PLOS WATER

RESEARCH ARTICLE

Optimizing machine learning for water safety: A comparative analysis with dimensionality reduction and classifier performance in potability prediction

Debashis Chatterjee[1], **Prithwish Ghosh**[2]*, **Amlan Banerjee**[3], **Shiladri Shekhar Das**[3]

1 Department of Statistics, Visva Bharati, Santiniketan, India, **2** Department of Statistics, North Carolina State University, Raleigh, North Carolina, United States of America, **3** Geological Studies Unit, Indian Statistical Institute, Kolkata, West Bengal, India

* pghosh4@ncsu.edu

Citation: Chatterjee D, Ghosh P, Banerjee A, Das SS (2024) Optimizing machine learning for water safety: A comparative analysis with dimensionality reduction and classifier performance in potability prediction. PLOS Water 3(8): e0000259. https://doi.org/10.1371/journal.pwat.0000259

Editor: Daniel Reddythota, Faculty of Water Supply & Environmental Engineering, ArbaMinch Water Technology Institute (AWTI), ETHIOPIA

Received: March 16, 2024

Accepted: July 16, 2024

Published: August 8, 2024

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pwat.0000259

Data Availability Statement: All data and related metadata underlying the findings reported is already provided as part of the submitted article.

Abstract

In this study, we investigated the effectiveness of machine learning techniques in predicting water potability based on water quality attributes. Initially, we applied seven classification-based methods directly to the original dataset, yielding varying accuracy scores. Notably, the Support Vector Machine (SVM) achieved the highest accuracy of 69%, while other methods such as XGBoost, k-Nearest Neighbors, Gaussian Naive Bayes, and Random Forest demonstrated competitive performance with scores ranging from 62% to 68%. Subsequently, we employed Principal Component Analysis (PCA) to reduce the dataset's dimensionality to six principal components, followed by reapplication of the machine learning techniques. The results showed an increase in accuracy across all classifiers, increasing to nearly 100%. This study provides insights into the impact of dimensionality reduction on predictive accuracy and underscores the importance of selecting appropriate techniques for water potability prediction.

Introduction

Safeguarding public health necessitates maintaining high water quality, as emphasized by the World Health Organization (WHO) [1]. Contaminated water poses a significant threat through waterborne diseases, as highlighted by the Centers for Disease Control and Prevention (CDC) [2]. The World Bank further underscores the link between water security and socioeconomic progress, where safe drinking water is essential for economic development [3]. Similarly, the United Nations Environment Programme (UNEP) emphasizes the connection between water quality and environmental health [4]. Traditional water quality assessment methods often face limitations, motivating the exploration of machine learning for water potability prediction [5]. These limitations drive the development of water potability prediction models, offering a faster and more efficient approach [6]. Furthermore, the growing challenge of water scarcity makes water potability prediction even more crucial, as highlighted by the United Nations (UN-Water) (2023) [7]. Climate change further adds complexity, impacting water quality. Water potability prediction models play a vital role in adaptation strategies (US

Funding: The author(s) received no specific funding for this work.

Competing interests: The authors have declared that no competing interests exist.

Environmental Protection Agency, 2023) [8]. In developing regions with challenges in centralized water treatment, water potability prediction models can be valuable tools [9]. Similarly, [10] demonstrates the application of machine learning for rapid water quality assessment in disaster response situations, where water potability prediction is critical for ensuring the health and safety of affected communities.

By harnessing the power of machine learning, water potability prediction offers a promising approach to safeguarding public health, promoting environmental well-being, and fostering sustainable development across diverse contexts. Maintaining high water quality is crucial for safeguarding public health, preserving environmental integrity, and fostering socioeconomic progress [11]. Evaluating and predicting water quality is complex as it depends on numerous chemical, physical, and biological factors [12]. Machine learning methodologies have emerged as invaluable tools in water quality assessment. These techniques empower researchers to analyze intricate datasets and generate accurate water quality predictions [11].

