基于微分和递归思想的太湖水悬浮污染物浓度预测模型

摘要

对于问题一, 注意到湖泊中的污染物总量是只一个关于时间变化的函数，所以首先建立用浓度和流速表示污染物总量的微分方程，并将污染物质量转化为用已知的湖泊容量表示，得到了湖水污染物浓度关于时间变化的线性微分方程。化简后采用一阶线性微分方程的通解来求解，并使用题目提供的污染物浓度数据来拟合曲线求解参数，建立出太湖水污染物浓度随时间序列的预测模型。代入时间，得到第 20 周的污染物监测浓度平均值是 2.95mg/L，第 30 周的污染物监测浓度平均值是 2.81mg/L。

对于问题二, 在问题一的基础上，首先将题设中给出的原始状态污染物浓度和改进工艺并吸附以后的污染物浓度排放浓度分别代入问题一得到的污染物浓度随时间序列的预测模型中。令 t=0, 得出初始状态下太湖水中的污染物浓度。随后按照问题一的思路建立微分方程，并将两次间隔时间（即 t=1）代入后，将微分方程转化为递推公式，得到问题二的污染物浓度预测模型。将初始状态太湖水中的污染物浓度值代入递推公式模型，即可以开始迭代求解。得到原始情况时污染物检测浓度时间序列，从第一周的 10.83mg/L 下降到第 30 周的 2.8mg/L；新措施实施后的污染物检测浓度序列，从第一周的 10.4mg/L 下降到第 30 周的 0.44mg/L。详细数据保存在题 2 前 30 周原始浓度数据.xlsx 和题 2 前 30 周新措施下浓度数据.xlsx 中。将原始状态下和工艺更新并吸附后的两组预测时间数据做差并与原始状态相比，就建立了浓度下降比例模型。得到污染物浓度比例从第一周的 4% 下降到第 30 周的 84%。

对于问题三, 在问题二的基础上，在微分方程的建立中加上生物分解质量的影响，在同样的建模思路下，得到问题三的污染物浓度预测模型，得到新措施实施后的污染物检测浓度从第一周的 9.79g/L 下降到第 15 周的 0.27mg/L。详细数据记录在题 3 前 15 周综合措施下浓度数据.xlsx 中。同样使用问题二中建立的浓度下降比例模型，得到污染物浓度比例从第一周的 9.57% 下降到第 15 周的 91.6%, 浓度下降比例记录在题 3 前 15 周浓度下降比例 (%).xlsx 中。

关键字:　　微分方程　曲线拟合　递推公式

一、 问题重述

1.1 问题背景

随着城市化和工业化的不断发展，造成水资源污染的工业排放日益增多。虽然湖水以其流动性和自净能力能在一定程度上降低污染物浓度，但着远远不足以达到遏制环境污染的程度。因此，采用先进手段在源头上对工业排放的污水进行净化，设立污染物浓度监测点开展常态化监测对保证生态环境的可持续发展具有重要意义。

1.2 问题要求

问题 1 要求不考虑其他因素的影响，在未知对产量配额后排放污水浓度的情况下，根据环保部门测得的第 1-15 周五个监测点的污染物浓度数据，建立数学模型，预测第 20 周和第 30 周的污染物浓度。

问题 2 要求在已知造纸厂排放流速，排出污染物浓度限额和活性炭吸附作用强度的情况下，建立数学模型描述湖水中污染物浓度与时间的变化关系，并计算污染物较原先下降的比例。

问题 3 要求在问题 2 的基础上加入微生物分解的影响，在已知微生物分解作用与总群数量的比例系数、太湖中微生物总群数量的监测数据的情况下，建立数学模型，描述湖水中污染物浓度的变化关系，并求出污染物在经历吸附，工艺提升和微生物分解之后下降的比例。

二、 问题分析

2.1 问题一分析

对于问题一，由于排入水中的污染物浓度，排入流速，排出的污染物浓度，排入流速都是未知的一个定值，对于多组污染物浓度数据建立方程是可行的，因此我们选择寻找不变量来建立等式。由于湖泊的容积是一个定值，同时由题意可知，湖泊中的污染物浓度可以视作一个随时间变化的函数，因此可以表示出湖水中的污染物总量。因为其也等价于流入的污染物与流出的之差，我们列出微分方程来求解。通过拟合求解一阶线性微分方程得到第 20 周和第 30 周的污染物浓度。

2.2 问题二分析

对于问题二，排入水中的污染物浓度数据由题目给出，因此我们首先把两个浓度数据代入问题一建立的模型中。随后按照问题一的思路建立类似的微分方程。在这里，因为要求求解污染物浓度变化关系，也就是求解污染物浓度的时间序列变化数据，我们将微分方程求解，并改写成递推公式的形式来迭代即可。在求解时先令 t=0，就可以得到初始污染浓度，在代入递推公式开始递推。结果记录在题 2 前 30 周原始浓度数据.xlsx 和题 2 前 30 周新措施下浓度数据.xlsx 中。

对于要求计算的污染物下降比例，只需将两个状态是的时间序列污染物浓度数据做差，并除以原始浓度即可得到浓度下降序列数据。结果记录在题 2 前 30 周浓度下降比例 (%).xlsx 中。

