Análise Numérica de Estratégias de Reinvestimento em Produtos Financeiros com Juros Simples

Arthur Hilario Tembikoski

São Paulo
2025

Resumo

Este trabalho apresenta uma análise interdisciplinar que combina conceitos de Cálculo Numérico e Engenharia Econômica para investigar o comportamento de investimentos financeiros com juros simples submetidos a estratégias de reinvestimento periódico. O estudo desenvolve um modelo matemático que descreve a transformação de juros simples em crescimento equivalente a juros compostos através do reinvestimento sistemático. A pesquisa aborda dois problemas principais: a determinação do tempo necessário para alcançar um lucro adicional específico exclusivamente do reinvestimento, e o cálculo do período requerido para atingir uma Taxa Mínima de Atratividade (TMA) predeterminada. Utilizam-se métodos numéricos, particularmente o método de Newton-Raphson, para resolver as equações não-lineares resultantes do modelo. A implementação computacional em Python permite a simulação detalhada do comportamento do investimento, incluindo a geração de fluxos de caixa e visualizações gráficas. Os resultados demonstram que estratégias de reinvestimento podem aumentar significativamente o retorno de investimentos com juros simples, validando a importância da disciplina financeira e proporcionando ferramentas analíticas para tomada de decisão de investimento.

Palavras-chave: Reinvestimento, Juros Simples, Juros Compostos, Método de Newton-Raphson, Engenharia Econômica, Análise Numérica.

Conteúdo

Lista de Figuras

Lista de Tabelas

1 Introdução

1.1 Contextualização do Problema

Durante o desenvolvimento das disciplinas de Cálculo Numérico e Engenharia Econômica no quadrimestre 2025.1, identificou-se uma oportunidade significativa para aplicação interdisciplinar dos conhecimentos teóricos adquiridos em um problema prático do mercado financeiro. O cenário analisado envolve produtos de investimento que oferecem juros simples, mas permitem resgates e reinvestimentos periódicos, criando uma dinâmica financeira interessante para análise.

O problema financeiro específico investigado refere-se a um produto bancário que oferece juros simples com taxa anual X%, contratos com duração trimestral e possibilidade de resgate e reinvestimento a cada três meses. Um investidor que adota a estratégia de reinvestir integralmente os rendimentos obtidos a cada trimestre transforma a natureza do crescimento do seu capital, gerando um efeito cumulativo semelhante ao dos juros compostos.

1.2 Formulação do Problema

O problema central deste estudo pode ser dividido em duas questões principais:

1. Determinação do número de períodos (trimestres) necessários para obter um rendimento adicional de X% exclusivamente através do efeito do reinvestimento, excluindo a contribuição dos juros simples.

2. Cálculo do número de períodos (trimestres) requeridos para atingir uma Taxa Mínima de Atratividade (TMA) de X% ao ano, considerando o efeito combinado dos juros simples e do reinvestimento.

Estas questões possuem relevância prática tanto para investidores individuais quanto para instituições financeiras, pois permitem a avaliação quantitativa de estratégias de reinvestimento e o planejamento financeiro de longo prazo.

1.3 Objetivos da Pesquisa

Os objetivos específicos deste trabalho são:

- Desenvolver um modelo matemático rigoroso que descreva o crescimento de investimentos com juros simples sujeitos a reinvestimento periódico total.

- Derivar analiticamente as equações fundamentais que governam o comportamento do investimento, com separação explícita dos componentes de juros simples e compostos.

- Implementar e comparar métodos numéricos para a resolução das equações não-lineares resultantes do modelo.
- Desenvolver uma solução computacional completa em Python para simulação e análise do comportamento do investimento.
- Realizar análises de sensibilidade e visualizações gráficas para facilitar a interpretação dos resultados.
- Extrair conclusões práticas sobre a eficácia de estratégias de reinvestimento em produtos financeiros com juros simples.

2 Fundamentação Teórica

2.1 Conceitos Financeiros Básicos

Para a análise proposta, é fundamental compreender os conceitos financeiros envolvidos:

- **Juros Simples**: Modalidade de remuneração em que os juros são calculados exclusivamente sobre o capital inicial, sem incorporação dos juros acumulados ao capital para cálculo dos juros subsequentes.
- **Juros Compostos**: Modalidade de remuneração em que os juros de cada período são incorporados ao capital para o cálculo dos juros do período seguinte.
- **Reinvestimento**: Estratégia financeira que consiste em aplicar novamente os rendimentos obtidos em um investimento, aumentando assim o capital base para geração de novos rendimentos.
- **Taxa Mínima de Atratividade (TMA)**: Taxa de retorno mínima exigida por um investidor para considerar um investimento viável, considerando o custo de oportunidade do capital.

2.2 Modelagem Matemática do Problema

2.2.1 Definição de Variáveis

Para uma análise quantitativa precisa, definem-se as seguintes variáveis:

- VP: Valor Presente do investimento (capital inicial)
- i: Taxa de juros simples por trimestre (expressa em decimal)
- n: Número de períodos de investimento (trimestres)

- VF_n: Valor Futuro do investimento após n trimestres
- k: Lucro adicional gerado exclusivamente pelo reinvestimento (fração do capital inicial)
- TMA: Taxa Mínima de Atratividade anual (expressa em decimal)

2.2.2 Dedução do Modelo de Crescimento

Considerando a estratégia de reinvestimento total dos rendimentos a cada trimestre, o crescimento do investimento pode ser descrito recursivamente:

$$\begin{aligned}
\text{Após } 1^{\underline{o}} \text{ trimestre:} \quad & VF_1 = VP(1+i) \\
\text{Após } 2^{\underline{o}} \text{ trimestre:} \quad & VF_2 = VF_1(1+i) = VP(1+i)^2 \\
\text{Após } 3^{\underline{o}} \text{ trimestre:} \quad & VF_3 = VF_2(1+i) = VP(1+i)^3 \\
& \vdots \\
\text{Após n trimestres:} \quad & VF_n = VP(1+i)^n
\end{aligned}$$

Esta dedução revela um resultado fundamental: embora o produto financeiro ofereça apenas juros simples, a estratégia de reinvestimento total transforma o crescimento em um comportamento equivalente a juros compostos.