The assessment and prediction of water quality are fundamental pillars in ensuring the potability of water sources [11]. This process aids in gauging the safety and suitability of water for human consumption, facilitating informed decisions regarding water treatment, and ensuring adherence to established quality standards [13].

Several studies have explored the application of machine learning for water quality prediction, recognizing its potential to improve public health and well-being (e.g., [14, 15]). These efforts often involve diverse machine learning algorithms and rigorous analysis to develop accurate predictive models. For instance, [14] employed a variety of algorithms and achieved high prediction accuracy. Similarly, [15] emphasized the importance of comprehensive evaluation and refinement to ensure the reliability of the proposed model for water safety monitoring. These studies highlight the growing interest and advancements in machine learning-based water quality prediction, paving the way for further development and real-world applications. Furthermore, [16, 17] emphasized the imperative to assess pesticide toxicity due to its association with carcinogenicity, advocating for adopting machine learning methodologies, such as support vector machines and decision trees to forecast the effects of novel pesticides, especially when data is limited. These approaches hold promise for fostering the development of safer pesticides and deepening our understanding of their health implications. [18, 19] underscores the practicality of machine learning in addressing diverse water-related challenges, ranging from predicting water quality parameters to mapping groundwater contaminants and aiding in water treatment processes. These insights offer valuable guidance for the intelligent advancement of water science.

Recent studies demonstrate the potential of machine learning techniques (e.g., XGBoost, Random Forest) for accurate water potability prediction using water quality data (e.g., [5, 20]). [5] explore XGBoost, Random Forest, and KNN for water potability prediction in Indian rivers, achieving high accuracy (e.g., XGBoost with 98.93%). [20] investigate XGBoost, Random Forest, etc., for water quality prediction with techniques for imbalanced data and explainable AI. [21] explore various machine learning algorithms (ANN, Random Forest, etc.) for water quality assessment, reporting promising accuracy for water potability classification. [22] focus on applying machine learning models and parameter optimization for predicting water quality indicators relevant to portability. [23] investigate machine learning models (SVM, Random Forest) for water potability prediction, comparing their performance.

Objective of this paper

The paper aims to assess the effectiveness of machine learning techniques in predicting water potability based on water quality attributes. It initially tests seven classification methods on the

original dataset, with the support vector machine achieving the highest accuracy of 69%. After employing Principal Component Analysis (PCA) to reduce dimensionality, the accuracy of all classifiers increases significantly, nearly reaching 100%. This study highlights the importance of selecting appropriate techniques and the impact of dimensionality reduction on predictive accuracy in water potability prediction.

The study [24] applied machine-learning algorithms directly to the water potability dataset. However, the accuracy of these algorithms was found to be suboptimal. Our research emphasizes the significance of selecting suitable techniques and the influence of dimensionality reduction on water potability prediction using principal component analysis. By implementing these strategies, we achieved an accuracy close to 100%, marking a significant improvement over the performance of the machine learning algorithms.

Work flow

Algorithm 1: Water Potability Analysis Workflow
Data Collection & Processing We have collected the dataset of water potability as mentioned in the Section.

Statistical Analysis

Missing value imputation: First we found some missing values in the To manage missing values, EM imputation was used: all missing values in samples labeled "potable" were replaced with the help of the EM algorithm of all non-missing "potable" samples, and the same method was applied to "non-potable" samples with missing values

Train Test Split: Then we used the Train test split of 70% and 30% randomly

Using algorithms without PCA: We used the machine learning algorithms to predict the potability, we found out that the highest accuracy which we could get from the results without using the Principal Component analysis(PCA) is 69%.

Using algorithms with PCA: We used PCA and created a mixture of all parameters(based on their variance and eigenvalue) and created new parameters of 6 dimensions to investigate whether it gives us better accuracy or not. After using the principal components, we get an accuracy of nearly 99.89%, which is clearly a good result compared to the analysis done earlier.

Checking the accuracy: We used the confusion matrix(mentioned in the section to get the accuracy score for both with PCA & without PCA analysis.

