**审定成绩：**

|  |  |  |  |
| --- | --- | --- | --- |
| **重庆邮电大学**  **毕业设计（论文）** | | | |
|  | | | |
|  | **中文题目** | **基于ADASYN和机器学习的水质检测系统研发** |  |
| **英文题目** | **Water quality detection system based on ADASYN and machine learning model** |
| **学院名称** | **国际学院** |
| **学生姓名** | **刘畅** |
| **专 业** | **软件工程** |
| **班 级** | **0231201** |
| **学 号** | **2023210769** |
| **指导教师** | **姓名 许汀汀 职称 副教授** |
| **答 辩 组**  **负 责 人** | **姓名 职称** |
|  | |

**2024年 4 月**

**重庆邮电大学教务处制**

国际学院本科毕业设计（论文）诚信承诺书

本人郑重承诺：

我向学院呈交的论文《基于ADASYN和机器学习模型的水质检测系统》，是本人在指导教师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明并致谢。本人完全意识到本声明的法律结果由本人承担。

年级 2024

专业 软件工程

班级 34082003

承诺人签名

年 月 日

学位论文版权使用授权书

本人完全了解重庆邮电大学有权保留、使用学位论文纸质版和电子版的规定，即学校有权向国家有关部门或机构送交论文，允许论文被查阅和借阅等。本人授权重庆邮电大学可以公布本学位论文的全部或部分内容，可编入有关数据库或信息系统进行检索、分析或评价，可以采用影印、缩印、扫描或拷贝等复制手段保存、汇编本学位论文。

（注：保密的学位论文在解密后适用本授权书。）

|  |  |
| --- | --- |
| 学生签名： | 指导老师签名： |
| 日期： 年 月 日 | 日期： 年 月 日 |

摘要

本研究旨在开发基于ADASYN和机器学习模型的智能水质监测系统，以有效应对日益严峻的水质污染问题对人类健康和环境的影响。随着工业化和城市化的不断推进，传统水质监测方法存在着周期长、成本高等问题，难以满足实时性和准确性的需求。因此，本研究利用了机器学习算法技术，构建了一款水质监测系统，采用了ADASYN和SMOTE等过采样算法，有效解决了水质数据不平衡的挑战。通过增加少数类样本数量，的模型更好地学习到了少数类别的特征，提升了模型的准确性和可靠性。接着，运用了KNN、GDBT、SVM和ANN等机器学习模型对过采样数据进行训练，实现了对水质数据的自动化、高效率预测，从而提高了监测效率和准确性。该系统采用了前后端分离架构，前端采用了Vue + Element UI，后端采用Django实现MVC框架，实现了前后端的高效交互。尽管如此，仍需针对特征选择模块的优化、引入更高效的算法和调参方法等方面进行进一步改进。未来的工作将侧重于系统的优化，并引入可视化页面，以提供更全面、直观的模型性能评估结果，以进一步提升系统的实时性和用户满意度。通过持续的改进和优化，智能水质监测系统有望在实际应用中发挥更为重要的作用，为保护水资源和环境做出更大的贡献。

**关键词：**水质检测；过采样算法；机器学习模型；Vue；Django

**Abstract**

The aim of this study is to develop an intelligent water quality monitoring system based on ADASYN and machine learning models to effectively respond to the impacts of the increasing water pollution problems on human health and the environment. With the continuous advancement of industrialization and urbanization, the traditional water quality monitoring methods suffer from the problems of long cycle time and high cost, which make it difficult to meet the demand of real-time and accuracy. Therefore, we have utilized machine learning algorithm technology to build a water quality monitoring system that employs oversampling algorithms such as ADASYN and SMOTE to effectively address the challenge of unbalanced water quality data. By increasing the number of minority class samples, our model better learns the features of the minority classes and improves the accuracy and reliability of the model. Then, we applied machine learning models such as KNN, GDBT, SVM, and ANN to train on the oversampled data, realizing automated and efficient prediction of water quality data, thus improving monitoring efficiency and accuracy. The system adopts a front-end and back-end separation architecture, the front-end adopts Vue + Element UI, and the back-end adopts Django to realize MVC framework, which realizes the efficient interaction between the front and back-end. Nevertheless, we still need to make further improvements for the optimization of the feature selection module, introduction of more efficient algorithms and tuning methods. Future work will focus on the optimization of the system and the introduction of visualization pages to provide more comprehensive and intuitive model performance evaluation results to further enhance the real-time performance and user satisfaction of the system. Through continuous improvement and optimization, our intelligent water quality monitoring system is expected to play a more important role in practical applications and make greater contributions to the protection of water resources and the environment.

**Keywords**: Water quality monitoring; Oversampling algorithm; Machine learning model; Vue; Django

目录

[第1章 引言 1](#_Toc166575781)

[1.1 研究背景和意义 1](#_Toc166575782)

[1.2 国内外研究现状 2](#_Toc166575783)

[1.2.1 国外研究现状 2](#_Toc166575784)

[1.2.2国内研究现状 2](#_Toc166575785)

[1.3 主要内容 3](#_Toc166575786)

[第2章 需求分析及系统设计 5](#_Toc166575787)

[2.1 需求分析 5](#_Toc166575788)

[2.1.1 功能需求 5](#_Toc166575789)

[2.1.2 非功能需求 5](#_Toc166575790)

[2.2 系统总体设计 6](#_Toc166575791)

[第3章 数据获取及模型方法 7](#_Toc166575792)

[3.1数据获取 7](#_Toc166575793)

[3.2自适应过采样算法 8](#_Toc166575794)

[3.2.1 ADASYN 8](#_Toc166575795)

[3.2.2 SMOTE 8](#_Toc166575796)

[3.3机器学习模型与超参数优化 9](#_Toc166575797)

[3.3.1 K最近邻算法（KNN） 9](#_Toc166575798)

[3.3.2 梯度提升树（GDBT） 10](#_Toc166575799)

[3.3.3 支持向量机（SVM） 11](#_Toc166575800)

[3.3.4人工神经网络（ANN） 12](#_Toc166575801)

[第4章 前后端技术栈及其核心代码展示 13](#_Toc166575802)

[4.1 前端技术栈以及页面展示 14](#_Toc166575803)

[4.1.1 前端技术栈 14](#_Toc166575804)

[4.1.2 页面展示 14](#_Toc166575805)

[4.2 后端技术栈以及核心代码展示 19](#_Toc166575806)

[4.2.1 后端技术栈 19](#_Toc166575807)

[4.2.2 核心代码展示 19](#_Toc166575808)

[第5章 实验结果与分析 26](#_Toc166575809)

[5.1 过采样算法实现结果 26](#_Toc166575810)

[5.2机器学模型分析结果 27](#_Toc166575811)

[第6章 结论与展望 31](#_Toc166575812)

[6.1 系统总结 31](#_Toc166575813)

[6.2 系统不足与展望 32](#_Toc166575814)

[参考文献 33](#_Toc166575815)

[致谢 35](#_Toc166575816)

# 

# 第1章 引言

## 1.1 研究背景和意义

随着工业化和城市化进程的加快，水质污染日益严重，给人类健康和环境造成了严重的威胁。因此，开展水质检测工作具有重要的现实意义，水质的安全性和稳定性直接关系到人民群众的生活质量和国家的生态安全，然而，传统的水质检测方法往往需要大量的人力、物力投入，且存在着检测周期长、成本高等问题，难以满足实时性和准确性的要求。近年来，利用人工智能和机器学习技术解决水质检测问题成为了研究的热点之一[1]。Ahmed等人回顾传统实验室检测水质耗时慢，成本高等基础上，提出了一个基于物联网的低成本系统，该系统采用机器学习技术实时监测水质，分析水质趋势，并检测异常事件，如水的故意污染[2]。因此利用机器学习模型和算法进行水质监测，则能够实现自动化、高效率的监测，提高了水质监测的效率和准确性。

在水质检测领域，研究工作目前重点聚焦于几个核心挑战：解决数据不平衡问题、选择及优化适合的机器学习模型。为了应对数据分布不均的问题，研究者们已经探索了多种策略，包括合成少数类过采样技术、自适应合成抽样以及欠采样技术。此外，许多研究尝试了多种机器学习模型，合适模型的选择与优化仍是研究备受关注的焦点。Olatinwo等人开发了基于高速公路双向长短期记忆网络（Highway-BiLSTM）的水质分类工具，旨在将机器学习模型的选择和优化集成到边缘计算支持的水质监测系统中，以便实现现场水质的快速分类[3]。研究中采用了合成少数过采样技术（SMOTE）和模型调优策略来处理数据不平衡，从而有效提升了模型的分类准确度。此外，Patel等人通过应用SMOTE技术改进了水质数据集的类别平衡，并采用多种模型来评估水质。他们的研究结果指出，随机森林和梯度提升模型在准确率上表现最佳。为了进一步增强模型透明度和解释性，研究中还引入了可解释的人工智能（XAI）技术，以确定影响模型预测的关键特征[4]。综上所述，确保数据平衡及对机器学习模型进行调优是水质监测系统研发中的关键步骤，这些策略有助于提高系统的准确性和可靠性，从而做出更精确的决策支持和环境管理。

未来，水质检测技术将朝着智能化、自动化、高效化的方向发展。随着人工智能和大数据技术的成熟和应用，有望在数据不平衡问题、机器学习模型优化以及系统构建等方面取得更多创新和突破，本论文旨在探索基于机器学习的水质检测技术，通过平衡水质数据和使用多个的机器学习模型，构建智能化的水质监测系统，选择及优化合适的机器学习模型，为解决当前水质监测中存在的问题提供新的解决方案。该研究对于推动水质监测技术的智能化、自动化发展，提高水质监测的效率和准确性，具有重要的理论和实践意义。

