基于神经网络的污水处理厌氧工艺关键指标预测与预警

摘要

党的十九大以来，我国越来越重视环保问题，而其中污水处理作为保护环境的重要一步，越来越多的学者投身到污水处理的相关研究之中。随着数据挖掘和人工智能技术的不断发展，如何运用计算机技术提升相关工艺已经成为一个热点问题。因此，本文从3台厌氧反应器运行数据着手，利用数据挖掘与深度学习技术探索各数据间的关系，实现对关键指标的预测与预警。

针对问题一，考虑到原始数据有较多脏数据，本文主要先把原始数据进行离群点剔除、异常值删除以及归一化处理，然后把预处理后的数据应用到BP神经网络模型中进行拟合（回归），并通过使用Keras的Sequential模型来简化神经网络结构的构建，进而实现对出水COD和出水VFA的数值预测。

针对问题二，本文主要从原始数据的分布特征出发，通过函数作图观察出水COD和出水VFA的数据分布，力求找到一个数据相对分布均衡的区间来作为厌氧反应器的正常运行时的合理区间，进而根据出水COD和出水VFA的物理意义来把合理区间的上界作为阈值，一旦相关指标超过阈值该系统就应该做出反馈，让相关工作人员采取合理措施确保厌氧反应器保持正常运作。

关键字： 环保　污水处理　厌氧工艺　机器学习　神经网络　BP

一、 问题的提出

1.1 问题背景

党的十九大以来，我国越来越重视环保问题，而随着社会发展，生活中产生的废水越来越多，污水处理成为保护环境的重要一步，越来越多的学者投身到污水处理的相关研究之中。人们日常生活中产生的污水大多为有机废水，有关高浓度有机废水处理的方法已经成为各污水处理厂人员的研究对象。而在高浓度有机废水处理过程中，厌氧处理是一种高效率，低能耗的工艺，在此工艺中需要精确预测相关关键指标的数据如出水 COD 和 VFA 才能更好的进行工艺处理，预测研究相关数据非常重要。但目前厌氧工艺关键指标的预测与预警难以把握导致污水厌氧处理无法达到理想效果，经过分析，我们发现主要原因有传统厌氧过程预测模型对相关重要参数如 VFA（Volatile Fatty Acids）和出水 COD 难以在线测量，以及现有计算机技术不足导致相关数据预测的建模过程复杂，难以实现编程化。

1.2 问题重述

近年来我国对污水处理问题非常重视，如何预测有关污水处理厌氧工艺的有关数据指标成为许多研究员的目标。因此，我们需要使用数学模型来完成以下目标:

（1）根据 3 台厌氧反应器运行数据，做出对出水 COD 和 (VFA，Volatile Fatty Acids) 指标的预测。

（2）根据数据分析正常运行时 VFA 合理区间，并对 3 台反应器出水 COD 和 VFA 做出预警，判断警戒值。

1.3 问题分析

1.3.1 问题一分析

题目要求根据三台厌氧反应器运行数据，做出对出水 COD 和 VFA 指标的预测。经过对反应器运行数据的分析，我们发现影响出水 COD 与 VFA 指标的因素有很多，包括污水厌氧处理过程的复杂和各类参数如进水量，进水 COD,HLR 等，众多的影响因素导致建模的难度大大增加。而经过查阅资料，BP 算法和神经网络模型有较好的容错性，自适应能力，适用于多因素，多条件的场合。因此，采用 BP 算法和神经网络模型来对出水 COD 和 VFA 指标进行预测具有一定的优势。[1]

1.3.2 问题二分析

此问要求对三台反应器出水 COD 和 VFA 做预警，在问题一中已经建立相关模型来对出水 COD 和 VFA 进行预测，并得到一些相关数据。在此问中，三台机器性能不同等原因导致三台机器的预警阈值不相同，于是再次建立一个新模型不利于解决此问题，相反，利用问题一模型所得的相关数据与原始数据对出水 COD 和 VFA 指标进行规律总结，更有利于分析三台机器的预警阈值。

二、 模型假设

(1) 假设文档所提供的数据都是真实可靠的；
(2) 假设机器的性能不随时间的变化而变化；
(3) 假设出水 COD 和 VFA 指标只与文档中给出的指标有关；

三、 常用符号说明

符号	意义
Xmean	平均值
Xstd	标准差
Cov(X,Y)	协方差
Var[x]	方差
η	学习率
X_{max}	最大值
X_{min}	最小值

其余符号将在陈述公式时在公式下作说明。

四、 问题 1 的建模与求解

问题一要求我们根据 3 台厌氧反应器运行数据，使用合理的数学模型做出水 COD 和 VFA 指标的预测。

4.1 厌氧反应器指标数据的预处理

在建立模型和训练参数之前，我们需要对文档中的脏数据进行处理:

