鄱阳湖湖泊水位的预测

摘要

本文在对湖泊水位的预测分析中，通过先对数据的预处理，对无效数据剔除处理，合并有效数据，将不平稳数据差分处理，从而形成较完整的数据集。通过分析三个问题的要求，在借助于 SPSSPRO,SPSS,以及 Matlab 等数据分析软件，选择合适的模型进行数据处理、模型训练、预测和结果可视化等任务，在比较中选出拟合效果较好的模型，使预测的相对误差最小，并且准确找到影响水位变化较大的外部因素。

针对问题一，本文首先对数据进行预处理，对附件 1 中的无效数据采取剔除处理，并将不平稳的数据进行一阶差分处理转化成平稳序列，然后采用 AIC 准则和 BIC 准则进行定阶生成模型，题目要求根据前几年的历史水位数据预测 2021 年的水位数据并用真实数据进行验证对比，本文在 ARMA 模型预测结果相对误差较大，引入时间分解模型将数据分为不同成分更好理解趋势和季节性，使预测相对误差能够在 0.1 以下。

针对问题二，本文首先对两个附件中的数据进行预处理和整合，得到一个更完整的数据集。然后，本文使用两个模型来分析鄱阳湖 A 站点水位变化的外部因素，并通过对比发现决策树回归模型相较于 Lasso 回归模型表现更好，因此选择该模型作为主要分析模型。最后，本文使用 MSE、RMSE、MAE、MAPE、R^2 等指标对模型进行评价，以找到影响鄱阳湖 A 站点水位变化较大的外部因素。

针对问题三，本文首先对三个附件中的数据进行预处理，然后根据问题二得到的三个主要变量进行数据分析。接着使用 BP 神经网络回归模型作为预测模型，并使用训练集数据进行模型训练。最后，将建立的模型应用到训练和测试数据中，并根据测试数据的预测精度对模型进行评价。需要注意的是，BP 神经网络模型具有随机性，每次运算的结果不一样，因此不能得到确定的方程。

关键字：鄱阳湖水位 预测 ARMA 影响因素 决策树回归 Lasso bp 神经网络

一、问题重述

1.1 问题背景

湖泊水位预测对于水资源管理和调度、可持续水电开发、防洪减灾等方面非常重要。精确的水位预测对于水利基础设施如拦河坝和堤坝的有效运行至关重要。湖泊水位的变化会对水质、河流运输和水生态系统产生重大影响，如高水位时可能导致水坝崩溃的风险，低水位时可能对鱼类繁殖等产生不利影响。

因此，开发一个湖泊水位提前预警模型系统可以帮助采取积极措施，降低财产损失和安全隐患。然而，湖泊水位受到多种水文因素的影响，包括降雨量、温度、蒸发、来水量、泄洪量和历史水位等。此外，湖泊水位的变化与这些水文因素之间的关系并不是简单的线性关系。另外，近年来气候变化异常现象的频繁发生也增加了水位预测的复杂性和难度。

1.2 问题的提出

问题一：只利用站点 A 的历史水位数据（附件 1）建立水位预测模型，并使用 2021 年数据验证模型的预测效果。

问题二：根据给出的站点 A 周边区县（德安县、都昌县、永修县、进贤县、南昌县、新建县、鄱阳县、余干县）的气象数据，以及流入鄱阳湖的五条主要河流（抚河、赣江、饶河、信江、修水）的入湖径流量数据，分析并找出对鄱阳湖 A 站点水位变化影响较大的关键外部因素。

问题三：综合利用站点 A 的历史水位数据（附件 1）、鄱阳湖周边县气象数据（附件 2）、鄱阳湖河流流入流量数据（附件 3），建立水位预测模型，并使用 2021 年数据验证模型的预测效果。

二、问题分析

2.1 问题一

针对问题一，为了对站点 A 历史水位数据建立水位预测模型，并使用 2021

年数据验证模型的预测效果，首先对提供的数据进行预处理，由于题目求解范围为2011年至2021年的十年，故需要对冗余的数据进行剔除等其他必要操作，然后需要对历史水位进行可视化分析，观察数据的趋势和波动，绘制时间序列图。其次进行平稳性检验，历史水位数据若不平稳要采用差分处理将其转化为平稳序列，本文采用的是一阶差分处理。处理后根据平稳的历史水位数据选择适当的时间序列模型来建立水位预测模型，本文采用的是ARMA模型和时间序列分解模型。

接下来，将过去十年的历史水位数据作为训练数据，将2021年的水位数据作为测试数据进行预测和对比。采用AIC准则和BIC准则进行定阶生成模型。在数据处理过程中，计算移动平均（MA）和季节指数（SI），通过季节指数（r）的修正、长期趋势（T）的计算和循环变动（C）的计算进行预测。对比ARMA模型和时间序列分解模型得到的预测结果的相对误差。经过比较发现，时间序列分解模型更适合本题。

2.2 问题二

针对问题二，为了满足题目要求建立模型分析并找出对站点A水位变化影响较大的关键外部因素，为保证最终得到的结果是客观准确的，要将两表的数据集的数据进行预处理后再合并为一个新的数据集。

设置样本量中参与模型训练的比例为70%，并将交叉验证折数设为三折。使用训练集数据来建立决策树回归模型，并获得决策树的结构。对模型评估结果进行分析，判断模型拟合程度，评估结果包括各项指标如均方误差（MSE）、决定系数（RMSE）等。根据题目要求，需要分析并找出对鄱阳湖A站点水位变化影响较大的关键外部因素。因此，在这种情况下，更重要的是拟合能力而不是仅仅预测概率的高低。选择拟合能力更高的决策树回归模型后根据模型得到的特征向量从而分析得出对站点A水位变化影响较大的关键外部因素。

2.3 问题三

针对问题三，相较于问题一是要综合分析三个附件的数据来建立水位预测模型，并使用2021年数据验证模型的预测效果，首先对提供的数据进行预处理，并根据问题二得到的结论，找出三个主要的影响因素，然后将数据合并整合成一个完整的数据集。

确定了bp神经网络的层数、节点数等结构参数后，用训练集数据来建立bp

神经网络模型，通过交叉验证集、训练集和测试集作为预测评价指标，并通过量化指标来衡量 bp 神经网络的预测效果。其中，通过交叉验证集的评价指标可以不断调整超参数，以得到可靠稳定的模型，用 MSE、RMSE、MAE 等指标来判断模型的拟合程度。根据题目要求，综合利用站点 A 的历史水位数据、鄱阳湖周边县气象数据、鄱阳湖河流入湖径流量数据，建立水位预测模型，并使用 2021 年数据验证模型的预测效果。选择拟合能力更高的，准确性更好的 bp 神经网络模型得到的预测值更接近准确值。