2.2.3 Separação de Componentes

A expressão $VF_n = VP(1+i)^n$ pode ser expandida utilizando o Teorema Binomial de Newton:

$$VF_n = VP(1+i)^n = VP\sum_{k=0}^{n}\binom{n}{k}i^k = VP\left[1 + ni + \frac{n(n-1)}{2!}i^2 + \frac{n(n-1)(n-2)}{3!}i^3 + \cdots + i^n\right] \tag{1}$$

Esta expansão permite identificar claramente os diferentes componentes do crescimento:

- $VP \cdot 1$: Capital inicial (componente constante)
- $VP \cdot ni$: Juros simples totais (componente linear)
- Termos de ordem superior: Efeito cumulativo dos juros compostos gerados pelo reinvestimento

A reorganização da fórmula permite uma separação explícita:

$$VF_n = VP(1 + ni) + VP[(1 + i)^n - (1 + ni)] \tag{2}$$

Onde:

- $VP(1 + ni)$: Valor com juros simples apenas (sem reinvestimento)
- $VP[(1 + i)^n - (1 + ni)]$: Valor adicional gerado exclusivamente pelo reinvestimento

2.2.4 Variáveis Analíticas Derivadas

Com base na separação dos componentes, definem-se duas variáveis analíticas fundamentais:

1. **Diferença entre juros compostos e simples** (k):

   $$k = (1 + i)^n - (1 + n \cdot i) \tag{3}$$

   Representa o lucro adicional, expresso como fração do capital inicial, gerado exclusivamente pela estratégia de reinvestimento.

2. **Percentual de Rendimento Anual Equivalente** (PRA):

   $$PRA = \left[\frac{(1 + i)^n - (1 + n \cdot i)}{n/4}\right] + 4i \tag{4}$$

   Converte o rendimento trimestral em uma taxa anual equivalente, considerando quatro trimestres por ano.

2.3 Métodos Numéricos Aplicáveis

2.3.1 Necessidade de Métodos Numéricos

As equações (3) e (4) são equações não-lineares em n que não possuem soluções analíticas fechadas em termos de funções elementares quando n é a variável desconhecida. Esta característica justifica a aplicação de métodos numéricos para obtenção de soluções aproximadas.

2.3.2 Método de Amostragem (Busca Incremental)

O método de amostragem constitui uma abordagem direta baseada em tentativa e erro sistemática:

1. Inicialização com valor inicial para n

2. Incremento de n em passos fixos Δn

3. Avaliação da função objetivo $k(n)$ em cada iteração

4. Interrupção quando $|k(n) - k_{\text{desejado}}| < \epsilon$

2.3.3 Método de Newton-Raphson

O método de Newton-Raphson representa um algoritmo iterativo sofisticado para encontrar raízes de funções não-lineares:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Este método apresenta convergência quadrática sob condições adequadas, oferecendo alta eficiência computacional.

2.3.4 Condições de Convergência do Método de Newton-Raphson

Para garantir a convergência do método de Newton-Raphson aplicado às equações não-lineares do modelo, é necessário verificar as condições do teorema de convergência para funções convexas. O teorema estabelece que, para uma função $f(x)$ contínua e diferenciável em um intervalo $[a, b]$, se:

1. $f(a) \cdot f(b) < 0$ (existe raiz no intervalo)

2. $f'(x) \neq 0$ para todo $x \in [a, b]$

3. $f''(x)$ mantém o mesmo sinal em $[a, b]$

Então o método de Newton-Raphson converge para a raiz única no intervalo, independentemente do chute inicial $x_0 \in [a, b]$.

Para o problema em análise, definem-se as seguintes funções e suas derivadas:

1. **Período para lucro k:**

$$\begin{aligned}
f_1(n) &= (1+i)^n - (1 + n \cdot i) - k \\
f_1'(n) &= (1+i)^n \cdot \ln(1+i) - i \\
f_1''(n) &= (1+i)^n \cdot [\ln(1+i)]^2
\end{aligned}$$

2. **Período para TMA**:

$$
\begin{aligned}
f_2(n) &= \frac{(1+i)^n - (1+n\cdot i)}{n/4} + 4i - TMA \\
f_2'(n) &= \frac{4\left[((1+i)^n \ln(1+i) - i)n - ((1+i)^n - (1+n\cdot i))\right]}{n^2} \\
f_2''(n) &= \frac{N(n)}{n^4}
\end{aligned}
$$

onde o numerador $N(n)$ é dado por:

$$
\begin{aligned}
N(n) = 4\Bigg[& n^2 \cdot (1+i)^n \ln(1+i) - 1 + n \cdot (1+i)^n [\ln(1+i)]^2 \\
& - [(1+i)^n \ln(1+i) - i] \\
& - 2n \cdot [n \cdot ((1+i)^n \ln(1+i) - i)] \\
& - [(1+i)^n - (ni+1)] \Bigg]
\end{aligned}
$$

Verificação para o problema do período necessário para alcançar um lucro adicional k Para $i = 0.05$ e $k = 0.20$, foram realizadas verificações no intervalo $[1, 25]$ trimestres:

Saída do código:

```
Valor da funcao_juros_simples_composto(0.05, 1, 0.2)  -0.2
Valor da funcao_juros_simples_composto(0.05, 25, 0.2)  0.9363549408993885
f(i, a, k)f(i, b, k) < 0 Intervalo [1,25] validacao: -0.18727098817987772

f'(i, a)f'(i, b) > 0 para validar que não tem raiz (f'(i, n) = 0)
no intervalo [1,25]: 0.0001416838517235493

f''(i, a)f''(i, b) > 0 para validar que nao tem troca de sinal
em f''(i, n) para todo n em [a,b] no intervalo [1,25]: 2.014887921977145e-05
```

Análise dos resultados:

- ✓ $f(1) \cdot f(25) = -0.1873 < 0$: Indica existência de raiz no intervalo.
- ✓ $f'(1) \cdot f'(25) = 1.417 \times 10^{-4} > 0$: A derivada não se anula no intervalo.
- ✓ $f''(1) \cdot f''(25) = 2.015 \times 10^{-5} > 0$: A segunda derivada mantém sinal positivo.

Todas as três condições do teorema de convexidade são satisfeitas, garantindo convergência quadrática do método de Newton-Raphson para esta equação.

Verificação para o problema do período necessário para atingir uma TMA específica Para $i = 0.05$ e $TMA = 0.40$, verificações no intervalo $[1, 30]$ trimestres:

Saída do código:

```
Valor da Funcao_Periodo_para_TMA(0.05, 1, 0.4)  -0.2
Valor da Funcao_Periodo_para_TMA(0.05, 25, 0.4)  0.04292565002008902
f(i, a, k)f(i, b, k) < 0 Intervalo [1,30] validacao: -0.008585130004017805

f'(i, a)f'(i, b) > 0 para validar que não tem raiz (f'(i, n) = 0)
no intervalo [1,30]: 5.4905651475722636e-05

f''(i, a)f''(i, b) > 0 para validar que nao tem troca de sinal
em f''(i, n) para todo n em [a,b] no intervalo [1,30]: 3.5484016950628785
```

Análise dos resultados:

- $\checkmark f(1) \cdot f(30) = -0.00859 < 0$: Indica existência de raiz no intervalo.
- $\checkmark f'(1) \cdot f'(30) = 5.491 \times 10^{-5} > 0$: A derivada não se anula no intervalo.
- $\checkmark f''(1) \cdot f''(30) = 3.5484 > 0$: A segunda derivada mantém sinal positivo.