About the dataset

This study uses a [25] dataset for assessing and predicting water potability based on relevant water quality attributes. It includes a total of 3,276 samples analyzed for nine essential hydrochemical parameters: pH value, hardness, total dissolved solids (TDS), chloramines, sulfate, conductivity, organic carbon, trihalomethanes, and turbidity. Each sample's potability, which indicates whether the water is safe for human consumption, is specified, with 1 indicating potable and 0 indicating non-potable. The dataset was randomly divided so that 2,457 samples (75%) were allocated for training and the remaining 819 samples (25%) for testing the models. To manage missing values, EM imputation was used: all missing values in samples labeled

"potable" were replaced with the help of the EM algorithm of all non-missing "potable" samples, and the same method was applied to "non-potable" samples with missing values.

In the Fig 1, We plotted a cylindrical plot concerning water potability where we can say that from the data, we have 61% Non-drinkable and 39% drinkable water information. The World Health Organization (WHO) recommends that the pH values of drinking water fall within the range of 6.5 to 9.5. Similarly, tap water is required in Germany to have a pH value within this range. Let's examine how many of our samples fall within these guidelines [26].

Basic classification based on physical parameters of water

While the pH value of water is crucial for ensuring disinfection and clarification, it alone does not determine water potability according to WHO guidelines. However, maintaining the appropriate pH is vital to prevent the corrosion of copper pipes and certain types of steel, as low pH levels can dissolve these materials, leading to metal contamination. Therefore, the pH value should be considered alongside other water quality indicators [27].

Based on WHO guidelines, we created Fig 2 to illustrate the pH levels of water. According to these standards, a pH level between 6.5 and 9 indicates drinkable water, while levels outside this range indicate non-drinkable water, as depicted in the figure. Similarly, Fig 3 shows the hardness levels of water. WHO categorizes water with a hardness level below 150 as moderately soft and water with levels above 150 as hard, as indicated in the figure. Additionally, Fig 4

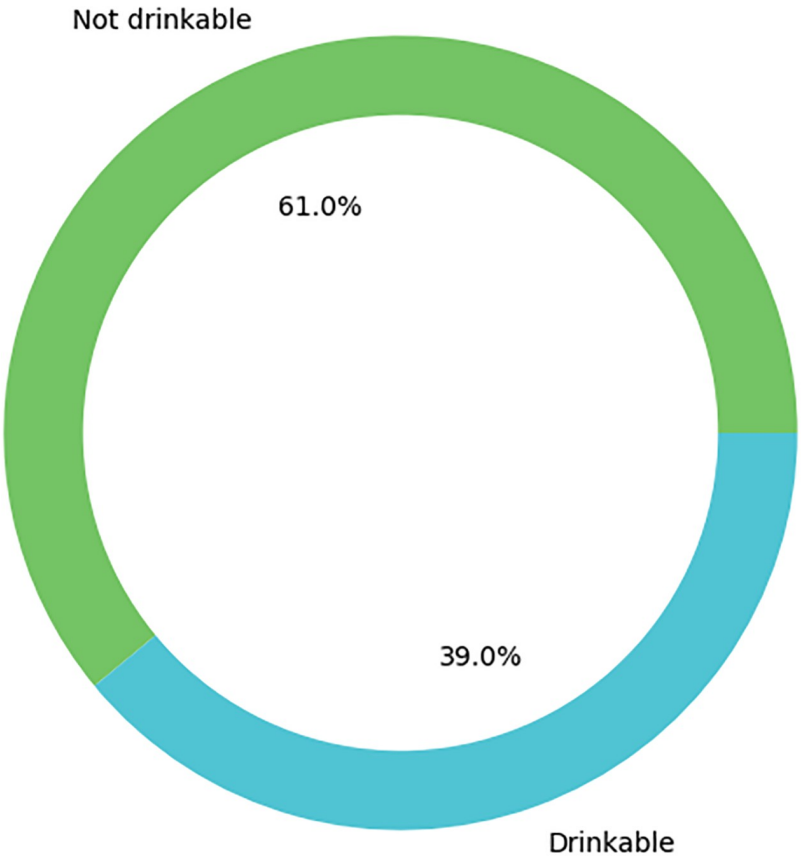

Fig 1. A cylindrical plot concerning water potability where we can say that we have 61% non-drinkable and 39% drinkable water information from our data.

