第三次上机报告

> 2233316027 王博想

1. 矩阵计算

问题重述

给定矩阵：

$$A = \begin{bmatrix} 3 & 0 & 4 & 0 \\ 2 & 2 & 2 & 2 \\ 0 & -7 & 0 & 0 \\ 5 & 3 & -2 & 2 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 10 \\ 20 \\ 30 \\ 40 \end{bmatrix}, \quad b_2 = \begin{bmatrix} 1 \\ 3 \\ 5 \\ 7 \end{bmatrix}, \quad \varepsilon = 10^{-10}$$

完成以下计算：

1. 求各阶顺序主子式的值
2. 使用 `linalg` 求解 $Ax = b_1$
3. 使用 Crame 法则解 $Ax = b_2$，并验证 $\|Ax - b_2\|_2 < \varepsilon$

实现思想

1. 主子式通过提取前 k 行列的子矩阵并计算行列式得到；
2. 方程组 $Ax = b_1$ 使用 NumPy 的 `linalg.solve`；
3. 使用克拉默法则计算 $x_i = \dfrac{\det(A_i)}{\det(A)}$，其中 A_i 为用 b_2 替换 A 的第 i 列的矩阵；验证误差是否小于 ε。

源代码

```
import numpy as np

A = np.array([[3, 0, 4, 0], [2, 2, 2, 2], [0, -7, 0, 0], [5, 3, -2,
2]], dtype=float)
b1 = np.array([10, 20, 30, 40], dtype=float)
b2 = np.array([1, 3, 5, 7], dtype=float)
eps = 1e-10
```

```
minors = [np.linalg.det(A[:k, :k]) for k in range(1, 5)]
print("Leading principal minors:", minors)

x1 = np.linalg.solve(A, b1)
print("Solution x1:", x1)

det_A = np.linalg.det(A)
x2 = np.zeros(4)
for i in range(4):
    Ai = A.copy()
    Ai[:, i] = b2
    x2[i] = np.linalg.det(Ai) / det_A
print("Solution x2 (Cramer):", x2)

error = np.linalg.norm(np.dot(A, x2) - b2)
print("Error:", error < eps)
```

结果及说明

```
Leading principal minors: [3.0000000000000004, 6.0,
-13.999999999999996, 336.0]
Solution x1: [ 5.71428571 -4.28571429 -1.78571429 10.35714286]
Solution x2 (Cramer): [ 0.95238095 -0.71428571 -0.46428571
1.72619048]
Error: True
```

说明：主子式依次为 $3, 6, -14, 336$；$Ax = b_1$ 解为 x_1，$Ax = b_2$ 使用克拉默法求得 x_2 并通过误差验证。

2. 线性方程组求解

问题重述

从 `function.txt` 文件中读取一个四阶线性方程组并用高斯消元法求解。

实现思想

1. 读取文件，跳过注释行，提取 n、系数矩阵 A 和常数向量 b；
2. 实现高斯消元法，进行消元和回代，求得唯一解。

源代码

```python
import numpy as np

def read_equation(file_path):
    with open(file_path, 'r', encoding='utf-8') as f:
        lines = [line.strip() for line in f if line.strip() and not line.startswith('#')]
        n = int(lines[0])
        A = np.array([list(map(float, line.split(','))) for line in lines[1:n+1]])
        b = np.array(list(map(float, lines[n+1].split(','))))
        return A, b

def gaussian_elimination(A, b):
    n = len(b)
    for i in range(n):
        max_row = i + np.argmax(np.abs(A[i:, i]))
        A[[i, max_row]] = A[[max_row, i]]
        b[i], b[max_row] = b[max_row], b[i]
        for j in range(i+1, n):
            factor = A[j][i] / A[i][i]
            A[j, i:] -= factor * A[i, i:]
            b[j] -= factor * b[i]
    x = np.zeros(n)
    for i in range(n-1, -1, -1):
        x[i] = (b[i] - np.dot(A[i, i+1:], x[i+1:])) / A[i][i]
    return x

A, b = read_equation('function.txt')
x = gaussian_elimination(A, b)
print("Solution:", x)
```

结果及说明

```
Solution: [ 4.18518519 -1.07407407 -0.25925926  0.55555556]
```

方程组解为 $x = [4.18518519, -1.07407407, -0.25925926, 0.55555556]$

3. 函数极值与绘图

问题重述

函数 $f(x) = x + \sin(x + \cos x)$，求其在区间 $[0, 4\pi]$ 上的所有极值点（保留 4 位小数）及对应极值，绘图并标出极值点。

实现思想

1. 使用 SymPy 进行符号求导，构造导数函数 $f'(x)$；
2. 在 $[0, 4\pi]$ 区间内寻找导数为零的点（数值解法）；
3. 判断这些点是否为极值点，并绘图展示。