2.3 问题三分析

对于问题三，问题二的思路类似，只需要在由质量关系建立微分方程时加入微生物的分解作用，即减去微生物分解掉的那一部分污染物质量，最后同样建立递推公式即可。将污染物浓度的变化关系记录在题 3 前 15 周综合措施下浓度数据.xlsx 中，浓度下降比例记录在题 3 前 15 周浓度下降比例 (%).xlsx 中。

三、 模型假设

为简化问题，本文做出以下假设：

- 假设 1 假设太湖是一个理想的均匀水体，污染物浓度在其中的分布是完全均匀的，水体不受到水流动，风力等的扰动，水中除了悬浮污染物和微生物种群没有其他颗粒物。
- 假设 2 假设太湖的容量为 44.3 亿立方千米，保持不变，在实际计算中，为了避免单位换算困难，使用微型太湖容量 44.3 升来计算求解。
- 假设 3 假设湖水的流入流出速度保持恒定不变。
- 假设 4 假设工厂排放的污水中污染物浓度与限制定额完全一致。

四、 符号说明

符号	说明	单位
m_1	单位时间流入太湖的污染物质量	mg
m_2	单位时间流出太湖的污染物质量	mg
v_1	单位时间排入水中的污染物流速	mg
v_2	单位时间的湖泊流速	m/s
c_1	单位时间排入水中的污染物浓度	mg/L
c_2	单位时间排出水中的污染物浓度	mg/L
V	太湖的湖泊容积	$10^8 \times m^3$
n	微生物总群数量	/
k	微生物分解作用与总群数量的比例系数	/
t	单位时间	周
$rate$	湖水中污染物浓度下降的比例	/
$M(t)$	湖水中的污染物总量	mg
$m(t)$	湖水中微生物分解的污染物质量	mg
$c(t)$	湖水中的污染物浓度	mg/L
$c11(t)$	初始时太湖中污染物浓度	mg/L
$c12(t)$	采用活性炭吸附后太湖中污染物浓度	mg/L

五、 问题一模型的建立和求解

5.1 问题一的模型建立

5.1.1 微分方程的建立

按照题设，由于上游造纸厂每周排污的流速和浓度均不变，设单位时间内排入水中污染物流速为 v1，浓度为 c1，则在单位时间内流入污染物的质量 m1 是一个定值。设单位时间为 dt，可得到等式

$$m_1 = v_1 \cdot c_1 \cdot dt \tag{1}$$

又由于湖泊的流速也不变，湖水中污染物的浓度可以视为一个随时间变化的函数，我们把湖泊流速记为 v2，湖水中的污染物浓度记为 c(t)，则在单位时间 dt 内, 流出湖泊的污染物质量 m2 同样可以用等式表达

$$m_2 = v_2 \cdot c(t) \cdot dt \tag{2}$$

因此，若我们把湖水中污染物总量记为 M(t)，很容易得到污染物质量随时间的变化为

$$\frac{dM(t)}{dt} = v_1 \cdot c_1 - v_2 \cdot c(t) \tag{3}$$

由于湖泊的总量是定值，为 44.3 亿立方米，我们把太湖的湖泊总容积记为 V，很容易得到等式

$$M(t) = V \cdot c(t) \tag{4}$$

根据公式 (4) 我们可以把方程 (3) 转化为

$$V \cdot \frac{dc(t)}{dt} = v_1 \cdot c_1 - v_2 \cdot c(t) \tag{5}$$

其中 v1 是单位时间排入污染水的流速，V2 是排出污染水的流速，V 是太湖的湖泊总容积，题中已给出为 44.3 亿立方米，c1 是单位时间内排入湖水中的污染物浓度，c2 是排出湖水中的污染物浓度。湖水中污染物总量记为 M(t)，湖水中污染物浓度记为 c(t)。

5.1.2 一阶线性微分方程的求解

将已知量代入后，令 y=c(t)，C 表示 v1 与 c1 的乘积，是一个确定的常数，由此可以将式子简记为

$$44.3y' - v_2 \cdot y = C \tag{6}$$

将 v2 和 C 看作方程两个给定的参量，y 视为唯一的未知数，我们可以求解这个一阶线性微分方程 (6)。从而得到关于 t 的污染物浓度关系函数 c(t)，也就是要得到 y 和 y’ 之间的映射关系。

首先把方程变形为

$$y' - \frac{v_2}{44.3}y = \frac{C}{44.3} \tag{7}$$

随后将其代入非齐次一阶线性微分方程的通解公式中

$$y^* = e^{\int P(x)dx}\left[\int Q(x)e^{\int P(x)dx}dx + C(\text{这里指通解中的常数})\right] \tag{8}$$

得到结果

$$c(t) = c_1 e^{-\frac{v_2}{44.3}}t + \frac{C}{v_2} \tag{9}$$

5.1.3 时间序列方程的建立

在此，我们将时间序列数据和 5.1.2 中得到的函数进行比对，试图求出未知的参数。在具体求解过程中，我们将时间序列数据导入 python 中，使用第三方库 scipy.optimize 中的 curve_fit 函数对每一个监测点分别进行拟合得到各个监测点的参数为