三、模型假设

1.数据完整性假设：附件所给的数据是完整的，没有任何遗漏或缺失。说明可以使用这些数据作为基础进行分析和建模，而不需要填补任何缺失的数值。

2.数据准确性假设：附件所给的数据的数值是准确的，没有明显的错误或误差。这说明可以在分析和建模过程中信任这些数值，而不需要对其进行额外的校正或纠正。

3.数据一致性假设：附件所给的数据在时间和空间上是一致的。说明可以将这些数据集合并在一起，并将它们视为相同时间段和地理位置下的观测数据。

四、符号说明

符号	说明	成分
X_t	时间序列在时刻的 t 观测值	
c	常数项	
ϕ_i	AR 模型的系数，表示过去观测值的权重	
ε_t	白噪声项，表示随机误差	
θ_t	MA 模型的系数，表示过去随机误差的权重	
p	AR 模型的阶数，表示过去观测值的个数	
q	MA 模型的阶数，表示过去随机误差的个数	
LIF	极大似然比	
$ParNum$	待估计参数个数	

$NumObs$	样本数	
L	最大阶数	
T_t	描述时间序列的长期变化趋势	长期趋势
S_t	表示时间序列的周期性变动	季节性
C_t	描述时间序列的非固定周期波动	循环变动
E_t	表示时间序列中的随机噪声或不可预测的波动	随机波动
MSE	均方误差,预测值与实际值之差平方的期望值	
$RMSE$	均方根误差,为 MSE 的平方根	
MAE	平均绝对误差,绝对误差的平均值，能反映预测值误差的实际情况	
$MAPE$	平均绝对百分比误差,是 MAE 的变形，是一个百分比值	
R^2	将预测值跟只使用均值的情况下相比，结果越靠近 1 模型准确度越高	

五、模型建立与求解

5.1 问题一模型建立与求解

5.1.1 数据分析

首先，取出站点 A 过去 10 年的历史水位数据存储在变量 x 中，将过去十年的历史水位数据作为训练数据。而后将 2021 年的水位数据取出作为测试数据，便于进行预测和对比。

原时间序列数据如下：

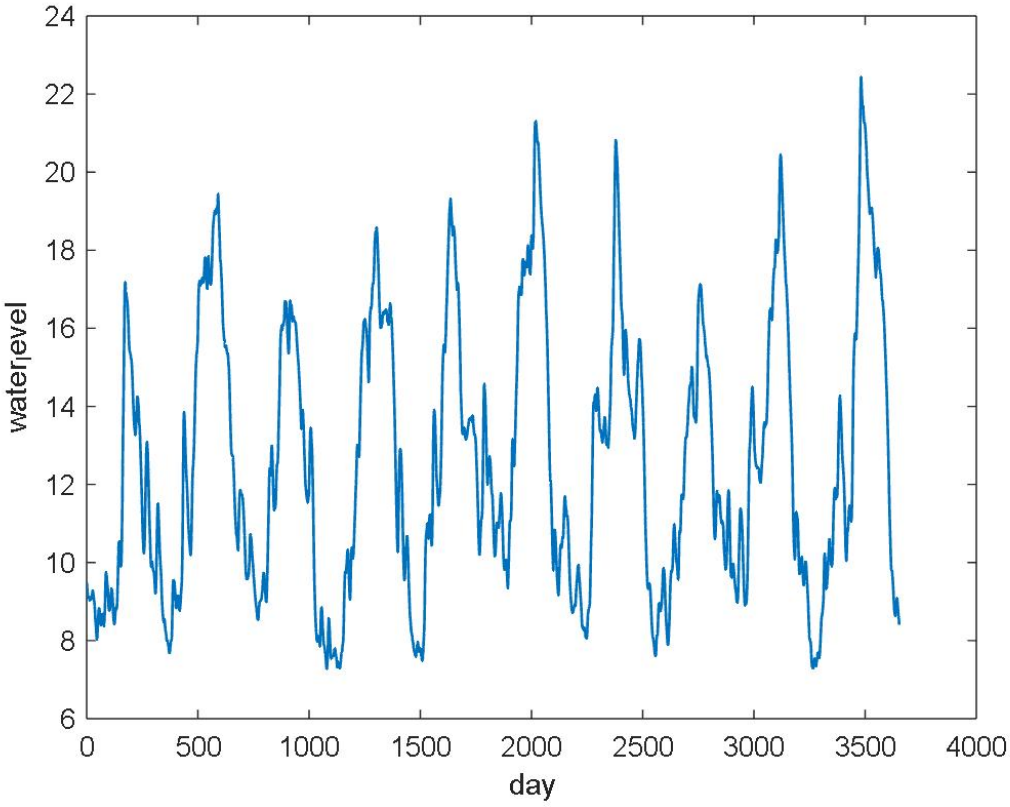

图 1 原时间序列数据和数据平稳判断

时间序列平稳性判断为不平稳，则进行差分处理，对不平稳的训练数据 train_data 进行一阶差分处理，将差分后的数据赋值给 train_data1。一阶差分可以用于减少数据的趋势和季节性。可见一阶差分处理后的序列如下，整体还是比较平稳的。

