

Atividade 2

Fórmula de Black-Scholes: modelo usado na precificação de derivativos.

$$d_1 = \frac{\ln(S/K) + (r - q + \sigma^2/2)\tau}{\sigma\sqrt{\tau}}$$

$$d_2 = d_1 - \sigma\sqrt{\tau}$$

S : preço da ação

K : strike price (preço de exercício)

r : taxa livre de risco

q : dividend yield anual

$\tau = T - t$: tempo para o vencimento

σ : volatilidade

Utilizando a Regra 1/3 de Simpson

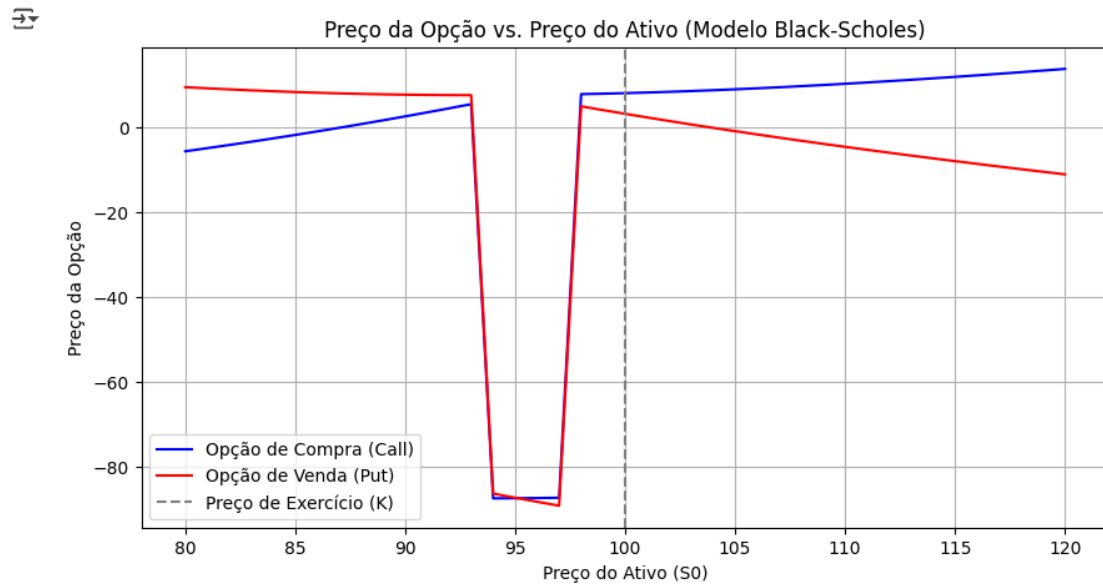
```

1 import numpy as np
2 import pandas as pd
3 import scipy.stats as stats
4 import matplotlib.pyplot as plt
5
6 def normal_cdf_simpson(x, n=1000):
7     if x < 0:
8         return 1 - normal_cdf_simpson(-x, n)
9
10    f = lambda t: np.exp(-t**2 / 2) / np.sqrt(2 * np.pi)
11    a, b = 0, x
12    h = (b - a) / n
13    integral = f(a) + f(b)
14
15    for i in range(1, n, 2):
16        integral += 4 * f(a + i * h)
17    for i in range(2, n-1, 2):
18        integral += 2 * f(a + i * h)
19
20    return (h / 3) * integral
21
22 def black_scholes(S0, K, T, r, sigma, option_type="call"):
23     # Preço de uma opção
24     d1 = (np.log(S0 / K) + (r + 0.5 * sigma**2) * T) / (sigma * np.sqrt(T))
25     d2 = d1 - sigma * np.sqrt(T)
26
27     N_d1 = normal_cdf_simpson(d1)
28     N_d2 = normal_cdf_simpson(d2)
29
30     if option_type == "call":
31         return S0 * N_d1 - K * np.exp(-r * T) * N_d2
32     elif option_type == "put":
33         return K * np.exp(-r * T) * normal_cdf_simpson(-d2) - S0 * normal_cdf_simpson(-d1)
34     else:
35         raise ValueError("O tipo da opção deve ser 'call' ou 'put'.")
36
37 # Parâmetros:
38 K = 100
39 T = 1 # 1 ano
40 r = 0.05 # 5% aa
41 sigma = 0.2 # 20% aa
42
43 # Gerar preços para diferentes valores de S0:
44 S0_values = np.arange(80, 121, 1) # De 80 a 120 em passos de 1
45 call_prices = [black_scholes(S0, K, T, r, sigma, "call") for S0 in S0_values]
46 put_prices = [black_scholes(S0, K, T, r, sigma, "put") for S0 in S0_values]
47
48 # Geração da planilha Excel:
49 df = pd.DataFrame({"S0": S0_values, "Call Price": call_prices, "Put Price": put_prices})
50 df.to_excel("black_scholes.xlsx", index=False)

```

Gráfico:

```
52 # Gráficos:
53 plt.figure(figsize=(10, 5))
54 plt.plot(S0_values, call_prices, label="Opção de Compra (Call)", color="blue")
55 plt.plot(S0_values, put_prices, label="Opção de Venda (Put)", color="red")
56 plt.axvline(K, color="gray", linestyle="--", label="Preço de Exercício (K)")
57 plt.xlabel("Preço do Ativo (S0)")
58 plt.ylabel("Preço da Opção")
59 plt.title("Preço da Opção vs. Preço do Ativo (Modelo Black-Scholes)")
60 plt.legend()
61 plt.grid()
62 plt.show()
63
64 print("Planilha 'black_scholes.xlsx' gerada com sucesso!")
```



Planilha 'black_scholes.xlsx' gerada com sucesso!