(1) 对于一些数据少数丢失，部分缺失的情况，我们采用平均值填充法进行处理。

(2) 对于一些数据丢失严重、甚至整行为空的情况，基于科学性与严谨性的原则，我们我们将这些情况从数据集中删除。

(3) 我们对输入指标进行 0-1 归一化处理。详情会在下文 4.3.5 中介绍。

(4) 对于一些离群点，我们予以剔除。下面用 2 号反应器出水 VFA 数据分布散点图为例，说明离群点剔除。由图 1 可知，红色圆点圈出的点属于离群点，下方的数据点分

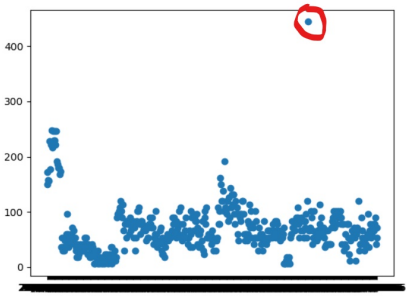

图 **1**　**2** 号反应器的出水 **VFA** 的数据分布散点图

布比较集中，因此我们将直接剔除这个点所属的数据行。

4.2 各指标数据的评估

面对众多的指标，我们采用计算相关系数的方法理顺指标之间的关系，有利于在后面的神经网络模型中选择最优的指标。相关系数公式为：

$$r(X,Y)=\frac{Cov(X,Y)}{\sqrt{Var[x]Var[Y]}}$$

利用 Excel 软件，我们计算出了指标之间的相关系数：相关系数越大，表示两个指标之间关系越紧密；反之，则越关系不大。这为我们选择模型指标提供了有力的支撑。

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1		OLR	F	inCOD	HLR	ALR	pH0	T	TSS	VFA	VMP	outCOD	CODrate	VFA1	pH1	T_in	T_high	T_low
2	OLR	1	0.7493115	0.347323	0.7492305	0.4030812	-0.161681	0.338369	-0.039747	0.2253106		0.2339635	0.2597332	-0.080216	0.0647226	0.3596795	0.115207	0.1030511
3	F	0.7493115	1	-0.334131	0.9999976	0.7154042	0.2203416	-0.032898	-0.228714	-0.296568		-0.198073	-0.192398	-0.162951	0.2296196	-0.01167	-0.138555	-0.165319
4	inCOD	0.347323	-0.334131	1	-0.334211	-0.449691	-0.548354	0.4810744	0.2903696	0.7856332		0.642894	0.6404061	0.1912647	-0.258828	0.4841534	0.2972838	0.3238354
5	HLR	0.7492305	0.9999976	-0.334211	1	0.7153823	0.2205015	-0.033132	-0.228738	-0.296329		-0.198044	-0.192543	-0.162543	0.2295921	-0.011912	-0.138886	-0.165652
6	ALR	0.4030812	0.7154042	-0.449691	0.7153823	1	0.3300728	-0.199495	-0.202853	-0.418194		-0.212862	-0.351056	-0.031294	0.240525	-0.184122	-0.259026	-0.293257
7	pH0	-0.161681	0.2203416	-0.548354	0.2205015	0.3300728	1	-0.370786	-0.39342	-0.336608		-0.397449	-0.299055	-0.098519	0.6065876	-0.365527	-0.229241	-0.265635
8	T	0.338369	-0.032898	0.4810744	-0.033132	-0.199495	-0.370786	1	0.2532278	0.254537		0.101058	0.5358742	-0.262449	-0.047154	0.9829771	0.8796048	0.8901377
9	TSS	-0.039747	-0.228714	0.2903696	-0.228738	-0.202853	-0.39342	0.2532278	1	0.3203427		0.1508934	0.2093302	0.1559277	-0.271247	0.25195	0.1521188	0.2048401
10	VFA	0.2253106	-0.296568	0.7856332	-0.296329	-0.418194	-0.336608	0.254537	0.3203427	1		0.5698983	0.4596502	0.3870981	0.0047387	0.2438437	0.039953	0.0606367
11	VMP																	
12	outCOD	0.2339635	-0.198073	0.642894	-0.198044	-0.212862	-0.397449	0.101058	0.1508934	0.5698983		1	-0.153689	0.4536375	-0.190508	0.105436	-0.075888	-0.054861
13	CODrate	0.2597332	-0.192398	0.6404061	-0.192543	-0.351056	-0.299055	0.5358742	0.2093302	0.4596502		-0.153689	1	-0.214069	-0.124209	0.5344606	0.4631272	0.4762907
14	VFA1	-0.080216	-0.162951	0.1912647	-0.162543	-0.031294	-0.098519	-0.262449	0.1559277	0.3870981		0.4536375	-0.214069	1	-0.125019	-0.267756	-0.380184	-0.341961
15	pH1	0.0647226	0.2296196	-0.258828	0.2295921	0.240525	0.6065876	-0.047154	-0.271247	0.0047387		-0.190508	-0.124209	-0.125019	1	-0.047545	-0.017276	-0.064266
16	T_in	0.3596795	-0.01167	0.4841534	-0.011912	-0.184122	-0.365527	0.9829771	0.25195	0.2438437		0.105436	0.5344606	-0.267756	-0.047545	1	0.8657045	0.883706
17	T_high	0.115207	-0.138555	0.2972838	-0.138886	-0.259026	-0.229241	0.8796048	0.1521188	0.039953		-0.075888	0.4631272	-0.380184	-0.017276	0.8657045	1	0.9612405
18	T_low	0.1030511	-0.165319	0.3238354	-0.165652	-0.293257	-0.265635	0.8901377	0.2048401	0.0606367		-0.054861	0.4762907	-0.341961	-0.064266	0.883706	0.9612405	1

