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	**34082003**
学　号	**2020215139**
指导教师	许汀汀　副教授
答 辩 组 负 责 人	万邦睿　高级工程师

二〇二四年六月

重庆邮电大学教务处制

___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和机器学习的水质检测系统研发》，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　2020

专业　软件工程

班级　34082003

承诺人签名

2024　年 5 月 31　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：机器学习模型；过采样算法；水质检测； 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: Machine learning model; Oversampling algorithm; Water quality monitoring; Django

目录

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，不能满足水质检测所需的准确性。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 GDBT 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

水质监测是环境监测的重要组成部分，国外大量研究集中在通过创新算法提升采样效率和准确性。Sahu 等人提出了一种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于获得准确结果的重要性[5]。在自适应采样算法领域，研究者们已经探索了其在多个领域的应用。Wong 等人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 等人介绍了一种用于预测娱乐水质的自适应合成采样算法（ADASYN），该算法结合了 K-均值近邻和支持向量机等机器学习技术，突出了自适应采样在提高水质预测准确性方面的重要性[8]。这些研究展示了自适应采样算法在提升水质监测和预测准确性方面的显著效果，研究重点涵盖了并行聚类、模糊推理系统和机器学习技术等多个领域，旨在优化传感器网络、节约能源并提高水质监测系统的整体质量。

总之，自适应采样算法在水质监测中的应用不仅提高了数据采集的效率和准确性，还显著降低了能源消耗，为环境保护和管理提供了强有力的技术支持。这些研究成果为后续研究提供了重要参考，推动了水质监测技术的持续发展。

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 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。

2.1 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。

表 2.1 变量解释表

变量名称	变量类型	变量类别	数据总量
温度（°C）	解释变量	连续变量	543
溶解氧（mg/L）	解释变量	连续变量	
pH 值	解释变量	连续变量	
电导率（μmhos/cm）	解释变量	连续变量	
生化需氧量（mg/L）	解释变量	连续变量	

硝酸盐和亚硝酸盐的平均值（mg/L）	解释变量	连续变量
粪大肠菌（MPN/100ml）	解释变量	连续变量
总大肠菌（MPN/100ml）	解释变量	连续变量
水质好坏	被解释变量	离散变量

2.2 前后端框架

2.2.1 前端技术栈

Vue3 是一个 JavaScript 框架，用于创建互动用户界面，具有丰富的生态系统，如路由、状态管理和 UI 库等，可以满足不同需求。其核心理念之一是组件化开发，即将页面分解为多个可复用组件，每个组件负责自己的视图和逻辑，从而提高了代码的维护性和复用性。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。

2.2.2 后端技术栈

Django 框架负责构建后台服务，遵循 MVC 设计模式，将应用分割为三个主要组件：模型、视图与控制器。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 配置与视图功能协同工作，确立了网络地址到视图处理的对应规则。视图层的主要职责是接收请求并生成恰当的反馈。总体来说，Django 的后端架构主要提供支持前端的 Python 算法接口，旨在维护系统的业务逻辑与数据操作，同时确保系统运行的稳定与可增长性。

2.3 自适应过采样算法

2.3.1 ADASYN

ADASYN (Adaptive Synthetic Sampling Approach for Imbalanced Learning) 是一种自适应合成抽样方法，专门用于处理不平衡数据集的过采样问题。该算法通过生成新的合成样本来平衡少数类和多数类样本的比例，从而提高模型的训练效果和准确性。其基本原理是通过计算少数类样本与其最近邻少数类样本的距离，确定每个少数类样本的过采样比例，进而生成新的合成样本。

如图 3.1，具体而言，对于每个少数类样本，ADASYN 算法首先计算其与 K 个最近邻少数类样本的距离，形成一个邻域集合。根据该集合中的多数类样本的比例，确定该少数类样本的过采样权重。权重越大，表示该少数类样本所在区域的多数类样本比例越高，所需生成的合成样本数量越多[16]，公式如下：

$$N_i = round\left(N \times \frac{G_{(i)}}{\sum_{i=1}^{m} G_{(i)}}\right) \tag{2.1}$$

其中，N_i是需要生成的新样本数量，N是总的新样本数量，m是少数类样本的数量，$G_{(i)}$是第i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，能够平衡少数类样本的数量，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。

ADASYN 的主要优点包括：通过生成新的少数类样本，显著平衡数据集中少数类和多数类样本的比例。其中合成样本基于少数类样本和其邻域样本生成，确保了新样本的多样性和分布一致性。同时，该算法能够自适应地调整每个少数类样本的过采样比例，针对不同的样本分布情况进行调整，避免了过度或不足采样的问题。在实际应用中，ADASYN 广泛用于各类不平衡数据集的处理，如金融欺诈检测、医疗诊断等领域，通过生成高质量的合成样本，有效提升了模型的预测性能。通过使用 ADASYN 算法，本研究能够有效平衡少数类样本的数量，确保模型在处理不平衡数据集时的稳定性和准确性，从而更好地评估企业金融化对经营绩效的影响。

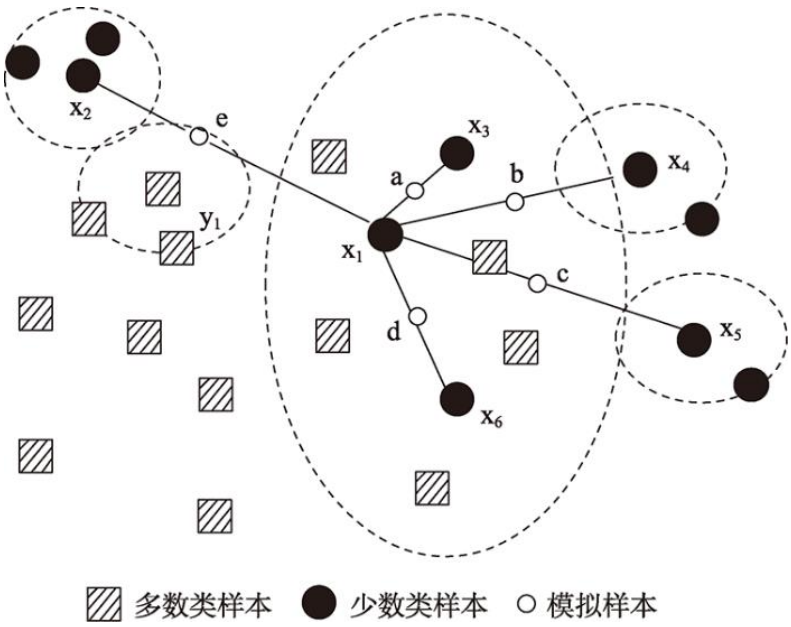


图 2.1 ADASYN 算法示意图

2.3.2 SMOTE

SMOTE (Synthetic Minority Over-sampling Technique) 是一种广泛应用于解决不平衡数据集问题的过采样方法。它通过生成新的合成样本来平衡少数类和多数类样本的比例，从而改善分类器的性能。SMOTE 算法的核心思想是通过插值法在少数类样本及其最近邻之间生成新的样本，从而扩展少数类的决策空间。对于数据集中的每个少数类实例，算法选取其 k 个相似的最近邻，并在这些邻居之间产生新的样本点[17]。具体来说，对于数据集中的每个少数类实例，SMOTE 算法首先选择其 k 个相似的最近邻样本。然后，在原始少数类样本和这些邻居之间生成新的合成样本。生成新样本的公式如下：

$$x_{new} = x_i + \delta \times (x_k - x_i) \tag{2.2}$$

x_i是原始的少数类样本，x_k是其最近邻样本，δ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。

在实际应用中，SMOTE 通过生成高质量的合成样本，能够显著提高模型在不平衡数据集上的表现。本研究利用 SMOTE 算法对数据进行预处理，以生成平衡的数据集，进而构建更加稳健的分类模型。这一过程有助于更准确地评估企业金融化对经营绩效的影响，并确保模型在处理不平衡数据时的稳定性和可靠性。

通过 SMOTE 的应用，本研究不仅提升了模型的预测能力，还为进一步的分析提供了坚实的数据基础。

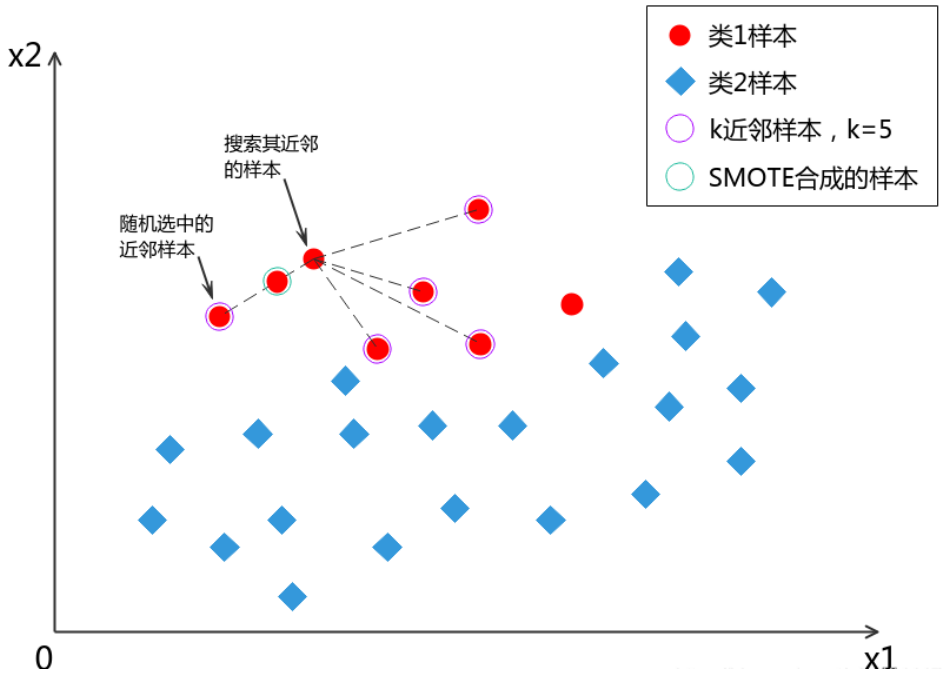


图 2.2 SMOTEN 算法示意图

过采样后的数据在一定条件下不会对模型训练结果的准确性产生负面影响，反而可以提高模型的性能和泛化能力。这是因为过采样技术，尤其是像 SMOTE 、ADASYN 这样的先进方法，旨在解决不平衡数据集带来的问题。过采样生成的合成样本通过扩展少数类的决策空间，帮助模型更全面地理解少数类样本的特征。这种扩展决策边界的方法提高了模型的泛化能力，使其在未见过的数据上表现更佳。在出现数据不平衡问题中，过采样算法只会对模型的鲁棒性有提高的作用，并不使得模型性能上的下降，这些改进共同作用，使得过采样后的数据能够有效提高模型训练的准确性和稳定性。因此，过采样不仅不会对模型训练结果产生负面影响，反而是处理不平衡数据集的一种有效方法。

2.4 机器学习模型与超参数优化

2.4.1 K 最近邻算法

K 最近邻（K Nearest Neighbors, KNN）是一种监督学习算法。当检测一个未知样本类别时，KNN 算法会在训练集中找出与该样本最近的 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]。在处理回归问题时，会把这 K 个邻居的平均或加权平均值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]。合适的 K 值可以通过交叉验证等方法来选择，除了 K 值外，KNN 算法还可以选择其他超参数，如距离度量方式（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，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。

2.4.2 梯度提升树

梯度提升树（Gradient Boosting Decision Tree）是一种集成学习方法，是通过有序地训练多个决策树来提升模型的准确度。梯度提升树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]。