https://doi.org/10.1371/journal.pwat.0000259.g001

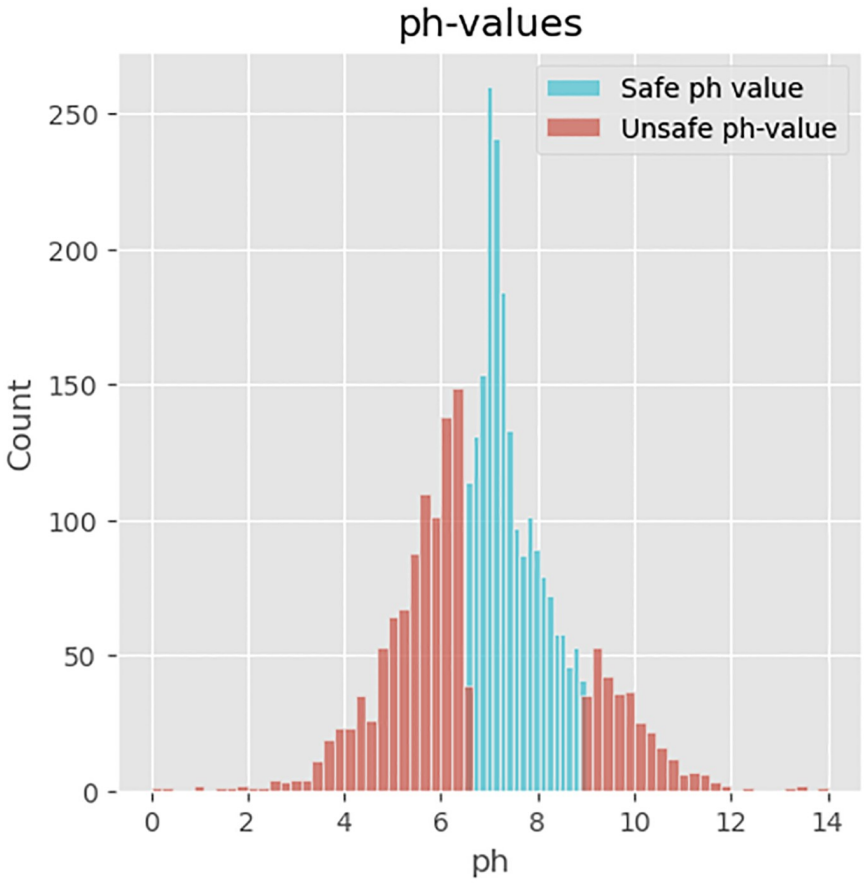

Fig 2. Plot of the pH level of water from our dataset where according to WHO, pH level > 6.5 and <9 means it is in the drinkable PH mask, and if the PH level is <6.5 and >9 means it is in the non-drinkable pH mask, which is shown concerning count in the picture. Here, the blue defines the safe pH values, and red is unsafe pH values.

https://doi.org/10.1371/journal.pwat.0000259.g002

presents the sulfate levels of water. According to WHO guidelines, sulfate levels below 250 fall within the EU limits, while levels between 250 and 500 fall within WHO limits, as shown in the figure [28] and all of those are combinely shown in Fig 5.

This dataset contains water quality measurements and assessments related to potability, which is the suitability of water for human consumption. The dataset's primary objective is to provide insights into water quality parameters and assist in determining whether the water is potable. Each row in the dataset represents a water sample with specific attributes, and the "Potability" column indicates whether the water is suitable for consumption.

This dataset is suitable for a supervised binary classification task, where machine learning models can be trained to predict water potability based on the provided water quality attributes. The models aim to classify water samples as potable (1) or not potable (0). This dataset is valuable for water quality assessment, water treatment planning, and ensuring the safety of drinking water supplies. It can be utilized by water treatment plants, environmental agencies, and researchers to make data-driven decisions regarding water quality and portability.

