Environmental Science

AIMS Environmental Science, 12(5): 916–935.
DOI: 10.3934/environsci.2025040
Received: 12 July 2025
Revised: 05 October 2025
Accepted: 14 October 2025
Published: 29 October 2025

https://www.aimspress.com/journal/environmental

Research article

Salinity forecasting in Chao Phraya River using XGBoost with missing data handling

Jiramate Changklom[1], Trang Prommana[1], Phakawat Lamchuan[2] and Adichai Pornprommin[1,]*

[1] Department of Water Resources Engineering, Faculty of Engineering, Kasetsart University, Bangkok 10900, Thailand

[2] Office of Engineering and Architectural Design, Royal Irrigation Department, Bangkok 10900, Thailand

* **Correspondence:** Email: fengacp@ku.ac.th.

Abstract: Machine learning models for water quality prediction often encounter challenges due to incomplete datasets resulting from issues such as sensor failures or data transmission errors. However, XGBoost can learn from incomplete data and make predictions even when some input values are missing, without requiring any manual intervention by the user. This offers a considerable advantage over other models that require complete datasets. To test this capability, we utilized XGBoost to forecast salinity levels in the Chao Phraya River, Thailand, based on investigating forecasts at lead times of 24 and 48 hours, using four predictors: the water levels from three stations and the salinity level. The tested model demonstrated high predictive performance across both complete and incomplete subsets of the data. Where the complete input dataset was available, our XGBoost model outperformed the single-level ANN model and was comparable to the multilevel ANN model. Additionally, we enhanced XGBoost's ability to forecast under incomplete data conditions by utilizing both real and synthetic datasets. The synthetic dataset was generated by removing 1–3 predictor variables from the complete dataset. Based on the results, incorporating synthetic data into the training dataset substantially enhanced the model's robustness against missing data. Notably, training with synthetic data that excluded one variable at a time was sufficient for accurate predictions. These results underscore XGBoost's practicality and reliability for real-world water quality forecasting, particularly under conditions of data scarcity.

Keywords: XGBoost; missing data; salinity forecasting; Chao Phraya River

1. Introduction

Saltwater intrusion into the Chao Phraya River, Thailand, poses major challenges, especially affecting freshwater availability and quality for agriculture, drinking water, and industry in central Thailand, including the Bangkok metropolitan area [1,2]. This intrusion is primarily driven by seasonal drought, reduced river discharge, tidal fluctuations, and sea-level oscillations, resulting in higher salinity levels, particularly during the dry season [1–3]. Changklom et al. (2022) developed machine learning-based models that have been used to forecast and manage this phenomenon [1]. They utilized multiple linear regression and artificial neural network (ANN) modeling to forecast salinity. Both models were divided into two approaches: a single-level model designed to forecast salinity in all situations and a multilevel model combining sub-models tailored to specific upstream water levels. Among these options, the multilevel ANN approach was the most accurate, with performance remaining reasonably acceptable up to the 48-hour forecast horizon [1]. However, such forecasting systems face practical challenges, including technical issues related to monitoring stations, such as sensor malfunctions, data transmission failures, or power supply losses, which result in incomplete input datasets for the forecasting model and, in turn, sometimes lead to its complete failure [4].

Studies on forecasting model development frequently overlook or minimally address this important issue of incomplete data [5,6]. Typically, studies have excluded incomplete records and focused only on evaluating predictive accuracy based on complete datasets [7]. While several research efforts have aimed to manage missing data, most have concentrated on imputing data before model development rather than directly addressing data gaps that arise during the operational forecasting phase [8–10].

Nevertheless, some studies have specifically attempted to solve missing data problems occurring during forecasting. For example, Matusowsky et al. [11] used machine learning-based virtual sensors to replace missing sensor values, ensuring system integrity even when physical sensors fail. Chen et al. [12] extracted helpful information from incomplete data samples and transferred these learned features to structurally complete datasets, thus enhancing fault diagnosis accuracy. Kaveh et al. [13] integrated robust preprocessing with the model in remote sensing applications, significantly improving the handling of missing data. Mena et al. [14] proposed innovative ensemble methods and sensor dropout techniques to increase the robustness of model predictions to maintain accuracy despite the absence of sensor input. Lastly, Kaya et al. [15] applied Single Plurality Voting System classifiers to improve robustness by aggregating predictions from multiple classifiers, maintaining high prediction accuracy even with multiple sensor failures.

However, each approach from the studies above is often complex and difficult to implement, requiring advanced technical expertise, extensive computational resources, or customized architectures that may limit their widespread application in real-time forecasting systems. In contrast, at present, there is a machine learning model called XGBoost (eXtreme Gradient Boosting) that can handle missing data challenges automatically, requiring no additional preprocessing or manual intervention [16]. Its algorithm assigns a default direction for tree branching when a missing input value is encountered during the training phase [17].

However, decision tree-based models, such as XGBoost, are often overlooked in time-series forecasting because they lack a built-in mechanism to capture sequential dependencies explicitly [18]. Often, most researchers use sequence-based architecture that aligns naturally with time-series data, particularly those in the recurrent neural network (RNN)-based models, with long short-term memory networks being among the most widely adopted [19,20]. Yet, RNNs cannot inherently handle missing data, which poses a major challenge when deployed in real-world forecasting systems [21–23].

XGBoost is not only capable of forecasting under incomplete data conditions but also outperforms many other machine learning models in various aspects, such as computational efficiency, scalability, regularization capabilities, and prediction accuracy, making it highly suitable for practical forecasting applications [17]. In 2015, its effectiveness was evident on Kaggle, a leading platform for data science competitions, where 17 out of the 29 winning solutions featured XGBoost. Additionally, all the top 10 teams in KDDCup 2015 used XGBoost, highlighting its superior performance [24].

Some studies have attempted to apply XGBoost for water quality forecasting, such as Makumbura et al. [25], who combined it with a feature explanation method for forecasting water quality indices. Al Saleem et al. [26] applied XGBoost to predict wastewater treatment features under varying weather conditions, surpassing traditional models. Ahmed et al. [27] optimized XGBoost with metaheuristic techniques for dye degradation prediction, achieving 98.99% accuracy. However, none of these studies applied XGBoost to water quality forecasting while simultaneously addressing the issue of missing data.

Therefore, the current study aimed to test XGBoost by applying it to salinity forecasting in the Chao Phraya River, Thailand. The forecasting results from XGBoost using complete input data were compared to the single-level and multilevel ANN models developed by Changklom et al. (2022). In addition, the current study examined how well XGBoost could forecast when the input dataset was incomplete and explored practical strategies for handling missing data to improve XGBoost's performance under such conditions.

2. Materials and methods

2.1. Study site and data collection

The forecast evaluation utilized data from the Samlae Raw Water Pumping Station (S1) water quality monitoring station on the Chao Phraya River. This facility is critical in the region's water supply system, as it pumps raw water from the Chao Phraya River to produce tap water [1,28]. Accurate forecasting at this location is essential for effective water resource management and to ensure the safety and reliability of the municipal water supply [1].

The dataset used to build the forecasting model was the same as that used in the research conducted by Changklom et al. (2022), allowing for a direct comparison of model performance under consistent data conditions. The dataset contained salinity data from the Metropolitan Waterworks Authority (MWA) at S1 (the target for prediction), water level data from the Inburi (CPY006) and Tha Ruea (PAS009) stations, provided by the Hydro Informatics Institute (HII), and tidal level data from the Bangkok station (BB), provided by the Royal Thai Navy (RTN). The locations of these stations are presented in Figure 1.

