

Atividade 1

Sistema de Lorenz

- previsões meteorológicas
- sistema de três equações diferenciais de primeira ordem:

$$\frac{dx}{dt} = \sigma(y - x)$$

$$\frac{dy}{dt} = x(\rho - z) - y$$

$$\frac{dz}{dt} = xy - \beta z$$

- intensidade de movimento do fluido atmosférico: x
- variação de temperatura na direção horizontal: y
- variação de temperatura na direção vertical: z

parâmetros:

$$\sigma = 10$$

$$\beta = 8/3$$

$$\rho = 28$$

condições iniciais:

$$x = y = z = 5$$

$$t = 0 \text{ a } 20$$

Método utilizado: Runge Kutta de quarta ordem de dupla precisão:

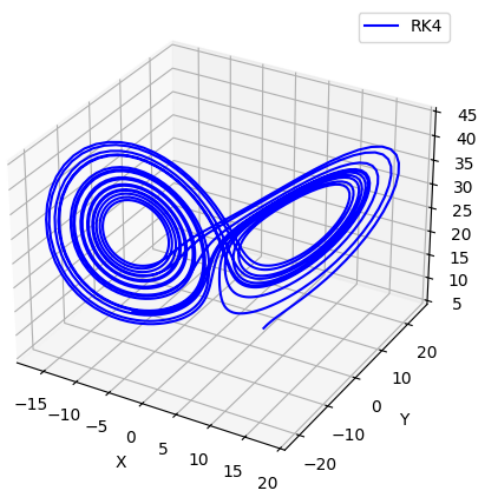
```
1 # Importando os pacotes:
2 import numpy as np
3 import pandas as pd
4 import matplotlib.pyplot as plt
5
6 # Parâmetros do Sistema de Lorenz:
7 sigma = 10
8 rho = 28
9 beta = 8 / 3
10
11 def lorenz(x, y, z):
12     dx = sigma * (y - x)
13     dy = x * (rho - z) - y
14     dz = x * y - beta * z
15     return np.array([dx, dy, dz])
16
17 # Método Runge Kutta:
18 def runge_kutta4(f, y0, t0, tf, h):
19     t = np.arange(t0, tf, h)
20     y = np.zeros((len(t), len(y0)))
21     y[0] = y0
22     for i in range(len(t) - 1):
23         k1 = f(*y[i])
24         k2 = f(*y[i] + (h / 2) * k1)
25         k3 = f(*y[i] + (h / 2) * k2)
26         k4 = f(*y[i] + h * k3)
27         y[i+1] = y[i] + (h / 6) * (k1 + 2 * k2 + 2 * k3 + k4)
28     return t, y
29
30 # Configuração inicial e parâmetros
31 y0 = np.array([5.0, 5.0, 5.0])
32 t0, tf, h = 0, 20, 0.01 # Intervalo de tempo e passo
33
34 t_rk4, y_rk4 = runge_kutta4(lorenz, y0, t0, tf, h)
35
36 df = pd.DataFrame({
37     "Tempo": t_rk4,
38     "RK4 X": y_rk4[:, 0],
39     "RK4 Y": y_rk4[:, 1],
40     "RK4 Z": y_rk4[:, 2],
41 })
42
43 # Salva Resultados no Excel:
44 df.to_excel("lorenz_results.xlsx", index=False)
45
46 print("Resultados salvos em 'lorenz_results.xlsx'")
```

Gráficos:

```
50 # Gráficos:
51 fig = plt.figure(figsize=(10, 5))
52
53 # Gráfico 3D
54 ax = fig.add_subplot(121, projection='3d')
55 ax.plot(y_rk4[:, 0], y_rk4[:, 1], y_rk4[:, 2], label="RK4", color="blue")
56 ax.set_title("Atrator de Lorenz (RK4)")
57 ax.set_xlabel("X")
58 ax.set_ylabel("Y")
59 ax.set_zlabel("Z")
60 ax.legend()
61
62 # Gráfico 2D X vs Z
63 ax2 = fig.add_subplot(122)
64 ax2.plot(t_rk4, y_rk4[:, 0], label="X (RK4)")
65 ax2.plot(t_rk4, y_rk4[:, 1], label="Y (RK4)")
66 ax2.plot(t_rk4, y_rk4[:, 2], label="Z (RK4)")
67 ax2.set_title("Evolução Temporal das Variáveis (RK4)")
68 ax2.set_xlabel("Tempo")
69 ax2.set_ylabel("Valores")
70 ax2.legend()
71
72 plt.tight_layout()
73 plt.show()
```

Resultados salvos em 'lorenz_results.xlsx'

Atrator de Lorenz (RK4)



Evolução Temporal das Variáveis (RK4)