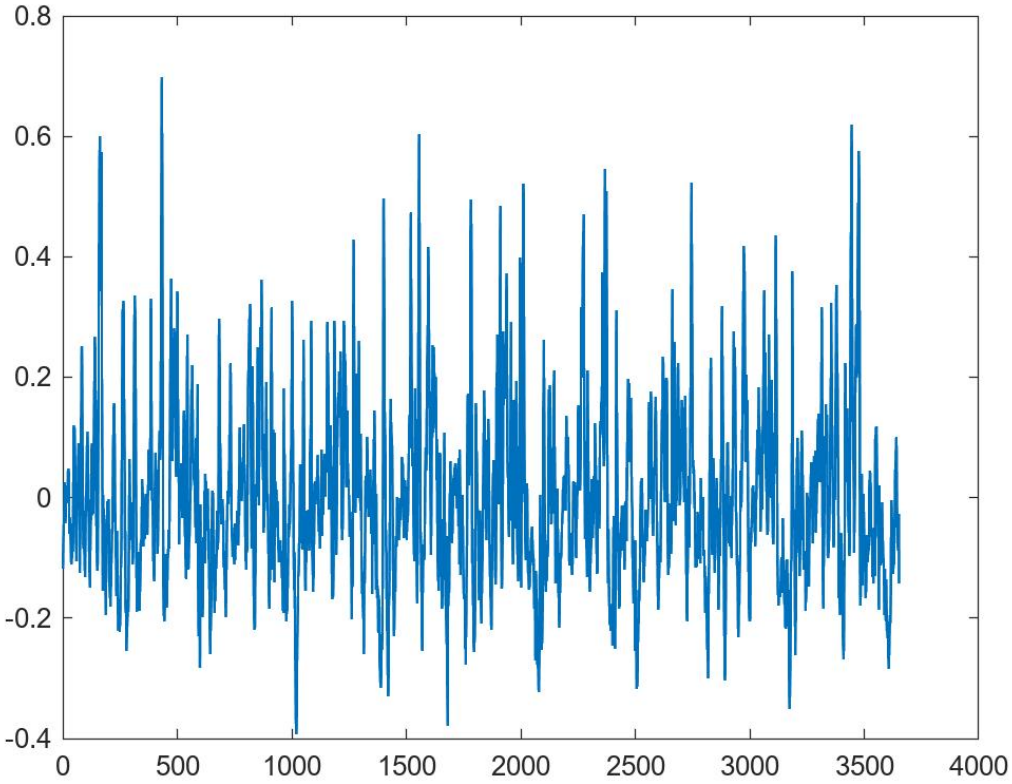

图 2 一阶差分处理后的序列

5.1.2 确定模型及定阶

根据相关性特征，可利用自相关函数与偏相关函数的截尾性来识别模型类型，并利用偏相关函数（PACF）,确定 AR 模型的之后阶数；利用自相关函数（ACF），确定 MA 模型的滞后阶数。

以下是三个模型的相关特征，如下表

模型	自相关函数（ACF）	偏相关函数（PACF）
自回归模型 AR(p)	拖尾	P 阶截尾
移动平均模型 MA(q)	q 阶截尾	拖尾
自回归移动平均模型 ARMA(p,q)	拖尾	拖尾

表 1 三个模型相关特征

截尾指的是从某阶开始均为（接近）0 的性质，拖尾指的是逐渐衰减为 0。

自回归模型 AR(p)计算方法如下如下：

$$AR(p):X_t = c + \sum_{i=1}^{p} \phi_i X_{t-i} + \varepsilon_t$$

移动平均模型 MA(q)计算方法如下如下：

$$MA(q):X_t = c + \varepsilon_t + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i}$$

自回归移动平均模型 ARMA(p,q)计算方法如下如下：

$$ARMA(p,q):X_t = c + \sum_{i=1}^{p} \phi_i X_{t-i} + \varepsilon_t + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i}$$