Todas as três condições também são satisfeitas para esta equação, garantindo convergência quadrática do método.

3 Metodologia

3.1 Abordagem Computacional

A implementação computacional foi desenvolvida em Python, organizada em células Jupyter Notebook para facilitar a execução passo a passo e a visualização intermediária de resultados. A escolha do Python justifica-se pela sua ampla adoção na comunidade científica, pela riqueza de bibliotecas disponíveis e pela sintaxe clara e expressiva.

3.2 Estrutura do Código

O código foi organizado em seis células principais, cada uma com funcionalidade específica:

1. **Célula 1**: Entrada de parâmetros do investimento
2. **Célula 2**: Cálculos iniciais e definição de funções básicas
3. **Célula 3**: Cálculo do período para lucro k usando dois métodos numéricos
4. **Célula 4**: Cálculo manual de rendimento anual equivalente

5. **Célula 5**: Método de Newton-Raphson para encontrar período dado TMA

6. **Célula 6**: Geração do fluxo de caixa detalhado

3.3 Parâmetros da Simulação

Para análise prática, estabeleceram-se os seguintes parâmetros realistas:

- Valor Presente (VP): R$ 10.000,00
- Taxa de juros trimestral (i): 5% (0,05)
- Período de referência (n): 15 trimestres (3,75 anos)
- TMA anual (TMA): 40% (0,40)
- Lucro desejado (k): 20% (0,20)

3.4 Visualização de Dados

Para complementar a análise numérica, desenvolveram-se visualizações gráficas que permitem a compreensão intuitiva do comportamento do investimento. As visualizações incluem:

- Evolução temporal do valor acumulado e componentes de juros
- Comportamento do retorno anual equivalente
- Composição percentual dos juros ao longo do tempo
- Participação relativa dos juros compostos no total de juros

4 Implementação Computacional

4.1 Célula 1: Entrada de Parâmetros

```python
import numpy as np
import pandas as pd
import math

# Entrada dos par metros b sicos do investimento
vp = float(input("Digite o valor de vp: "))
i = float(input("Digite o valor de i (Juros Simples): "))
n = float(input("Digite o valor de n (Periodo): "))
TMA = float(input("Digite o valor da TMA (%): "))/100

```

```
print("\n\n")
```

Listing 1: Entrada de parâmetros do investimento

Descrição: Esta célula implementa a interface de entrada de dados, coletando os parâmetros fundamentais do modelo. A conversão da TMA para decimal assegura consistência nas operações matemáticas subsequentes.

4.2 Célula 2: Cálculos Iniciais

```
def Diff_Juros_Simples_Compostos(i,n):
    js = n*i + 1
    jc = ((i+1)**n)
    return jc - js

K = Diff_Juros_Simples_Compostos(i,n)
print(f"Valor % de K calculado para o Periodo {n}: {K*100}")
print(f"Valor % de lucro total (Simples + Composto) calculado para o Periodo {n}: {(n*i+K)*100}")
print(f"Valor total acumulado (aproximado): {vp*K*n*i*vp+vp}")
print(f"Rendimento anual: {(K*100/(n/4)) + i*4*100}")
print("\n\n")

k = float(input("Digite o valor % de lucro por reinvestimento (k): "))/100

def Juros_Total(i,k):
    return 1+n*i+k

juros_total = Juros_Total(i,k)
print(f"Calculando Juros total (%) com o valor de k = {k}: {(juros_total -1)*100}")
print(f"Valor de Lucro Juros compostos: {vp*k}")
print(f"Valor de Resgate total (aproximado): {round(vp*juros_total, 2)}")

print("\n\n")
```

Listing 2: Cálculos iniciais e definição de funções

Descrição: Realiza cálculos preliminares essenciais, incluindo a diferença entre juros compostos e simples e o valor total do investimento considerando ambos os componentes.

4.3 Célula 3: Cálculo do Período para Lucro k

```
def aproximacaoPi(i,n):
```

```python
    return round((i+1)**(n) - (n*i +1), 6)

# Versao 1 Amostragem
parte1 = 0
periodo = 0
iteracoes = 0
while (k != parte1):
    periodo += 0.000001
    parte1 = round(aproximacaoPi(i,periodo),6)
    iteracoes += 1

print(f"Calculo da raiz no loop: {parte1}")
print(f"Periodo aproximado para k = {k}: {periodo}")
print(f"Lucro % aproximado juros compostos: {k*100}")
print(f"Numero de interacoes: {iteracoes}")

# Versao 2 Newton-Raphson
def Funcao_juros_Simples_Composto(i,n,k):
    return (i+1)**(n) - (n*i + 1) - k

def Derivada_Funcao_juros_Simples_Composto(i,n):
    return (i+1)**(n) * math.log(i+1) - i

def Segunda_Derivada_Funcao_juros_Simples_Composto(i,n):
    return (i+1)**(n) * math.log(i+1)**2

# Teorema de convexidade
funcao_juros_simples_composto_1 = Funcao_juros_Simples_Composto(i, 1, k)
funcao_juros_simples_composto_25 = Funcao_juros_Simples_Composto(i, 25, k)
print(f" Valor da funcao_juros_simples_composto({i}, 1, {k}) {funcao_juros_simples_composto_1}")
print(f" Valor da funcao_juros_simples_composto({i}, 25, {k}) {funcao_juros_simples_composto_25}")
print(f" f(i, a, k)f(i, b, k) < 0 Intervalo [1,25] validacao: {funcao_juros_simples_composto_1*funcao_juros_simples_composto_25}")

print("\n")

derivada_a = Derivada_Funcao_juros_Simples_Composto(i, 1)
derivada_b = Derivada_Funcao_juros_Simples_Composto(i, 25)
validacao_derivada = derivada_a*derivada_b

print(f" f'(i, a)f'(i, b) > 0 para validar que n o tem raiz (f'(i, n) = 0) no intervalo [1,25]: {validacao_derivada} ")

derivada2_a = Segunda_Derivada_Funcao_juros_Simples_Composto(i, 1)
```

```python
derivada2_b = Segunda_Derivada_Funcao_juros_Simples_Composto(i, 25)
validacao_derivada2 = derivada2_a*derivada2_b

print(f" f''(i, a)f''(i, b) > 0 para validar que nao tem troca de sinal em f''(i, n) para todo n em [a,b] no intervalo [1,25]: {validacao_derivada2} ")

print("\n")

# Escolhendo um chute inicial no intervalo [1, 25]
n0 = 5  # Pode ser qualquer valor dentro do intervalo [1, 25]

for j in range(50):
    n0 = n0 - Funcao_juros_Simples_Composto(i,n0,k)/Derivada_Funcao_juros_Simples_Compos

print(f"Periodo (n) aproximado Metodo Newton-Raphson para k = {k} : {n0}")
```

Listing 3: Cálculo do período usando dois métodos diferentes

Descrição: Implementa e compara dois métodos numéricos para resolver a equação $k(n) = k_{desejado}$, proporcionando validação cruzada dos resultados.