## 1.2 国内外研究现状

### 1.2.1 国外研究现状

水质是环境监测的一个重要方面，国外各种研究都侧重于创新算法，以提高采样效率和准确性。Sahu 等人（2011 年）提出了一种用于预测水质的自适应网络模糊推理系统（ANFIS），强调了适当的训练和参数选择对获得准确结果的重要性。在自适应采样算法领域，已有多项研究探索了其在不同领域的应用[5]。Wong 等人（2016 年）成功部署了一种自适应采样算法，用于优化实时水质监测的传感器网络，展示了物联网服务在环境传感方面的优势[6]。Shu 等人（2017）开发了一种用于自动水质监测的高能效自适应采样算法，与固定采样率相比，显著节省了电池能耗[7]。此外，Xu 等人（2020 年）介绍了一种用于预测娱乐水质的自适应合成采样算法（ADASYN），利用了 K-均值近邻和支持向量机等机器学习技术强调了自适应采样在提高水质预测准确性方面的重要性[8]。总之，这些研究展示了自适应采样算法在提高水质监测和预测准确性方面的重要意义，其研究侧重于并行聚类、模糊推理系统和机器学习技术等各个方面，在优化传感器网络、节约能源和提高水质监测系统的整体质量方面发挥着至关重要的作用。

### 1.2.2国内研究现状

在国内水质检测已经成为衡量环境质量的重要标准之一，为评估全国不同水源中存在的污染物和指标，已开展了多项研究。刘玲华等人（2010 年）调查和评估了中国水源中的挥发性有机化合物，强调了数据采样的重要性，了解水污染物组成[9]。张维为等人（2012 年）重点研究了降雨对青岛第一海水浴场微生物水质的影响，强调有必要进行密集采样，以开发针对水质快速变化的科学预警系统[10]。彭子康等人基于颜色特征值分析和决策树模型机器学习，分类并处理大量水样集，构建出水体浑浊度及水体氨氮浓度之间的相关性模型，检测精度可达90.24%[11]。因此对于采样过程中存在不均衡的问题需要得到重点关注，通过选择合适数据特征值以及平衡训练数据，可以避免检测水质过程出现的结果不可靠，不准确的问题。

过采样是一种处理数据不平衡问题的方法，它通过增加少数类样本的数量来平衡各类别之间的样本分布，常用的过采样算法包含随机过采样，smote以及自适应过采样算法等。凌煦等人通过结合ADASYN过采样和XGBoost算法研究影响光伏出力的关键要素，结果有效提升模型的准确性[12]。李瑞平等人提出一种基于欧氏距离改进的Borderline-Smote过采样算法，使用梯度提升树模型预测冠心病，准确率提高8.4%，精确率提高2.9%，召回率提高9.1%，AUC提高4.6%[13]。陈虹等人提出了一种自适应过采样算法（ADASYN）与改进堆叠式降噪自编码器（SDA）结合的入侵检测模型实验结果表明，ADASYN-SDA模型相较于SDA、AE-DNN和MSVM模型，在平均准确率、检测率和误判率上均有一定程度的提高[14]。以上研究均表明，通过过采样增加了少数类样本的数量，可以使模型更好地学习到少数类别的特征，此外，新的合成样本有助于使模型在学习过程中更好地拟合数据分布，减少了过拟合的风险。因此在水质检测方面，过采样技术可以有效解决水质数据不平衡问题。

国内过采样技术在水质检测方面的研究有所空缺，因此将使用自适应过采样算法（ADASYN）以及合成少数派过采样技术（SMOTE）算法增加异常样本的数量，并结合K最近邻算法（KNN），梯度提升树（GDBT），支持向量机（SVM）以及人工神经网络（ANN）模型对水质数据集进行训练预测，开发一个水质检测系统。

## 1.3 主要内容

第1章引言部分将介绍水质研究的重要性，以及国内外相关研究背景，并突出本文水质检测系统的创新之处。

第2章深入阐述了水质检测系统的需求分析以及总体设计框架。在需求分析部分，详细讨论了系统应满足的功能性和非功能性要求，总体设计部分则着重于讲解了系统架构的各模块设计以及数据流向

第3章将详细介绍数据获取方法以及研究方法。这包括相关数据集的获取途径，水质检测特征的分析，以及过采样算法和机器学习模型的选择和原理。

第4章将重点展示前端和后端所采用的技术栈，并展示相关页面设计和后端核心代码部分，以便读者全面了解系统的实现方式。

第5章将呈现算法模型的运行结果，着重于不同机器学习模型的超参数调优过程，并展示交叉验证集的准确率结果，以评估模型的性能。

第6章作为论文的结尾部分，将对毕业设计的工作进行总结，并提出今后可能的研究方向，为读者提供进一步探索的思路。

# 第2章 需求分析及系统设计

## 2.1 需求分析

在水质监测系统的需求分析中，首先确定了系统的核心目标：实现水质数据的过采样处理，利用机器学习算法对水质进行分类和预测，并通过用户友好的界面让用户能轻松访问和解读水质结果。

### 2.1.1 功能需求

在功能需求方面，系统需包括数据上传模块，负责接收检测水质的原始数据，指标包含pH值、溶解氧、浊度等关键参数，并具备容错能力以确保数据的完整性和准确性。数据预处理模块将负责数据清洗，包括过采样处理、降噪和异常值检测，将数据格式化以适应后续分析处理的需求，前端页面数据面板会展示处理后的数据集。模型训练和预测模块将使用机器学习算法来训练水质分类模型，包括KNN、ANN、GDBT及SVM，通过交叉验证等技术选择最优的超参数，自动优化和更新模型，使用训练好的模型预测新数据的水质状况。前端显示模块需要提供直观的用户界面设计，以显示实时和历史水质数据，并提供数据可视化工具，如图表和趋势线，同时允许用户查看预测的结果数据。后端服务模块将处理来自前端的请求，调用适当的数据处理和机器学习模型，并采取安全措施以保护数据，包括用户认证和访问控制。

### 2.1.2 非功能需求

在非功能需求方面，水质监测系统必须满足一系列的性能、可用性、兼容性和可维护性标准。性能方面，要求系统应具备高可用性和响应速度，性能要求系统能够快速处理大量数据，并且能够在高并发条件下稳定运行，以确保即使在数据量上万的情况下，系统的响应时间仍能保持在可接受的范围内，系统的处理能力应通过压力测试和性能基准测试来验证，以确保满足预定的性能指标。可用性方面，要求系统界面直观易用，让用户无需专业训练即可操作，同时系统应该保证高可用性，这意味着系统的正常运行时间要尽可能接近100%，并且在进行维护或升级时要最小化对用户的影响。兼容性方面，确保系统API的标准化和文档化，方便与其他系统集成和数据共享，支持主流浏览器和多种移动设备，确保用户可以在不同的终端上访问系统。可维护性方面，设计时应采用模块化和松耦合的架构，以便在未来可以方便地添加新功能或升级现有功能。

## 2.2 系统总体设计

本水质检测系统采用模块化设计，旨在提高数据处理效率、模型准确性以及用户交互的便捷性。系统的总体架构分为数据上传、数据预处理、模型训练与预测、前端展示和后端服务五个主要部分。以下详细描述各部分的功能和数据流向：

数据上传模块负责收集检测水质的原始数据。这些数据包括但不限于水温、pH值、溶解氧、浊度等多个指标。收集到的数据将以csv格式发送后端中，以便后续的数据预处理和分析。

数据预处理模块接收原始数据，并执行清洗、归一化、特征选择等步骤，以提高数据质量和模型训练效率。为了解决数据不平衡问题，本模块集成了ADASYN和SMOTE算法，通过合成少数样本来平衡数据集，为后续的模型训练创建更加健壮的基础。

模型训练与预测模块是系统的核心，包含多个机器学习算法，如KNN、ANN、GDBT和SVM。该模块利用预处理后的数据训练模型，并对新的水质数据进行预测。模型训练过程中，通过交叉验证等技术选择最优的超参数，以确保预测的准确性和可靠性。

前端展示模块使用Vue3结合Element UI框架开发，提供用户友好的界面。用户可以通过该界面查看数据处理结果、模型调优过程以及预测报告。此外，用户还可以通过前端界面上传新的水质数据并接收模型的预测结果。

后端服务模块基于Django框架搭建，接收来自前端展示模块的HTTP请求，包括用户对水质数据的查询、预测结果的请求以及其他相关操作。当新的数据通过数据采集模块进入系统时，后端服务模块调用数据预处理模块来清洗和平衡数据，确保数据的可靠性。利用处理过的数据，后端服务模块调用机器学习模型进行训练和预测。将机器学习模型的预测结果整合，并通过RESTful API响应前端的请求，将结果以结构化的格式（如JSON）返回给前端展示模块。

系统的数据流向开始于数据采集模块，原始数据被传输到后端。预处理模块对原始数据进行必要的处理后，经过预处理的数据被用于训练机器学习模型。当模型训练完成后，它将用于对新的或实时数据进行预测。训练好的模型用于对新数据进行预测，预测结果由后端服务模块接收，并进行必要的格式化和封装处理，以适配前端展示模块的展示需求。最终，处理后的预测结果通过API接口发送给前端展示模块。用户可以通过前端界面查询实时和历史预测结果，进行数据可视化和进一步分析。

