

Curso de Cálculo Numérico

Exemplos de Equações Diferenciais Ordinárias (EDOs) em Ciência da Computação

Professor: Paulo César Linhares da Silva

Introdução

As equações diferenciais ordinárias de primeira ordem (EDOs de primeira ordem) são amplamente utilizadas em ciências da computação para modelar diversos fenômenos.

Nos slides a seguir temos alguns exemplos de problemas que envolvem EDOs de primeira ordem aplicados à área:

Carga de Capacitores em Circuitos

Contexto: Modelar a carga de um capacitor em um circuito elétrico, como em um computador ou dispositivo eletrônico.

Equação:

$$\frac{dQ}{dt} = \frac{V - Q/C}{R}$$

Onde:

- $Q(t)$ é a carga no capacitor no tempo t ,
- V é a voltagem aplicada,
- R é a resistência,
- C é a capacitância.

Aplicação: Pode ser usado para simular o comportamento de circuitos em sistemas embarcados.

Modelos de Propagação de Vírus em Redes

- **Contexto:** Modelar a propagação de vírus ou malware em uma rede de computadores.
- **Equação:**

$$\frac{dI}{dt} = \beta SI - \gamma I$$

Onde:

$I(t)$ é o número de computadores infectados no tempo t

$S(t)$ é o número de computadores suscetíveis a infecção

β é a taxa de infecção,

γ é a taxa de recuperação.

Aplicação: Pode ser usado para prever a disseminação de malware e planejar estratégias de contenção.

Algoritmos de Aprendizado de Máquina (Gradiente Descendente)

- **Contexto:** Modelar a dinâmica de otimização em algoritmos de aprendizado de máquina.
- **Equação:**

$$\frac{d\theta}{dt} = -\nabla J(\theta)$$

Onde:

$\theta(t)$ são os parâmetros do modelo no tempo t ,

$J(\theta)$ é a função custo.

Aplicação: Pode ser usado para analisar a convergência de algoritmos de otimização.

Simulação de Filas (Teoria das Filas)

- **Contexto:** Modelar o comportamento de filas em sistemas de computação, como filas de processamento ou requisições em servidores.
- **Equação:**

$$\frac{dN}{dt} = \lambda - \mu N$$

Onde:

- $N(t)$ é o número de tarefas na fila no tempo t ,
- λ é a taxa de chegada de tarefas,
- μ é a taxa de processamento.

Aplicação: Pode ser usado para otimizar o desempenho de sistemas de servidores ou balanceamento de carga.

Dinâmica de Memória Cache

- **Contexto:** Modelar a taxa de acertos e falhas em uma memória cache.
- **Equação:**

$$\frac{dH}{dt} = \alpha(1 - H) - \beta H$$

Onde:

$H(t)$ é a taxa de acertos no tempo t ,

α é a taxa de acertos potenciais,

β é a taxa de falhas.

Aplicação: Pode ser usado para otimizar políticas de substituição de cache.

Calcular a área de objetos em uma imagem ou a intensidade média de pixels.

- Aqui está um exemplo usa a regra dos trapézios repetida para calcular a intensidade média de pixels em uma imagem.
- A ideia é tratar a imagem como uma função bidimensional $f(x,y)$, onde $f(x,y)$ representa a intensidade do pixel na posição (x,y) .
- A intensidade média pode ser calculada como a integral da função de intensidade dividida pela área da imagem.

Explicação do código

Regra dos Trapézios Repetida em 2D:

- A função `trapezoidal_rule_2d` calcula a integral de uma função bidimensional usando a regra dos trapézios repetida.
- A função `np.trapz` é usada para aplicar a regra dos trapézios ao longo de cada eixo.

Carregamento da Imagem:

- A imagem é carregada e convertida para escala de cinza usando a biblioteca PIL (Pillow).
- A intensidade de cada pixel é representada por um valor entre 0 (preto) e 255 (branco).

Função de Intensidade:

- A função `intensity_function` retorna a intensidade do pixel na posição (x,y) usando interpolação bilinear.

Visualização

- A imagem é exibida usando matplotlib, com uma barra de cores indicando a intensidade dos pixels.

Exemplo de Saída

- **Intensidade total:** Soma das intensidades de todos os pixels.
- **Área da imagem:** Número total de pixels.
- **Intensidade média:** Média das intensidades dos pixels.

Observações

- A regra dos trapézios é adequada para imagens porque aproxima a integral de forma simples e eficiente.
- Para imagens grandes, o número de subintervalos pode ser ajustado para equilibrar precisão e desempenho.
- Este método pode ser estendido para imagens coloridas, calculando a intensidade média para cada canal (R, G, B).