图 **2**　相关系数

4.3 模型的建立

我们使用 python 语言中的 TensorFlow 框架搭建了 BP 神经网络模型。

4.3.1 人工神经网络模型

人工神经网络（Artificial Neural Network，ANN ）诞生于 20 世纪 50 年代，自上世纪末到现在，都是学界的研究热点。ANN 是一种按照人脑的组织和活动原理而构造的一种数据驱动型非线性映射模型, 由大量的神经元构成。每个神经元代表着一种函数，神经元之间的连接代表着权重。其在工程预测评估、组合优化、工程实践等领域已有广泛的应用。[2]

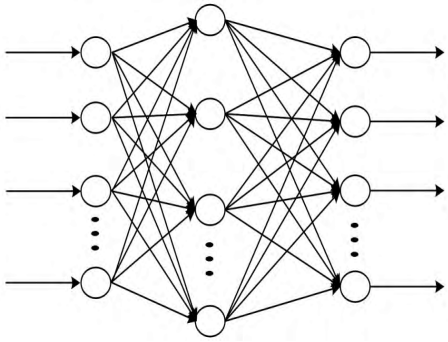

图 **3**　人工神经网络示意图

4.3.2 BP 神经网络模型

近年来，百分之 80 到 90 的人工神经网络模型是 BP 神经网络模型（Back -Propagation Network，前馈反向传播网络），其被称为神经网络最精华的部分，是前向网络的核心。[3]

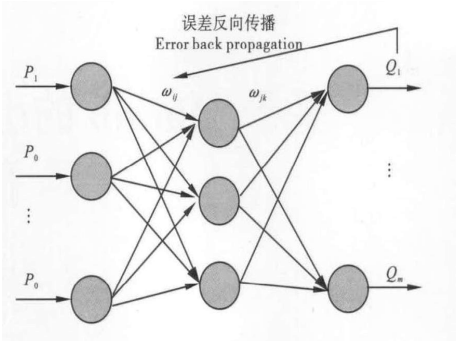

图 **4**　前馈反向传播网络示意图

BP 神经网络一般由输入层、隐含层、输出层组成。其具体的思想和核心是：在经过前向传播将输入层的值通过隐含层传播到输出层后，不断地计算输出层的预期值与真

实值之间的误差，然后将误差经过隐含层，传输到输入层。在一次一次的传播中不断地修改模型的参数，一旦模型的误差有增大的趋势则停止传输。这时，神经网络模型就得到了极大的改善，能够良好地拟合输入与输出之间的关系 [4]。下面按顺序介绍 BP 神经网络各计算阶段及其公式。[5]