# 第3章 数据获取及模型方法

本章主要介绍水质数据集的特征值及获取途径，并使用相关的过采样模型算法对数据集进行平衡，包括ADASYN和SMOTE算法。此外，还将使用不同的机器学习模型，包括ANN、GDBT和SVM，通过选择合适的超参数对水质数据进行训练和预测。

## 3.1数据获取

本研究所使用的水质数据集来源于Kaggle平台。数据集中包含了多个特征值，这些特征值是对水质状况进行评估和监测的关键参数。下面是最佳水质中每个特征值指标最优取值范围：温度：水体的最佳温度范围应该在20°C到30°C之间，对于淡水体来说，温度的偏离可能意味着环境的异常变化。溶解氧：水中溶解氧的含量应该在4（mg/L）到8（mg/L）之间，这是维持水中生物生存的基本要求。pH值：水质的pH值范围应该在6到8之间，这对于维持水体中的生态平衡至关重要。电导率：水质中的电导率理想情况下应该在150到500 μmhos/cm之间，这是评估水体中溶解物质含量的重要指标。生化需氧量（BOD）：水中的BOD值应低于5（mg/L），以确保水质清洁度。硝酸盐和亚硝酸盐的平均值：不应超过5.5（mg/L），否则可能对水体造成污染。粪大肠菌：其值不应超过200 MPN/100ml，超过此值可能表明水体受到了污染。总大肠菌（包括粪大肠菌）：其值不应超过500 MPN/100 ml，超过此值可能对人类健康产生危害[15]。这些特征值反映了水质状况的多个方面，对于水质评估和监测具有重要意义。

## 3.2自适应过采样算法

### 3.2.1 ADASYN

ADASYN算法是一种自适应的合成抽样方法，专门用于处理不平衡数据集的过采样问题。此算法对于每个少数类样本，会计算其与最近邻少数类样本的距离，并计算出每个样本的比例因子，该因子表示样本的过采样权重。然后，根据比例因子，对每个少数类样本进行合成样本的生成，生成的样本数量与其比例因子成正比，即距离较远的少数类样本生成更多的合成样本[16]。即对于分布密度较低的样本，生成的新样本数量较多，而对于密度较高的样本，生成的新样本数量较少。公式如下：

其中，是需要生成的新样本数量，是总的新样本数量，是少数类样本的数量，是第个少数类样本的分布密度。通过这种自适应的合成抽样方法，ADASYN算法能够有效地增加少数类样本的数量，同时保持了样本分布的多样性，提高了模型的训练效果和准确性。

### 3.2.2 SMOTE

SMOTE算法是一种经典的过采样方法，通过合成新的少数类样本来平衡数据集中不同类别之间的样本数量。其对于每个少数类样本，从其k个最近邻的少数类样本中随机选择一个样本，在选定的少数类样本和其选定的最近邻样本之间，随机生成新的合成样本。合成样本的特征值在两个样本之间的线性插值得到，生成的样本数量可以根据设定的过采样比例来确定[17] 。具体公式如下：

中，是原始的少数类样本，是其最近邻样本，是0到1之间的随机数。通过这种基于样本插值的方法，SMOTE算法能够有效地增加少数类样本的数量，从而实现数据集的平衡化，提高模型的训练效果和准确性。

## 3.3机器学习模型与超参数优化

### 3.3.1 K最近邻算法（KNN）

K最近邻（K Nearest Neighbors）是一种基本的监督学习算法，用于分类和回归问题。对于一个未知类别的样本，KNN算法会找出训练集中与该样本最近的K个邻居，然后根据这K个邻居的类别进行投票，将该样本归类为票数最多的类别[18]。处理回归问题，其会根据这K个邻居的值进行平均或加权平均，作为该样本的预测值，Juna等便提出了一种九层多层感知器（MLP），该感知器与 K-nearest neighbor（KNN）归类器一起使用来检测水质情况，在使用KNN计算器的情况下，所提出的九层MLP模型的水质预测准确率可达0.99[19]，此研究说明了KNN在预测水质方面的可行性。

在选择超参数时，KNN算法最重要的超参数是K值，即选择的邻居数量。较小的K值会导致模型更加敏感，容易受到噪声的影响，而较大的K值则可能会忽略样本间的局部特征[20]。通常可以通过交叉验证等方法来选择合适的K值，除了K值外，KNN算法还可以根据需要选择其他参数，如距离度量方式（如欧氏距离、曼哈顿距离等）以及邻居权重计算方式等。通过调整这些超参数，可以优化KNN模型的性能，提高其预测准确性和泛化能力。

如图一所示，KNN算法当预测一个新样本的类别时，根据它距离最近的 K 个样本点是什么类别来判断该新样本属于哪个类别，如果k=3 （实线圆圈），测试样本（绿点）应该被分类为红色三角形，因为在内圆圈里有2个三角形和1个正方形。如果k=5 （虚线圆圈），它被分配给蓝色方形，因为有3个正方形和2个三角形在外圆圈内。K值选取可以通过交叉验证来选择，通过选择不同k值，计算验证集合的准确率，可以选择准确率最高的k值，从而获得最优模型。

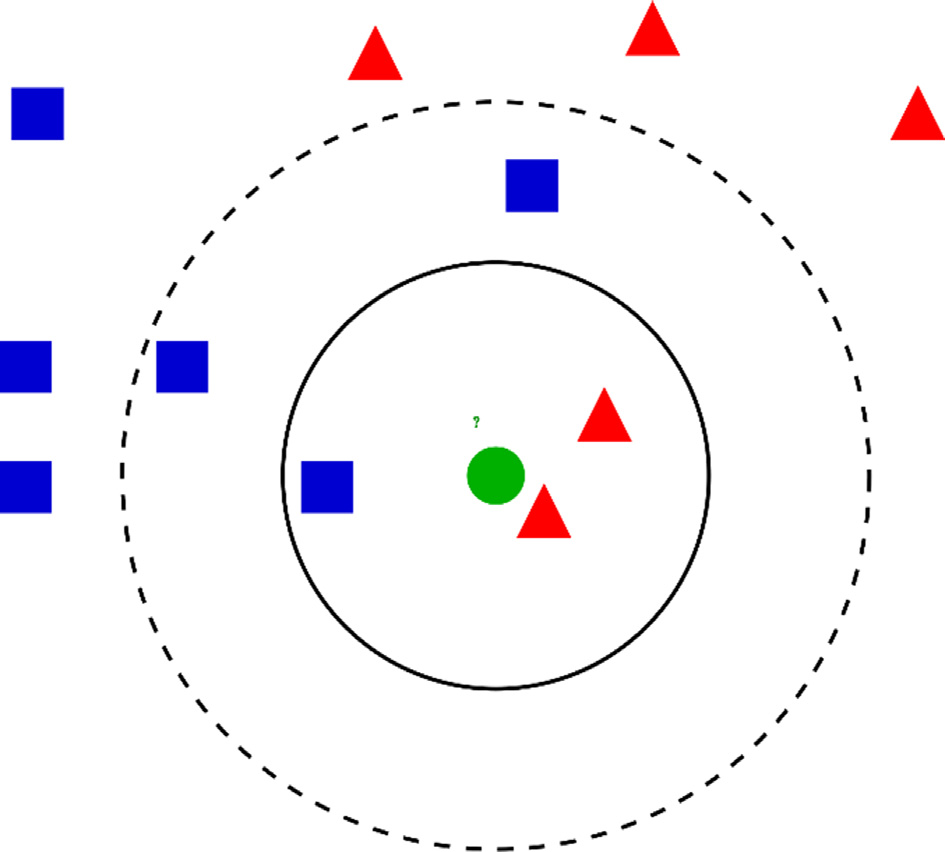


图 1 KNN算法示意图

### 3.3.2 梯度提升树（GDBT）

梯度提升树（Gradient Boosting Decision Tree，GDBT）是一种集成学习方法，通过顺序地训练一系列决策树来提升模型的性能。 GDBT通过将多个弱分类器（决策树）组合成一个强分类器，从而提高模型的准确性和泛化能力，例如Peng等人使用梯度提升决策树（GBDT）和随机森林组成的集合学习模型，对戊型肝炎历史流行病例与环境因素进行训练和预测，结论是集合学习模型的预测效果优于经典模型，说明GDBT相较于普通决策树，模型泛化性能和准确度更高[21]。

在选择超参数时，GDBT模型的关键超参数包括树的深度、学习率、子采样比例等。树的深度决定了模型的复杂度和学习能力，学习率控制了每棵树的贡献程度，子采样比例控制了每棵树训练时使用的样本比例[22]。其通过拟合一个初始模型（一般为简单模型，如平均值）来预测目标值。每一轮迭代都构建一个新的决策树来纠正前一轮模型的残差。将每个决策树的预测结果进行加权求和，得到最终的集成模型。最终的模型预测结果是所有树的预测结果的加权求和：