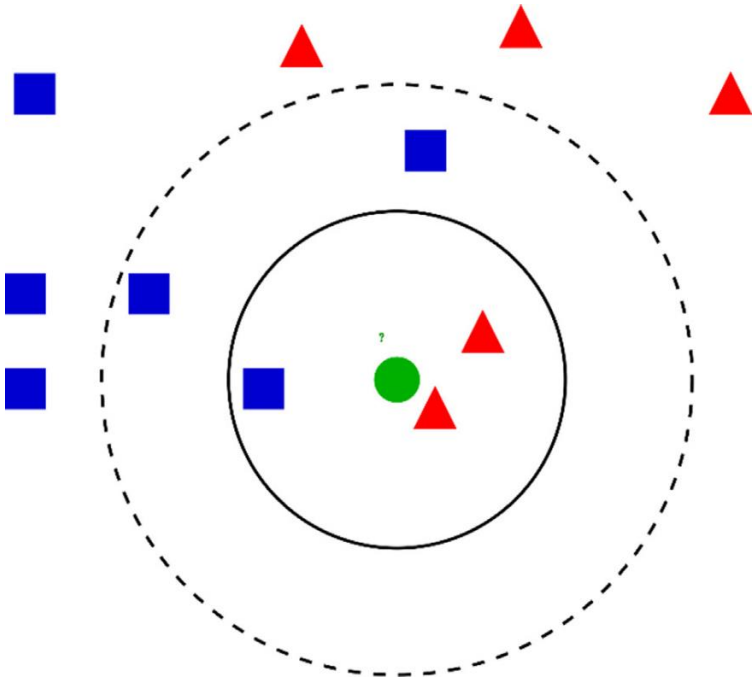


图 2.3 KNN 算法示意图

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。在每一轮迭代中，都会构建一个新的决策树来修正前一轮模型的残差。将每棵决策树的预测结果按权重进行加和，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总和：

$$y_i = \sum_{t=1}^{T} \mu * h_t(x_i) \tag{2.3}$$

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

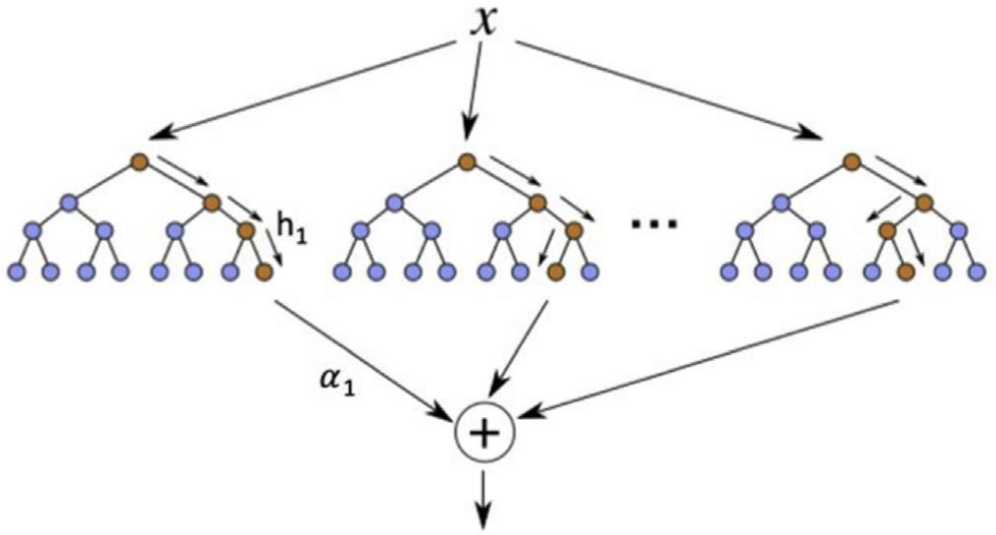


图 2.4 提升决策树:h 为弱分类器，a 表示分配给 h 的程度权重。

2.4.3 支持向量机

支持向量机（Support Vector Machine, SVM）是一种广泛应用的监督学习模型，特别适用于分类问题。SVM 通过寻找一个最优超平面来将不同类别的样本分开，并最大化该超平面到最近样本点的距离，从而实现分类效果的优化[23]。张森等人提出了一种结合偏最小二乘法与 SVM 的水质预测方法，有效解决了多重共线性导致的预测精度低的问题，进一步表明 SVM 模型能够高效处理高维、非线性和小样本问题[24]。

在选择 SVM 模型的超参数时，主要考虑核函数类型和惩罚参数 C。核函数类型决定了数据在高维空间中的映射方式，常用的核函数包括线性核、多项式核和高斯核。惩罚参数 C 则决定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 值可能导致模型欠拟合[25]。SVM 的目标函数是最大化分类间隔，即最小化权重向量w的范数，同时使得所有样本点满足一定的约束条件：

$$y_i = (w^t x_i + b) \geq 1 - \xi_i, \forall_i \tag{2.4}$$

其中，ξ_i 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。

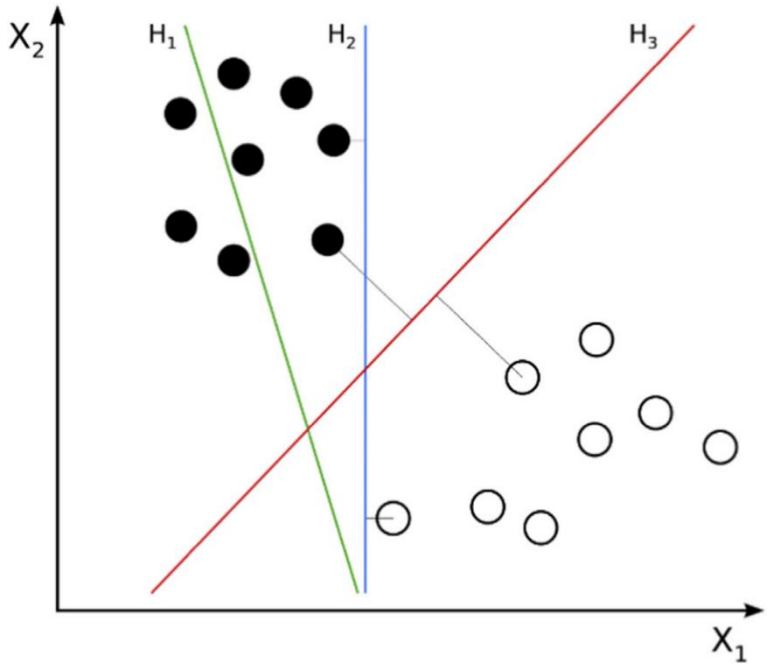


图 2.5 SVM 算法展示。H1 不区分类。H2 有，只有很小的差距。H3 用最大边距分隔它们

在水质检测方面，SVM 模型具有显著优势。首先，SVM 能够处理高维数据，这对于包含多种水质指标的复杂数据集尤为重要。其次，SVM 在处理非线性关系时表现优异，通过核函数将数据映射到高维空间，从而有效捕捉数据的复杂结构。此外，SVM 对于小样本数据具有良好的泛化能力，能够在有限的训练数据下仍然保持较高的预测精度。最后，SVM 的稳健性和鲁棒性使其在面对噪声和异常值时仍能保持稳定的性能，这对于实时水质监测中的数据波动情况尤为关键。

综上所述，SVM 模型凭借其在处理高维、非线性和小样本数据方面的卓越表现，以及其在水质检测中的实际应用优势，成为环境监测领域的一种重要工具，能够有效提高水质预测的准确性和可靠性。

2.4.4 人工神经网络

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。输入样本通过网络层进行前向传播，经过多次线性和非线性变换，最终在输出层生成预测结果。通过反向传播

算法，将预测结果与真实标签的差异进行比较，并使用梯度下降算法来调整网络中的权重和偏置，以最小化损失函数，使模型的预测结果逐渐逼近真实标签[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直接影响了网络的复杂度和学习能力：更多的隐藏层和节点数可以捕捉更复杂的数据模式，但也增加了计算复杂度和过拟合的风险。学习率控制了每次权重更新的步长：较高的学习率可以加速训练过程，但可能导致不稳定的收敛；较低的学习率则收敛较慢但更稳定。正则化参数用于控制模型的复杂度，通过增加惩罚项来防止过拟合，从而提高模型的泛化能力。

此外，ANN 在处理非线性关系、复杂模式识别和大规模数据方面表现出色，广泛应用于图像识别、语音处理和自然语言处理等领域。在水质检测方面，ANN 模型通过大量历史数据的训练，可以有效识别和预测水质变化趋势，从而为环境监测和决策提供可靠的支持。通过优化网络结构和调整超参数，ANN 能够在多维数据中提取有效特征，提高模型的预测精度和稳定性。

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]。说明其可以通过反向传播算法来学习连接权重，实现了对输入数据的有效建模和预测。

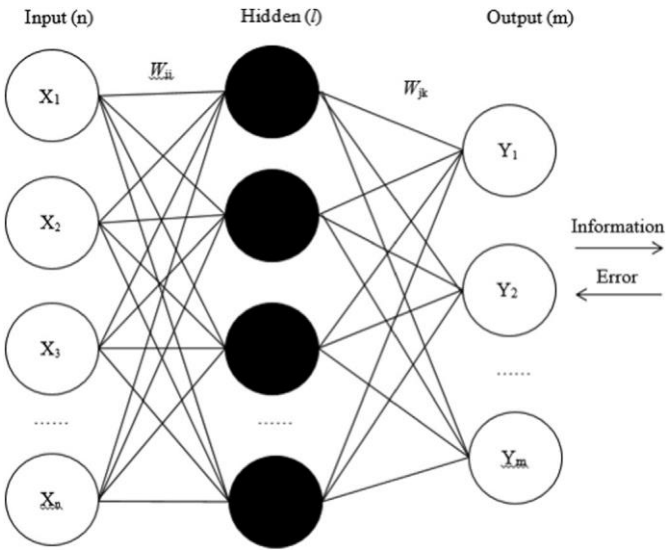


图 2.6 BP-TLP 神经网络的体系结构

2.5 系统工作流程

采样后的数据构成原始数据库，其中包含 7 个训练字段和 1 个标签字段（总大肠菌）。系统工作流程如图 3.5 所示，首先，利用总大肠菌值将样本分为两类，即总大肠菌高于或低于 200。接着，采用 ADASYN 算法生成高于阈值的合成样本，以平衡数据集。随后，从低于阈值 200 的原始样本中随机选择数据，并与合成样本结合，形成新的训练数据集。该训练数据集用作四种机器学习算法的输入。剩余的未选择样本和原始超过阈值的样本被耦合，形成验证和测试数据集，并用于模型结果的验证过程。

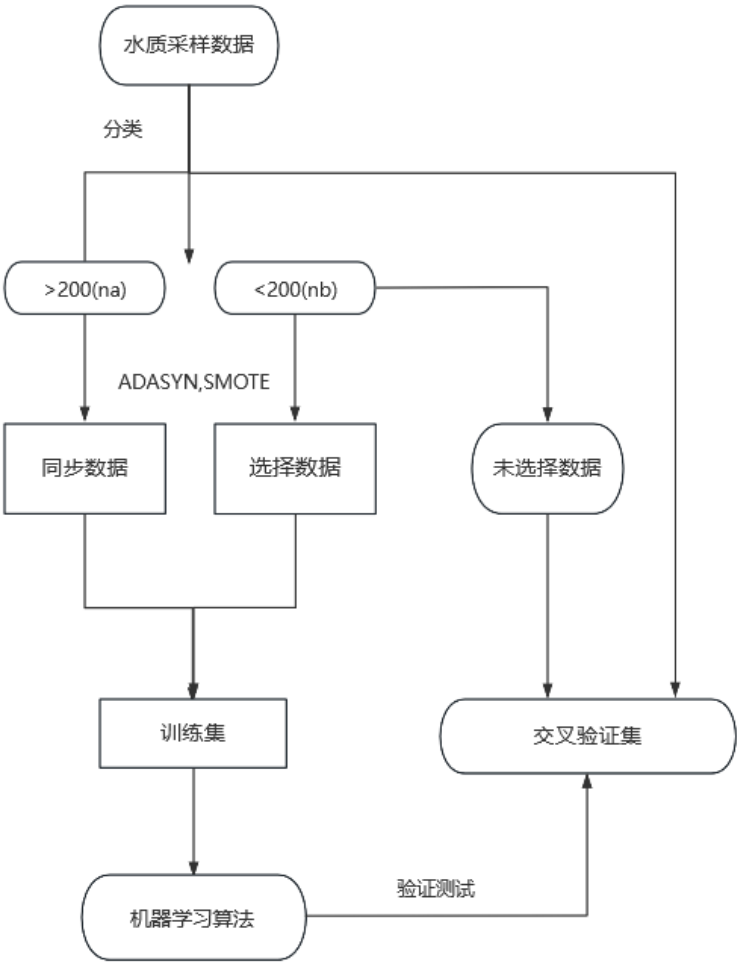


图 2.7 系统工作流程图

第 3 章 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。

3.1 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模块和机器学习模型，并采取安全措施以保护数据，包括用户认证和访问控制。

