Journal of
Hydroinformatics

Journal of Hydroinformatics Vol 25 No 6, 2541 doi: 10.2166/hydro.2023.158

Modeling ion constituents in the Sacramento-San Joaquin Delta using multiple machine learning approaches

Peyman Namadi *, Minxue He and Prabhjot Sandhu

California Department of Water Resources, 1516 9th Street, Sacramento, CA, USA
*Corresponding author. E-mail: peyman.hosseinzadehnamadi@water.ca.gov

PN, 0000-0002-5729-2951

ABSTRACT

Salinity is of paramount importance in shaping water quality, ecosystem health, and the capacity to sustain diverse human and environmental demands in estuarine environments. Electrical conductivity (EC) is commonly utilized as an indirect measure of salinity, serving as a proxy for estimating other ion constituents within the Sacramento-San Joaquin Delta (Delta) of California, United States. This study investigates and contrasts four machine learning (ML) models (Regression Trees, Random Forest, Gradient Boosting, and Artificial Neuronal Networks) for approximating ion constituent concentrations based on EC measurements, emphasizing the enhancement of conversion for constituents exhibiting pronounced non-linear relationships with EC. Among the four models, the Artificial Neuronal Networks model outshines the others in predicting ion constituents from EC, especially for those displaying strong non-linear relationships with EC. All four ML models surpass traditional parametric regression equations in terms of accuracy in estimating ion concentrations. Furthermore, an interactive web browser-based dashboard is developed, catering to users with or without programming expertise, enabling ion level simulation within the Delta. By furnishing more precise ion constituent estimations, this research enriches the understanding of salinity's effects on water quality in the Delta and fosters well-informed water management decisions.

Key words: artificial neural networks, decision trees, gradient boosting, ion constituents, random forests, Sacramento-San Joaquin Delta

HIGHLIGHTS

- The study applies four machine learning models to enhance the prediction of ion concentrations in the Sacramento-San Joaquin Delta based on electrical conductivity measurements.
- The research introduces an interactive web-based dashboard, facilitating simulations of ion levels and providing a user-friendly platform for understanding salinity's impact on water quality in the Delta.

1. INTRODUCTION

Estuaries, as dynamic and complex environments where rivers meet the sea, host a wide variety of ecological processes and support diverse ecosystems. These transitional zones are characterized by the interplay between fresh water and saltwater, resulting in a gradient of salinity levels. Salinity in estuarine areas is a critical factor influencing the distribution and survival of aquatic species (Attrill & Rundle 2002), and biogeochemical cycles (Nixon 1981). Furthermore, estuaries are often integral to human activities, such as commercial and recreational fishing, transportation, and coastal development. Consequently, understanding and accurately simulating salinity in estuaries is essential for preserving ecosystem integrity, managing water resources, and ensuring the sustainability of human activities in these areas.

In light of these considerations, researchers and water resource managers have been focusing on developing effective methods to estimate and predict salinity in estuarine environments (Cloern *et al.* 2014). Numerical models, remote sensing techniques, and data-driven approaches are some of the tools employed to simulate and monitor salinity dynamics. Accurate salinity simulations are crucial in informing decisions related to water resource allocation, environmental protection, and the sustainable management of estuarine ecosystems. In this context, the present study investigates salinity in the Sacramento-

San Joaquin Delta (Delta) of California, a critical estuarine region, with the aim of advancing the understanding of its influence on water quality and facilitating well-informed water management decisions.

Salinity in the Delta plays a vital role in determining water quality, the overall health of the ecosystem, and its ability to support various human and environmental needs, such as agricultural irrigation, habitat preservation, and drinking water supply (Rabalais *et al.* 2002; Kemp *et al.* 2005; Savenije 2005; Cloern & Jassby 2012). Salinity is a measure of the concentration of dissolved salts in water, and its accurate estimation is crucial for effective water resource management. Typically, salinity is measured indirectly as electrical conductance (EC) using automatic sensors and reported as specific conductance (Hutton *et al.* 2016; Rath *et al.* 2017; Bañón *et al.* 2021). EC, which represents the ability of water to conduct electric current, has a strong correlation with the concentration of dissolved ions in water.

EC can be employed as a predictor for other ion constituents in the Delta, including total dissolved solids (TDS), dissolved chloride (Cl^-), dissolved sulfate (SO_4^{2-}), dissolved sodium (Na^+), dissolved calcium (Ca^{2+}), dissolved magnesium (Mg^{2+}), dissolved potassium (K^+), dissolved bromide (Br^-), and alkalinity. These ion constituents have diverse impacts on water quality and its suitability for various uses. For example, high concentrations of chloride and sulfate ions can adversely affect crop yields (Maas & Hoffman 1977), while elevated levels of sodium and calcium ions can lead to issues such as soil degradation and scaling in water infrastructure.

The maximum safe levels of ion constituents in water depend on the intended use. For aquatic species living in the Delta, excessive concentrations of certain ions can be toxic, leading to negative effects on their survival, growth, and reproduction. The United States Environmental Protection Agency (US EPA) provides guidelines for maximum ion concentrations in fresh water to protect aquatic life (US EPA 2018). For drinking water, the World Health Organization (WHO) and US EPA establish maximum contaminant levels (MCLs) for various ions to ensure water safety and protect public health (WHO 2011; US EPA 2021). Given these guidelines, failure to accurately simulate ion concentrations can have severe implications. Incorrect estimations of ions like chloride or sulfate could result in the inappropriate allocation of water for agricultural use, leading to reduced crop yields or soil degradation. Similarly, elevated levels of certain ions could be toxic to aquatic life, disrupting local food chains and reducing biodiversity. Inaccurate simulations could also compromise the safety of drinking water, putting public health at risk. Therefore, the need for precise and reliable ion concentration simulation is not merely academic but has direct, tangible implications for environmental sustainability and human well-being.

However, the concentration of each ion constituent is measured from discrete water samples (i.e., grab samples) and is available much less frequently than EC. This limitation poses a challenge for continuous monitoring of water quality and informed decision-making. The need to convert EC to other constituents arises from the increasing interest in reporting model results in terms of other constituents, as salinity is currently modeled in terms of EC. Therefore, developing accurate and reliable regression models between EC and other ion constituents is critical for understanding the impacts of salinity on water quality in the Delta and for making informed management decisions.

Various regression models have been developed in previous studies, with EC as the predictor and individual constituents as the predictand, based on grab sample data in the Delta (Jung 2000; Suits 2002; Hutton 2006; Hutton *et al.* 2022; Denton 2015). Most recently, Hutton *et al.* (2022) developed simplified statistical equations to estimate salinity constituent concentrations from EC, assuming that three sources govern the salinity level in the Delta: seawater intrusion (Ocean source), fresh water (Sacramento River), and the agricultural source (drainage-influenced San Joaquin River). While their work significantly contributed to the field, the accuracy of their equations for ions with strong non-linear relationships with EC, such as alkalinity and potassium (K^+), requires improvement.

This study aims to address these limitations by developing machine learning (ML) models that can emulate and potentially improve upon the existing regression equations developed by Hutton *et al.* (2022) to simulate ion constituents from EC. A more recent study (Namadi *et al.* 2022) demonstrated the potential of ML algorithms as an alternative to parametric regression models for predicting ion constituents in the Delta. This study focused on the South Delta, testing ML models at seven stations during a short period from 2018 to 2020 when the grab samples were regularly collected. The findings provided preliminary evidence that ML can be successfully applied to estimate ion constituents from EC in the region.

Building on these promising results, the current study aims to further test the effectiveness of ML models for ion constituent prediction by expanding the analysis to a larger area and a more extended period. By using a more comprehensive dataset, we hope to validate the applicability of ML models to a broader context and establish their utility for estimating ion constituents across the entire Sacramento-San Joaquin Delta. Ultimately, this research will contribute to enhancing water quality monitoring and support better-informed water management decisions, ensuring the protection of aquatic life and the safety of drinking water supplies.

2. METHODOLOGY

2.1. Study locations and study dataset

The study area encompasses the interior Delta (Figure 1). In accordance with Hutton *et al.* (2022), the study area is divided into three distinct subregions to account for unique source water influences that vary by hydrology and season: Old-Middle River (OMR) Export Corridor, San Joaquin River Corridor, and South Delta. Figure 1 presents the boundaries of the three subregions, as well as the locations of the grab samples.