4.3.1 Resultados obtidos

Saída do método de amostragem:

```
Calculo da raiz no loop: 0.2
Periodo (n) aproximado para k = 0.2: 12.109449997688804
Lucro % aproximado juros compostos: 20.0
Numero de interacoes: 12109450
```

Saída do método de Newton-Raphson:

```
Periodo (n) aproximado Metodo Newton-Raphson para k = 0.2 : 12.109463124395804
```

Ambos os métodos convergem para resultados muito próximos (12.10945 vs 12.10946 trimestres), validando a correção das implementações. O método de Newton-Raphson requer apenas 50 iterações, enquanto o método de amostragem necessita de mais de 12 milhões de iterações.

4.4 Célula 4: Cálculo de Rendimento Anual

```python
def Calcula_Porc_Juros_Anual(jurosimples,juroscompostos, periodo):
    qtde_ano = periodo/4
```

```
    return (juroscompostos/(qtde_ano)) + jurosimples

porc_lucro = Calcula_Porc_Juros_Anual(i*100*4,k*100+1, n0)
print(f"Porcentagem de rendimento anual % para k = {k*100}%:
   {porc_lucro}")
```

Listing 4: Cálculo manual do rendimento anual

Descrição: Converte o rendimento trimestral em taxa anual equivalente, facilitando a comparação com outros investimentos.

4.5 Célula 5: Cálculo do Período para TMA

```
def Funcao_Periodo_para_TMA(i,n,TMA):
    return (((i+1)**(n) - ((n)*i + 1))/(n/4)) + (4*i) - (TMA)

def Derivada_Funcao_Periodo_para_TMA(i,n):
    numerador = 4*(((((i+1)**n) * math.log(i+1) - i)*n) - (((i+1)**n) -
       (n*i+1)))
    denominador = n**2
    return numerador / denominador

def Segunda_Derivada_Funcao_Periodo_para_TMA(i,n):
    numerador = 4*((n**2) * ( (((i+1)**n) * math.log(i+1)) - 1 +
                  n*(((i+1)**n) * ((math.log(i+1))**2)) -
                  (((i+1)**n)*(math.log(i+1)) - i) -
                  (2*n*(n*( ((i+1)**n) * (math.log(i+1)) - i))) -
                  (((i+1)**n) - (n*i+1)*1)))
    denominador = n**4
    return numerador / denominador

# Versao 1 Amostragem
def calcula_porc_lucro(jurosimples,juroscompostos, periodo):
    return (juroscompostos/(float(periodo)/4)) + jurosimples

def calcula_k(i,n):
    new_var = (i+1)**(n)
    new_var0 = (n)*i + 1
    return new_var-new_var0

parte2 = 0
periodo = 0
PRA = 0
while (True):
    if(PRA >= TMA*100):
        break
    else:
```

```python
        periodo += 1
        K1 = calcula_k(i,periodo)
        PRA = calcula_porc_lucro(i*100*4,K1*100, periodo)

print(f"Periodo (n): {periodo}")
print(f"Rendimento total (juros composto) %: {K1*100}")
print(f"Porcentagem de rendimento anual: {PRA}")

# Versao 2 Newton-Raphson
# Teorema de convexidade
funcao_periodo_tma_1 = Funcao_Periodo_para_TMA(i, 1, TMA)
funcao_periodo_tma_30 = Funcao_Periodo_para_TMA(i, 30, TMA)
print(f" Valor da Funcao_Periodo_para_TMA({i}, 1, {TMA}) {funcao_periodo_tma_1}")
print(f" Valor da Funcao_Periodo_para_TMA({i}, 25, {TMA}) {funcao_periodo_tma_30}")
print(f" f(i, a, k)f(i, b, k) < 0 Intervalo [1,30] validacao: {funcao_periodo_tma_1*funcao_periodo_tma_30}")

print("\n")

derivada_a = Derivada_Funcao_Periodo_para_TMA(i, 1)
derivada_b = Derivada_Funcao_Periodo_para_TMA(i, 25)
validacao_derivada = derivada_a*derivada_b

print(f" f'(i, a)f'(i, b) > 0 para validar que n o tem raiz (f'(i, n) = 0) no intervalo [1,30]: {validacao_derivada} ")

derivada2_a = Segunda_Derivada_Funcao_Periodo_para_TMA(i, 1)
derivada2_b = Segunda_Derivada_Funcao_Periodo_para_TMA(i, 25)
validacao_derivada2 = derivada2_a*derivada2_b

print(f" f''(i, a)f''(i, b) > 0 para validar que nao tem troca de sinal em f''(i, n) para todo n em [a,b] no intervalo [1,30]: {validacao_derivada2} ")

print("\n")

p0 = 5
for j in range(50):
    p0 = p0 - Funcao_Periodo_para_TMA(i,p0, TMA) / Derivada_Funcao_Periodo_para_TMA(i,p0)

print(f"Periodo (n) para atingir a TMA {TMA*100}% usando o metodo Newton-Raphson: {p0}")
```

Listing 5: Método Newton-Raphson para encontrar período dado TMA

Descrição: Resolve a equação para encontrar o período necessário para atingir uma TMA específica, utilizando o método de Newton-Raphson.

4.5.1 Resultados obtidos

Saída do método de amostragem:

```
Periodo (n): 27
Rendimento total (juros composto) %: 138.3456322341576
Porcentagem de rendimento anual: 40.4956492198752
```

Saída do método de Newton-Raphson:

```
Periodo (n) para atingir a TMA 40.0% usando o metodo Newton-Raphson: 26.58344858609
```

O método de amostragem encontra 27 trimestres como solução inteira, enquanto o método de Newton-Raphson fornece uma solução mais precisa de 26.58 trimestres.

