第三届“泰迪杯”

全国大学生数据挖掘竞赛

优

秀

作

品

作品名称：城市供水处理混凝投药过程的建模与控制

荣获奖项：一等奖

作品单位：华南师范大学

作品成员：杨程炜　胡小粤　许仕达

指导教师：杨坦

基于改进的 BP 神经网络的最佳混凝剂投药量模型

摘　要：本文首先通过插值变换样本的时间窗口，再构造反映混凝剂净水效果的指标，并通过该指标与 PAC 投入量在不同的滞后阶数下的相关系数，得到原水添加混凝剂反应到沉淀结束出水的时间约为 124 分钟。再使用改进模型——基于有限次残差拟合的 BP 神经网络组合模型，学习净水规律，即在不同的原水水质和原水流量下，要想得到一定的出水浊度所需的混凝剂投入量。一方面，模型的绝对误差百分比为 18.84%，说明总体的预测效果一般，另一方面，证实改进模型确实比经典 BP 神经网络更有效，预测精度提高了 16.89%。然后，利用上述模型生成当前状况下的去浊率曲线，并取最高效率点为最优点。最后,引入温度作为自变量，重新建立 BP 神经网络组合模型，通过前后对比，发现预测误差减少 10.31%，在残差拟合 4 次后预测误差降低 7.92%，说明温度变量对预测有效，且通过分析，温度对 PAC 投入量的影响是非线性的。

关键词：混凝剂　BP 神经网络　去浊率　最优点

Model of the Best Amount of Coagulant Delivery Based on Improved BP Neural Network

Abstract: In this paper, first, we fabricate a interpolation to transform the window of sample,and construct a index that reflect the effect of coagulant purifying water.Next,we gain the correlation between the index and PAC investment when lag coefficient is in different, in order to receive the length of time between raw water added coagulant and outputting water is 124 minutes.After that,there is a improved model,Combined model of BP neural network based of limited error fitting,and we make use of that model to learn the orderliness of water purification,which can gain the enough coagulant when the water quality of raw water and rate of flow of raw water is realized different.On the one hand,the absolute error percentage of this model is equal to 18.84%,which can emerge this model has normal effect of forecast.On the other hand,modified model that raised the effect of forecast by 16.89% is more effective than classical BP neural network.Then,using that model to generate curve of turbidity rate under current conditions,and choice the point that represent achieving maximum efficiency as the optimal point.Ultimately,the model is introduced temperature to be one of independent variable to fabricate a new expanded BP neural network model,after we observe the contrast of the above models finding that the prediction error has reduced by 10.31% and when error fitting has been done by 4 times the prediction error has reduced by 7.92% ,which could show that temperature variable is effective to prediction and nonlinear to PAC investment.

Key words: coagulant Back Propagation turbidity rate optimal point

目　录

1. 研究目标

本次建模目标是利用 2013.08.08～2014.09.05 的 9397 条数据，利用数据挖掘，计算出添加混凝剂反应到沉淀结束出水所需的时间，并根据该反应时间（即控制时滞），利用原水水质数据、流量数据、沉淀池浊度及混凝剂投加量等数据建立最佳混凝剂投药量的模型，并通过引入温度变量，再追加温度作为影响因素，建立更完善的最佳混凝剂投药量模型，实现对污水处理自动化的实时控制，从而增强水厂净水的高效性和便利性。

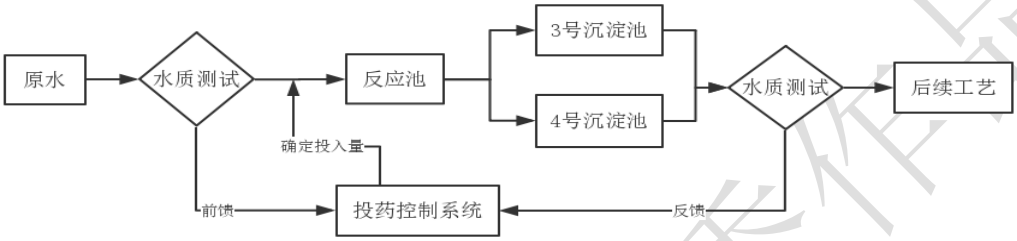

图 1　净水总过程图

2. 分析方法与过程

2.1. 总体流程

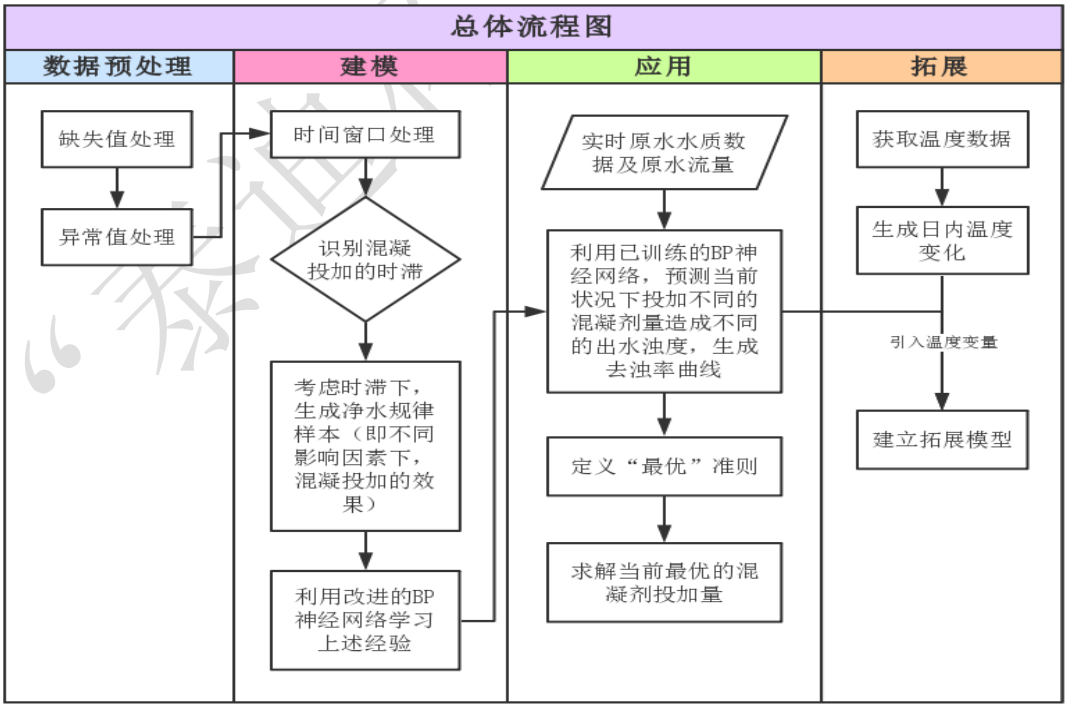

图 2　研究过程图

[注]：本文在求出控制时滞后，跳过题目要求的需求二，直接添加沉淀池的浊度作为自变量开始建模，原因如下：

如果没有添加沉淀池的浊度作为自变量，那么只通过原水 PH、原水浊度、取水量预测 PAC 投放量，学习到的是：过去在不同原水状况下净水厂不同的混凝剂投放量，得到的只是过去投放经验的总结.