表 1　参数一览表

监测点	c1	v2	C
1	9.87001	9.09041	25.48343
2	9.89060	8.89718	24.33779
3	9.84575	9.07173	25.58456
4	9.86457	8.97011	24.92296
5	9.84157	9.03423	25.42367

至此，我们建立了**太湖水污染物浓度随时间序列的预测模型**
即

$$c(t) = c_1 e^{-\frac{v_2}{44.3}}t + \frac{C}{v_2} \tag{10}$$

其中湖水中污染物浓度记为 c(t),t 是经历的时间周期，而 c1 是排入污染水中污染物浓度，v2 是单位时间内排出水的流速，C 是 c1 和 v1 的乘积，由参数表提供。

5.2 问题一的求解

问题一的求解过程较为简单，只需要将题目要求的第 20 周和第 30 周作为时间代入太湖污染物浓度随时间序列的预测模型中，就可以得到对应 5 个检测点位的预测污染浓度数据。

5.3 求解结果

表 2　问题一求解结果

监测点	第 20 周的污染物浓度	第 30 周的污染物浓度
1	2.96625	2.82426
2	2.91359	2.75936
3	2.98414	2.84140
4	2.95036	2.80114
5	2.98077	2.83583

六、 问题二模型的建立和求解

6.1 问题二模型的建立

6.1.1 时间序列方程的确定

由题意可知，未采取新技术时在单位时间内排入水中的污染物浓度为 c11=25mg/L。令 c11=c1, 将其代入问题一建立的模型公式 (9) 中，得到初始时太湖污染物浓度随时间序列的预测模型

$$c_{11}(t) = 25 \cdot e^{-\frac{v_2}{44.3}t} + \frac{C}{v_2} \tag{11}$$

又由于题目给出的活性炭吸附作用以比例系数 0.25 计，即可以简单认为活性炭能够去除废水中 25% 的漂浮污染物。也就是说，经过活性炭吸附，可以得到

$$c_2 = 5 \times (1 - 0.25)mg/L = 3.75mg/L \tag{12}$$

我们设采用新工艺以后在单位时间内排入水中的污染物浓度为 c12=3.75mg，令 c12=c1, 得到采用新工艺后太湖污染物浓度随时间序列的预测模型

$$c_{12}(t) = 3.75 \cdot e^{-\frac{v_2}{44.3}t} + \frac{C}{v_2} \tag{13}$$

令其他参数与问题一中的模型保持不变，由此，我们得到了问题二的**太湖水污染物浓度的时间序列预测模型**

$$\begin{cases} c_{11}(t) = 25 \cdot e^{-\frac{v_2}{44.3}t} + \frac{C}{v_2} \\ c_{12}(t) = 3.75 \cdot e^{-\frac{v_2}{44.3}t} + \frac{C}{v_2} \end{cases} \tag{14}$$

6.1.2 微分方程的建立和求解

和问题一的建模过程类似，我们需要建立微分方程求解。

***Step1* 微分方程的建立**和问题一的建模类似，根据质量守恒定律，湖泊中的污染物质量变化率等于流入的污染物质量减去流出的污染物质量。因此我们可以得到

$$\frac{d}{dt}\left(V \cdot c(t)\right) = v_1 \cdot c_1 - v_2 \cdot c(t) \tag{15}$$

化简得到

$$\frac{dc}{dt} = \frac{v_1 \cdot c_1}{V} - \frac{v_2}{V} \cdot C \tag{16}$$

***Step2* 微分方程的求解**易知，方程 (16) 是一个线性微分方程，解得其解为

$$c(t) = \left(C_n - \frac{v_1 \cdot c_1}{v_2}\right) e^{-\frac{v_2}{V}} t + \frac{v_1 \cdot c_1}{v_2} \tag{17}$$

6.1.3 递推公式的确立

在此，我们在 6.1.1 中得到的微分方程的基础上，尝试构造递推公式

令 Cn 是第 n 周的污染浓度，n=0,1,2,3⋯⋯，我们将 t(两次检测之间的时间间隔) 代入 c(t)，即可求得下一次检测时湖泊中的污染物浓度 Cn+1:

$$C_{n+1} = \left(C_n - \frac{v_1 \cdot c_1}{v_2}\right) e^{-\frac{v_2}{V}} t + \frac{v_1 \cdot c_1}{v_2} \tag{18}$$

由此，我们就把微分方程改写成了递推公式。

又由于 t=1(周) 在此是确定的常量，我们直接将其代入，得到

$$C_{n+1} = \left(C_n - \frac{v_1 \cdot c_1}{v_2}\right) e^{-\frac{v_2}{V}} + \frac{v_1 \cdot c_1}{v_2} \tag{19}$$

6.1.4 浓度下降比例预测模型的建立

首先，根据题目描述我们很容易得到关于污染物浓度下降比例的计算公式

$$\text{下降比例} = \frac{\text{原始浓度} - \text{新措施后的浓度}}{\text{原始浓度}} \times 100\% \tag{20}$$

也就是说，我们只要将太湖中不进行任何措施的时候的浓度的时间序列数据与太湖中进行新措施的时候的浓度的时间序列数据作差比较，并除以太湖中不进行任何措施的时候的浓度的时间序列数据，即可得到下降的污染物浓度比例。