4.6 Célula 6: Geração do Fluxo de Caixa

```python
def juroscompostos(i,n):
    return ((i+1)**(n))

def jurossimples(i,n):
    return i*n

# Recoleta de par metros
vp = float(input("Digite o valor de vp: "))
i = float(input("Digite o valor de i: "))
n = float(input("Digite o valor de n: "))

# C lculo dos componentes
kc = juroscompostos(i,n)
ks = jurossimples(i,n)
kc_puro = kc-ks
k_total = kc_puro+ks

print(f"Valor de Kc calculado: {kc_puro}")
print(f"Valor de Ks calculado: {ks}")
print(f"Valor de K (Simples + Composto) calculado: {k_total}")
print(f"Valor de vp*k: {vp*k_total}")

# Gera  o do fluxo de caixa
print("\n\nFluxo de caixa:")
qtde_fluxo = int(input("Digite quanto tempo de fluxo de caixa: "))

```

```
# Inicializa  o de listas
trimestres = []
valores_acumulados = []
juros_simples_lista = []
juros_compostos_lista = []
retorno_anual_lista = []

n_periodo = 0
while (n_periodo <= qtde_fluxo):
    kc = juroscompostos(i, n_periodo)
    ks = jurossimples(i, n_periodo)
    kc_puro = kc - ks
    k_total = kc_puro + ks

    valor_acumulado = k_total * vp

    # C lculo do retorno anual
    if n_periodo > 0:
        retorno_anual = ((kc - 1) / (n_periodo/4)) * 100
    else:
        retorno_anual = 0

    # Armazenamento
    trimestres.append(n_periodo)
    valores_acumulados.append(valor_acumulado)
    juros_simples_lista.append(ks * vp)
    juros_compostos_lista.append(kc_puro * vp)
    retorno_anual_lista.append(retorno_anual)

    print(f"{n_periodo} Trimestre: {valor_acumulado:.2f}")
    n_periodo += 1
```

Listing 6: Geração do fluxo de caixa do investimento

Descrição: Gera fluxo de caixa detalhado do investimento, armazenando dados para posterior análise e visualização.

4.7 Célula 7: Código para Geração de Gráficos

```
import matplotlib.pyplot as plt
import numpy as np

# Dados da simula  o
trimestres = list(range(0, 16))
valor_acumulado = [10000, 10500, 11025, 11576.25, 12155.06, 12762.82,
                   13400.96, 14071, 14774.55, 15513.28, 16288.95,
                   17103.39, 17958.56, 18856.49, 19799.32, 20789.28]
```

```
juros_simples = [0, 500, 1000, 1500, 2000, 2500, 3000, 3500, 4000,
                 4500, 5000, 5500, 6000, 6500, 7000, 7500]
juros_compostos = [0, 0, 25, 76.25, 155.06, 262.82, 400.96, 571,
                   774.55, 1013.28, 1288.95, 1603.39, 1958.56,
                   2356.49, 2799.32, 3289.28]
retorno_anual = [0, 20, 20.5, 21.02, 21.55, 22.10, 22.67, 23.26,
                 23.87, 24.50, 25.16, 25.83, 26.53, 27.25, 28.00, 28.77]

# C lculo da participa  o percentual
participacao_compostos = []
for i in range(len(trimestres)):
    if (juros_simples[i] + juros_compostos[i]) > 0:
        participacao = (juros_compostos[i] / (juros_simples[i] +
            juros_compostos[i])) * 100
    else:
        participacao = 0
    participacao_compostos.append(participacao)

# Configura  o do estilo
plt.style.use('seaborn-v0_8-darkgrid')
fig, axs = plt.subplots(2, 2, figsize=(15, 12))
fig.suptitle('An lise Gr fica do Investimento com Reinvestimento
    Trimestral\nVP = R$ 10.000,00 | i = 5% a.t. | n = 15 trimestres',
    fontsize=16, fontweight='bold')

# Gr fico 1: Evolu  o do valor acumulado
axs[0, 0].plot(trimestres, valor_acumulado, 'b-', linewidth=2.5,
    marker='o', label='Valor Acumulado')
axs[0, 0].plot(trimestres, juros_simples, 'r--', linewidth=2,
    marker='s', label='Juros Simples Acum.')
axs[0, 0].plot(trimestres, juros_compostos, 'g:', linewidth=2,
    marker='^', label='Juros Compostos Acum.')
axs[0, 0].set_xlabel('Trimestre', fontsize=12)
axs[0, 0].set_ylabel('Valor (R$)', fontsize=12)
axs[0, 0].set_title('Evolu  o do Valor Acumulado e Componentes de
    Juros', fontsize=14, fontweight='bold')
axs[0, 0].legend(loc='upper left', fontsize=10)
axs[0, 0].grid(True, alpha=0.3)
axs[0, 0].set_xticks(trimestres)

# Gr fico 2: Retorno anual equivalente
axs[0, 1].plot(trimestres[1:], retorno_anual[1:], 'purple',
    linewidth=3, marker='D')
axs[0, 1].set_xlabel('Trimestre', fontsize=12)
axs[0, 1].set_ylabel('Retorno Anual Equivalente (%)', fontsize=12)
axs[0, 1].set_title('Evolu  o do Retorno Anual Equivalente',
    fontsize=14, fontweight='bold')
```

```python
axs[0, 1].grid(True, alpha=0.3)
axs[0, 1].set_xticks(trimestres)
for i, txt in enumerate(retorno_anual):
    if i > 0:
        axs[0, 1].annotate(f'{txt:.1f}%', (trimestres[i], retorno_anual[i]),
                          textcoords="offset points", xytext=(0,10),
                          ha='center', fontsize=8)

# Gr fico 3: Composi  o percentual dos juros
bar_width = 0.35
x_indices = np.arange(len(trimestres))
axs[1, 0].bar(x_indices - bar_width/2, juros_simples, bar_width,
              label='Juros Simples', color='red', alpha=0.7)
axs[1, 0].bar(x_indices + bar_width/2, juros_compostos, bar_width,
              label='Juros Compostos', color='green', alpha=0.7)
axs[1, 0].set_xlabel('Trimestre', fontsize=12)
axs[1, 0].set_ylabel('Valor (R$)', fontsize=12)
axs[1, 0].set_title('Composi  o dos Juros por Trimestre', fontsize=14, fontweight='bold')
axs[1, 0].legend(loc='upper left', fontsize=10)
axs[1, 0].grid(True, alpha=0.3, axis='y')
axs[1, 0].set_xticks(x_indices)
axs[1, 0].set_xticklabels(trimestres)

# Gr fico 4: Participa  o percentual dos juros compostos
axs[1, 1].plot(trimestres, participacao_compostos, 'orange', linewidth=3, marker='*')
axs[1, 1].set_xlabel('Trimestre', fontsize=12)
axs[1, 1].set_ylabel('Participa  o (%)', fontsize=12)
axs[1, 1].set_title('Participa  o dos Juros Compostos no Total de Juros', fontsize=14, fontweight='bold')
axs[1, 1].grid(True, alpha=0.3)
axs[1, 1].set_xticks(trimestres)
axs[1, 1].axhline(y=50, color='r', linestyle='--', alpha=0.5, label='Meta 50%')
axs[1, 1].legend(loc='lower right', fontsize=10)
for i, txt in enumerate(participacao_compostos):
    if i > 0:
        axs[1, 1].annotate(f'{txt:.1f}%', (trimestres[i], participacao_compostos[i]),
                          textcoords="offset points", xytext=(0,5),
                          ha='center', fontsize=8)

plt.tight_layout()
plt.subplots_adjust(top=0.92)