Water Quality Index (WQI)

The Water Quality Index (WQI) is a comprehensive measure of water quality, reflecting various characteristics indicative of its overall condition. Traditionally, nine parameters contribute

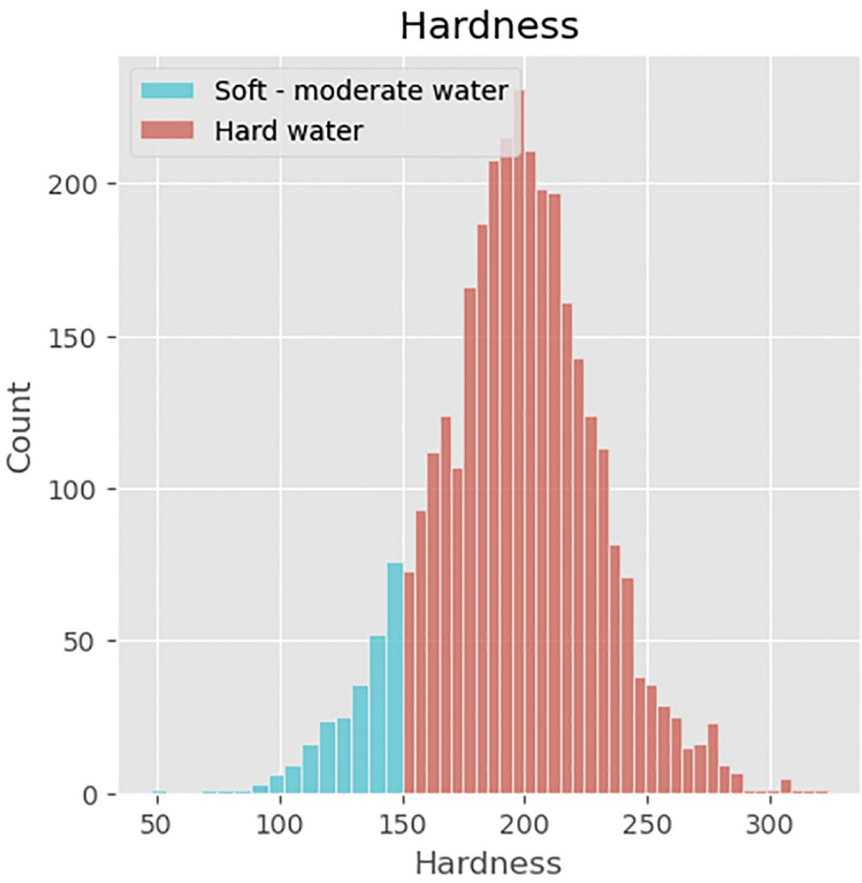

Fig 3. Plot of the hardness level of water from our dataset where according to who hardness level <150 means it is in the soft, moderate water mask and if the hardness level is >150 means it is in the hard water mask, which is shown concerning count in the picture. Here, blue is the soft, moderate water value, and red is the Hard water value.

https://doi.org/10.1371/journal.pwat.0000259.g003

to the calculation of WQI. The formula used for WQI calculation is represented as follows: [24]

$$\mathrm{WQI} = \frac{\sum_{i=1}^{N} q_i \times w_i}{\sum_{i=1}^{N} w_i}$$

where N denotes the number of attributes, q_i signifies the quality rating scale, computed according to the formula:

$$q_i = 100 \times \left(\frac{V_i - V_{\mathrm{Ideal}}}{S_i - S_{\mathrm{Ideal}}} \right)$$