自相关函数图和偏相关函数图如下，可知自相关函数与偏相关函数均为截尾形态。可判断使用 ARMA 模型。

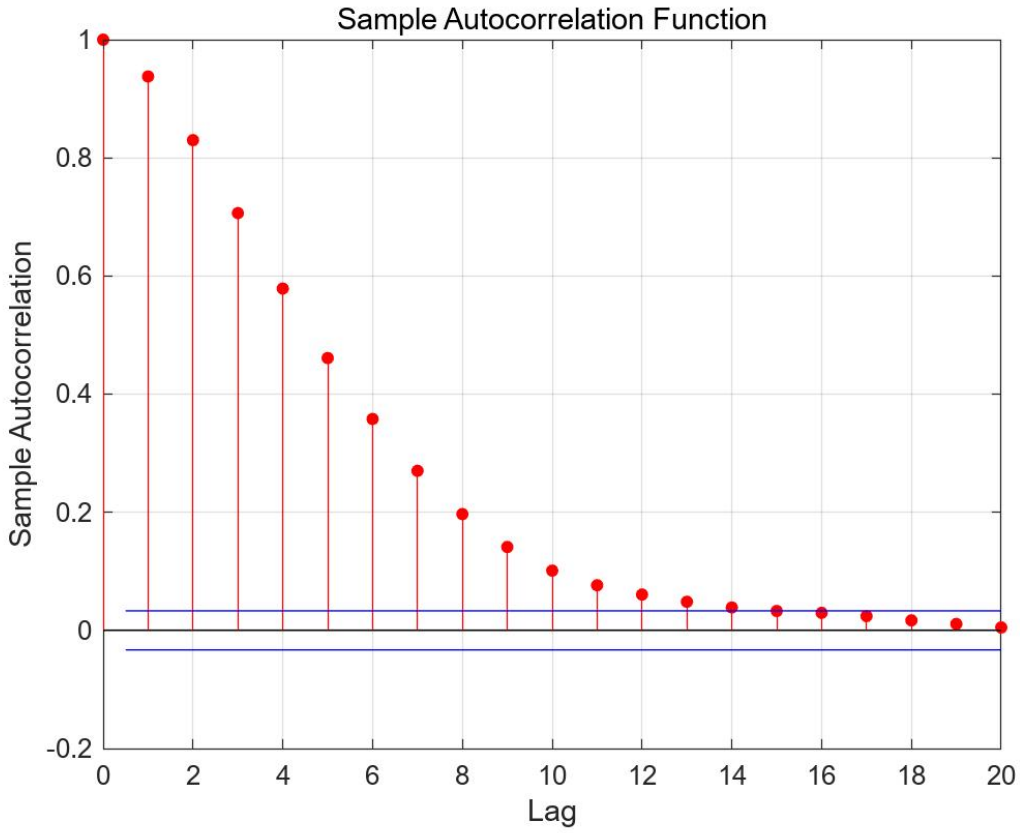

图 3 自相关函数图

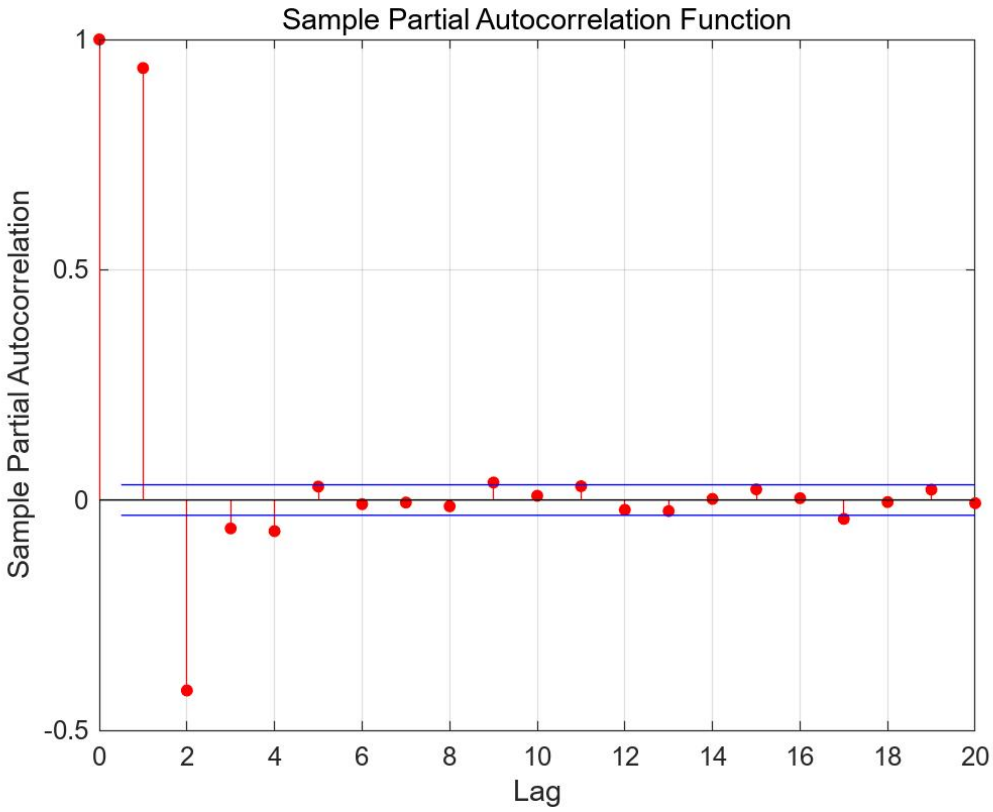

图 4 偏相关函数图

通过自相关函数和偏相关函数图不好判断模型的阶数，采用定量分析 AIC 准则和 BIC 准则来计算。

MATLAB 中 AIC 的计算公式为：

$$AIC = (-2 \times LIF) + 2 \times ParNum$$

BIC 的计算公式为:

$$BIC = (-2 \times LIF) + ParNum \times \log(NumObs)$$

较小的 AIC 或 BIC 值意味着模型具有较好的拟合和较小的复杂度。最大阶数 $L = [NumObs / 10, NumObs / 5]$，当样本个数太大时自定义一个最大阶数，阶数太大计算量大的同时模型的复杂程度也会大大增大。

利用 AIC 准则定阶找到 AIC 最小值对应的 p,q 可以确定使用 ARMA(3,2)模型。

图 5 AIC 定阶热力图

5.1.3 参数估计生成模型

（代码见附录 A）

得到的模型如下：

```
model =
Discrete-time ARMA model: A(z)y(t) = C(z)e(t)
  A(z) = 1 - 2.835 z^-1 + 2.523 z^-2 - 0.002481 z^-3 - 1.008 z^-4 - 0.6247 z^-5 + 2.438 z^-6 - 1.906 z^-7 - 0.1704 z^-8 + 0.9

  C(z) = 1 - 0.539 z^-1 - 0.3655 z^-2 + 0.4445 z^-3 + 0.2457 z^-4 - 0.7106 z^-5 + 0.5374 z^-6 + 0.5047 z^-7 - 0.2521 z^-8

采样时间: 1 seconds

Parameterization:
   Polynomial orders:   na=10   nc=8
   Number of free coefficients: 18
   Use "polydata", "getpvec", "getcov" for parameters and their uncertainties.

Status:
Estimated using ARMAX on time domain data "train_data2".
Fit to estimation data: 98.67% (prediction focus)
FPE: 0.002164, MSE: 0.002131

Model Properties
```

图 6 生成模型

5.1.4 调用模型预测

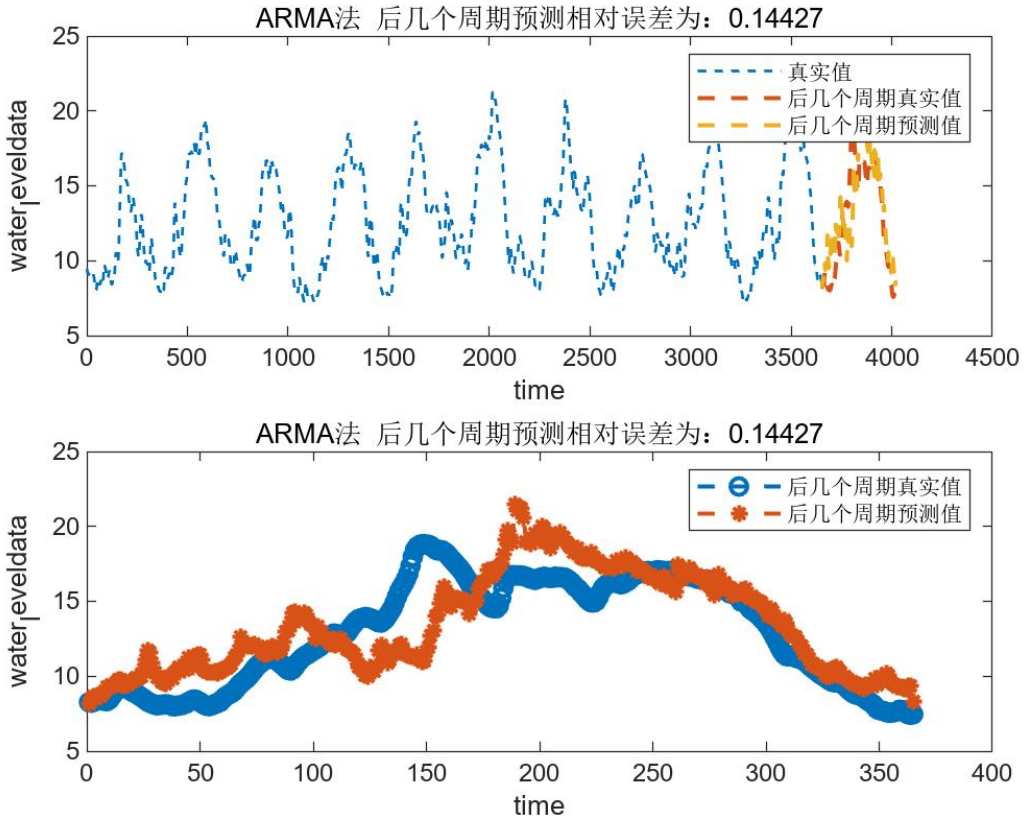

图 7 ARMA 法后几个周期预测相对误差图

5.1.5 结果分析与验证

以上采用了 ARMA 模型通过自回归和移动平均部分来描述时间序列的特征，计算出后几个周期预测相对误差为 0.14427，相对误差仍较大，在统计分析附件 1 给的数据后发现数据具有明显长期趋势和季节性，如下图。