而如果增加沉淀池的浊度作为自变量，那么通过原水状况、出水时的浊度预测 PAC 投放量，则模型学习到的是：当前原水状况下，要使得未来出水时达到一定浊度所需要的 PAC 投放量，则学习到的是客观的净水规律。

因此，我们直接跳过题目的需求二，直接引入沉淀池浊度变量。

本用例主要包括如下步骤：

步骤一：样本数据预处理

步骤二：计算原水添加混凝剂反应到沉淀结束出水的时间

步骤三：建立原水水质、取水量、沉淀池浊度、混凝剂投加量之间的数学模型

步骤四：定义混凝剂最优投入量的含义

步骤五：求解当前最优的混凝剂投入量

步骤六： 获取并生成温度数据

步骤七： 建立含温度变量的拓展模型

2.2. 具体步骤

步骤一：样本数据预处理

✧ 缺失值处理

在原始计量数据，发现前 288 条数据存在大量的样本 PAC 消耗量缺失的现象（其中仅有 2 条含相应记录，但依然无法进行有效的插值处理，因此忽略），为确保建模数据的有效性，删除上述 288 条数据，占原始数据的 3.06%。

✧ 异常值处理

在取水量和供水量的折线图中，发现 2014 年 3 月 28 日 16 点数据异常大（且 15 点数据缺失），2014 年 5 月 27 日 15 点数据异常大（且 14 点数据缺失），2014 年 6 月 20 日 18 点数据异常大（且 7 点-17 点数据缺失），2014 年 9 月 4 日 13 点的取水量为负值，显然不符合实际，为了保持时间的连贯性，进行线性插值，使总数据增加到 9156

条（即共有 381.5 天），增加的数据占总数据的 0.51%。

步骤二：计算原水添加混凝剂反应到沉淀结束出水的时间

✧ 时间窗口变换（插值处理）

原始数据样本两两间隔为一个小时，相差较大，不利于较精确地进行分析和计算，对数据进行线性插值，得到以一分钟为间隔的数据。

✧ 净水效果指标的构造

(1) 浊度的理解

1NTU 溶液的浑浊程度与悬浮物及胶体状态颗粒为 1mg/L 的溶液的浑浊程度是等价的，因此，给定数据中的浊度可理解为悬浮物及胶体状态颗粒的浓度(mg/L)。

(2)指标构造

通过浊度在净水沉淀前后的变化量，作为体现 PAC 净水效果的指标，用 Q_t 表示净化效果，D_{t}、W_{t} 分别表示 t 时刻（分钟）的原水浊度、取水量，$D_{\mathrm{t+n}}'$、$W_{\mathrm{t+n}}'$ 代表净化后出水时刻 $\mathrm{t+n}$（分钟）的沉淀池浊度、供水量，考虑到净水过程中水的损耗，净化后的浊度需要一定的折算，表达式如下：

$$Q_t = D_{\mathrm{t}} - D_{\mathrm{t+n}}' \times \frac{W_{\mathrm{t+n}}'}{W_t}$$

✧ 净水效果与 PAC 投入的相关系数

PAC 的投入量与净水效果有着直接的关系，因此计算两者之间的相关系数 $\rho(Q,Pac)$，通过观察相关系数何时取得最大值，确定为控制时滞。并绘制如下曲线图：

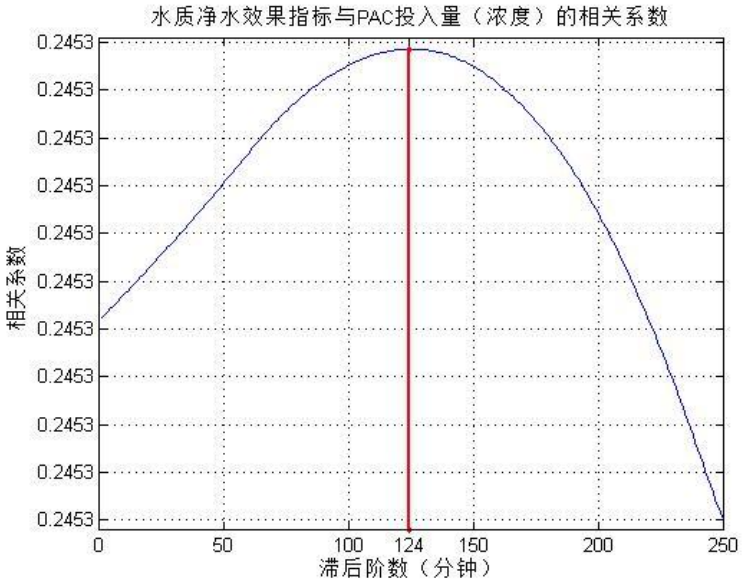

图 3　净水效果与PAC投入量相关系数及滞后阶数图

由图 3 看出，总体的相关系数并不大，主要原因是因为以分钟为时间窗口的数据是由插值而来的，数据品质较为一般。

发现当滞后阶数为 124 时，水质浊度下降量与 PAC 投入量的相关系数达到最大值，即确定从原水添加混凝剂反应到沉淀结束出水的时间为 124 分钟。

步骤三：建立原水水质、原水流量、混凝剂投加量之间的数学模型

✧ 改良模型：基于有限次残差拟合的 BP 神经网络组合模型

（1） 提出背景

经典 BP 神经网络初始权值阈值由于是随机选取的，因此容易陷入局部极小值，而诸如融合遗传算法的 BP 神经网络等模型虽有效提高预测精度，但运算慢是其无法忽视的缺点，尤其对于本题，插值后的数据以分钟为时间窗口，有几十万个样本，在一般的微型计算机中训练一次需时数分钟。而遗传算法中，每一个种群就是一次训练，而且还需几十、数百代的遗传，因此，效率十分低，不利于本题的情况，需要进行适当改进。

（2） 原理简介

一方面，若训练陷入局部极小值，则相比于陷入全局最优解的情况，其拟合残差仍含有可预测的信息。另一方面，如果能充分利用残差中的信息，其预测精度甚至会比陷入全局最优解时更好。原因如下，对于 BP 神经网络，一旦网络结构、激励函数被确定，那么其拟合的函数形式就被确定下来，即使由于函数形式因包含激励函数而具有较强的非线性映射能力.但仍无法完全避免设定偏误，因此残差往往隐含可用于预测输出值的信息。

因此，反复构建新的 BP 神经网络，对残差进行学习、拟合，达到充分利用隐含在残差的有价值的信息的目的。

（3） 组合模型的建立过程

首先以 $X=(X_1,X_2,...,X_n)^T$ 作为输入变量、$Y=(Y_1,Y_2,...,Y_m)^T$ 作为预测变量建立 BP 神经网络，其预测输出为 $O=(O_1,O_2,...,O_m)^T$，预测残差为 $e^{(1)}=(e_1^{(1)},e_2^{(1)},...,e_m^{(1)})^T$；以 (X,O) 作为输入变量、以 $e^{(1)}$ 作为预测变量构建第二个 BP 神经网络，对第一个神经网络的残差进行预测.其预测输出记为 $\hat{e}^{(1)}$，其残差为 $e^{(2)}=e^{(1)}-\hat{e}^{(1)}$；再以 $(X,O,\hat{e}^{(1)})$ 作为输入变量、以 $e_k^{(2)}$ 作为预测变量构建第三个 BP 神经网络,对第二个 BP 神经网络的残差

进行预测，如此反复（如图 4）.假定对残差进行 N 次拟合.则输出节点 Y_k 最终的预测结果为：

$$\hat{Y}_k = O_k + \hat{e}_k^{(1)} + \hat{e}_k^{(2)} + \ldots + \hat{e}_k^{(N)} \qquad (k = 1,2,\ldots,m)$$