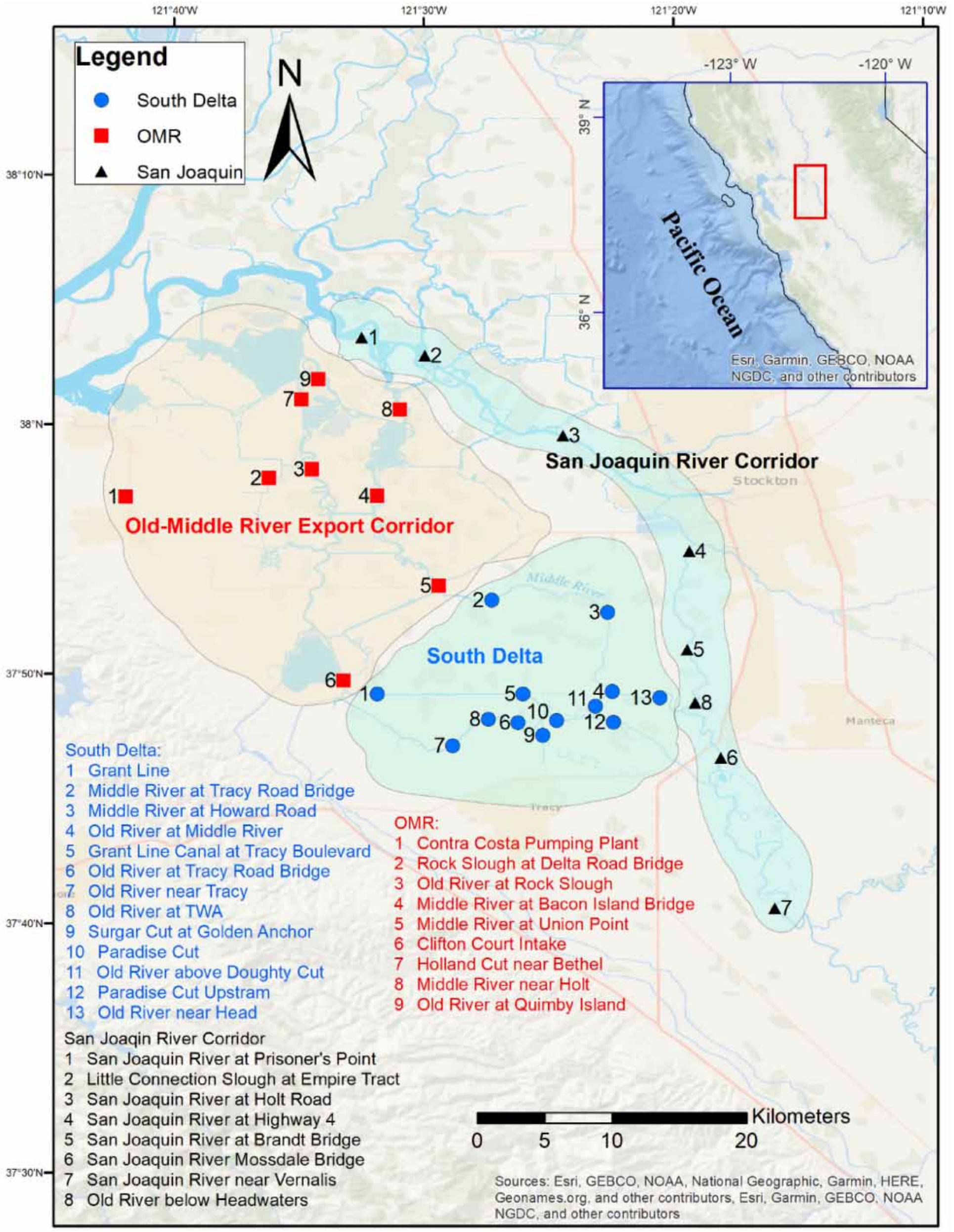

121°40'W
121°30'W
121°20'W
121°10'W
38°10'N
38°N
37°50'N
37°40'N
37°30'N
Legend
● South Delta
🟥 OMR
▲ San Joaquin
N
-123° W
-120° W
39° N
36° N
Pacific Ocean
Esri, Garmin, GEBCO, NOAA NGDC, and other contributors
San Joaquin River Corridor
Stockton
Old-Middle River Export Corridor
South Delta
Middle River
Manteca
Tracy
South Delta:
1 Grant Line
2 Middle River at Tracy Road Bridge
3 Middle River at Howard Road
4 Old River at Middle River
5 Grant Line Canal at Tracy Boulevard
6 Old River at Tracy Road Bridge
7 Old River near Tracy
8 Old River at TWA
9 Surgar Cut at Golden Anchor
10 Paradise Cut
11 Old River above Doughty Cut
12 Paradise Cut Upstram
13 Old River near Head
San Joaqin River Corridor
1 San Joaquin River at Prisoner's Point
2 Little Connection Slough at Empire Tract
3 San Joaquin River at Holt Road
4 San Joaquin River at Highway 4
5 San Joaquin River at Brandt Bridge
6 San Joaquin River Mossdale Bridge
7 San Joaquin River near Vernalis
8 Old River below Headwaters
OMR:
1 Contra Costa Pumping Plant
2 Rock Slough at Delta Road Bridge
3 Old River at Rock Slough
4 Middle River at Bacon Island Bridge
5 Middle River at Union Point
6 Clifton Court Intake
7 Holland Cut near Bethel
8 Middle River near Holt
9 Old River at Quimby Island
Kilometers
0 5 10 20
Sources: Esri, GEBCO, NOAA, National Geographic, Garmin, HERE, Geonames.org, and other contributors, Esri, Garmin, GEBCO, NOAA NGDC, and other contributors

Figure 1 | Map showing 30 study stations in the interior Delta. *Note*: The inset map shows the location of the San Francisco Bay and Sacramento-San Joaquin Delta (Bay-Delta), containing the Delta study area (highlighted in the red rectangle).

To assemble the most comprehensive ion sample dataset in the Delta to date, this study combines grab samples from three sources. The primary dataset is derived from Hutton *et al.* (2022), which includes ion grab samples, EC, and X2 position collected between 1959 and 2018 at 19 stations within the study area (Figure 1). The second dataset consists of samples collected by the Department of Water Resources between 2018 and 2020 at seven stations in the South Delta sub-regions (Stations: 5, 8, 9, 10, 11, 12, 13) (Figure 1). The third dataset covers 13 stations in the interior Delta, with samples collected from 2018 to 2022 (Figure 1).

Incorporating the second and third datasets provides several advantages, including an increased sample size, data from a wider range of hydrologic conditions, and coverage of the most recent critical drought years (i.e., 2021–2022). Additionally, the combined dataset captures a large variation of ions in the Delta, allowing for validation and testing of the models under more extreme conditions. Ultimately, this approach enhances the robustness of our models and improves their predictive capabilities. The final dataset for this study contains ion data spanning from 1959 to 2022 and encompasses 30 locations in the study area.

Figures 2–4 display the relationships between ion constituents (*y*-axis) and EC (*x*-axis) based on the augmented dataset, demonstrating the range of EC and ion constituents covered in this study. The nine ion constituents are categorized into three groups according to their relationship with EC: (1) linear relationship (Mg^{2+} and TDS) (Figure 2), (2) bifurcation relationship (Na^+, Cl^-, and Ca^{2+}) (Figure 3), and (3) non-linear relationship (alkalinity, K^+, SO_4^{2-}, and Br^-) (Figure 4). Groups 2 (bifurcation relationship) and 3 (non-linear relationship) consist of ion constituents influenced by multiple sources, such as fresh water (Sacramento River), agricultural drainage (San Joaquin River), and seawater (ocean source). This comprehensive dataset ensures a thorough evaluation of the ML models and their performance in simulating ion constituents across various conditions in the Delta.

2.2. Model development

This study employs four non-parametric supervised ML techniques to estimate ion constituents based on the EC at the study stations: regression trees (RT), random forest (RF), gradient boosting (GB), and artificial neural network (ANN). The equations from Hutton *et al.* (2022) serve as benchmark models for comparison.

Due to the complexity of the Delta's channel network and bathymetry, as well as the varying impacts of ocean tides, channel diversions, island drainage, pumping, and San Joaquin River inflow on local hydrodynamics, the source of the water and the proportions of water quality constituents at each study location can differ significantly. Consequently, we utilize subregions as categorical variables (Old and Middle River (OMR), South Delta, and San Joaquin Corridor) within the input data for the ML models.

Since ML algorithms cannot directly process categorical variables, we employ one-hot encoding to convert the names of the three subregions into binary vectors. This encoding assigns a separate column to each subregion category, with 0 and 1 indicating the absence or presence of that category, respectively. This approach eliminates the arbitrary assignment of numerical values that could mislead the learning algorithm.

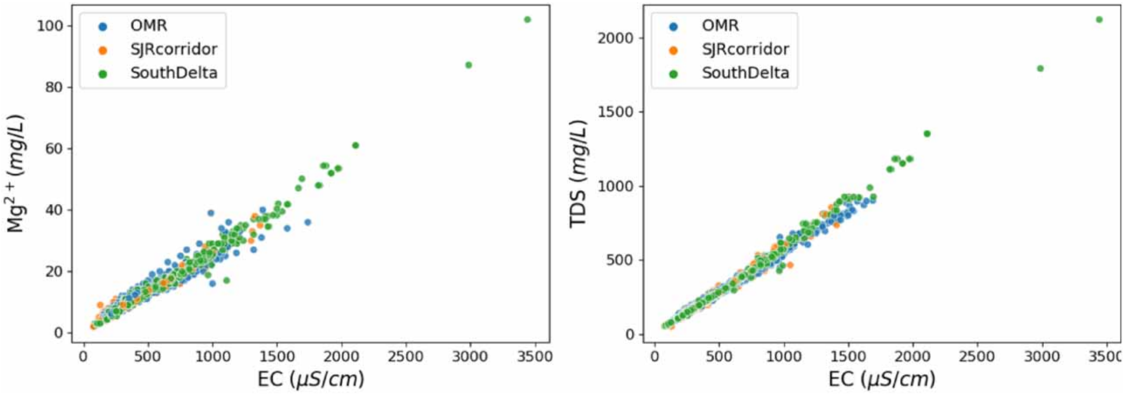


Figure 2 | Scatter plots showing the relationship between salinity (represented by EC) and ion constituents with linear relationship with EC (Group 1).

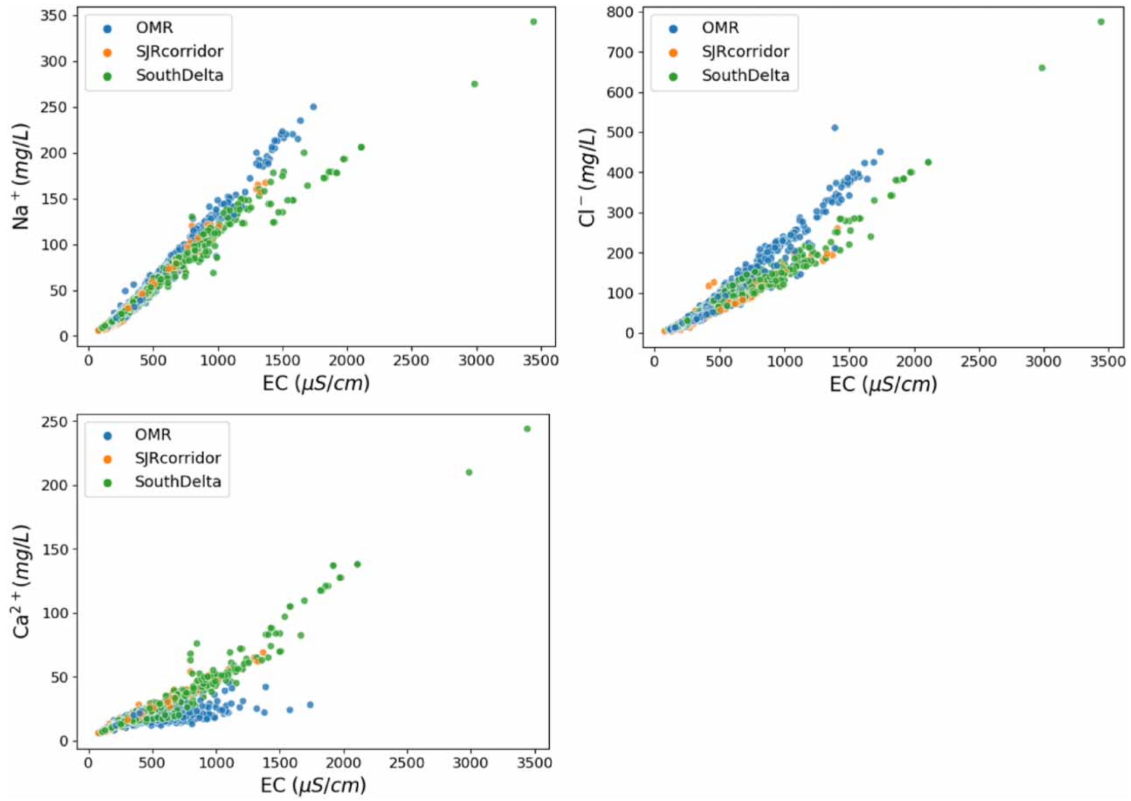


Figure 3 | Scatter plots showing the relationship between salinity (represented by EC) and ion constituents with bifurcation relationship with EC (Group 2).

Additionally, we include the month, water year type (WYT), and X2 position (when a specific sample was taken) as input features to evaluate their potential impacts on model outcomes. The Sacramento River X2 position is a water quality management indicator that represents the distance from the Golden Gate Bridge to the location where the salinity level reaches 2 parts per thousand (ppt). This indicator helps to monitor the fresh water–saltwater interface and its influence on the Delta's water quality. WYT is a classification based on water availability, which varies from year to year depending on factors such as precipitation and snowpack. WYT categories include wet, above normal, below normal, dry, and critical. These variables provide essential information about the hydrologic conditions affecting the study area.