CPY006 reflects upstream seawater flushing and river discharge variability regulated by the Chao Phraya Diversion Dam, providing information on how upstream flow management influences downstream salinity intrusion at S1. PAS009 represents the Pasak River's contribution and the effects of controlled flushing operations at the Pasak Reservoir and Rama VI Diversion Dam, which can alter flow timing and magnitude and thereby mitigate or exacerbate salinity intrusion at S1. BB, located near the river mouth, monitors tidal impacts and sea-level oscillations from the Gulf of Thailand, offering a direct measure of tidal forcing. Together, these stations capture the main hydrological drivers of salinity intrusion at S1—tidal forcing, runoff variability, and dam operations—ensuring that the forecasting model integrates both natural and regulated influences on salinity dynamics at S1.

Previous studies have also demonstrated that controlled releases from upstream dams can counteract tidal intrusion by increasing freshwater gradients, as shown in one-dimensional salinity intrusion simulations and estuarine observations [29]. These works highlight that salinity dynamics in the Chao Phraya estuary result from the combined influences of river discharge, tidal forcing, and sea-level fluctuations, consistent with the hydrological context of this study.

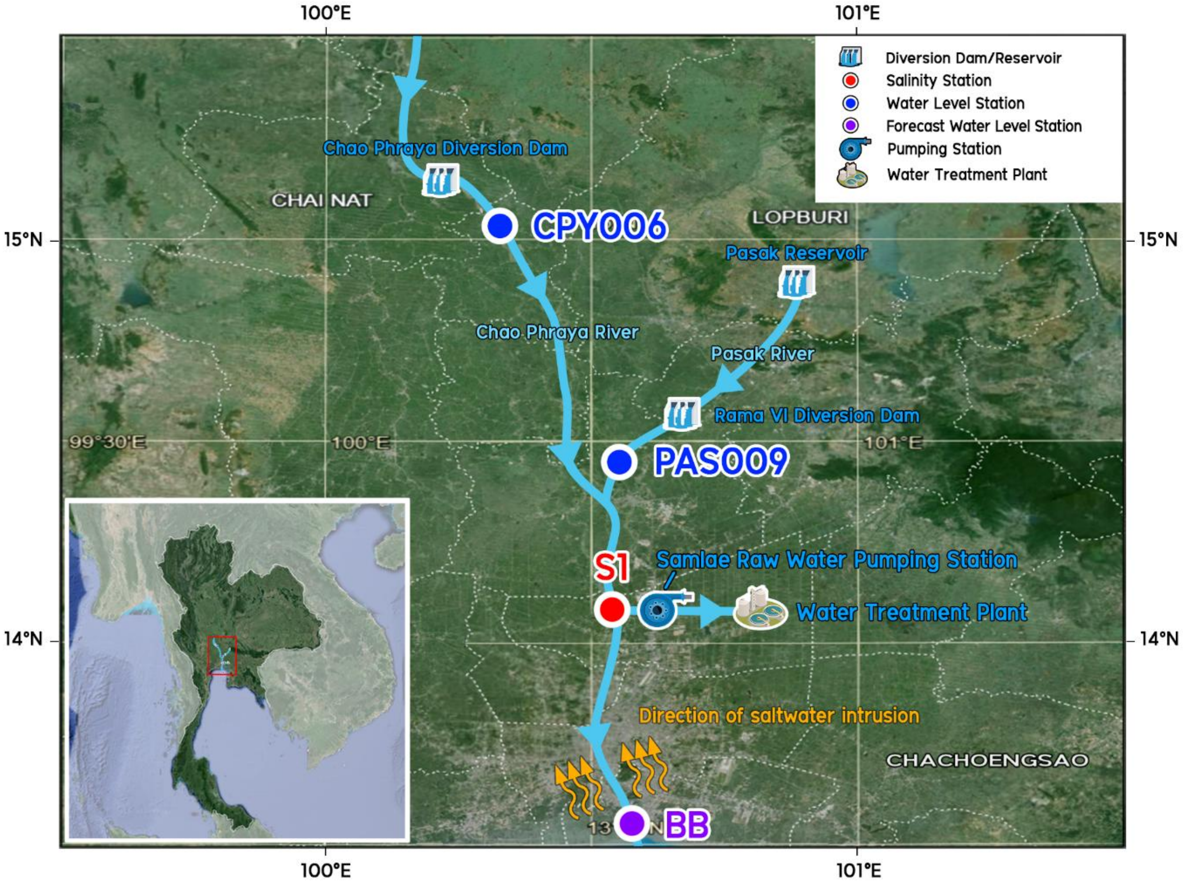

Figure 1. Location maps of measurement stations utilized in the current study. The inset map shows the location of Thailand, with the red box highlighting the study area in the downstream part of the Chao Phraya River Basin.

Data from various organizations were collected at different time resolutions. The MWA and HII data were recorded at 10-min intervals, while the RTN data were logged hourly. Therefore, the MWA and HII data were each aggregated into hourly averages to match the RTN data. The data period used in the current study was identical to that of Changklom et al. (2022), spanning 2014–2020. The combined dataset was split into training and test subsets. April and October 2020 were designated test periods, while the remaining data were used for model training. These two months represent the peaks of Thailand's dry and wet seasons, allowing for the capture of distinct salinity patterns.

The dataset contained 61,081 hourly records; however, a substantial portion of the records were missing, with the S1 station having 4% missing records, CPY006 having 11% missing records, and PAS009 having 27% missing records. In contrast, the BB station had no missing data. The missing records occurred in various forms: some were true gaps represented as NaN values, while others appeared as abnormal entries such as placeholder codes (e.g., -9999) or extremely large values outside the expected range. In this study, all abnormal values exceeding defined thresholds were treated as missing data.

The missingness in this dataset arose from a combination of random and systematic causes, but the majority were systematic, as long continuous gaps were observed due to sensor malfunctions and data transmission failures. For example, CPY006's gaps were in January to July 2018, and PAS009's gaps were in April to July 2018 (the time series is shown in Figure S1 in the Supplementary). These gaps pose challenges for modeling salinity at S1 and will serve as a test for evaluating the XGBoost model's performance in handling missing data. Details on data availability and missing data for each station are summarized in Table 1.

Table 1. Summary of data collected from multiple stations during 2014–2020.

Station	Parameter	Value			Amount of data in hours/percentage of total		
		Min	Mean	Max	Total	Available	Missing
S1	Salinity (g/L)	0.01	0.18	2.5	61,081	58,621 (96%)	2,460 (4%)
CPY006	Water level (m)	2.65	4.85	14.28	61,081	54,266 (89%)	6,815 (11%)
PAS009	Water level (m)	-0.65	1.09	4.47	61,081	44,312 (73%)	16,769 (27%)
BB	Water level (m)	-2.24	0.03	1.53	61,081	61,081 (100%)	0 (0%)

(Min = minimum; Max = maximum)

2.2. Lag time selection

The XGBoost model is designed for tabular data [17] and does not inherently consider temporal dependencies in time series data. To apply the XGBoost model to salinity forecasting, the time series data must be transformed into a tabular format, where each row represents a sample and each column represents a predictor variable at specific time lags.

An effective and widely accepted method for determining the appropriate time lags for each predictor variable is cross-correlation analysis [30]. This method measures the similarity between two time series as a function of the time displacement of one relative to the other. The time lag at which the correlation reaches its peak can be identified by calculating the cross-correlation between the predictor variables and the target salinity at S1. Then, these optimal lag values were used to construct lagged predictor features in the tabular dataset.

The results of the cross-correlation analysis are illustrated in Figure 2. The autocorrelation analysis of S1 revealed a distinct 24-hour cyclical pattern, reflecting tidal influences. Conversely, the cross-correlation between S1 and CPY006 (located upstream) had no 24-hour pattern; instead, there was a negative correlation, where higher upstream water levels reduced salinity at S1. The strongest correlation occurred with a 125-hour lag, indicating the approximate travel time for water from CPY006 to S1. Similarly, S1 and PAS009 were negatively correlated with the 24-hour tidal patterns. Lastly, the BB station, located near the sea, had a clear 24-hour periodicity, alternating between positive and negative correlations due to tidal cycles.

