气体污染源定位问题

摘要

空气污染不仅对人类健康有害，也对生态系统和环境造成损害，影响植物生长、水质和土壤质量，破坏生态平衡。有效的气体污染治理有助于保护环境和生物多样性。为防治气体污染，相关部门在边长为 100 米正方形区域每隔 10 米部署一个气体采集传感器记录污染气体浓度，建立数学模型利用采集数据确定污染源的位置。首先利用插值分析方法对数据进行插值求解，扩大数据量，然后利用高斯扩散模型对污染源的具体位置或是数目求解。问题一是单一污染源问题，得到污染源的估计位置在 $(9.96, 9.97) \pm 0.05$ 米。问题二是多个污染源问题，得到污染源有两个，分别估计位置在 $(10.00, 10.00) \pm 0.05$ 米和 $(40.00, 30.00) \pm 0.09$ 米。

关键字：　污染源　空间插值技术　高斯扩散模型

一、 问题重述与分析

1.1 问题背景

空气污染不仅对人类健康有害，也对生态系统和环境造成损害，影响植物生长、水质和土壤质量，破坏生态平衡。有效的气体污染治理有助于保护环境和生物多样性。为防治气体污染，相关部门在边长为 100 米正方形区域每隔 10 米部署一个气体采集传感器记录污染气体浓度，建立数学模型利用采集数据确定污染源的位置。

1.2 数据集和要求

附件 Sheet1 [1] 提供了正方形区域内横坐标以及纵坐标为 0m，10m，20m，30m，40m，50m，60m，70m，80m，90m，100m 的某污染源气体浓度采集数据（单位：mg），形成 11×11 网格；附件 Sheet2 提供了该区域内未知污染源个数与位置的气体采集数据（单位：mg）。题目要求建立数学模型讨论以下问题：

1. 确定这一个污染源的具体位置，预测每隔 1 米的浓度数据（共 101×101 个）并填入 Sheet3；
2. 确定多个污染源的具体数量及其相应的位置，预测每隔 1 米的浓度数据并填入附件 Sheet4；
3. 对以上进行误差分析。

1.3 问题分析

污染源的物理模型可以联想到气体扩散模型，其遵循正态分布进行扩散，根据数据集进行拟合，可以求特定位置。对于单一污染源，用一个高斯函数便可；对于多污染源，需多个高斯函数相加，再根据猜测污染源数量，需要考虑高斯函数的数目。

由于样本较少，在预测污染源位置之前，可以按照题目要求先对每隔 1 米的浓度数据进行估计。已知空间离散的点 (x, y)，且不考虑 z，可采用空间插值技术，Mathematica 编程软件中有 `SpatialEstimate` 函数，可以直接进行使用。

二、 模型假设

a. 假定数据集为某一时间点（包括昼夜、季节）的各个位置的气体浓度，即不考虑时间变量；

[1]表格“H4”的数据类型错误，因为数字过小（4^{-311}），直接做零处理

b. 假定研究区域地形平坦，铅直方向（z）气体浓度均一，即只考虑二维平面气体扩散模型；

c. 假定气体污染扩散遵循高斯分布，气体浓度不受污染源类型影响，且考虑污染源为点源类型。

三、 符号说明

符号	符号含义	单位
(x, y)	横轴、纵轴位置坐标	米 m
h	两点之间的间隔	米 m
γ	变异函数	毫克 mg
z	Kriging 估计值	毫克 mg
C	高斯空间浓度	毫克 mg
σ_x, σ_y	横轴、纵轴扩散宽度	米 m

四、 问题一的模型建立与求解

首先我们将给定的数据进行可视化处理，了解数据的分布和潜在的污染区域，以直观感受污染物可能的位置。根据图 1，可以得到单一污染源最有可能的位置在点 $(10, 10)$ 附近。

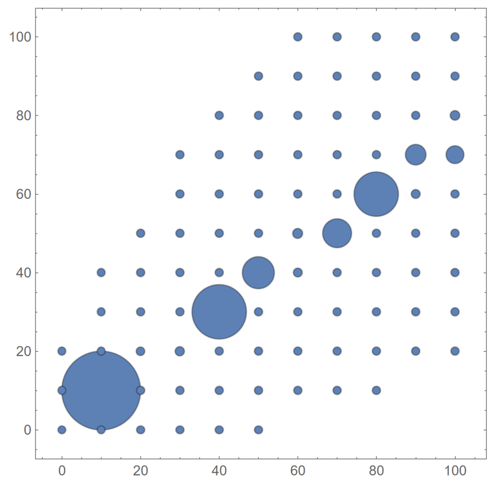

图 **1**　问题一的数据可视化

4.1 空间插值求点

在求得具体的污染源位置之前，我们将所给的空间数据点进行处理，利用空间插值技术得到每隔 1 米的气体浓度值。具体技术之一为 Kriging 空间插值方法，它主要用于地理统计数据的插值，下面对此进行简单介绍。