由于初始时太湖污染物浓度被记为为 c11(t), 新措施使用活性炭吸附以后的污染物浓度记为 c12(t), 我们可以将式子简单写成

$$rate = \frac{c_{11}(t) - c_{12}(t)}{c_{11}(t)} \times 100\% \tag{21}$$

6.2 污染物浓度和下降比例的预测模型

由上述建模过程可以得到问题二的数学模型为:

$$\begin{cases} C_{n+1} = \left(C_n - \frac{v_1 \cdot c_1}{v_2}\right) e^{-\frac{v_2}{V}} + \frac{v_1 \cdot c_1}{v_2} \\ rate = \frac{c_1 - c_2}{c_1} \times 100\% \\ c_{11} = 25 \\ c_{12} = 3.75 \end{cases} \tag{22}$$

6.3 问题二模型求解

问题二的求解过程采用了递推的方法，也就是对已经建立的递推公式不断迭代。在得到两个关于时间序列的太湖水悬浮污染物浓度预测序列数据后，将其代入浓度下降比例预测模型中，得到比例随时间的变化关系。

由于在做题过程中，我们发现到达 30 周后，浓度曲线才趋于平缓，所以我们计算出了第 1-30 周的浓度数据情况，便于得到较为正确的浓度下降比例结果。

Step1：求解初始状态下 30 周的污染物浓度预测数据

首先对初始状态时太湖污染物浓度随时间序列的预测模型 (11) 中的 t 取 0，得到第 0 周也就是初始状态时的污染物浓度数值，随后依次读入题设中的第 1-15 周的造纸悬浮颗粒污染物浓度测量值数据，并将得到的初始浓度代入递推公式，这样就完成了递推的条件。于是对递推公式 (19) 进行迭代，得到初始状态下 30 周的污染物浓度预测数据。输出为题 2 前 30 周原始浓度数据.xlsx。

Step2：求解新措施实施后 30 周的污染物浓度预测数据

类似的，我们对改进工艺后的太湖污染物浓度随时间序列的预测模型 (13) 同样执行 Step1 中的步骤，得到使用新工艺和活性炭吸附以后 30 周的污染物浓度预测数据。输出为题 2 前 30 周新措施下浓度数据.xlsx。

Step3：求解浓度下降比例预测结果

采用双重循环遍历两个浓度预测数据列表，并依次将对应周数的两个浓度数据代入浓度下降比例预测方程 (21), 即可得到第 1-30 周的浓度下降比例的数据。输出为题 2 前 30 周浓度下降比例 (%).xlsx，同时采用作图工具将其可视化。

6.4 求解结果

6.4.1 数据描述

求解结果具体见附录:

题 2 前 30 周浓度下降比例 (%).xlsx

题 2 前 30 周原始浓度数据.xlsx

题 2 前 30 周新措施下浓度数据.xlsx

大体上来说，污染物浓度下降比例从第一周的 4% 下降到第 30 周的 84%, 下降了 **80%**;

原始情况时污染物检测浓度从第一周的 10.83mg/L 下降到第 30 周的 2.8mg/L, 新措施实施后的污染物检测浓度从第一周的 10.4mg/L 下降到第 30 周的 0.44mg/L。

