國立臺灣大學工學院環境工程學研究所

碩士論文

Graduate Institute of Environmental Engineering

College of Engineering

National Taiwan University

Master’s Thesis

以機器學習模型預測污水處理廠放流水中的化學需氧量

Prediction of Effluent COD of Wastewater Treatment Plant Using Machine Learning

劉軒成

Xuan-Cheng Liu

指導教授：林逸彬 博士

Advisor: Yi-Pin Lin, Ph.D.

中華民國 114 年 7 月

July, 2025

doi:10.6342/NTU202503249

doi:10.6342/NTU202503249

摘要

隨著人口急速成長和都市化，生活污水量持續攀升，再加上工廠排放與農業灌溉帶來的複雜污染物，污水處理廠（WWTPs）面臨更大處理量與更複雜污染物的雙重挑戰。因此，如何有效監控並預測出流水水質成為當務之急。本研究結合現場感測器數據與機器學習模型，對一工業區污水處理廠出流水 COD（COD.out）進行預測。首先對感測器數據進行清洗以排除異常值，並加入時間延遲分析來捕捉污水處理過程中的停留效應，接著比較隨機森林（Random Forest, RF）、梯度提升機（Gradient Boosting Machine, GBM）與極限梯度提升（Extreme Gradient Boosting, XGB）三種模型，結果顯示 RF 在預測 COD.out 的表現最佳，平均絕對百分比誤差 (Mean Absolute Percentage Error, MAPE) 為 6.22%。此外，以夏普利加成解釋 (SHapley Additive exPlanations, SHAP) 分析各輸入參數包含進水 pH（pH.in）、進水溫度（Temp.in）、氧化渠溫度（Temp.Ox.ditch）、出水 pH（pH.out）、出水溫度（Temp.out）、出水懸浮固體濃度（SS.out）對模型輸出的影響程度，結果顯示，放流池、氧化渠溫度（TEMP.out 和 TEMP.Ox.ditch）以及放流池懸浮固體濃度 (SS.out) 對 COD.out 的影響最為顯著。當 Temp.Ox.ditch 與 Temp.out 維持在 27~32°C 之間、SS.out 低於 2.5 mg/L 時，模型預測的 COD.out 呈下降趨勢。透過重點監測並維持這三個關鍵參數可以有效預測 COD.out。

關鍵字：污水處理廠、出流水化學需氧量、機器學習、水質預測、模型可解釋性

doi:10.6342/NTU202503249

ABSTRACT

As population grows rapidly and urbanization accelerates, the volume of domestic wastewater continues to increase. Coupled with complex pollutants from industrial discharges and agricultural irrigation, wastewater treatment plants (WWTPs) face the dual challenges of higher influent loads and greater pollutant complexity. Therefore, effective monitoring and prediction of effluent water quality in the WWTPs have become a crucial task. In this study, on-site sensor data from an industrial WWTP are combined with machine learning (ML) models to forecast effluent chemical oxygen demand (COD.out). First, sensor readings are cleaned to remove outliers, and time lag analysis is incorporated to capture the retention effects occurring throughout the treatment process. Subsequently, three models, random forest (RF), gradient boosting machine (GBM) and extreme gradient boosting (XGB), were trained and compared. RF delivered the best performance in predicting COD.out, achieving a mean absolute percentage error (MAPE) of 6.22%. SHapley Additive exPlanations (SHAP) analysis was employed to evaluate the influences of each input parameter, including influent pH (pH.in), influent temperature (Temp.in), oxidation ditch temperature (Temp.Ox.ditch), effluent pH (pH.out), effluent temperature (Temp.out), and effluent suspended solids concentration (SS.out) on the model output. The results indicate that the TEMP.Ox.ditch, TEMP.out, and SS.out have the most significant influences on COD.out. When Temp.Ox.ditch and Temp.out are maintained between 27 and 32 °C and SS.out is kept below 2.5 mg/L, the model predicts a declining trend in effluent COD. By focusing on monitoring these three key parameters, the COD.out can be effectively predicted.

Keywords: Wastewater Treatment Plant, Effluent Chemical Oxygen Demand, Machine Learning, Water Quality Prediction, Model Interpretability

doi:10.6342/NTU202503249

CONTENTS

doi:10.6342/NTU202503249

doi:10.6342/NTU202503249

List Of Abbreviations

Term	Abbreviation
adaptive boosting	AdaBoost
artificial neural network	ANN
backpropagation neural network	BP-NN
biochemical oxygen demand	BOD
chemical oxygen demand	COD
coefficient of determination	R^2
continuous water monitoring system	CWMS
covariance matrix	S
decision trees	DT
deep neural network	DNN
dissolved oxygen	DO
electrical conductivity	EC
effluent quality index	EQI
extreme gradient boosting	XGB
gradient boosting machine	GBM
k-nearest neighbors	KNN
light gradient boosting machine	Light GBM
logistic regression	LR
long short-term memory	LSTM
mahalanobis distance	MD
machine learning	ML
mean absolute error	MAE
mean absolute percentage error	MAPE
minimum covariance determinant	MCD
multilayer perceptron	MLP
oxidation-reduction potential	ORP
random forest	RF
recurrent neural network	RNN
robust distance	RD
root mean square error	RMSE
SHapley Additive exPlanations	SHAP
support vector regression	SVR
suspended solids	SS
total dissolved solids	TDS
total nitrogen	TN
total phosphorus	TP
total suspended solids	TSS
visual studio code	VSCode
wastewater treatment plants	WWTPs

doi:10.6342/NTU202503249

LIST OF FIGURES

doi:10.6342/NTU202503249

doi:10.6342/NTU202503249

LIST OF TABLES

doi:10.6342/NTU202503249

doi:10.6342/NTU202503249

Chapter 1 Introduction

1.1 Background

With the rapid growth of the global population and accelerated urbanization, domestic wastewater generated has been continuously increasing. Moreover, the expansion of industrial activities and the rising demand for agricultural irrigation have further exacerbated water pollution (Moss, 2008). As a result, wastewater treatment plants (WWTPs) face the dual challenge of processing larger volumes of wastewater and dealing with increasingly complex pollutants. Ensuring the effectiveness of the treatment processes has become an indispensable responsibility of wastewater treatment industry, in which water quality monitoring and water quality prediction play important roles.

Machine learning (ML) models which are developed using historical data collected from on-site sensors have been employed to uncover the complex relationships between water quality parameters and to predict effluent water quality in WWTPs (Müller and Guido, 2016). Since ML models rely entirely on sensor data, ensuring data integrity is critical. In addition, past studies often overlook the wastewater retention and transport between treatment units, which prevents the data from reflecting true water quality conditions and leads to incorrect patterns and poor accuracy (Wang et al., 2021). Moreover, ML models are frequently regarded as “black boxes” and unable to explain the contributions of input parameters to the final predictions. These challenges must be addressed to ensure the proper applications of ML models to assist water quality predictions and optimal operations of WWTPs.

doi:10.6342/NTU202503249

1.2 Research Objectives

The goal of this study is to apply ML models for effluent water quality predictions in WWTPs. Specific objectives include:

1. To develop a data screening procedure to ensure the integrity of sensing data.
2. To incorporate the “time lag” concept in data processing workflow to reflect the water retention in treatment units.
3. To evaluate the performance of different ML models in predicting effluent water quality.
4. To quantify and explain the impact of each input data feature on the ML predictions.

doi:10.6342/NTU202503249

Chapter 2 Literature review

2.1 Water Quality Monitoring in WWTPs

In WWTPs, multiple key water quality parameters are monitored, including temperature for evaluating thermal contamination, biochemical oxygen demand (BOD) and chemical oxygen demand (COD) for assessing organic load; total suspended solids (TSS) for evaluating particle removal efficiency; electrical conductivity (EC) for assessing the concentration of dissolved ions, serving as an indirect measure of total dissolved solids, and total nitrogen (TN) and total phosphorus (TP) for evaluating nutrient removal performance (Tchobanoglous et al., 2014). If a WWTP fails to control its effluent quality to meet the standards required by environmental regulations, it can cause ecological damage to natural water bodies. For example, when wastewater with a high residual organic content is discharged into a natural water body, it consumes excess dissolved oxygen (DO), which adversely affects the survival of aquatic organisms (Von Sperling, 2007).

Traditional laboratory analyses provide high accuracy measurements of water quality parameters but require a relatively long time (hours or days) to complete sampling and testing. Therefore, abnormal influent water quality, failures in water treatment processes, and violations of discharge standards could not be detected and responded immediately. In contrast, on-site sensors continuously monitor water quality, providing real time information to prompt corrective measures if the situations mentioned above occur. Currently in Taiwan, a continuous water monitoring system (CWMS) is implemented according to the Ministry of Environment’s regulations, in which temperature, pH, EC, COD and suspended solids (SS) in the effluent are continuously

doi:10.6342/NTU202503249

monitored in industrial WWTPs (Ministry of Environment, 2024).

There is an increasing demand to digitalize traditional WWTPs to improve water quality management. The application of ML models using historical data collected from on-site sensors has been considered to reveal complex relationships among different water quality parameters and to predict water quality under different scenarios, enabling WWTPs operators to take action before abnormality occurs (Safder et al., 2022). Although ML model is a powerful tool, its performance critically depends on the quality of the input data. Water quality sensors can be affected by malfunctions, connectivity errors, and other factors that degrade data quality (Szeląg et al., 2017). Consequently, it is essential to identify anomalous data before feeding them into ML models.

2.2 Categories of ML

ML can be categorized into supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning (Mohammed et al., 2016). Among these, supervised learning relies on labeled datasets, meaning each input sample is paired with a known output. It encompasses traditional algorithms such as decision trees (DT), logistic regression (LR), multiple linear regression (MLR), support vector regression (SVR) and k-nearest neighbors (KNN), and extends to ensemble learning like random forests (RF), gradient boosting machines (GBM) and extreme gradient boosting (XGB), adaptive boosting (AdaBoost), light gradient boosting machine (Light GBM) (Murphy, 2012). Ensemble learning combines multiple simple weak learners, typically DT, which can either function independently as a single model or serve as the fundamental units of a larger ensemble (Hastie et al., 2009). Ensemble learning trains models on different random subsets of data and averages their predictions to smooth out noise. It can also build models sequentially, with each new model focusing on correcting the remaining

doi:10.6342/NTU202503249

errors to gradually reduce overall errors and boost accuracy. By aggregating many weak learners, ensemble learning can even surpass a single complex model, such as a deep neural network (Zhou, 2012). A deep neural network is inspired by the human brain and consists of layers of interconnected nodes that learn to recognize relationships in data. Supervised learning also incorporates deep learning models such as artificial neural network (ANN), multilayer perceptron (MLP), backpropagation neural network (BP-NN), deep neural network (DNN), recurrent neural network (RNN), and long short-term memory (LSTM) (Chollet, 2017).

Unsupervised learning relies on unlabeled datasets, allowing models to discover internal data structures and perform clustering (James et al., 2013). Semi-supervised learning is between these two approaches: it initially trains a model with a small set of labeled datasets and then uses a large amount of unlabeled datasets for refinement (Mohammed et al., 2016). Reinforcement learning does not depend on labeled or unlabeled datasets but instead learns by interacting with the environment and receiving feedback (Alpaydin, 2006).

2.3 Application of ML models in Water Quality Management in WWTPs

In recent years, ML has been applied to WWTPs for water quality management. For instance, Qambar and Al Khalidy (2022) proposed a dynamic ML model for real time influent BOD prediction to optimize the operation of aeration tanks in WWTPs. Unlike the traditional use of DO control threshold, the ML model adjusts the DO concentration based on actual influent conditions, leading to a 23% reduction in energy consumption. Wang et al. (2022) used RF, XGB and LightGBM to predict effluent TSS at WWTP. The

doi:10.6342/NTU202503249

results showed that XGB performed best among the three models and that influent temperature is a critical parameter. Nasir and Li (2024) used ANN, GBM, RF, XGB and a hybrid RF-GBM model to predict effluent BOD at WWTP. The ANN achieved the most accurate predictions, enabled real-time BOD monitoring, eliminating the 5-7 day waiting time for traditional lab tests and reducing labor costs. Mahanna et al. (2024) used ML models such as LR, RF, GBM, and SVR to predict the removal efficiencies of SS, COD, and BOD in WWTPs. The results showed that RF was the best model in predicting COD and SS removal efficiencies while GBM performed the best in predicting BOD removal efficiency. The importance of input parameters was also analyzed, revealing that influent COD and total dissolved solids (TDS) were the most influential parameters for both COD and BOD removal efficiencies, whereas influent SS and TDS were most critical for SS removal efficiency. Accordingly, operators should closely monitor these parameters. Manav-Demir et al. (2024) used XGB, LightGBM, SVR and RF to predict effluent COD and BOD at WWTP. The results indicated that SVR achieved the best accuracy. By integrating on-site sensors, the model can provide real-time predictions and alerts to support WWTP operations and decision-making. Cechinel et al. (2024) used SVM, LSTM, MLP and RF to predict effluent COD at WWTP. The results showed that LSTM performed best. The importance of input parameters was also analyzed, revealing that influent TSS was the most significant parameter affecting effluent COD. Ye et al. (2024) used MLR, BP-NN, SVR, DNN and XGB to predict effluent BOD at WWTP. The results showed that XGB outperformed other models. Effluent COD was identified as the most influential parameter affecting effluent BOD. Bo-Qi et al. (2025) used AdaBoost, BP-NN, SVR, XGB and GBM to predict a composite effluent quality index (EQI) at WWTP. EQI was defined as the weighted sum of effluent BOD, COD, TN, TP and SS concentrations. The results showed that XGB outperformed other models and can be used to forecast effluent water quality in real time, enabling operators to adjust operational parameters, such as aeration rate, preventing pollutant concentrations from exceeding discharge stahdards. Wang et al. (2025) used RF, LSTM, RNN and SVM to predict N_2O emissions at WWTP. The results showed that RF was the model with the highest accuracy. The study demonstrated that traditional monitoring methods are limited by high costs, time consuming and complex procedures, while ML can directly extract hidden relationships from historical sensor data for N_2O level prediction, thereby enabling early warnings for possible abnormalities.