Thus, our study consists of two numerical predictors (EC and X2 position) and three categorical predictors (location, month, and WYT), with nine targets (ion constituents). We maintain consistency with Hutton *et al.* (2022) by selecting the same predictors used in their study.

The input–output datasets are randomly divided into two groups for training (80% of the dataset) and testing (20% of the dataset). We evaluate the performance of the four ML models using two criteria: R^2 (Equation (1)) and mean absolute error (MAE) (Equation (2)). R^2 values range from 0 to 1, with values closer to 1 indicating that model simulations capture most of the variability in the observed data. MAE is a positive number, with values close to 0 signifying that the model-simulated values are very close to the observed values.

Using both R^2 and MAE allows us to assess the performance of the models from different perspectives. R^2 measures the proportion of variance in the observed data that can be explained by the model, while MAE provides a direct measure of the average error between observed and predicted values. Considering both criteria enables us to ensure that the selected model captures the overall data trends (as indicated by a high R^2 value) while also minimizing the average error between observed and predicted values (as indicated by a low MAE value). This comprehensive evaluation helps to identify the most accurate and reliable model for predicting ion constituents in the Delta.

The following sections provide a brief overview of the non-parametric supervised ML techniques used in this study. Detailed descriptions of each technique, along with the process of model development, training, and validation, are presented

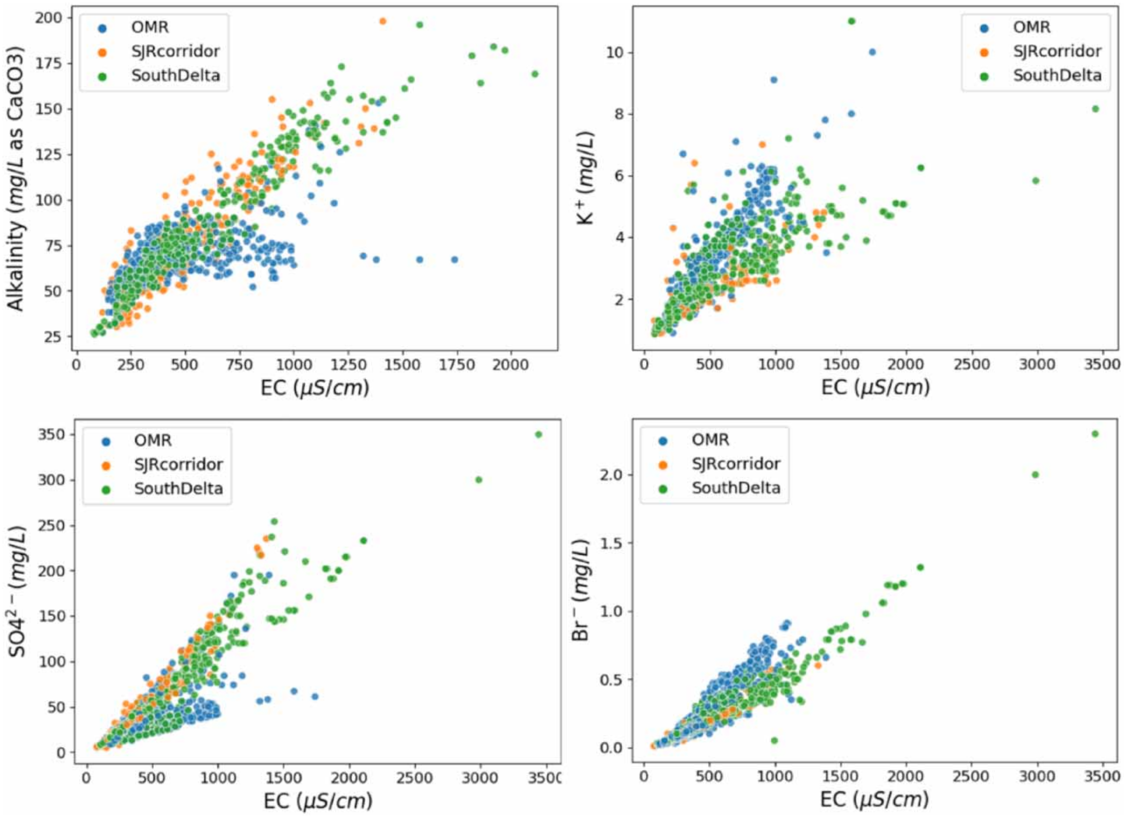


Figure 4 | Scatter plots showing the relationship between salinity (represented by EC) and ion constituents with non-linear relationship with EC (Group 3).

in the methodology section. By comparing the performance of these models using R^2 and MAE, we aim to identify the most suitable approach for predicting ion constituents in the Delta, ultimately contributing to improved water quality management and informed decision-making.

$$R^2 = 1 - \frac{\text{SSE}}{\text{SSTotal}} \tag{1}$$

where SSE is the sum of squared error (or residuals). SSE $= \sum_i (y_i - \hat{y}_i)^2$; SSTotal is the sum of squared deviations from the mean $\bar{y}$ (total variation of y without model adjustment). SSTotal $= \sum_i (y_i - \hat{y}_i)^2 + \sum_i (\hat{y}_i - \bar{y})^2$, where y_i is observed values; $\hat{y}_i$ is simulated values; and $\bar{y}$ is the mean of observed values.

$$\text{MAE} = \frac{\sum_{i=1}^{n} |\hat{y}_i - y_i|}{n} \tag{2}$$

2.2.1. Decision trees

Decision trees are popular ML methods that can be applied to both regression and classification problems. This method stratifies the predictor space into several rectangular regions and assigns the mean of each region to all observed data included in that specific region (Loh 2011; James *et al.* 2013). Tree-based ML models are useful for interpretation, as their results indicate the importance of predictors. The split points, which are specific values where the tree decides to divide the data into different paths or branches, suggest the best threshold for each predictor.

To generate each decision tree, the first step is to identify the best predictor and cutpoint at each node of the tree. This is done by evaluating all possible splits of the data based on each predictor and selecting the split that maximizes the separation between the outcomes of the response variable. The model implements the recursive binary splitting method that splits the dataset into two new branches. The decision tree considers all predictors and all possible cutpoints for each predictor, and then chooses the predictor and cutpoints for which the residual sum of squares (RSS) is the minimum. Equation (3) shows the RSS criteria to be minimized at each splitting point, where R_1 and R_2 are the two new branch regions after each splitting process, j is the predictor indicator, and S is the cutpoint. y_i represents the targets, which are ion constituents in this study. The individual functions of the Regression Tree (RT) model were determined using the 'sklearn' library in the Python environment.

$$\mathrm{RSS} = \sum_{i:x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2 \tag{3}$$

2.2.2. Random forest

Random Forest (RF) is an ensemble learning method that has demonstrated strong predictive performance in addressing a wide range of classification and regression analysis problems (Breiman 2001; Liaw & Matthew 2002). One of the key features of RF is the use of the bootstrap technique, which is a resampling method that helps reduce the variance of statistical learning methods.

Bootstrap works by creating multiple samples of the original dataset by randomly selecting observations with replacement. This means that each sample can have multiple copies of the same observation, and some observations may be left out altogether. By creating multiple samples, the bootstrap technique helps to create variability in the data, which in turn helps to reduce the variance of the statistical learning method (Tibshirani & Efron 1993). This allows for the production of new populations from the primary population by resampling data (James *et al.* 2013).

RF combines the results of all decision trees that were produced using the bootstrapping technique. In other words, if β separate training datasets were produced by the bootstrapping method, $\hat{f}^1(x)$, $\hat{f}^2(x)$, $\ldots$, $\hat{f}^\beta(x)$ will be the result of each decision tree. Equation (4) shows the final result of the RF method, the average of all decision trees, which generates a single low-variance statistical learning model with higher accuracy.

$$\hat{f}_{\mathrm{avg}}(x) = \frac{1}{\beta}\sum_{b=1}^{\beta} \hat{f}^b(x) \tag{4}$$

In addition to bootstrapping, RF also incorporates the concept of feature randomness. At each split in a decision tree, a random subset of features is considered, which further increases the diversity of individual trees and reduces overfitting (Cutler *et al.* 2007). This combination of bootstrapping and feature randomness results in a robust and accurate ensemble model. The individual functions of the RF model were determined using the 'sklearn' library in the Python environment.

2.2.3. Gradient boosting

Gradient Boosting (GB) uses an ensemble of models instead of a single model for some particular learning tasks. The ensemble approach relies on combining many weak simple models to obtain a stronger ensemble model (Natekin & Knoll 2013). One common example of an ensemble model is the aforementioned RF model (Breiman 2001), but the boosting method is based on a different strategy as adapted in the RF model. The main idea behind boosting models is a sequential process. In the first iteration of this process, one weak ML model, which is a regression tree in our study, is developed, and in the next iterations, a new weak learner model will train with respect to the error of all weak learner models in the previous iterations. The combination of the boosting algorithm and the gradient descent formulation used in the iteration process is called gradient boosting (Freund & Schapire 1997; Friedman *et al.* 2000; Friedman 2001). Equation (5) shows the formulation of the GB algorithm that aims to minimize the loss function $\Psi(y, f)$. The individual functions of the GB model were determined by using the '*sklearn*' library in the Python environment.

$$y = \arg\min \Psi(y, f(x)) = \arg\min E_x[E_y(\Psi[y, f(x)])|x] \tag{5}$$

where $E_y(\Psi[y, f(x)])$ is expected y loss; and $\min E_x[E_y(\Psi[y, f(x)])|x]$ is expectation over the whole dataset.

2.2.4. Artificial neural network

Artificial intelligence-based neural network (ANN) models have emerged as popular predictive tools in various domains, providing valuable insights for model identification, analysis, and forecasting. The ANN's effectiveness is largely due to its ability to model non-linear relationships between dependent and independent variables, which is particularly beneficial when dealing with complex real-world problems (Hopfield 1988; Zhang *et al.* 2015). Over the years, ANNs have found applications in diverse fields such as finance, environmental modeling, healthcare, and engineering (Gurney 1997; Zhang *et al.* 1998; Maier & Dandy 2000). The power of ANNs stems from their structure, which is inspired by the human brain's neural network. This allows them to learn and adapt their internal parameters to improve their predictive performance iteratively. The learning process involves adjusting the weights and biases within the network to minimize the error between the predicted and actual outputs (Haykin 1998; Bishop & Nasrabadi 2006). As a result, ANNs can uncover hidden patterns and relationships in data that might be missed by traditional linear regression or other ML techniques (Cybenko 1989; Hornik *et al.* 1989).

Moreover, ANNs have demonstrated the ability to handle noisy or incomplete data, making them particularly suitable for modeling complex systems where data quality may be an issue (Gardner & Dorling 1998; Karlik & Olgac 2011). Furthermore, ANNs have the advantage of being universal function approximators, meaning that they can theoretically approximate any continuous function to a desired level of accuracy, given an appropriate network structure and sufficient training data (Hornik *et al.* 1989). In summary, the versatility, robustness, and adaptability of ANNs have led to their widespread adoption as powerful predictive models for tackling a broad range of classification and regression analysis problems across various fields.

