

数学建模作业一报告

原创 私密 lzrrrrr 2020-07-09 10:46:33 1 收藏

编辑 版权

文章目录

数学建模作业一报告

三维网格图绘制

等高线与地标绘制

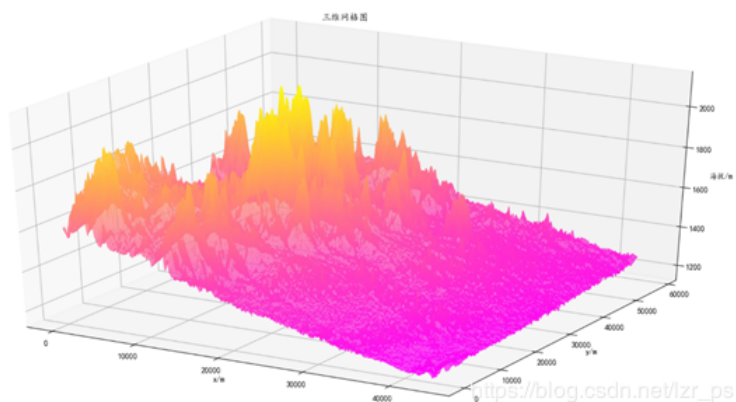
求区域表面积近似值

附录

数学建模作业一报告

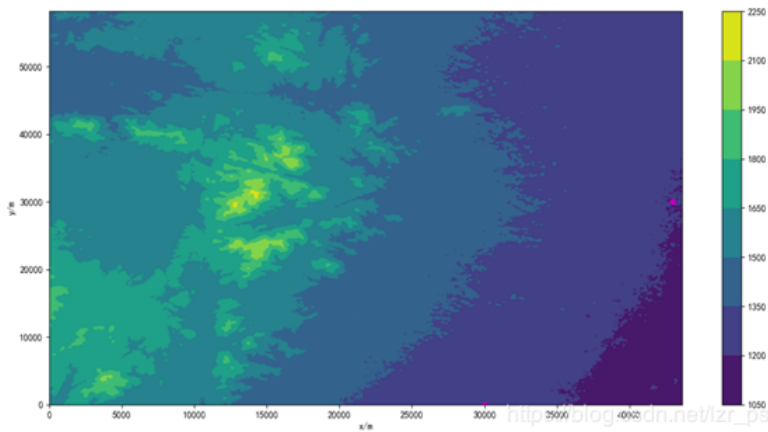
三维网格图绘制

- a.用pandas读取excel文件并将数据保存在数组中
 - b.`ax3d=plt.axes(projection= '3d')`设置成3维模式
 - c.用`ax3d.plot_surface`绘制三维网格图
- 结果：



等高线与地标绘制

- a.用pandas读取excel文件并将数据保存在数组中
 - b.用`plt.contourf`绘制投影
 - c.用`plt.plot`标记点A,B的位置
- 结果：



求区域表面积近似值

- 将每个网格上区域的表面积近似为海拔高度构成的两个三角形面积之和
- 用勾股定理求得三角形边长，然后用海伦定理求解面积
- 汇总所有网格得表面积得到总面积

结果：

总面积近似为：2575226615.1594605

附录

代码：

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

region_data_path = "./data.xlsx" # 文件路径
data = pd.read_excel(region_data_path, header=None)
data = data.values
plt.rcParams['font.sans-serif'] = ['KaiTi'] # 指定默认字体
plt.rcParams['axes.unicode_minus'] = False # 解决保存图像是负号 '-' 显示为方块的问题

# 画3D网格图
def Grid_3D():
    fig = plt.figure() # 创建一张图片
    ax3d=plt.axes(projection='3d')
    x = np.arange(data.shape[0])
    y = np.arange(data.shape[1])
    x, y = np.meshgrid(x*50, y*50)
    ax3d.set_title('三维网格图')
    ax3d.plot_surface(x, y, data.transpose([1,0]), rstride=1, cstride=1, cmap=plt.cm.spring) # 3D网格图
    ax3d.set_xlabel('x/m')
    ax3d.set_ylabel('y/m')
    ax3d.set_zlabel('海拔/m')
    plt.show()

# 画等高线图
def Contourline(A,B):
    x1 = A[0]*1000
    x2 = B[0]*1000
```

```

y1 = A[1]*1000
y2 = B[1]*1000
x = np.arange(data.shape[0])
y = np.arange(data.shape[1])
x, y = np.meshgrid(x * 50, y * 50)
cset = plt.contourf(x,y,data.transpose([1,0])) # 等高线绘画
plt.colorbar(cset) # 设置颜色条
plt.plot(x1, y1, 'om') # A点标注
plt.plot(x2, y2, 'om') # B点标注
plt.xlabel('x/m')
plt.ylabel('y/m')
plt.show()

# 求区域表面积
def compute_area():
    a = np.arange(data.shape[0])
    b = np.arange(data.shape[1])
    z = data.transpose([1,0])
    area = 0
    for i in b[:-1]:
        for j in a[:-1]:
            h1 = z[i][j]
            h2 = z[i+1][j]
            h3 = z[i][j+1]
            h4 = z[i+1][j+1] # 每个网格的四个海拔高度
            l1 = (abs(h1-h3)**2+50**2)**0.5
            l2 = (abs(h1-h2)**2+50**2)**0.5
            l3 = (abs(h3-h4)**2+50**2)**0.5
            l4 = (abs(h2-h4)**2+50**2)**0.5 # 近似四边形的四条边长度
            l5 = (abs(h1-h4)**2+2*(50**2))**0.5 # 对角线长度
            # 用海伦公式求两个三角形面积
            p1 = (l1+l3+l5)/2
            p2 = (l2+l4+l5)/2
            tri_area1 = (p1*(p1-l1)*(p1-l3)*(p1-l5))**0.5
            tri_area2 = (p2*(p2-l2)*(p2-l4)*(p2-l5))**0.5
            zarea = tri_area1+tri_area2
            area+=zarea # 汇总面积
    print('总面积近似为: '+str(area))

if __name__ == '__main__':
    A = [30, 0]
    B = [43, 30]
    Grid_3D()
    Contourline(A,B)
    compute_area()

```