6.4.2 可视化结果

以检测点一为例，以可视化的方式展示求解结果

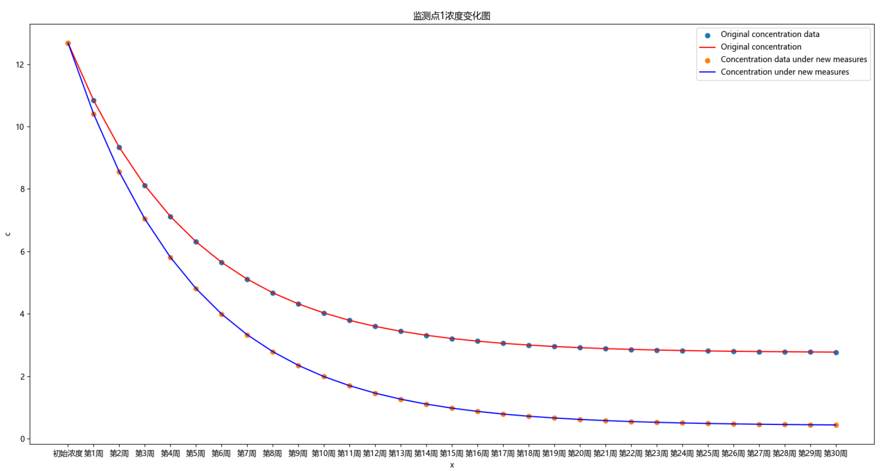

图 1　两种情况下的污染物浓度下降趋势图

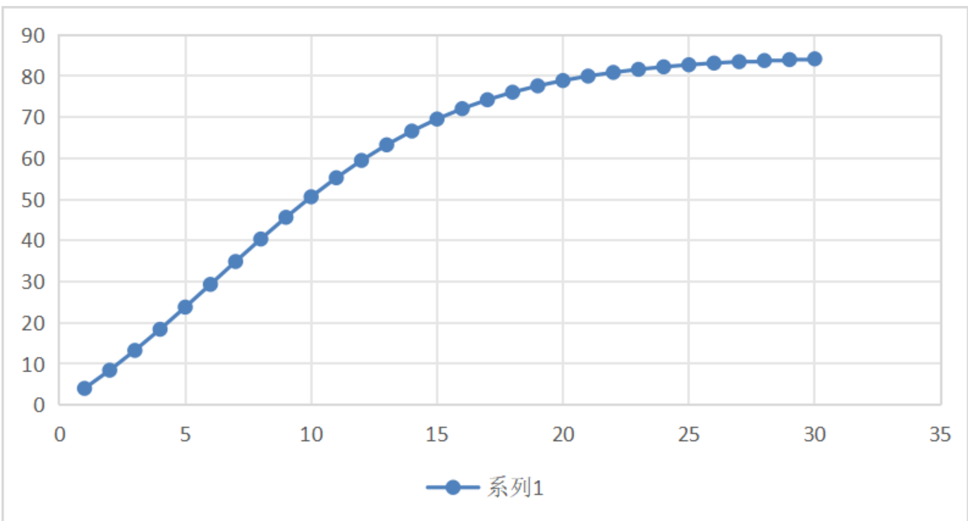

图 2　两种情况下的污染物浓度下降趋势图

七、问题三模型的建立和求解

7.1 问题三模型的建立

首先，在问题二建模的基础上，同样根据质量守恒定律，湖泊中的污染物质量变化率等价于流入的污染物质量减去流出的污染物质量，**还要再减去微生物分解的污染物质量**。由题目条件可知，微生物对悬浮污染物的分解作用与总群数量成 0.03 的线性比例关系。

我们令比例系数 k 为 0.03, 设题目提供的微生物总群总量记为 n, 容易得出微生物分解的污染物质量 m(t) 为

$$m(t)=k\cdot n\cdot V\cdot c(t) \tag{23}$$

因此对问题二中的等式进行调整，可以得到

$$\frac{d}{dt}\left(V\cdot c(t)\right)=v_1\cdot c_1-v_2\cdot c(t)-k\cdot n\cdot V\cdot c(t) \tag{24}$$

化简得

$$\frac{dc}{dt}=\frac{v_1\cdot c_1}{V}-\left(\frac{v_2}{V}+k\cdot n\right)\cdot C \tag{25}$$

对这个线性微分方程求解，其解为

$$c(t)=\left(C_n-\frac{v_1\cdot c_1}{v_2+k\cdot n\cdot V}\right)e^{-\left(\frac{v_2}{V}+k\cdot n\right)}+\frac{v_1\cdot c_1}{v_2+k\cdot n\cdot V} \tag{26}$$

在计算下一次检测时的浓度时，我们将 t(两次检测之间的时间间隔) 代入 c(t)，令 t=1, 即可求得下一次检测时湖泊中的污染物浓度 Cn+1:

$$C_{n+1}=\left(C_n-\frac{v_1\cdot c_1}{v_2+k\cdot n\cdot V}\right)e^{-\left(\frac{v_2}{V}+k\cdot n\right)}+\frac{v_1\cdot c_1}{v_2+k\cdot n\cdot V} \tag{27}$$

由此我们得到了问题三的**太湖中的悬浮污染物浓度变化模型**

$$\begin{cases} C_{n+1}=\left(C_n-\frac{v_1\cdot c_1}{v_2+k\cdot n\cdot V}\right)e^{-\left(\frac{v_2}{V}+k\cdot n\right)}+\frac{v_1\cdot c_1}{v_2+k\cdot n\cdot V} \\ k=0.03 \end{cases} \tag{28}$$

由于浓度下降比例预测模型仍为公式 (21) 不变，所以得到问题三的数学模型为:

$$\begin{cases} C_{n+1}=\left(C_n-\frac{v_1\cdot c_1}{v_2+k\cdot n\cdot V}\right)e^{-\left(\frac{v_2}{V}+k\cdot n\right)}+\frac{v_1\cdot c_1}{v_2+k\cdot n\cdot V} \\ rate=\frac{c_1-c_2}{c_1}\times 100\% \\ k=0.03 \end{cases} \tag{29}$$

7.2 模型求解

问题三的模型求解过程和问题二类似，只需要在新工艺吸附的前提下加上生物的分解作用，将问题三题设中给出的 15 周的太湖中微生物总群数量监测预估数据导入建立的数学模型后，综合前两问的求解结果和参数即可求解。

Step1：求解初始状态下 15 周的污染物浓度预测数据

与问题二的求解过程相同，不再赘述。所得结果记录在题 3 前 15 周原始浓度数据.xlsx 中。**Step2**：求解综合治理之后 15 周的污染物浓度预测数据

将递推公式替换为 (27)，其余过程保持不变，这样就计入了生物分解的作用，输出结果存放在题 3 前 15 周综合措施下浓度数据.xlsx 中。

Step3：求解浓度下降比例预测结果

同样使用浓度下降比例预测模型 (21), 得出污染物浓度下降的比例数据，记录在题 3 前 15 周浓度下降比例 (%).xlsx 中。

7.3 求解结果

7.3.1 数据描述

由于求解结果数据量较大，具体见附录：

题 3 前 15 周浓度下降比例 (%).xlsx

题 3 前 15 周原始浓度数据.xlsx

题 3 前 15 周综合措施下浓度数据.xlsx

总体来说，污染物浓度比例从第一周的 9.57% 下降到第 15 周的 91.6%, 新措施实施后的污染物检测浓度从第一周的 9.79g/L 下降到第 15 周的 0.27mg/L。