A typical ANN model consists of three primary layers: an input layer, one or more hidden layers, and an output layer. In this study, the ANN model was designed with six layers: an input layer, four hidden layers, and an output layer. The input layer contains five input variables: EC, X2 position, location, month, and WYT. To find the optimal ANN structure for each ion constituent, a random hyperparameter search was performed in the Python environment. As a result, all models have four hidden layers, but the number of neurons and activation functions differ among the ion constituents. Figure 5 illustrates the general architecture of the ANN, detailing the arrangement of input, hidden, and output layers. The ANN model was implemented using the open-source TensorFlow library in the Python environment (Abadi *et al.* 2016). This allowed for efficient experimentation with various configurations and hyperparameters, ultimately leading to the selection of the best-performing ANN models for each ion constituent.

After detailing the individual characteristics, configurations, and justifications for employing each of the four ML models – DT, RF, GB, and ANN – we present a comparative summary table (Table 1). This table succinctly encapsulates the key features of each model, offering a side-by-side view to highlight their unique attributes and commonalities. By examining the table, readers can gain an overview of the models' underlying architectures, loss functions, learning algorithms, scalability, and other relevant features, thereby providing a comprehensive understanding of the tools used in our study.

2.3. K-fold cross-validation

After finalizing the ion constituent simulation models using various ML techniques, we compared their performance based on two evaluation criteria: R^2 and MAE. Based on these criteria, we selected the best-performing model for our study. To further validate the performance and robustness of our chosen model, we employed the K-fold cross-validation method.

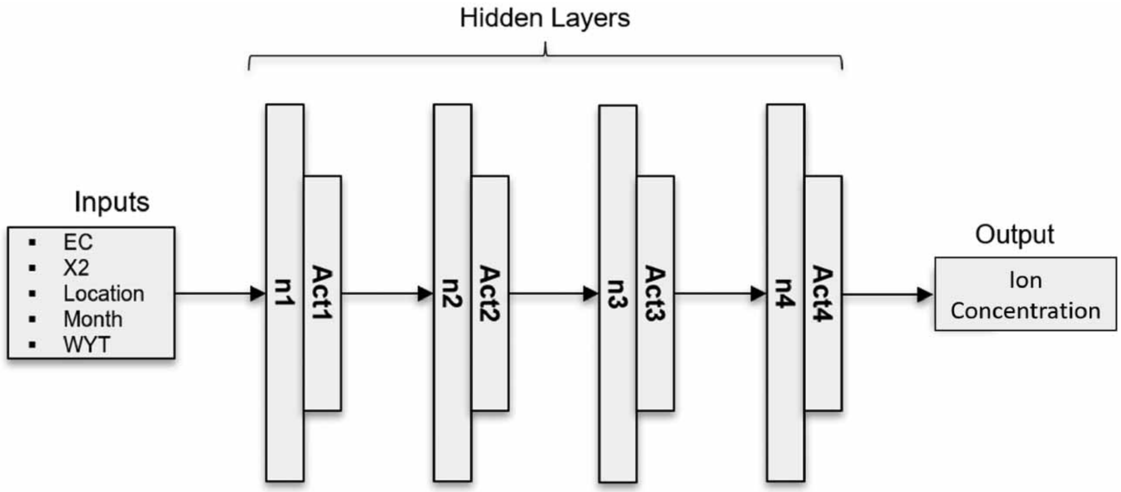


Figure 5 | Artificial neural network architecture.

Table 1 | Comparative overview of key features across selected ML models

Feature/Model	Decision trees	Random forests	Gradient boosting	Artificial neural networks
Model type	Tree-based	Ensemble	Ensemble	Neural network
Basic unit	Decision tree	Decision trees	Weak learners	Neurons
Hidden layers	None	None	None	One or more
Loss function	Gini/Entropy	Gini/Entropy	Various	MSE, cross-entropy, etc.
Learning algorithm	ID3, CART, etc.	Bagging	Boosting	Gradient descent, Adam, etc.
Regularization	Pruning	Voting/Averaging	Shrinkage	Dropout, weight decay, etc.
Scalability	Moderate	High	Moderate to high	High
Robustness	Moderate	High	High	Varies
Interpretability	High	Moderate	Low	Low
Speed/Efficiency (Training)	Fast	Moderate	Moderate	Varies
Speed/Efficiency (Inference)	Fast	Fast	Fast	Fast
Applications	Classification, regression	Classification, regression, anomaly detection	Classification, regression, ranking	Classification, regression, NLP, image processing

K-fold cross-validation is a widely used technique for assessing the performance of a model in ML and statistical modeling. It involves dividing the original dataset into K equal-sized subsets or 'folds' and then iteratively training the model on K–1 folds while using the remaining fold as a validation set. The process is repeated K times, with each fold serving as the validation set exactly once (Stone 1974; Geisser 1975; Efron 1983). In this example, we have used $K = 5$, dividing the dataset into five equal-sized subsets. These subsets were created randomly to ensure each fold was a good representation of the whole dataset, and we used a fixed random seed for reproducibility. Each fold represents 100% of the data, with 20% of the data being used for testing and 80% for training in each iteration. For each iteration, four of these subsets are used as the training set, and the remaining subset is used as the validation set. We opted for random K-fold cross-validation as opposed to stratified K-fold because our dataset did not exhibit significant imbalances in the distribution of target variables that would necessitate stratification.

The main advantage of K-fold cross-validation is that it allows us to assess the model's performance on different subsets of the data, providing a more comprehensive understanding of its generalization capability. This helps to prevent overfitting and ensures that the model is not biased toward a specific subset of the data. Since the model is tested on the entire dataset throughout the K iterations, it provides a more reliable performance estimation (Hastie *et al.* 2009).

In our study, we used $K = 5$, meaning that we divided our dataset into five equal-sized subsets. For each iteration, four of these subsets were used as the training set, and the remaining subset was used as the validation set. After completing all five iterations, we averaged the performance metrics (R^2 and MAE) across the five validation sets to obtain the final performance estimation of our selected model.

2.4. Dashboard

In this study, we have developed an interactive and user-friendly dashboard to provide a convenient tool for users with or without programming knowledge to simulate ion levels in the Delta using our ML models. The dashboard allows users to explore the results of the four ML simulators (RT, RF, GB, and ANN) for nine ion constituents based on the selected hydrological conditions. Users can interactively input the five predictor variables (EC, Sacramento X2, Location, Month, and WYT) to generate simulations of ion levels from the four ML models.

To access the dashboard, users can navigate to the following URL using a web browser: https://dwrdashion.azurewebsites.net/Dashboard. A step-by-step guide for using the dashboard is also available at the website to assist first-time users in effectively leveraging its features.

Our pre-trained models are stored on a GitHub repository, and their functionality is made available through Microsoft Azure. By connecting the Azure server to the GitHub repository, the models are hosted and executed on the Azure server, ensuring seamless integration and accessibility.

The development of the interactive and user-friendly dashboard offers several advantages that make it a valuable resource for stakeholders, researchers, and decision-makers. Some of these advantages include:

- Accessibility: The dashboard is designed to be accessible to users with various levels of technical expertise, removing the barrier of programming knowledge and allowing a wider audience to benefit from the simulation results.
- Real-time results: By leveraging the power of Microsoft Azure, the dashboard provides real-time results based on user inputs, enabling users to explore different scenarios and assess the impacts of various hydrological conditions on ion levels in the Delta.
- Scalability: Microsoft Azure offers robust scalability, allowing the dashboard to handle a large number of users and data inputs without compromising performance or reliability. This feature ensures that the tool remains available and responsive even during periods of high demand (Microsoft 2021).
- Ease of maintenance and updates: Hosting the pre-trained models on GitHub and utilizing Microsoft Azure for the dashboard makes it easier to maintain and update the models as new data becomes available or as improvements are made to the algorithms. This ensures that users have access to the latest and most accurate information at all times.
- Data security and privacy: Microsoft Azure is a secure and reliable platform that adheres to strict security standards, ensuring that user data and the models are protected and confidential information is not compromised (Microsoft 2021).
- Collaboration: The dashboard provides a common platform for stakeholders, researchers, and decision-makers to collaborate and share insights, fostering a data-driven approach to understanding and addressing the challenges faced in managing the Delta's water system.

In summary, the dashboard, in combination with Microsoft Azure, offers a powerful, accessible, and scalable solution for simulating ion levels in the Delta, promoting data-driven decision-making, and facilitating collaboration among various stakeholders.

3. RESULTS

This section first presents the performance of the equations developed by Hutton *et al.* (2022) in simulating nine ion constituents at 30 locations in the Delta. The performance of the proposed models is evaluated next.

3.1. Simulation of ion constituents using the benchmark model

The performance of the benchmark model on nine ion constituents is evaluated using two metrics, R^2 and MAE (Table 2). The ion constituents are divided into three groups (Group 1, Group 2, and Group 3), as mentioned in the methodology section. The number of samples (sample size) and data range for each ion are provided in Table 2. The standard deviation (SD) for each ion is given, which represents the variability or dispersion of the ion concentration values in the dataset.

The sample size for each ion constituent varies, with alkalinity having the minimum sample size of 1,039 grab samples and Cl-having the maximum sample size of 1,972 samples. The benchmark model, based on the equations by Hutton *et al.* (2022),

Table 2 | Performance of the benchmark model in simulating ion constituents in the Delta

Group	Ion	Sample size	Data range	SD	R^2	MAE
Group 1	TDS	1,466	49–2,120	204	0.99	12.7
	Mg^{2+}	1,336	2–102	8.6	0.96	1.24
Group 2	Na^{+}	1,575	6–343	44	0.94	4.77
	Ca^{2+}	1,335	5.8–244	18	0.87	3.31
	Cl^{-}	1,972	4–775	77	0.92	10.26
Group 3	SO_4^{2-}	1,066	5–350	46.5	0.52	14.61
	Br^{-}	1,239	0.01–2.3	0.22	0.9	0.04
	Alkalinity	1,039	26–198	27.6	0.79	9.52
	K^{+}	1,148	0.87–11	1.35	0.62	0.51

demonstrates the best performance for ion constituents in Group 1, which have a strong linear relationship with EC. The performance of the model decreases for ion constituents in Groups 2 and 3.

In particular, the R^2 values for SO_4^{2-} and K^+ are 0.52 and 0.62, respectively, suggesting that there is significant room for improvement in the estimation of these two constituents. Although the R^2 value for TDS is quite high at 0.99, it does not necessarily imply that the model is near perfect. The second metric, MAE, is 12.7 milligrams per liter (mg/l), which is not negligible. One of the objectives of this study is to decrease the MAE difference between observed and simulated values, even for ion constituents yielding high R^2 values in Hutton *et al.*'s (2022) study.

3.2. Simulation of ion constituents via proposed models

This section assesses the performance of four alternative ion simulation models and aims to compare their performance with each other in order to select the best model. Once the best model is identified, its performance is compared against the benchmark Hutton *et al.*'s (2022) model. The generalization performance of a model developed via an ML method is based on its ability to predict test data not used in training. Assessment of this performance is crucial for selecting the most suitable model and measuring its usefulness. Test error, which is the model prediction error over a test sample of data not used in training, serves as a key metric in this evaluation. One of the best approaches for training and testing a model is to randomly divide the data into two parts: training data and test data. The training data are used to fit or develop the models, while the test data are used to assess the model generalization error by comparing simulated ion concentrations to observed values not used in the model development. This approach helps mitigate the risk of overfitting, ensuring that the chosen model provides meaningful results for a variety of conditions.