```

```python
# Gr fico adicional: Vis o consolidada
fig2, ax2 = plt.subplots(figsize=(14, 8))
ax2.set_xlabel('Trimestre', fontsize=14)
ax2.set_ylabel('Valor Acumulado (R$)', color='blue', fontsize=14)
line1 = ax2.plot(trimestres, valor_acumulado, 'b-', linewidth=3, marker='o',
                 label='Valor Acumulado (R$)')
ax2.tick_params(axis='y', labelcolor='blue')
ax2.set_xticks(trimestres)
ax2.grid(True, alpha=0.3)

ax2b = ax2.twinx()
ax2b.set_ylabel('Retorno Anual Equivalente (%)', color='purple', fontsize=14)
line2 = ax2b.plot(trimestres, retorno_anual, 'purple', linewidth=2.5, marker='s',
                  label='Retorno Anual Eq. (%)')
ax2b.tick_params(axis='y', labelcolor='purple')

ax2.fill_between(trimestres, 10000, valor_acumulado, alpha=0.2, color='green',
                 label='Juros Totais')
ax2.fill_between(trimestres, [10000 + js for js in juros_simples], valor_acumulado,
                 alpha=0.4, color='orange', label='Juros Compostos')

lines = line1 + line2
labels = [l.get_label() for l in lines]
ax2.legend(lines, labels, loc='upper left', fontsize=12)

plt.title('Vis o Consolidada: Valor Acumulado e Retorno com Reinvestimento Trimestral\n(VP = R$ 10.000,00, i = 5% a.t.)', fontsize=16, fontweight='bold')
plt.tight_layout()

# Salvar figuras
fig.savefig('analise_investimento_4graficos.png', dpi=300, bbox_inches='tight')
fig2.savefig('visao_consolidada_investimento.png', dpi=300, bbox_inches='tight')

print("Gr ficos gerados e salvos com sucesso!")
plt.show()
```

Listing 7: Código Python para geração dos gráficos de análise

5 Resultados e Análise

5.1 Resultados Numéricos

5.1.1 Comparação de Métodos Numéricos

Tabela 1: Comparação dos métodos numéricos para encontrar período dado $k = 20\%$

Método	**Período (trimestres)**	**Iterações**	**Eficiência Relativa**
Amostragem	12.10945	12.109.450	1x
Newton-Raphson	12.10946	50	242.189x

A Tabela 1 apresenta a comparação entre os métodos numéricos utilizados. Observa-se que ambos convergem para resultados muito próximos (12.10945 vs 12.10946 trimestres), validando a correção das implementações. Contudo, o método de Newton-Raphson demonstra superioridade significativa em eficiência computacional, sendo aproximadamente 242.000 vezes mais eficiente que o método de amostragem.

5.1.2 Período para Lucro Adicional de 20%

A análise revela que são necessários aproximadamente 12.109 trimestres (equivalente a 3.027 anos) para que o reinvestimento gere um lucro adicional de 20% sobre o capital inicial, excluindo os juros simples. Este resultado demonstra que:

- Em aproximadamente 3 anos, a estratégia de reinvestimento contribui com 20% do lucro total
- O retorno anual equivalente neste ponto é de 26.94%
- A disciplina de reinvestimento sistemático produz resultados significativos em horizonte médio

5.1.3 Período para TMA de 40% ao Ano

Para atingir uma Taxa Mínima de Atratividade de 40% ao ano, são necessários aproximadamente 26.58 trimestres (equivalente a 6.645 anos) segundo o método de Newton-Raphson. O método de amostragem encontra 27 trimestres como solução inteira. Este resultado indica que:

- A estratégia de reinvestimento permite duplicar o retorno anual efetivo (de 20% para 40%) em aproximadamente 6.5 anos
- O retorno total neste ponto atinge 138.35%

- Investimentos com horizonte de longo prazo beneficiam-se significativamente do efeito cumulativo do reinvestimento

5.1.4 Fluxo de Caixa Detalhado

Tabela 2: Fluxo de Caixa do Investimento - Análise Trimestral (Juros)

Trimestre	**Juros Simples (R$)**	**Juros Compostos (R$)**	**Total Juros (R$)**
0	0.00	0.00	0.00
1	500.00	0.00	500.00
2	1.000,00	25.00	1.025,00
3	1.500,00	76.25	1.576,25
4	2.000,00	155.06	2.155,06
5	2.500,00	262.82	2.762,82
6	3.000,00	400.96	3.400,96
7	3.500,00	571.00	4.071,00
8	4.000,00	774.55	4.774,55
9	4.500,00	1.013,28	5.513,28
10	5.000,00	1.288,95	6.288,95
11	5.500,00	1.603,39	7.103,39
12	6.000,00	1.958,56	7.958,56
13	6.500,00	2.356,49	8.856,49
14	7.000,00	2.799,32	9.799,32
15	7.500,00	3.289,28	10.789,28

Tabela 3: Fluxo de Caixa do Investimento - Análise Trimestral (Valores Acumulados)

Trimestre	Valor Acumulado (R$)	Retorno Total (%)	Retorno Anual Eq. (%)
0	10.000,00	0.00	0.00
1	10.500,00	5.00	20.00
2	11.025,00	10.25	20.50
3	11.576,25	15.76	21.02
4	12.155,06	21.55	21.55
5	12.762,82	27.63	22.10
6	13.400,96	34.01	22.67
7	14.071,00	40.71	23.26
8	14.774,55	47.75	23.87
9	15.513,28	55.13	24.50
10	16.288,95	62.89	25.16
11	17.103,39	71.03	25.83
12	17.958,56	79.59	26.53
13	18.856,49	88.56	27.25
14	19.799,32	97.99	28.00
15	20.789,28	107.89	28.77

A Tabela 3 apresenta o fluxo de caixa detalhado do investimento ao longo de 15 trimestres. Observam-se os seguintes padrões:

- Crescimento exponencial do valor acumulado, característico dos juros compostos
- Aumento progressivo da participação dos juros compostos no total de juros
- Crescimento contínuo do retorno anual equivalente

5.2 Visualização Gráfica dos Resultados

5.2.1 Evolução do Valor Acumulado e Componentes

A Figura 1 ilustra a evolução temporal do valor acumulado e seus componentes. Observa-se claramente:

- Crescimento exponencial do valor acumulado (linha azul)
- Comportamento linear dos juros simples acumulados (linha vermelha tracejada)
- Crescimento acelerado dos juros compostos acumulados (linha verde pontilhada)
- A curva de valor acumulado torna-se cada vez mais íngreme, indicando aceleração do crescimento