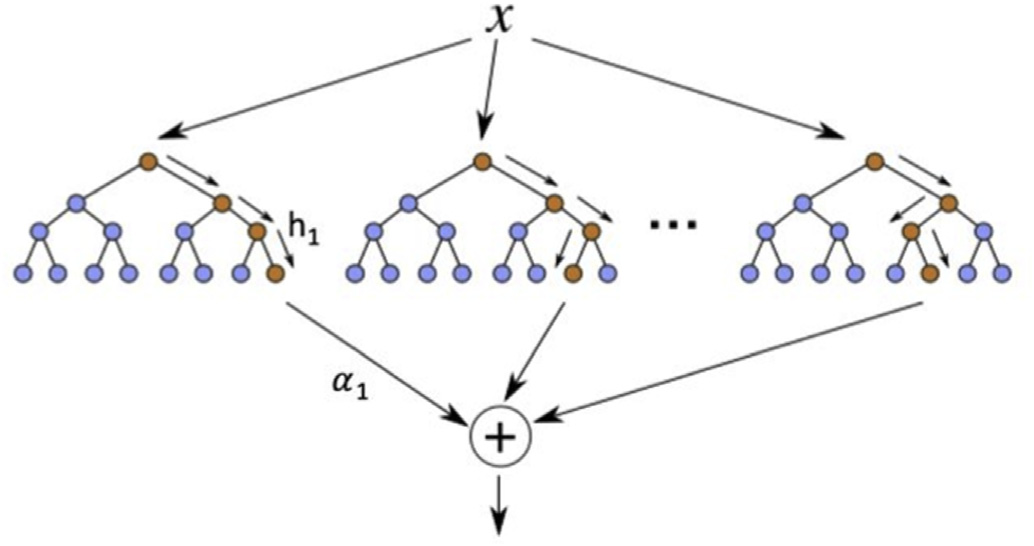
其中，𝑇 是迭代次数， 是学习率， 是第𝑡轮迭代的决策树对样本𝑖的预测值。

图 2 提升决策树:h为弱分类器，a表示分配给h的程度权重。

### 3.3.3 支持向量机（SVM）

支持向量机（Support Vector Machine，SVM）是一种用于分类和回归分析的监督学习模型，通过找到数据间的最优超平面来进行分类。SVM通过找到能够将不同类别的样本分开的最优超平面来进行分类，使得超平面到最近的样本点的距离最大化[23]。张森等人提出了一种将偏最小二乘法和支持向量机（SVM）相耦合的水质预测方法，解决了多重相关性造成预测精度低的问题，并且说明支持向量机建模可以较好地解决高维非线性小样本问题[24]。

在选择超参数时，SVM模型的超参数包括核函数类型、惩罚项参数C、等。核函数类型决定了数据在高维空间中的映射方式，核函数的参数影响了核函数的复杂度和映射效果常用的核函数包括线性核、多项式核和高斯核。惩罚项参数C控制了模型对误分类样本的惩罚程度，较大的C值可能导致模型过拟合，而较小的C值则可能导致模型欠拟合[25]。SVM的目标函数是最大化间隔（Margin），即最小化权重向量𝑤的范数，同时使得所有样本点满足以下约束条件：

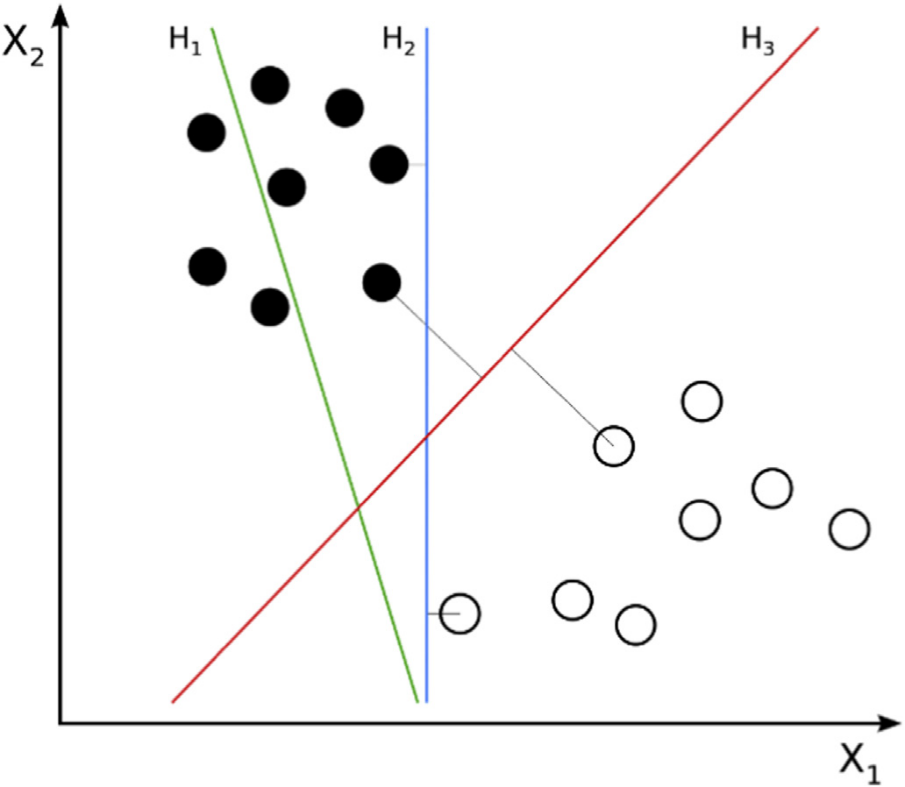
其中， 是松弛变量（Slack Variables），允许一些样本点落在错误的一侧，𝐶 是惩罚项参数，用于控制误分类样本的惩罚程度。

图 3 SVM算法展示。H1不区分类。H2有，但只有很小的差距。H3用最大边距分隔它们。

### 3.3.4人工神经网络（ANN）

人工神经网络（Artificial Neural Network，ANN）是一种模拟人脑神经系统的计算模型，由多个神经元（节点）组成多层网络。输入样本通过网络的各层进行前向传播，经过一系列的线性和非线性变换，最终得到输出层的预测结果。而后通过向后传播，通过比较预测结果与真实标签的差异，利用梯度下降算法来更新网络中的权重和偏置，以最小化损失函数，使模型的预测结果逼近真实标签[26]。

在选择超参数时，ANN模型的超参数包括隐藏层的数量和节点数、学习率、正则化参数等。隐藏层的数量和节点数影响了网络的复杂度和学习能力，而学习率控制了权重更新的步长，正则化参数用于控制模型的复杂度，防止过拟合。

在BP-TLP神经网络中， BP算法通过反向传播误差来更新连接权重，从而使得网络能够逐步优化拟合训练数据，达到降低损失函数的目的。王晓萍等人使用BP神经网络对钱塘江水质指标进行了预测，结果表明，BP神经网络模型对大部分水质指标能够得到较好的预测值,相对误差的绝对值小于6% [27]。说明BP神经网络体系结构可以通过反向传播算法来学习连接权重，以实现对输入数据的有效建模和预测。

# 图示 描述已自动生成第4章 前后端技术栈及其核心代码展示

图 4 BP-TLP神经网络的体系结构

在本章节中，将详细介绍开发所需的技术栈以及相关的核心代码，以及展示页面的功能和交互。在开发过程中，采用了一系列现代化的前端和后端技术，以确保页面的性能、交互性和可扩展性。其中前端页面使用Vue3+ Element UI实现前端渲染，后端使用Django技术实现前端与后端python算法交互。通过 Vue.js 3 和 Element UI 构建前端界面，结合 Django 构建后端逻辑和数据处理，可以实现前后端分离的开发模式，提高了开发效率和代码的可维护性。同时，Vue.js 和 Django 都有活跃的社区和丰富的文档资源，能够为开发人员提供充足的支持和帮助。

## 4.1 前端技术栈以及页面展示

### 4.1.1 前端技术栈

Vue 3 是一个用于构建交互式用户界面的 JavaScript 框架，拥有丰富的生态系统，包括路由管理、状态管理和 UI 库等，能够满足各种需求。其核心概念之一是组件化开发，即将页面拆分成多个可复用的组件，每个组件负责自己的视图和逻辑，这种方式提高了代码的可维护性和复用性。借助 Vue 3，开发人员能够快速构建交互性强、拓展性高的用户界面。同时，结合使用Element UI组件库，其是一套基于 Vue.js 的 UI 组件库，提供了丰富的主题定制和样式配置选项，开发人员可以根据项目需求进行定制和扩展，根据提供的API文档，可以轻松的为组件添加合适的属性，事件等，可以快速构建一个简洁明了的前端页面。