In this study, a random hyperparameter search was performed to optimize the ANN models for each ion constituent. The results of this search, presented in Table 3, show the selected number of neurons (N) and activation functions (Act) for each ion constituent model. Interestingly, the number of neurons in the ANN models increases for ion constituents belonging to Group 2 and further increases for Group 3. This finding suggests that as the non-linearity and complexity of the ion constituent models increase, the ANN models require more neurons to accurately simulate these constituents. This observation aligns with the expectation that more complex relationships between input and output variables necessitate a higher number of neurons in the hidden layers of the ANN models, providing better performance in capturing non-linear relationships.

Table 3 | Optimal number of neurons and activation functions for each ion constituent model

	TDS		**Mg^{2+}**		**Na^+**	
Hidden layer	**N**	**Act**	**N**	**Act**	**N**	**Act**
1	30	elu	30	relu	30	tanh
2	30	sigmoid	30	elu	30	elu
3	30	elu	30	tanh	30	sigmoid
4	30	relu	30	relu	30	elu
	Ca^{2+}		**Cl^-**		**SO_4^{2-}**	
Hidden layer	**N**	**Act**	**N**	**Act**	**N**	**Act**
1	40	elu	30	relu	44	relu
2	40	sigmoid	30	elu	44	relu
3	40	relu	30	simoid	44	relu
4	30	tanh	30	elu	22	relu
	Br^-		**Alkalinity**		**K^+**	
Hidden layer	**N**	**Act**	**N**	**Act**	**N**	**Act**
1	44	elu	30	tanh	44	relu
2	44	sigmoid	30	relu	44	relu
3	30	elu	30	tanh	44	relu
4	30	tanh	30	elu	22	relu

We evaluated the performance of four ML models, namely RT, RF, GB, and ANN, for simulating ion constituents in three different groups. The results are presented in Figures 6–8 and Supplementary Appendix Figures A.1–A.6, where each figure contains four panels (a), (b), (c), and (d), corresponding to the Regression Trees (RT), Random Forest (RF), Gradient Boosting (GB), and Artificial Neural Networks (ANN) models, respectively. The *x*-axis of each panel represents the observed data, while the *y*-axis depicts the simulated data. A dashed line in each graph indicates a perfect 1:1 relationship between observed and simulated values, highlighting a model that simulates the observations without any errors. In these panels, red points represent the training samples, and blue points denote the test dataset. Additionally, a table within each panel displays the model's performance for both training and test datasets based on R^2 and MAE metrics.

Overall, the results demonstrate that the ANN model outperforms the other models in simulating Group 1 constituents (TDS and Mg^{2+}), Group 2 constituents (Na^+, Ca^{2+}, and Cl^-), and two constituents in Group 3 (Br^- and SO_4^{2-}), based on both R^2 and MAE values during testing. For alkalinity and K^+ in Group 3, the RF model exhibited slightly better performance. These findings indicate the potential of ANNs in improving the estimation of ion concentrations in various groups when compared to traditional RT, RF, and GB models.

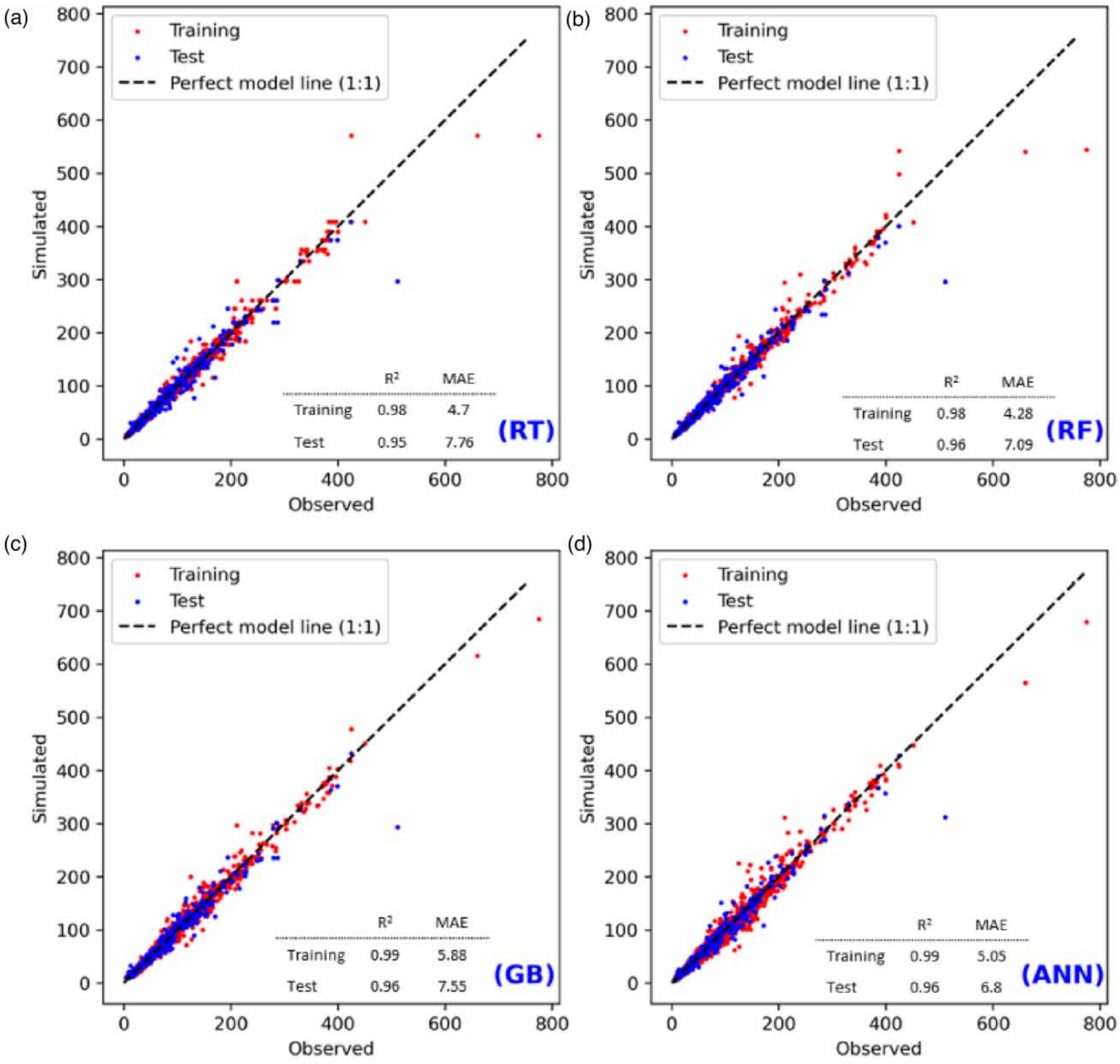


Figure 6 | Performance of four alternative models (train for 80% and test for 20% of samples) to simulate Cl^-. RT, regression trees; RF, random forest; GB, gradient boosting; ANN, artificial neural network.

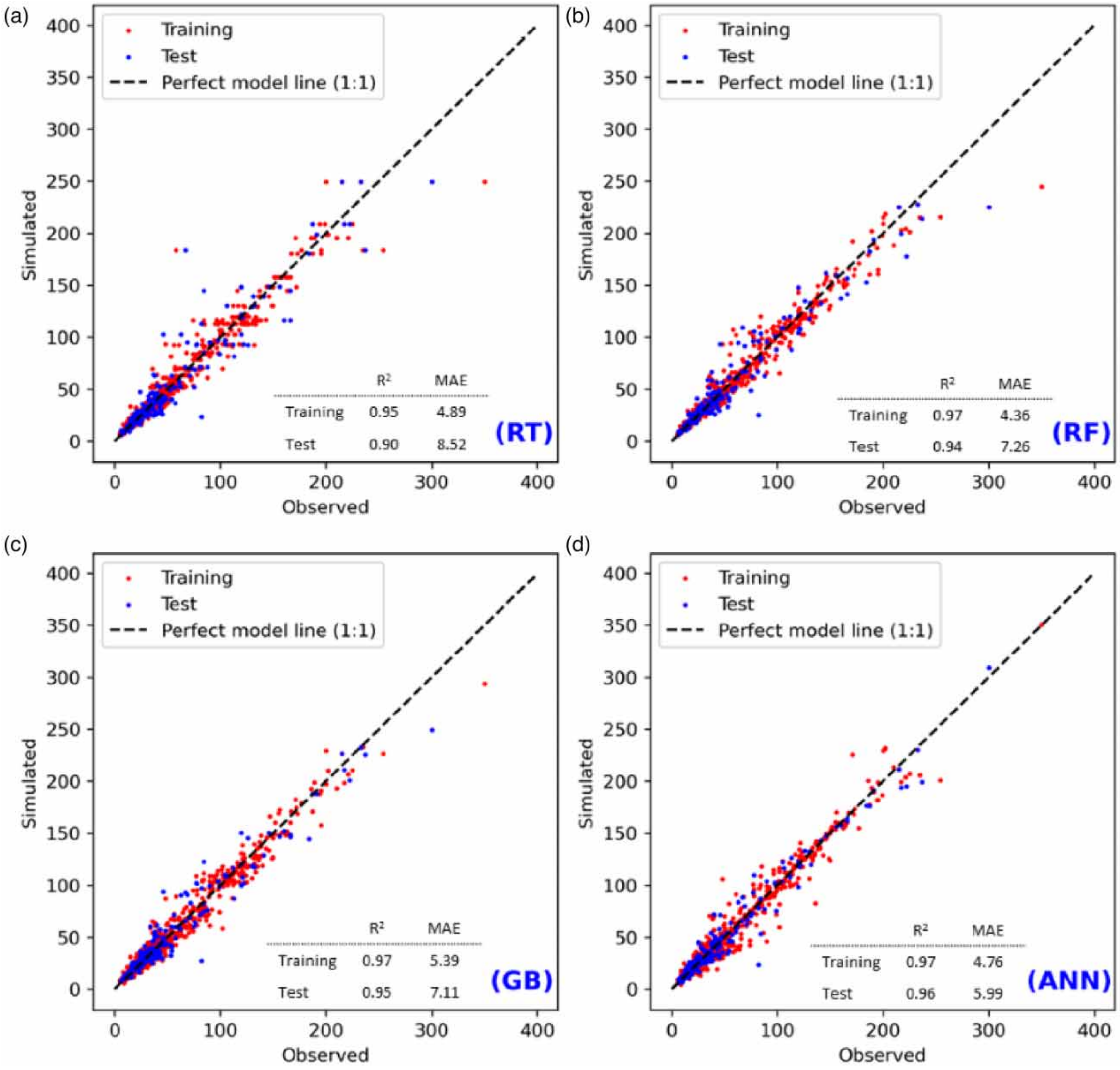


Figure 7 | Performance of four alternative models (train for 80% and test for 20% of samples) to simulate SO_4^{2-}.