首先是正向传播：

1. 从输入层到隐含层：

$$\alpha_h = \sum_{i=1}^{d} V_{ih} * x_i$$

其中 α_h 表示第 h 个隐藏神经元的输入,V_{ih} 表示输出层到隐含层 ih 间的权重，Xi 表示第 i 个输入神经元。

2. 隐含层中的激活函数：

$$b_h = f(\alpha_h)$$

其中 f 表示激活函数。

3. 从隐含层到输出层：

$$\beta_j = \sum_{h=1}^{q} w_{hj} * b_h$$

其中 β_j 表示第 j 个输出神经元的输入，w_{hj} 表示隐含层到输出层 hj 间的权重。

4. 输出层激励函数：

$$y_j' = g(\beta_j)$$

其中 g 表示输出层的激励函数。

5. 计算损失函数：

$$L^k = \frac{\sum_{h=1}^{m}(y_j'^k - y_j^k)}{2}$$

其中 y_j^k 表示真实值, $y_j'^k$ 表示预测值。

接下来是反向传播, 在我们的模型的模型中，我们一般采用优化器来实现这一步骤：

为了方便书写，下面我们采用 sigmoid 函数作激活函数。其中 sigmoid 函数为：

$$f(x) = \frac{1}{1+e^{-x}}$$

6. 损失函数对输出层的梯度：

$$\Delta w_{hj} = -\eta(y_j'^k - y_j^k) * y_j'^k * (1 - y_j'^k) * b_h$$

其中 η 为学习率。

7. 损失函数对隐含层到输出层参数的梯度:

$$\Delta\theta = (y_j'^k - y_j^k) * f'(\beta_j - \theta_j)$$

其中 f 为激励函数 sigmoid。

8. 损失函数对输入层到隐含层参数的梯度:

$$\Delta V_{ih} = -\eta X_i * (\sum_{j=1}^{m}(y_j'^k - y_j^k) * f'(\beta_j - \theta_j) * w_{hj}) * f'(\alpha_h)$$

至此，BP 神经网络各计算阶段和公式已介绍完毕。在实际计算中，我们将不断迭代此过程，直至达到迭代次数，获得拟合良好的模型。

4.3.3 激励函数的选择

在上一部分中，我们为了公式书写的整洁，采用了 Sigmoid 激励函数，但是在实际建模中，我们采用了 ReLU 激励函数。下面我们就来对这两个不同的激励函数进行介绍与比较，阐述我们选择 Relu 激励函数的原因。

sigmoid 激励函数为：

$$f(x) = \frac{1}{1+e^{-x}}$$

Sigmoid 激励函数是一种比较原始的激励函数，多在早期被用于各种神经网络中。其求导后的形式简洁，性质优良，在许多神经网络教材中被广泛介绍。但在应用过程中效果不太良好，主要缺点有两点：一是会有梯度弥散，影响弥合效果；二是计算中含有 exp 计算，耗时较多。

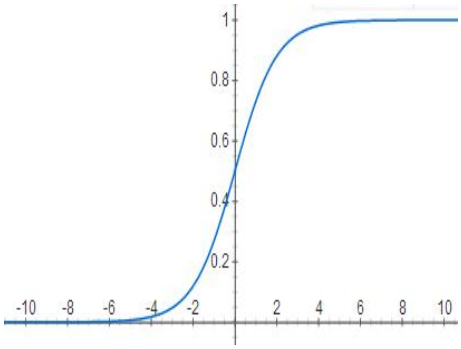

图 **5** **Sigmoid** 激励函数

Relu 激励函数为：

$$f(x) = max(0, x)$$

ReLU(Rectified Linear Unit) 激励函数在如今已被广泛应用在许多神经网络中，颇受学界和业界的欢迎。其部分解决了 Sigmoid 函数梯度弥散的问题，拟合效果更好；同时计算公式简单，收敛的速度更快。缺点则是神经元较为脆弱，容易死亡。[7]

理所当然地，我们选择了 ReLU 函数作为我们的激励函数以求更好的拟合效果与运算速度。

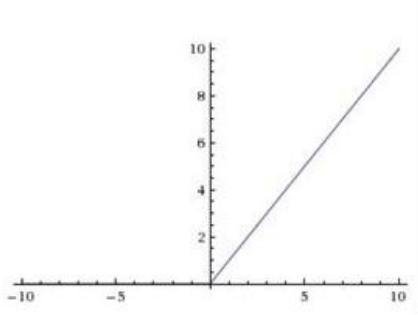

图 **6** **ReLU** 激励函数

4.3.4 优化器的选择

在我们的模型中，我们采用了 Adam 优化器。而在实际应用中，SGD、AdaGrad 也是经典的优化器。下面，我们就来介绍与对比三者之间的异同，阐述我们选择 Adam 优化器的原因。