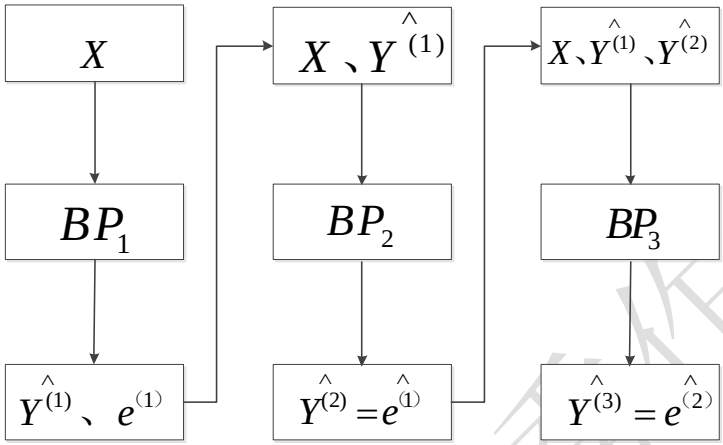

图 4　组合模型示意图

✧　建立改进模型

取 PH、原水浊度、沉淀池浊度、取水量作为自变量，PAC 投入量作为因变量，使用改进模型——基于有限次残差拟合的 BP 神经网络组合模型，得到原水水质、PH、原水流量、沉淀池浊度、混凝剂投入量之间的数学模型。

其中每一个 BP 神经网络的隐层神经元个数 N 利用以下经验公式得到：

$$N = 2I + 1$$

I 表示输入神经元的个数。

✧　模型诊断及评估

以上模型建立后，我们将改良模型与经典 BP 神经网络进行对比，结果如图 5。

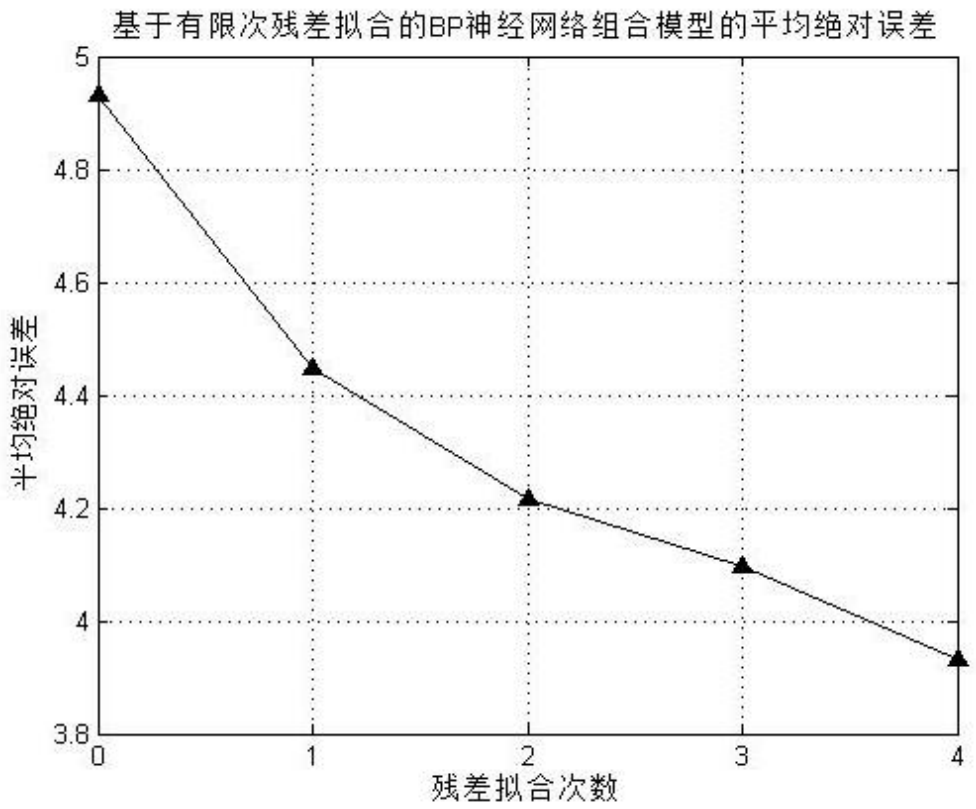

图 5　平均绝对误差折线图

残差拟合次数	0	1	2	3	4
平均绝对误差	4.931181	4.447976	4.215024	4.098183	3.931073
绝对误差百分比	49.54	39.43	30.73	22.50	18.84

表 1　平均绝对误差表

基于有限次残差拟合的 BP 神经网络组合模型（残差拟合次数为 0）的平均绝对误差百分比为 49.54%，而残差拟合 4 次以后的组合模型的平均绝对误差百分比为 18.84%，证明上述改进有效而且必要。

模型精度受制于以下两个方面——

① 在实际操作中，PAC 是以小时为单位进行投放的，而在数据处理过程中采用插值的方式获得连续型数据，一定程度上影响了数据精度；

② 由于水流流速、沉淀物沉积等因素，导致每单位时间的沉淀池浊度会受到前后时刻的影响，从而影响了建模结果。

③ PAC 的混凝效果仍可能受更多的因素影响，如温度、微生物等，而没有纳入上述模型。

④ PAC 的混凝过程是复杂的物理、化学过程，因此系统较为复杂。

基于上述因素，我们认为上述建模精度是可以接受的。

利用残差拟合次数为 4 的组合模型，其残差的频率直方图如下，发现预测误差主要集中在[-5,5]之间，占了预测样本的 77.21%。另外，求得预测误差的平均值为 -0.0932，比较接近 0。

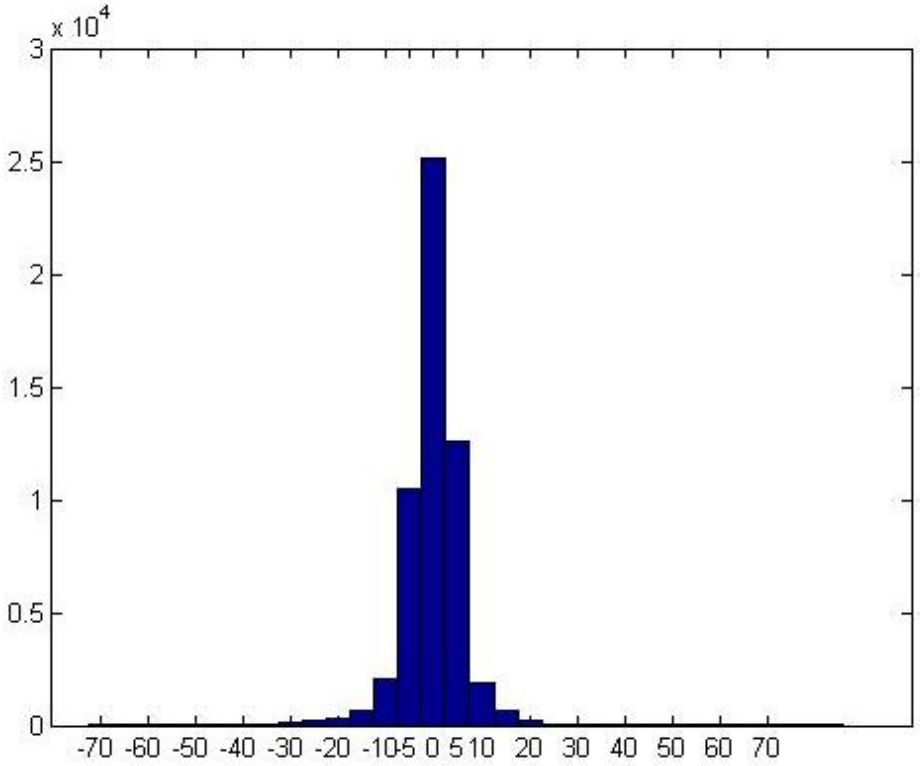

图 6　残差频率直方图

步骤四：定义混凝剂最优投入量的含义

✧　最低成本点

最低成本点即在净水处理中，取沉淀池出水浊度要求上限相适应的 PAC 投入量为混凝剂最佳投入量，优点为控制点明确，在符合出水的合格要求下成本达到最低，不足的是效率未达到最高值。