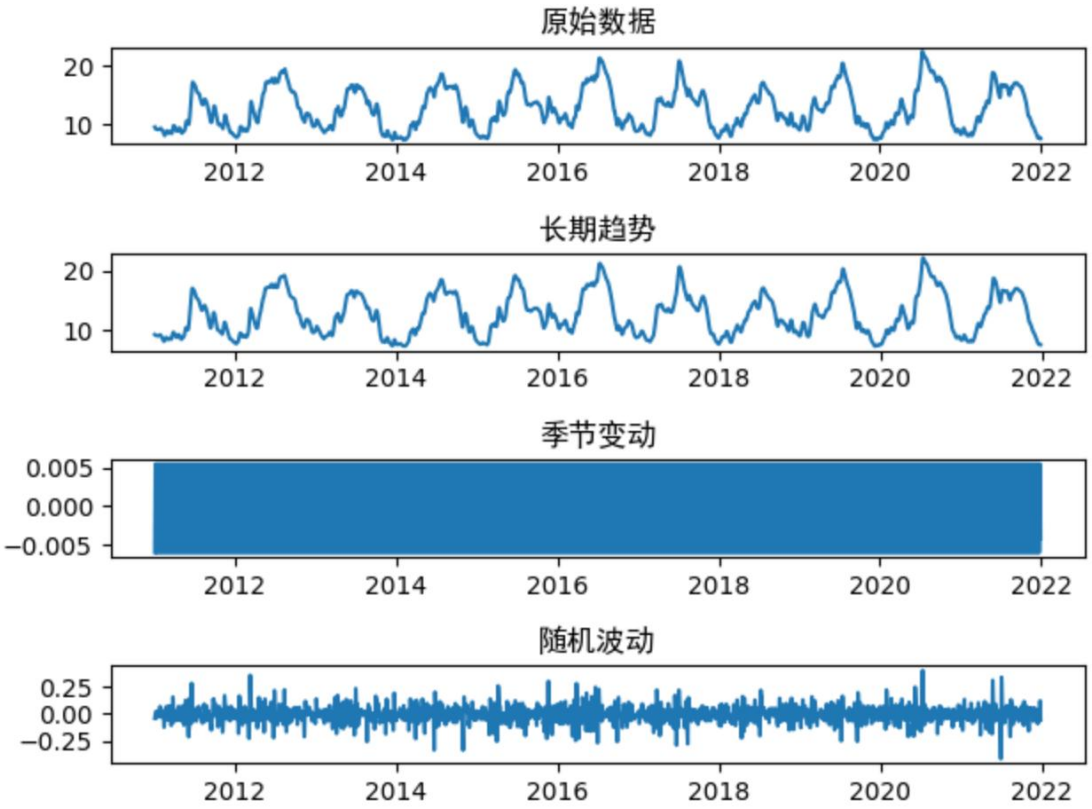

图 8 历史水位数据拆分为长期趋势、季节变动和随机波动三个成分总图

故对代码模型进行修改，采用时间序列分解法，通过将时间序列数据分解为不同成分，使我们能够更好地理解数据的趋势和季节性，独立处理各个成分，减少噪声的影响，并在预测和分析中提供更多的信息和洞察力。

时间序列分解法计算公式为：

$$X_t = T_t + S_t + C_t + S_t$$

最终对模型选择上进行优化，优化结果如下图，

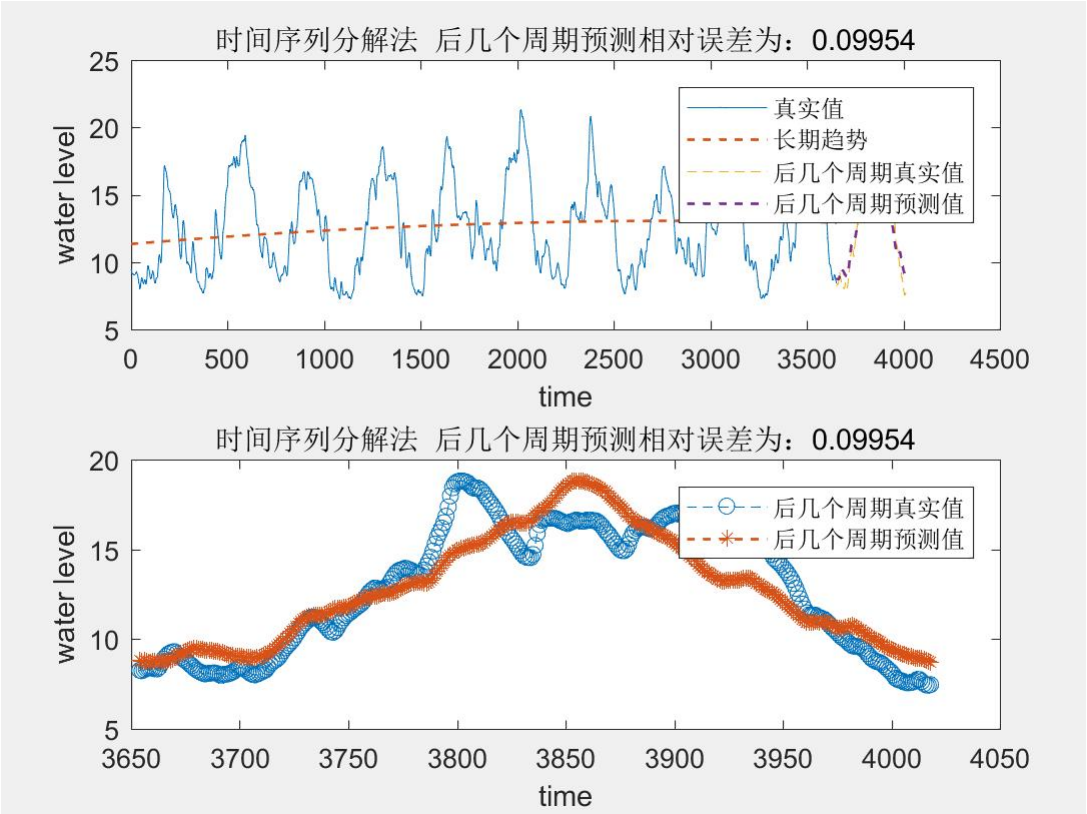

图 9 时间序列分解法后几个周期预测相对误差图

时间序列分解法的后几个周期预测相对误差相较于 ARMAM 模型更小，结果为 0.09954，为未来水位预测提供了依据。

5.2 问题二模型建立与求解

5.2.1 模型的建立

首先，我们将样本量中参与模型训练的比例设置为 70%，并将交叉验证折数设为三折。使用训练集数据来建立决策树回归模型，并获得决策树的结构。下表展示了模型各项参数配置以及模型训练时长。

参数名	参数值
训练用时	0.576s
数据切分	0.7
数据洗牌	否
交叉验证	3
节点分裂评价准则	friedman_mse

特征划分点选择标准	best
划分时考虑的最大特征比例	None
内部节点分裂的最小样本数	2
叶子节点的最小样本数	1
叶子节点中样本的最小权重	0
树的最大深度	10
叶子节点的最大数量	50
节点划分不纯度的阈值	0