The K-fold cross-validation analysis results further support the effectiveness of the selected ANN models. The average R^2 and MAE values obtained from the 5-fold cross-validation demonstrate the excellent performance of these models. Figure 9 presents box plots illustrating the distribution of MAE values for all nine ion constituents across the 5-fold cross-validation. In this figure, stars indicate the performance of the benchmark model based on MAE.

A comparison of the box plots with the stars reveals that the ANN models consistently outperform the benchmark model in terms of MAE during the K-fold cross-validation iterations. This finding underscores the superiority of the ANN models in simulating ion concentrations more accurately compared to the benchmark model, further highlighting the potential of ANNs in enhancing the estimation of ion concentrations in various groups.

Figure 10 presents the comparison between the benchmark models and the average of K-fold iterations (using the K-fold results instead of the single model results to ensure a fair comparison) for the ANN models. The first column displays the percentage improvement of the model based on R^2, and the second column shows the percentage decrease in MAE. The improvement in R^2 ranges between 0 and 85% for the nine ion constituents. Although the benchmark model (parametric regression equation) can yield fairly reasonable simulations for Groups 1 and 2 ion constituents, the ANN models can yield even better simulations with notably smaller errors (measured by MAE). In contrast, significant improvement is observed for ion constituents in Group 3 (SO_4^{2-}, Br^-, alkalinity, and K^+) when compared with the benchmark model. For

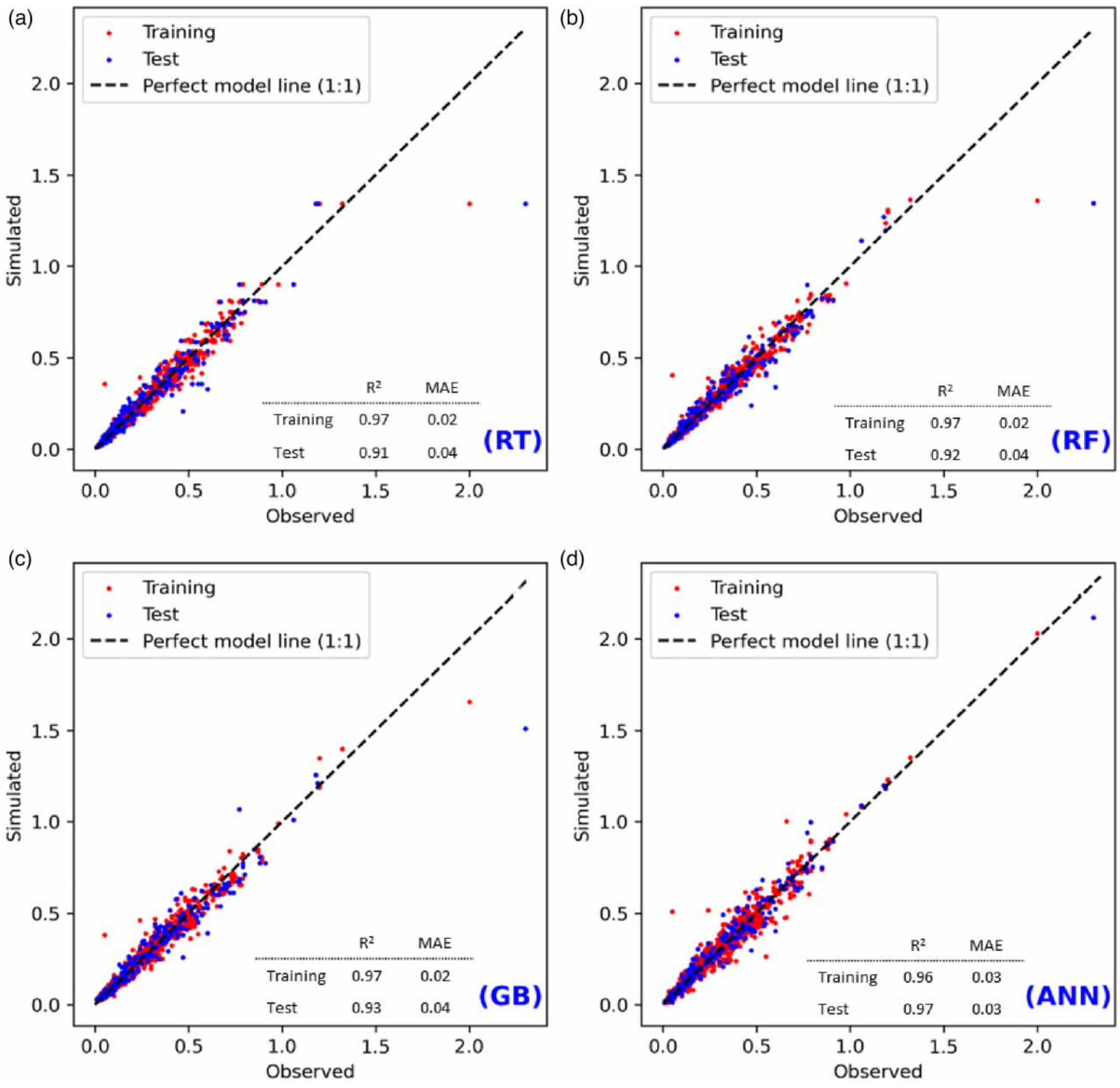


Figure 8 | Performance of four alternative models (train for 80% and test for 20% of samples) to simulate Br$^-$.

instance, the ANN model increases R^2 by 0% but reduces MAE by 24% over the benchmark model for TDS. Moreover, the ANN models improve MAE by 33, 32, 40, 34, 59, 25, 26, and 20% for Mg^{2+}, Na^+, Ca^{2+}, Cl^-, SO_4^{2-}, Br^-, alkalinity, and K^+, respectively. These observations demonstrate the superior performance of ANN models over the benchmark model, particularly for ion constituents in Group 3.

3.3. Ion simulator dashboard

Our study has also resulted in the development of a user-friendly dashboard, which effectively demonstrates the capabilities of the proposed ANN models and other ML models in simulating ion constituents in the Delta region. Figure 11 presents a snapshot of the dashboard interface, showcasing its clean and intuitive design, allowing users to easily explore the results of the four ML simulators for nine ion constituents.

The dashboard features dropdown menus, sliders, and other interactive elements that allow users to easily customize their queries. For instance, the EC value can be adjusted via a slider, while the location and WYT can be selected from dropdown lists. After the desired inputs are selected, the user can click a 'Compute' button, and the ion concentration predictions for each of the four models will be displayed in graphical form, as preferred by the user.

The dashboard's interactive features enable users to adjust input parameters and visualize the outcomes for different hypothetical hydrological conditions, comparing the performance of ANN, RT, RF, and GB models.

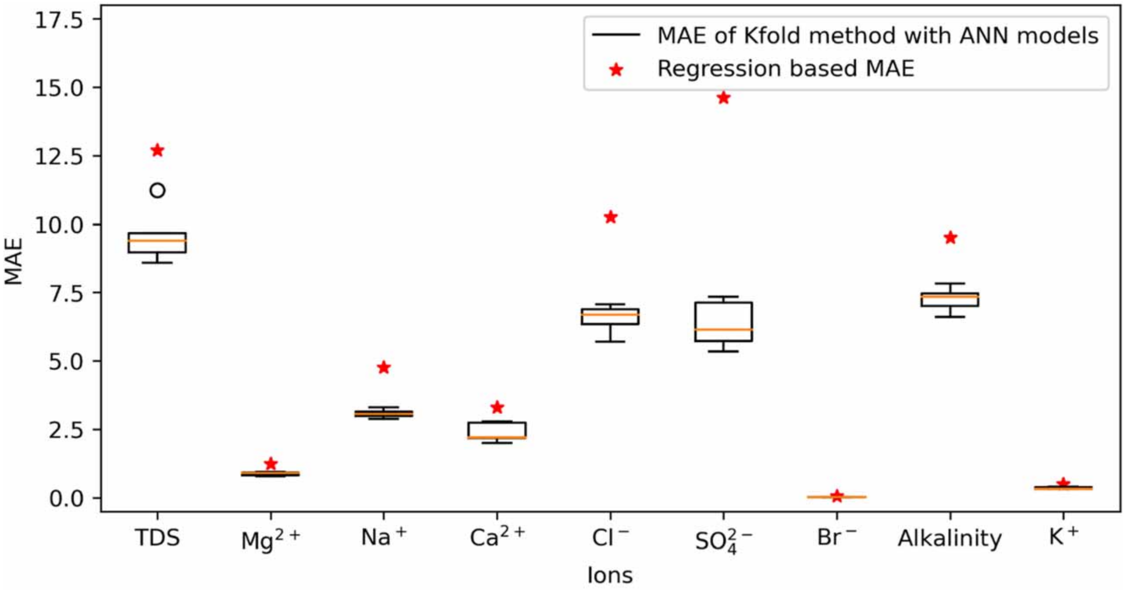


Figure 9 | MAE values for the nine ion constituents across the 5-fold cross-validation using the selected ANN models vs. MAE of benchmark model.