2.4 Challenges and Limitations of Water Parameters Predictions in WWTPs

Although ML models have been applied for water quality management in WWTPs, relatively poor prediction performances are also reported in some studies. For instance, Bagherzadeh et al. (2021) found that GBM reached an R^2 of 0.58 for influent TN prediction; Cechinel et al. (2024) showed that SVR attained an R^2 of 0.60 for effluent COD prediction; and Manav-Demir et al. (2024) reported that when predicting TN and TP in the effluent of a WWTP, RF and XGB predictions produced relatively high MAPE of 0.34 and 0.27, respectively.

One possible reason for the poor predictions is that water retention in each treatment processes in WWTPs were not considered. Clarifying the impact of the retention or “time lag” in each treatment process is essential to correctly capture the relationship between the water quality parameters inside the WWTP and those in the effluent (Wang et al., 2021). Toivonen and Räsänen (2024) found that the influent COD impacts the effluent COD after about 23.25 hr. They also showed that DO begins to affect COD removal

doi:10.6342/NTU202503249

efficiency after roughly 100 hr. Therefore, incorporating time lag of data and understanding the appropriate time series between water quality parameters are important for ML application in WWTPs. Moreover, even the ML models demonstrate good performance, their results remain difficult to interpret due to the “black box” natures of the ML models. Therefore, search for a method that can be used to interpret the results from ML models is essential for the application of ML models to WWTP operations.

doi:10.6342/NTU202503249

Chapter 3 Materials and Methods

3.1 Research Flowchart

The research flowchart of this study is shown in Figure 1. The water quality data acquired from different treatment processes in an industrial WWTP are collected and processed through a series of data cleaning steps to maintain data integrity. The cleaned data were analyzed using correlation analysis to identify the key water quality parameters affecting effluent COD. The data were then used as the input for various ML models to evaluate their prediction performance to determine the best ML model. Finally, SHAP is used to quantify the contribution of each important feature to the COD predictions, thereby identifying the most influential factors affecting water quality.

This simulation was implemented using Python 3.10 and conducted within the Visual Studio Code (VSCode) development environment for programming and execution. The construction, training, optimization, and evaluation of all ML models were performed in VSCode. The simulation was conducted on a workstation equipped with an AMD Ryzen 9950X3D CPU, which is the main processor that runs the computer and handles most tasks; 32 GB of DDR5 RAM, which is the memory that helps the computer run programs faster by temporarily storing data; and an NVIDIA RTX 5080 GPU with 16 GB of dedicated memory, a special processor that helps speed up calculations, especially those used in ML.

doi:10.6342/NTU202503249

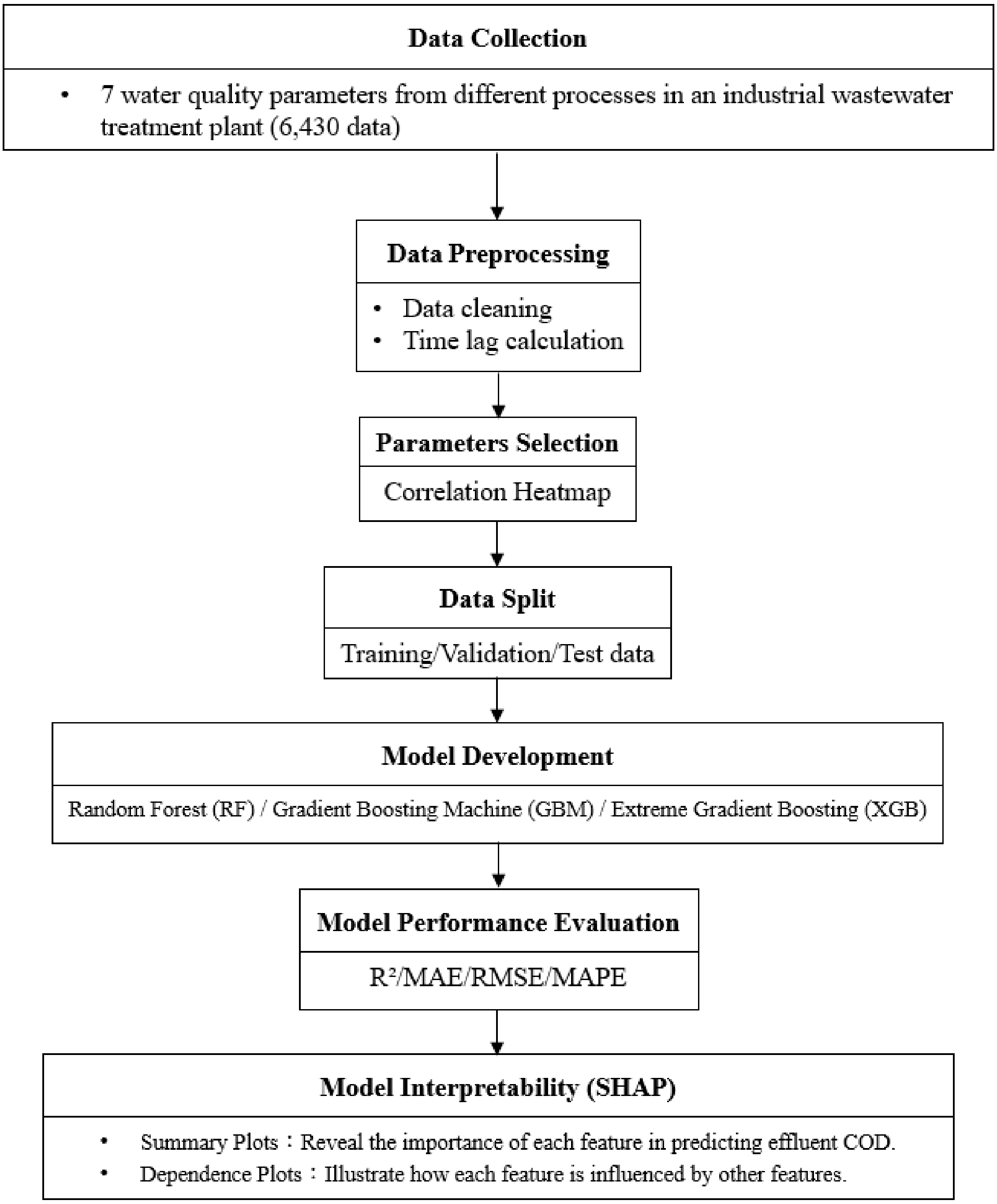

Data Collection
• 7 water quality parameters from different processes in an industrial wastewater treatment plant (6,430 data)
Data Preprocessing
• Data cleaning
• Time lag calculation
Parameters Selection
Correlation Heatmap
Data Split
Training/Validation/Test data
Model Development
Random Forest (RF) / Gradient Boosting Machine (GBM) / Extreme Gradient Boosting (XGB)
Model Performance Evaluation
R²/MAE/RMSE/MAPE
Model Interpretability (SHAP)
• Summary Plots：Reveal the importance of each feature in predicting effluent COD.
• Dependence Plots：Illustrate how each feature is influenced by other features.

Figure 1. Research Flowchart

doi:10.6342/NTU202503249

3.2 WWTP data collection

The data were collected from an industrial WWTP located in central Taiwan. The treatment processes employed in the WWTP are shown in Figure 2. The wastewater first passes through physical treatment processes, including bar screens, grit chamber, and primary clarifier, to remove large particles. Then, the wastewater enters an oxidation ditch to break down organic matter, followed by the secondary clarifier, allowing sludge to settle and separate from the clarified supernatant. Finally, the supernatant undergoes coagulation, flocculation and sedimentation before discharge.

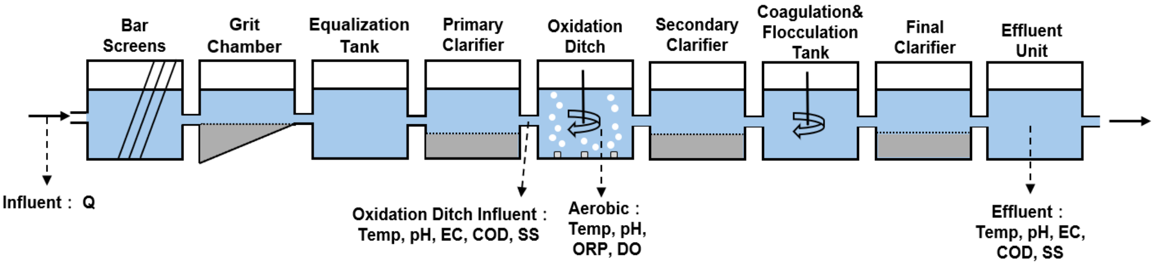

Bar Screens
Grit Chamber
Equalization Tank
Primary Clarifier
Oxidation Ditch
Secondary Clarifier
Coagulation& Flocculation Tank
Final Clarifier
Effluent Unit
Influent : Q
Oxidation Ditch Influent : Temp, pH, EC, COD, SS
Aerobic : Temp, pH, ORP, DO
Effluent : Temp, pH, EC, COD, SS

Figure 2. Treatment processed employed in the WWTP

Data collected from on-site sensors between January 1, 2024, and September 30, 2024, were used for this study. The monitored water quality parameters include temperature (Temp), pH, EC, COD, SS, oxidation-reduction potential (ORP), and DO. The sensor deployment is shown in Table 1. All data points are collected at an hourly frequency, with a total of 6,430 data points.

doi:10.6342/NTU202503249

Table 1. Sensor deployment in the WWTP

Sensors Location	**Monitored Parameters**
Oxidation ditch inlet	Temperature (TEMP.in), pH (pH.in), Conductivity (EC.in), Chemical Oxygen Demand (COD.in), Suspended Solids (SS.in)
Oxidation ditch	Temperature (TEMP.Ox.ditch), pH (pH.Ox.ditch), Oxidation Reduction Potential (ORP.Ox.ditch), Dissolved Oxygen (DO.Ox.ditch)
Effluent unit	Temperature (TEMP.out), pH (pH.out), Conductivity (EC.out), Chemical Oxygen Demand (COD.out), Suspended Solids (SS.out)

3.3 Data Preprocessing

3.3.1 Data Cleaning

To ensure data validity, data preprocessing to remove missing data, invalid data, and outliers is required. For the monitored water quality parameters (Table 1), except ORP.Ox.ditch, should be greater than or equal to zero and any negative values found in the data must be removed. Additionally, if the data contain ten or more consecutive identical values, which may be caused by sensor malfunction or connection anomalies, these repeated values are considered invalid and removed from the dataset.

To determine the outliers, multivariate analysis incorporating the correlation between parameters was employed. The covariance matrix (S, Equation (1)) was used to evaluate the correlation between two parameters and the mahalanobis distance (MD, (Equation (3)) was calculated. MD measures the distribution of univariate data points using standard deviations, taking the correlation between variables in a multivariate context into account. If the variables are highly correlated, the S value is large and the MD value is reduced, indicating that the point is close to the mean and is not seen as an

doi:10.6342/NTU202503249

outlier. Conversely, if the variables are less correlated, the S value is small and the MD value is enlarged, reflecting that the point is an outlier. The covariance matrix and Mahalanobis Distance are integrated with the Chi-Square Distribution to identify outliers (Murphy, 2012).

$$\mathrm{S} = \begin{bmatrix} S_{11} & \cdots & S_{1k} \\ \vdots & \ddots & \vdots \\ S_{k1} & \cdots & S_{kk} \end{bmatrix} \qquad \text{Equation (1)}$$

$$S_{pq} = \frac{1}{n-1}\sum_{i \in N}(x_{i,p} - \bar{x}_p)\,(x_{i,q} - \bar{x}_q)^T \qquad \text{Equation (2)}$$

$$MD_i = \sqrt{(x_i - \bar{x})^T \cdot S^{-1} \cdot (x_i - \bar{x})}\ \ i = 1,2,\ldots,\mathrm{n} \qquad \text{Equation (3)}$$

where, k is number of features, p and q ranges from 1 to k, n represents the total number of samples, x_i represents a data point, $\bar{x}$ denotes the mean vector of the data, *N* is the sample index set, *T* denotes the transpose operator, which converts a column vector into a row vector and vice versa.