7.3.2 可视化结果

以检测点一为例，以可视化的方式展示求解结果

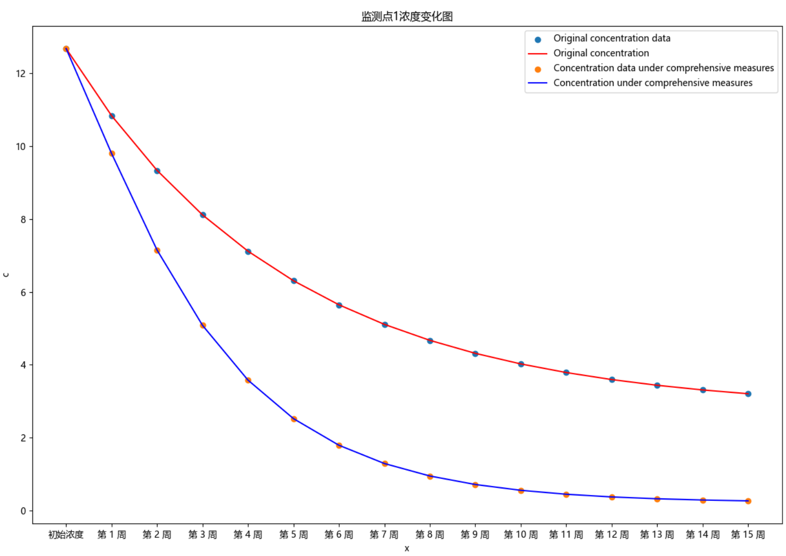

图 3　两种情况下的污染物浓度下降趋势图

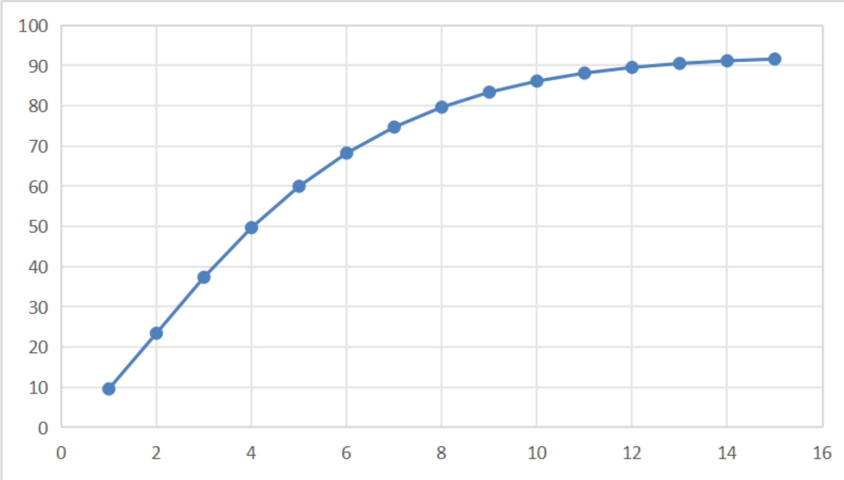

图 4　污染物浓度比例下降趋势图

section 模型的分析与检验 由于在本题的建模过程中，我们沿用了微分思想，并都使用微分方程进行求解。因此，我们仅需使用问题一中建立的模型来进行误差分析，即可验证模型的准确性。

将问题一中题目给出的实际 1-15 周的污染物浓度数据与用曲线拟合得到的微分方程计算出的数据对比，得到五个监测点的拟合率数值。

表 3　拟合率分析

监测点	$\mathrm{R}^2 score$
1	0.99987
2	0.99989
3	0.99987
4	0.99989
5	0.99987

易知，模型的拟合率非常好，都达到了 999.9% 以上，所以建立的模型具有良好的描述太湖水中悬浮污染物的能力。

当然，我们的模型是在基于绝对理想的情况下建立的。而在实际情况中，太湖是一个非常复杂的天然水域，水体的悬浮颗粒物上会有众多其他物质吸附，使得这些颗粒在一定的水动力下在水体的底部积聚。同时，水体里面还要很多其他化学成分，沙土，微小湍流等等。因此，太湖的水体应该以一个二维甚至三位的视角来建模解决问题。这也是我们的模型欠缺的地方。

八、 模型的评价

8.1 模型的优点

- 优点 1 综合性：模型综合考虑了湖泊污染物浓度变化的多个过程，能够较全面地反映湖泊中污染物浓度的动态变化。
- 优点 2 数学严谨性：通过建立微分方程并求解，模型具有数学上的严谨性。这种方法能够更准确地描述污染物在湖泊中的动态平衡过程。
- 优点 3 参数可调：模型中的参数可以根据实际情况进行调整，使得模型更加符合具体湖泊的污染情况
- 优点 4 预测能力：模型能够预测未来某一时刻湖泊中的污染物浓度，为环保部门提供决策支持。

8.2 模型的缺点

- 缺点 1 未考虑其他因素：模型没有考虑其他可能影响湖泊污染物浓度的因素，如湖泊底部的沉积物释放、湖泊中的其他生物活动（如藻类生长和死亡）、气候变化等。
- 缺点 2 计算复杂度：虽然模型在数学上是严谨的，但在实际应用上求解微分方程需要一定的计算资源。

参考文献

[1] 华祖林, 王鹏. 河湖水环境数学模型与应用[M]. 北京: 科学出版社, 2021.