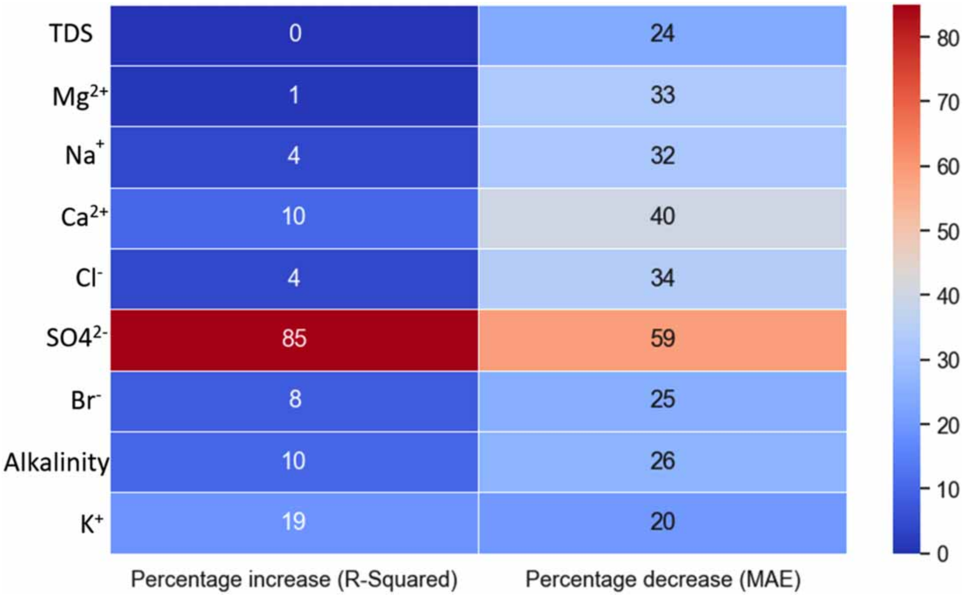
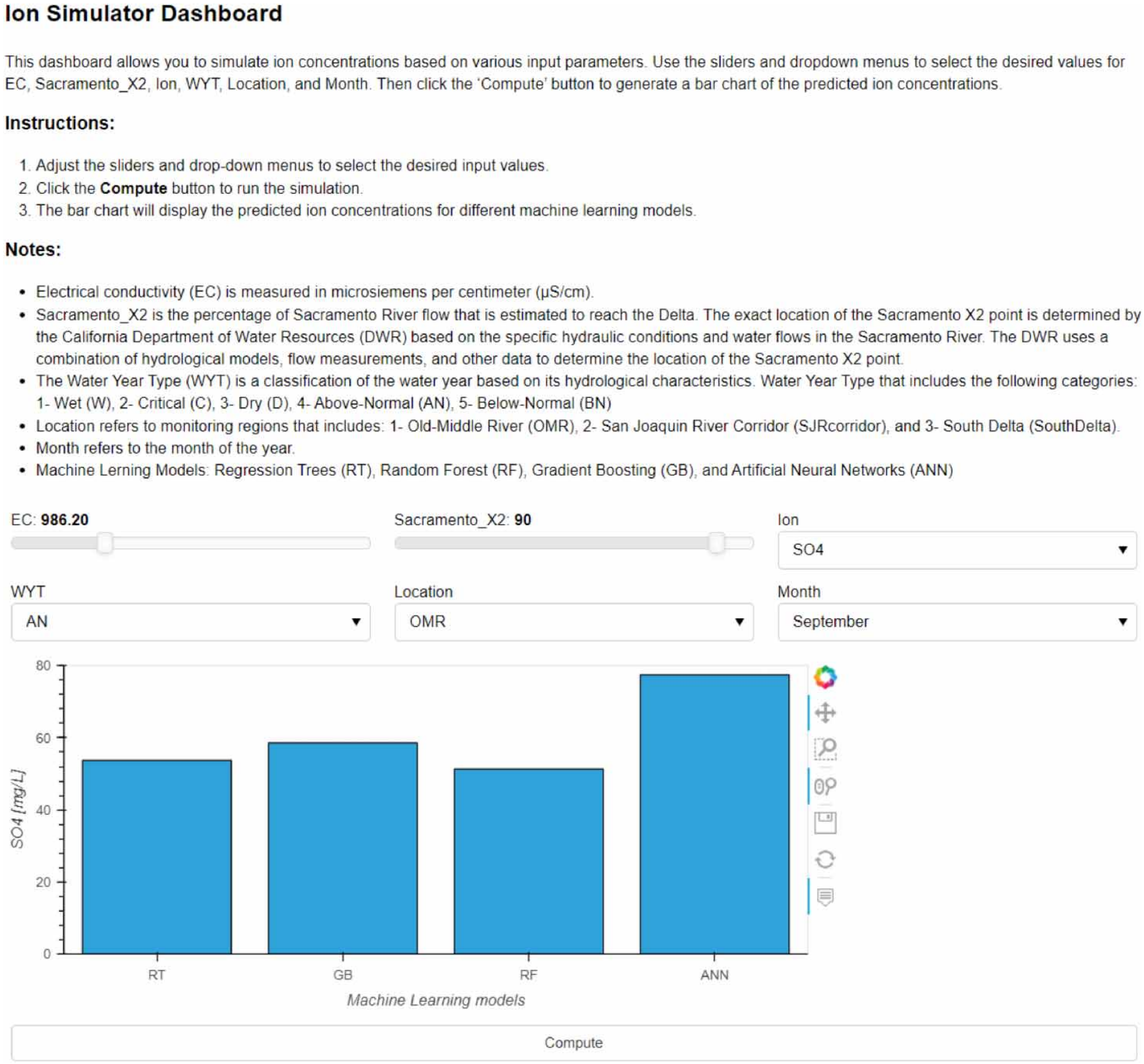


Figure 10 | ANN model performance on simulating the concentrations of nine ion constituents based on percent improvement from the benchmark model represented by R^2 and MAE.

4. DISCUSSION

Numerous studies have explored the use of parametric regression equations to predict ion constituents in various water bodies (Jung 2000; Suits 2002; Hutton 2006, Hutton *et al.* 2022; Denton 2015). Each successive study has built upon the previous work to improve the predictive performance of these equations. Hutton *et al.* (2022) represent the latest advancement in this line of research, offering the most accurate parametric regression models to date.

Contrary to previous studies that relied on parametric regression equations, our study found that ML models not only simplify the prediction process but also improve the accuracy of ion constituent estimates. Specifically, our models showed better



Ion Simulator Dashboard

This dashboard allows you to simulate ion concentrations based on various input parameters. Use the sliders and dropdown menus to select the desired values for EC, Sacramento_X2, Ion, WYT, Location, and Month. Then click the 'Compute' button to generate a bar chart of the predicted ion concentrations.

Instructions:

1. Adjust the sliders and drop-down menus to select the desired input values.
2. Click the **Compute** button to run the simulation.
3. The bar chart will display the predicted ion concentrations for different machine learning models.

Notes:

- Electrical conductivity (EC) is measured in microsiemens per centimeter (μS/cm).
- Sacramento_X2 is the percentage of Sacramento River flow that is estimated to reach the Delta. The exact location of the Sacramento X2 point is determined by the California Department of Water Resources (DWR) based on the specific hydraulic conditions and water flows in the Sacramento River. The DWR uses a combination of hydrological models, flow measurements, and other data to determine the location of the Sacramento X2 point.
- The Water Year Type (WYT) is a classification of the water year based on its hydrological characteristics. Water Year Type that includes the following categories: 1- Wet (W), 2- Critical (C), 3- Dry (D), 4- Above-Normal (AN), 5- Below-Normal (BN)
- Location refers to monitoring regions that includes: 1- Old-Middle River (OMR), 2- San Joaquin River Corridor (SJRcorridor), and 3- South Delta (SouthDelta).
- Month refers to the month of the year.
- Machine Lerning Models: Regression Trees (RT), Random Forest (RF), Gradient Boosting (GB), and Artificial Neural Networks (ANN)

EC: **986.20**

Sacramento_X2: **90**

Ion: SO4

WYT: AN

Location: OMR

Month: September

Figure 11 | Screenshot of the interactive dashboard interface, displaying the results of the four ML models (ANN, RT, RF, and GB) for simulating nine ion constituents in the Delta region.

performance for ions that have non-linear relationships with EC, addressing a significant limitation in existing models. This demonstrates the adaptability and robustness of ML algorithms in dealing with complex hydrological data.

In a previous study, we (Namadi *et al.* 2022) took the pioneering step of applying ML models to simulate ion constituents' levels in the South Delta, using a dataset that spanned from 2018 to 2020. Although that study laid important groundwork, its scope was limited, both in terms of geographic coverage and time span. The current study addresses these limitations and significantly expands on this foundational work. By integrating three datasets, we have been able to cover a more extensive range of stations and a prolonged period of time, thus ensuring a more representative sample of water quality conditions in the region. Specifically, we included data from 30 stations and extended the time span to 64 years, thereby providing a robust foundation for our analysis.

To simulate ion constituents, we employed four different ML models: RT, RF, GB, and ANN. The rationale behind testing four distinct algorithms lies in determining which model is best suited to address our research problem, as each model offers unique advantages. However, it is crucial to further elaborate on the limitations of these models for a more comprehensive understanding.

RTs are easily interpretable, as they provide a clear visualization of the decision-making process. However, they can suffer from overfitting, which may limit their generalization performance and pose a problem when applying these models to new, untested locations. RFs use an ensemble of decision trees to alleviate the overfitting problem associated with single RT, but their interpretability can be compromised due to the ensemble approach.

GB combines multiple weak learners to produce a strong learner. It offers improved predictive performance compared to individual trees but may be computationally intensive, making it less suitable for real-time applications where quick predictions are needed. ANNs are powerful models capable of capturing complex, non-linear relationships; however, they can be more difficult to interpret and require a substantial amount of data for training, limiting their usefulness in settings with limited data availability.

Moreover, all the models have inherent limitations when handling missing or imbalanced data, a frequent issue in hydrological studies. These limitations could potentially impact the reliability and applicability of our findings. Therefore, acknowledging these constraints not only provides a more balanced view of our study but also helps identify areas for future research and iterative model refinement.

In our evaluation, we used two performance metrics, R^2 and MAE, to assess the models' performance. The use of both metrics is essential because relying on just one of them can be misleading. While R^2 measures the proportion of the variance in the dependent variable that is predictable from the independent variables, it may not fully capture the model's accuracy, especially when the errors are large. On the other hand, MAE provides a more direct measure of the average magnitude of the errors, making it a valuable complementary metric to R^2 in determining the model's overall performance.

The results indicate that ANNs outperformed the other models for simulating Group 1 (TDS and Mg^{2+}) and Group 2 (Na^+, Ca^{2+}, and Cl^-) ion constituents, as well as Br^- and SO_4^{2-} in Group 3. ANN can generate comparable results to RF for alkalinity and K^+. One possible explanation for this is that ANNs are adept at capturing complex, non-linear relationships in data, a feature particularly useful for hydrological variables like ion concentrations that may not follow linear patterns. Unlike traditional regression models, ANNs can identify hidden layers of abstraction or features in the data, enabling more accurate predictions. Moreover, the architecture of the ANN allows for more intricate connections between variables, which could be critical in capturing the multi-faceted relationships in ion concentrations. The use of K-fold cross-validation, with $K = 5$, confirmed the robustness of the ANN model, demonstrating that the model is not overfitted and can generalize well to new data.

While the benchmark model provided satisfactory results for Groups 1 and 2, the ANN models demonstrated even better performance with notably smaller errors (measured by MAE). For Group 3 ion constituents (SO_4^{2-}, Br^-, alkalinity, and K^+), where the ANN models showed a remarkable improvement in performance compared to the benchmark model. Specifically, for these Group 3 ion constituents, the ANN model improved MAE by a range of 20–59%. This marked increase in performance emphasizes the distinct advantages of using ANN models over traditional parametric regression equations, particularly in capturing the complex, non-linear relationships between ion constituents and EC. The development of a user-friendly dashboard has made it possible for users with or without programming knowledge to interact with and visualize the results of the ML simulators. The dashboard, which is hosted on Microsoft Azure, offers a convenient way to explore hypothetical hydrological conditions and compare the results of different ML models for ion constituents.

Despite the promising results achieved in this study, there are some limitations that should be acknowledged. One of the main limitations is that the ML models developed in this study were trained for 30 different water quality stations in the Delta. Consequently, the applicability of these models is restricted to these specific stations. This limitation may hinder the utility of the models in predicting ion constituents in areas of the Delta not covered by the current dataset. To address this limitation, follow-up work is planned to expand the scope and applicability of the ML models. Specifically, we will develop and apply ML models to additional locations in the Delta, thus ensuring a more comprehensive understanding of water quality dynamics in the region. This will involve collecting new data and updating the models to accommodate the additional information, allowing for more accurate predictions in previously unexplored areas. Also, this may include the use of transfer learning techniques, which can adapt the models to new locations using a limited amount of new data.