首先，介绍梯度下降类优化器的一般优化原理。

1. 计算目标函数参数的当前梯度：

$$g_t = \bigtriangledown f(w_t)$$

其中，f 为目标函数，w_t 为目标参数。

2. 根据历史梯度计算一阶动量：

$$A_t = \alpha(g1, ..., gt)$$

3. 根据历史梯度计算二阶动量：

$$B_t = \beta(g1, ..., gt)$$

4. 计算下降梯度：

$$\gamma_t = \frac{\eta * A_t}{\sqrt{B_t}}$$

5. 优化参数：

$$w_{t+1} = w_t - \gamma_t$$

接下来分别介绍三种优化器：

1.SGD 优化算法为：

$$\gamma_t = \eta * g_t$$

无动量的计算，其余同上。

SGD（Stochastic Gradient De-scent, 随机梯度下降法）的特点就是较为简单易懂，没有动量，因此没有追溯历史梯度，下降梯度即使用目标函数参数的当前梯度。SGD 是最

原始的神经网络优化算法，用一些简单的高等数学知识即可理解，在沿着梯度下降后目标参数会停留在一个局部最优点。SGD 方法原始简单，但下降速度较其他优化算法慢。[6]

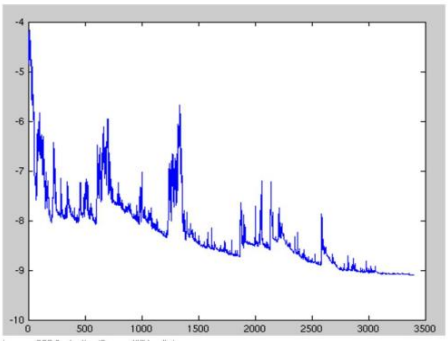

图 **7**　梯度下降

2.AdaGrad 优化算法为：

$$B_t = \sum_{i=1}^{t} g_i^2$$

$$\eta = \eta / \sqrt{B_t}$$

无一阶动量的计算，其余与普遍优化原理相同。即等同于在 SGD 算法中增加了二阶动量的计算。

AdaGrad 采用了牛顿法（二阶优化方法），是一种“自适应学习率”的优化算法。它在 SGD 的基础上增加了二阶动量的计算。对于经常被用到并更新的参数，我们不希望它被新的反馈影响太大，因此我们需要将学习率调低；反之，很少被使用到的参数，如果遇到更新，我们希望在此次更新中学习更多一些，因此把学习率调高。AdaGrad 的缺点在于 B_t 单调递增使得 η 单调递减至 0，可能导致后期学习率过小，简介使得训练提前结束，无法从剩余样本中继续训练。

3.Adam 优化算法为：

$$A_t = \beta_1 * m_{t-1} + (1 - \beta_1) * g_t$$

$$B_t = \beta_2 * B_{t-1} + (1 - \beta_2) * g_t^2$$

其中，$\beta_1 \beta_2$ 为超参数。

即与一般优化原理基本相同，在 SGD 的基础上增加了一阶动量的计算和二阶动量的计算。

Adam 可以说是前面两种算法和其他算法的集大成者。Adam 在神经网络领域是十分流行的算法，它同时充分利用了一阶矩计算和二阶矩计算，计算了梯度的指数移动均

值（exponential moving average），超参数 $\beta_1\beta_2$ 用于控制其衰减率。由此，调参工作量大大减少，甚至用默认参数就可以解决大部分的问题。

显而易见，Adam 不仅拥有更高的性能、更优秀的特性，对于参数的选择也十分友好。因此，我们选择 Adam 优化器。

4.3.5 归一化模型的选择与反归一化的处理

这个模型中，需要对出水 COD 和 VFA 进行预测需要进行大量的数据运算，而影响出水 COD 和 VFA 指标变化的参数复杂多样，且数值大小差距过大，直接使用原始数据进行处理不易获得出水 COD 和 VFA 的预测值。经过研究，我们决定对其数据进行归一化处理，经查阅资料，我们发现以下三种归一化方法是对数据进行整合处理的常用方法。

（1）z-score 标准化（zero-mean normalization）

其公式为：

$$Z = X - X_{mean}/X_{std}$$

z-score 标准化是将数据按比例缩放至符合标准正态分布，即均值为 0，标准化为 1 的方法。适用于未知最大值与最小值或有超出离群的情况。

(2) min-max 标准化（Min-max normalization）/0-1 标准化（0-1 normalization）

其公式为:

$$X = \frac{X - X_{min}}{X_{max} - X_{min}}$$

min-max 标准化是对原始数据进行线性代使之落到 [0,1] 区间的方法。适用于大多数数据处理，但当出现新数据时需要重新定义最大最小值。

（3）Decimal scaling 小数定标标准化

其公式为:

$$X = X/(10 * j)$$

其中其中 j 是同时满足 $-1 <= X_{max}/(10 * j) <= 1$ 与 $-1 <= X_{min}/(10 * j) <= 1$ 的条件的值。