✧　最高经济点（效率最高）

最高经济点同时考虑出水浊度的合格要求和混凝剂的净水效率，因为一般情况下浊度净化率并不随着混凝剂投入量的增加而等量提高，会在某些区间混凝剂的投入量增加到某个点时，浊度净化率出现明显变化。一定区间内，混凝剂净化率的增加在某点之后随着投入量增加出现明显减少，称该点为最高经济点，在该点混凝剂的效率达到最高（即

数学上的函数拐点）。

本文认为，取最高经济点对应的混凝剂投入量为混凝剂最佳投入量，优点为充分利用了混凝剂的作用，使效率达到最大，不足的是会适当增加成本。

步骤五：求解当前最优的混凝剂投入量

在本文中，选取最高经济点（效率最高）作为最优的定义，使用已建立的神经网络模型，求得在符合出水合格要求的条件下，PAC 投入量与去浊率的关系，绘制直观图像如下：（以 PH=6.8，原水浊度=100NTU，取水量为 9000L/h 为例）

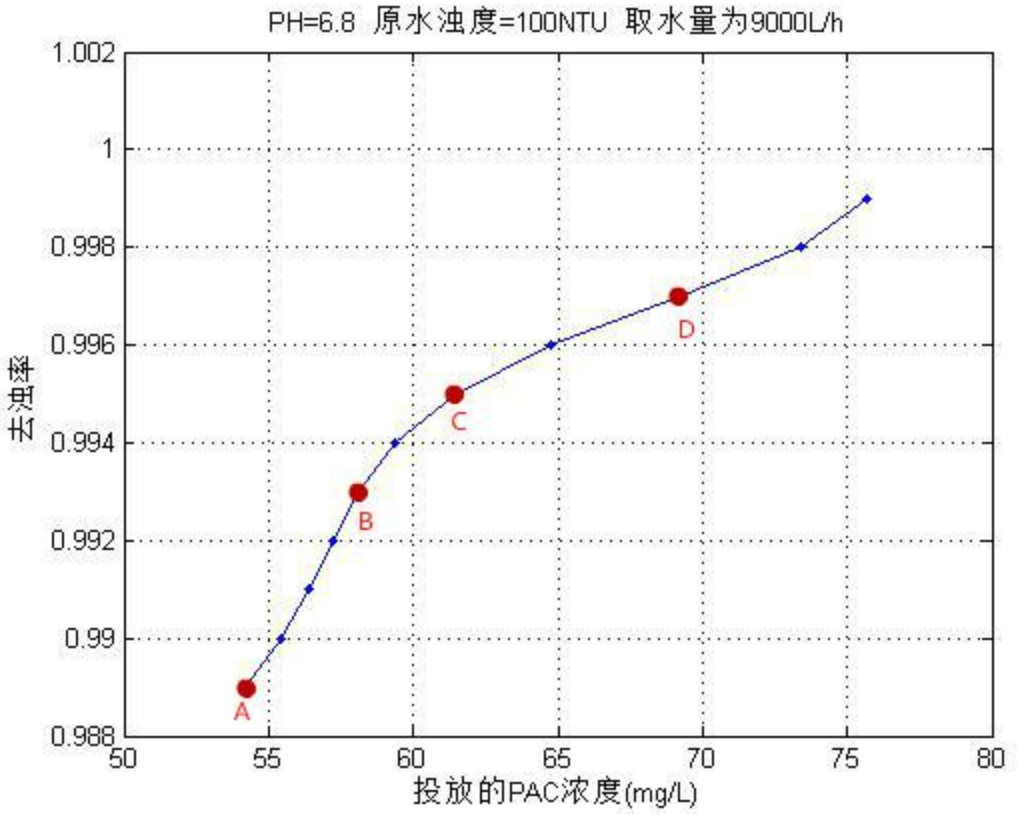

图 7　去浊度与 PAC 投入量示意图

此图中，A 点为满足出水浊度要求的对应的 PAC 最少投入量，而 B 点和 C 点均为拐点，而一般取第一个拐点作为最优经济点。

由于原始样本中沉淀池浊度最低为 0.35，对应此时的去浊率约为 95.8%（此时为 D 点），去浊率高于 95.8%的情况由于缺少样本的学习，因此上图中 D 点之后的线段不具有足够的可信度，在此忽略。因此根据经济原则，选取 B 点的投入量作为最佳混凝剂投药量。

步骤六： 获取并生成温度数据

✧ 获取每日温度数据

根据相关信息，我们决定选取深圳的气温数据作为温度数据，通过深圳气象局官方网站获取到 2013 年 8 月 20 日至 2014 年 9 月 5 日间的每日最高气温及最低气温。

✧ 生成对应时间窗口的实时温度

模型中的数据的时间窗口已通过线性插值改变为分钟，通过网络获取的温度数据仅有每日最低值和最高值，不利于引入，且根据资料可知每日气温变化具有一定的规律，本文中我们假设每日最低气温在 5 点取到，最高气温在 18 点取到，然后对气温数据进行三次样条插值，得到以一分钟为间隔的数据，任取三天的温度数据插值结果绘制示意图，如下：