The estimation of the covariance matrix is based on the entire dataset, and the presence of outliers may affect the accuracy of the covariance matrix. To reduce the interference of outliers on matrix estimation, the minimum covariance determinant (MCD, Equation (4)) was adopted (Yoon et al., 2019). This method randomly selects a subset of the dataset of size h = 0.75 n and calculates its covariance determinant. The selection is repeated until the subset with the smallest determinant is found. Based on this subset, the mean and covariance matrix are calculated, thereby improving the matrix's accurate reflection of the central tendency of the data. The distance calculated using the above matrix is referred to as the robust distance (RD, Equation (6)). When combined with the

Chi-Square Distribution, it can be used to identify outliers. Data points falling outside the range defined by the Chi-Square Distribution are regarded as outliers and removed.

$$S_{MCD} = \begin{bmatrix} (S_{MCD})_{11} & \cdots & (S_{MCD})_{1k} \\ \vdots & \ddots & \vdots \\ (S_{MCD})_{k1} & \cdots & (S_{MCD})_{kk} \end{bmatrix} \qquad \text{Equation (4)}$$

$$(S_{MCD})_{pq} = \frac{1}{h-1} \sum_{i \in H} (x_{i,p} - \bar{x}_{MCD,p})\,(x_{i,q} - \bar{x}_{MCD,q})^T \qquad \text{Equation (5)}$$

$$RD_i = \sqrt{(x_i - \bar{x}_{MCD})^T \cdot S_{MCD}^{-1} \cdot (x_i - \bar{x}_{MCD})} \quad i = 1,2,\ldots,\mathrm{h} \qquad \text{Equation (6)}$$

where, h represents the number of samples in the subset, H is the sample index set.

3.3.2 Time Lag Calculation

In the WWTP, wastewater flows through different treatment units in sequence. The data synchronously collected by the sensors in different treatment units at the same time reflect the characteristics of different batches of wastewater. This fact results in time delays between the data collected by sensors in different treatment units and affects the correlations between water quality parameters. To address this time delay issue, correlation analysis is employed to explore the relationships between each parameter and effluent COD to calibrate the data collected at the same time.

To quantitatively analyze the time delay effect, two correlation assessment methods were used, namely the Pearson product-moment correlation and Spearman’s rank correlation. Additionally, the Jackknife method was used to enhance their robustness (Stehlík et al., 2023). These two methods calculate the correlation coefficient between each water quality parameter and the target effluent COD at different time delays. The

Pearson product-moment correlation coefficient (r) measures the linear relationship between variables, while the Spearman's rank correlation coefficient calculates the correlation after ranking the data to capture nonlinear relationships. The Jackknife method calculates the Pearson product-moment correlation and Spearman's rank correlation coefficient by removing one data point at a time and averaging all the recalculated values. This method helps to assess the robustness of the results and reduces the influence of any single data point on the final correlation coefficient. The formula applies to Pearson product-moment correlation coefficient on raw data and to Spearman's rank correlation coefficient after converting the raw data into ranks.

$$r = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \overline{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \overline{Y})^2}} \qquad \text{Equation (7)}$$

where X_i and Y_i represent the observations of two data sets, and $\overline{X}$ and $\overline{Y}$ represent the respective mean values of these two data sets, and n is the total number of data points.

The results from the two statistical methods will be compared and the lag value corresponding to the largest absolute correlation coefficient will be selected as the best time delay. This captures the time delay effect in each parameter's data and recovers the characteristics of different batches of wastewater, providing more accurate data for training models.

3.4 Feature Selection and Data Extraction

The water quality parameters that showed relatively higher Pearson product–moment correlation coefficients with the target variable (COD.out) are designated as "features". Only these features are selected for subsequent model training to prevent

doi:10.6342/NTU202503249

potential interferences from model prediction and to avoid an excessive computational load that would prolong the training time.

After feature selection, the data were split into a 70% training dataset (4,501 data points) and a 30% test dataset (1,929 data points). The training dataset is used to fit the ML models, allowing them to learn patterns and relationships between features and the target variable. The test dataset is used to evaluate the model performance on unseen data.

3.5 ML Model Selection

Three ML models, including Random Forest (RF), Gradient Boosting Machine (GBM), eXtreme Gradient Boosting (XGB), are employed in this study for effluent COD prediction.

RF is an ensemble learning method that combines the predictions of multiple DT for judgment. This method uses Bagging to resample the dataset, ensuring that the training samples for each DT are different. During the node splitting process of each DT, random features are selected for splitting to reduce the correlation between the trees (Lakshmanaprabu et al., 2019). As shown in the Figure 3, for regression problems with predicted values, the final prediction of the RF is the average of the predictions from all DT. RF excels in handling high-dimensional data, capturing nonlinear relationships between variables.

doi:10.6342/NTU202503249

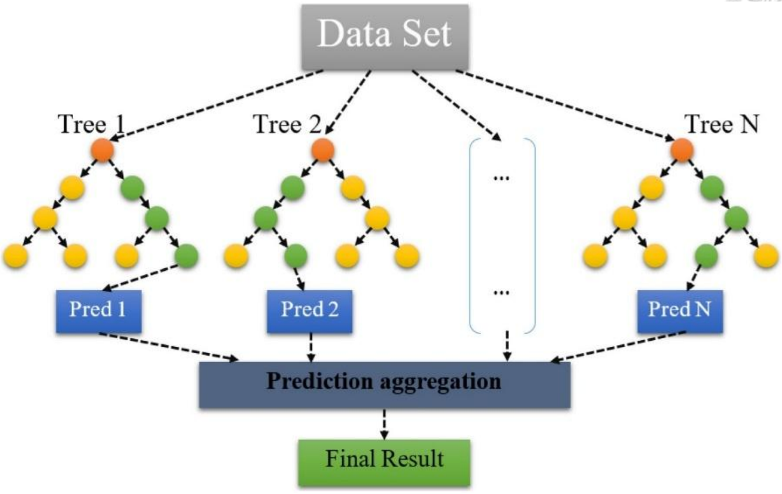


Figure 3. RF structure (Bagherzadeh et al., 2021)

GBM is an ensemble learning model based on boosting that employs gradient descent to minimize the loss function. The gradient can be seen as a direction for adjusting the model, guiding how the model should update in order to reduce prediction error. The gradient is calculated by taking the partial derivative of the loss function L(y, F)=$0.5(\mathrm{y}-\mathrm{F}(\mathrm{x}))^2$ with respect to F(x), resulting in $-(\mathrm{y}-\mathrm{F}(\mathrm{x}))$ (Friedman, 2001). The gradient and the residual usually only differ by a negative sign. As a result, the gradient can be interpreted through the residual, with the negative gradient serving as the direction for adjustment. Typically, the residual, which is the difference between the predicted value and the actual value, i.e., y - F(x), is calculated from the prediction F(x) of the current model. Then, a weak learner is trained to fit this residual, such that h(x) $\approx$ y - F(x). The weak learner h(x) is added to the original model F(x) to obtain a new model F(x) + h(x). This process is repeated iteratively, with the residuals of the previous model training the new weak learner, until the predetermined number of training iterations is reached or the model's performance is satisfactory. The final model is the sum of all these terms: F(x) + h(x) + ⋯ .

XGB is an ensemble learning algorithm that combines the advantages of both

doi:10.6342/NTU202503249

Bagging and Boosting. XGB incorporates random feature selection when constructing each tree, while maintaining the learning characteristics of gradient boosting, enabling each tree to correct the errors made by the previous tree (Chen and Guestrin, 2016). Furthermore, XGB incorporates L1 regularization (alpha) and L2 regularization (lambda) on leaf weights to control model complexity and prevent overfitting.

To prevent the model from overfitting (memorizing noise in the training data) or underfitting (failing to capture underlying trends), K-fold validation was employed to evaluate model performance, setting K = 5. In 5-fold cross-validation, the training set is evenly divided into 5 subset. In each fold, 4 subsets are used for training and the remaining subset is used for validation. This approach ensures that each data point serves as both a training and a validation example at different stages, preventing the model from focusing on only one portion of the dataset.

3.6 Model Performance Evaluation

Selecting the best model is a key step in the ML process. In regression models, the smaller the difference between the predicted results and the actual values, the better the model performance. To conduct a comprehensive and objective evaluation, four commonly used evaluation metrics, namely mean absolute error (MAE), mean absolute percentage error (MAPE), root mean square error (RMSE), and coefficient of determination (R^2) were employed.

MAE is the average absolute error between the predicted values and the actual values. Since the error is taken as the absolute value, it avoids the issue of positive and negative errors canceling each other, making it less sensitive to extreme values compared to other metrics. MAE is suitable for situations where extreme values are prevalent in the training data. MAPE is the absolute percentage error between the actual values and the predicted

doi:10.6342/NTU202503249

values, averaged across all data points. The lower the MAPE, the more accurate the predictions. MAPE is useful because it converts the error into a percentage, avoiding the need to consider the unit of the data. However, it cannot be used when the target value (y) contains zero. RMSE is based on mean square error (MSE), which calculates the average of the squared differences between the actual values and the predicted values. The square feature penalizes extreme values (outliers), making RMSE more sensitive to them. RMSE is derived by taking the square root of MSE, with the primary goal of keeping the unit consistent with the actual values. Finally, R² measures the goodness-of-fit of the model by calculating the difference between the variation of the actual values and the squared errors of the predicted values. The closer R² is to 1, the higher the model's goodness-of-fit. Through the comprehensive analysis of these evaluation metrics, the model performance can be fully evaluated. The equations for calculating MAE, MAPE, RMSE, and R² are shown in Equations (8)-(11).

$$R^2 = \frac{\sum_{i=1}^{N}(\widehat{y_i} - \bar{y})^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \qquad \text{Equation (8)}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\widehat{y_i} - y_i| \qquad \text{Equation (9)}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\widehat{y_i} - y_i)^2} \qquad \text{Equation (10)}$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \widehat{y_i}}{y_i}\right| \times 100\% \qquad \text{Equation (11)}$$

where y_i represents the actual value, $\hat{y}_i$ represents the model's predicted value, $\bar{y}$ represents the mean value, N is the total number of data points.

doi:10.6342/NTU202503249

3.7 SHapley additive exPlanations (SHAP)

Because the ML models are regarded as "black boxes," their internal computations are not transparent to users. To verify which features the model relies on, the SHAP is used. SHAP is a feature attribution method based on cooperative game theory (Lundberg and Lee, 2017). By comparing predictions with and without a given feature, SHAP computes each feature's contribution to the final prediction, thereby quantifying its impact on the model's output.

$$\phi_i = \sum_{S \in F \setminus \{i\}} \frac{|S|!(M-|S|-1)!}{M!}[f_x(S \cup \{i\}) - f_x(S)] \qquad \text{Equation (12)}$$

where ϕ_i is the SHAP value of feature i, *F* represents the set of features (with a total of M features), *S* represents any subset of features that does not include the i-th feature, $f_x()$ represents the model's prediction function, $S \cup \{i\}$ is the subset *S* plus feature i.

Chapter 4 Results and Discussion

4.1 Analysis and Preprocessing of WWTP Data

Water quality data detected using 14 sensors (Table 1) in an industrial WWTP over a period of 9 months were used in this study to develop a ML model for effluent COD prediction. A total of 6,430 data were recorded. Since some of the data could be erroneous due to equipment malfunctions, connection interruptions, sensor fouling and other issues, direct use of these data in model training could introduce bias and deteriorate the model development. Therefore, these data must first undergo cleaning to remove missing data, invalid data, or outlier before model development.

Blanks resulting from the “no response” of the sensors in the dataset are treated as missing data. Negative values, except the ORP in the oxidation ditch (ORP.Ox.ditch), and ten or more identical consecutive measurements are considered as invalid data. The numbers of missing and invalid data detected in the dataset are summarized in Table 2.

For outlier detection, the RD for each sample based on the MCD was calculated and subjected to the chi-square distribution test (Section 3.3.1). At a significance level of α = 0.01 with 14 degrees of freedom, the corresponding chi-square critical value is 29.14. Therefore, when the RD of a sample exceeds 29.14, it is classified as an outlier at the 99% confidence level. Multivariate detection was performed to compute the RD value for each data point. Among the 6,430 data points, a total of 262 were classified as outliers. Values obtained from linear interpolation were used to replace the removed data points as data continuity is required for model development.

doi:10.6342/NTU202503249

Table 2. Summary of Missing and Invalid Data Counts for Each Water Quality Parameter

	Missing data	Invalid data	
		Negative values	Consecutive identical values
Q	0	0	0
pH.in	5	6	0
TEMP.in	5	0	0
EC.in	5	0	0
SS.in	5	0	124
COD.in	5	0	0
pH.Ox.ditch	0	54	0
TEMP.Ox.ditch	0	0	88
ORP.Ox.ditch	0	--	665
DO.Ox.ditch	0	53	1509
pH.out	37	0	10
TEMP.out	37	0	0
EC.out	37	0	0
SS.out	59	0	0
COD.out	59	0	60

Figure 4, Figure 5, and Figure 6 illustrate the data for the water quality parameters over the 9-month period before and after data cleaning in oxidation ditch influent, oxidation ditch, and effluent, respectively. Using the pH in oxidation ditch effluent as an example, before data cleaning several pH values dropped into negative ranges or far below the normal pH range (Figure 4(a)). After cleaning, all negative values and anomalous spikes were removed and linear interpolation restored the pH values to a reasonable range (Figure 4(f)). Similar for water temperature, the readings occasionally approached 0 °C or deviated significantly from the reasonable range (Figure 4(b)). After cleaning, these erroneous points were removed and replaced to restore the temperature to a reasonable range of approximately 17 °C to 35 °C to reflect the realistic temperature condition in central Taiwan (Figure 4(g)).