如图 3.1 显示了系统用例图，该用例图展示了用户在数据处理和模型训练中的主要操作流程和各个用例之间的关系。用户可以通过“查看本次数据指标”来了解当前水质指标，这一用例通过“显示数据集”用例来实现数据的展示。用户可以选择“上传 csv 文件”来导入新的数据集，这个操作同样会触发“显示数据集”用例以展示上传的数据。

在数据处理环节，用户可以选择“过采样数据集”，包括使用 ADASYN 技术和 SMOTE 技术，这些技术可以扩展出“返回过采样前后的数据分布”用例，以便用户对比数据分布的变化。这一环节也会通过“显示数据集”用例来展示。

在模型训练方面，用户可以选择不同的算法进行训练，包括 KNN 训练、GBDT 训练、ANN 训练和 SVM 训练。每一个训练用例都会包含“训练结果评估”

用例，该用例进一步扩展出“返回准确率、召回率和 F1 得分为指标的评估结果”用例，以提供详细的模型评估指标。

预测数据集的功能允许用户选择不同的预测算法，包括 KNN 预测、ANN 预测、SVM 预测和 GBDT 预测。每一个预测用例都会包含“选择模型优值”和“选择模型学习率”用例，以便用户调整预测模型的参数。最终的预测结果也会通过“显示数据集”用例展示给用户。

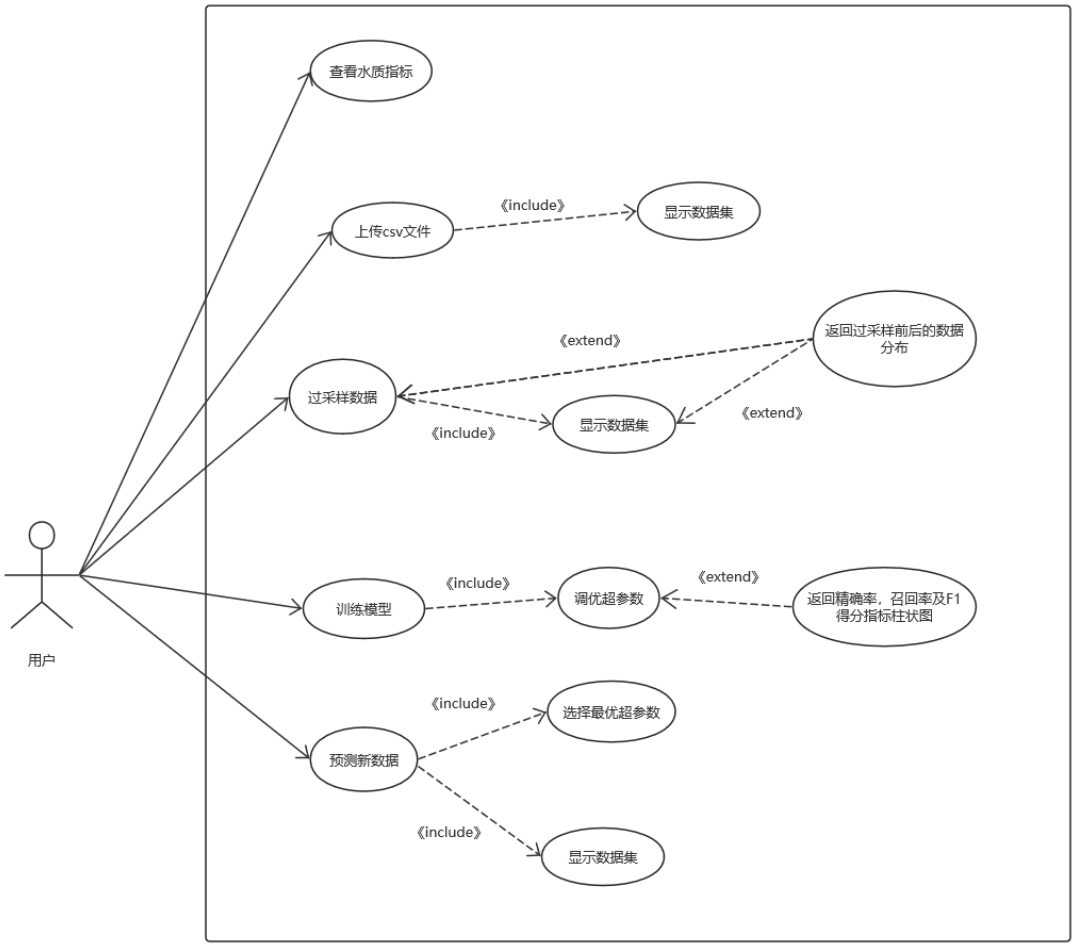


图 3.1 用例图

在需求分析阶段，本研究通过分析项目的一个组织和结构设计得到系统的业务流程图。该业务流程图展示了水质检测系统的详细操作流程。首先，用户通过登录进入系统，然后上传训练水质数据。系统进行数据清洗和特征提取，以确保数据的质量和有效性。根据数据的不平衡情况，系统采用 ADASYN 或 SMOTE 算法进行数据过采样，生成平衡后的数据集。接下来，系统选择合适的机器学习模

型进行训练，并通过交叉验证等技术选择最佳超参数，确保模型的准确性和可靠性。完成模型训练后，用户可以上传新的预测数据集，系统将根据训练好的模型进行水质预测，并显示预测结果。此外，系统还可以显示水质指标和采样点信息，提供全面的水质监测服务。整个流程确保了数据处理、模型训练与预测的科学性和有效性，满足水质检测系统的需求分析要求。

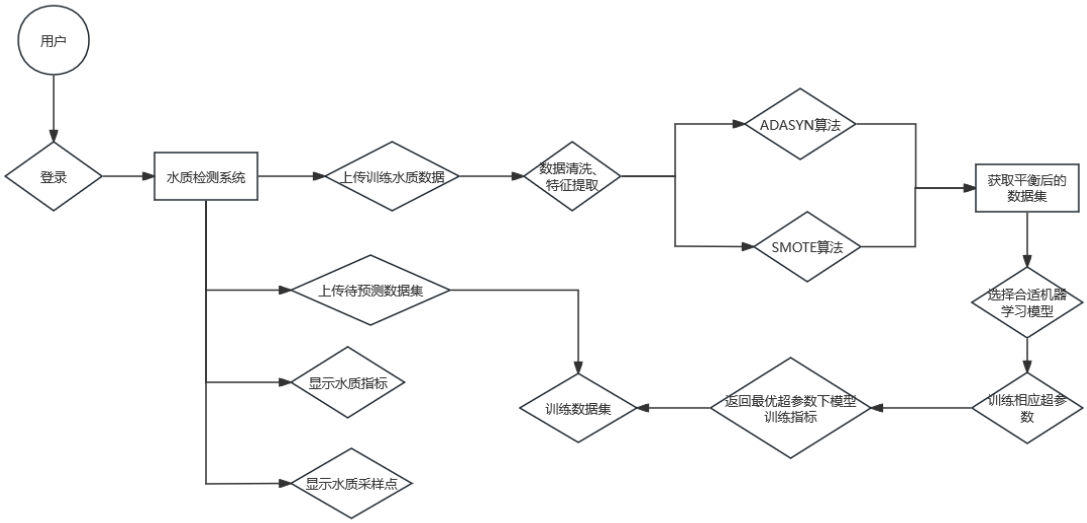


图 3.2 业务流程图

在水质监测系统中，数据预处理、数据上传及模型训练和预测模块是关键功能模块，确保系统能够有效处理、存储和分析水质数据，实现准确的水质预测和监测。数据预处理模块包括数据清洗、特征选择和数据平衡处理，生成高质量的训练数据集；数据上传模块则负责用户原始水质数据文件的上传和存储，保证数据的完整性和可用性；模型训练及预测模块利用预处理后的数据进行机器学习模型（如 KNN、GDBT、SVM 和 ANN）的训练，并通过最优参数选择进行模型预测，下面是相关模块的用例规约：

表 3.1 详细描述了数据上传模块的功能和操作流程，确保用户能够顺利地将原始水质数据文件上传到系统中，为后续的数据预处理和模型训练提供必要的数据支持，从而提高整个水质监测系统的准确性和可靠性。

表 3.1 数据上传用例规约

用例名称	数据上传。
用例描述	该用例描述了系统如何接受用户上传的原始水质数据文件，并将其存储到系统中以备后续的预处理和模型训练使用。
参与者	用户
前置条件	• 用户已成功登录系统。 • 用户已准备好需要上传的原始水质数据文件（格式为 CSV、Excel 等）。
后置条件	• 原始水质数据文件成功上传并存储在系统中。 • 系统生成上传成功的确认信息，并显示数据预览。
主成功场景	1. 用户点击“上传数据”按钮。 2. 系统显示文件选择窗口，用户选择需要上传的水质数据文件。 3. 用户确认文件选择，点击“上传”按钮。 4. 系统接收文件并验证文件格式和内容。 5. 系统将文件内容存储在数据库中。 6. 系统生成上传成功的确认信息，并展示数据预览，包括数据的基本统计信息和部分样本数据。 7. 用户确认上传结果，完成数据上传操作。
扩展场景	4a. 如果文件格式不符合要求（如不是 CSV 或 Excel 格式），系统提示用户上传失败，并要求重新选择文件。 4b. 如果文件内容有误（如缺少必要字段或包含不合法字符），系统提示用户上传失败，并建议检查文件内容。
非功能性需求	• 系统界面应简洁友好，用户操作简单明了。 •系统应具有良好的扩展性，便于后续支持更多文件格式和上传方式。

表 3.2 详细描述了数据预处理模块的功能和操作流程，确保用户能够在清晰的指导下完成数据的清洗、特征选择和过采样处理，从而为后续的模型训练和预测提供高质量的数据支持，提高整个水质监测系统的准确性和可靠性。

表 3.2 数据预处理用例规约

用例名称	数据预处理
用例描述	该用例描述了系统如何对用户上传的原始水质数据进行清洗、特征选择和过采样处理，为后续模型训练和预测提供高质量的数据。
参与者	用户
前置条件	• 用户已成功登录系统。 • 用户已上传原始水质数据。
后置条件	• 数据经过清洗、特征选择和过采样处理，生成高质量的平衡数据集，准备用于模型训练。
主成功场景	8. 用户点击“数据预处理”按钮。 9. 系统显示数据预处理选项，用户选择需要的预处理步骤，包括数据清洗、特征选择和过采样处理。 10. 系统执行数据清洗操作，处理特殊字符，将数据转换为浮点型，并删除包含 NaN 值的行。 11. 系统执行特征选择操作，保留温度、溶氧量、PH 值、电导率等重要特征。 12. 系统执行过采样处理，用户选择 ADASY 或者 SMOTE 算法平衡数据集。 13. 系统完成数据预处理，并展示预处理后的数据分布情况。 14. 用户确认预处理结果，系统保存预处理后的数据集。
扩展场景	3a. 如果数据清洗过程中发现异常数据，系统提示用户数据异常并建议用户检查上传的原始数据。 4a. 如果特征选择不合理导致关键特征丢失，系统提示用户重新选择特征。 5a. 如果过采样处理失败，系统提示用户选择其他算法并重新尝试。
非功能性需求	• 系统界面应简洁友好，用户操作简单明了。 • 系统应具有良好的扩展性，便于后续增加新的数据预处理方法和优化现有方法。

表 3.3 详细描述了模型训练及预测模块的功能和操作流程，确保用户能够在清晰的指导下完成模型的训练和预测工作，从而提高水质监测系统的智能化和自动化水平。

表 3.3 模型训练及预测用例规约

用例名称	模型训练及预测
用例描述	该用例描述了系统如何利用预处理后的数据进行模型训练，并使用训练好的模型进行预测。
参与者	用户
前置条件	• 用户已成功登录系统。 • 数据已上传并经过预处理，包括数据清洗、特征选择和过采样处理。
后置条件	• 系统返回模型的训练结果，包括模型的准确率、召回率、F1 得分等性能指标。 • 用户可以使用训练好的模型进行新数据的预测，并查看预测结果。
主成功场景	15. 用户点击“模型训练”按钮。 16. 系统弹出选择模型界面，用户选择需要训练的模型（如 KNN、GDBT、SVM、ANN）。 17. 系统设定默认超参数（如 KNN 的 K 值，GDBT 的学习率，SVM 的 C 值，ANN 的学习率和层数）。 18. 系统开始训练所选模型，并进行超参数调优。 19. 系统完成训练后，显示训练结果，包括模型的准确率、召回率、F1 得分等性能指标。 20. 用户点击“模型预测”按钮，上传新数据。 21. 系统使用训练好的模型对新数据进行预测，并返回预测结果。 22. 用户查看预测结果。
扩展场景	5a. 如果训练过程中发生错误，系统显示错误信息，系统可以重新设定超参数并重试。 7a. 如果新数据上传失败，系统提示用户重新上传数据。
非功能性需求	• 系统界面应简洁友好，用户操作简单明了。 • 系统应具有良好的扩展性，便于后续增加新模型和优化现有模型。