Furthermore, it is essential to recognize that water quality conditions and their drivers may change over time due to various factors such as climate change, land use changes, or evolving water management practices. To ensure the continued relevance and accuracy of our models, we will regularly update them with the most recent data and evaluate their performance against emerging trends and conditions. To maintain this ongoing relevance, our update process will adhere to a structured framework. This involves initial data collection from reliable sources, preprocessing the acquired data, retraining the models with both old and new data, and conducting a rigorous testing phase for model validation. Finally, the updated models will be deployed to replace the older versions in the dashboard. By addressing these limitations in our follow-up work, we aim to provide even more valuable tools for understanding the impacts of salinity on water quality in the Delta and informing water management decisions that protect aquatic life and ensure the safety of drinking water supplies. For instance, water resource managers and policymakers can use the models and dashboard to simulate various water management scenarios and assess their impacts on ion constituent concentrations in the Delta. This will enable them to make well-informed decisions regarding the allocation of water resources and the implementation of water conservation measures to maintain optimal water quality and preserve aquatic ecosystems. Furthermore, public health officials can use the dashboard to monitor water quality in real-time and identify areas where ion concentrations exceed regulatory standards. This will allow for prompt action to protect public health by issuing advisories, implementing treatment processes, or taking other necessary measures to ensure the safety of drinking water supplies.

5. CONCLUSION

This study represents a paradigm shift in the approach to water quality modeling in the Delta region. By leveraging ML techniques, our research not only demonstrates equal or better performance compared to traditional parametric regression equations but also introduces a level of adaptability previously absent in the field. Particularly, the use of ANN and RF models showcased superior ability in simulating ion constituent concentrations, addressing the limitations of non-linear relationships that have constrained previous models.

A pioneering aspect of our work is the creation of an interactive dashboard accessible to users across various disciplines and levels of technical expertise. This tool serves as a nexus for data-driven, informed decision-making, significantly demystifying the complexities of water quality dynamics for stakeholders, researchers, and policymakers. This platform has immediate utility, providing real-time guidance for water management scenarios, thus potentially leading to more sustainable practices and better public health outcomes.

Our research marks a substantial contribution to water quality modeling by introducing ML as a robust alternative to traditional modeling techniques. It opens up exciting avenues for future work, such as extending ML models to other geographical locations within the Delta, refining models based on real-time data, and possibly incorporating additional variables like climate change factors. Such directions could lead to a more comprehensive, adaptable, and forward-looking approach to water quality management in the Delta and beyond. These advancements not only deepen our understanding of water quality in the Delta but also offer actionable insights for real-world applications, ranging from resource allocation to environmental conservation and public health protection.

ACKNOWLEDGEMENTS

We appreciate the support of the Municipal Water Quality Investigations (MWQI) program. Our sincere gratitude is extended to Tetra Tech, Inc. for generously providing the data essential for this study. We also wish to thank our colleagues in the Water Quality Evaluation Section of the North Central Region Office, whose diligent efforts in collecting the grab samples were invaluable to our research.

DATA AVAILABILITY STATEMENT

All relevant data are available from https://github.com/PeymanHNamadi/Ion_Study_Dashboard/tree/main.

CONFLICT OF INTEREST

The authors declare there is no conflict.

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irvin, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y. & Zheng, X. 2016 Tensorflow: a system for large-scale machine learning. In: *12th USENIX Symposium on Operating Systems Design and Implementation* (OSDI '16) (Vol. 16, No. 2016, pp. 265–283).

Attrill, M. J. & Rundle, S. D. 2002 Ecotone or ecocline: Ecological boundaries in estuaries. *Estuarine, Coastal and Shelf Science* **55** (6), 929–936.

Bañón, S., Álvarez, S., Bañón, D., Ortuño, M. F. & Sánchez-Blanco, M. J. 2021 Assessment of soil salinity indexes using electrical conductivity sensors. *Scientia Horticulturae* **285**, 110171.

Bishop, C. M. & Nasrabadi, N. M. 2006 *Pattern Recognition and Machine Learning*, Vol. 4, No. 4. Springer, New York, p. 738.

Breiman, L. 2001 Random forests. *Machine Learning* **45**, 5–32. https://doi.org/10.1023/A:1010933404324.

Cloern, J. E. & Jassby, A. D. 2012 Drivers of change in estuarine-coastal ecosystems: Discoveries from four decades of study in San Francisco Bay. *Reviews of Geophysics* **50** (4), RG4001.

Cloern, J. E., Foster, S. Q. & Kleckner, A. E. 2014 Phytoplankton primary production in the world's estuarine-coastal ecosystems. *Biogeosciences* **11** (9), 2477–2501.

Cutler, D. R., Edwards Jr, T. C., Beard, K. H., Cutler, A., Hess, K. T., Gibson, J. & Lawler, J. J. 2007 Random forests for classification in ecology. *Ecology* **88** (11), 2783–2792.

Cybenko, G. 1989 Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems* **2** (4), 303–314.

Denton, R. 2015 Delta Salinity Constituent Analysis. Richard Denton and Associates, prepared for the State Water Project Contractors Authority.

Efron, B. 1983 Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the American Statistical Association* **78** (382), 316–331.

Freund, Y. & Schapire, R. E. 1997 A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences* **55** (1), 119–139.

Friedman, J. H. 2001 Greedy function approximation: A gradient boosting machine. *Annals of Statistics* **29** (5), 1189–1232.

Friedman, J., Hastie, T. & Tibshirani, R. 2000 Additive logistic regression: A statistical view of boosting (with discussion and a rejoinder by the authors). *The Annals of Statistics* **28** (2), 337–407.

Gardner, M. W. & Dorling, S. R. 1998 Artificial neural networks (the multilayer perceptron)–A review of applications in the atmospheric sciences. *Atmospheric Environment* **32** (14–15), 2627–2636.

Geisser, S. 1975 The predictive sample reuse method with applications. *Journal of the American Statistical Association* **70** (350), 320–328.

Gurney, K. 1997 *An Introduction to Neural Networks*. CRC Press, London.

Hastie, T., Tibshirani, R., Friedman, J. H. & Friedman, J. H. 2009 *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Vol. 2. Springer, New York, pp. 1–758.

Haykin, S. 1998 *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, Englewood Cliffs.

Hopfield, J. J. 1988 Artificial neural networks. *IEEE Circuits and Devices Magazine* **4** (5), 3–1.

Hornik, K., Stinchcombe, M. & White, H. 1989 Multilayer feedforward networks are universal approximators. *Neural Networks* **2** (5), 359–366.

Hutton, P. 2006 Validation of DSM2 volumetric fingerprints using grab sample mineral data. In *Power Point Presentation at CWEMF Annual Meeting*.

Hutton, P. H., Rath, J. S., Chen, L., Ungs, M. J. & Roy, S. B. 2016 Nine decades of salinity observations in the San Francisco Bay and Delta: Modeling and trend evaluations. *Journal of Water Resources Planning and Management* **142** (3), 04015069.

Hutton, P., Sinha, A. & Roy, S. 2022 Simplified Approach for Estimating Salinity Constituent Concentrations in the San Francisco Estuary & Sacramento-San Joaquin River Delta. User guide, July 2022.

James, G., Witten, D., Hastie, T. & Tibshirani, R. 2013 *An Introduction to Statistical Learning*. Springer, New York, p. 112.

Jung, M. 2000 Revision of Representative Delta Island Return Flow Quality for DSM2 and DICU Model Runs. Prepared for the CALFED Ad-Hoc Workgroup to Simulate Historical Water Quality Conditions in the Delta by Marvin Jung and Associates, Inc. Consultant's Report to the Department of Water Resources Municipal Water Quality Investigations Program (MWQI-CR#3), December 2000.

Karlik, B. & Olgac, A. V. 2011 Performance analysis of various activation functions in generalized MLP architectures of neural networks. *International Journal of Artificial Intelligence and Expert Systems* **1** (4), 111–122.

Kemp, W. M., Boynton, W. R., Adolf, J. E., Boesch, D. F., Boicourt, W. C., Brush, G., Cornwell, J. C., Fisher, T. R., Glibert, P. M., Hagy, J. D., Harding, L. W., Houde, E. D., Kimmel, D. G., Hiller, W. D., Newell, R. I. E., Smith, E. M. & Stevenson, J. C. 2005 Eutrophication of Chesapeake Bay: Historical trends and ecological interactions. *Marine Ecology Progress Series* **303**, 1–29.

Liaw, A. & Matthew, W. 2002 Classification and regression by random forest. *R News* **2** (3), 18–22.

Loh, Y. 2011 Classification and regression trees. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* **1** (1), 14–23.

Maas, E. V. & Hoffman, G. J. 1977 Crop salt tolerance – Current assessment. *Journal of the Irrigation and Drainage Division* **103** (2), 115–134.

Maier, H. R. & Dandy, G. C. 2000 Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications. *Environmental Modelling & Software* **15** (1), 101–124.

Microsoft 2021 Scalability. Available from: https://azure.microsoft.com/en-us/overview/scalability/.

Namadi, P., He, M. & Sandhu, P. 2022 Salinity-constituent conversion in South Sacramento-San Joaquin Delta of California via machine learning. *Earth Science Informatics* **15** (3), 1749–1764.

Natekin, A. & Knoll, A. 2013 Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics* **7**, 21.

Nixon, S. W. 1981 Remineralization and Nutrient Cycling in Coastal Marine Ecosystems. In: Neilson, B.J., Cronin, L.E. (eds) *Estuaries and Nutrients. Contemporary Issues in Science and Society*. Humana Press, Totowa, pp. 111–138.

Rabalais, N. N., Turner, R. E. & Wiener, M. 2002 Gulf of Mexico hypoxia, aka 'The dead zone'. *Annual Review of Ecology and Systematics* **33** (1), 235–263.

Rath, J. S., Hutton, P. H., Chen, L. & Roy, S. B. 2017 A hybrid empirical-Bayesian artificial neural network model of salinity in the San Francisco Bay-Delta estuary. *Environmental Modelling & Software* **93**, 193–208.

Savenije, H. H. 2005 *Salinity and Tides in Alluvial Estuaries*. Gulf Professional Publishing, Delft.

Stone, M. 1974 Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society: Series B (Methodological)* **36** (2), 111–133.

Suits, B. 2002 Chapter 5, Relationships between Delta water quality constituents as derived from grab samples. In: *DWR's 'Methodology for Flow and Salinity Estimates in the Sacramento-San Joaquin Delta and Suisun Marsh'*. 23rd Annual Progress Report, June 2002.

Tibshirani, R. J. & Efron, B. 1993 An introduction to the bootstrap. *Monographs on Statistics and Applied Probability* **57** (1), 1–436.

US Environmental Protection Agency (US EPA) 2018 *National Recommended Water Quality Criteria – Human Health Criteria Table*. Available from: https://www.epa.gov/wqc/national-recommended-water-quality-criteria-human-health-criteria-table.

US Environmental Protection Agency (US EPA) 2021 *Drinking Water Standards and Regulations*. Available from: https://www.epa.gov/dwstandardsregulations.

World Health Organization (WHO) 2011 *Guidelines for Drinking-Water Quality*, 4th edn. World Health Organization, Geneva.

Zhang, G., Patuwo, B. E. & Hu, M. Y. 1998 Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting* **14** (1), 35–62.

Zhang, Z., Deng, Z., Rusch, K. & Walker, N. 2015 Modeling system for predicting enterococci levels at Holly Beach. *Marine Environmental Research* **109**, 140–147.

First received 22 June 2023; accepted in revised form 16 October 2023. Available online 31 October 2023