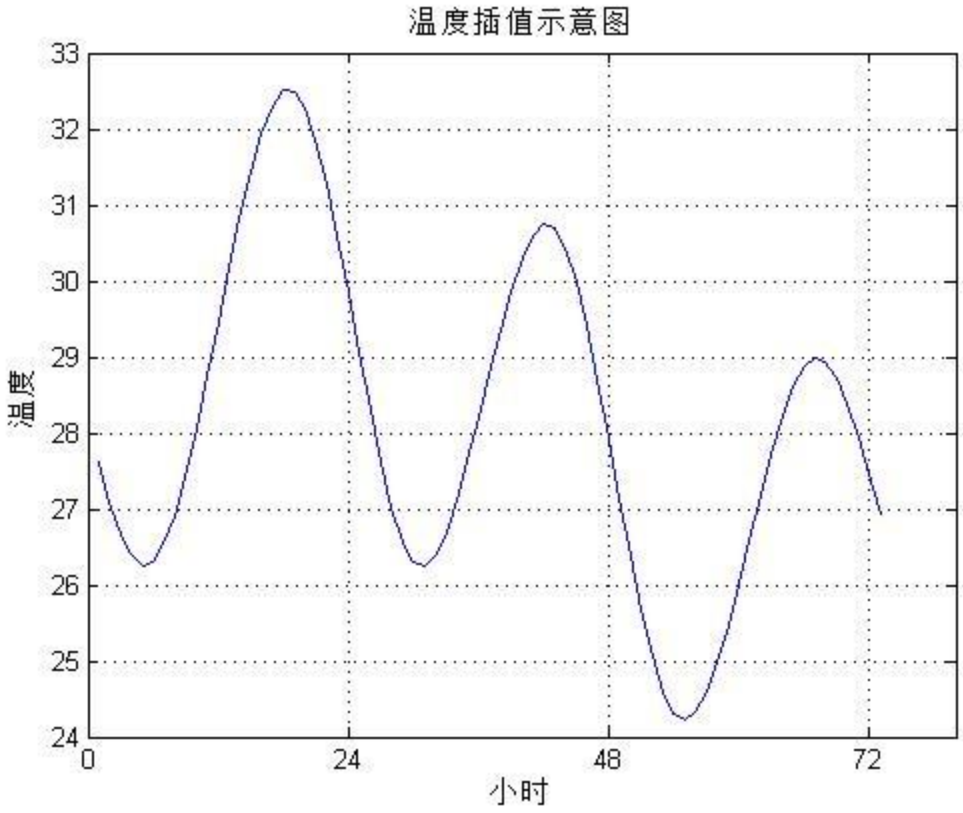

图 8 温度插值示意图

步骤七： 建立含温度变量的拓展模型

引入温度变量后，重新建立新的基于有限次残差拟合的 BP 神经网络组合模型，将其在预测样本上的平均绝对误差与没有引入温度变量时的误差进行对比，结果如图 9：

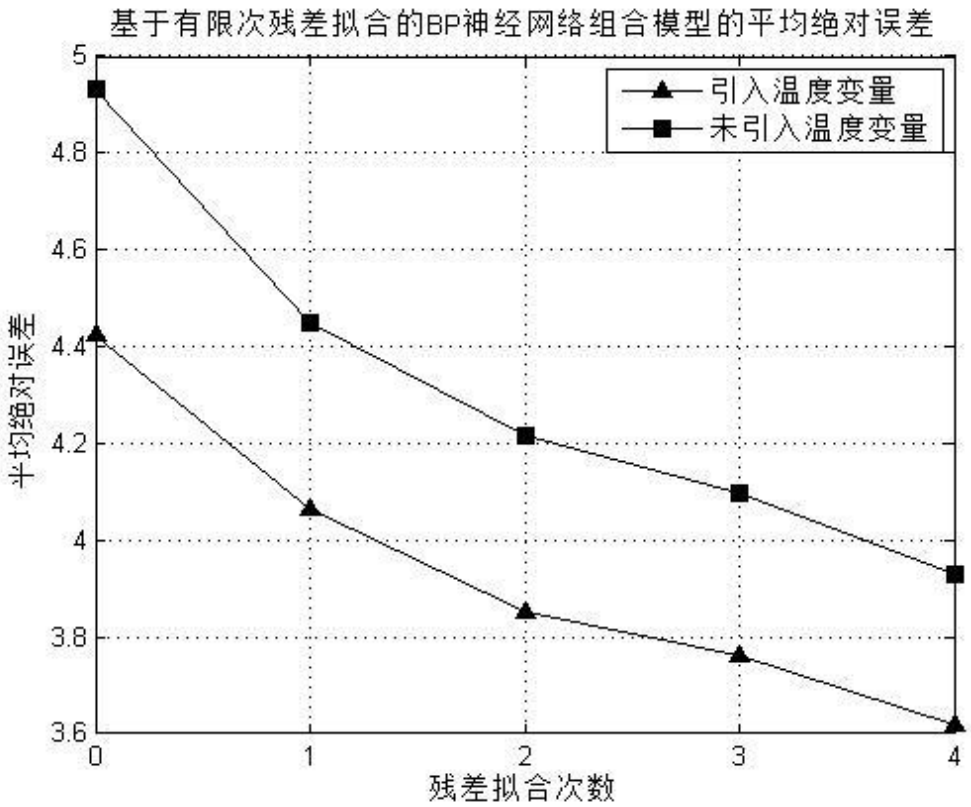

图 9　　引入温度变量的预测误差对比图

在还没有对残差进行拟合时，没有引入温度变量的模型误差为 4.931，引入温度为误差为 4.423，预测误差减少 10.31%；在残差拟合 4 次后，没有引入温度变量的模型平均绝对误差为 3.931，引入温度变量后为 3.6197，预测误差降低了 7.92%。

总的来说，在没有引入温度变量的模型，即使不断对残差进行拟合，充分学习残差中的残余信息，其误差仍然不能收敛于引入温度变量的模型的误差，因此可认为温度变量在预测 PAC 投放量与去浊效果的模型中是有用信息。

另外，由于上述温度数据是经过一定的假设与插值得到的，在不同的天气状况下，其当日最低、最高温度可能会发现偏离。例如，多云天气可能使最高气温点提早出现，在不同季节，受太阳直射点偏移的影响，最低、最高气温出现的时刻也会不同，因此，上述温度数据的品质属一般，因此，其预测误差降低相对 7.92%仍然是可以证明温度变量是有效的。

2.3. 结果分析

引入温度变量后，尝试在一定的条件下（PH=6.8，原水浊度=100NTU，取水量为 9000L/h）下，在不同的温度下探究去浊率曲线，取温度分别为 5 摄氏度、10 摄氏度、