3.2 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。

第 4 章 系统总体设计及详细设计

4.1 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：

数据上传模块负责上传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上传模块进入系统时，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上传模块，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水质好坏的预测结果，进行数据可视化。

如图 4.1 展示了系统功能模块图，展示了水质监测系统的整体架构及其各个功能模块。系统分为多个关键模块，包括数据上传、数据预处理、模型训练和模型预测。用户首先上传水质数据，系统将数据进行清洗、特征选择和自适应过采样（如 ADASYN 或 SMOTE），以平衡数据分布并提高模型训练的效果。经过预处理的数据用于训练多种机器学习模型，包括 KNN、GDBT、SVM 和 ANN 模型。每个模型通过调参优化以获得最佳性能，并根据模型指标评估其准确性和可靠性。训练完成后，用户可以上传新数据进行预测，系统将返回预测结果。此外，系统还包括展示水质采样点和水质指标、显示数据分布等功能，以提供全面的水质监测信息。这些模块的设计确保了系统的功能完整性和高效性。

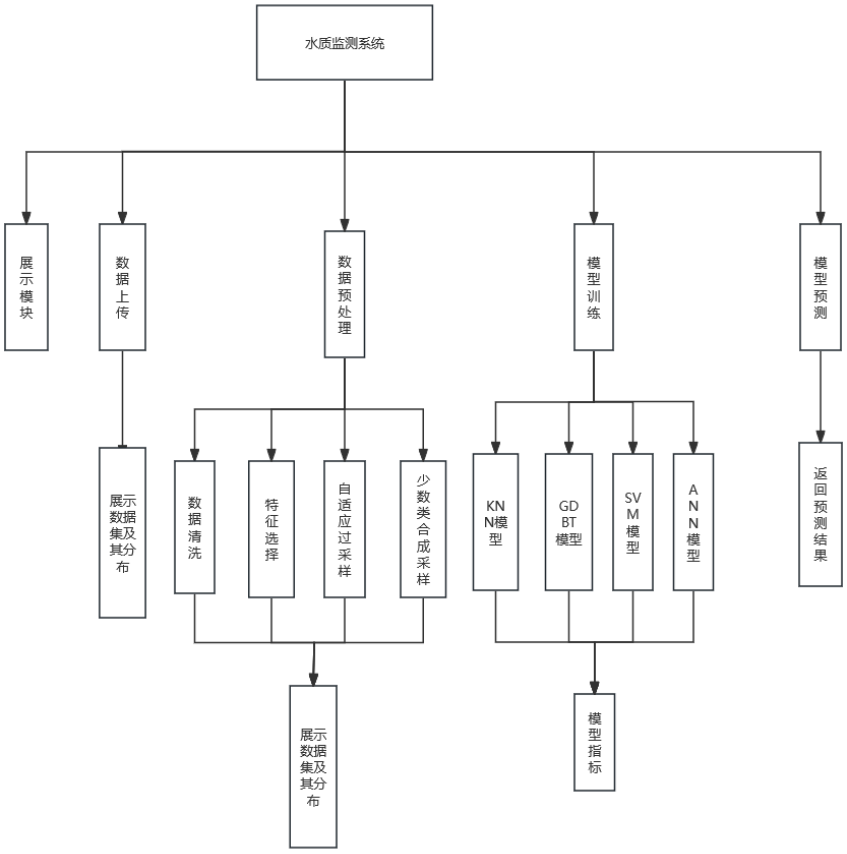


图 4.1 系统功能模块图

如图 4.2，本文详细介绍了水质检测系统的架构设计及其各层次的功能模块。系统架构分为五个层次：访问层、前端 UI 层、展示层、业务层和数据层。在访问层，用户通过电脑终端进行操作，进入系统。前端 UI 层采用 Vue3、HTML5、Element.UI、CSS 和 JavaScript 等技术栈构建用户界面，提供友好的交互体验和高效的前端功能。展示层通过 RESTful API 实现前后端的数据交互，支持 POST 和 GET 请求，确保数据传输的安全性和高效性。业务层包含数据上传模块、数据预处理模块、模型训练与预测模块以及后端服务模块。用户通过数据上传模块将水质数据上传至系统，数据预处理模块对数据进行清洗和特征提取，模型训练与预测模块则负责模型的选择、训练和预测。后端服务模块处理来自前端的请求，并协调各业务模块的工作，确保业务逻辑的顺畅执行。数据层负责数据的存储和业务处理，确保系统的数据安全和高效访问。通过各层次模块的合理分工和相互协作，整个系统架构设计确保了水质检测任务的高效执行和数据处理的准确性，满足了水质检测系统的业务需求。

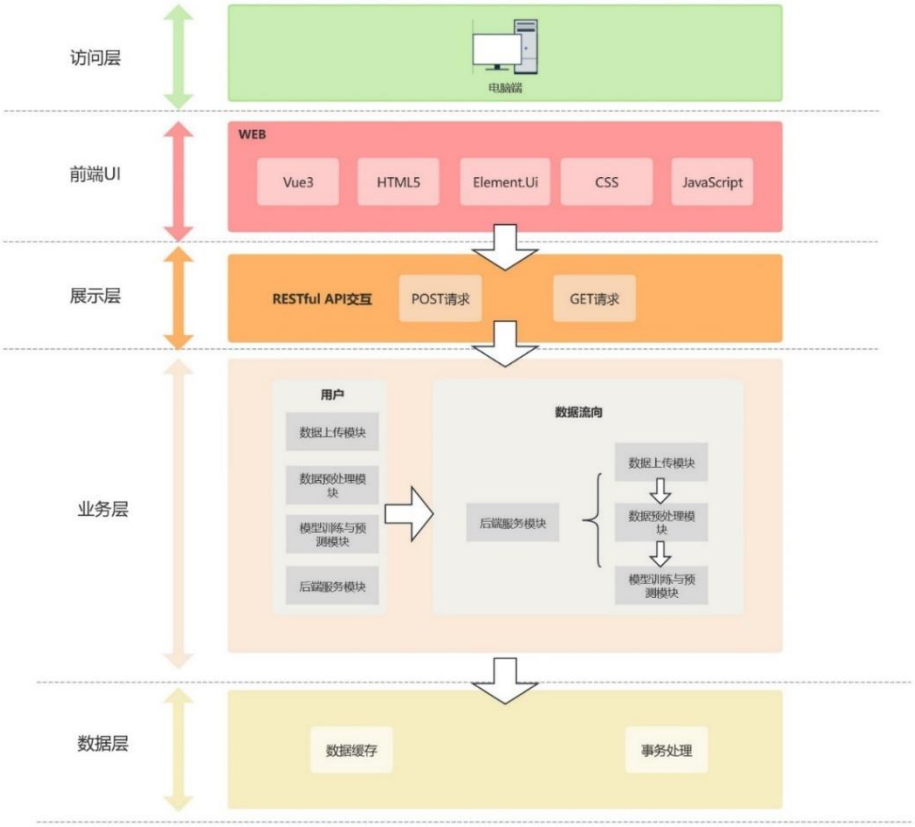


图 4.2 系统架构图

4.2 详细系统设计

4.2.1 展示模块

该时序图展示了用户在系统页面上查看水质检测所需特征及水质采样点图片的过程。首先，用户上传地标图片到系统页面，系统页面调用图片显示控件来展示用户上传的图片。接着，用户访问主页面，系统页面通过调用服务端参数，获取并显示水质指标等相关数据。这些数据包括水质采样地点的显示信息和页面上其他相关的水质指标数据。通过该流程，系统确保用户可以直观地查看水质采样点的位置图片和相关的水质检测特征，提升了系统的可用性和用户体验。

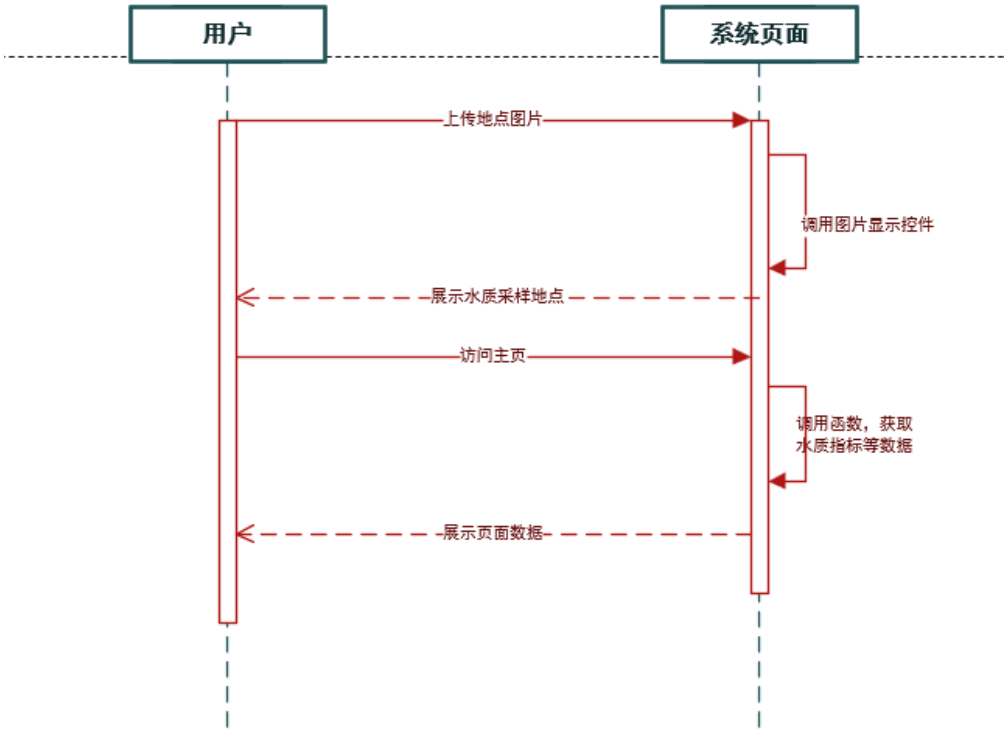


图 4.3 展示模块时序图

4.2.2 数据上传模块

下面详细介绍了水质检测系统中数据上传模块，展示了用户、系统页面和系统业务三者之间的交互流程。如图 4.4 所示，用户通过电脑终端访问系统页面，

系统页面调用后端服务获取水质指标等初始数据，并将这些数据展示给用户。用户在页面上点击数据上传按钮，系统页面捕获该操作并将数据传递给系统业务处理模块。系统业务模块接收到数据后，进行数据的初步清洗和预处理，确保数据的完整性和有效性。处理完成后，系统业务模块将清洗后的数据返回给系统页面，系统页面再将处理结果展示给用户。整个过程通过清晰的时序图展示了用户操作、系统页面响应和后台业务处理之间的紧密协作，确保了数据上传过程的流畅和高效，满足了系统的业务需求。