After data cleaning, the maximum, minimum, mean, and standard deviation for each water parameter are presented in Table 3. For the target variable COD.out, the maximum,

doi:10.6342/NTU202503249

minimum, mean and standard deviations are 97.4 mg/L, 9.8 mg/L, 27.0 mg/L, and 7.8 mg/L, respectively.

doi:10.6342/NTU202503249

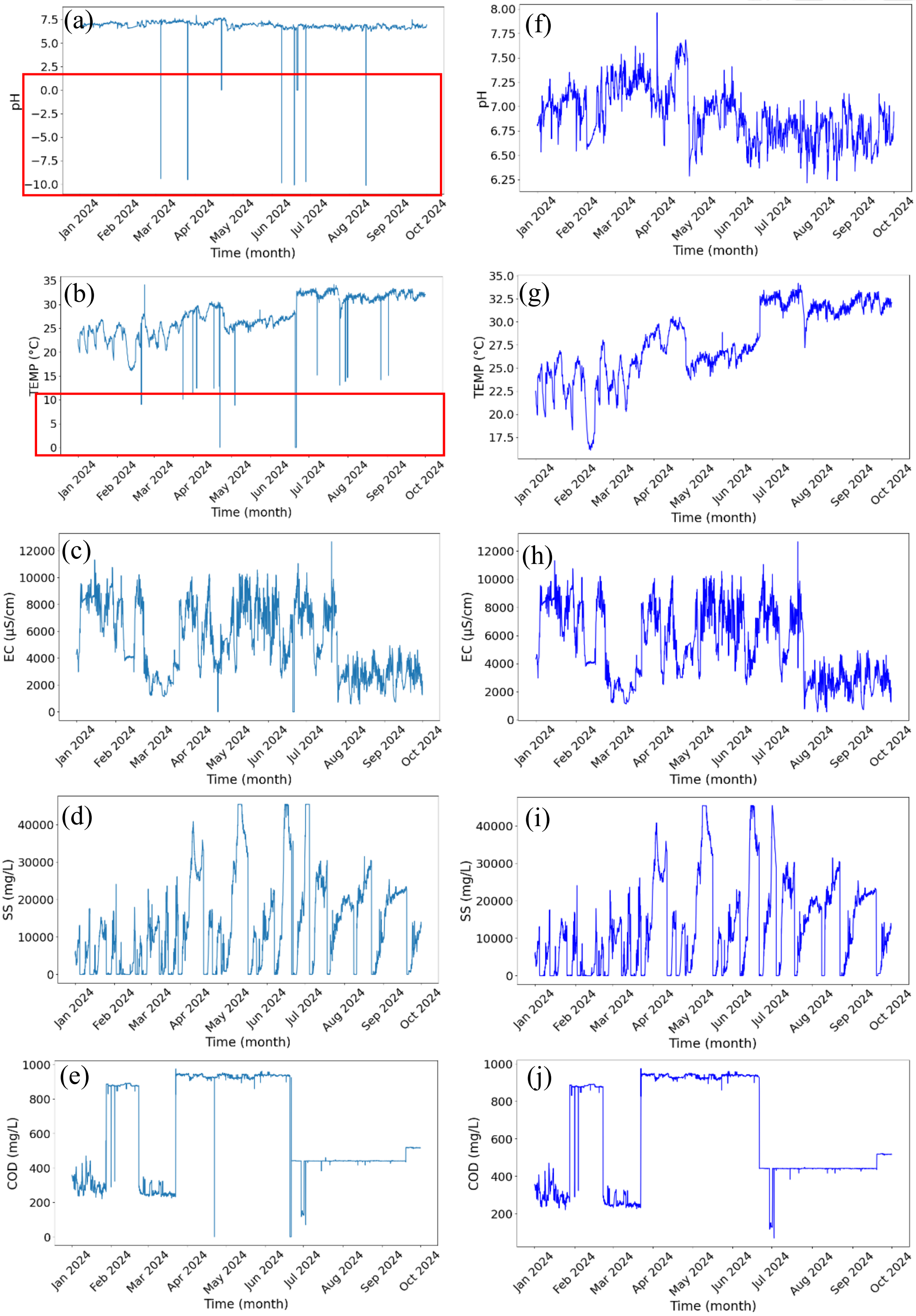


Figure 4. Water quality parameters at the oxidation ditch influent over a 9-month period. (a) pH, (b) TEMP, (c) EC, (d) SS, and (e) COD before data cleaning; (f) pH, (g) TEMP, (h) EC, (i) SS, and (j) COD after data cleaning

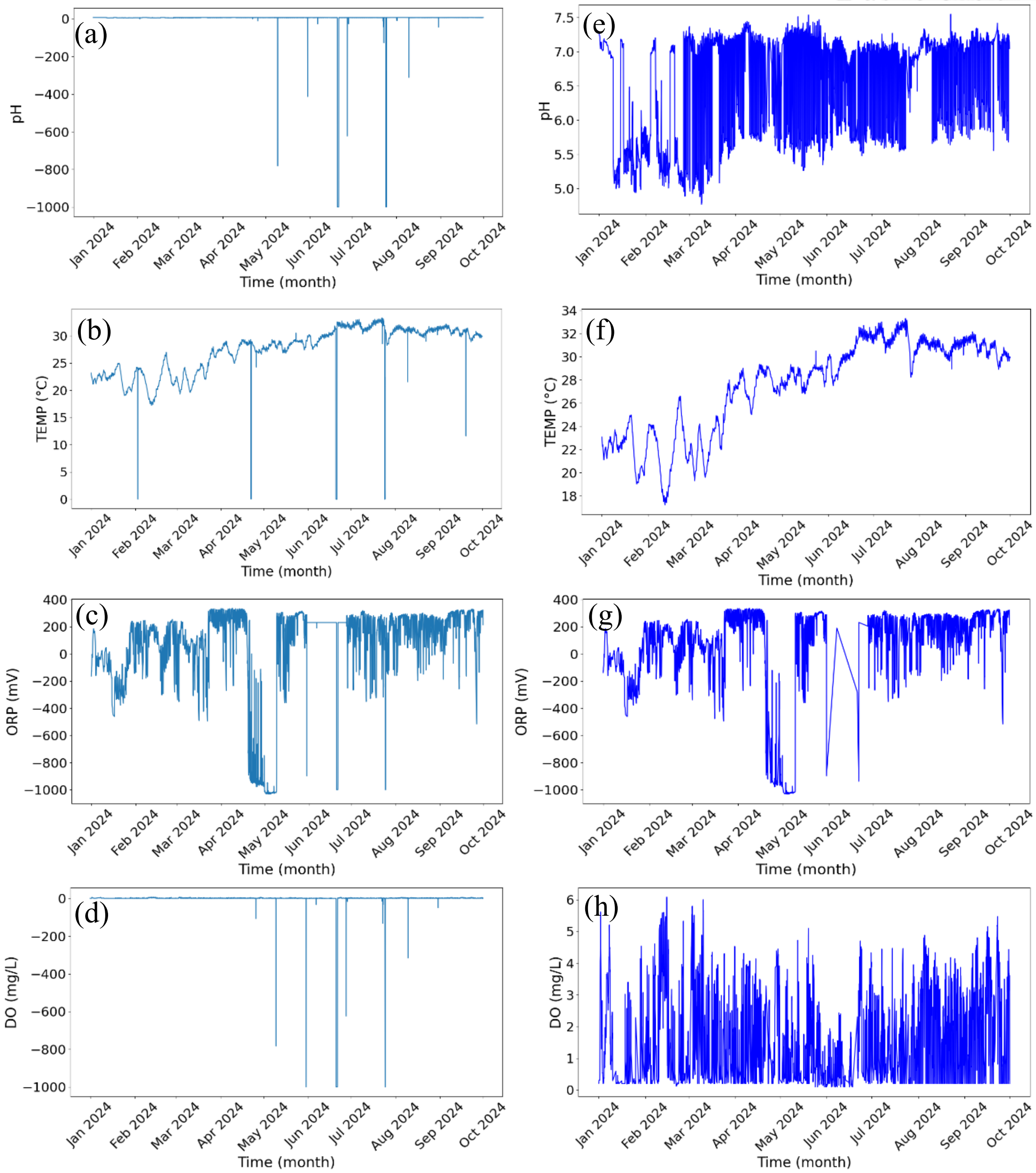


Figure 5. Water quality parameters in the oxidation ditch over a 9-month period : (a) pH, (b) TEMP, (c) ORP, and (d) DO before data cleaning; (e) pH, (f) TEMP, (g) ORP, and (h) DO after data cleaning

doi:10.6342/NTU202503249

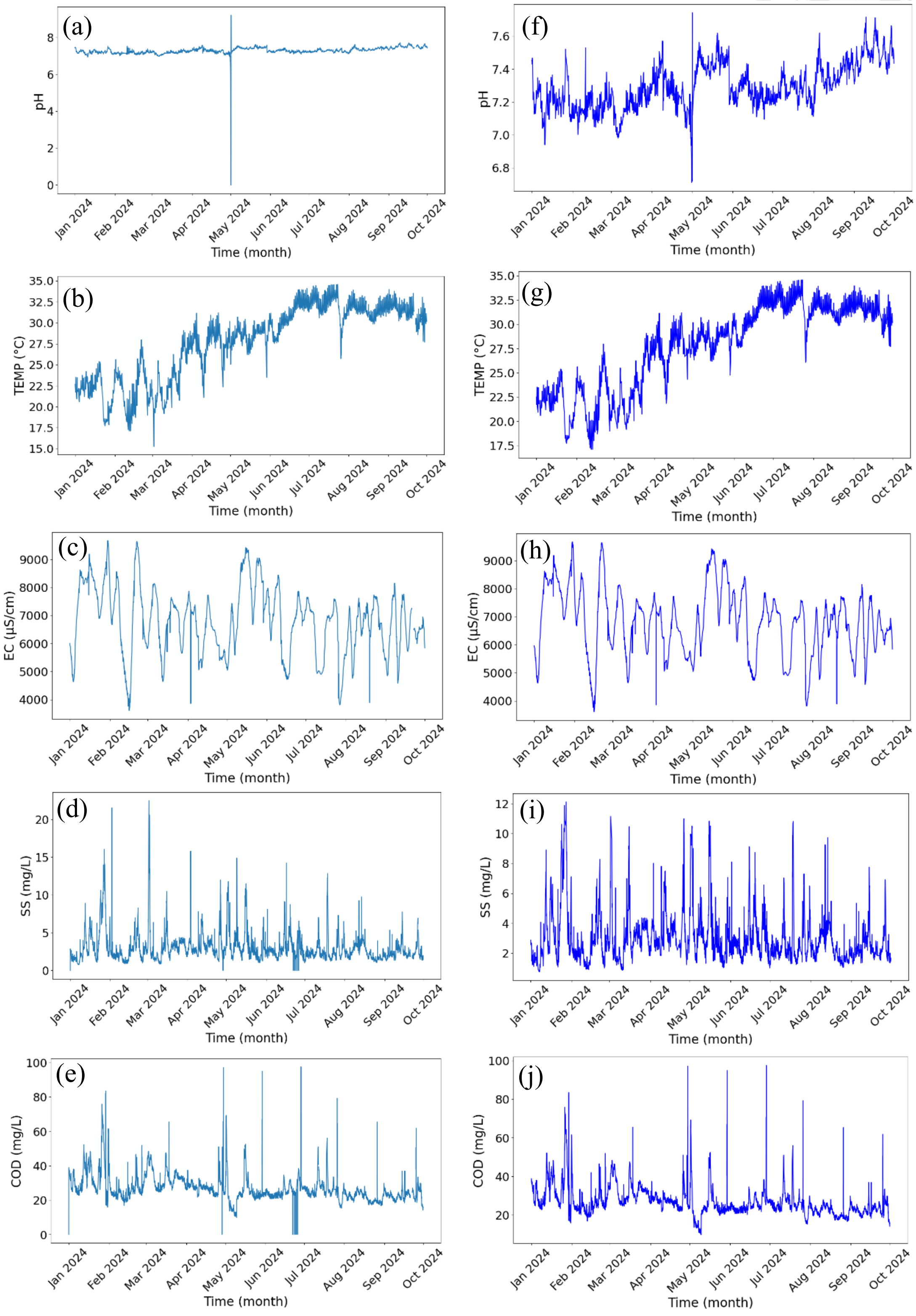


Figure 6. Water quality parameters for the effluent over a 9-month period: (a) pH, (b) TEMP, (c) EC, (d) SS, and (e) COD before data cleaning; (f) pH, (g) TEMP, (h) EC, (i) SS, and (j) COD after data cleaning

doi:10.6342/NTU202503249

Table 3. The maximum, minimum, mean and standard deviations of each water quality after data cleaning

Parameters	Units	Max	Min	Average	SD
Q	M^3/day	7092.0	459.0	3760.2	1426.0
$pH_{.in}$	-	8.0	6.2	6.9	0.3
$TEMP_{.in}$	°C	34.2	16.1	27.7	3.9
$EC_{.in}$	μS/cm	12662.7	576.4	5312.5	2578.7
$SS_{.in}$	mg/L	45450.0	7.3	12757.9	12088.6
$COD_{.in}$	mg/L	974.8	69.7	614.5	276.7
$pH_{.Ox.ditch}$	-	7.5	4.8	6.5	0.8
$TEMP_{.Ox.ditch}$	°C	33.3	17.2	27.4	4.0
$ORP_{.Ox.ditch}$	mV	335.3	-1032.3	25.9	330.5
$DO_{.Ox.ditch}$	mg/L	6.0	0.1	1.4	1.3
$pH_{.out}$	-	7.7	6.7	7.3	0.1
$TEMP_{.out}$	°C	34.6	17.1	27.6	4.4
$EC_{.out}$	μS/cm	9671.8	3610.9	6821.4	1207.6
$SS_{.out}$	mg/L	12.1	0.8	3.0	1.8
$COD_{.out}$	mg/L	97.4	9.8	27.0	7.8