5.2.2 Retorno Anual Equivalente

A Figura 2 apresenta a evolução do retorno anual equivalente. Os principais insights são:

- O retorno anual inicia em 20% (igual à taxa nominal de juros simples)
- Cresce continuamente ao longo do tempo, atingindo 28.77% após 15 trimestres
- A trajetória ascendente demonstra o efeito cumulativo do reinvestimento
- Cada ponto no gráfico representa o retorno anualizado considerando todo o período até aquele trimestre

5.2.3 Composição dos Juros por Trimestre

A Figura 3 mostra a composição dos juros em cada trimestre através de um gráfico de barras. Observações relevantes:

- Nos primeiros trimestres, os juros simples dominam completamente a composição
- Progressivamente, os juros compostos aumentam sua participação relativa
- No trimestre 15, os juros compostos representam R$ 3.289,28 contra R$ 7.500,00 de juros simples
- A largura das barras permanece constante, enquanto as alturas das componentes evoluem

5.2.4 Participação Percentual dos Juros Compostos

A Figura 4 apresenta a evolução da participação percentual dos juros compostos no total de juros. Aspectos notáveis:

- Inicia em 0% (primeiro trimestre sem juros compostos)
- Cresce monotonicamente, atingindo 30.5% no trimestre 15
- A trajetória sugere crescimento contínuo, eventualmente superando 50% com horizonte mais longo
- A linha de referência em 50% marca um ponto de viragem importante

5.2.5 Visão Consolidada

A Figura 5 oferece uma visão integrada do comportamento do investimento. Características principais:

- Combina valor acumulado (eixo primário, azul) e retorno anual (eixo secundário, roxo)
- Áreas coloridas representam a decomposição entre juros simples e compostos
- Permite correlação visual direta entre crescimento do capital e evolução do retorno
- Facilita a compreensão da relação entre tempo, valor acumulado e rentabilidade

5.3 Análise Comparativa com Cenário sem Reinvestimento

Para contextualizar os resultados, compara-se o cenário com reinvestimento total com um cenário base sem reinvestimento:

Tabela 4: Comparação entre cenários com e sem reinvestimento após 15 trimestres

Parâmetro	**Sem Reinvestimento**	**Com Reinvestimento**
Valor Acumulado	R$ 17.500,00	R$ 20.789,28
Retorno Total	75.00%	107.89%
Retorno Anual Médio	20.00%	28.77%
Juros Simples	R$ 7.500,00	R$ 7.500,00
Juros Compostos	R$ 0,00	R$ 3.289,28

A Tabela 4 revela que a estratégia de reinvestimento total proporciona:

- Incremento de R$ 3.289,28 no valor acumulado (18.8% de aumento)
- Aumento de 32.89 pontos percentuais no retorno total (43.9% de aumento relativo)
- Elevação de 8.77 pontos percentuais no retorno anual médio (43.9% de aumento relativo)
- Geração de R$ 3.289,28 em juros compostos adicionais

6 Discussão

6.1 Interpretação dos Resultados

Os resultados obtidos demonstram de forma conclusiva que estratégias de reinvestimento sistemático podem transformar significativamente o perfil de retorno de investimentos com juros simples. A análise quantitativa revela:

1. **Transformação do crescimento**: O reinvestimento periódico converte o crescimento linear característico dos juros simples em crescimento exponencial típico dos juros compostos.

2. **Eficiência temporal**: São necessários aproximadamente 3 anos para que o reinvestimento gere um lucro adicional de 20% sobre o capital inicial, e aproximadamente 6.5 anos para duplicar o retorno anual efetivo.

3. **Composição dinâmica**: A participação dos juros compostos no total de juros evolui de 0% para mais de 30% em 3.75 anos, com tendência clara de crescimento contínuo.

4. **Superioridade metodológica**: O método de Newton-Raphson demonstra eficiência computacional significativamente superior ao método de amostragem para resolver as equações não-lineares do modelo.

5. **Garantia de convergência**: As verificações do teorema de convexidade confirmam que ambas as equações atendem às condições necessárias para convergência quadrática do método de Newton-Raphson.

6.2 Implicações Práticas

As descobertas deste estudo possuem várias implicações práticas para diferentes stakeholders:

6.2.1 Para Investidores Individuais

- **Importância da disciplina**: A estratégia de reinvestimento sistemático requer disciplina financeira, mas produz retornos significativos a médio e longo prazo.

- **Planejamento de horizonte**: Investidores devem considerar horizontes temporais adequados (3+ anos) para maximizar os benefícios do reinvestimento.

- **Avaliação de produtos**: Além da taxa nominal, deve-se considerar o potencial de reinvestimento na avaliação de produtos financeiros.

6.2.2 Para Instituições Financeiras

- **Design de produtos**: Produtos que facilitam o reinvestimento automático podem ser mais atrativos para clientes.

- **Ferramentas de simulação**: Oferecer ferramentas baseadas neste modelo pode auxiliar clientes no planejamento financeiro.

- **Educação financeira**: Comunicar os benefícios do reinvestimento pode melhorar a relação com clientes.

6.2.3 Para Educadores e Pesquisadores

- **Exemplo interdisciplinar**: O trabalho serve como caso de aplicação integrada de cálculo numérico e engenharia econômica.
- **Metodologia replicável**: A abordagem desenvolvida pode ser adaptada para análise de outros problemas financeiros.
- **Validação prática**: Os resultados validam a teoria de métodos numéricos em cenário real.

6.3 Limitações do Estudo

Reconhecem-se as seguintes limitações no presente trabalho:

1. **Simplificações do modelo**:
   - Assume reinvestimento imediato e total, sem considerar períodos de carência
   - Não inclui custos de transação ou impostos sobre rendimentos
   - Considera taxa de juros constante ao longo do tempo
2. **Escopo da análise**:
   - Foca em um cenário específico de parâmetros
   - Não explora completamente a sensibilidade a variações paramétricas
   - Limita-se a estratégia de reinvestimento total
3. **Aspectos metodológicos**:
   - Compara apenas dois métodos numéricos
   - Não inclui análise estatística rigorosa dos resultados
   - Baseia-se em simulação teórica sem dados empíricos

6.4 Sugestões para Trabalhos Futuros

Com base nas limitações identificadas, sugerem-se as seguintes direções para pesquisas futuras:

1. **Extensões do modelo**:

- Incorporação de efeitos fiscais (imposto de renda sobre rendimentos)
- Inclusão de inflação para análise de retornos reais
- Consideração de reinvestimento parcial (frações dos rendimentos)
- Adaptação para diferentes frequências de reinvestimento

2. **Avanços metodológicos**:
   - Implementação de métodos numéricos adicionais (bisseção, secante, etc.)
   - Desenvolvimento de análise de sensibilidade paramétrica
   - Criação de interface gráfica interativa para exploração de cenários
   - Aplicação de técnicas de otimização para estratégias de reinvestimento

3. **Aplicações práticas**:
   - Análise de produtos financeiros reais do mercado
   - Desenvolvimento de ferramentas de decisão para investidores
   - Integração com planejamento financeiro pessoal
   - Estudo comparativo entre diferentes estratégias de reinvestimento

7 Conclusões

7.1 Conclusões Principais

Este trabalho desenvolveu e aplicou uma abordagem interdisciplinar combinando conceitos de cálculo numérico e engenharia econômica para análise de estratégias de reinvestimento em produtos financeiros com juros simples. As principais conclusões são:

1. **Validação do modelo matemático**: Confirmou-se que o reinvestimento periódico transforma o crescimento de investimentos com juros simples em comportamento equivalente a juros compostos, conforme descrito pela equação $VF_n = VP(1+i)^n$.