而针对我们所建的模型需要的数据，Decimal scaling 小数定标标准化处理得到的数据比较单调线性化，单纯的小数点移动来处理没有其他两种归一化的方法所得数据更适用。而我们经过对原始数据的分析，z-score 标准化在处理未知最大最小值和有超出范围值的数据的优势在此处无法发挥很大的作用，而 min-max 标准化相较于 z-score 标准化计算过程更加简洁，所得到的数据也更加方便实用，于是我们采取 min-max 标准化对数据进行归一化。

用原始数据经过 min-max 标准化处理后得到的数据，我们使用所建的 keras sequential 模型对出水 COD 和 VFA 指标进行预测，在训练数据时，我们将数据进行归一化，得

到的预测数据属于 [0,1] 区间。为了更好的进行误差计算与精度，减小误差的计算复杂程度，我们需要将数据进行反归一化，用预测得到的真实数据与原始数据对比，使计算误差的过程更简洁。

而由于原始的预测数据不应该是在 [0，1] 区间，为了方便对我们训练出来的预测数据和提供的样本预测数据进行 MAE（平均绝对误差）评估度量，我们需要对预测数据进行反归一化处理。公式如下：

$$MAE = \frac{\sum_{i=1}^{m} |h(x_i) - y_i|}{m}$$

4.3.6 使用 Keras Sequential 模型建模

在理清 BP 神经网络结构，选择好激励函数 ReLU 和优化器 Adam 后，我们使用了 Keras Sequential 模型实现我们的 BP 神经网络，模型结构参照 BP 神经网络结构，代码在附件中。参考了相关性系数和实际的建模情况后，我们选择了“进水量”,“进水 COD”,“反应器内 pH”,“反应器内温度”,“TSS”“进水 VFA”六个参数作为输入参数训练模型，拟合输出值“出水 COD”“VFA”。在下一章节中，我们将简述一下模型的拟合结果，解决题目中的问题。

4.4 模型的拟合结果

从图中可以看到，在 250 个 Epoch 后，我们的模型梯度下降效果可观。其中，loss 采用的计算方法为 MSE，其公式为：

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(y_i - y_i^{pred})^2$$

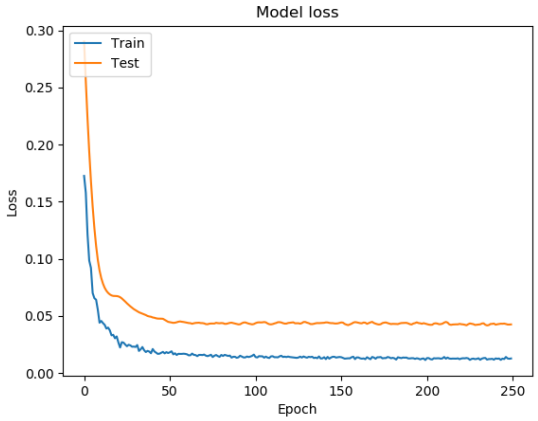

图 **8**　**Loss**

表 **1**　预测出水 **COD** 正确率（左边精度 **0.9**，右边 **0.8**）

次数	1 号反应器	2 号反应器	3 号反应器	次数	1 号反应器	2 号反应器	3 号反应器
1	0.7123	0.57	0.6535	1	0.9041	0.85	0.901
2	0.6301	0.58	0.4851	2	0.8904	0.85	0.8416
3	0.6301	0.53	0.604	3	0.9589	0.73	0.9505
4	0.6301	0.57	0.6337	4	0.8904	0.92	0.7723
5	0.6712	0.45	0.6634	5	0.9452	0.75	0.8911
6	0.6027	0.47	0.5743	6	0.9315	0.91	0.8812
7	0.726	0.64	0.604	7	0.8767	0.85	0.9307
8	0.6164	0.57	0.5545	8	0.9041	0.87	0.9109
9	0.6164	0.66	0.5743	9	0.9315	0.83	0.901
10	0.4521	0.61	0.5941	10	0.8493	0.86	0.9109
平均	0.62874	0.565	0.59409	平均	0.90821	0.842	0.88912