Here, V_i represents the observed value, V_{Ideal} denotes the ideal value, and S_i signifies the standard value of the parameter. The parameter's standard value, w_i, is calculated using the equation:

$$w_i = \frac{K}{S_i}$$

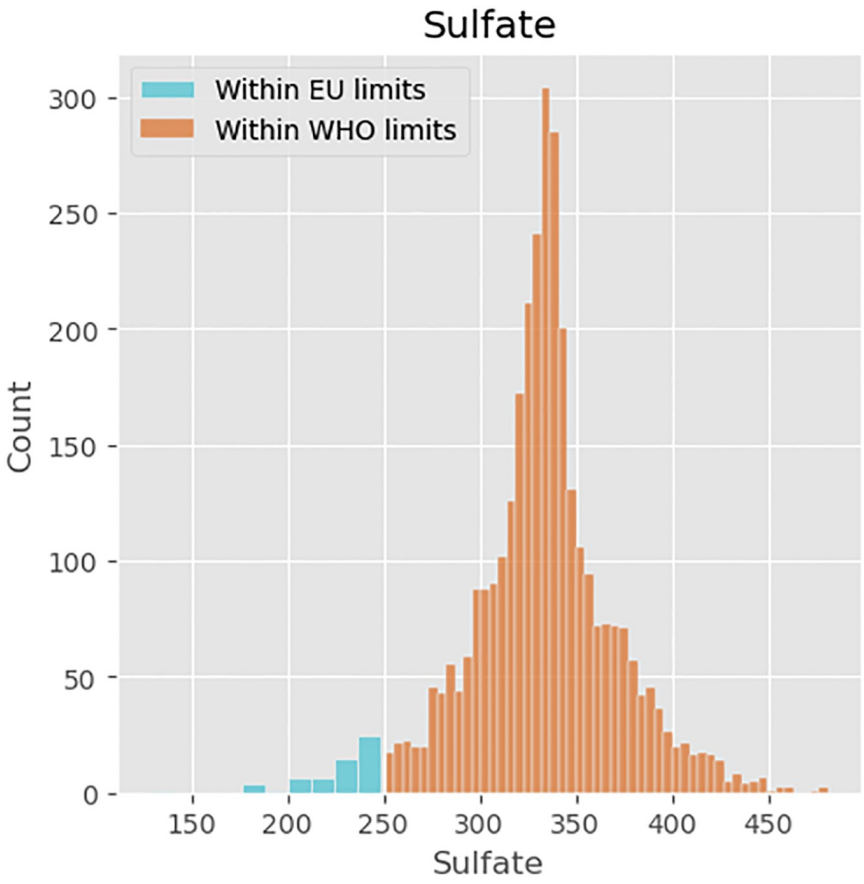

Fig 4. We have plotted the sulfate level of water where according to who sulfate level <250 which is within EU limits sulfate mask and if the sulfate level is >250 and <500 means it is within WHO limit sulfate mask, which is shown concerning count in the picture. Here, blue defines the EU as limiting sulfate values, and orange is within WHO's limits.

https://doi.org/10.1371/journal.pwat.0000259.g004

where the proportionality constant K is determined as:

$$K = \frac{1}{\sum_{i=1}^{N} S_i}$$

Methodologies

EM algorithm

The Expectation-Maximization (EM) algorithm is an iterative method for finding maximum likelihood or maximum a posteriori estimates of parameters in statistical models where the model involves latent variables. The algorithm consists of two steps: the E-step and the M-step [29].

The EM (Expectation-Maximization) algorithm is an iterative method used to estimate the missing values in a dataset when some values are not observed. In the context of the parameters $x_1, x_2, x_3, \ldots, x_{10}$, where the parameters are 'ph', 'Hardness', 'Solids', 'Chloramines', 'Sulfate', 'Conductivity', 'Organic_carbon', 'Trihalomethanes', 'Turbidity', and 'Potability', the EM algorithm can be applied to impute missing values in the dataset. [30]

Start with initial estimates for the missing values. These estimates can be random or based on some simple imputation method. Compute the expected values of the missing data given