15 摄氏度、20 摄氏度。结果如图下：

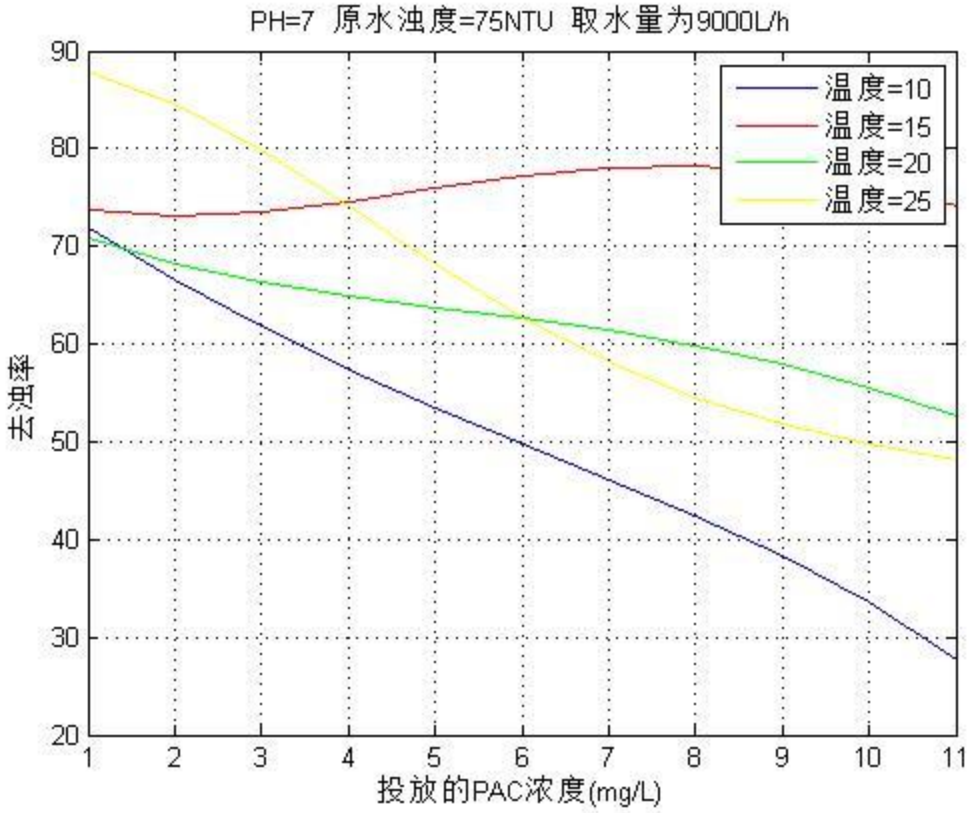

图 9　温度对去净水效果影响的示意图 1

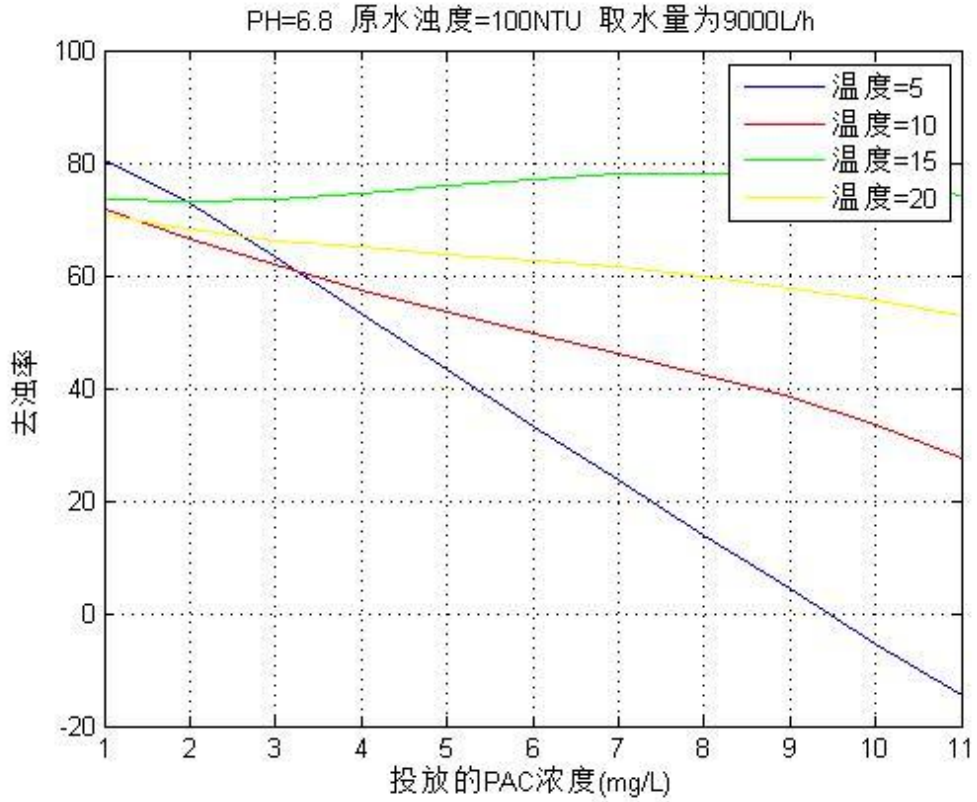

图 10　温度对去净水效果影响的示意图 2

观察上面两图，可得以下结论：

（1） 在不同的温度，随着投放 PAC 浓度的增加，去浊率的增减性是不同的。而且，在部分适宜温度下，去浊率会随 PAC 浓度的增加而增加，而相比适宜温度，过低、过高的温度都会使得去浊率随 PAC 浓度的增加而减少，说明温度对净水效果的影响是非线性的。

（2） 同样在 PH=6.8，原水浊度=100NTU，取水量为 9000L/h 情况下，在没有引入温度前，去浊率曲线没有出现极大值（图 7），而引入温度变量后，去浊率出现了极大值（图 10 红线），这在部分文献中称为效果最优点（净水效果达到最大）。一方面说明 PAC 浓度对净水效果的影响是非线性的，另一方面，说明温度对上述模型的预测效果起到显著作用。

（3） 对比上述两图，在不同的温度下，最经济点不一定存在。例如，可在图 9 情况，若当天温度为 25 摄氏度左右，则最经济点不存在，此时最低成本点是最好的选择。

（4） 承接（1）结论，相比适宜温度（使得去浊率随 PAC 浓度增加而增加的温度），过高过低的温度都会使得去浊率随 PAC 浓度增加而减少，而且，温度条件越偏离适宜温度，去浊率曲线下降得越快，即此时净水效果对 PAC 浓度更敏感。

3. 结论

本文构造了反映混凝剂净水效果的指标，并通过该指标与 PAC 投入量在不同的滞后阶数下的相关系数，得到原水添加混凝剂反应到沉淀结束出水的时间为 124 分钟。使用改良模型——基于有限次残差拟合的 BP 神经网络组合模型，学习在不同影响因素下的净水规律，把考虑原水水质、原水流量、沉淀池浊度作为自变量，预测混凝剂投入量，该模型的绝对误差百分比的平均值为 18.84%，效果一般，但预测精度比经典 BP 神经网络提高 16.89%。本文定义混凝剂的最佳投入量为最佳经济点，同时利用上述 BP 神经网络组合模型即可进行预测，最后增加温度变量作为自变量，重新建立的 BP 神经网络组合模型，通过前后对比，发现误差降低 10.31%，说明温度变量对模型的预测有作用，且温度对去浊效果的影响是复杂和非线性的。

4. 参考文献

[1]潘世英等. 混凝实验条件下混凝剂最佳投加量的选择方法研究[J].工业水处理，2011,31（10）：25-27

[2]胡茗,陈征. 前馈神经网络在水厂混凝投药中的应用[J]. 南昌大学学报（理科版），2010,34（1）：94-97

[3]甘艳珍.城市供水处理混凝投药过程的建模与控制研究[D].广州，2008

[4]白桦,李圭白.混凝投药智能控制系统实现方法的研讨[J].给水排水，2003,8（10）：81-83

[5]沈世锰. 神经网络系统理论及其应用[M].北京：科学出版社,1998

[6]李士勇. 模糊控制·神经控制和智能控制论[M].哈尔滨：哈尔滨工业大学出版社,1996

5. 附录

```
%%第一问：从290个数据开始，因为NULL
A=xlsread('C:\Users\Administrator\Desktop\泰迪数据挖掘竞赛\附件三_投药控制数据集
\Q1.xls');                                          %导入数据，变量命名
ph=A(:,1);
former=A(:,2);
b=A(:,3:4);
b_mean=mean(b,2);
in=A(:,5);
out=A(:,6)./60;
pac=A(:,7)./60;
N1=max(size(b))-1;
N2=N1.*60;
x=0:60:N2;
X=0:1:N2;                                           %x为原数据时间尺度，X为插值尺度
b_mean=mean(b,2);

Rho=[];                                                     %插值
Former=interp1(x,former,X,'linear');
B_mean=interp1(x,b_mean,X,'linear');
Pac=interp1(x,pac,X,'linear');
In=interp1(x,in,X,'linear');
Out=interp1(x,out,X,'linear');
for n=1:250
    %效果因子=原水浊度(NTU->mg/L)*进水-  出水浊度(NTU->mg/L)*出水
    effect=Former(1:end-n+1)-B_mean(n:end).*Out(n:end)./In(1:end-n+1);
    %effect=Former(1:end-n+1)-B_mean(n:end);
    %Pac_sum=Pac(1:end-n+1).*Out(1:end-n+1);
    Rho_Matrix=corrcoef(effect,Pac(1:end-n+1));
    %Rho_Matrix=corrcoef(effect,Pac_sum);
    rho=Rho_Matrix(1,2);
    Rho=[Rho;rho];
```