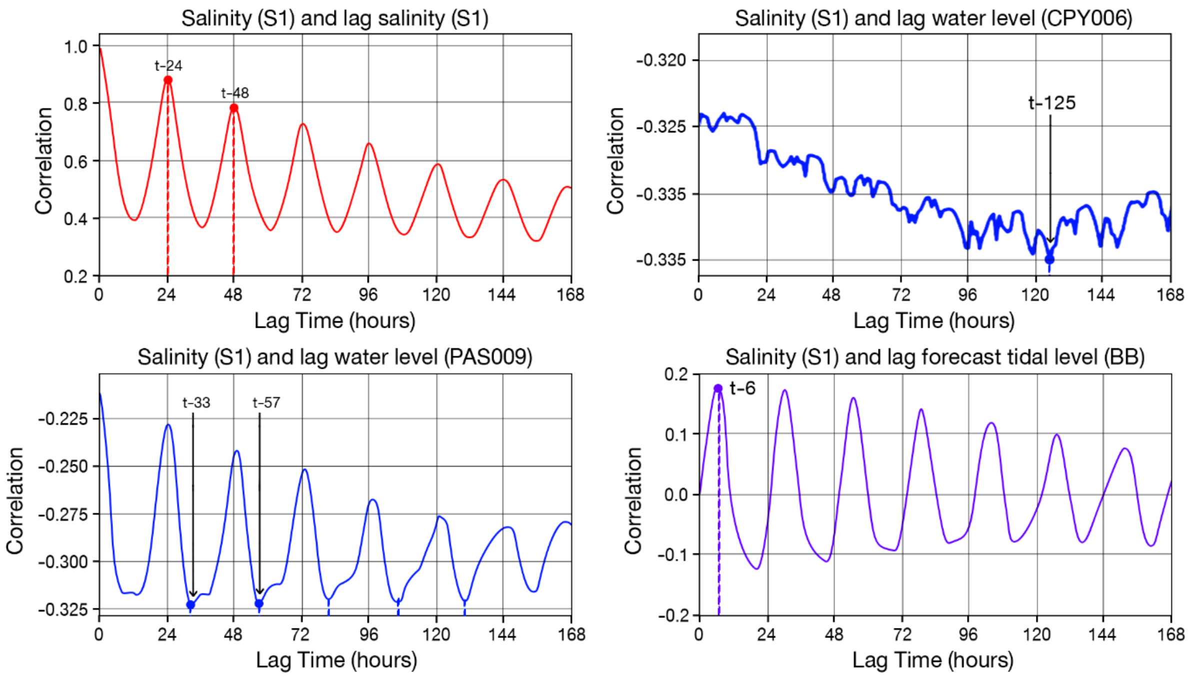

Figure 2. Cross-correlation analysis of S1 with predictor variables.

This study focused on forecasting salinity 24 and 48 hours in advance, which are time horizons where the forecasting performance from Changklom et al. (2022) remained within an acceptable accuracy range. For S1 and PAS009, predictor variables were selected based on the first prominent peak in the cross-correlation that occurred at or beyond the forecast horizon. The first peak for the 24-hour forecast model occurred at 24 hours for the S1 variable and at 33 hours for the PAS009 variable. The first peak for the 48-hour forecast occurred at 48 hours for the S1 variable and at 57 hours for the PAS009 variable. The peak correlation of CPY006 with the target salinity at S1 occurred at 125 hours, which exceeded both the forecast lead times. For the BB station, predictor variables were chosen from the earliest peak in the cross-correlation, which occurred at a 6-hour lag, shorter than all forecast horizons. This was possible because BB's tidal data were derived from harmonic tidal forecasting conducted by RTN, meaning the values were available in advance. Therefore, the earliest major peak could be selected, regardless of the forecast lead time. The relationships between the target and predictor variables, along with their selected time lags for both models, can be summarized using Eq (1):

$$S1_t = f(S1_{t-FP}, CPY006_{t-125}, BB_{t-6}, PAS009_{t-FP-9}) \tag{1}$$

where the subscripts denote specific points within the time series, *t* represents the forecast time, and *FP* is the forecast period, with all time intervals expressed in hours.

2.3. Availability of sample dataset

From Eq (1), data at different time lags were paired into a dataset for training and testing the XGBoost model, referred to as the sample dataset. Due to lag time effects and the time alignment required to pair target and predictor variables, the final sample size was reduced to 58,504 from the original 61,081 records. The resulting dataset was divided into 57,051 samples for model training and 1453 samples for model testing.

In contrast, the model used in Changklom et al. (2022) could only utilize approximately 35,000 samples (training dataset) due to its requirement for complete datasets (no missing data) and more predictor variables used. This limitation did not apply to the XGBoost model, which is capable of handling incomplete records. As a result, the XGBoost model in the current study benefited from a training dataset that was about 37% larger and a testing dataset that was approximately 13% larger than those used in Changklom et al. (2022). Table 2 summarizes the number of training and testing samples used in the single-level and multilevel ANN models of Changklom et al. (2022) and in the current study.

Table 2. Sample dataset sizes after all unavailable datasets have been removed.

Forecast period	Amount of data in hours/percentage of total				
	Single-level ANN	Multilevel ANN	XGBoost		
	Total	Total	Total	Complete	Incomplete
Training dataset					
24 hours	35148	35033	57051	36254 (64%)	20797 (36%)
48 hours	35016	34920	57051	36132 (63%)	20919 (37%)
Test dataset					
24 hours	1253	1259	1453	1262 (87%)	191 (13%)
48 hours	1251	1259	1453	1262 (87%)	191 (13%)

2.4. Model evaluation criteria

The performance of the models was evaluated using two statistical metrics, root mean square error (RMSE) and Nash–Sutcliffe efficiency (NSE), presented in Eqs (2) and (3), respectively:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(P_i - O_i)^2} \tag{2}$$

$$NSE = 1 - \frac{\sum_{i=1}^{n}(P_i - O_i)^2}{\sum_{i=1}^{n}(O_i - \bar{O})^2} \tag{3}$$

where P_i is the predicted value, O_i is the observed value, $\bar{O}$ is the mean of the observed values, and n is the total number of observations.

RMSE measures the average magnitude of prediction errors, emphasizing larger errors due to their squared term, providing insights into the absolute performance of the model. NSE evaluates the predictive skill of hydrological models, comparing the relative magnitude of residual variance against observed data variance. NSE values range from negative infinity to 1, where 1 indicates perfect predictions, values above 0 suggest acceptable performance, and negative values imply predictions worse than using the mean observed data as predictions [31].

2.5. Impact analysis of individual predictors

The impact of missing data from each predictor variable was analyzed to investigate the individual contribution of each predictor variable to the overall forecasting performance. Typically, this is done by simulating a missing input for each predictor variable individually, one at a time, and observing the resulting change in model accuracy [32–35]. The analysis began by filtering the test dataset to include only samples with complete values for all variables. This filtering step yielded 1262 fully complete records.

Based on this subset, a series of controlled missing-data scenarios were constructed by systematically removing the input values of one predictor variable at a time, while keeping the others unchanged. This generated five distinct test configurations: one baseline scenario, where all the predictors were present; and four scenarios, where each of the following predictors was individually omitted: $S1_{t\text{-}FP}$, $CPY006_{t\text{-}125}$, $PAS009_{t\text{-}FP\text{-}9}$, and $BB_{t\text{-}6}$.

Each of these scenarios was evaluated using the same trained XGBoost model, and the performance was assessed using the RMSE and NSE metrics. The differences in performance between each incomplete-data scenario and the baseline provided insight into how crucial each predictor variable was for maintaining model accuracy. This method enables a clearer understanding of the model's sensitivity to missing input data and highlights which variables are most influential for reliable salinity forecasts in the Chao Phraya River system.