### 4.1.2 页面展示

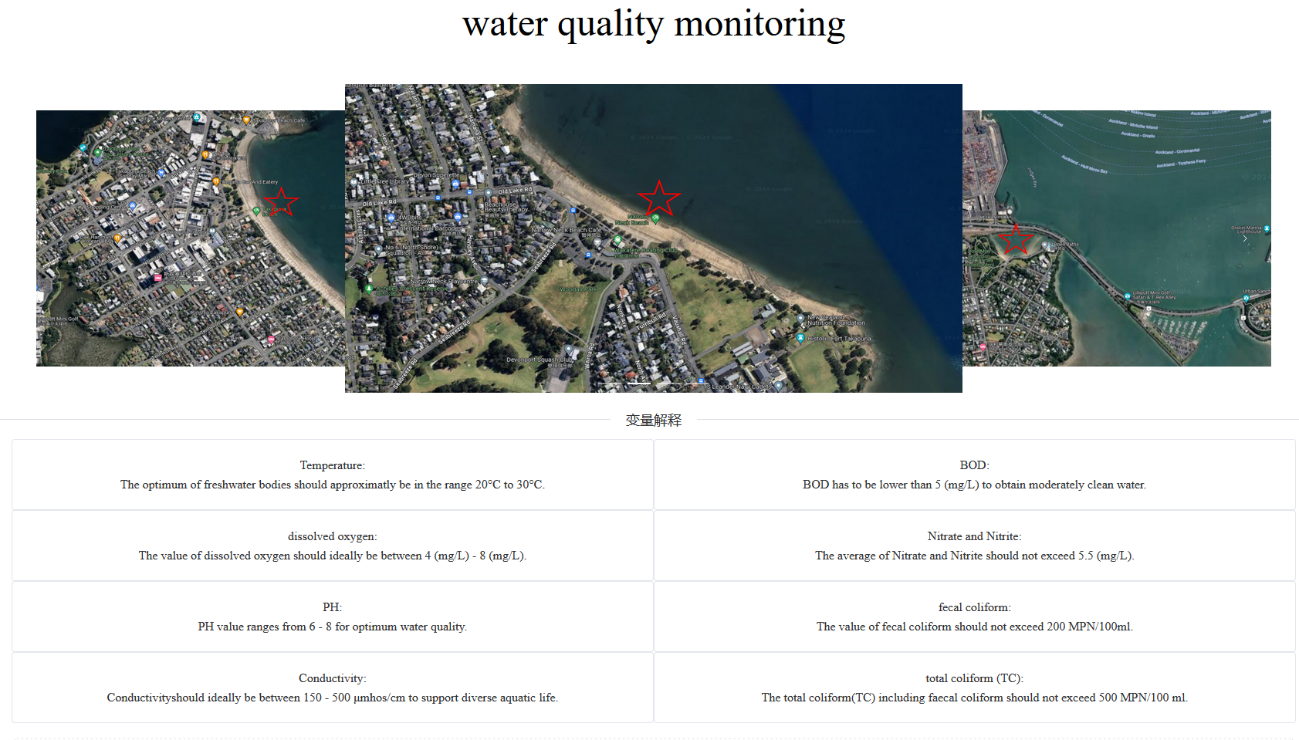
在图5中，以走马灯的形式展示了数据集获取地点。紧接着，提供了关于水质检测的关键指标，包括温度、溶氧量、PH值、导电率、生化需氧量、硝酸盐和亚硝酸盐的平均值，以及粪大肠菌和总大肠菌等八大特征。通过给出优质水源的相关指标范围，能够确定采集的数据集中水质优劣的分布情况，并进行过采样处理，为模型算法提供所需的训练集。

图 5 展示采样地点和变量解释

图片包含 表格

描述已自动生成如图6所示，本页面功能为上传原始数据，用户可通过点击文件上传或拖拽方式上传原始数据集。上传完成后，页面将显示上传文件名称，并弹出上传完成状态提示。请注意，该上传按钮仅支持上传一个文件。上传完成后，页面下方将显示所有数据信息。用户可以选择分页器来设置每页显示的条目个数或跳转到指定的页数，以便查看全部数据并实现数据区域的跳转。

图 6 导入csv文件以及数据展示

图表

描述已自动生成如图7所示，该页面展示了数据过采样功能，提供了两个按钮，分别实现了ADASYN和SMOTE算法的功能。在页面下方展示了处理前后的数据分布情况，其中，0表示不符合优质水源指标的数据数量，而1表示所有指标达标优质水源范围的数据数量。通过对比处理前后的数据分布，可以看出，处理后值为1的数据量增加至与值为0的数据量相等，实现了数据平衡处理。同时，页面下方文字展示了总数据量大小的前后变化。

图 7 过采样前后数据图表展示

如图8所示，页面下方展示了KNN算法对过采样处理后的数据进行训练的功能。用户可以通过对不同K值进行训练，并根据交叉验证集的准确率选择最优K值。页面上可视化展示了不同K值下的模型准确率，并显示了最优K值下模型的精确率、召回率以及F1得分的柱状图，以便了解模型最终的训练性能。同时，页面下方的注释显示了最大准确率、最优K值以及模型类型的相关信息。

图表

描述已自动生成

图 8 KNN算法调参过程及结果展示

图表

描述已自动生成如图9所示，页面下方展示了GDBT算法对过采样处理后的数据进行训练的功能。用户可以通过对不同学习率进行训练，并根据交叉验证集的准确率选择最优学习率。页面上可视化展示了不同学习率下的模型准确率，并显示了最优学习率下模型的精确率、召回率以及F1得分的柱状图，以便了解模型最终的训练性能。同时，页面下方的注释显示了最大准确率、最优学习率以及模型类型的相关信息。

图 9 GDBT算法调参过程及结果展示

如图10所示下方展示了SVM算法对过采样处理后的数据进行训练，本功能可以通过对不同C值进行训练，根据交叉验证集的准确率选择最优C值，在按钮下方可视化展示不同C值下的模型准确率，同时显示了最优C值下模型的精确率，召回率以及F1得分的柱状图，以便了解模型最终的训练性能。下方的注释显示了图表

描述已自动生成最大准确率，最优C值以及模型类型的相关信息。

图 10 SVM算法调参过程及结果展示

图表

描述已自动生成如图11所示，页面下方展示了ANN算法对过采样处理后的数据进行训练的功能。用户可以通过对不同学习率进行训练，并根据交叉验证集的准确率选择最优学习率。页面上可视化展示了不同学习率下的模型准确率，并显示了最优学习率下模型的精确率、召回率以及F1得分的柱状图，以便了解模型最终的训练性能。同时，页面下方的注释显示了最大准确率、最优学习率以及模型类型的相关信息。

图 11 ANN算法调参工程及结果展示

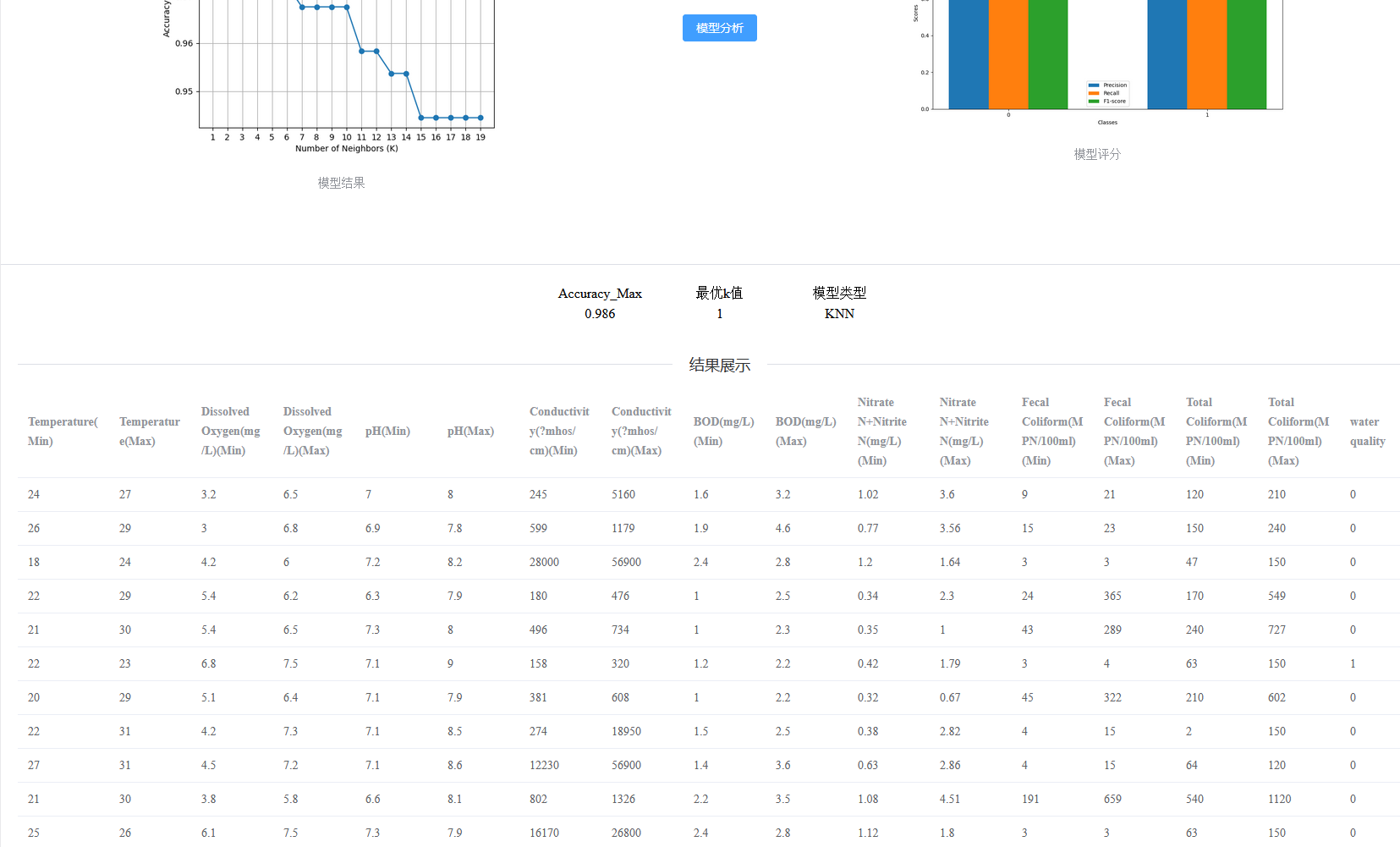
点击相应按钮后，实现相应模型训练过平衡后的数据集后，点击按钮模型分析，用户可以上传新的 CSV 数据文件，对数据集进行数据预处理。然后，使用带有最优超参数的相应模型进行预测分析。模型分析成功后，页面将显示预测后的数据集，如图12所示。