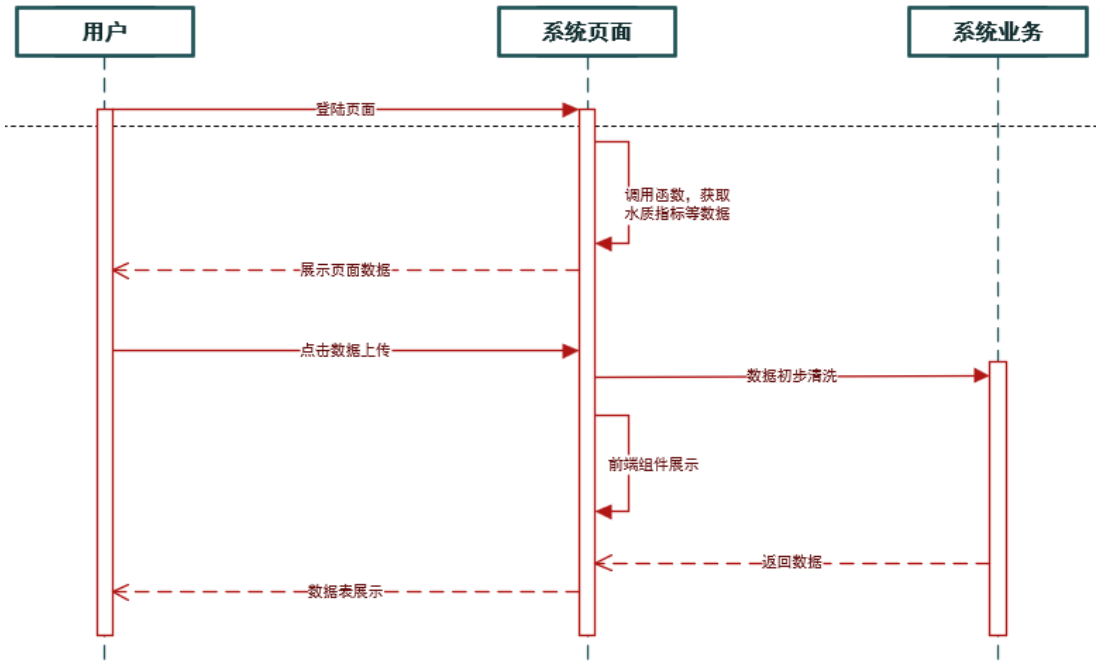


图 4.4 上传数据时序图

4.2.2 数据预处理模块

在本研究中，数据预处理模块分为数据清洗、特征选择和过采样处理三个主要部分。数据清洗是数据预处理的首要步骤，旨在确保数据质量。具体操作包括将特殊字符替换为 NaN 值、去除特殊字符、将数据转换为浮点型以及删除包含 NaN 值的行，以确保数据一致性和完整性。特征选择是通过挑选对预测结果有显著影响的变量来优化模型性能。本研究选择了温度、溶氧量、PH 值、电导率、生化需氧量、硝酸盐和亚硝酸盐的平均值、粪大肠菌和总大肠菌等特征，这些变量均是反映水质的重要指标。为了处理数据集中的类别不平衡问题，本研究应用了

ADASYN 和 SMOTE 两种过采样技术。ADASYN 是一种自适应的合成抽样方法，通过合成新的少数类样本来平衡数据集，使得合成样本更加接近真实分布；而 SMOTE 则通过在少数类样本的最近邻之间生成新的合成样本来增加少数类样本数量，改善模型的泛化能力。通过上述数据预处理步骤，确保了数据集的高质量和均衡性，为后续模型训练和分析提供了可靠的基础。

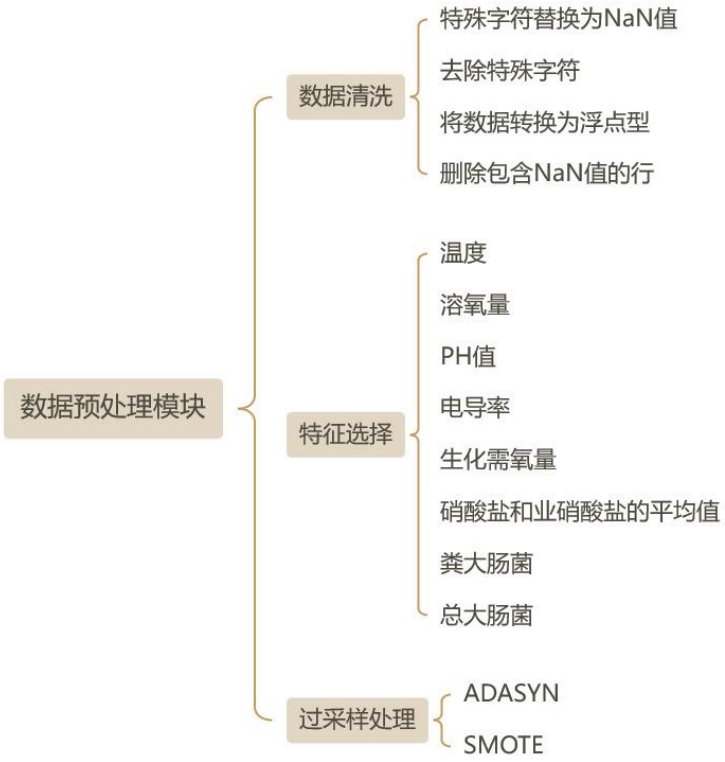


图 4.5 数据预处理模块

下面详细说明了水质检测系统中数据预处理模块的时序图及其工作流程。首先，用户在系统页面上点击数据上传按钮，将水质数据上传至系统。系统页面接收到用户上传的数据后，立即调用系统业务层的函数，获取水质指标等相关数据。系统业务层接收到请求后，开始进行数据预处理，包括数据清洗、特征提取和数据平衡等步骤。预处理完成后，系统业务层将处理后的数据返回给系统页面。系统页面接收到处理后的数据后，将数据展示给用户，包括数据的预处理结果和采样点的分布情况。整个流程确保了用户上传的数据能够快速、准确地进行预处理，并将结果及时反馈给用户，为后续的模型训练和预测提供了高质量的数据基础。

通过这种设计，系统实现了数据预处理的高效性和准确性，确保了水质检测任务的顺利进行。

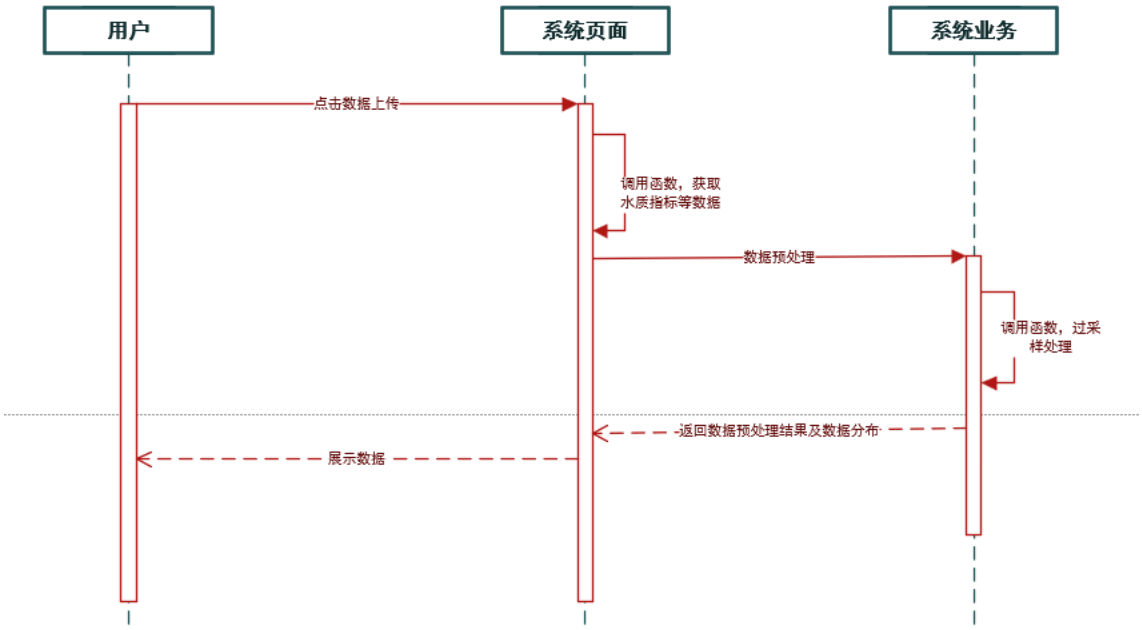


图 4.6 数据预处理时序图

4.2.3 模型训练与预测模块

在模型训练与预测模块中，利用预处理后的数据对四种不同的机器学习算法进行训练和优化，这四种算法分别是 K 最近邻（KNN）、梯度增强决策树（GDBT）、支持向量机（SVM）和人工神经网络（ANN）。每种算法都有其特定的超参数需要调优，以确保模型能够达到最佳性能。

首先，对于 KNN 算法，主要的超参数是 K 值，即选择的邻居数量。通过交叉验证技术，选择使得模型在验证集上表现最优的 K 值。对于 GDBT 和 ANN，主要调优的超参数是学习率，学习率决定了每次迭代更新权重的步长大小，选择合适的学习率能够加速模型的收敛并提高预测精度。对于 SVM，主要调优的超参数是 C 值，C 值是正则化参数，用于平衡分类边界的平滑性和训练误差的权重，同样通过交叉验证确定最优的 C 值。

在模型训练过程中，通过精确率、召回率和 F1 得分等性能指标对模型进行评估。精确率衡量的是模型预测的正类样本中实际为正类的比例；召回率衡量的是

实际为正类的样本中被模型正确预测为正类的比例；F1 得分是精确率和召回率的调和平均数，用于综合评估模型的性能。通过这些指标，可以全面衡量每个模型的优劣，确保选择的模型在实际应用中具有较高的准确性和可靠性。

最终，基于这些评估指标，选择最优的模型用于预测。该模块的核心在于，通过科学的模型训练和评估方法，确保所选择的模型在新数据上的预测性能最佳，从而实现高效且准确的水质监测和预测任务。通过这样的系统化训练和评估流程，可以大大提高模型的泛化能力和实际应用效果，为水质监测提供可靠的技术支持。

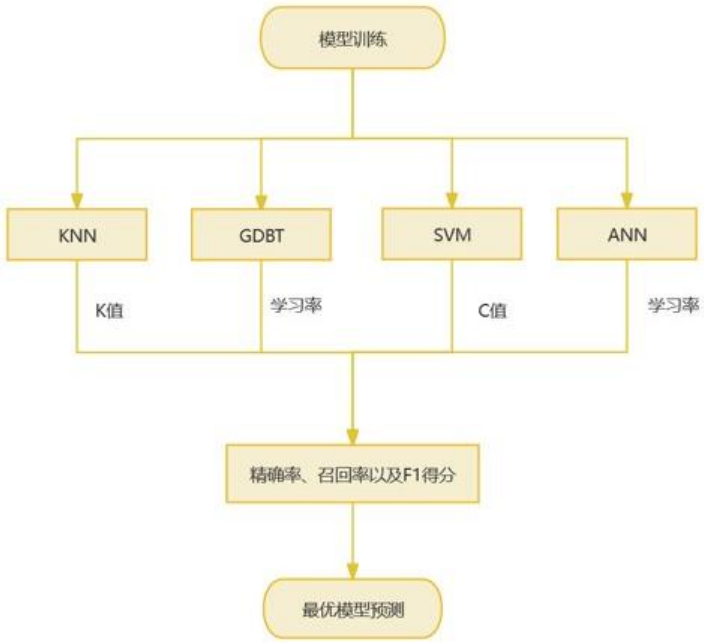


图 4.7 模型训练及预测模块

下面展示及说明了水质检测系统中模型训练及预测模块的时序图及其工作流程。首先，用户在系统页面上上传数据，系统页面接收到数据后，调用前端组件进行数据展示。用户点击相应按钮，选择特定模型进行训练，系统页面调用业务层函数，传递训练请求。系统业务层接收到请求后，调用相应的机器学习算法进行模型训练，并返回模型的精确率、召回率及 F1 得分等指标。系统页面接收到这些指标后，更新界面显示训练结果，用户可以查看这些指标以评估模型表现。

接下来，用户再次上传新的数据用于预测，系统页面重复调用前端组件展示数据，并调用业务层进行预测。系统业务层接收到预测请求后，调用已优化的模型进行预测，并将预测结果返回给系统页面。系统页面接收到预测结果后，将结

果展示给用户。整个流程确保了用户能够方便地进行模型训练和预测，系统通过高效的交互和数据处理，提供了准确的模型性能评估和预测结果，为水质检测提供了可靠的技术支持。通过这种设计，系统实现了模型训练和预测的高效性和准确性，确保了水质检测任务的顺利进行。