为了进一步研究我们这个模型的拟合性，我们定义了一个精度 p：

$$p = \frac{|y_{pred} - y_{real}|}{y_{real}}$$

若数据运算后所得的结果大于等于精度 p，我们就认为这个数据正确。

从表中我们可以看出，在预测出水 COD 时，在精度 0.8 的情况下得到了百分之 90 左右的正确率，在精度 0.9 的情况下得到了百分之 60 左右的正确率，模型有效，可以用此模型实现对出水 COD 的预测。从表中我们可以看出，在预测出水 VFA 时，在精度 0.7 的情况下得到了百分之 50 左右的正确率。虽然 loss 函数已呈现下降的态势，但拟合程度依然不高。即使如此，我们相信该模型也可用于 VFA 粗略的大致预测。

综上，利用此神经网络模型，可以比较准确地预测出出水 COD 的值，粗略预测出出水 VFA 的值，基本完成问题一。

表 **2**　预测出水 **VFA(**精度 **0.7)**

次数	1 号反应器	2 号反应器	3 号反应器
1	0.3014	0.43	0.4851
2	0.6438	0.23	0.495
3	0.589	0.3	0.5347
4	0.3562	0.45	0.396
5	0.4932	0.46	0.495
6	0.3836	0.51	0.5149
7	0.2192	0.5	0.5545
8	0.4658	0.32	0.5743
9	0.589	0.32	0.5248
10	0.6027	0.5	0.4455
平均	0.46439	0.402	0.50198

五、 问题 2 的建模与求解

1. 题目要求分析出水 COD 和出水 VFA 的合理区间，以及提出出水 COD 和出水 VFA 的阈值。

2. 求解方法主要是通过分析数据的分布特征，通过编写程序分别画出两个输出参数出水 COD、出水 VFA 随着时间变化趋势的分布散点图，再分别画出两个输出参数出水 COD、出水 VFA 的频率分布直方图，通过几幅图中数据的分布特性来综合分析出合理的区间以及确定阈值。

3. 根据污水处理过程中厌氧工艺处理的相关技术，我们可以知道出水 COD 表征了厌氧反应器对于污染物的去除能力，当出水 COD 越小时，反应器运行效率越高，因此我们确定出水 COD 合理区间时，主要是看出水 COD 数据分布在某段区间的频率，在区间值不过大的合理范围之内继续找到数据更加趋于平均分布的区间，进而把区间上界设置为阈值。另外，出水 VFA 也可以反映出厌氧反应器是否处于合理处理负荷，当机器正常运行时 VFA 应该维持在一定范围之内，而当 VFA 偏高时，厌氧反应有失衡趋势，如果继续偏高的话，那么系统 PH 将会迅速降低，对于厌氧反应有酸化效果，进而导致反应器运行失败，因此我们确定出水 VFA 合理区间时，同样是看出水 VFA 数据分布在

某段区间的频率，在区间值相对较小的范围内确定合理区间，进而把区间上界设置为阈值。

4. 阈值的设定可以对于厌氧反应器的厌氧反应过程起到预警作用，一旦监测到出水 COD 或出水 VFA 超过设定好的阈值，系统就应该通知到相关人员马上采取如加入缓冲剂适当提升碱度、降低进水负荷等合理措施来确保反应器正常运作。