图 12 模型预测结果

## 4.2 后端技术栈以及核心代码展示

### 4.2.1 后端技术栈

后端使用 Django 框架实现后端业务逻辑，采用 MVC（Model-View-Controller）架构模式，将应用程序分为模型、视图和控制器三层。在后端实现自适应过采样算法（ADASYN）和 SMOTE 算法，提供前端接口，对上传的文件进行过采样处理，以解决数据不平衡问题。此外，通过调用相应的机器学习库，对经过过采样处理的数据进行训练，并提供接口供前端调用进行数据预测和分析。在 Django 中，路由功能由URL配置和视图函数组成，URL 配置定义了 URL 和视图函数之间的映射关系，视图函数则负责处理用户请求并返回相应的响应。总体来说，Django可以有效的为前端提供python算法接口后端架构设计旨在实现系统的业务逻辑和数据处理功能，保证系统的稳定性和可扩展性。

### 4.2.2 核心代码展示

1. **路由功能代码**

下面代码定义了Django项目中的URL配置，主要用于处理与模型分析相关的HTTP请求。Django 将根据URL配置找到相应的视图函数，并将请求传递给该视图函数进行处理，最终返回响应给用户，通过这些路径和关联的视图函数，用户可以实现对上传文件的数据分析、过采样处理以及各种机器学习模型的训练和结果展示等功能。

|  |
| --- |
| from django.urls import path, include  from .views import (analyze\_file, analyze\_ada, analyze\_knn, analyze\_model\_knn, analyze\_smote,analyze\_gdbt,analyze\_model\_gdbt,analyze\_svm, analyze\_model\_svm, analyze\_ann, analyze\_model\_ann)  urlpatterns = [  path('analyze', analyze\_file, name='analyze\_file'),  path('ADASYN', analyze\_ada, name='analyze\_ada'),  path('smote', analyze\_smote, name='analyze\_smote'),  path('KNN', analyze\_knn, name='analyze\_knn'),  path('model\_knn', analyze\_model\_knn, name='analyze\_model\_knn'),  path('GDBT', analyze\_gdbt, name='analyze\_gdbt'),  path('model\_gdbt', analyze\_model\_gdbt, name='analyze\_model\_gdbt'),  path('SVM', analyze\_svm, name='analyze\_svm'),  path('model\_svm', analyze\_model\_svm, name='analyze\_model\_svm'),  path('ANN', analyze\_ann, name='analyze\_svm'),  path('model\_ann', analyze\_model\_ann, name='analyze\_model\_svm'),  # Other URLs...  ] |

1. **接收文件处理代码**

下列代码实现了处理上传文件的功能，接收包含文件的 POST 请求，从请求中获取上传的 CSV 文件，然后使用 pandas 库读取文件数据。接着，调用传入的处理函数对数据进行处理，该处理函数应返回一个包含 Base64 编码的图片、数据行数以及处理后的数据框。最后，将处理结果以 JSON 格式返回给客户端。如果请求不包含文件或文件未上传，则返回错误信息，并返回状态码 400。其中图片是模型参数与准确率之间关系示意图以及模型结果精确率，召回率以及F1得分柱状图，dataframe返回自适应算法或机器学习模型处理后的数据集

|  |
| --- |
| def handle\_uploaded\_file(request, processing\_func):  if request.method == 'POST' and request.FILES.get('file'):  uploaded\_file = request.FILES['file']  data = pd.read\_csv(uploaded\_file)  plot\_base64, number, dataframe = processing\_func(data)  json\_data = dataframe.to\_json(orient='records')  return JsonResponse({'plot': plot\_base64, 'row': number, 'value': json\_data})  else:  return JsonResponse({'error': 'File not provided'}, status=400) |

1. **数据预处理以及特征工程**

第一段代码定义了数据集所需要的特征值有哪些，以方便后续对数据集进行特征工程操作。第二段代码定义了一个名为 `condition\_func` 的函数，用于根据一系列水质指标判断水质是否符合优质水源的条件。通过逐一比较每个指标的取值范围是否在优质水源标准内，生成一个布尔值条件，表示该行数据是否符合条件。

第三段代码实现了数据预处理的功能，包括对原始数据进行清洗和处理，使其适合用于后续的分析和建模。清洗过程包括将特殊字符替换为NaN值、去除不需要的字符、将数据转换为浮点型，并且删除包含NaN值的行。这样可以确保数据质量和完整性，为后续的数据分析和建模提供可靠的数据基础。

|  |
| --- |
| features = ['Temperature(Min)', 'Temperature(Max)', 'Dissolved Oxygen(mg/L)(Min)', 'Dissolved Oxygen(mg/L)(Max)', 'pH(Min)','pH(Max)', 'Conductivity(?mhos/cm)(Min)', 'Conductivity(?mhos/cm)(Max)', 'BOD(mg/L)(Min)', 'BOD(mg/L)(Max)','Nitrate N+Nitrite N(mg/L)(Min)', 'Nitrate N+Nitrite N(mg/L)(Max)', 'Fecal Coliform(MPN/100ml)(Min)', 'Fecal Coliform(MPN/100ml)(Max)', 'Total Coliform(MPN/100ml)(Min)', 'Total Coliform(MPN/100ml)(Max)']  def condition\_func(row):  condition=((row['Temperature(Min)']>=20)& (row [ 'Temperature(Max) '] <= 30) &  ( row ['Dissolved Oxygen(mg/L)(Min)'] >= 4) & (row ['Dissolved Oxygen(mg/L)(Max)'] <= 8) &  ( row ['pH(Min)'] >= 6) & (row['pH(Max)'] <= 8) & (row ['Conductivity(?mhos/cm)(Min)'] >= 150) &  (row ['Conductivity(?mhos/cm)(Max)'] <= 500) & ( row ['BOD(mg/L)(Max)'] <= 5)&  (row ['Nitrate N+Nitrite N(mg/L)(Max)'] <= 5.5) & (row ['Fecal Coliform (MPN/100ml)(Max) '] <= 200) &  (row ['Total Coliform(MPN/100ml)(Max)'] <= 500))  return condition  def data\_preprocessing(original\_df):  original\_df = original\_df.replace('-', np.nan)  original\_df = original\_df.replace('\n4', '', regex=True)  original\_df = original\_df.replace('\n', ' ', regex=True)  original\_df = original\_df.astype(float)  original\_df.dropna(inplace=True)  return original\_df |

1. **ADASYN算法实现**

下面三段代码共同实现了 ADASYN 过采样处理的功能。首先，adasyn\_processing\_func 函数接收原始数据并调用 adasyn\_processing 函数进行过采样处理，然后根据处理后的数据计算水质情况的分布，并调用 generate\_plot 函数生成数据分布图，并返回图表的 Base64 编码、数据行数和处理后的数据框。generate\_plot 函数负责绘制数据分布图，并将图表转换为 Base64 编码。adasyn\_processing 函数实现了 ADASYN 过采样算法的具体逻辑，首先对原始数据进行预处理，然后调用 ADASYN 进行过采样处理，最后将处理后的特征和标签合并为一个数据框并返回，SMOTE算法实现代码同理。

|  |
| --- |
| def adasyn\_processing\_func(data):  df\_resampled = adasyn\_processing(data)  df\_resampled['water quality'] = df\_resampled.apply(condition\_func, axis=1). astype(int)  data\_distribution = df\_resampled['water quality'].value\_counts()  return generate\_plot(data\_distribution), df\_resampled.shape[0], df\_resampled  def generate\_plot(data\_distribution):  # Plot data distribution  plt.bar(data\_distribution.index, data\_distribution.values)  plt.xlabel('Category')  plt.ylabel('Count')  plt.title('Data Distribution')  plt.xticks([0, 1], ['0', '1'])  # Convert plot to base64  buffer = io.BytesIO()  plt.savefig(buffer, format='png')  buffer.seek(0)  plot\_base64 = base64.b64encode(buffer.getvalue()).decode()  plt.close()  return plot\_base64  def adasyn\_processing(data):  original\_df = data\_preprocessing(data)  df\_wq = original\_df.copy(deep=True)  df\_wq['water quality'] = df\_wq.apply(condition\_func, axis=1).astype(int)  oversample = ADASYN(n\_neighbors=3, random\_state=40)  x = df\_wq[features]  y = df\_wq['water quality']  x\_resampled, y\_resampled = oversample.fit\_resample(x, y)  x\_df = pd.DataFrame(x\_resampled, columns=features)  y\_df = pd.DataFrame(y\_resampled, columns=['water quality'])  df\_resampled = pd.concat([x\_df, y\_df], axis=1)  return df\_resampled |

1. **KNN算法实现代码**

第一段代码实现了KNN模型的训练过程。首先，调用ADASYN算法对数据进行过采样处理，然后将处理后的数据拆分为训练集和测试集，并使用不同的K值（1到20）进行KNN模型的训练，并评估每个K值下模型的准确率。接着，绘制了K值与准确率之间的关系图，并找出在测试集上准确率最高的K值。最后，使用最优K值重新训练KNN模型，并对整个数据集进行预测，并生成分类报告的可视化图表。