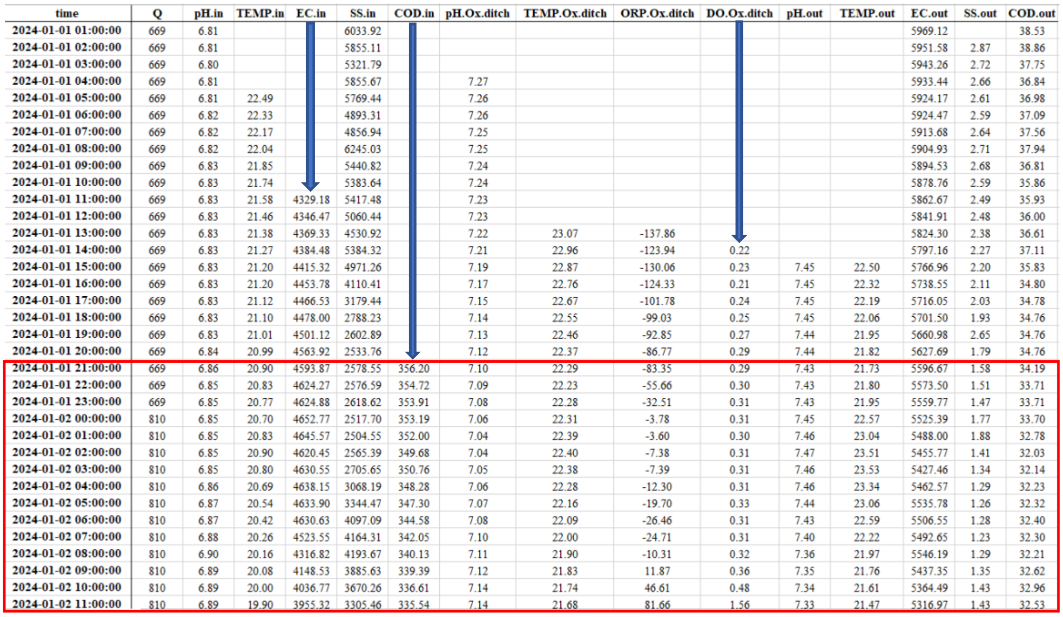
doi:10.6342/NTU202503249

Since all water quality parameter sensors record data simultaneously, a time lag must be introduced to reflect the fact that different batches of water are detected by these sensors. The optimal time lag for each of the 14 water quality parameters relative to COD.out was calculated using the method described in Section 3.3.2. and the results are shown in Table 4. The optimal time lag varies across water quality parameters, likely due to factors such as hydraulic retention times and the reaction kinetics affecting each water quality parameters in different treatment units before the parameter's association with COD.out becomes evident. For example, Influent COD must experience all treatment units with a cumulative time lag of 20 hr to be associated with COD.out. Similarly, ORP and DO in the oxidation ditch reflect the redox status in the ditch that affect microbial activity, which require a sufficient retention time to reveal their impact on COD removal that ultimately affect COD.out. Figure 7 shows an example of the water quality parameters are shifted according to the optimal time lag in Table 4. In the subsequent model training, the shifted data (marked by the red box in Figure 7) are used as the input.

doi:10.6342/NTU202503249

Table 4. Optimal time lag between each parameter and COD.out

Parameters	Best time Lag
pH$_{.in}$	0
TEMP$_{.in}$	4
EC$_{.in}$	10
SS$_{.in}$	0
COD$_{.in}$	20
pH$_{.Ox.ditch}$	3
TEMP$_{.Ox.ditch}$	12
ORP$_{.Ox.ditch}$	12
DO$_{.Ox.ditch}$	13
pH$_{.out}$	14
TEMP$_{.out}$	14
EC$_{.out}$	0
SS$_{.out}$	1



time	Q	pH.in	TEMP.in	EC.in	SS.in	COD.in	pH.Ox.ditch	TEMP.Ox.ditch	ORP.Ox.ditch	DO.Ox.ditch	pH.out	TEMP.out	EC.out	SS.out	COD.out
2024-01-01 01:00:00	669	6.81			6033.92								5969.12		38.53
2024-01-01 02:00:00	669	6.81			5855.11								5951.58	2.87	38.86
2024-01-01 03:00:00	669	6.80			5321.79								5943.26	2.72	37.75
2024-01-01 04:00:00	669	6.81			5855.67		7.27						5933.44	2.66	36.84
2024-01-01 05:00:00	669	6.81	22.49		5769.44		7.26						5924.17	2.61	36.98
2024-01-01 06:00:00	669	6.82	22.33		4893.31		7.26						5924.47	2.59	37.09
2024-01-01 07:00:00	669	6.82	22.17		4856.94		7.25						5913.68	2.64	37.56
2024-01-01 08:00:00	669	6.82	22.04		6245.03		7.25						5904.93	2.71	37.94
2024-01-01 09:00:00	669	6.83	21.85		5440.82		7.24						5894.53	2.68	36.81
2024-01-01 10:00:00	669	6.83	21.74		5383.64		7.24						5878.76	2.59	35.86
2024-01-01 11:00:00	669	6.83	21.58	4329.18	5417.48		7.23						5862.67	2.49	35.93
2024-01-01 12:00:00	669	6.83	21.46	4346.47	5060.44		7.23						5841.91	2.48	36.00
2024-01-01 13:00:00	669	6.83	21.38	4369.33	4530.92		7.22	23.07	-137.86				5824.30	2.38	36.61
2024-01-01 14:00:00	669	6.83	21.27	4384.48	5384.32		7.21	22.96	-123.94	0.22			5797.16	2.27	37.11
2024-01-01 15:00:00	669	6.83	21.20	4415.32	4971.26		7.19	22.87	-130.06	0.23	7.45	22.50	5766.96	2.20	35.83
2024-01-01 16:00:00	669	6.83	21.20	4453.78	4110.41		7.17	22.76	-124.33	0.21	7.45	22.32	5738.55	2.11	34.80
2024-01-01 17:00:00	669	6.83	21.12	4466.53	3179.44		7.15	22.67	-101.78	0.24	7.45	22.19	5716.05	2.03	34.78
2024-01-01 18:00:00	669	6.83	21.10	4478.00	2788.23		7.14	22.55	-99.03	0.25	7.45	22.06	5701.50	1.93	34.76
2024-01-01 19:00:00	669	6.83	21.01	4501.12	2602.89		7.13	22.46	-92.85	0.27	7.44	21.95	5660.98	2.65	34.76
2024-01-01 20:00:00	669	6.84	20.99	4563.92	2533.76		7.12	22.37	-86.77	0.29	7.44	21.82	5627.69	1.79	34.76
2024-01-01 21:00:00	669	6.86	20.90	4593.87	2578.55	356.20	7.10	22.29	-83.35	0.29	7.43	21.73	5596.67	1.58	34.19
2024-01-01 22:00:00	669	6.85	20.83	4624.27	2576.59	354.72	7.09	22.23	-55.66	0.30	7.43	21.80	5573.50	1.51	33.71
2024-01-01 23:00:00	669	6.85	20.77	4624.88	2618.62	353.91	7.08	22.28	-32.51	0.31	7.43	21.95	5559.77	1.47	33.71
2024-01-02 00:00:00	810	6.85	20.70	4652.77	2517.70	353.19	7.06	22.31	-3.78	0.31	7.45	22.57	5525.39	1.77	33.70
2024-01-02 01:00:00	810	6.85	20.83	4645.57	2504.55	352.00	7.04	22.39	-3.60	0.30	7.46	23.04	5488.00	1.88	32.78
2024-01-02 02:00:00	810	6.85	20.90	4620.45	2565.39	349.68	7.04	22.40	-7.38	0.31	7.47	23.51	5455.77	1.41	32.03
2024-01-02 03:00:00	810	6.85	20.80	4630.55	2705.65	350.76	7.05	22.38	-7.39	0.31	7.46	23.53	5427.46	1.34	32.14
2024-01-02 04:00:00	810	6.86	20.69	4638.15	3068.19	348.28	7.06	22.28	-12.30	0.31	7.46	23.34	5462.57	1.29	32.23
2024-01-02 05:00:00	810	6.87	20.54	4633.90	3344.47	347.30	7.07	22.16	-19.70	0.33	7.44	23.06	5535.78	1.26	32.32
2024-01-02 06:00:00	810	6.87	20.42	4630.63	4097.09	344.58	7.08	22.09	-26.46	0.31	7.43	22.59	5506.55	1.28	32.40
2024-01-02 07:00:00	810	6.88	20.26	4523.55	4164.31	342.05	7.10	22.00	-24.71	0.31	7.40	22.22	5492.65	1.23	32.30
2024-01-02 08:00:00	810	6.90	20.16	4316.82	4193.67	340.13	7.11	21.90	-10.31	0.32	7.36	21.97	5546.19	1.29	32.21
2024-01-02 09:00:00	810	6.89	20.08	4148.53	3885.63	339.39	7.12	21.83	11.87	0.36	7.35	21.76	5437.35	1.35	32.62
2024-01-02 10:00:00	810	6.89	20.00	4036.77	3670.26	336.61	7.14	21.74	46.61	0.48	7.34	21.61	5364.49	1.43	32.96
2024-01-02 11:00:00	810	6.89	19.90	3955.32	3305.46	335.54	7.14	21.68	81.66	1.56	7.33	21.47	5316.97	1.43	32.53

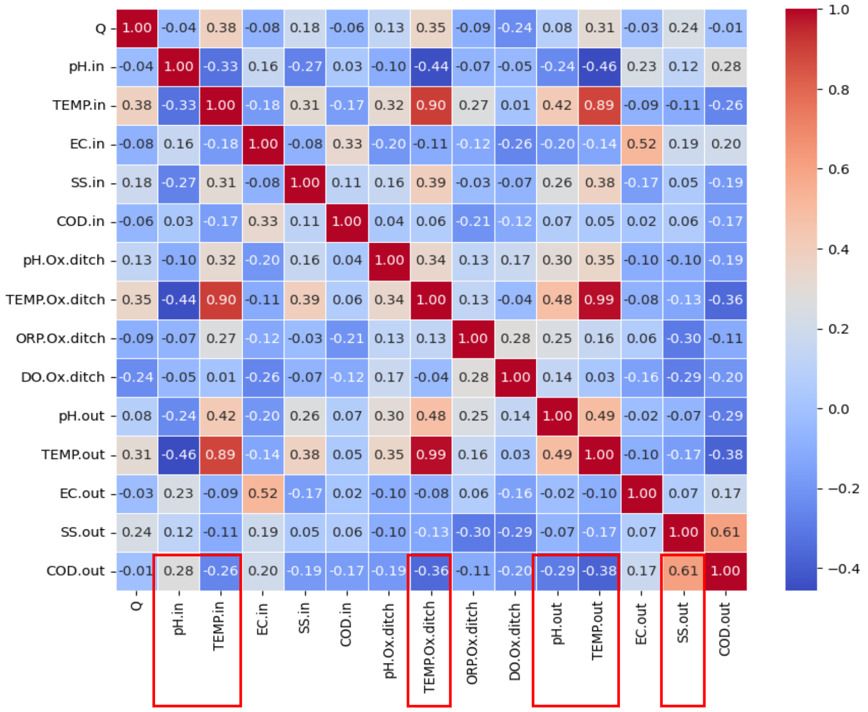
Figure 7. Data shifted by optimal time lag

doi:10.6342/NTU202503249

4.2 Model Development and Performance Evaluation

4.2.1 Feature Selection

To investigate the relationship between each water quality parameter (feature) and COD.out, the Pearson correlation coefficient between each feature and COD.out was calculated (see section 3.4) and plotted as a heatmap as shown in Figure 8. The red color indicates positive correlation and the blue color indicates negative correlation. Among the 14 features, only those with a higher correlation with COD.out were selected for the following model development as irrelevant features could increase computational load and interfere with the model performance. Based on the heatmap, pH.in, Temp.in, Temp.Ox.ditch, pH.out, Temp.out, and SS.out were selected.