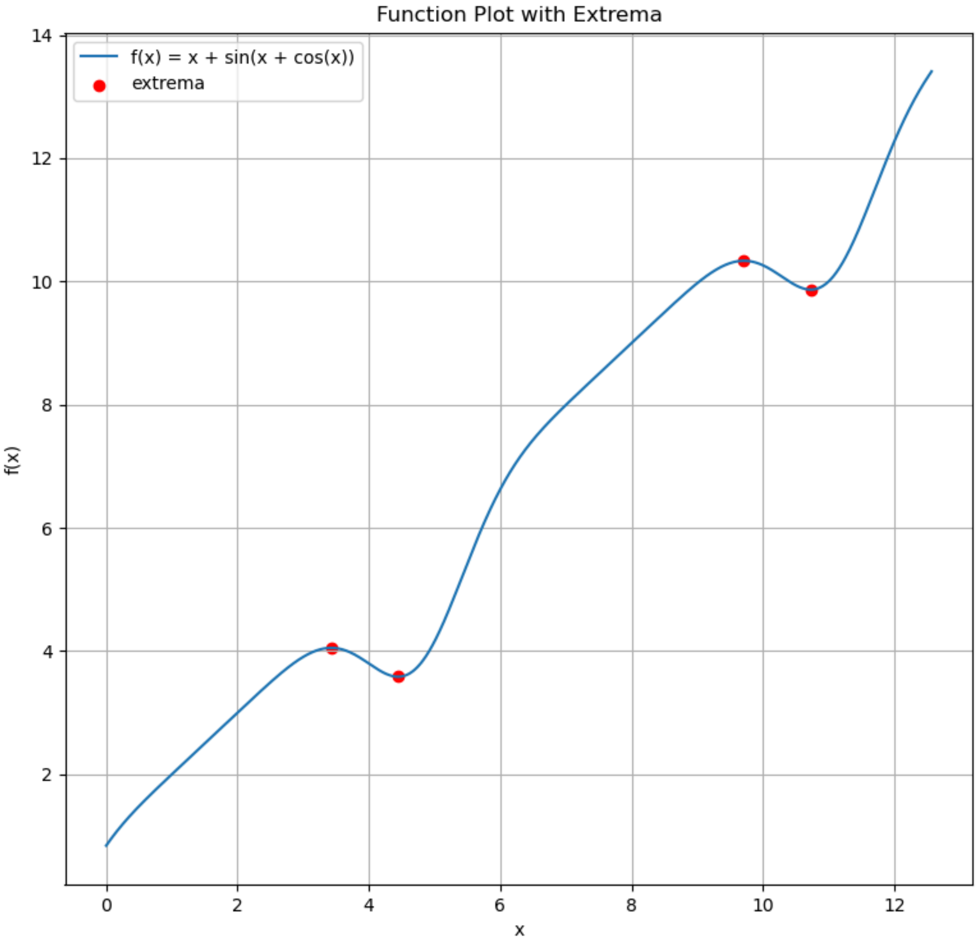
源代码

```
import numpy as np
import matplotlib.pyplot as plt
import sympy as sp
from scipy.optimize import fsolve

x = sp.symbols('x')
f = x + sp.sin(x + sp.cos(x))
f_prime = sp.diff(f, x)

f_prime_func = sp.lambdify(x, f_prime, 'numpy')
f_func = sp.lambdify(x, f, 'numpy')

initial_guesses = np.linspace(0, 4*np.pi, 50)
crit_points = []
for guess in initial_guesses:
    root, = fsolve(f_prime_func, guess)
    if 0 <= root <= 4*np.pi:
        if not any(abs(root - cp) < 1e-4 for cp in crit_points):
            crit_points.append(root)

extrema_x = np.array(sorted(crit_points))
extrema_y = f_func(extrema_x)

x_vals = np.linspace(0, 4*np.pi, 1000)
y_vals = f_func(x_vals)

plt.plot(x_vals, y_vals, label='f(x) = x + sin(x + cos(x))')
plt.scatter(extrema_x, extrema_y, color='red', label='extrema')
plt.title('Function Plot with Extrema')
plt.xlabel('x')
plt.ylabel('f(x)')
plt.legend()
plt.grid(True)
plt.show()

for x0, y0 in zip(extrema_x, extrema_y):
    print(f"x = {x0:.4f}, f(x) = {y0:.4f}")
```

结果及说明

```
x = 3.4263, f(x) = 4.0512
x = 4.4437, f(x) = 3.5830
x = 9.7095, f(x) = 10.3344
x = 10.7269, f(x) = 9.8662
```

极值点已标出并在图像中以红点显示，计算值保留四位小数。


Function Plot with Extrema
f(x) = x + sin(x + cos(x))
extrema
f(x)
x
14
12
10
8
6
4
2
0
2
4
6
8
10
12