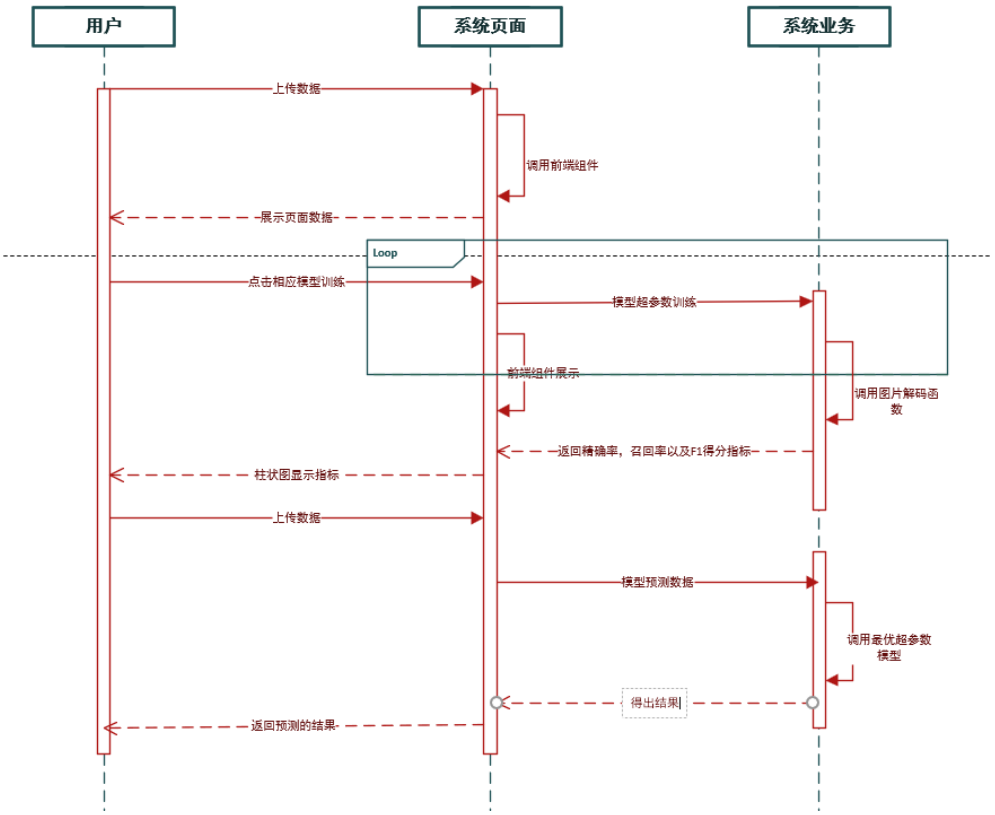


图 4.8 模型训练及预测时序图

第 5 章 系统测试

5.1 系统页面展示及数据处理过采样测试

在本项目中，我们采用了黑盒测试方法对系统进行功能验证和质量保证。黑盒测试通过对系统的各个功能模块进行逐项验证，确保系统在不同输入条件下能够产生正确的输出结果，而不关注其内部实现细节。我们设计了多个详细的测试用例，包括首页轮播图显示、水质指标文字显示、文件上传模块、过采样算法处理模块、模型训练测试以及模型分析测试等。每个用例都明确了测试的用例说明、前置条件、输入、执行步骤、预期结果以及重要程度，并逐一执行测试，下面介绍一下核心功能测试。

在第一个测试用例中，测试的功能点是首页的文件上传模块。该用例说明文件上传模块是否能够正常处理 CSV 文件并展示上传后的数据列表。前置条件是首页能够正常显示。输入为一个 CSV 文件。执行步骤为点击上传按钮上传 CSV 文件。预期结果包括页面弹出上传成功的提示，并正确显示上传后的数据列表。由于文件上传是系统核心功能之一，重要程度被评为高。测试结果显示该用例通过，证明文件上传模块功能正常。

第二个测试用例测试了首页的过采样算法处理模块。用例说明测试了该模块能否正确处理上传的文件并展示过采样前后的数据分布。前置条件是文件已经成功上传。执行步骤为点击过采样算法按钮，处理已上传的 CSV 文件。预期结果包括页面展示 ADASYN 和 SMOTE 两个过采样算法按钮，正确显示过采样前后的数据分布柱状图，并返回过采样方法名称及新生成数据集总量。此模块的重要程度也评为高，因为它直接影响数据处理的准确性和后续模型训练。测试结果显示该用例通过，表明过采样算法处理模块功能正常。

表 5-1 测试用例表

功能点	用例说明	前置条件	输入	执行步骤	预期结果	重要程度	测试结果
首	轮播图	首页	无	点击轮播图下方	界面图片显示正确，按钮	低	通

页	能够正常显示图片切换	能够正常显示		按钮进行图片切换	齐全，控件整齐		过
首页	水印质量文字指标显示正确	首页能够正常显示	无	无	正确显示了页面文字，控件齐全，布局整齐	低	通过
首页	文件上传模块测试	首页能够正常显示	CSV 文件	点击上传按钮，上传 CSV 文件	页面弹出上传成功的弹出，并显示上传后的数据列表	高	通过
首页	过采样算法处理模块测试	文件以及成功上传	无	点击过采样算法按钮，执行算法，处理已上传的 CSV 模块	1. 页面会正确展示 ADASYN、SMOTE 两个过采样算法按钮 2.页面正确显示过采样前后的数据分布的柱状图，并弹出处理成功的弹窗 3.页面返回过采样方法名称及新生成数据集总量 4.显示过采样算法处理后的新数据	高	通过

5.2 系统模型预测及分析测试

在第二个测试用例表中，详细记录了机器学习模型关键功能点的测试过程及结果，分别为 KNN 模型、GDBT 模型、SVM 模型以及 ANN 模型训练测试。针对模型训练测试，首先确保数据已通过过采样处理，接着点击机器学习模型按钮，执行四种机器学习模型的训练及超参数调优。预期结果包括页面正确显示模型按钮，成功调用后弹出提示框，并展示参数调优过程中的参数与准确率相关性的折线图，以及在最优超参数下的各模型的准确率、召回率及 F1 得分的柱状图。测试结果显示页面能够准确显示模型训练后的性能指标、最优超参数值及模型名称，测试通过。测试结果表明，系统能够准确显示个模型模型训练后的性能指标、最优超参数值及模型名称，测试通过。通过这四个测试用例，验证了系统在处理过采样数据并进行机器学习模型训练和调优方面的功能正常，为后续数据分析和决策提供了可靠支持。随后详细记录了模型预测测试用例详情，其前置条件为模型已正确被训练，并返回了最有超参数、准确率等模型评估指标，输入 CSV 文件作为模型预测数据集，通过在模型训练后，点击模型分析，会对新数据集进行预测分析，测试结果表明页面正确显示模型分析按钮，同时根据上个模型训练选择的

机器学习模型，执行相应分析预测，在执行成功后会弹出成功弹窗，并在前端展示数据，测试通过

表 5-2 测试用例表

功能点	用例说明	前置条件	输入	执行步骤	预期结果	重要程度	测试结果
首页	KNN模型训练测试	数据已经过采样处理	无	点击机器学习模型按钮，执行模型的训练及超参数调优，训练过采样处理后的数据集	1.点击按钮，会有成功调用的弹窗，页面正确显示参数调优过程中的参数与准确率相关性的折线图 2.页面显示最优超参数下的模型的准确率，召回率及 F1 得分指标的柱状图。	高	通过
首页	GDBT模型训练测试	数据已经过采样处理	无	点击机器学习模型按钮，执行模型的训练及超参数调优，训练过采样处理后的数据集	1.点击按钮，会有成功调用的弹窗，页面正确显示参数调优过程中的参数与准确率相关性的折线图 2.页面显示最优超参数下的模型的准确率，召回率及 F1 得分指标的柱状图。	高	通过
首页	SVM模型训练测试	数据已经过采样处理	无	点击机器学习模型按钮，执行模型的训练及超参数调优，训练过采样处理后的数据集	1.点击按钮，会有成功调用的弹窗，页面正确显示参数调优过程中的参数与准确率相关性的折线图 2.页面显示最优超参数下的模型的准确率，召回率及 F1 得分指标的柱状图。	高	通过
首页	ANN模型训练测试	数据已经过采样处理	无	点击机器学习模型按钮，执行模型的训练及超参数调优，训练过采样处理后的数据集	1.点击按钮，会有成功调用的弹窗，页面正确显示参数调优过程中的参数与准确率相关性的折线图 2.页面显示最优超参数下的模型的准确率，召回率及 F1 得分指标的柱状图。	高	通过
首页	模型预测测试	数据已被模型训练	上传文件	点击模型分析按钮，执行模型预测	页面显示最优超参数下的模型的准确率，召回率及 F1 得分指标的柱状图。	高	通过

通过这些详细的黑盒测试用例，我们全面验证了系统各个功能模块的工作情况，确保系统在不同输入条件下都能产生正确的输出，进而提高了系统的稳定性和可靠性。这些测试用例的成功通过，为系统的正式上线和使用提供了强有力的质量保障。

第 6 章 前后端展示及研究结果

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，前后端分离的开发模式可以提高开发效率和代码可维护性。Vue.js 和 Django 拥有活跃的社区和丰富的文档资源，为开发人员提供了大量支持和帮助。

6.1 前后端展示

6.1.1 前端页面展示

在图 6.1 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，能够确定上传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。

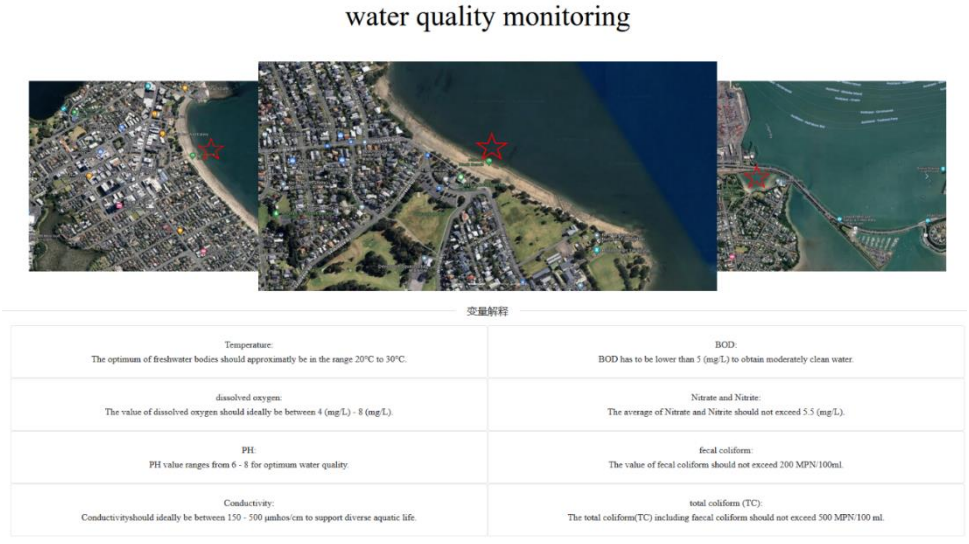


图 6.1 展示采样地点和变量解释

如图 6.2 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 6.2 导入 csv 文件以及数据展示

如图 6.3 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。

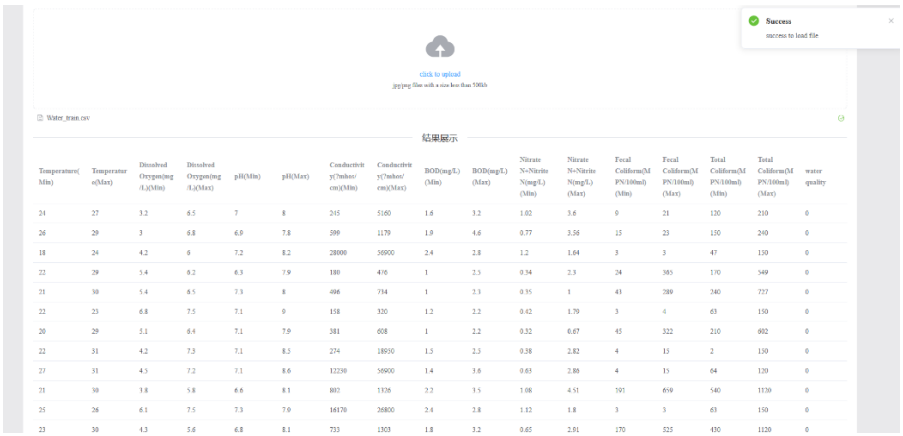


图 6.3 过采样前后数据图表展示

如图 6.4 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 6.4 KNN 算法调参过程及结果展示

图 6.5 GDBT 算法调参过程及结果展示

如图 6.5 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。

如图 6.6 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。下方的注释显示了最大准确率，最优 C 值以及模型类型的相关信息。