	Q	pH.in	TEMP.in	EC.in	SS.in	COD.in	pH.Ox.ditch	TEMP.Ox.ditch	ORP.Ox.ditch	DO.Ox.ditch	pH.out	TEMP.out	EC.out	SS.out	COD.out
Q	1.00	-0.04	0.38	-0.08	0.18	-0.06	0.13	0.35	-0.09	-0.24	0.08	0.31	-0.03	0.24	-0.01
pH.in	-0.04	1.00	-0.33	0.16	-0.27	0.03	-0.10	-0.44	-0.07	-0.05	-0.24	-0.46	0.23	0.12	0.28
TEMP.in	0.38	-0.33	1.00	-0.18	0.31	-0.17	0.32	0.90	0.27	0.01	0.42	0.89	-0.09	-0.11	-0.26
EC.in	-0.08	0.16	-0.18	1.00	-0.08	0.33	-0.20	-0.11	-0.12	-0.26	-0.20	-0.14	0.52	0.19	0.20
SS.in	0.18	-0.27	0.31	-0.08	1.00	0.11	0.16	0.39	-0.03	-0.07	0.26	0.38	-0.17	0.05	-0.19
COD.in	-0.06	0.03	-0.17	0.33	0.11	1.00	0.04	0.06	-0.21	-0.12	0.07	0.05	0.02	0.06	-0.17
pH.Ox.ditch	0.13	-0.10	0.32	-0.20	0.16	0.04	1.00	0.34	0.13	0.17	0.30	0.35	-0.10	-0.10	-0.19
TEMP.Ox.ditch	0.35	-0.44	0.90	-0.11	0.39	0.06	0.34	1.00	0.13	-0.04	0.48	0.99	-0.08	-0.13	-0.36
ORP.Ox.ditch	-0.09	-0.07	0.27	-0.12	-0.03	-0.21	0.13	0.13	1.00	0.28	0.25	0.16	0.06	-0.30	-0.11
DO.Ox.ditch	-0.24	-0.05	0.01	-0.26	-0.07	-0.12	0.17	-0.04	0.28	1.00	0.14	0.03	-0.16	-0.29	-0.20
pH.out	0.08	-0.24	0.42	-0.20	0.26	0.07	0.30	0.48	0.25	0.14	1.00	0.49	-0.02	-0.07	-0.29
TEMP.out	0.31	-0.46	0.89	-0.14	0.38	0.05	0.35	0.99	0.16	0.03	0.49	1.00	-0.10	-0.17	-0.38
EC.out	-0.03	0.23	-0.09	0.52	-0.17	0.02	-0.10	-0.08	0.06	-0.16	-0.02	-0.10	1.00	0.07	0.17
SS.out	0.24	0.12	-0.11	0.19	0.05	0.06	-0.10	-0.13	-0.30	-0.29	-0.07	-0.17	0.07	1.00	0.61
COD.out	-0.01	0.28	-0.26	0.20	-0.19	-0.17	-0.19	-0.36	-0.11	-0.20	-0.29	-0.38	0.17	0.61	1.00

Figure 8. Coefficient correlation heatmap

4.2.2 Model Construction and Hyperparameter Settings

After feature selection, the data were divided into a 70% training set (4,501 data points) and a 30% testing set (1,929 data points). To prevent overfitting that will result in noise memory in the training data or underfitting that will hinder the model to capture the underlying trends, the K-fold cross validation (K=5) as shown in Figure 9 was employed. In the 5-fold cross validation, the 4,501 training data points are divided equally into 5 subsets. In each fold, 4 subsets (3,601 data points) are used for model training, and the remaining subset (900 data points) serves as the validation set. This process is repeated 5 times to reduce the risk that a single partition excessively influences the evaluation metrics. After completing the 5 folds, the evaluation metrics from each fold are averaged for each hyperparameter combination. These averaged metrics are then compared across all configurations to identify the best hyperparameters.

Different models require different types of hyperparameters which are critical in the model training and ultimately the model performance. The best hyperparameters used for training the RF, GBM, and XGB models are summarized in Table 5-7, respectively. Each model was retained using the entire training set (4,501 samples) according to the best hyperparameters and their performances were evaluated using the test set for unseen data (1,929 data points).

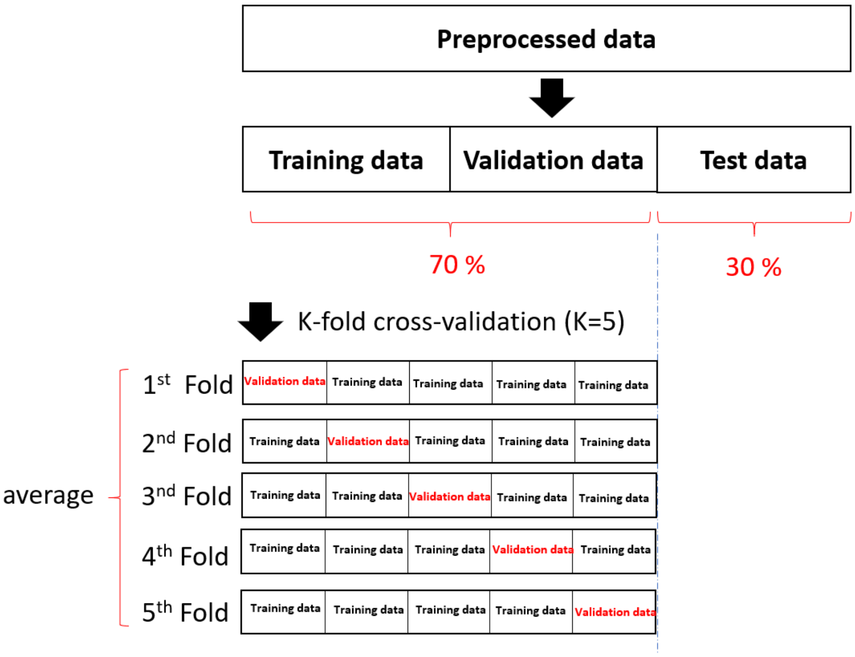


Figure 9. K-fold cross validation (K=5)

doi:10.6342/NTU202503249

Table 5. Hyperparameter settings for RF

Hyperparameters	Meaning of Hyperparameters	Value Settings	Best value
n_estimators	Specifies how many decision trees the model will build. • Too low：underfitting • Too high：overfitting	50, 100, 200	200
max_depth	Defines how deep each decision tree can grow. • Too low：underfitting • Too high：overfitting	None (No Limit), 10, 20, 30	20
min_samples_split	Determines how many samples must exist in a node before the algorithm attempts to split it. • Too low：overfitting • Too high：underfitting	2, 5, 10	2
min_samples_leaf	Indicates how many samples each leaf node must contain. • Too low：overfitting • Too high：underfitting	1, 2, 4	1
max_features	Specifies how many features to consider when searching for the best split at each node. • Too low：underfitting • Too high：overfitting	Sqrt($\sqrt{\text{Number of features}}$), log2($log_{2Number\ of\ features}$)	3

doi:10.6342/NTU202503249

Table 6. Hyperparameter settings for GBM

Hyperparameters	Meaning of Hyperparameters	Value Settings	Best value
n_estimators	Specifies how many weak learners (decision trees) the model uses to iteratively reduce error. • Too low：underfitting • Too high：overfitting	50, 100, 200	200
max_depth	Defines how deep each decision tree can grow. • Too low：underfitting • Too high：overfitting	3, 5, 7	7
learning_rate	Scales each weak learner's contribution at every iteration. • Too low：underfitting • Too high：overfitting	0.01, 0.1, 0.2	0.2
subsample	Determines what fraction of training samples to draw at each iteration. • Too low：underfitting • Too high：overfitting	0.8, 1.0	1.0

doi:10.6342/NTU202503249

Table 7. Hyperparameter settings for XGB

Hyperparameters	Meaning of Hyperparameters	Value Settings	Best value
n_estimators	Specifies how many weak learners (decision trees) the model uses to iteratively reduce error. ・ Too low：underfitting ・ Too high：overfitting	50, 100, 200	200
max_depth	Defines how deep each decision tree can grow. ・ Too low：underfitting ・ Too high：overfitting	3, 5, 7	7
learning_rate	Scales each weak learner's contribution at every iteration. ・ Too low：underfitting ・ Too high：overfitting	0.01, 0.1, 0.2	0.2
subsample	Determines what fraction of training samples to draw at each iteration. ・ Too low：underfitting ・ Too high：overfitting	0.8, 1.0	0.8
colsample_bytree	Specifies what fraction of features each tree randomly uses when building. ・ Too low：underfitting ・ Too high：overfitting	0.6, 0.8, 1.0	1.0
gamma	Specifies the minimum loss reduction required to split a node. * A loss function is the error between predicted and actual values. ・ Too low：overfitting ・ Too high：underfitting	0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6	0.1
alpha	Controls how strongly the model applies an L_1 penalty to leaf outputs. * It sets the outputs of noisy leaves to zero. ・ Too low：overfitting ・ Too high：underfitting	0.05, 0.1, 1, 2, 3	3
Lambda	Controls how strongly the model applies an L_2 penalty to leaf outputs. *It shrinks all leaf outputs without making them zero. ・ Too low：overfitting ・ Too high：underfitting	0.05, 0.1, 1, 2, 3	3

doi:10.6342/NTU202503249

4.2.3 Model Performance Evaluation

Figure 10 shows the scatter plots using training and test datasets for RF, GBM, and XGB and the obtained performance evaluation metrics including R^2, RMSE, MAE, and MAPE are shown in Table 8. For the training dataset, RF, GBM, and XGB all perform very well, with the R^2 of 0.98, 1.00, and 0.99, respectively; while for the test dataset, the R^2 are 0.83, 0.79, and 0.82 respectively. R^2 reflects how closely predicted values match actual values, so higher values indicate better fit; whereas RMSE, MAE, and MAPE are used to evaluate the deviation between the model predictions and the true values, so lower values indicate more accurate performance. In addition to the scatter plots, Figure 11 shows the trend charts comparing predicted and actual values for RF, GBM, and XGB on the test datasets. These trend charts help visualize how well each model captures the overall patterns. Considering both the evaluation metrics and the trend charts, RF and XGB slightly outperform GBM on the test dataset, although the differences are not large.

doi:10.6342/NTU202503249

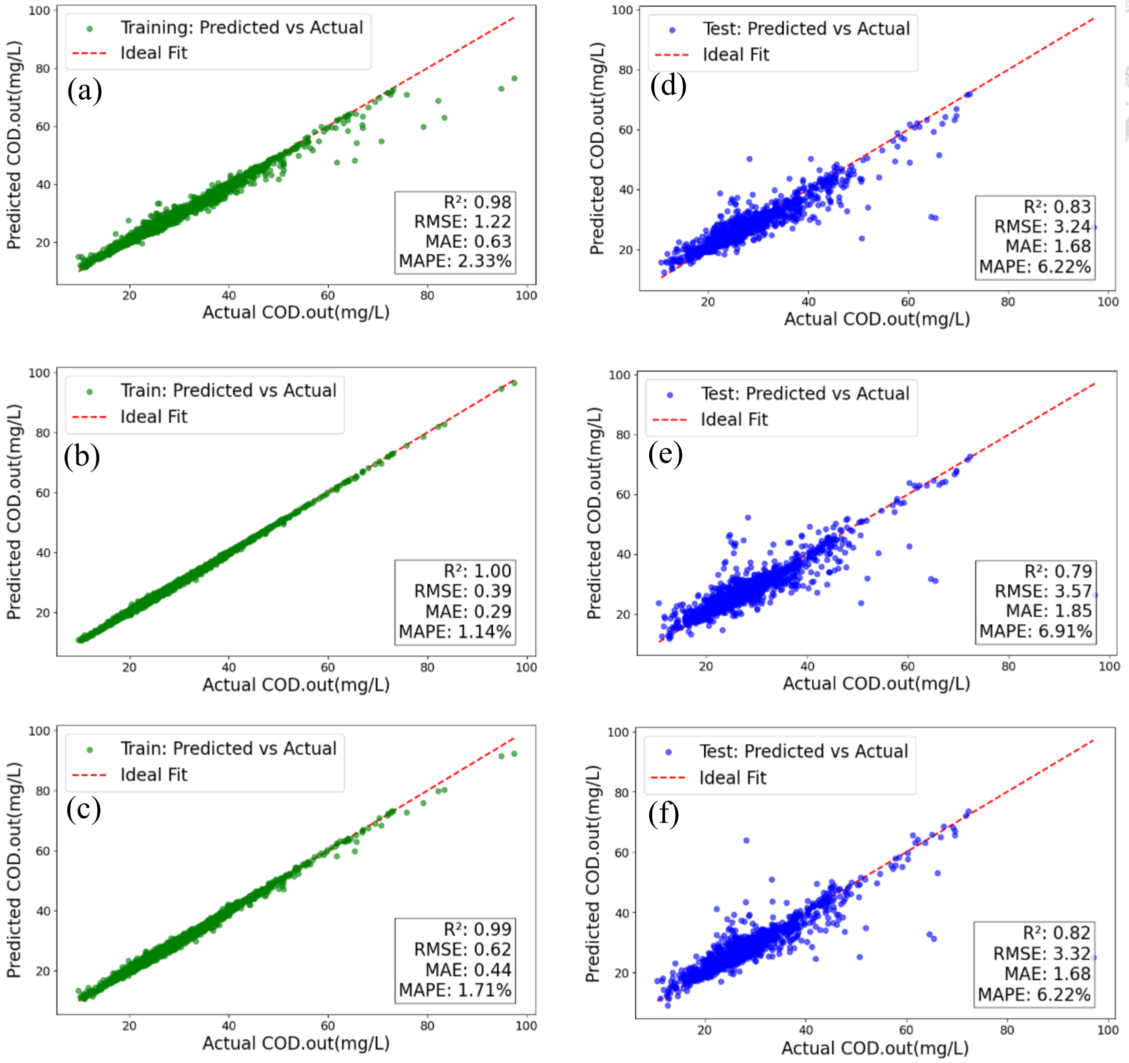


Figure 10. Scatter plots for the training dataset: (a) RF, (b) GBM, (c) XGB; and Scatter plots for the test dataset: (d) RF, (e) GBM, (f) XGB

Table 8. Performance evaluation of RF, GBM, and XGB on the training/test dataset

Model	**Training Dataset**				**Test Dataset**				**Training Time (sec)**
	R^2	RMSE	MAE	MAPE (%)	R^2	RMSE	MAE	MAPE (%)	
RF	0.98	1.22	0.63	2.33	0.83	3.24	1.68	6.22	128
GBM	1.00	0.39	0.29	1.14	0.79	3.57	1.85	6.91	48
XGB	0.99	0.62	0.44	1.71	0.82	3.32	1.68	6.22	2645