表 2 模型参数

下图展示了决策树结构，内部节点给出了被分枝特征的具体切分情况，即根据某个特征的某个切分值进行划分。mse/friedman_mse/mae 等用以确定对哪一个特征进行切分，样本数量是该节点拥有的样本数量，节点样本均值是该节点全部样本的均值。

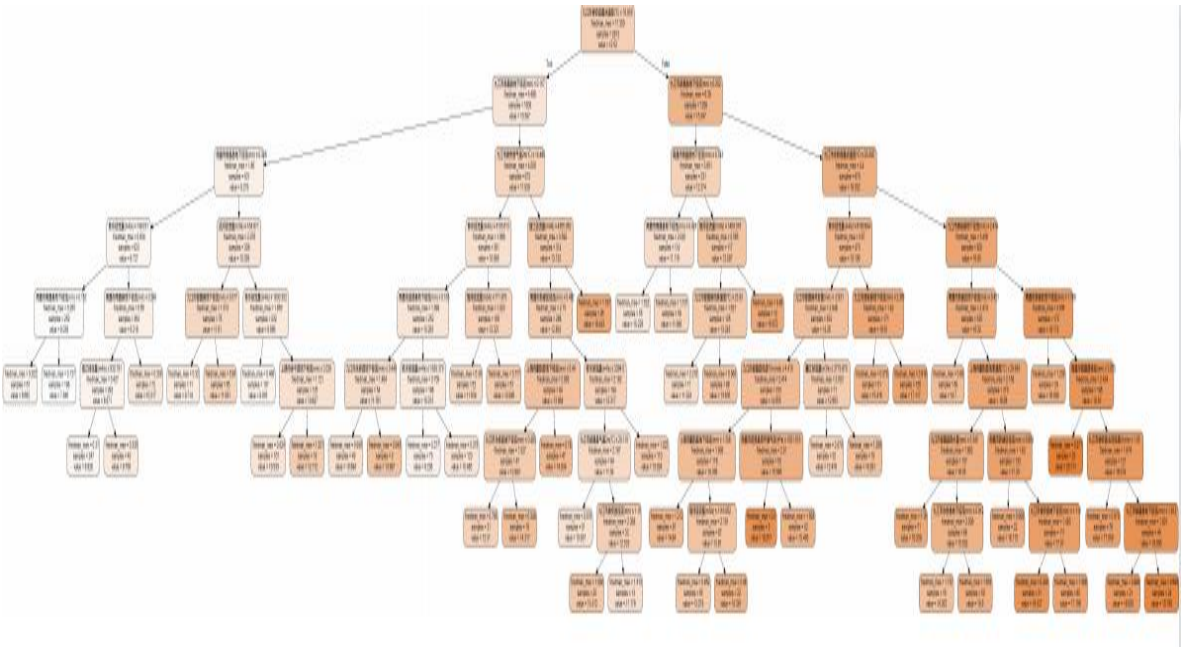

图 10 决策树结构

5.2.2 调用模型获取特征重要性

接下来，利用建立的决策树模型计算特征重要性。特征重要性表示在模型中各个特征对结果的贡献程度。下面的柱形图选取了特征重要性靠前的自变量。

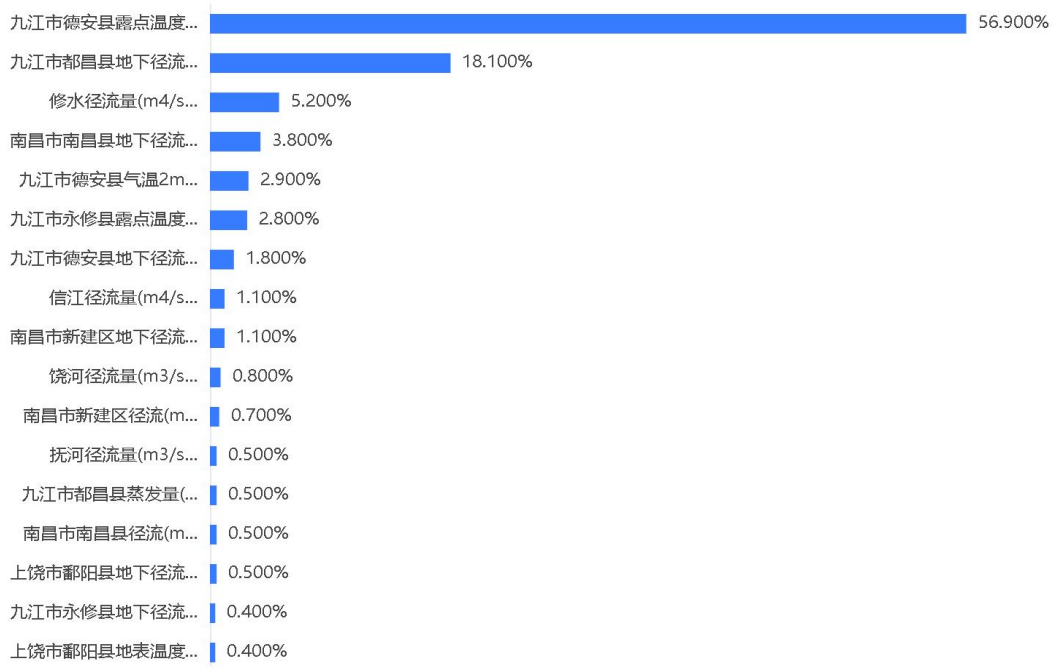

图 11　各特征（自变量）的重要性比例

5.2.3 分析模型评估结果

将建立的决策树回归模型应用于训练集和测试集数据，并得到模型的评估结果。评估结果包括各项指标如均方误差（MSE）、决定系数（RMSE）等，用于评价模型的拟合程度和预测能力。

下表中展示了交叉验证集、训练集和测试集的预测评价指标，通过量化指标来衡量决策树的预测效果。其中，通过交叉验证集的评价指标可以不断调整超参数，以得到可靠稳定的模型。

	MSE	RMSE	MAE	MAPE	R^2
训练集	0.76	0.872	0.653	5.236	0.933
交叉验证集	4.012	1.995	1.546	12.372	0.64
测试集	4.29	2.071	1.61	12.813	0.688

表 3 模型评估结果

根据模型评估结果，本文对训练集、交叉验证集和测试集的性能进行了分析。在训练集上，模型表现较好，其均方误差（MSE）为 0.76，均方根误差（RMSE）为 0.872，平均绝对误差（MAE）为 0.653，平均绝对百分比误差（MAPE）为 5.236%，