第二段代码则实现了对训练好的KNN模型进行分析的功能。首先，同样调用ADASYN算法对数据进行过采样处理，然后使用训练过的最优KNN模型对数据进行预测，并将预测结果与原数据合并，返回包含预测结果的数据框。其他机器学习实现同理，SVM会使用不同C值对过采样数据进行评估，选择准确率最高时的C值，GDBT和ANN则会对学习率进行调优评估，最终代均实现使用不同的机器学习模型对过采样数据集进行训练，而后使用最优模型对新数据集进行分析预测，将结果返回至前端。

|  |
| --- |
| def knn\_train(data):  df\_resampled = adasyn\_processing(data)  x = df\_resampled[features]  y = df\_resampled['water quality']  x\_train, x\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size=0.2, random\_state=42)  k\_values = range(1, 20)  accuracies = []  for k in k\_values:  knn = neighbors.KNeighborsClassifier(n\_neighbors=k)  knn.fit(x\_train, y\_train)  y\_pred = knn.predict(x\_test)  accuracy = accuracy\_score(y\_test, y\_pred)  accuracies.append(accuracy)  plt.plot(k\_values, accuracies, marker='o')  plt.xlabel('Number of Neighbors (K)')  plt.ylabel('Accuracy')  plt.title('KNN Performance')  plt.xticks(k\_values)  plt.grid(True)  # Convert plot to base64  buffer = io.BytesIO()  plt.savefig(buffer, format='png')  buffer.seek(0)  plot\_base64 = base64.b64encode(buffer.getvalue()).decode()  plt.close()  knn\_best= neighbors.KNeighborsClassifier (k\_values[accuracies.index (max (accuracies))])  knn\_best.fit(x, y)  y\_hat = knn\_best.predict(x)  report = classification\_report(y, y\_hat, output\_dict=True)  plot2\_base64 = report\_plot(report)  return plot\_base64,round(max(accuracies),3), k\_values[accuracies. Index (max (accuracies))] , knn\_best, plot2\_base64  def knn\_model\_analyse(data):  df\_resampled = adasyn\_processing(data)  x = df\_resampled[features]  \_, \_, \_, knn\_best, \_ = knn\_train(data)  y\_pred = knn\_best.predict(x)  x\_df = pd.DataFrame(x, columns=features)  y\_df = pd.DataFrame(y\_pred, columns=['water quality'])  df\_resampled = pd.concat([x\_df, y\_df], axis=1)  return df\_resampled |

# 第5章 实验结果与分析

## 5.1 过采样算法实现结果

过采样算法的实现结果分析表明，ADASYN 和 SMOTE 算法都成功地将原始数据集中的不平衡样本增加到了更平衡的状态，从而为后续的机器学习模型提供了更好的数据基础。

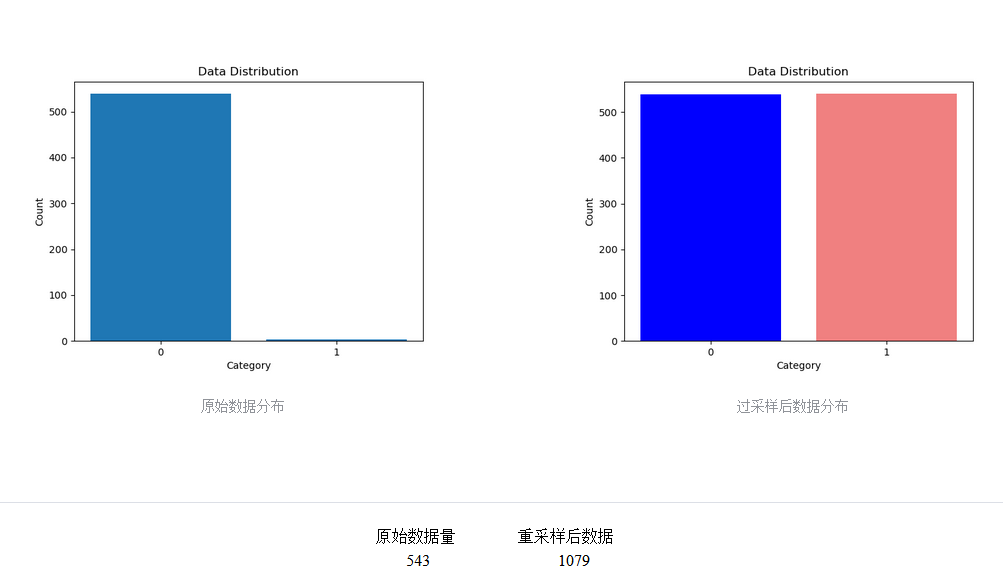
通过 ADASYN 算法，原始数据集的样本数量从 543 增长至 1079，使得水源质量好坏的分布更加平衡。这种平衡的数据分布有助于机器学习模型更准确地学习和预测样本的类别，提高了模型的泛化能力和预测准确性。

图 13 ADASYN算法结果

同样地，SMOTE 算法也取得了类似的效果，将原始数据集的样本数量由 543 增长至 1078，使得数据量更加平衡。通过直方图可视化分析，可以清晰地观察到样本数量的平衡情况，从而确保模型在训练和预测过程中能够充分考虑到不同类别样本的影响，提高了模型的稳定性和可靠性。

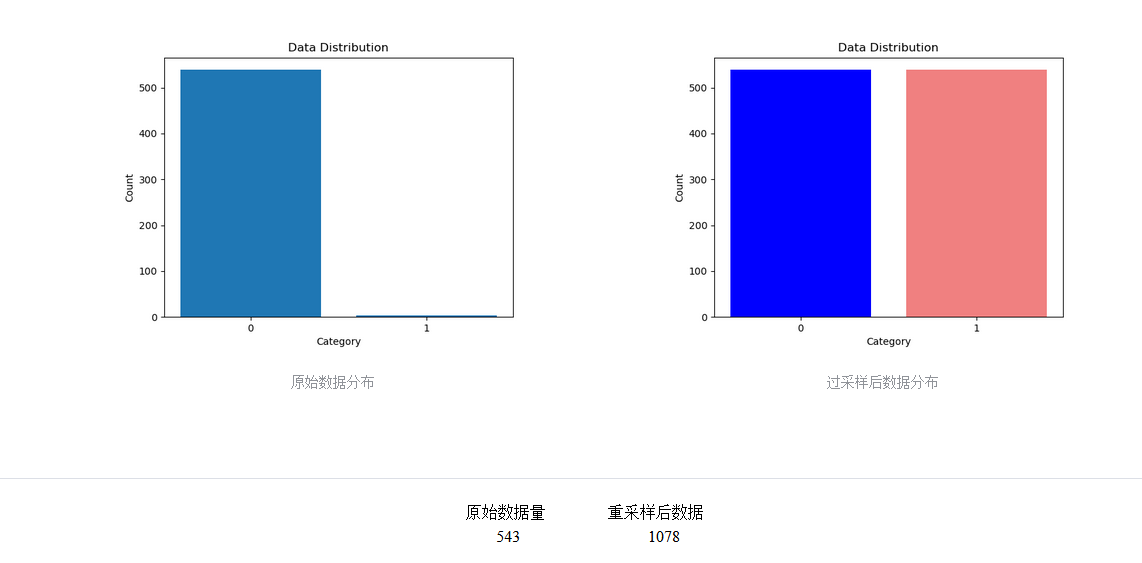


图 14 SMOTE算法结果

综上所述，ADASYN 和 SMOTE 算法的实现结果表明其在解决数据不平衡问题上具有显著的效果，为后续的机器学习模型提供了更可靠的数据基础，从而提高了模型的性能和预测能力。

## 5.2机器学模型分析结果

图表, 折线图

描述已自动生成在交叉验证过程中，KNN模型的最优K值为1或2，其准确率达到了0.986。KNN模型是根据最近邻居的标签对数据进行分类。KNN模型无需训练过程，适用于小型数据集。然而，KNN模型在处理大型数据集时可能会面临计算复杂度高的问题，而且对异常值和噪声敏感，如图所示，为KNN算法的可视化结果。

图 15 KNN算法K值与准确率关系图

图表, 条形图

描述已自动生成

图 16 KNN模型评估指标

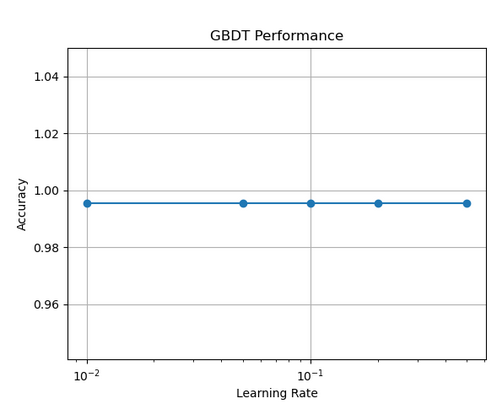
GDBT模型的最优学习率为0.01，其交叉验证准确率达到了0.995。GDBT模型是一种集成学习算法，通过迭代地训练多个决策树来改进模型性能。它具有高准确性、能够处理大型数据集和高维特征的能力。然而，GDBT模型也存在一些缺点，如对于超参数的调优较为繁琐，训练时间较长，容易过拟合。

图 17 GDBT算法学习率与准确率关系图

图表, 条形图

描述已自动生成

图 18 GDBT算法评估指标

图表, 折线图

描述已自动生成SVM模型的最优C值为1，交叉验证准确率达到了1。SVM模型是一种二分类模型，通过找到最佳的超平面来实现数据的分类。它具有良好的泛化能力和对异常值的鲁棒性，适用于高维数据集和小样本问题。然而，SVM模型在处理大型数据集时可能会面临内存消耗过大的问题，而且对于超参数的选择比较敏感。