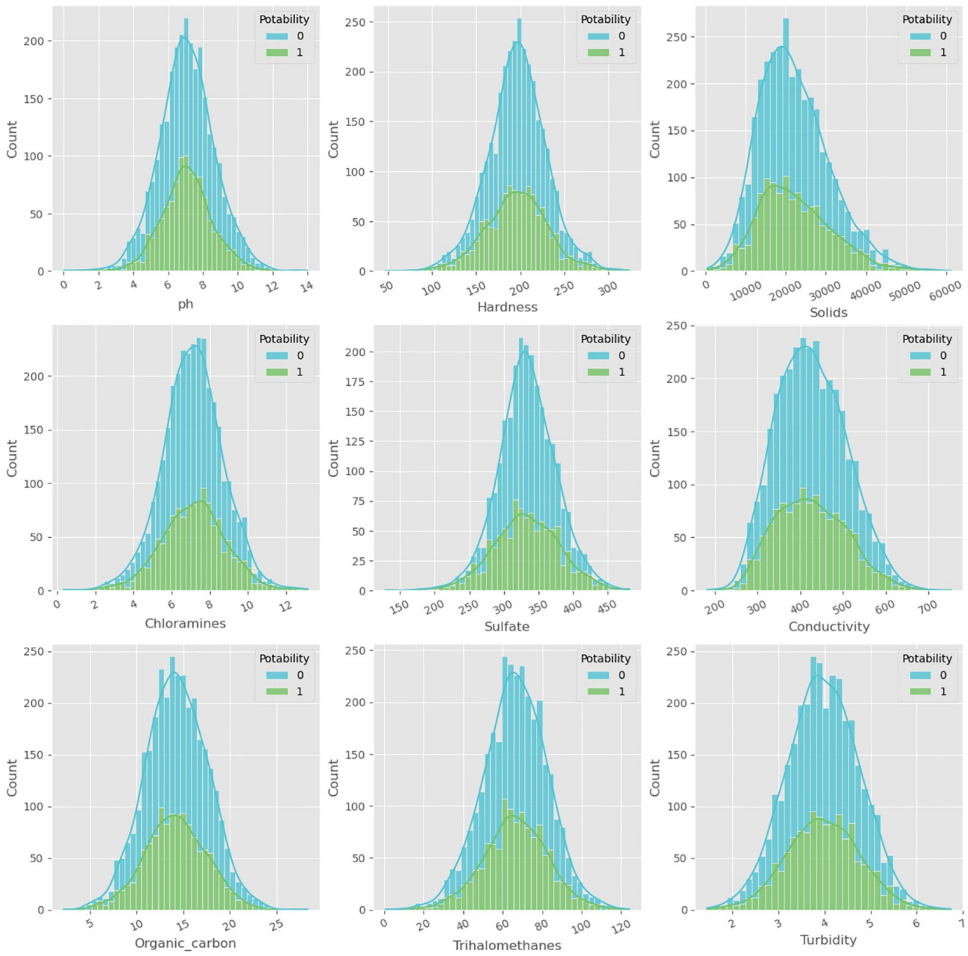

Fig 5. Visualization(Matrix plot) of water potability from our dataset across nine features (pH, hardness, sulfate, solids, conductivity, etc.). The plot illustrates water potability, with blue indicating potability as 0 and green as 1.

https://doi.org/10.1371/journal.pwat.0000259.g005

the observed data and the current parameter estimates. This step estimates the probability distribution of the missing values given the observed data and the current parameter values. Update the parameter estimates based on the observed data and the expected values of the missing data computed in the E-step. This step involves maximizing the likelihood function to find the parameter values that best fit the observed data and the estimated missing values. Repeat the E-step and M-step until convergence, i.e., until the parameter estimates and the imputed missing values no longer change significantly between iterations. Once convergence is reached, the final parameter estimates and imputed missing values are obtained.

Principal Component Analysis (PCA)

PCA is a technique used for dimensionality reduction and feature extraction. It transforms the original variables into a new set of uncorrelated variables called principal components. The PCA algorithm finds the directions (principal components) along which the data varies the most. [31]

Given a dataset with n observations and p variables represented by the matrix X, where each row corresponds to an observation and each column corresponds to a variable, PCA computes the covariance matrix C of X.

The covariance matrix C is then decomposed into its eigenvectors and eigenvalues. The eigenvectors represent the directions of maximum variance in the data, while the eigenvalues represent the magnitude of the variance along each eigenvector [32].