end

figure, %相关系数 & 滞后项数

plot(Rho(1:250)),xlabel('滞后阶数（分钟）'),ylabel('相关系数'),title('水质净水效果指标与PAC投入量（浓度）的相关系数')

grid on;

hold on,plot([124 124],[min(Rho)-(Rho(5)-Rho(1)) Rho(124)],'r.-','linewidth',2);

set(gca,'XTick',[0:50:100,124,150:50:250]);

axis([0 250 min(Rho)-(Rho(5)-Rho(1)) max(Rho)+(Rho(5)-Rho(1))])

```
%-----------------------第二问代码-------------------------
A=xlsread('C:\Users\Administrator\Desktop\泰迪数据挖掘竞赛\附件三_投药控制数据集\Q1.xls');
%导入数据，变量命名
ph=A(:,1);    former=A(:,2);   b=A(:,3:4);   b_mean=mean(b,2);
in=A(:,5);   out=A(:,6);   pac=A(:,7);
N1=max(size(b))-1;       N2=N1.*60;
x=0:60:N2;           X=0:1:N2;  %x为原数据时间尺度，X为插值尺度
b_mean=mean(b,2);
Rho=[];
%插值
Ph=interp1(x,ph,X,'linear').';
Former=interp1(x,former,X,'linear').';
B_mean=interp1(x,b_mean,X,'linear').';
In=interp1(x,in,X,'linear').';
Out=interp1(x,out,X,'linear').';
Pac=interp1(x,pac,X,'linear').';
%滞后n0=124,生成样本
n0=124;
input=[Ph(1:end-n0),Former(1:end-n0),B_mean(n0+1:end),In(1:end-n0)];
output=Pac(1:end-n0);
k=rand(1,max(size(output)));
[m,n]=sort(k);
%找出训练数据和预测数据 （旁置法）
n1=round(0.9*max(size(output)));%90%为作为训练样本与检验样本
input_train=input(n(1:n1),:)';
output_train=output(n(1:n1))';
input_test=input(n(n1+1:end),:)';
output_test=output(n(n1+1:end))';


%残差拟合N次
N=4;
N=N+1;
%不同阶的神经网络、归一化、预测输出（预测样本）、残差的保存
NET=cell(N,1);one=cell(N,2);Y=[];E=output_train;%Y是预测样本的预测值,Y_train是训练过程中训练样本拟合值,E是每次训练的目标值E(0)是训练输出,E(1)为第一次残差
for i=1:N
%训练样本输入输出数据归一化
[inputn,inputps]=mapminmax(input_train);
[outputn,outputps]=mapminmax(output_train);
one{i,1}=inputps;
```

```
one{i,2}=outputps;
% %初始化网络结构
[p,~]=size(inputn);
net=newff(inputn,outputn,2*p+1);
net.trainParam.epochs=50;  %最大迭代次数
net.trainParam.lr=0.1;
net.trainParam.goal=0;
%网络训练
net=train(net,inputn,outputn);
NET{i,1}=net;
%训练样本预测（用于求残差序列）
an_train=sim(net,inputn);
BPoutput_train=mapminmax('reverse',an_train,outputps);
e=output_train-BPoutput_train;
E=[E;e];

%循环利用代码：
input_train=[input_train;BPoutput_train];
output_train=e;

end

%k阶预测模块(输入:input_test,output_test)
for j=1:N
%预测样本预测过程
inputn_test=mapminmax('apply',input_test,one{j,1});
an=sim(NET{j,1},inputn_test);
BPoutput=mapminmax('reverse',an,one{j,2});
Y=[Y;BPoutput];
input_test=[input_test;BPoutput];
end


%各阶神经网络总输出合并及最终残差、最终误差百分比
Ysum=[];error=[];abserror=[];abserrorpercent=[];
for k=1:N
    Ysum=[Ysum;sum(Y(1:k,:),1)];
    error=[error;output_test-Ysum(end,:)];
    %mse=[mse;error(end,:)*error(end,:).'./max(size(error))];
    abserror=[abserror;mean(abs(error(end,:)))];

abserrorpercent=[abserrorpercent;abs((output_test-Ysum(end,:))./output_test
)];
end

figure,
plot(0:N-1,abserror,'k-^','markerfacecolor','k');
title('基于有限次残差拟合的BP神经网络组合模型的平均绝对误差');
xlabel('残差拟合次数')
ylabel('平均绝对误差')
set(gca,'XTick',[0:1:N-1]);
grid on
%误差直方图
figure,
hist(error(end,:));
title('预测误差的频率直方图')
%平均绝对误差
abserror(end)
```

```
%绝对误差百分比的平均值
mean(abserrorpercent(:,[1:37532,37534:end]),2)



%%实验的例子(Sample):  PH=6.8, 原水浊度=100NTU  ，取水量为 9000L/h
%生成样本
figure,
x_sample=[6.8;100;9000];
b_sample=0.1:0.1:1.1;  %出水浊度
n_sample=max(size(b_sample));
X_sample=[ones(1,n_sample).*x_sample(1);ones(1,n_sample).*x_sample(2);b_sam
ple;ones(1,n_sample).*x_sample(3)];
%BP 网络预测
Y=[];
for j=1:N
X_sample_test=mapminmax('apply',X_sample,one{j,1});
an=sim(NET{j,1},X_sample_test);
BPoutput=mapminmax('reverse',an,one{j,2});
Y=[Y;BPoutput];
X_sample=[X_sample;BPoutput];
end
Y_final=sum(Y,1);
lv=(x_sample(1)-b_sample)./x_sample(1);
%plot(Y_final,b_sample);
plot(Y_final,lv,'b.-');
grid on
title('PH=6.8  原水浊度=100NTU  取水量为 9000L/h')
xlabel('投放的 PAC 浓度(mg/L)')
ylabel('去浊率')
hold on

%-----------------------第三问代码-------------------------
%%温度导入与插值
B=xlsread('C:\Users\Administrator\Desktop\泰迪数据挖掘竞赛\Q3\tempeture');
Thigh=B(:,1);Tlow=B(:,2);
NT=max(size(Thigh));
T0=[];% 未插值温度（最低、最高交错）
for i=1:NT
T0=[T0,Tlow(i),Thigh(i)];
end
t0=[];%low 为 5 点,high 为 18 点
for i=1:NT
t0=[t0,5+24*(i-1),18+24*(i-1)];
end
t1=1:24*NT;
T1=interp1(t0,T0,t1,'spline');
t1=t1(1:end-12);
T1=T1(1:end-12);
%温度插值示意图
plot(T1(24:24*4));
xlabel('小时');ylabel('温度')
set(gca,'XTick',[0:24:24*3]);
plot(T1(24:24*4));
xlabel('小时');ylabel('温度')
```

```
set(gca,'XTick',[0:24:24*3]);
grid on
title('温度插值示意图')
```

```
%%-------------------------------加入温度变量的模型
-------------------------------------
A=xlsread('C:\Users\Administrator\Desktop\泰迪数据挖掘竞赛\Q3\Q3-2.xls');
%导入数据，变量命名
ph=A(:,1);    former=A(:,2);   b=A(:,3:4);   b_mean=mean(b,2);
in=A(:,5);   out=A(:,6);   pac=A(:,7);  temp=A(:,8);
N1=max(size(b))-1;       N2=N1.*60;