Kriging 的关键组成部分是变异函数，它量化了数据点之间的空间相关性。对此污染源问题，我们可以采用高斯变异，常用于连续以及平滑的空间点之间，它的表达式为：

$$\gamma(h)=\begin{cases} c_1+c_2\left[1-\exp\left\{-\left(\frac{h}{a}\right)^2\right\}\right], & h\leq 0 \\ 0, & h<0 \end{cases} \tag{1}$$

其中 $\gamma(h)$ 是变异函数在变量 h 时的值，h 表示点与点之间的间隔，c_1 为 nugget，表示测量误差或微观尺度变化，c_2 为 sill，表示模型的总方差，a 为 range，表示超过这些距离的点不会受到彼此的影响。

下面正式引入 Kriging 插值公式：

$$z(x,y)=\sum_{i=1}^{n}\lambda_i z_i(x_i,y_i) \tag{2}$$

其中 $z(x,y)$ 就是待求的估计值，$z_i(x_i,y_i)$ 为测量值，λ_i 表示权重系数。Kriging 估计算法是利用从变异函数求得的线性方程，得到 λ_i 的值，以保证最小的方差。解系统的线性方程列为：

$$\begin{bmatrix} \gamma(x_1,x_1) & \cdots & \gamma(x_1,x_n) & 1 \\ \vdots & \ddots & \vdots & \vdots \\ \gamma(x_n,x_1) & \cdots & \gamma(x_n,x_n) & 1 \\ 1 & \cdots & 1 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \cdots \\ \lambda_n \\ \mu \end{bmatrix} = \begin{bmatrix} \gamma(x_1,x_0) \\ \cdots \\ \gamma(x_n,x_0) \\ 1 \end{bmatrix} \tag{3}$$

其中 μ 是 Lagrange 乘子，用于限制参数，式中 (x_1,x_n) 不代表坐标，意思是 x_1 和 x_n 两点间的距离，即 h。由此我们就可以得到 λ 的值，再将它们代入 Kriging 插值公式，就可以得出待求的估计点。

根据实际编程，数据量较为庞大，运算时间较久，无法有效得出 101×101 个点的具体浓度值，因此，这里直接采用 Mathematica 编程库里的`SpatialEstimate` 函数。此函数也是基于 Kriging 空间插值进行运算。输入数据，可以得到图 2 所示空间浓度分布。再将这些点进行输出，转到 Sheet3 表格中。

4.2 单一污染源具体位置

以上我们将数据进行了扩展，已知单一污染源，下面将确定其具体位置。已经假设气体扩散遵循高斯函数分布，那么可以使用逆高斯模型来估计污染源的位置。这种方法

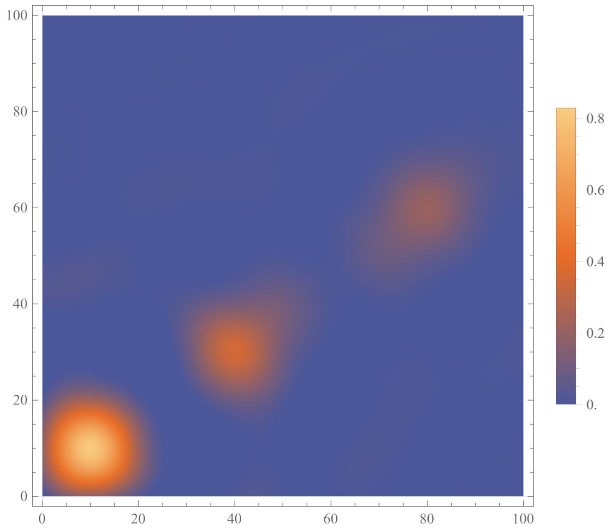

图 **2**　问题一空间浓度分布

基于假设气体浓度随距离增加而指数级衰减，且再污染源附近达到最高。已知高斯扩散模型的基本表达式为：

$$C(x,y)=\frac{Q}{2\pi\sigma_x\sigma_y}\exp\left\{-\frac{(x-x_0)^2}{2\sigma_x^2}-\frac{(y-y_0)^2}{2\sigma_y^2}\right\} \tag{4}$$

其中 $C(x,y)$ 是点 (x,y) 处的气体浓度，Q 是污染源强度（或总释放量），通常是一个未知的常数，(x_0,y_0) 表示污染源位置，σ_x 和 σ_y 是扩散在 x 方向和 y 方向上的标准偏差，描述了扩散的宽度。

在 Mathematica 中，这里使用了内置函数`NonlinearModelFit`对高斯扩散模型表达式进行非线性拟合，得到表 1 的五个拟合最优参数值。其中，$A=Q/(2\pi\sigma_x\sigma_y)$。