同时决定系数（R^2）为0.933。这表明模型在训练集上能够较好地拟合数据，预测结果与实际值较为接近。

然而，在交叉验证集和测试集上，模型的性能较差。在交叉验证集上，模型的MSE为4.012，RMSE为1.995，MAE为1.546，MAPE为12.372%，R^2为0.64。在测试集上，模型的MSE为4.29，RMSE为2.071，MAE为1.61，MAPE为12.813%，R^2为0.688。这表明模型在未见过的数据上的预测误差较大，且解释数据的能力较低。

根据这道题目的要求，需要分析并找出对鄱阳湖A站点水位变化影响较大的关键外部因素。因此，在这种情况下，更重要的是拟合能力而不是仅仅预测概率的高低。通过选择一个能够更好地拟合现有数据的模型，可以得到关键外部因素与鄱阳湖A站点水位变化之间的具体关系。这将有助于深入理解外部因素对水位变化的影响，并为进一步的分析和预测提供基础。

因此，在这道题目中，优先选择拟合能力更高的决策树回归模型会更为合适。得到的结果为对鄱阳湖A站点水位变化影响较大的关键外部因素是九江市德安县露点温度。

5.3 问题三模型建立与求解

5.3.1 模型的建立

首先，基于上一题模型得到的结果，根据领域知识和统计方法，选择对结果有影响的重要特征，对于那些对结果影响较小的数据基于统计指标、相关性分析、特征重要性评估等方法进行剔除。然后根据预处理的需求和模型的输入要求，将数据进行整理。下表展示了模型各项参数配置以及模型训练时长。

参数名	参数值
训练用时	3.222s
数据切分	0.9
数据洗牌	是
交叉验证	6

激活函数	identity
求解器	lbfgs
学习率	0.1
L2 正则项	1
迭代次数	1000
隐藏第 1 层神经元数量	100

表 4 模型参数

5.3.2 模型的评估

通过训练集数据来建立 bp 神经网络回归模型，根据得到的模型对 21 年水位数据进行预测，并与真实值相比较。

下面表 5 展示的是 bp 神经网络回归模型训练的结果。

	MSE	RMSE	MAE	MAPE	R^2
训练集	1.219	1.104	0.831	6.545	0.899
交叉验证集	1.601	1.265	0.935	7.341	0.867
测试集	1.529	1.236	0.904	7.058	0.865

表 5 测试数据训练结果

下表 6 展示的是模型预测效果通过这些指标，可以评估 bp 神经回归模型的预测性能。在这些结果中，测试集的表现最好，其次是训练集和交叉验证集。R^2 接近 0.71 表示该模型对数据的解释能力还不错。

	MSE	RMSE	MAE	MAPE	R^2
训练集	3.512	1.874	1.463	11.859	0.707
交叉验证集	3.556	1.885	1.469	11.888	0.703
测试集	3.505	1.872	1.471	11.617	0.71

表 6 模型预测效果

下表 7 展示的是在利用已有数据使用建立的 bp 神经回归模型来进行预测得到的结果。从评价指标的值可以看出，bp 神经网络回归模型的预测效果一般，

与第一题的模型相比，准确度相差较大。

评价指标	评价结果
MSE	2.585881722
RMSE	1.608067698
MAE	1.305487555
R^2	0.794054491
MAPE	11.19023591

表 7 对模型的预测和应用

5.3.3 模型效果分析

在多变量输入的条件下，神经网络模型预测没有单变量时间序列模型准确。可能是以下原因：

1.数据预处理不当：在使用神经网络模型进行预测之前，需要对数据进行预处理，例如归一化、平滑等操作。如果预处理不当，例如将训练集与测试集的数据进行不同的预处理，或者使用不同的预处理方法，可能会导致模型预测结果不准确。

2.模型参数选择不当：神经网络模型中有很多参数需要调整，例如网络层数、隐层节点数、学习率等，这些参数会影响模型的预测结果。如果参数选择不当，可能会导致模型预测不准确。

3.模型结构不合适：神经网络模型结构的设计也会影响模型的预测结果，例如隐层节点数、激活函数的选择等。如果模型结构不合适，可能会导致模型预测不准确。

4.数据之间的相关性：在多变量输入的条件下，不同变量之间的相关性可能会影响模型的预测效果。如果不同变量之间存在较大的相关性，可能会导致模型预测不准确。本题中，可能存在变量选取不当。

参考文献

[1]刘保东，宿洁，陈建良，数学建模基础教程[M]，北京：高等教育出版社，2015(9):279

[2] 周志华. 机器学习[M]. 清华大学出版社, 2016.

[3] 刘超,回归分析—方法、数据与 R 的应用,高等教育出版社,2019.

[4] 柯郑林. Lasso 及其相关方法在多元线性回归模型中的应用[D].北京交通大学,2011.

附录

（一）问题一

未优化模型前的 ARMA 模型代码：

```
clc;clear;close all
data= xlsread('附件 1-鄱阳湖 A 水文站逐日水位数据.xlsx');
% 取出站点 A 过去 10 年的历史水位数据
x=data(1:3653);
train_data=data(1:3653);
test_data=data(3654:end)';%测试数据，用于预测对比
%%  看数据是否平稳，不平稳进行差分处理
figure(1)
plot(train_data,'LineWidth',1)
pinwen=adftest(train_data); %1 代表平稳，0 代表不平稳
%非平稳进行差分处理
xlabel('day')
ylabel('water_level')
if pinwen==1
    disp('平稳序列')
    train_data1=train_data;
else
    disp('不平稳序列')
    figure(2)
    train_data1=diff(train_data);
    %train_data1=train_data;
    plot(train_data1,'LineWidth',1)
end
%% 利用自相关图和偏相关图判断模型类型和阶次
figure(3)
autocorr(train_data1) %绘制自相关函数
[ACF,Lags,Bounds]=autocorr(train_data);
figure(4)
```

```
parcorr(train_data1) %绘制偏相关函数
%% 自相关和偏相关函数难以判断时可以用 AIC 准则求出最好阶数
%确定阶数的上限
lim=round(length(train_data)/10); %数据总长度的 1/10
if lim>=10
    lim=10;%如果数据太长了，就限定阶数
end
train_data=train_data';
train_data2=iddata(train_data');
save_data=[];
for p=1:lim
    for q=1:lim
        num=armax(train_data2,[p,q]);  %armax 对应 FPE 最小
        AIC=aic(num);
        save_data=[save_data;p q AIC];
        reli_juzheng(p,q)=AIC;
    end
end
%AIC 越小越好
%% 绘制阶数热力图
figure(5)
for i=1:lim
    y_index(1,i)={['AR' ,num2str(i)]};
    x_index(1,i)={['MA' ,num2str(i)]};
end
H = heatmap(x_index,y_index, reli_juzheng, 'FontSize',12, 'FontName','宋体');
H.Title = 'AIC 定阶热力图';
figure(6)
%% 利用阶数得到模型
min_index=find(save_data(:,3)==min(save_data(:,3)));
p_best=save_data(min_index,1);  %p 的最优阶数
q_best=save_data(min_index,2);  %q 的最优阶数
model=armax(train_data2,[p_best,q_best])
%% 利用模型预测
L=length(test_data);
train_data1=train_data1';
pre_data=[train_data1';zeros(L,1)];
pre_data1=iddata(pre_data);
pre_data2=predict(model,pre_data1,L);
pre_data3=get(pre_data2);%得到结构体
pre_data4=pre_data3.OutputData{1,1}(length(train_data1)+1:length(train_data1)+L);%从
结构体里面得到数据
%显示全部
data=[train_data1';pre_data4];%全部的差分值
```

```
if pinwen==0 %非平稳时进行差分还原
    data_pre1=cumsum([train_data(1);data]);%还原差分值
elseif pinwen==1
    data_pre1=data;
end
data_pre2=data_pre1(length(train_data)+1:end);%最终预测值
subplot(2,1,1)
plot(1:length(train_data),train_data,'--','LineWidth',1)
hold on
plot(length(train_data)+1:length(train_data)+L,test_data,'--','LineWidth',1.5)
hold on
plot(length(train_data)+1:length(train_data)+L,data_pre2,'--','LineWidth',1.5)
hold on
xlabel('time')
ylabel('water_leveldata')
legend('真实值','后几个周期真实值','后几个周期预测值')
wucha=sum(abs(data_pre2'-test_data)./test_data)./length(data_pre2);
title_str=['ARMA 法','  后几个周期预测相对误差为：',num2str(wucha)];
title(title_str)
subplot(2,1,2)
plot(1:L,test_data,'--o','LineWidth',1.5)
hold on
plot(1:L,data_pre2,'--*','LineWidth',1.5)
hold on
xlabel('time')
ylabel('water_leveldata')
legend('后几个周期真实值','后几个周期预测值')
title_str=['ARMA 法','  后几个周期预测相对误差为：',num2str(wucha)];
title(title_str)
```