[2] 宁秀美. 河流混合污染物浓度二维移流扩散研究[M]. 北京: [出版者不详].

附录 A　文件列表

文件名	功能描述
Q1 时间序列微分方程.py	问题一程序代码
Q2 递推微分模型.py	问题二程序代码
Q3 改良递推微分模型.py	问题三程序代码
题 1 模型参数.xlsx	问题一模型参数
题 1 结果.xlsx	问题一求解结果
题 2 前 30 周浓度下降比例 (%).xlsx	问题二求解结果
题 2 前 30 周新措施下浓度数据.xlsx	问题二求解结果
题 2 前 30 周原始浓度数据.xlsx	问题二求解结果
题 3 前 15 周浓度下降比例.xlsx(%)	问题三求解结果
题 3 前 15 周综合措施下浓度数据.xlsx	问题三求解结果
题 3 前 15 周原始浓度数据.xlsx	问题三求解结果

附录 B　代码

Q1 时间序列微分方程.py

```
import numpy as np
from scipy.optimize import curve_fit
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.metrics import r2_score


# 定义拟合微分函数模型
def modelFuction(x, C1, V2, C2):
    return C1 * np.exp(-(V2 / 44.3) * x) + C2 / V2

# 读取数据
df=pd.read_excel("表1.xlsx")
x=np.array([i for i in range(1,len(df.index.values)+1)])
args=[]

for i in range(len(df.columns[1:])):
```

```
    lst=df[df.columns[1:][i]].values

    # 使用curve_fit进行拟合
    pop, pov = curve_fit(modelFuction, x, lst)

    # 保存参数至列表
    args.append(pop)
    # 输出拟合得到的参数值
    print("监测点{} C□: {:.5f}, V□: {:.5f}, C: {:.5f}".format(i+1, pop[0], pop[1], pop[2]))

    # 计算拟合值
    y_fit = modelFuction(x, *pop)

    # 计算R^2分数
    r2 = r2_score(lst, y_fit)
    print("监测点{} R^2 score: {:.5f}".format(i+1, r2))

    # 绘制数据和拟合曲线
    plt.figure(figsize=(8, 4))
    plt.scatter(x, lst, label='Data')
    plt.plot(x, y_fit, 'r-', label='Fit: $y = C□ \cdot e^{-V□x/44.3} + C/V□$')
    plt.xlabel('x')
    plt.ylabel('y')
    plt.legend()
    plt.savefig(f"监测点 {i + 1}.png")
    plt.show()

    # 预测第20周和第30周的太湖污染物浓度
    print(f"监测点{i + 1} 第 20 周的污染物浓度:", round(modelFuction(20, *pop), 5))
    print(f"监测点{i + 1} 第 30 周的污染物浓度:", round(modelFuction(30, *pop), 5))

```

```
# 保存模型参数
pd.DataFrame(args, index=df.columns[1:], columns=["C□", "V□", "C"]).to_excel("题1模型参数.xlsx")
```

Q2 递推微分模型.py

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

V = 44.3  #太湖容量

# 时间序列微分模型方程
def time_series_model(x, C1, V2, C2):
    return C1 * np.exp(-(V2 / V) * x) + C2 / V2

# 递推微分模型方程
def recursive_model(Cn, V2, C2):
    return (Cn - C2 / V2) * np.exp(-(V2 / V)) + C2 / V2

# 读取数据
df = pd.read_excel("题1模型参数.xlsx")
args, args_copy, c_start, data_orig, data_changed, percentage, points, week = [], [], [], [], [], [], [], [f"第{i + 1}周" for i in range(30)]    #c_start: 太湖初始污染物浓度
c_orig, c_change, v_in, k = 25, 5, 1, 0.25
for i in range(len(df.index)):
    points.append(df.iloc[i].values[0])
    args.append(df.iloc[i].values[1:].tolist())
    c_start.append(time_series_model(0, *args[-1]))
    args[-1][0] = c_start[-1]
    args[-1][2] = c_orig * v_in
args_copy = args.copy()

# 原始浓度
for i in range(len(c_start)):
```

```
    point = []
    for j in range(30):
        point.append(recursive_model(*args_copy[i]))
        args_copy[i][0] = point[-1]
    data_orig.append(point)
print(data_orig)
pd.DataFrame(data_orig, index=points, columns=week).to_excel("题2前30周原始浓度数据.xlsx")

# 新措施下参数
for i in range(len(args)):
    args[i][0] = c_start[i]
    args[i][2] = c_change * v_in * (1 - k)
args_copy = args.copy()