表 **1**　扩散模型拟合参数

x_0 (m)	y_0 (m)	σ_x (m)	σ_y (m)	A (mg)
9.95668	9.97053	6.05064	6.12333	0.846451

最终我们可以得到问题一的单一污染源的位置在坐标 (9.96, 9.70) 附近。另外，污染源的宽度大约为 6 米。

五、 问题二的模型建立与求解

对于问题二的污染源为多个，同样地，首先对数据进行可视化处理。如图 3 所示，污染源可能为 2 到 6 个。

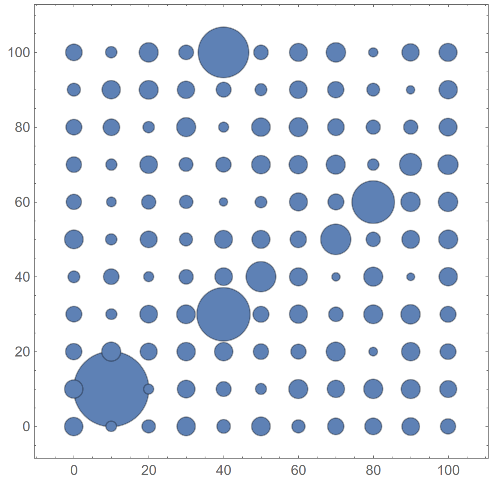

图 3　问题二数据可视化

5.1 空间插值求点

和问题一一样，首先进行空间插值，求出尽可能多的点，得到图 4 浓度分布。与问题一所得不同，问题二的污染源扩散分布范围小。

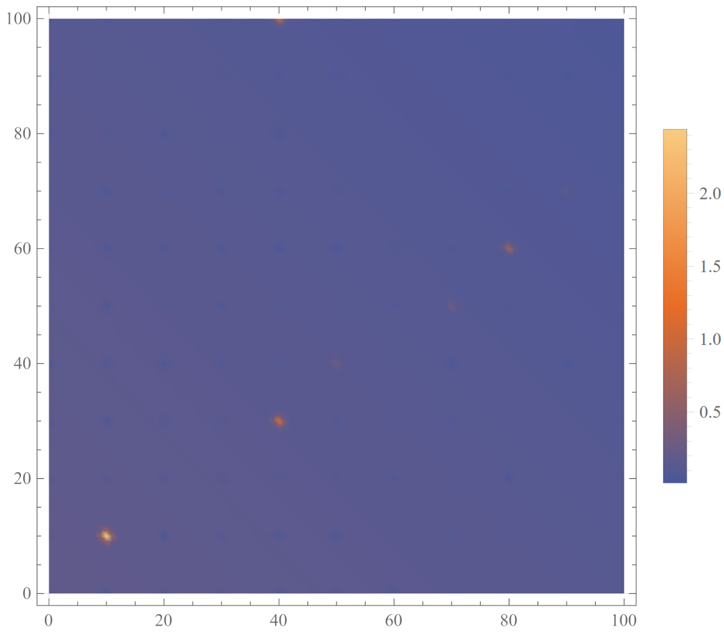

图 4　问题二空间浓度分布

5.2 多污染源数量与位置

对于多个污染源问题，我们可以通过拓展单一高斯分布模型来适应多个源。将多个高斯分布的叠加用于拟合数据，从而得以估计每个污染源的位置、强度和扩散特性。根据编程所得结果，当污染源大于等于 3 个的时候，所得方差较大，因此，只考虑了 2 个污染源，得到如表 2 所示结论。

表 **2**　问题二扩散模型拟合参数

	x_0 (m)	y_0 (m)	σ_x (m)	σ_y (m)	A (mg)
污染源 1	9.9994	9.99941	0.613674	0.613674	2.40986
污染源 2	39.9986	29.9986	0.688405	0.688405	1.08569

最后得出结论，共有两个污染源，其位置分别为 $(10, 10)$ 和 $(40, 30)$ 附近。另外，污染源 1 的宽度大概在 0.61 米，污染源 2 在 0.68 米，这和图 4 所示吻合。

六、 误差分析

对于问题一进行误差分析，得到表 3。

表 **3**　问题一误差表格

	估计值	方差	置信区间
x_0	9.95668	0.0479695	(9.86265, 10.0507)
y_0	9.97053	0.0486499	(9.87517, 10.0659)
σ_x	6.05064	0.0514488	(5.94979, 6.15149)
σ_y	6.12333	0.0523012	(6.02081, 6.22585)
A	0.846451	0.00666647	(0.833383, 0.859519)

对于问题二进行误差分析，得到表 4。

表 **4**　问题二误差表格

	估计值	方差	置信区间
x_1	9.9994	0.0454122	(9.91038, 10.0884)
y_1	9.99941	0.0454122	(9.91039, 10.0884)
x_2	39.9986	0.0922547	(39.8177, 40.1794)
y_2	29.9986	0.0922547	(29.8178, 30.1794)