x=0:60:N2;           X=0:1:N2;  %x 为原数据时间尺度，X 为插值尺度
b_mean=mean(b,2);
Rho=[];
%插值
Ph=interp1(x,ph,X,'linear').';
Former=interp1(x,former,X,'linear').';
B_mean=interp1(x,b_mean,X,'linear').';
In=interp1(x,in,X,'linear').';
Out=interp1(x,out,X,'linear').';
Pac=interp1(x,pac,X,'linear').';
Temp=interp1(x,temp,X,'linear').';
%滞后 n0=124,生成样本
n0=124;
input=[Ph(1:end-n0),Former(1:end-n0),B_mean(n0+1:end),In(1:end-n0),Temp(1:e
nd-n0)];
output=Pac(1:end-n0);
k=rand(1,max(size(output)));
[m,n]=sort(k);
%找出训练数据和预测数据 （旁置法）
n1=round(0.9*max(size(output)));%90%为作为训练样本与检验样本
input_train=input(n(1:n1),:)';
output_train=output(n(1:n1))';
input_test=input(n(n1+1:end),:)';
output_test=output(n(n1+1:end))';


%残差拟合 N 次
N=4;
N=N+1;
%不同阶的神经网络、归一化、预测输出（预测样本）、残差的保存
NET=cell(N,1);one=cell(N,2);Y=[];E=output_train;%Y 是预测样本的预测值,Y_train 是
训练过程中训练样本拟合值,E 是每次训练的目标值 E(0)是训练输出,E(1)为第一次残差
for i=1:N
%训练样本输入输出数据归一化
[inputn,inputps]=mapminmax(input_train);
[outputn,outputps]=mapminmax(output_train);
```

```
one{i,1}=inputps;
one{i,2}=outputps;
% %初始化网络结构
[p,~]=size(inputn);
net=newff(inputn,outputn,2*p+1);
net.trainParam.epochs=50;  %最大迭代次数
net.trainParam.lr=0.1;
net.trainParam.goal=0;
%网络训练
net=train(net,inputn,outputn);
NET{i,1}=net;
%训练样本预测（用于求残差序列）
an_train=sim(net,inputn);
BPoutput_train=mapminmax('reverse',an_train,outputps);
e=output_train-BPoutput_train;
E=[E;e];

%循环利用代码：
input_train=[input_train;BPoutput_train];
output_train=e;

end




%k阶预测模块(输入:input_test,output_test)
for j=1:N
%预测样本预测过程
inputn_test=mapminmax('apply',input_test,one{j,1});
an=sim(NET{j,1},inputn_test);
BPoutput=mapminmax('reverse',an,one{j,2});
Y=[Y;BPoutput];
input_test=[input_test;BPoutput];
end
%各阶神经网络总输出合并及最终残差、最终误差百分比
Ysum=[];error=[];abserror=[];abserrorpercent=[];
for k=1:N
    Ysum=[Ysum;sum(Y(1:k,:),1)];
    error=[error;output_test-Ysum(end,:)];
    %mse=[mse;error(end,:)*error(end,:).'./max(size(error))];
    abserror=[abserror;mean(abs(error(end,:)))];

abserrorpercent=[abserrorpercent;abs((output_test-Ysum(end,:))./output_test
)];
end

figure,
plot(0:N-1,abserror,'k-^','markerfacecolor','k');
title('基于有限次残差拟合的BP神经网络组合模型的平均绝对误差');
xlabel('残差拟合次数')
ylabel('平均绝对误差')
set(gca,'XTick',[0:1:N-1]);
grid on
e2=[4.931181,4.44797,4.21502,4.0981,3.9310];%第三问模型结果
hold on
plot(0:N-1,e2,'k-s','markerfacecolor','k');
```

```
legend('引入温度变量','未引入温度变量')

%误差直方图
figure,
hist(error(end,:),x);
title('预测误差的频率直方图')
%平均绝对误差
abserror(end)
%绝对误差百分比的平均值
mean(abserrorpercent(:,[1:37532,37534:end]),2)


(e2(1)-abserror(1))./e2(1)    %引入温度变量后误差减少的百分比
(e2(end)-abserror(end))./e2(end)    %引入温度变量后误差减少的百分比



%%实验的例子1(Sample):  PH=6.8, 原水浊度=100NTU  ，取水量为9000L/h,温度为
15,20,25,30
%生成样本
YY_final=[];
for ttemp=[15,20,25,30]
x_sample=[6.8;100;9000;ttemp];
b_sample=0.1:0.1:1.1;  %出水浊度
n_sample=max(size(b_sample));
X_sample=[ones(1,n_sample).*x_sample(1);ones(1,n_sample).*x_sample(2);b_sam
ple;ones(1,n_sample).*x_sample(3);ones(1,n_sample).*x_sample(4)];

%BP网络预测
Y=[];
for j=1:N
X_sample_test=mapminmax('apply',X_sample,one{j,1});
an=sim(NET{j,1},X_sample_test);
BPoutput=mapminmax('reverse',an,one{j,2});
Y=[Y;BPoutput];
X_sample=[X_sample;BPoutput];
end
Y_final=sum(Y,1);
YY_final=[YY_final;Y_final];
lv=(x_sample(2)-b_sample)./x_sample(2);
hold on
%plot(Y_final,b_sample);
%plot(Y_final,lv,'b-');
end
figure,
plot(YY_final(1,:))
grid on
hold on
title('PH=6.8  原水浊度=100NTU  取水量为9000L/h')
xlabel('投放的PAC浓度(mg/L)')
ylabel('去浊率')
hold on
legend('温度=15','温度=25','温度=35','温度=45');
plot(YY_final(2,:),'r')
plot(YY_final(3,:),'g')
plot(YY_final(4,:),'y')
```

```
%%实验的例子2(Sample):  PH=7, 原水浊度=75NTU  ，取水量为9000L/h,温度为15,20,25,30
%生成样本
YY_final=[];
for ttemp=[10,15,20,25]
x_sample=[6.8;100;9000;ttemp];
b_sample=0.1:0.1:1.1;  %出水浊度
n_sample=max(size(b_sample));
X_sample=[ones(1,n_sample).*x_sample(1);ones(1,n_sample).*x_sample(2);b_sam
ple;ones(1,n_sample).*x_sample(3);ones(1,n_sample).*x_sample(4)];

%BP网络预测
Y=[];
for j=1:N
X_sample_test=mapminmax('apply',X_sample,one{j,1});
an=sim(NET{j,1},X_sample_test);
BPoutput=mapminmax('reverse',an,one{j,2});
Y=[Y;BPoutput];
X_sample=[X_sample;BPoutput];
end
Y_final=sum(Y,1);
YY_final=[YY_final;Y_final];
lv=(x_sample(2)-b_sample)./x_sample(2);
hold on
%plot(Y_final,b_sample);
%plot(Y_final,lv,'b-');
end
figure,
plot(YY_final(1,:))
grid on
hold on
title('PH=7  原水浊度=75NTU  取水量为9000L/h')
xlabel('投放的PAC浓度(mg/L)')
ylabel('去浊率')
hold on
plot(YY_final(2,:),'r')
plot(YY_final(3,:),'g')
plot(YY_final(4,:),'y')
legend('温度=10','温度=15','温度=20','温度=25');
```