

from scipy import optimize as opt

import numpy as np

from scipy.optimize import minimize

# 目标函数

def objective(x):

return x[0] \*\* 2 + x[1] \*\* 2 + x[2] \*\* 2 + 8

# 约束条件

def constraint1(x):

return x[0] \*\* 2 - x[1] + x[2] \*\* 2 # 不等约束

def constraint2(x):

return -(x[0] + x[1] \*\* 2 + x[2] \*\* 2 - 20) # 不等约束

def constraint3(x):

return -x[0] - x[1] \*\* 2 + 2

def constraint4(x):

return x[1] + 2 \* x[2] \*\* 2 - 3 # 不等约束

# 边界约束

b = (0.0, None)

bnds = (b, b, b)

con1 = {'type': 'ineq', 'fun': constraint1}

con2 = {'type': 'ineq', 'fun': constraint2}

con3 = {'type': 'eq', 'fun': constraint3}

con4 = {'type': 'eq', 'fun': constraint4}

cons = ([con1, con2, con3, con4]) # 4个约束条件

x0 = np.array([0, 0, 0])

# 计算

solution = minimize(objective, x0, method='SLSQP', bounds=bnds, constraints=cons)

x = solution.x

print('目标值: ' + str(objective(x)))

print('答案为')

print('x1 = ' + str(x[0]))

print('x2 = ' + str(x[1]))

# ----------------------------------

# 输出：

# 目标值: 10.651091840572583

# 答案为

# x1 = 0.5521673412903173

# x2 = 1.203259181851855