendo amplamente utilizada em diversas áreas do conhecimento.

Por fim, como proposta futura, pretende-se dar continuidade ao projeto “Laboratório de Matemática Aplicada e Computacional” com a seleção e solução de novos problemas envolvendo conceitos econômicos, tal como investimentos no mercado de ações.

**Agradecimentos:** Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq, através do Programa Institucional de bolsas de Iniciação Científica – PIBIC, pelo auxílio financeiro que possibilitou a dedicação ao projeto.

#### Referências

1. Anton, H. A.; Rorres, C. *Álgebra linear com aplicações*, 10 ed.; Bookman: Porto Alegre, Brasil, 2012.
2. Barbosa, L. C.; Rodriguez, B. D. do A.; Poffal, C. A. O Ensino de Limites utilizando a Linguagem Python. *Anais do X Encontro Regional de Matemática Aplicada do Rio Grande Do Sul*, 2020.
3. Barbosa, A. C. de C.; Rojas, A.; Concordido, C. F. R.; Carvalhaes, C. G. Cálculo Diferencial e Integral de Funções de uma Variável com Python. 1. ed. Rio de Janeiro: Ciência Moderna, 2019.
4. Batista, N. A.; Alves, G. B.; Gonzaga, A. L.; Brandão, M. A..GitSED: Um Conjunto de Dados com Informações Sociais baseado no GitHub. *Proceedings of the Satellite Events of the 32nd Brazilian Symposium on Databases*, 2017.
5. Duval, R. Aprendizagem em Matemática: Registros de Representação Semiótica e Funcionamento Cognitivo da Compreensão em Matemática; Papirus: São Paulo, Brasil, 2003.



6. Fassarella, L. S. Desconexão procedimental e programação no ensino-aprendizagem da Matemática: considerações a partir da Teoria dos Registros de Representação Semiótica. *Revista Eletrônica de Educação Matemática – REVEMAT*, 2020, 15, pp. 01-24.
7. Frasson, F., Lamburú, C. E., Zompero, A. de F. Aprendizagem Significativa Conceitual, Procedimental e Atitudinal: Uma Releitura da Teoria Ausubeliana. *Contexto e Educação*, 2019, 34 (108), pp. 303-318.
8. Jupyter. Disponível em: <https://jupyter.org/>. Acesso em: 08 mai. 2023.
9. Menezes, N. N. C. Introdução à programação com Python: algoritmos e lógica de programação para iniciantes. São Paulo: Novatec Editora, 2010.
10. Oliveira, R. Economia e mundo do trabalho do amanhã dependem da matemática. Disponível em: <https://porvir.org/economia-e-mundo-do-trabalho-do-amanha-dependem-da-matematica>. Acesso em: 07 fev. 2023.
11. Pereira, B. A.; Pereira, S. A.; Filho, M. R. L. Educação Matemática e linguagens de programação: um levantamento de artigos sobre a Educação Básica. *Ensino da Matemática em Debate*, 2022, v. 9, n. 3, pp. 94-116.
12. Pereira, G. M. de A., Rodriguez, B. D. do A., Poffal, C. A. Linguagem Python e Jupyter Notebook: Um problema de distribuição de temperatura em uma placa. *Encontro Regional de Matemática Aplicada e Computacional RJ*, 2021.
13. Ruggiero, M. A., Gomes, V. L. *Cálculo Numérico: Aspectos teóricos e computacionais*, 2 ed.; Pearson Makron Books: São Paulo, Brasil, 2019.
14. Santos, Mônica Romana de Oliveira; Fassarella, Lúcio Souza. “Programação aplicada ao ensino de matemática sob a perspectiva da teoria dos registros de representação semiótica: uma experiência didática com as noções de variáveis dependente e independente na função afim”. *Revista Paranaense de Educação Matemática*, 2023, v. 12, n. 28, pp. 287-310.
15. Santos, V. L. dos, Lourenço, M. M., Granja, A. P., Lopes, Camila A., Blass, L. Aplicação do Método de Newton, Secante E Bisseção para Definição de Preço Adequado à Venda. *Anais do 10º Salão Internacional de Ensino, Pesquisa e Extensão – SIEPE*, 2018, v. 10, n. 1.
16. Silva, L. M. O. da; Machado, M. A. S. *Matemática aplicada à administração, economia e contabilidade: funções de uma e mais variáveis.*; Cengage Learning: São Paulo, Brasil, 2010.
17. Sgorla, K., Custódio, K. Z., Máximo, J. C. Projeto Novos Rumos 4.0: Pedagogia Crítica, Metodologias Ativas e Desenvolvimento Humano No Ensino De Programação Básica. *Texto Livre: Linguagem e Tecnologia*, 2020, v. 13, n. 1, pp. 120-136.