2.6. Synthetic data augmentation

With approximately 37% of the training dataset comprising incomplete records, there is a risk that this amount of incomplete data may adversely impact the model's ability to generalize effectively in missing data scenarios. This issue was addressed by synthesizing additional incomplete data samples and incorporating them into the training dataset. In many studies, synthetic data was added to training datasets, with the results showing that such augmentation improved model accuracy. For example, López-Chacón et al. [36] improved streamflow predictions using synthetic hydrological data. Yang et al. [37] enhanced medical image classification with synthetic data. Duffy et al. [38] boosted legal document accuracy using synthetic data augmentation. Chawla et al. [39] generated synthetic minority-class instances to address class imbalance. Li et al. [40] showed that synthetic text improves model performance in natural language understanding.

Therefore, synthetic data augmentation was applied to generate additional training samples with structured missing inputs for the current study. Initially, the training dataset was filtered to include only fully complete samples. From this filtered set, synthetic datasets were created by removing variables in various structured patterns: one variable at a time, two variables at a time, and three variables at a time. The rationale for this design is that the dataset contains four predictor variables in total, so removing one to three of them systematically covers all possible cases of incomplete inputs that may occur during operational forecasting. Then, these artificially incomplete datasets were combined incrementally with the original training dataset, creating three augmented datasets: 1) a dataset containing synthetic records missing exactly one variable; 2) a dataset containing synthetic records missing one or two variables; and 3) a dataset containing synthetic records missing one, two, or three variables.

Each augmented training dataset was evaluated to determine the optimal amount of synthetic data necessary to enhance predictive accuracy when forecasting with incomplete inputs. Once the best augmentation strategy had been identified, the final model was tested comprehensively across various scenarios, consisting of the full test dataset, subsets of the fully complete test data, and subsets of the incomplete test data, followed by a detailed assessment of how missing each specific predictor individually influenced the forecasting performance. This method provided insight into the optimal use of synthetic data augmentation and its potential to substantially enhance forecasting reliability under incomplete data conditions. To clearly illustrate the overall methodological process of this study, a flowchart summarizing the steps from data collection through model evaluation is presented in Figure 3.

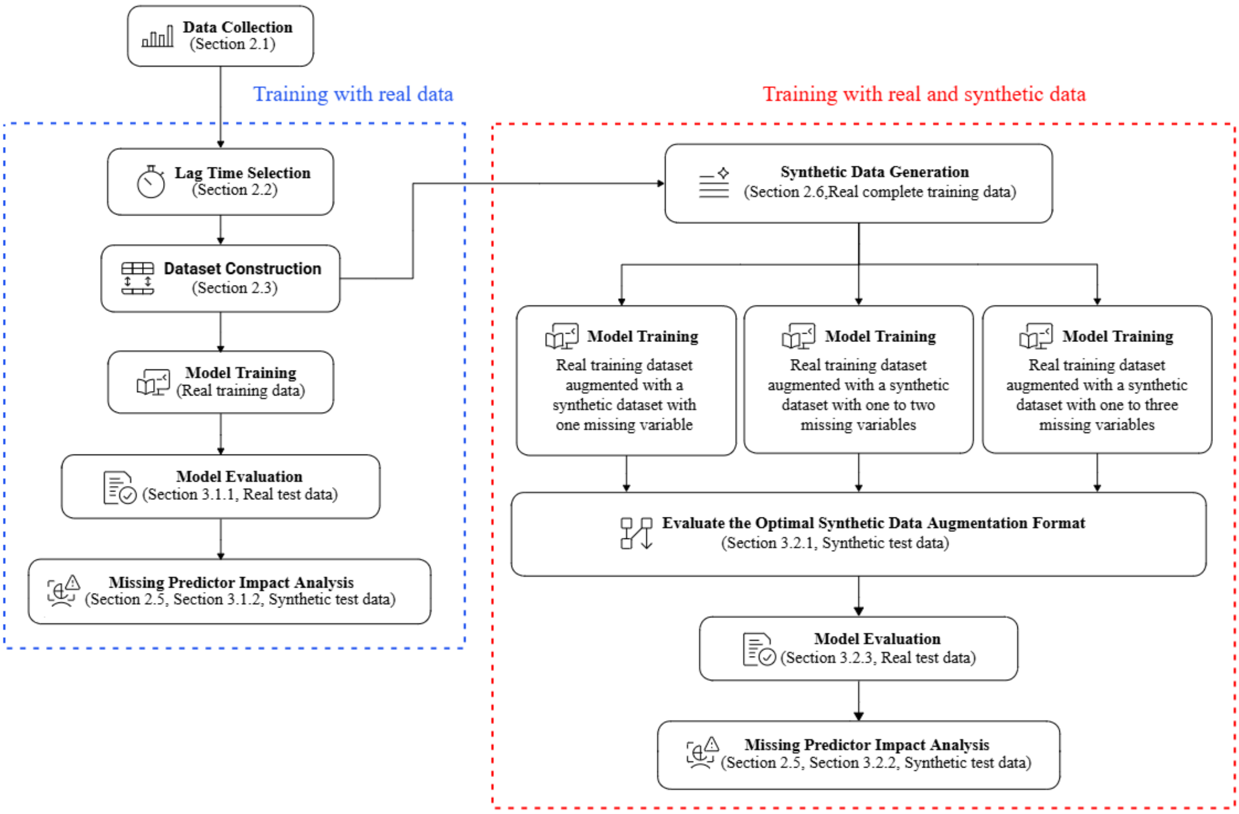
Data Collection
(Section 2.1)
Training with real data
Lag Time Selection
(Section 2.2)
Dataset Construction
(Section 2.3)
Model Training
(Real training data)
Model Evaluation
(Section 3.1.1, Real test data)
Missing Predictor Impact Analysis
(Section 2.5, Section 3.1.2, Synthetic test data)
Training with real and synthetic data
Synthetic Data Generation
(Section 2.6,Real complete training data)
Model Training
Real training dataset augmented with a synthetic dataset with one missing variable
Model Training
Real training dataset augmented with a synthetic dataset with one to two missing variables
Model Training
Real training dataset augmented with a synthetic dataset with one to three missing variables
Evaluate the Optimal Synthetic Data Augmentation Format
(Section 3.2.1, Synthetic test data)
Model Evaluation
(Section 3.2.3, Real test data)
Missing Predictor Impact Analysis
(Section 2.5, Section 3.2.2, Synthetic test data)

Figure 3. Methodological workflow summarizing data processing, model development, and evaluation steps.

3. Results and discussion

3.1. Evaluation model performance using real data

3.1.1. Continuous salinity forecast evaluation during testing periods

This section evaluates the XGBoost model's forecasting accuracy by comparing its predictions to observed salinity data at the Samlae Raw Water Pumping Station. Figure 4 illustrates the model's performance during two critical testing periods: April 2020 (the dry season) and October 2020 (the wet season). The figure displays forecasted salinity values at 24- and 48-hour lead times compared to the actual, observed salinity levels.

During the dry season (Figure 4a), the model accurately captured the timing and magnitude of salinity peaks, particularly at the 24-hour horizon, with a close alignment between observed and predicted values. The 48-hour forecasts maintained reasonable accuracy, although slightly less precise during peak periods, especially under conditions of incomplete data.

In the wet season (Figure 4b), the observed salinity was significantly lower and more stable. XGBoost consistently provided accurate predictions at 24 and 48-hour horizons, demonstrating robustness even with scattered, incomplete predictor data. Predictions remained below the MWA surveillance threshold, confirming the model's practical applicability in operational monitoring scenarios.