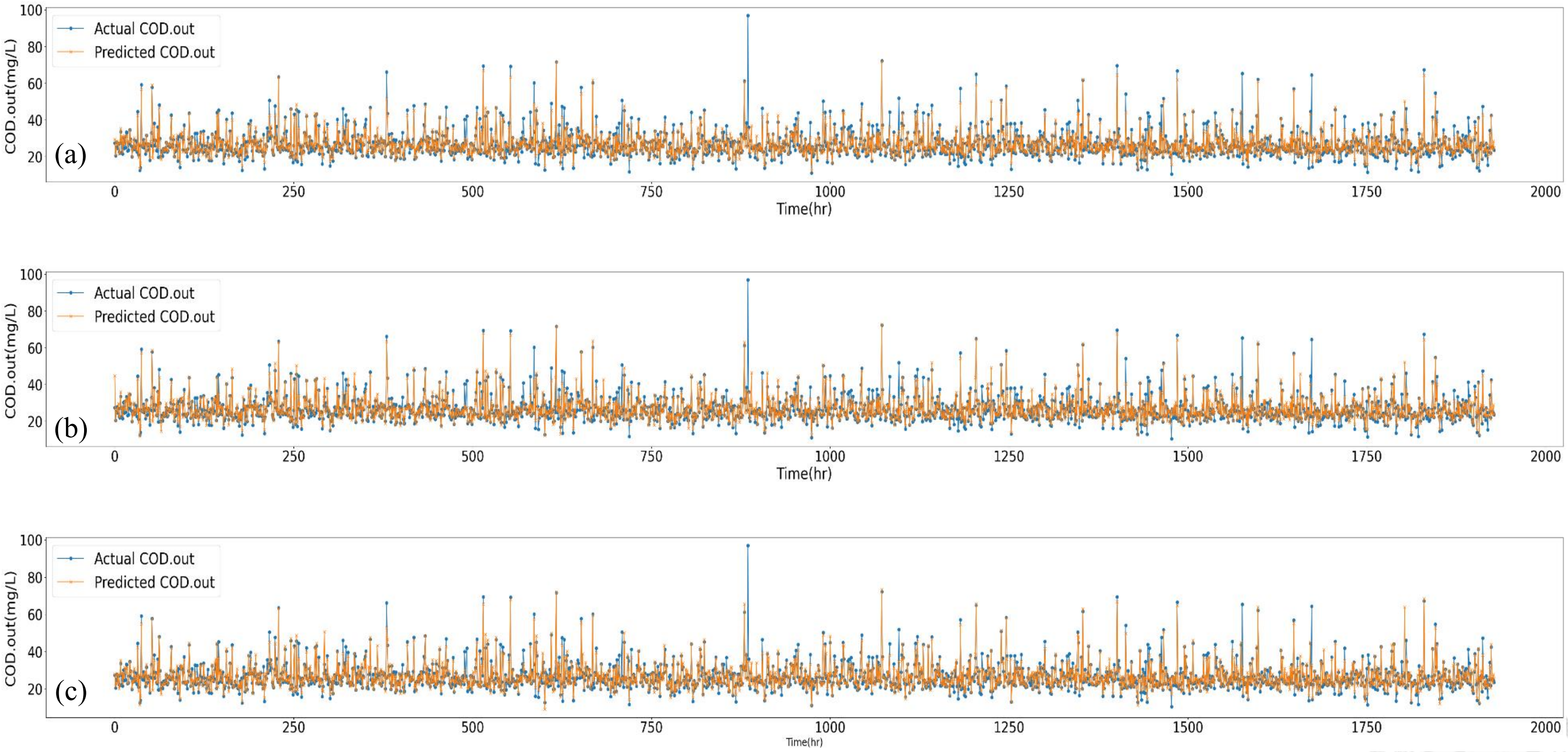


Figure 11. Trend charts for the test dataset: (a) RF, (b) GBM, (c) XGB

doi:10.6342/NTU202503249

From the perspective of computational efficiency, XGB requires tuning more hyperparameters (Table 7), which often leads to a longer training time. Therefore, XGB was not considered further. In contrast, RF builds independent decision trees simultaneously and averages their predictions for the final output. Since each tree works separately, errors in one tree do not influence other trees, making the model more robust to overfitting. RF also needs fewer hyperparameters and trains faster. Considering these factors, RF was considered as the best model in this study.

4.3 Feature Contribution Analysis for COD.out Prediction Using SHAP

SHAP analysis (Figure 12) was applied to quantify each feature's contribution to RF model predictions. Figure 12(a) shows the distribution of SHAP values for each feature across all instances, in which each instance represents a complete set of feature values. The color from blue to red represents the feature value from low to high. When the SHAP value is positive, the feature increases the model's predicted value. For example, the red dots for SS.out lie at SHAP > 0, indicating that higher values of SS.out increase the model's predicted values for COD.out, while the blue dots lie at SHAP < 0, indicating that lower SS.out values decrease the predicted COD.out values.

Moreover, to compare each feature's overall contribution to the model's predictions, the absolute SHAP values for all instances for each feature were averaged and plotted as a bar chart as shown in Figure 12(b). The bar chart shows that the top three most influential features for predicting COD were SS.out, Temp.Ox.ditch, and Temp.out.

doi:10.6342/NTU202503249

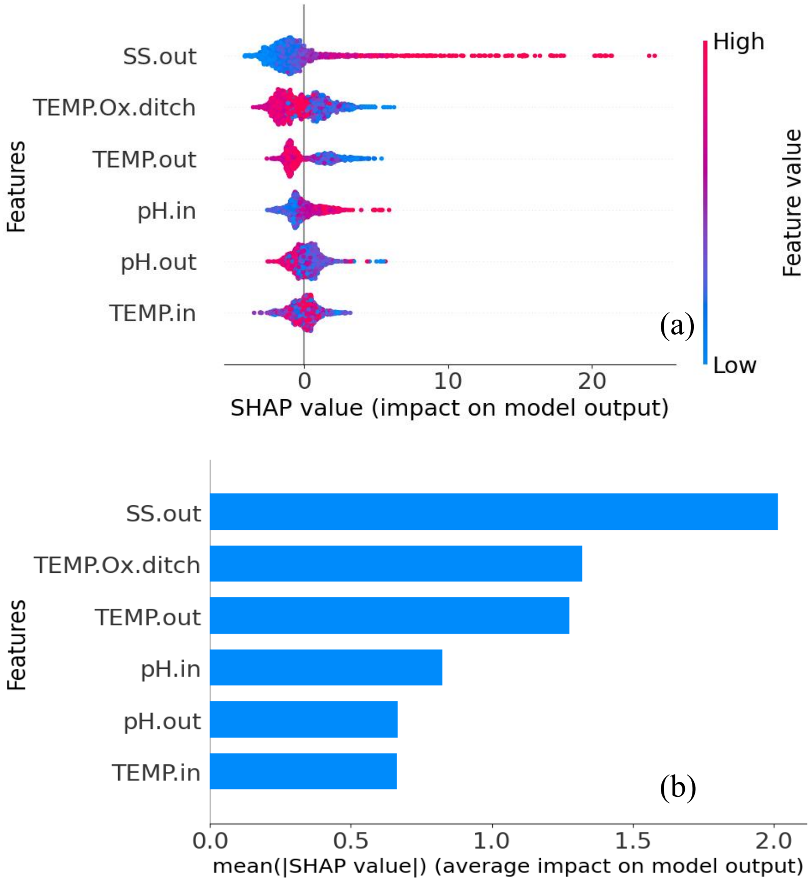


Figure 12.(a) Distribution of SHAP values for each feature in predicting COD.out, (b) Feature contribution on COD.out prediction (measured by mean absolute SHAP values)

Based on the feature contribution ranking shown in Figure 12(b), Figure 13 presents SHAP interaction plots of SS.out vs.Temp.Ox.ditch and SS.out vs. Temp.out. The x-axis shows the SS.out value, the left y-axis displays the SS.out's SHAP value, and the right y-axis shows the corresponding Temp.Ox.ditch or Temp.out values. As shown in Figure 13(a) and 13(b), it is observed that when SS.out is below 2.5 mg/L, the corresponding SHAP values are negative, indicating that the model predicts a lower COD.out. Once SS.out exceeds 2.5 mg/L, SHAP values become positive and increase almost linearly, showing that a higher SS.out leads to a higher predicted COD.out. In particular, when SS.out is between 11 and 12.5 mg/L and both water temperatures (Temp.Ox.ditch and

Temp.out) are between 20 and 22 °C, the SHAP values are the highest, indicating that under these conditions, COD.out rises significantly and requires attention.

SS consists of both organic and inorganic particles. When organic particles dominate, SS and COD are positively correlated, while when inorganic particles dominate, they are only weakly correlated. Figure 13(a) and 13(b) show that a higher SS.out leads to a higher COD.out, illustrating that organic particles resulting from microbial biomass dominates. To reduce COD.out, SS.out should be controlled below 2.5 mg/L to minimize the organic loading to COD.out.

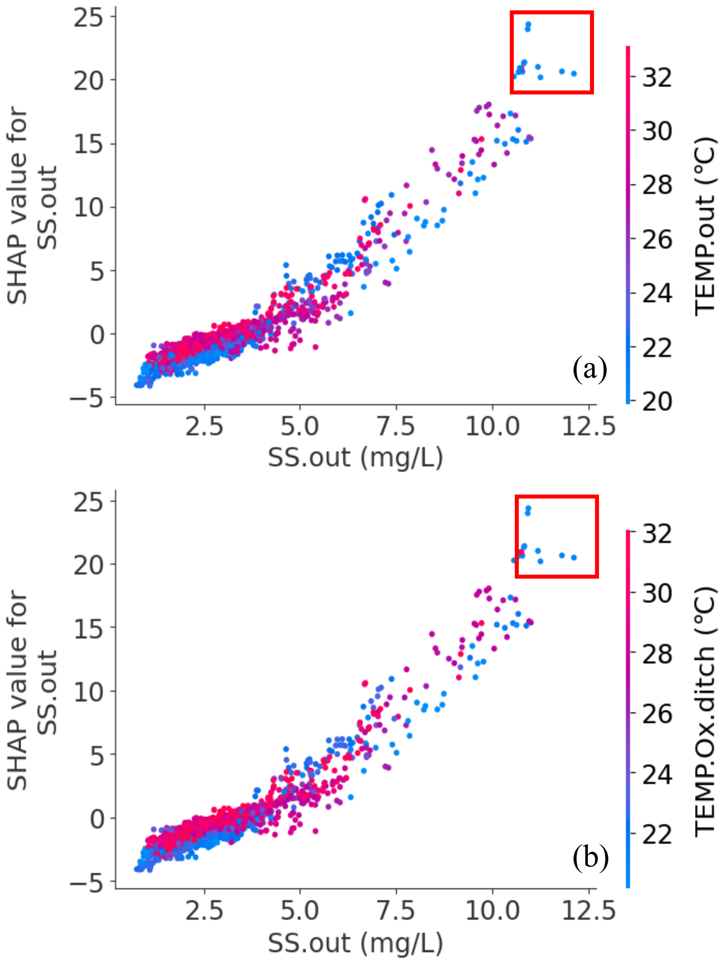


Figure 13.(a) Interaction between SS.out and TEMP.out, (b) Interaction between SS.out and Temp.Ox.ditch

doi:10.6342/NTU202503249

Figure 14 presents the SHAP interaction plots of Temp.Ox.ditch vs. SS.out and Temp.Ox.ditch vs. Temp.out. The x-axis shows the Temp.Ox.ditch value, the left y-axis displays the Temp.Ox.ditch's SHAP value, and the right y-axis shows the corresponding SS.out or Temp.out. As shown in Figure 14(a) and 14(b), when Temp.Ox.ditch is below approximately 27 °C, the SHAP value is positive, indicating that a lower temperature in the oxidation ditch is associated with an increased COD.out. When Temp.Ox.ditch ranges from 27-32 ℃, the SHAP value becomes negative, showing that Temp.Ox.ditch in this range contributes to a lower COD.out. When the Temp.Ox.ditch exceeds 32 ℃, the SHAP value rises again to positive, implying that an excessively high temperature in Temp.Ox.ditch also leads to a higher COD.out. Overall, the results in Figure 14(b) can be divided into three categories: blue for Temp.Ox.ditch = 18-24 ℃, purple for Temp.Ox.ditch = 24-28 ℃, and red for Temp.Ox.ditch > 28 ℃. When Temp.Ox.ditch is between 27 ℃ and 32 ℃, the corresponding Temp.out also lies in the high temperature region above 27 ℃, indicated by purple and red colors in Figure 14(b). In this range of Temp.Ox.ditch, the SHAP values are negative, indicating that the model predicts a lower COD.out under these conditions. Thus, this range represents an optimal temperature in the oxidation ditch.

doi:10.6342/NTU202503249

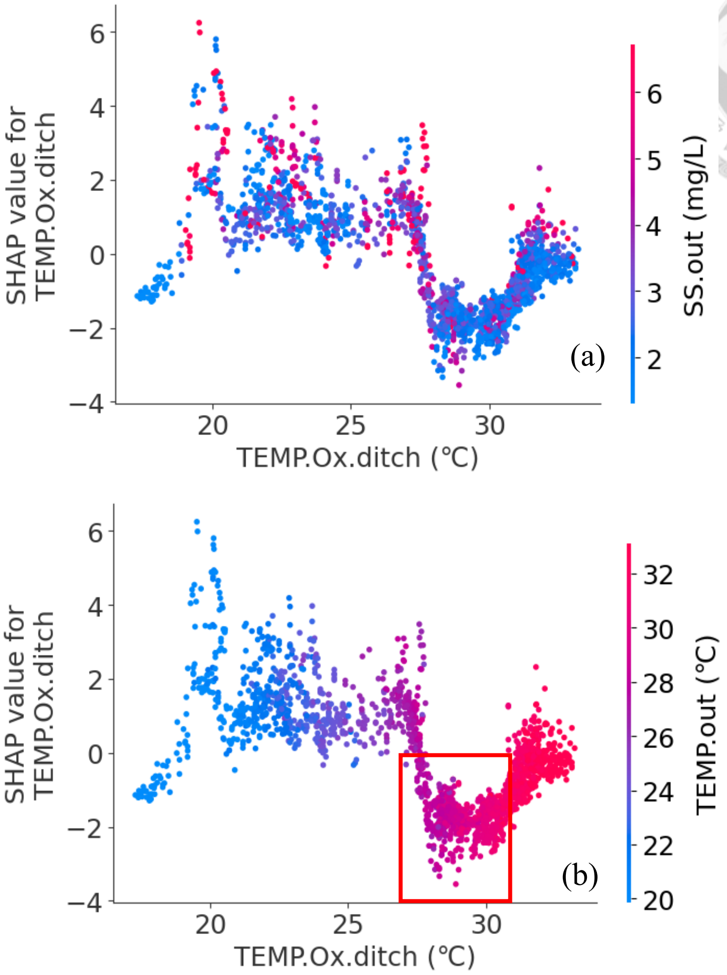


Figure 14.(a) Interaction between Temp.Ox.ditch and SS.out, (b) Interaction between Temp.Ox.ditch and TEMP.out

Figure 15 presents the SHAP interaction plots of Temp.out vs. SS.out and Temp.out vs. Temp.Ox.ditch. The x-axis shows the Temp.out value, the left y-axis displays the Temp.out's SHAP value, and the right y-axis shows the corresponding SS.out or Temp.Ox.ditch. As shown in Figure 15(a) and 15(b), when Temp.out is below approximately 27 °C, the SHAP value is positive, indicating that a lower Temp.out is associated with an increased COD.out. Once Temp.out exceeds 27 °C, the SHAP value becomes negative, meaning that a higher temperature leads to a decreased COD.out.