5. 我们以 1 号反应器为例，如下图展示了 1 号反应器的散点图和频率分布直方图，以及根据相关分析得出的合理区间和阈值。

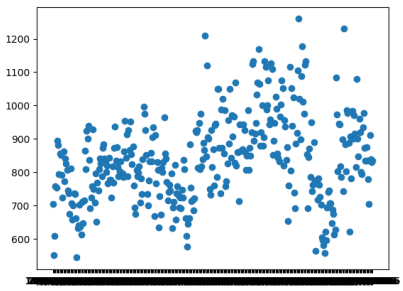

图 **9**　**1** 号反应器的出水 **COD** 随着时间变化的趋势散点图

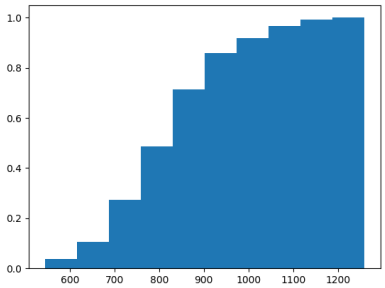

图 **10**　**1** 号反应器的出水 **COD** 的频率分布直方图（累积型）

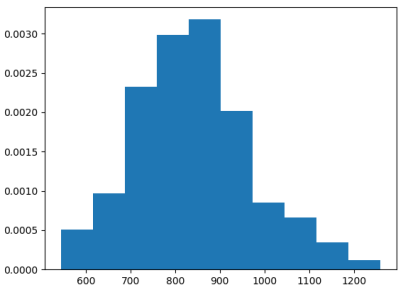

图 **11**　**1** 号反应器的出水 **COD** 的频率分布直方图

结果分析：

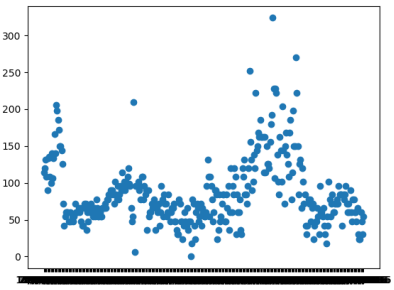

图 **12**　**1** 号反应器的出水 **VFA** 随着时间变化的趋势散点图

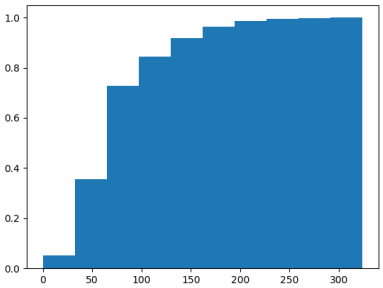

图 **13**　**1** 号反应器的出水 **VFA** 的频率分布直方图（累积型）

根据图 9-图 11，我们发现较多数据平均分布在 650-1000 这个区间，而且 1000 这个值也近似达到了整体分布频率的 0.9，足够用来确定出水 COD 的合理运行区间以及划分阈值。

根据图 12-图 14，我们发现较多数据平均而且集中地分布在 30-150 这个区间，150 这个值也近似达到了整体分布频率的 0.9，可以划分出出水 VFA 的正常运行时的参数区间以及确定 150 这个阈值。

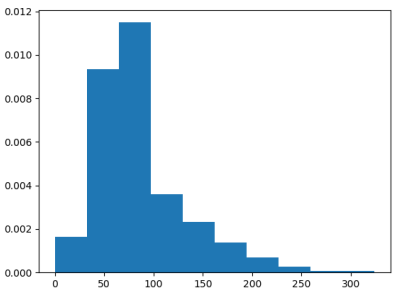

图 **14**　**1** 号反应器的出水 **VFA** 的频率分布直方图

表 **3**　出水 **COD** 合理区间和阈值

反应器编号	出水 COD 合理区间	出水 COD 阈值
1 号反应器	650-1000	1000
2 号反应器	650-950	950
3 号反应器	680-930	930

表 **4**　出水 **VFA** 合理区间和阈值

反应器编号	出水 VFA 合理区间	出水 VFA 阈值
1 号反应器	30-150	150
2 号反应器	0-125	125
3 号反应器	0-110	110

六、 模型优缺点

6.1 模型优点

1. 我们使用平均值填充、偏移值去除等方法实现了对脏数据的预处理，使数据更为可信，模型更具科学性与严谨性。

2. 我们基于 keras Sequential 搭建的 BP 神经网络模型以及选择的一系列结构一方面具有优良的性质和较高的效率，另一方面模型均采用了常见的激活函数、优化器等部件，在实际生产中较易实现，门槛较低。

3. 模型对出水 COD 的预测较为准确，在精度 0.8 的情况下可以实现百分之九十的正确率，拟合度较高，可以为实际生产带来价值。

4. 针对问题二我们的模型采用了大量的各式图表，直观地反映了数据的分布规律,简单易懂。

6.2 模型缺点

1. 针对问题一 VFA 的拟合精度不足，这一点有待改进。

2. 针对问题二没有运用严格的数学公式进行公理化的证明，严谨性略有不足。

七、 参考文献与引用

[1] 曹为阳. 污水处理出水 COD 软测量预测建模方法 [J]. 安徽工业大学,2017,:.

[2][3][4] 苏高利. 论基于 MATLAB 语言的 BP 神经网络的改进算法 [J]. 科技通报,2003,2:

[5] 刘希玉.BP 神经网络预测算法的改进及应用 [J]. 山东师范大学信息科学与工程学院,,:.

[6] 郭敏钢. 基于 Tensorflow 对卷积神经网络的优化研究 [J]. 计算机工程与应用,2019,:.

[7] 常永虎. 基于梯度的优化算法研究 [J]. 现代计算机,2019,:.