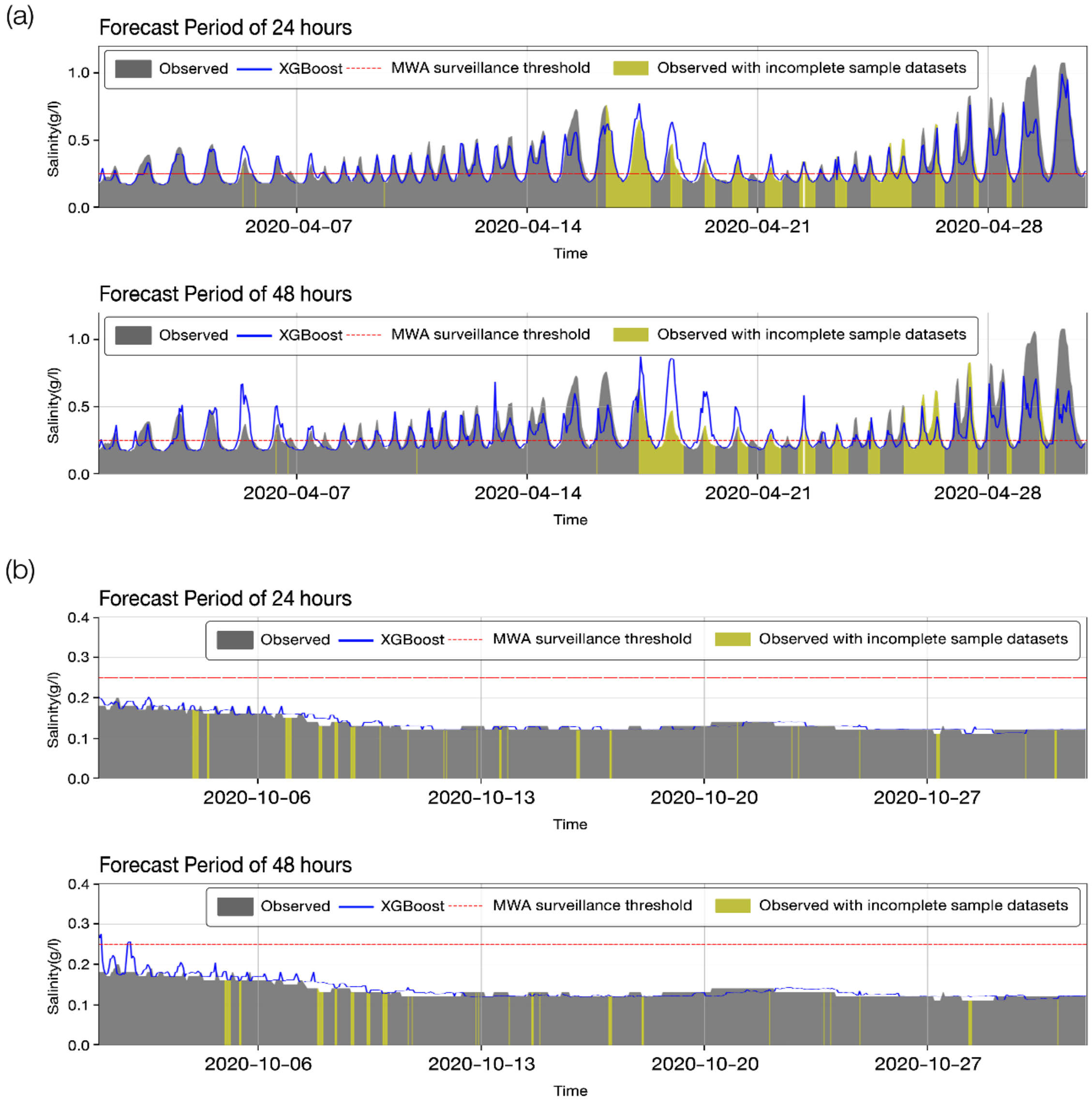

Figure 4. Forecasted versus observed salinity during (a) April 2020 (dry season) and (b) October 2020 (wet season) for forecast horizons of 24 and 48 hours. Shaded areas in gray represent observed data, blue lines indicate XGBoost predictions, yellow bars highlight time steps associated with incomplete predictor data, and the red dashed line denotes the MWA surveillance threshold.

Table 3 presents a comprehensive comparison of the forecast performance between the proposed XGBoost model and the ANN models developed by Changklom et al. (2022), including both single-level and multilevel ANN models, using the RMSE and NSE metrics across two forecast horizons: 24 and 48 hours. Performance was further assessed under three distinct conditions: when all input predictor variables are available (complete datasets), when some predictor data are missing (incomplete datasets), and across the entire test dataset (total datasets).

Under complete data conditions—the only scenario where the single-level and multilevel ANN models were applicable due to their inability to process missing inputs—XGBoost outperformed the single-level ANN at both forecast horizons (e.g., at 24 hours, RMSE = 0.057 g/L vs. 0.069 g/L; NSE = 0.862 vs. 0.796) and delivered accuracy comparable to the multilevel ANN model (e.g., RMSE = 0.054 g/L, NSE = 0.875 at 24 hours). Paired bootstrap resampling (10,000 iterations) confirmed that XGBoost significantly outperformed the single-level ANN, while its performance was generally comparable to the multilevel ANN, with no consistent evidence of significant differences across horizons (Supplementary Table S1). These results highlight that XGBoost offers a more effective modeling approach than ANN in this context. When compared to the single-level ANN, XGBoost consistently produced better accuracy. Furthermore, despite being a single unified model, XGBoost achieved predictive performance comparable to the multilevel ANN, consisting of several sub-models tailored to different input scenarios. This demonstrates XGBoost's capacity to match more complex architectures while retaining simplicity and efficiency.

Table 3. Forecast performance of XGBoost, single-level ANN, and multilevel ANN across 24- and 48-hour horizons under complete, incomplete, and total data scenarios.

Forecast period	Model	Sample dataset	Metric	
			RMSE (g/L)	NSE
24 hours	Single-level ANN	Complete	0.069	0.796
	Multilevel ANN	Complete	0.054	0.875
	XGBoost	Complete	0.057	0.862
	XGBoost	Incomplete	0.060	0.796
	XGBoost	Total	0.057	0.857
48 hours	Single-level ANN	Complete	0.096	0.605
	Multilevel ANN	Complete	0.085	0.687
	XGBoost	Complete	0.084	0.694
	XGBoost	Incomplete	0.120	0.230
	XGBoost	Total	0.090	0.648

Where input data was incomplete—a condition that ANN cannot accommodate—XGBoost had a clear advantage by maintaining predictive performance. Specifically, on the 24-hour forecast horizon, the model’s accuracy remained relatively stable compared to its performance under complete data (RMSE = 0.060 g/L, NSE = 0.796), showing only a minor reduction in the RMSE and NSE values. However, at the 48-hour forecast horizon, the impact of missing data was more pronounced, resulting in a more noticeable decline in accuracy (RMSE = 0.120 g/L, NSE = 0.230). Nonetheless, because incomplete samples constituted only 13% of the test dataset, the overall predictive accuracy when using the full dataset (both complete and incomplete samples) did not deteriorate substantially from that achieved with only the complete samples, for both the 24-hour (RMSE = 0.057 g/L, NSE = 0.857) and 48-hour (RMSE = 0.090 g/L, NSE = 0.648) models.

3.1.2. Impact of missing individual predictors

This section assesses the sensitivity of the XGBoost model to the absence of individual predictor variables. Specifically, it investigates how the exclusion of one predictor at a time affected forecast accuracy at two lead times: 24 and 48 hours; an ablation analysis is shown in Table 4.

When the input data was complete, the model achieved high accuracy at the 24-hour forecast horizon (RMSE 0.057 g/L, NSE 0.862). On the 48-hour horizon, the accuracy remained high (RMSE 0.084 g/L, NSE 0.694). Examining the model's sensitivity to the absence of individual predictors, $S1_{t\text{-}FP}$ and $BB_{t\text{-}6}$ were the most influential. On the 24-hour forecast horizon, $BB_{t\text{-}6}$ had the most impact on prediction performance when omitted (RMSE = 0.240 g/L, NSE = -1.445), while at the 48-hour horizon, the $S1_{t\text{-}FP}$ variable became the most important (RMSE = 0.156 g/L, NSE = -0.048). Conversely, the absence of the $CPY006_{t\text{-}125}$ or $PAS009_{t\text{-}FP\text{-}9}$ variables had relatively minimal impact, suggesting the model could still maintain acceptable accuracy without these inputs.