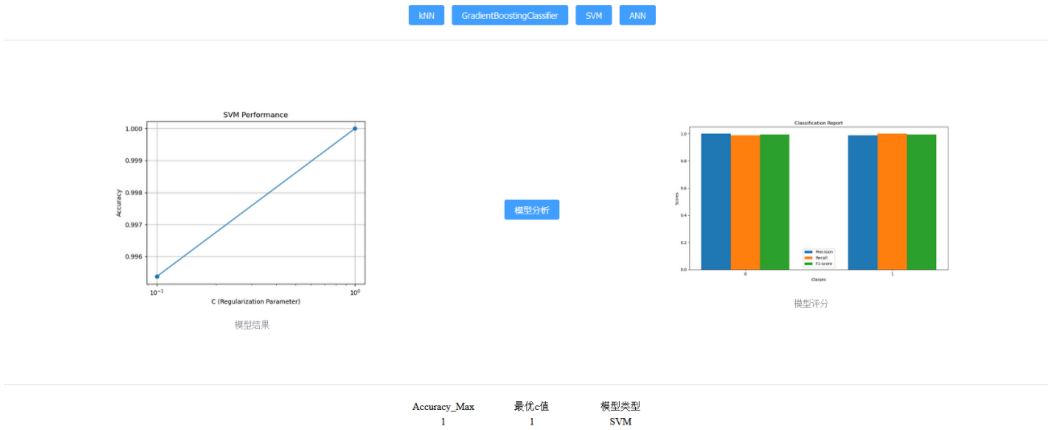


图 6.6 SVM 算法调参过程及结果展示

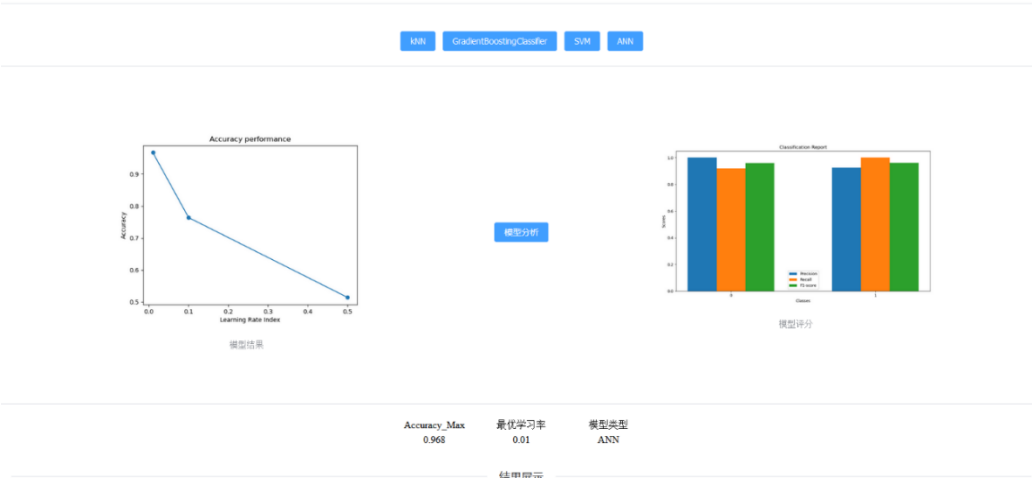
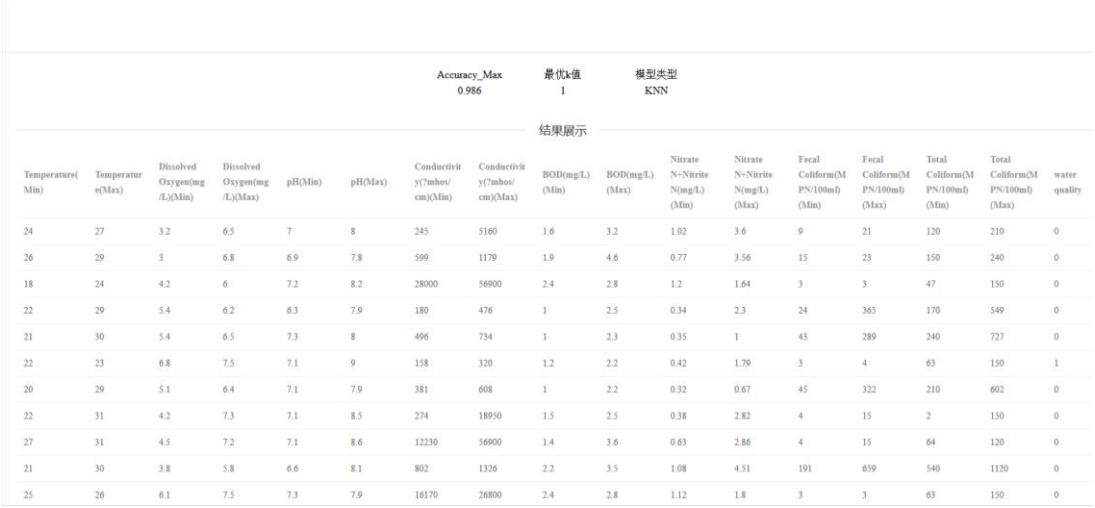


图 6.7 ANN 算法调参工程及结果展示

如图 6.7 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。

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 6.8 所示。



Accuracy_Max 0.986　最优k值 1　模型类型 KNN

结果展示

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)	water quality
24	27	3.2	6.5	7	8	245	5160	1.6	3.2	1.02	3.6	9	21	120	210	0
26	29	3	6.8	6.9	7.8	599	1179	1.9	4.6	0.77	3.56	15	23	150	240	0
18	24	4.2	6	7.2	8.2	28000	56900	2.4	2.8	1.2	1.64	3	3	47	150	0
22	29	5.4	6.2	6.3	7.9	180	476	1	2.5	0.34	2.3	24	365	170	549	0
21	30	5.4	6.5	7.3	8	496	734	1	2.3	0.35	1	43	289	240	727	0
22	23	6.8	7.5	7.1	9	158	320	1.2	2.2	0.42	1.79	3	4	63	150	1
20	29	5.1	6.4	7.1	7.9	381	608	1	2.2	0.32	0.67	45	322	210	602	0
22	31	4.2	7.3	7.1	8.5	274	18950	1.5	2.5	0.38	2.82	4	15	2	150	0
27	31	4.5	7.2	7.1	8.6	12230	56900	1.4	3.6	0.63	2.86	4	15	64	120	0
21	30	3.8	5.8	6.6	8.1	802	1326	2.2	3.5	1.08	4.51	191	659	540	1120	0
25	26	6.1	7.5	7.3	7.9	16170	26800	2.4	2.8	1.12	1.8	3	3	63	150	0

图 6.8 模型预测结果

6.1.2 后端核心代码展示

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...
    ]
```

2) 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，后端获取上传的 CSV 文件，使用 pandas 库读取数据集。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)
```

3) 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，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,确保数据质量和完整性，为后续的数据训练提供可靠的数据。

```
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
```

4) 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
```

5) 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
```

6.2 研究结果

6.2.1 过采样算法实现结果

过采样算法的实现结果分析表明，ADASYN（Adaptive Synthetic Sampling）和 SMOTE（Synthetic Minority Over-sampling Technique）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。具体而言，通过 ADASYN 算法，原始数据集的样本数量从 543

显著增长至 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。ADASYN 算法通过自适应地生成少数类样本，重点关注难以分类的样本区域，从而有效地改善了分类器对少数类的识别能力。

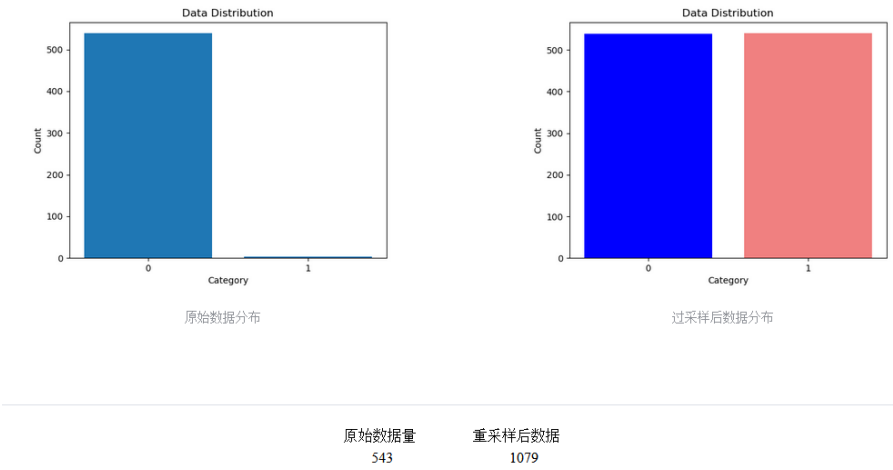


图 6.9 ADASYN 算法结果

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。

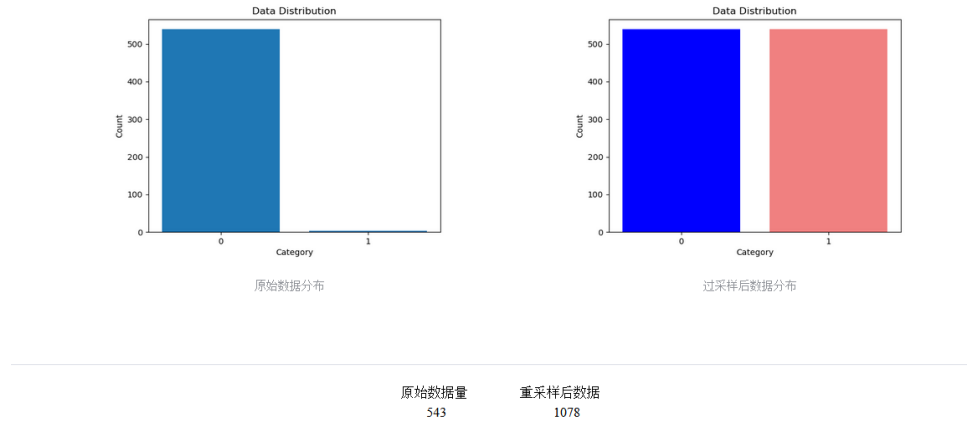


图 6.10 SMOTE 算法结果

6.2.2 机器学模型分析结果

本文对 KNN 模型的结果进行了详细分析。通过超参数迭代实验，图 6.11 展示了 K 值与准确率之间的关系。在不同的 K 值下，模型的准确率有所不同，最终选择 K=1 作为最优超参数，因为此时模型的准确率最高，接近 0.99。KNN 模型的优点在于其简单易懂、无需训练过程以及对异常值不敏感。

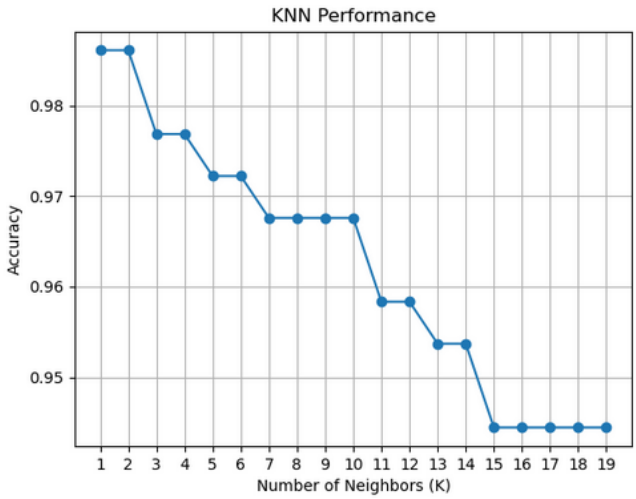


图 6.11 KNN 算法 K 值与准确率关系图

从分类报告图中可以看出，KNN 模型在精确率、召回率和 F1 指标上均表现优异，几乎达到了 1.0。这表明模型在分类任务中具有极高的准确性和稳定性。

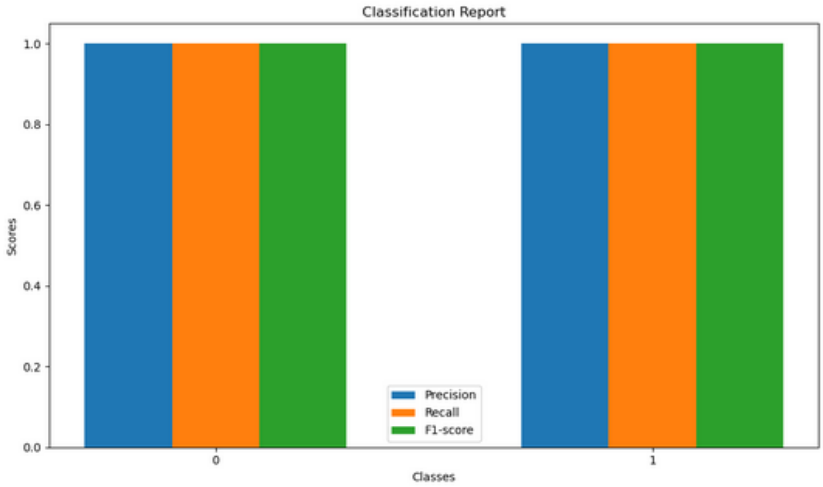


图 6.12 KNN 模型评估指标

本文对 GBDT 模型的结果进行了详细分析。通过超参数迭代实验，图 6.13 展示了学习率与准确率之间的关系。实验确定 GBDT 模型的最优学习率为 0.01，在此学习率下，模型的交叉验证准确率达到了 0.995。GBDT 是一种集成方法，通过顺序构建多棵决策树来提升预测精度。其优点包括强大的预测能力、对数据分布的鲁棒性和处理非线性关系的能力。