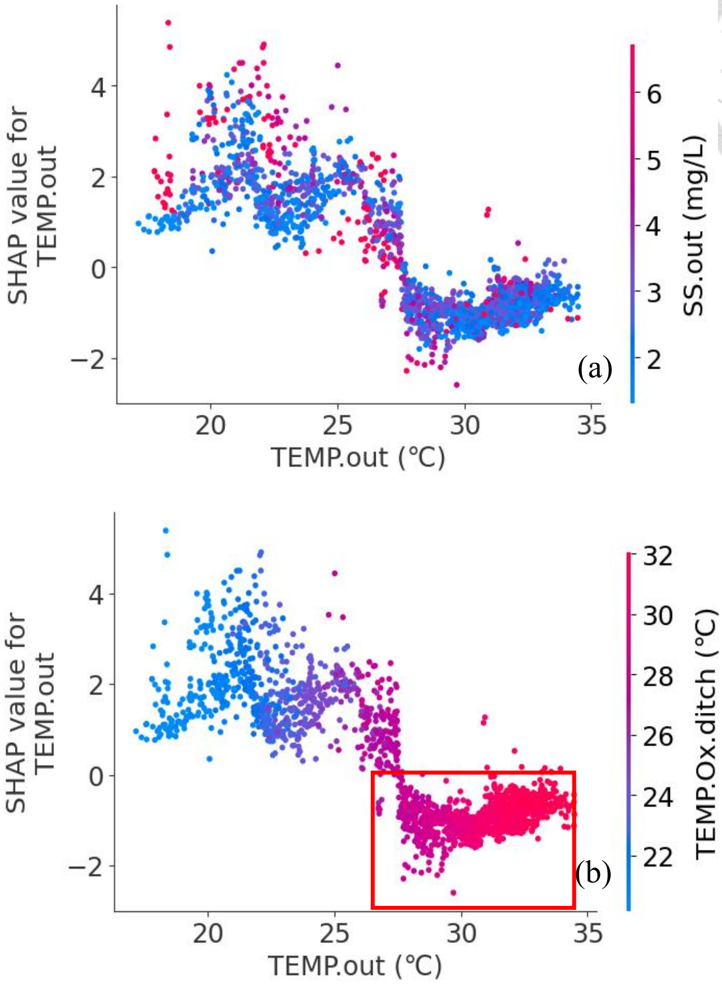


Figure 15. (a) Interaction between TEMP.out and SS.out, (b) Interaction between TEMP.out and Temp.Ox.ditch

Based on the above SHAP analysis, it is recommended the following to achieve a lower effluent COD: 1. maintain a relatively high temperature in the oxidation ditch and effluent to boost the microbial activity to degrade organics and 2. keep SS in the effluent below 2.5 mg/L by enhancing the performance of clarifies after the oxidation ditch.

doi:10.6342/NTU202503249

Chapter 5 Conclusions and Recommendations

5.1 Conclusions

In this study, ML models were used to predict the effluent COD in an industrial WWTP. For the ML model development, a data screening procedure was employed to ensure the integrity of historical sensing data and a “time lag” concept was incorporated to reflect the water retention in treatment units in the data processing workflow. The performance of different ML models for predicting effluent COD was then evaluated. Finally, the impacts of each input data feature on the ML predictions were quantified and explained. The conclusions are summarized as follows:

1. Among the 6,430 data points collected from the industrial WWTP, missing data, invalid data and outliers were identified and removed. To preserve the continuity required for subsequent model development, these removed points were imputed using the values determined from linear interpolation.
2. Optimal time lags for 14 water quality parameters relative to effluent COD were determined by Pearson product-moment and Spearman’s rank correlations. It was found that the optimal time lag varied across parameters. Accounting for time delays is essential for reflecting the water retention in treatment units.
3. Three ML models, including RF, GBM, and XGB, were tested. For the test dataset, RF and XGB were found to slightly outperform GBM. However, XGB was not considered due to its longer training time. Moreover, RF is less prone to overfitting and requires less training time. Consequently, RF was selected as the optimal model, achieving a MAPE of 6.22%, indicating close alignment between predicted and real values.
4. SHAP analysis identified effluent SS, temperature in the oxidation ditch and effluent

doi:10.6342/NTU202503249

temperature as the most influential parameters for RF predictions of effluent COD. It is recommended to maintain high temperatures in both the oxidation ditch and effluent and to keep effluent SS below 2.5 mg/L to achieve a low effluent COD in the WWTP.

5.2 Recommendations

Sensor readings for influent SS (up to 40,000 mg/L) and influent COD (up to 900 mg/L) were not removed from ML development since they meet the data screening criteria. However, these values could not reflect true water quality and indictive for probable sensor failures. Although these variables were not important features for the effluent COD prediction using RF model, they highlight the importance of acquiring accurate data for ML models. Below are some recommendations for future study.

1. Develop sensors with automatic cleaning and self-calibration to prevent fouling and thus improve data quality.
2. Currently, only water quality monitoring data from the oxidation ditch influent, oxidation ditch, and the effluent are available. However, parameters for the influent, grit chamber, and secondary sedimentation tank are unavailable, and operational parameters like aeration rate and sludge retention time have not been recorded. It is recommended to install sensors at each treatment unit and record operational parameters to gain a comprehensive understanding of the WWTP’s conditions, thereby enhancing the model’s ability to predict effluent COD.
3. There are no instances of effluent COD violations in the historical dataset. Without including any exceedance data, the model can not provide early warnings. It is recommended to continuously collect data during any exceedance events at the WWTP and incorporate these records into the dataset to strengthen the model’s

doi:10.6342/NTU202503249

capability to detect abnormal conditions.

4. The RF model is currently being used for effluent COD prediction and achieves an R^2 of 0.83. This indicates fairly high accuracy, yet there remains an opportunity for improvement. It is recommended to simulate the performance of deep learning models not yet tested, such as LSTM and RNN, and carry out comparative evaluations with the RF model to determine the optimal ML model for this WWTP.

doi:10.6342/NTU202503249

Reference

Alpaydin, E. (2006). *Introduction to machine learning*. MIT Press.

Bagherzadeh, F., Mehrani, M.-J., Basirifard, M., & Roostaei, J. (2021). Comparative study on total nitrogen prediction in wastewater treatment plant and effect of various feature selection methods on machine learning algorithms performance. *Journal of Water Process Engineering, 41*, 102033.

Bo-Qi, L., Ding-Jie, Z., Yang, Z., & Long-Yu, S. (2025). Comparative analysis of supervised learning models for effluent quality prediction in wastewater treatment plants. *PLoS One, 20*(6), e0325234.

Cechinel, M. A. P., Neves, J., Fuck, J. V. R., de Andrade, R. C., Spogis, N., Riella, H. G., Padoin, N., & Soares, C. (2024). Enhancing wastewater treatment efficiency through machine learning-driven effluent quality prediction: A plant-level analysis. *Journal of Water Process Engineering, 58*, 104758.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining* (pp. 785-794).

Chollet, F. (2017). *Deep learning with Python*. Simon and Schuster.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 1189-1232.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning with applications in R*. Springer.

Lakshmanaprabu, S. K., Shankar, K., Ilayaraja, M., Nasir, A. W., Vijayakumar, V., & Chilamkurti, N. (2019). Random forest for big data classification in the internet of things using optimal features. *International Journal of Machine Learning and Cybernetics, 10*(10), 2609-2618.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems, 30*.

Müller, A. C., & Guido, S. (2016). *Introduction to machine learning with Python: A guide for data scientists*. O'Reilly Media, Inc.

Mahanna, H., El-Rashidy, N., Kaloop, M. R., El-Sapakh, S., Alluqmani, A., & Hassan, R. (2024). Prediction of wastewater treatment plant performance through machine learning techniques. *Desalination and Water Treatment, 319*, 100524.

Manav-Demir, N., Gelgor, H. B., Oz, E., Ilhan, F., Ulucan-Altuntas, K., Tiwary, A., & Debik, E. (2024). Effluent parameters prediction of a biological nutrient removal (BNR) process using different machine learning methods: A case study. *Journal of Environmental Management, 351*, 119899.

Ministry of Environment. (2024). Effluent water quality parameters and limits for other industrial park dedicated sewers.

Mohammed, M., Khan, M. B., & Bashier, E. B. M. (2016). *Machine learning: Algorithms and applications*. Crc Press.

Moss, B. (2008). Water pollution by agriculture. *Philosophical Transactions of the Royal Society B: Biological Sciences, 363*(1491), 659-666.

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT Press.

Nasir, F. B., & Li, J. (2024). Comparative analysis of machine learning models and explainable artificial intelligence for predicting wastewater treatment plant variables. *Advances in Environmental and Engineering Research, 5(4)*, 1-23.

Qambar, A. S., & Al Khalidy, M. M. (2022). Optimizing dissolved oxygen requirement and energy consumption in wastewater treatment plant aeration tanks using machine learning. *Journal of Water Process Engineering, 50*, 103237.

Safder, U., Kim, J., Pak, G., Rhee, G., & You, K. (2022). Investigating machine learning applications for effective real-time water quality parameter monitoring in full-scale wastewater treatment plants. *Water, 14*(19), 3147.

Stehlík, M., Ibacache-Quiroga, C., Dinamarca, M. A., González-Pizarro, K., Valdivia-Carrera, C. A., Gonzales-Gustavson, E., Ho-Palma, A. C., & Barraza-Morales, B. (2023). On asymmetric relations and robustified cross-correlation approach to surveillance based on detection of SARS-CoV-2 in wastewater in Chile and Peru. *Chemometrics and Intelligent Laboratory Systems, 242*, 104987.

Szeląg, B., Barbusiński, K., Studziński, J., & Bartkiewicz, L. (2017). Prediction of wastewater quality indicators at the inflow to the wastewater treatment plant using data mining methods. In *E3S Web of Conferences* (pp. 00174).

Tchobanoglous, G., Stensel, H. D., Tsuchihashi, R., & Burton, F. (2014). *Wastewater engineering treatment and resource recovery* (5th ed.). McGraw-Hill College.

Toivonen, E., & Räsänen, E. (2024). Time-series analysis approach to the characteristics and correlations of wastewater variables measured in paper industry. *Journal of Water Process Engineering, 61*, 105231.

Von Sperling, M. (2007). *Wastewater characteristics, treatment and disposal*. IWA.

Wang, D., Thunéll, S., Lindberg, U., Jiang, L., Trygg, J., & Tysklind, M. (2022). Towards better process management in wastewater treatment plants: Process analytics based on SHAP values for tree-based machine learning methods. *Journal of Environmental Management, 301*, 113941.

Wang, D., Thunéll, S., Lindberg, U., Jiang, L., Trygg, J., Tysklind, M., & Souihi, N. (2021). A machine learning framework to improve effluent quality control in wastewater treatment plants. *Science of The Total Environment, 784*, 147138.

Wang, Z., Wei, A., Tang, K., Shi, H., Zou, J., Hu, H., & Zhu, Y. (2025). Enhanced accuracy and interpretability of nitrous oxide emission prediction of wastewater treatment plants through machine learning of univariate time series: A novel approach of learning feature reconstruction. *Journal of Water Process Engineering, 71*, 107263.

Ye, G., Wan, J., Deng, Z., Wang, Y., Zhu, B., Yan, Z., & Ji, S. (2024). Machine learning-based prediction of biological oxygen demand and unit electricity consumption in different-scale wastewater treatment plants. *Journal of Environmental Chemical Engineering, 12*(2), 111849.

Yoon, S., Kim, S.-S., Chae, S.-H., & Park, N.-S. (2019). Introducing new outlier detection method using robust statistical distance in water quality data. *Desalination and Water Treatment, 149*, 157-163.

Zhou, Z.-H. (2012). *Ensemble methods: Foundations and algorithms*. CRC Press.

doi:10.6342/NTU202503249