Although both $S1_{t\text{-}FP}$ and $BB_{t\text{-}6}$ proved important when missing, further insight from the correlation analysis in Figure 2 indicated that the BB variable had a relatively low correlation with the target, suggesting it should not be a highly influential predictor. In addition, we conducted a feature importance analysis (presented in Supplementary Figure S2), which yielded consistent results, reinforcing that BB is unlikely to exert a disproportionately strong influence on the forecasts. Therefore, the unexpectedly strong decline in model accuracy when $BB_{t\text{-}6}$ was absent was likely due to the nature of the training data: BB had no missing values in the training set, which may have biased the model to over-rely on this variable. This observation motivated the next set of experiments described in the following section, where artificially introduced missing values were added to the training set to assess whether model robustness could be improved.

Table 4. Forecast accuracy degradation when each predictor variable is removed.

Forecast period	Variable removed	Metric	
		RMSE (g/L)	NSE
24 hours	-	0.057	0.862
	$S1_{t\text{-}FP}$	0.152	0.013
	$CPY006_{t\text{-}125}$	0.055	0.871
	$PAS009_{t\text{-}FP\text{-}9}$	0.054	0.876
	$BB_{t\text{-}6}$	0.240	-1.445
48 hours	-	0.084	0.694
	$S1_{t\text{-}FP}$	0.156	-0.048
	$CPY006_{t\text{-}125}$	0.094	0.623
	$PAS009_{t\text{-}FP\text{-}9}$	0.079	0.733
	$BB_{t\text{-}6}$	0.094	0.616

3.2. Evaluation of model performance using combined real and synthetic data

3.2.1. Effect of synthetic data augmentation on forecast performance

This section evaluates how different synthetic data augmentation strategies affected the forecast performance of the XGBoost model under various conditions of missing input data. The aim was to determine which augmentation method produced the best balance between prediction accuracy and robustness when some predictor variables were unavailable during inference. Table 4 presents the results evaluating the effect of synthetic data augmentation on the XGBoost model's performance under different incomplete data conditions. The table presents four augmentation strategies—no augmentation, synthetic data with one missing variable, synthetic data with one to two missing variables, and synthetic data with one to three missing variables. These strategies were tested across datasets containing complete inputs, missing one, missing two, and missing three predictor variables. Each forecast horizon (24 and 48 hours) was assessed separately based on the RMSE and NSE metrics.

For the dataset with complete predictor variables, adding synthetic data slightly degraded performance. For example, the 24-hour RMSE increased from 0.057 to 0.063 g/L, and NSE decreased from 0.862 to 0.831 in the most extreme augmentation case. However, for the datasets with missing predictor variables, synthetic augmentation provided clear benefits. Specifically, in the dataset with one missing variable, the 24-hour NSE improved from 0.079 (no augmentation) to 0.689 (three-variable synthetic), while the RMSE dropped from 0.125 to 0.078 g/L. Similar trends were observed at the 48-hour horizon and for datasets with two and three missing variables. For example, in the test set with two missing variables, the NSE increased from -1.066 (no augmentation) to 0.375 (augmentation with 1–3 missing variables), and the RMSE reduced from 0.181 to 0.117 g/L.

These findings confirmed that synthetic augmentation greatly enhanced model robustness, particularly when multiple inputs were missing during inference. Among the augmentation strategies, Table 5 demonstrates that augmenting synthetic data containing only one missing variable was sufficient to achieve substantial gains in accuracy. For example, at the 24-hour horizon with one missing variable, the NSE improved from 0.079 (no augmentation) to 0.681 with single-variable augmentation, while more complex augmentation strategies did not show consistent improvement—NSE dropped slightly to 0.679 with 1–2 missing variables, and only marginally increased to 0.689 with 1–3 missing variables. Similar patterns were apparent at the 48-hour horizon, where the NSE increased from 0.481 (no augmentation) to 0.542 with single-variable augmentation, while barely improving with more complex strategies. This outcome is consistent with the amount of simultaneous loss of multiple predictors being relatively low (approximately 5.8% with two missing variables and 0.2% with three missing variables of the dataset) compared to single-variable missingness (approximately 30.0% of the dataset). More complex augmentation schemes will therefore introduce unrealistic missing patterns or lead to over-regularization, which explains why they did not provide additional benefits. These findings indicated that augmenting with only one missing variable was the optimal approach, offering nearly the highest performance improvement without added complexity. Consequently, this strategy was adopted in the subsequent analyses.

Table 5. Forecast performance (RMSE and NSE) of the XGBoost model under varying synthetic data augmentation strategies and test data completeness levels.

Synthetic data augmentation scenario	Forecast period	Metric RMSE (g/L)	NSE
Filtered test dataset with complete variables			
No augmentation	24 hours	0.057	0.862
	48 hours	0.084	0.694
Synthetic data with one missing variable	24 hours	0.061	0.842
	48 hours	0.089	0.656
Synthetic data with one and two missing variables	24 hours	0.063	0.831
	48 hours	0.092	0.634
Synthetic data with one to three missing variables	24 hours	0.060	0.848
	48 hours	0.089	0.662
Filtered test dataset with one missing variable			
No augmentation	24 hours	0.125	0.079
	48 hours	0.106	0.481
Synthetic data with one missing variable	24 hours	0.080	0.681
	48 hours	0.101	0.542
Synthetic data with one and two missing variables	24 hours	0.080	0.679
	48 hours	0.103	0.532
Synthetic data with one to three missing variables	24 hours	0.078	0.689
	48 hours	0.101	0.542
Filtered test dataset with two missing variables			
No augmentation	24 hours	0.175	-0.884
	48 hours	0.181	-1.066
Synthetic data with one missing variable	24 hours	0.106	0.432
	48 hours	0.118	0.368
Synthetic data with one and two missing variables	24 hours	0.104	0.465
	48 hours	0.118	0.371
Synthetic data with one to three missing variables	24 hours	0.102	0.473
	48 hours	0.117	0.375
Filtered test dataset with three missing variables			
No augmentation	24 hours	0.191	-0.744
	48 hours	0.207	-1.179
Synthetic data with one missing variable	24 hours	0.131	0.184
	48 hours	0.137	0.155
Synthetic data with one and two missing variables	24 hours	0.128	0.231
	48 hours	0.134	0.190
Synthetic data with one to three missing variables	24 hours	0.128	0.223
	48 hours	0.136	0.174

3.2.2. Impact of synthetic data augmentation on predictor sensitivity

This section examines the impact of synthetic data augmentation on the sensitivity of the

XGBoost model to individual missing predictor variables. It expands upon the analysis in Section 3.1.2 by re-running the predictor removal test on a model trained with synthetic data that includes one-variable missing patterns. The goal was to assess whether this augmentation strategy improved model robustness when key inputs were unavailable. Table 6 compares forecasting accuracy between the complete input data and the scenarios where each predictor variable had been systematically removed during testing. Forecast performance was reported using both the RMSE and NSE metrics for 24-hour and 48-hour lead times.

Compared with the earlier results using only real training data (Table 4), the model trained with synthetic data augmentation improved consistency and reduced sensitivity to missing predictors. The degradation in accuracy was less pronounced across both the 24 and 48-hour forecast horizons. Notably, the RMSE remained less than 0.1 g/L and the NSE greater than 0.6 for most missing-variable scenarios. The impact of missing $S1_{t\text{-}FP}$ was reduced; however, the performance drop was clearly visible, indicating that this variable continued to play a critical role in model accuracy, even with enhanced robustness from data augmentation.