2. **Eficácia da estratégia de reinvestimento**: Para os parâmetros analisados (VP = R$ 10.000, i = 5% a.t.), a estratégia de reinvestimento total:
   - Gera lucro adicional de 20% exclusivamente do reinvestimento em aproximadamente 3 anos (12.109 trimestres)
   - Eleva o retorno anual efetivo de 20% para 28.77% em 3.75 anos
   - Permite atingir TMA de 40% ao ano em aproximadamente 6.5 anos (26.58 trimestres)

3. **Superioridade do método Newton-Raphson**: O método de Newton-Raphson demonstrou eficiência computacional aproximadamente 242.000 vezes superior ao método de amostragem para resolver as equações não-lineares do modelo.

4. **Garantia de convergência**: As verificações do teorema de convexidade confirmaram que ambas as equações do modelo atendem às condições necessárias para convergência quadrática do método de Newton-Raphson.

5. **Importância do horizonte temporal**: Os benefícios do reinvestimento são cumulativos e se aceleram com o tempo, com a participação dos juros compostos evoluindo de 0% para mais de 30% em 3.75 anos.

6. **Valor agregado quantificável**: Em comparação com estratégia sem reinvestimento, o reinvestimento total proporcionou aumento de 32.89% no retorno total e 43.9% no retorno anual médio após 3.75 anos.

7.2 Contribuições do Trabalho

Este estudo contribui para o conhecimento nas seguintes áreas:

- **Integração interdisciplinar**: Demonstra aplicação prática da integração entre cálculo numérico e engenharia econômica na solução de problemas financeiros reais.
- **Metodologia analítica**: Desenvolve abordagem completa desde modelagem matemática até implementação computacional e visualização de resultados.
- **Ferramenta de análise**: Oferece ferramenta computacional replicável para análise de estratégias de reinvestimento.
- **Insights práticos**: Fornece insights quantitativos sobre os benefícios de estratégias de reinvestimento para diferentes horizontes temporais.
- **Validação de convergência**: Apresenta verificação formal das condições de convergência do método de Newton-Raphson para equações financeiras não-lineares.

7.3 Recomendações Finais

Com base nos resultados obtidos, formulam-se as seguintes recomendações:

1. **Para investidores**:
   - Adotem disciplina de reinvestimento sistemático de rendimentos
   - Considerem horizontes temporais adequados (médio/longo prazo) para maximizar benefícios

- Utilizem ferramentas de simulação para planejamento financeiro

2. **Para instituições financeiras**:
   - Desenvolvam produtos que facilitem o reinvestimento automático
   - Ofereçam ferramentas educacionais baseadas em simulações
   - Comuniquem claramente os benefícios de estratégias de reinvestimento

3. **Para educadores**:
   - Utilizem este caso como exemplo de aplicação interdisciplinar
   - Enfatizem a importância de métodos numéricos na solução de problemas práticos
   - Integrem conceitos teóricos com aplicações do mundo real

4. **Para pesquisadores**:
   - Explorem as extensões sugeridas para modelos mais realistas
   - Desenvolvam ferramentas computacionais avançadas para análise financeira
   - Realizem estudos empíricos comparativos com dados do mercado
   - Investiguem condições de convergência para outros tipos de equações financeiras

7.4 Considerações Finais

Este trabalho demonstrou que a aplicação integrada de conhecimentos de cálculo numérico e engenharia econômica permite a análise quantitativa rigorosa de estratégias financeiras complexas. A abordagem desenvolvida - desde a modelagem matemática até a implementação computacional e visualização de resultados - fornece um framework analítico poderoso para tomada de decisão de investimento.

Os resultados obtidos validam a importância da disciplina financeira e do planejamento de longo prazo, destacando o valor agregado por estratégias sistemáticas de reinvestimento. Além disso, o trabalho contribui para a educação financeira ao fornecer ferramentas e insights que facilitam a compreensão de conceitos financeiros complexos.

A verificação formal das condições de convergência do método de Newton-Raphson adiciona rigor matemático à análise, garantindo a confiabilidade dos resultados numéricos obtidos.

Em um contexto de crescente complexidade dos mercados financeiros e necessidade de educação financeira, trabalhos como este desempenham papel importante na ponte entre teoria acadêmica e prática profissional, capacitando indivíduos e organizações para decisões financeiras mais informadas e fundamentadas.

Referências Bibliográficas

Referências

[1] ASSAF NETO, Alexandre. **Matemática Financeira e suas Aplicações**. São Paulo: Atlas, 2012.

[2] FRANCO, Neide Bertoldi. **Cálculo Numérico**. São Paulo: Pearson Prentice Hall, 2006.

Agradecimentos

Agradeço primeiramente a minha mulher Letícia por tornar esse projeto possível e aos professores das disciplinas de Cálculo Numérico e Engenharia Econômica pela orientação e pelos conhecimentos transmitidos durante o quadrimestre. Agradeço também à Universidade Federal do ABC pela infraestrutura e oportunidades de aprendizado.

A Código Fonte Completo

A.1 Código Principal

O código fonte completo está disponível em repositório online e pode ser acessado através do link: `https://github.com/arthurht122/Projeto-Engenharia-Economica-e-Calculo-`

A.2 Dependências e Requisitos

Para execução do código, são necessárias as seguintes dependências:

- Python 3.8 ou superior
- NumPy 1.20 ou superior
- Matplotlib 3.4 ou superior
- Jupyter Notebook (opcional, para execução interativa)

B Dados da Simulação

B.1 Parâmetros Utilizados

- Valor Presente (VP): R$ 10.000,00
- Taxa de juros trimestral (i): 5% (0,05)
- Período de análise (n): 15 trimestres
- TMA anual (TMA): 40% (0,40)
- Lucro desejado (k): 20% (0,20)

B.2 Resultados Detalhados

Os resultados completos da simulação, incluindo dados intermediários e arquivos de saída, estão disponíveis no repositório online do projeto.