将历史水位数据拆分为长期趋势、季节变动和随机波动三个成分代码：

```
# -*- coding: utf-8 -*-
import matplotlib
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.api as sm

# 读取 CSV 文件
data = pd.read_csv('附件 1-鄱阳湖 A 水文站逐日水位数据.csv', parse_dates=['时间'],
index_col='时间', encoding='gbk')

# 进行时间序列分解
result = sm.tsa.seasonal_decompose(data['数据'], model='additive')
```

```
# 设置全局字体
matplotlib.rcParams['font.sans-serif'] = 'Helvetica'
title_font = {'fontname': 'SimHei'}

# 绘制原始数据
plt.subplot(4, 1, 1)
plt.plot(data['数据'])
plt.title('原始数据', **title_font)

# 绘制长期趋势
plt.subplot(4, 1, 2)
plt.plot(result.trend)
plt.title('长期趋势', **title_font)

# 绘制季节变动
plt.subplot(4, 1, 3)
plt.plot(result.seasonal)
plt.title('季节变动', **title_font)

# 绘制随机波动
plt.subplot(4, 1, 4)
plt.plot(result.resid)
plt.title('随机波动', **title_font)

plt.tight_layout()
plt.show()
```

优化模型后时间序列分解模型代码：

```
clc;clear
data= xlsread('附件 1-鄱阳湖 A 水文站逐日水位数据.xlsx');
% 取出站点 A 过去 10 年的历史水位数据
x=data(1:3653);
train_data=data(1:3653)';
test_data=data(3654:end)';%测试数据，用于预测对比
N=365;    %周期
for t=1:(length(train_data)-N+1)
    MA(t)=sum(train_data(t:(t+N-1)))/N;
end
SI=100*train_data((N-1):end-1)./MA;
%用各年同季平均，去掉 SI 的随机性，得到季节指数 r，并修正季节指数 R
N_list=[N-1:N,1:N-2];%SI 是从 N-1 开始的，循环一个周期
for i=1:length([N-1:N,1:N-2])
```

```
    N1=N_list(i);
    r(i)=mean(SI(N1:N:end));
end
R=r./mean(r);
%拟合得到长期趋势 T
x=1:length(train_data);
p=polyfit(x,train_data,2); %用二次方程拟合
T=polyval(p,x);%长期趋势
% 计算循环变动 C
C=MA./T((N-1):end-1);
%预测后面一个季度的数据
N_p=1;%预测后两季度
T_n=length(train_data);
T_p=(T_n+1):(T_n+N_p*N);%预测后面两个周期
T_p1=polyval(p,T_p);  %长期趋势
C1=mean(C);
X_p=T_p1.*C1.*repmat(R,1,N_p);
subplot(2,1,1)
plot(train_data)
xlabel('time')
ylabel('water_leveldata')
hold on
plot(T,'--','LineWidth',1)%长期趋势
hold on
plot(T_p,test_data,'--','LineWidth',0.5)
hold on
plot(T_p,X_p,'--','LineWidth',1)
hold on
legend('真实值','长期趋势','后几个周期真实值','后几个周期预测值')
wucha=sum(abs(X_p-test_data)./test_data)./length(X_p);
title_str=['时间序列分解法','  后几个周期预测相对误差为：',num2str(wucha)];
title(title_str)
subplot(2,1,2)
plot(T_p,test_data,'--o','LineWidth',0.5)
hold on
plot(T_p,X_p,'--*','LineWidth',1)
hold on
xlabel('time')
ylabel('water_leveldata')
legend('后几个周期真实值','后几个周期预测值')
wucha=sum(abs(X_p-test_data)./test_data)./length(X_p);
title_str=['时间序列分解法','  后几个周期预测相对误差为：',num2str(wucha)];
title(title_str)
```

（二）问题二

```
#在 spsspro 软件中导入数据进模型生成结果代码
import numpy
import pandas
from spsspro.algorithm import supervised_learning
#生成案例数据
data_x = pandas.DataFrame({
    "A": numpy.random.random(size=100),
    "B": numpy.random.random(size=100)
})
data_y = pandas.Series(data=numpy.random.random(size=100), name="C")
#决策树回归
result = supervised_learning.decision_tree_regression(data_x=data_x, data_y=data_y)
print(result)
```

（三）问题三

```
#在 spsspro 软件中导入数据进模型生成结果代码
import numpy
import pandas
from spsspro.algorithm import supervised_learning
#生成案例数据
data_x = pandas.DataFrame({
    "A": numpy.random.random(size=100),
    "B": numpy.random.random(size=100)
})
data_y = pandas.Series(data=numpy.random.choice([1, 2], size=100), name="C")
result = supervised_learning.mlp_regression(data_x=data_x, data_y=data_y)
print(result)
```