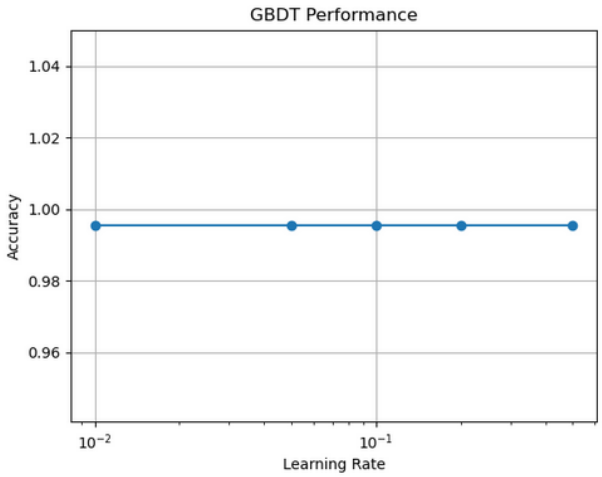


图 6.13 GDBT 算法学习率与准确率关系图

从分类报告图中可以看出，GBDT 模型在精确率、召回率和 F1 指标上均表现优异，达到了 0.989。这表明模型在分类任务中具有极高的准确性和稳定性。综上所述，GBDT 模型在最优学习率为 0.01 时，模型的各项指标均接近完美，显示出了极高的分类能力。然而，在实际应用中，仍需考虑 GBDT 模型的训练时间和参数调优复杂性，以确保其在大规模数据集上的有效性。

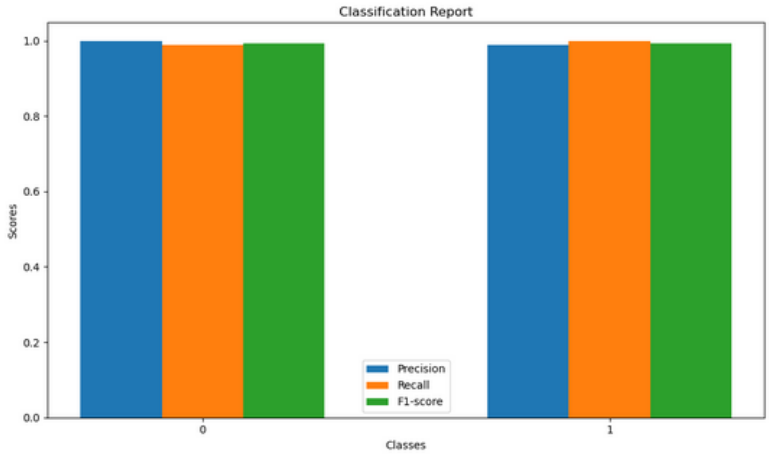


图 6.14 GDBT 算法评估指标

图 6.15 展示了 SVM 的超参数 c 值与准确率之间的关系图，可以看出最优超参数为 1。此外从分类报告图 6.16 中可以看出，SVM 模型在精确率、召回率和 F1 指标上均表现优异，F1 指标几乎达到了 0.994。这表明模型在分类任务中具有极高的准确性和稳定性。综上所述，SVM 模型在本次水质检测任务中表现出色，尤其是在最优 C 值为 1 时，模型的各项指标均接近完美，显示出了极高的分类能力。

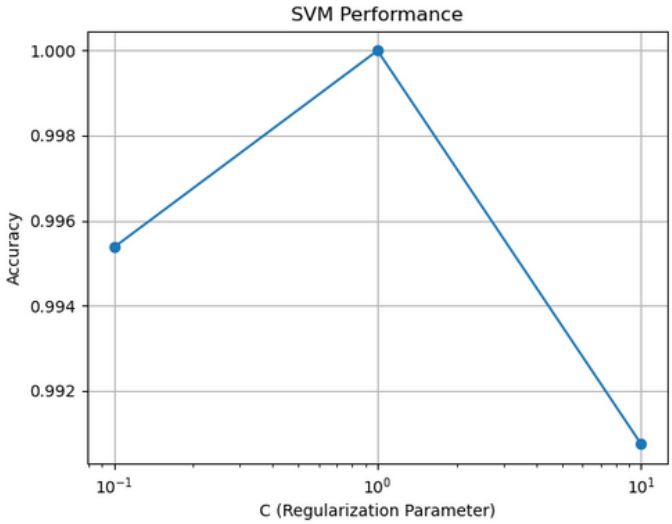


图 6.15 SVM 算法学习率与准确率关系图

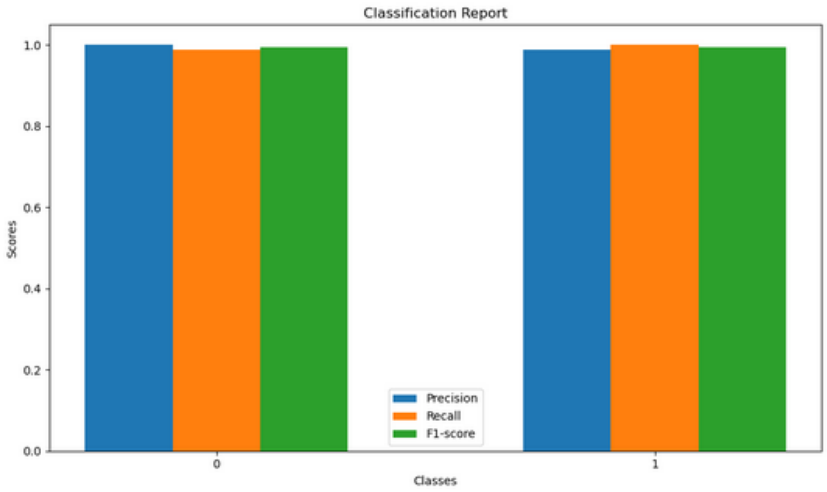


图 6.16 SVM 模型评估指标

从图 6.16 和图 6.17 中可以看出，人工神经网络（ANN）在不同学习率下的表现以及其分类评估指标。图 6.16 展示了 ANN 算法的学习率与准确率之间的关系。在超参数迭代过程中，经过多次实验，最终确定最优学习率为 0.01。此时，模型在训练集和测试集上的表现最佳，准确率为 0.968。图 6.17 展示了 ANN 模型的分

类评估指标。对于类别 0，精确率、召回率和 F1 指标在 0.9 左右，表明模型在识别类别 0 样本时表现良好。对于类别 1，这些指标均接近 1.0，显示出模型在识别类别 1 样本时具有极高的准确性和稳定性。

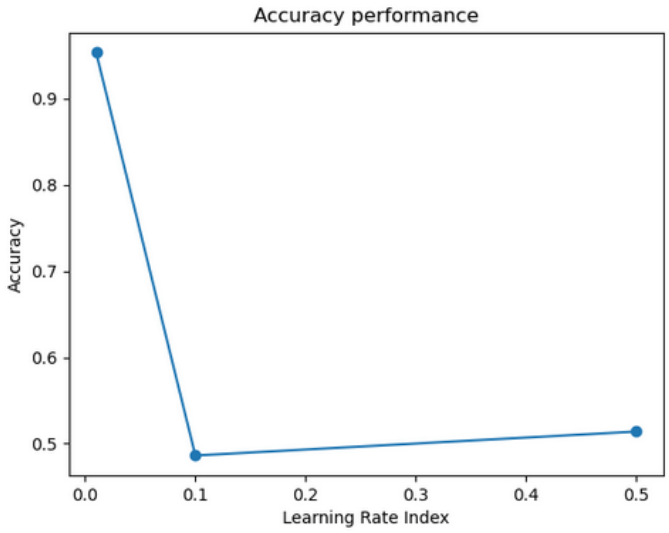


图 6.17 ANN 算法学习率与准确率关系图

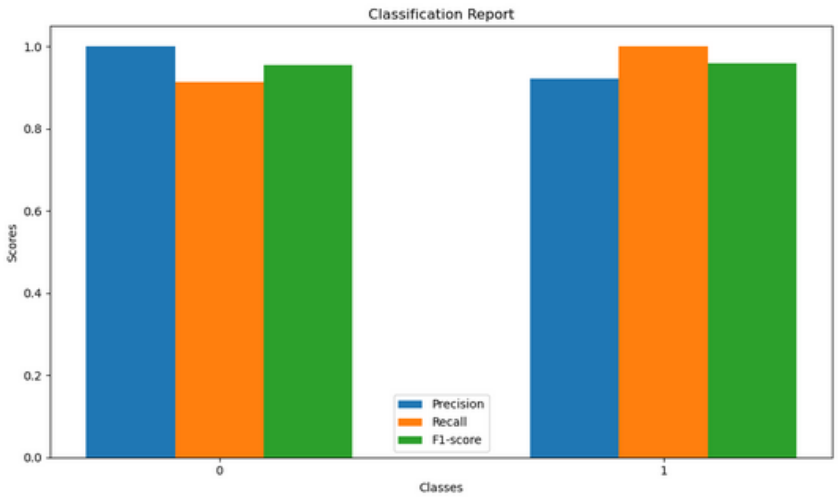


图 6.18 ANN 模型评估指标

6.2.3 各机器学模型结果对比

在对比 KNN、SVM、GBDT 和 ANN 这四种机器学习模型的性能时，我们细致地考察了它们在不同类别上的精确率、召回率和 F1 得分。这些指标帮助我们理解每个模型在训练集上的表现，并指导我们选择最合适的模型。

具体来说，KNN 模型在类别 0 和类别 1 上均获得了完美的分数，精确率、召回率和 F1 得分都达到了理想的 1.0，表明其在这两个类别上的预测是无可挑剔的。

SVM 模型同样表现出色，尤其是在类别 0 和类别 1 上，它的精确率和召回率都达到了 0.989，而 F1 得分为 0.994，这显示了 SVM 在这些类别上的高度准确性。

GBDT 模型在类别 0 上的精确率和召回率达到了 0.998，F1 得分也是 0.998，显示了其在这一类别上的强大实力。在类别 1 上，它的表现与 SVM 相当，精确率和召回率都是 0.989，F1 得分为 0.994，这证明了其在处理这些类别时的有效性。

最后，ANN 模型在类别 0 上也实现了完美的表现，精确率和召回率都是 1.0，F1 得分自然也是 1.0。然而在类别 1 上，尽管召回率非常高，达到了 0.998，但精确率只有 0.915，导致 F1 得分为 0.956。这表明虽然 ANN 模型能够识别出大多数正类样本，但在准确性方面还有提升空间。

表 6.1 模型结果对比

模型名称	最优超参数	类别	精确率	召回率	F1 得分指标
KNN	K 值（1 或 2）	0	1	1	1
		1	1	1	1
SVM	C 值（1）	0	1	0.989	0.994
		1	0.989	1	0.994
GDBT	学习率（0.01）	0	0.998	0.989	0.993
		1	0.989	0.998	0.994
ANN	学习率（0.01）	0	1	0.907	0.951
		1	0.915	1	0.956

总体而言，这些模型在特定的类别上都有其优势和局限性。选择最佳模型需要综合考虑各项指标，以及模型在实际应用中的需求。

第 7 章 结论与展望

7.1 系统总结

本论文致力于开发一个水质检测系统，旨在利用过采样算法和多种机器学习技术对水质数据进行数据处理、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，这类过采样技术可以有效提升数据集中的小众类别样本数量，从而优化类别的分布并增强随后应用的机器学习模型的表现。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。

7.2 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细心的指导与校对，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。

感谢答辩组的各位老师，在答辩过程中的严谨评审和宝贵建议，使我更加深刻地认识到自身研究中的不足与改进方向。感谢各位老师悉心指导和鼓励，您们的智慧与经验不仅让我在论文答辩中受益匪浅，更为我未来的学术道路指明了前进的方向。

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，衷心祝愿大家未来可。