



…………………………………………………..

import numpy as np

import matplotlib.pyplot as plt

import scipy as sp

from scipy.optimize import minimize

f = lambda x: (x[0] - 1)\*\*2 + (x[1] - 2.5)\*\*2 #here function is x,y dependent so it as vector(lambda),

#x[0]=first argunment of function that is x,

#x[1]=second argunment of function that is y,

cons = ({'type': 'ineq', 'fun': lambda x: x[0] - 2 \* x[1] + 2}, #constrains as tuple form

{'type': 'ineq', 'fun': lambda x: -x[0] - 2 \* x[1] + 6},

{'type': 'ineq', 'fun': lambda x: -x[0] + 2 \* x[1] + 2})

bnds = ((0, None), (0, None)) #boundary condition x from zero to infinity ,similarly for y.

res = minimize(f, (2, 0), bounds=bnds, constraints=cons) #(2,0) initial guess

print("function minimize at x,y=", res.x[0],res.x[1])

res.x

…………………………………………………….