七、 模型的评价与推广

模型最后的分析可以再进行结果可视化处理，将数据点和拟合的曲面结合在一起展示。对于多源复杂性处理，在实际应用中，可能需要根据特征调整模型，比如增加更多污染源，或者调整模型结构以反映更复杂的物理现象。其次，可以采用更加灵活的模型，如构建一个能够根据污染源数量动态变化的高斯模型。另外，对于非线性拟合，由于需要对参数进行合理的初始猜测，可以对数据进行进一步预处理，如聚类分析等，以达到更加准确的解。

模型未考虑的环境影响等，在实际的环境科学和工业应用中也尤为重要。Kriging 通常在地理信息系统（GIS）非常有用，如用于管理、分析和可视化地理空间数据，特别是在进行空间分析和热点识别时非常有用。

参考文献

[1] 正态分布 https://en.wikipedia.org/wiki/Normal_distribution

[] 克里格插值分析 https://en.wikipedia.org/wiki/Kriging

[2] https://xg1990.com/blog/archives/222

[3] https://pro.arcgis.com/en/pro-app/latest/tool-reference/3d-analyst/how-kriging-works.htm

附录 A 代码

```
(*数据输入及可视化*)
oneImport=Import[" \\data.xlsx",{"Sheets","Sheet1"}];
multiImport=Import[" \\data.xlsx",{"Sheets","Sheet2"}];
one=Flatten[Rest/@Rest[oneImport]];
multi=Flatten[Rest/@Rest[multiImport],1];
locs=Flatten[Table[{x,y},{x,0,100,10},{y,0,100,10}],1];
data1=MapThread[Append,{locs,one}];(*问题一数据, (x,y,z), z表示浓度*)
PointValuePlot[locs->one,"Size"](*问题一可视化*)
data2=MapThread[Append,{locs,multi}];(*问题二数据*)
PointValuePlot[locs->multi,"Size"](*问题二可视化*)
(*101*101个点*)
locs100=Flatten[Table[{x,y},{x,0,100,1},{y,0,100,1}],1];

(*问题一插值估计*)
sef1=SpatialEstimate[locs->one];(*空间估计算函数*)
estimate1=Table[sef1[{x,y}],{x,0,100,1},{y,0,100,1}];
results1=MapThread[Append,{locs100,Clip[Flatten[estimate1],{0,Infinity}]}];(*Clip:
    去除负值*)
ListDensityPlot[results1,PlotRange->All,PlotTheme->"Scientific"]
(*问题二插值估计*)
sef2=SpatialEstimate[locs->multi];
estimate2=Table[sef2[{x,y}],{x,0,100,1},{y,0,100,1}];
results2=MapThread[Append,{locs100,Clip[Flatten[estimate2],{0,Infinity}]}];
ListDensityPlot[results2,PlotRange->All,PlotTheme->"Scientific"]

(*将结果输入 Sheet 3 和 Sheet 4*)
Export[" \\quesetion1.xlsx",estimate1,"XLSX"];
Export[" \\quesetion2.xlsx",estimate2,"XLSX"];


(*高斯扩散模型*)
(*问题一: 一个污染源*)
gaussianModel[x_,y_,x0_,y0_,\[Sigma]x_,\[Sigma]y_,A_]:=A
    Exp[-((x-x0)^2/(2\[Sigma]x^2))-(y-y0)^2/(2\[Sigma]y^2)];
fitParas=NonlinearModelFit[results1,
  {gaussianModel[x,y,x0,y0,\[Sigma]x,\[Sigma]y,A],{\[Sigma]x>0,\[Sigma]y>0,A>0}},
  {x0,y0,\[Sigma]x,\[Sigma]y,A},{x,y}];
params=fitParas["BestFitParameters"](*拟合参数结果*)
fitParas["ParameterTable"](*误差分析*)
fitParas["ParameterConfidenceIntervalTable"]

(*问题二: 多个污染源*)
multigaussianModel[x_,y_,multiParams_List]:=
   Sum[multiParams[[5i-4]] Exp[-((x-multiParams[[5i-3]])^2/(2multiParams[[5i-1]]^2))
```

```
      -(y-multiParams[[5i-2]])^2/(2multiParams[[5i]]^2)],{i,Length[multiParams]/5}];
p=Array[a,2*5];
twoFitParas=NonlinearModelFit[results2,{multigaussianModel[x,y,p],And@@Thread[p>0]},
   {p[[1]],{p[[2]],10},{p[[3]],10},p[[4]],p[[5]],p[[6]],{p[[7]],40},{p[[8]],30},p[[9]],p[[10]]},
   {x,y}];
twoParams=twoFitParas["BestFitParameters"]
twoFitParas["ParameterConfidenceIntervalTable"]
```