The impact of the missing $BB_{t\text{-}6}$, which previously caused a large performance drop, was now greatly mitigated. On the 24-hour forecast horizon, the NSE for $BB_{t\text{-}6}$ improved from -1.445 (without augmentation) to 0.829, and the RMSE dropped from 0.240 to 0.063 g/L. Similar improvements were observed at the 48-hour forecast horizon, indicating that the model became more resilient to the absence of the $BB_{t\text{-}6}$ variable after being exposed to similar missing-data patterns during training. This suggests that synthetic augmentation enabled the model to better generalize and maintain stability, especially in cases where those variables had few-to-no missing values in the original training data.

Table 6. Forecast accuracy degradation when each predictor variable is removed, after training with synthetic missing variable data.

Forecast period	Variable removed	Metric	
		RMSE (g/L)	NSE
24 hours	-	0.061	0.842
	$S1_{t\text{-}FP}$	0.137	0.196
	$CPY006_{t\text{-}125}$	0.062	0.834
	$PAS009_{t\text{-}FP\text{-}9}$	0.056	0.864
	$BB_{t\text{-}6}$	0.063	0.829
48 hours	-	0.089	0.656
	$S1_{t\text{-}FP}$	0.139	0.175
	$CPY006_{t\text{-}125}$	0.090	0.653
	$PAS009_{t\text{-}FP\text{-}9}$	0.081	0.718
	$BB_{t\text{-}6}$	0.094	0.620

3.2.3. Overall impact of synthetic data on real-world forecasting

This section examines the effect of combining synthetic and real data on the overall forecasting performance of the XGBoost model, with a particular focus on its performance under various test data conditions: total, complete, and incomplete. Table 7 presents the RMSE and NSE metrics for the forecast horizons of 24 and 48 hours, comparing models trained with real data only to those trained with both real and synthetic data that included one-variable missing patterns.

Under these incomplete test conditions, the models trained with both the real and synthetic data had improved robustness compared to models trained on real data alone. The 24-hour forecast RMSE improved from 0.060 to 0.054 g/L, and the forecast NSE increased from 0.796 to 0.830. A similar improvement occurred in the 48-hour forecast, where the RMSE dropped from 0.120 to 0.113 g/L and the NSE increased from 0.230 to 0.319.

However, when the test dataset contained only complete samples, the combined training data resulted in a slight reduction in performance. At the 24-hour forecast horizon, the RMSE increased slightly from 0.057 to 0.061 g/L, and the NSE decreased from 0.862 to 0.842; at the 48-hour horizon, the RMSE increased from 0.084 to 0.089 g/L, and the NSE decreased from 0.694 to 0.656. The total dataset performance had similar small declines because the real-world test dataset was composed almost entirely of complete samples (87%), causing performance on the total dataset to closely follow the trends observed in the complete subset.

These results highlighted that synthetic data augmentation was effective, especially when the model was expected to operate under conditions with missing inputs. While it introduced minor trade-offs in accuracy when all inputs are available, the gains in robustness under more realistic, imperfect data conditions made it a worthwhile strategy. However, this trade-off could be managed by designing a synthetic augmentation to reflect the real data loss pattern as much as possible. For example, BB is a predictor derived from tidal forecasts and had no missing values in the original training dataset, making it unlikely to be missing in a real-world application. Therefore, omitting BB in synthetic augmentation was not necessary and may even have introduced unrealistic scenarios, which could have degraded model performance.

Table 7. Forecast performance (RMSE and NSE) of the XGBoost model trained on real data versus combined real and synthetic data under different test data completeness levels.

Forecast period	Training dataset	Test dataset	Metric	
			RMSE (g/L)	NSE
24 hours	Real data	Total sample	0.057	0.857
		Complete sample	0.057	0.862
		Incomplete sample	0.060	0.796
	Combined real and synthetic data	Total sample	0.060	0.843
		Complete sample	0.061	0.842
		Incomplete sample	0.054	0.830
48 hours	Real data	Total sample	0.090	0.648
		Complete sample	0.084	0.694
		Incomplete sample	0.120	0.230
	Combined real and synthetic data	Total sample	0.093	0.624
		Complete sample	0.089	0.656
		Incomplete sample	0.113	0.319

4. Conclusions

The applicability and effectiveness of XGBoost were investigated for salinity forecasting in the Chao Phraya River, particularly under conditions involving incomplete data. XGBoost provided robust predictive capabilities, even when some predictor inputs were missing, a notable advantage over the

ANN-based model that required complete datasets. Under complete-data conditions, XGBoost achieved better predictive accuracy than the single-level ANN model and comparable accuracy to multilevel ANN models that rely on multiple specialized sub-models.

Synthetic data augmentation greatly enhanced the robustness of applying the model to incomplete data scenarios. Specifically, the introduction of synthetic samples with only one missing predictor consistently provided substantial improvements in accuracy and robustness. Although this approach slightly reduced forecast accuracy under conditions of complete data, the trade-off was minimal and manageable. Future model performance should be enhanced by further optimization of synthetic augmentation strategies to more closely mimic realistic patterns of missing data, such as avoiding the unnecessary omission of predictors (like the tidal variable), which are unlikely to contain missing data.

In conclusion, XGBoost showed considerable potential for practical, real-time water quality forecasting applications. Its built-in capacity to handle missing data effectively addresses common real-world issues such as sensor failures and intermittent data losses. Our framework for effectively handling missing data was developed using the Chao Phraya River system, with its complex interactions between salinity, tidal forces, and regulated discharge. Nevertheless, it should be applicable to other river systems with different hydrological regimes. Future studies on different river basins and data types are expected to further demonstrate its versatility and practicability. In addition, it could include refining synthetic data augmentation techniques and integrating additional advanced machine learning approaches to further enhance forecasting performance and resilience.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The authors gratefully thank the Metropolitan Waterworks Authority (MWA), the Hydro Informatics Institute (HII), and the Hydrographic Department of the Royal Thai Navy (RTN) for providing the data used in this study. The authors would also like to express their sincere gratitude to Asst. Prof. Dr. Thanawin Rakthanmanon for his valuable guidance and insightful advice throughout this research. T.P. was supported by a doctoral fellowship from the Faculty of Engineering, Kasetsart University.

Conflict of interest

The authors declare that there is no conflict of interest in this paper.

References

1. Changklom J, Lamchuan P, Pornprommin A (2022) Salinity forecasting on raw water for water supply in the Chao Phraya River. *Water* 14: 741. https://doi.org/10.3390/w14050741
2. Tomkratoke S, Kongkulsiri S, Narenpitak P, et al. (2025) Drought and salinity intrusion in the Lower Chao Phraya River: Variability analysis and modeling mitigation approaches. *EGUsphere* https://doi.org/10.5194/egusphere-2024-4052