图 19 SVM算法C值与准确率关系图

图表, 条形图

描述已自动生成

图 20 SVM模型评估指标

图表, 折线图

描述已自动生成ANN模型的最优学习率为0.01，交叉验证准确率达到了0.968。ANN模型是一种深度学习模型，具有强大的非线性建模能力和适应性。它适用于复杂的数据模式识别和特征学习，能够处理大型数据集和高维特征。然而，ANN模型也存在训练时间长、需要大量数据和计算资源、调参困难等缺点。

图 21 ANN算法学习率与准确率关系图

图表, 条形图

描述已自动生成

图 22 ANN模型评估指标

# 第6章 结论与展望

## 6.1 系统总结

本论文致力于开发一个水质检测系统，旨在利用机器学习技术对水质数据进行分析和预测，以帮助监测水体的质量并提前预防潜在的污染事件。本文介绍了系统的设计与实现，重点关注了过采样模型算法和机器学习模型的应用。使用了ADASYN和SMOTE算法对水质数据集进行了平衡处理。这两种过采样算法能有效地增加少数类样本，从而改善数据集的类别分布，提高了后续机器学习模型的性能。其次，采用了不同的机器学习模型，包括ANN、GDBT和SVM，对平衡后的数据集进行了训练和预测。通过选择合适的超参数，优化了模型的性能，并提高了水质数据的预测准确度。

在系统实现方面，使用了Vue3和Element UI作为前端框架，利用它们提供的丰富组件和功能实现了用户友好的前端界面。而后端部分则采用了Django技术，实现了前端与后端的数据交互和算法调用，保证了系统的稳定性和可靠性。

## 6.2 系统不足与展望

尽管此次毕业设计中取得了一定的成果，但仍存在一些需要改进的地方。系统目前存在几个值得改进的地方，一方面，对于数据集来说，系统对于在特征选择方面存在局限性，只能根据指定的特征进行水质分析，无法灵活选择更多特征值来进行分析，这限制了系统的适用性和灵活性，此外，由于数据集规模较小，训练模型的泛化性能受到限制。另一方面，在处理高维度数据时，调节不同机器学习模型的超参数可能会消耗大量的模型运行时间，影响系统的实时性和响应速度，尤其是SVM模型，在以线性为内核函数时，会花费较长时间训练出不同C值与准确率之间的相关性。此外，系统对于模型运行评估指标的分析还不够充分，缺乏系统化的评估和反馈机制，需要依赖操作者个人分析精确率，召回率以及F1得分三者指标，这降低了系统的自动化程度和用户体验，普通人难以根据各指标评估系统结果。

为了改进系统，未来可以优化系统的特征选择模块，引入更灵活的特征选择机制，使用户能够根据实际需求自定义特征进行水质分析，例如在前端页面表单中选择合适的特征值，发送至后端对原始数据进行预处理，从而实现对特定特征进行分析预测水资源质量。针对处理高维度数据时遇到的时间成本较高的问题，可以通过优化模型训练和调参过程，引入更高效的算法和调参方法，以减少模型运行时间，提高系统的实时性。对于模型运行评估指标难以理解的问题，可以通过完善系统的评估指标分析模块，引入更多的评估指标和可视化工具，例如 PR 曲线、ROC 曲线以及 AUC 曲线等，为用户提供更全面、直观的模型性能评估结果，提升系统的可操作性。

总的来说，此水质监测系统为水质检测领域的研究和应用提供了一个基础平台，为未来进一步深入研究和实践奠定了基础。通过不断改进和优化，相信这个系统能够在实际应用中发挥更大的作用，为保护水资源和环境做出更大的贡献。

# 参考文献

[1] 王晓辉, et al., *水质生物毒性检测方法研究进展.* 河北工业科技, 2007(01): p. 58-62.

[2] Ahmed, U., et al., *Water quality monitoring: from conventional to emerging technologies.* Water Supply, 2019. **20**(1): p. 28-45 %@ 1606-9749.

[3] Olatinwo, S.O., T.H. Joubert, and D.D. Olatinwo, *Water Quality Assessment Tool for On-site Water Quality Monitoring.* IEEE Sensors Journal, 2024: p. 1-1.

[4] Patel, J., et al., *A Machine Learning-Based Water Potability Prediction Model by Using Synthetic Minority Oversampling Technique and Explainable AI.* Computational Intelligence and Neuroscience, 2022. **2022**: p. 9283293.

[5] Sahu, M., et al., *Prediction of water quality index using neuro fuzzy inference system.* Water Quality, Exposure and Health, 2011. **3**: p. 175-191.

[6] Wong, B.P. and B. Kerkez, *Adaptive measurements of urban runoff quality.* Water Resources Research, 2016. **52**(11): p. 8986-9000.

[7] Shu, T., et al., *An Energy Efficient Adaptive Sampling Algorithm in a Sensor Network for Automated Water Quality Monitoring.* Sensors, 2017. **17**(11): p. 2551.

[8] Xu, T., G. Coco, and M. Neale, *A predictive model of recreational water quality based on adaptive synthetic sampling algorithms and machine learning.* Water research, 2020. **177**: p. 115788.

[9] Liu, L. and H. Zhou, *Investigation and assessment of volatile organic compounds in water sources in China.* Environmental Monitoring and Assessment, 2011. **173**(1): p. 825-836.

[10] Zhang, W., et al., *Effects of rainfall on microbial water quality on Qingdao No. 1 Bathing Beach, China.* Marine Pollution Bulletin, 2013. **66**(1): p. 185-190.

[11] 彭子康, et al., *基于机器学习和图像处理的水质检测方法.* 自动化应用, 2023. **64**(10): p. 188-191.

[12] 凌煦, et al., *基于ADASYN-XGBoost算法的光伏出力预测研究.* 中国农村水利水电: p. 1-9.

[13] 李瑞平 and 朱俊杰, *基于改进Borderline-Smote-GBDT的冠心病预测.* 中国医学物理学杂志, 2023. **40**(10): p. 1278-1284.

[14] 陈虹, et al., *改进ADASYN-SDA的入侵检测模型研究.* 计算机工程与应用, 2020. **56**(02): p. 97-105.

[15] Tyagi, S., et al., *Water quality assessment in terms of water quality index.* American Journal of water resources, 2013. **1**(3): p. 34-38.

[16] He, H., et al. *ADASYN: Adaptive synthetic sampling approach for imbalanced learning*. in *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*. 2008. Ieee.

[17] Chawla, N.V., et al., *SMOTE: synthetic minority over-sampling technique.* Journal of artificial intelligence research, 2002. **16**: p. 321-357.

[18] 闭小梅 and 闭瑞华, *KNN算法综述.* 科技创新导报, 2009(14): p. 31.

[19] Juna, A., et al., *Water Quality Prediction Using KNN Imputer and Multilayer Perceptron.* Water, 2022. **14**(17): p. 2592.

[20] Guo, G., et al. *KNN Model-Based Approach in Classification*. in *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*. 2003. Berlin, Heidelberg: Springer Berlin Heidelberg.

[21] Peng, T., et al., *The Prediction of Hepatitis E through Ensemble Learning.* International Journal of Environmental Research and Public Health, 2021. **18**(1): p. 159.

[22] Lv, Q., *Hyperparameter tuning of GDBT models for prediction of heart disease*. International Conference on Electronic Information Engineering and Computer Science (EIECS 2022). Vol. 12602. 2023: SPIE.

[23] Kazemi, A., et al. *Two-Layer SVM, Towards Deep Statistical Learning*. in *2022 International Engineering Conference on Electrical, Energy, and Artificial Intelligence (EICEEAI)*. 2022.

[24] 张森, et al., *基于偏最小二乘回归和SVM的水质预测.* 计算机工程与应用, 2015. **51**(15): p. 249-254.

[25] Duan, K.-B. and S.S. Keerthi. *Which is the best multiclass SVM method? An empirical study*. in *International workshop on multiple classifier systems*. 2005. Springer.

[26] Agatonovic-Kustrin, S. and R. Beresford, *Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research.* Journal of pharmaceutical and biomedical analysis, 2000. **22**(5): p. 717-727.

[27] 王晓萍, 孙继洋, and 金鑫, *基于BP神经网络的钱塘江水质指标的预测.* 浙江大学学报(工学版), 2007(02): p. 361-364.

# 致谢

落笔至此，内心翻涌。四年时光，转瞬即逝，回首间纵有众多留恋不舍，难忘自己初入校园满眼期待，难忘父母悉心问候支持鼓励，难忘导师学业困境指点迷津，难忘益友相识相知苦乐与共。学浅才输，此文乃学业之终，意义非凡于己，故怀敬业之心，思辨、斟酌、思虑数日，敬师恩似海、益友相伴，敬父母之劳苦，吾学业之终成。

感谢许汀汀老师，在学术路上对我的指引和帮助，对每次研究过程中给予指导与细心的校对，给予我未来学术生涯中更多的机遇与可能。在过往的论文撰写与论文拒稿，许老师的耐心指导校正和积极的鼓励支持，总让我遇挫后重拾对研究的信心与热情。感谢许老师不吝赐教，诲人不倦，涓涓师恩，铭记于心。

感谢我的同窗挚友和女友。朝夕相处的四年时光，相互帮助，共同进步，与优秀的你们相遇是我生活中不可夺得幸运。感谢女友在我艰难低谷时期的陪伴，给予我的鼓励和关怀。

杨绛先生说过读书不是为了文凭和发财，而是成为个有温度懂情趣会思考的人，路漫漫其修远兮，吾将上下而求索，也祝愿我在往后的生活工作学习中能够不忘初心直勇敢的走下去。剑未配妥，出门便已是江湖流光易逝，终有别时，衷心祝愿大家未米可期。