# 新措施下浓度
for i in range(len(c_start)):
    point = []
    for j in range(30):
        point.append(recursive_model(*args_copy[i]))
        args_copy[i][0] = point[-1]
    data_changed.append(point)
print(data_changed)
pd.DataFrame(data_changed, index=points, columns=week).to_excel("题2前30周新措施下浓度数据.xlsx")

# 可视化数据
plt.rc("font", family='Microsoft YaHei')
for i in range(len(data_orig)):
    plt.figure(figsize=(20, 10))
    plt.scatter(["初始浓度"]+week, [c_start[i]]+data_orig[i], label='Original concentration data')
    plt.plot(["初始浓度"]+week, [c_start[i]]+data_orig[i], 'r-', label="Original concentration")
    plt.scatter(["初始浓度"]+week, [c_start[i]]+data_changed[i
```

```python
    ], label='Concentration data under new measures')
    plt.plot(["初始浓度"]+week, [c_start[i]]+data_changed[i], 'b-', label="Concentration under new measures")
    plt.xlabel('x')
    plt.ylabel('c')
    plt.title(f'监测点{i+1}浓度变化图')
    plt.legend()
    plt.savefig(f"2.监测点 {i + 1}.png")
    plt.show()

# 下降比例
for i in range(len(data_changed)):
    per = []
    for j in range(len(data_changed[i])):
        per.append(100 * np.abs(data_changed[i][j] - data_orig[i][j]) / data_orig[i][j])
    percentage.append(per)
print(percentage)
pd.DataFrame(percentage, index=points, columns=week).to_excel("题2前30周浓度下降比例(%).xlsx")
```

Q3 改良递推微分模型.py

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

# 读取数据
df = pd.read_excel("题1模型参数.xlsx")
df1 = pd.read_excel("表2.xlsx")
args, args_copy, c_start, data_orig, data_changed, percentage, points, week, number = [], [], [], [], [], [], [], df1["监测时间"].tolist(), df1["预估数量"].tolist()    #c_start: 太湖初始污染物浓度
c_orig, c_change, v_in, adsorb_ratio, decompose_ratio = 25, 5, 1, 0.25, 0.03
```

```python
V = 44.3  #太湖容量

# 时间序列微分模型方程
def time_series_model(x, C1, V2, C2):
    return C1 * np.exp(-(V2 / V) * x) + C2 / V2

# 递推微分模型方程
def recursive_model(Cn, V2, M1):
    return (Cn - M1 / V2) * np.exp(-(V2 / V)) + M1 / V2

# 改良递推微分模型方程
def improved_recursive_model(Cn, V2, M1, num):
    return (Cn - M1 / (V2 + decompose_ratio * num * V)) * np.exp(-((V2 / V) + decompose_ratio * num)) + M1 / (V2 + decompose_ratio * num * V)

for i in range(len(df.index)):
    points.append(df.iloc[i].values[0])
    args.append(df.iloc[i].values[1:].tolist())
    c_start.append(time_series_model(0, *args[-1]))
    args[-1][0] = c_start[-1]
    args[-1][2] = c_orig * v_in
args_copy = args.copy()

# 原始浓度
for i in range(len(c_start)):
    point = []
    for j in range(15):
        point.append(recursive_model(*args_copy[i]))
        args_copy[i][0] = point[-1]
    data_orig.append(point)
print(data_orig)
pd.DataFrame(data_orig, index=points, columns=week).to_excel("题3前15周原始浓度数据.xlsx")

```

```python
# 综合措施下参数
for i in range(len(args)):
    args[i][0] = c_start[i]
    args[i][2] = c_change * v_in * (1 - adsorb_ratio)
args_copy = args.copy()

# 综合措施下浓度
for i in range(len(c_start)):
    point = []
    for j in range(15):
        point.append(improved_recursive_model(*args_copy[i], number[j]))
        args_copy[i][0] = point[-1]
    data_changed.append(point)
print(data_changed)
pd.DataFrame(data_changed, index=points, columns=week).to_excel("题3前15周综合措施下浓度数据.xlsx")

# 可视化数据
plt.rc("font", family='Microsoft YaHei')
for i in range(len(data_orig)):
    plt.figure(figsize=(15, 10))
    plt.scatter(["初始浓度"]+week, [c_start[i]]+data_orig[i], label='Original concentration data')
    plt.plot(["初始浓度"]+week, [c_start[i]]+data_orig[i], 'r-', label="Original concentration")
    plt.scatter(["初始浓度"]+week, [c_start[i]]+data_changed[i], label='Concentration data under comprehensive measures')
    plt.plot(["初始浓度"]+week, [c_start[i]]+data_changed[i], 'b-', label="Concentration under comprehensive measures")
    plt.xlabel('x')
    plt.ylabel('c')
    plt.title(f'监测点{i+1}浓度变化图')
    plt.legend()
    plt.savefig(f"3.监测点 {i + 1}.png")
```

```
    plt.show()

# 下降比例
for i in range(len(data_changed)):
    per = []
    for j in range(len(data_changed[i])):
        per.append(100 * np.abs(data_changed[i][j] - data_orig[i][j]) / data_orig[i][j])
    percentage.append(per)
print(percentage)
pd.DataFrame(percentage, index=points, columns=week).to_excel("题3前15周浓度下降比例(%).xlsx")
```