3. Kulmart K, Charoenroongruang C (2024) Mathematical forecasting simulation of salinity intrusion in Chao Phraya River. *J Inf Optim Sci* 45: 181–198. https://doi.org/10.47974/JIOS-1410
4. Zhao Q, Zhu Y, Wan D, et al. (2018) Research on the data-driven quality control method of hydrological time series data. *Water* 10: 1712. https://doi.org/10.3390/w10121712
5. Ribeiro SM, de Castro CL (2022) Missing data in time series: A review of imputation methods and case study. *Learn Nonlinear Models* 20: 31–46.
6. Nijman SWJ, Leeuwenberg AM, Beekers I, et al. (2022) Missing data is poorly handled and reported in prediction model studies using machine learning: a literature review. *J Clin Epidemiol* 142: 218–229. https://doi.org/10.1016/j.jclinepi.2021.11.023
7. Gravesteijn BY, Sewalt CA, Venema E, et al. (2021) Missing Data in Prediction Research: A Five-Step Approach for Multiple Imputation, Illustrated in the CENTER-TBI Study. *J Neurotrauma* 38: 1842–1857. https://doi.org/10.1089/neu.2020.7218
8. Wen H, Pinson P, Gu J, et al. (2022) Wind energy forecasting with missing values within a fully conditional specification framework. *Int J Forecasting* https://doi.org/10.1016/j.ijforecast.2022.10.012
9. Nijman SWJ, Hoogland J, Groenhof TKJ, et al. (2021) Real-time imputation of missing predictor values in clinical practice. *Eur Heart J Digit Health* 2: 154–164. https://doi.org/10.1093/ehjdh/ztaa016
10. Zhang Y, Thorburn PJ (2022) Handling missing data in near real-time environmental monitoring: A system and a review of selected methods. *Future Gener Comput Syst* 128: 63–72. https://doi.org/10.1016/j.future.2021.09.033
11. Matusowsky M, Ramotsoela DT, Abu-Mahfouz AM (2020) Data imputation in wireless sensor networks using a machine learning-based virtual sensor. *J Sens Actuator Netw* 9: 25. https://doi.org/10.3390/jsan9020025
12. Chen D, Yang S, Zhou F (2019) Transfer learning based fault diagnosis with missing data due to multi-rate sampling. *Sensors* 19: 1826. https://doi.org/10.3390/s19081826
13. Kaveh M, Mesgari MS, Kaveh M (2025) A novel evolutionary deep learning approach for PM2.5 prediction using remote sensing and spatial–temporal data: A case study of Tehran. *ISPRS Int J Geo-Inf* 14: 42. https://doi.org/10.3390/ijgi14020042
14. Mena F, Arenas D, Dengel A (2024) Increasing the robustness of model predictions to missing sensors in Earth observation. *arXiv* preprint arXiv:2407.15512. https://doi.org/10.48550/arXiv.2407.15512
15. Kaya A, Keçeli AS, Catal C, et al. (2020) Sensor failure tolerable machine learning-based food quality prediction model. *Sensors* 20: 3173. https://doi.org/10.3390/s20113173
16. Ok E, Emmanuel M. Handling Missing Data in XGBoost. ResearchGate, 2024. Available from: https://www.researchgate.net/publication/390138079_Handling_Missing_Data_in_XGBoost
17. Chen T, Guestrin C (2016) XGBoost: A scalable tree boosting system. *Proc ACM SIGKDD Int Conf Knowl Discov Data Min* 22: 785–794. https://doi.org/10.1145/2939672.2939785
18. März A, Rasul K (2024) Forecasting with Hyper-Trees. *arXiv* preprint arXiv:2405.07836. https://doi.org/10.48550/arXiv.2405.07836
19. Petneházi G (2019) Recurrent neural networks for time series forecasting. *arXiv* preprint arXiv:1901.00069. https://doi.org/10.48550/arXiv.1901.00069
20. Khaldi R, El Afia A, Chiheb R, et al. (2024) What is the best RNN-cell structure to forecast each time series behavior? *Expert Syst Appl* https://doi.org/10.48550/arXiv.2303.07844

21. Che Z, Purushotham S, Cho K, et al. (2018) Recurrent neural networks for multivariate time series with missing values. *Sci Rep* 8:6085. https://doi.org/10.1038/s41598-018-24271-9
22. Mehdipour Ghazi M, Nielsen M, Pai A, et al. (2018) Robust training of recurrent neural networks to handle missing data for disease progression modeling. *arXiv* preprint arXiv:1808.05500. https://arxiv.org/abs/1808.05500
23. Lipton ZC, Kale DC, Wetzel R (2016) Modeling Missing Data in Clinical Time Series with RNNs. Proceedings of the Machine Learning for Healthcare Conference. Available from: https://proceedings.mlr.press/v56/Lipton16.pdf
24. Bekkerman R (2015) The present and the future of the KDD Cup competition: an outsider's perspective. LinkedIn. Available from: https://www.linkedin.com/pulse/present-future-kdd-cup-competition-outsiders-ron-bekkerman
25. Makumbura RK, Mampitiya L, Rathnayake N, et al. (2024) Advancing water quality assessment and prediction using machine learning models, coupled with explainable artificial intelligence (XAI) techniques like shapley additive explanations (SHAP). *Results Eng* 23: 102831. https://doi.org/10.1016/j.rineng.2024.102831
26. Al Saleem M, Harrou F, Sun Y (2024) Explainable machine learning methods for predicting water treatment plant features under varying weather conditions. *Results Eng* 21: 101930. https://doi.org/10.1016/j.rineng.2024.101930
27. Ahmed Y, Dutta KR, Chowdhury Nepu SN, et al. (2025) Optimizing photocatalytic dye degradation: A machine learning and metaheuristic approach for predicting methylene blue in contaminated water. *Results Eng* 25: 103538. https://doi.org/10.1016/j.rineng.2024.103538
28. Metropolitan Waterworks Authority. (2022). *MWA Consumer Confidence Report 2022*. Available from: https://www.mwa.co.th/wp-content/uploads/2023/03/2022-Annual-Water-Quality-Report-%E0%B8%AD%E0%B8%B1%E0%B8%87%E0%B8%81%E0%B8%A4%E0%B8%A9.pdf
29. Pokavanich T, Guo X (2024) Saltwater intrusion in Chao Phraya Estuary: A long, narrow and meandering partially mixed estuary influenced by water regulation and abstraction. *J Hydrol Reg Stud* 52: 101686. https://doi.org/10.1016/j.ejrh.2024.101686
30. Heyse J, Sheybani L, Vulliémoz S, et al. (2021) Evaluation of directed causality measures and lag estimations in multivariate time-series. *Front Syst Neurosci* 15: 620338. https://doi.org/10.3389/fnsys.2021.620338
31. Nash JE, Sutcliffe JV (1970) River flow forecasting through conceptual models part I — A discussion of principles. *J Hydrol* 10: 282–290. https://doi.org/10.1016/0022-1694(70)90255-6
32. Schouten RM, Lugtig P, Vink G (2018) Generating missing values for simulation purposes: A multivariate amputation procedure. *J Stat Comput Simul* 88: 2909–2930. https://doi.org/10.1080/00949655.2018.1491577
33. Rubright JD, Nandakumar R, Glutting JJ (2014) A simulation study of missing data with multiple missing X's. *Pract Assess Res Eval* 19: 10. https://doi.org/10.7275/9ew5-zd12
34. Bourguignon L, Lukas LP, Guest JD, et al. (2024) Studying missingness in spinal cord injury data: challenges and impact of data imputation. *BMC Med Res Methodol* 24:5. https://doi.org/10.1186/s12874-023-02125-x
35. Saar-Tsechansky M, Provost F (2007) Handling missing values when applying classification models. *J Mach Learn Res* 8: 1623–1657.

36. López-Chacón SR, Salazar F, Bladé E (2023) Combining synthetic and observed data to enhance machine learning model performance for streamflow prediction. *Water* 15: 2020. https://doi.org/10.3390/w15112020
37. Yang S, Kim K-D, Ariji E, et al. (2023) Evaluating the performance of generative adversarial network-synthesized periapical images in classifying C-shaped root canals. *Sci Rep* 13: 18038. https://doi.org/10.1038/s41598-023-45290-1
38. Duffy W, O'Connell E, McCarroll N, et al. (2025) Evaluating rule-based and generative data augmentation techniques for legal document classification. *Knowl Inf Syst*. https://doi.org/10.1007/s10115-025-02454-x
39. Chawla NV, Bowyer KW, Hall LO, et al. (2002) SMOTE: Synthetic Minority Over-sampling Technique. *J Artif Intell Res* 16: 321–357. https://doi.org/10.1613/jair.953
40. Li Y, Bonatti R, Abdali S, et al. (2024) Data generation using large language models for text classification: An empirical case study. *Proc Mach Learn Res* 235: 1–17.