The principal components are obtained by projecting the original data onto the eigenvectors. These components are ordered by the corresponding eigenvalues, with the first principal component capturing the most variance in the data, the second capturing the second most, and so on.

One approach to impute missing values using PCA is to reconstruct the missing values using the information captured by the principal components. The missing values for each observation can be estimated by projecting the observed values onto the principal components and then back-transforming them to the original variable space.

The imputed values are obtained by multiplying the projected values by the loading matrix, representing the relationship between the original variables and the principal components. The loading matrix can be obtained from the eigenvectors of the covariance matrix.

This process allows for imputing missing values while preserving the underlying data structure captured by the principal components [33].

Consider a dataset $x_1, x_2, x_3, \ldots, x_9$, where the parameters are 'ph', 'Hardness', 'Solids', 'Chloramines', 'Sulfate', 'Conductivity', 'Organic_carbon', 'Trihalomethanes', 'Turbidity', in the dataset $\mathbf{X}$ with n samples and 9 features arranged as an $n \times p$ matrix.

First, center the data by subtracting the mean of each feature:

$$\mathbf{X}_{\text{centered}} = \mathbf{X} - \bar{\mathbf{X}}$$

where $\bar{\mathbf{X}}$ is a vector of length p containing the means of each feature.

Calculate the covariance matrix $\mathbf{S}$ of the centered data:

$$\mathbf{S} = \frac{1}{n-1}(\mathbf{X}_{\text{centered}}^T \cdot \mathbf{X}_{\text{centered}})$$

Perform an eigenvalue decomposition on $\mathbf{S}$:

$$\mathbf{S} = \mathbf{V} \cdot \Lambda \cdot \mathbf{V}^T \tag{7}$$

where $\mathbf{V}$ is a matrix containing the eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix containing the eigenvalues.

Logistic regression

Logistic regression is a statistical method commonly used for binary classification problems. The logistic function, also known as the sigmoid function, is the core of logistic regression [34].

Consider a dataset $x_1, x_2, x_3, \ldots, x_9$, where the parameters are 'ph', 'Hardness', 'Solids', 'Chloramines', 'Sulfate', 'Conductivity', 'Organic_carbon', 'Trihalomethanes', 'Turbidity', and the target variable is potability.

The logistic regression model predicts the probability that a given observation belongs to a particular class (in this case, whether water is potable or not) based on the values of the input features. It uses the logistic function (also known as the sigmoid function) to map the linear combination of input features to the probability of belonging to a certain class [35].

The logistic function is defined as:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Where z is the linear combination of input features and their corresponding weights:

$$z = w_0 + w_1x_1 + w_2x_2 + \ldots + w_9x_9 \quad (9)$$

In logistic regression, the weights w_0, w_1, w_2, . . ., w_9 are learned during training to minimize the error between the predicted probabilities and the actual class labels.

The probability that an observation belongs to the positive class (potable water) can be expressed as: [36]

$$P('\text{potability}' = 1|x_1, x_2, \ldots, x_9) = \sigma(w_0 + w_1x_1 + w_2x_2 + \ldots + w_9x_9)$$

The probability that it belongs to the negative class (non-potable water) is:

$$P('\text{potability}' = 0|x_1, x_2, \ldots, x_9) = 1 - \sigma(w_0 + w_1x_1 + w_2x_2 + \ldots + w_9x_9)$$

During training, the weights w_0, w_1, w_2, . . ., w_9 are learned by maximizing the likelihood of the observed data given the model parameters. This is typically done using optimization techniques such as gradient descent.

Once the model is trained, it can be used to impute missing values by predicting the probability of the target variable (potability) based on the available features x_1, x_2, . . ., x_9 and using these probabilities to impute the missing values.

Decision tree classifier

Let X_1, X_2, X_3, . . ., X_9 represent the parameters:

X_1: ph, X_2: Hardness, X_3: Solids, X_4: Chloramines, X_5: Sulfate, X_6: Conductivity, X_7: Organic_carbon, X_8: Trihalomethanes, X_9: Turbidity,

Let Y represent the target variable: potability.

1. **Initialization**: Start with the root node.
2. **Splitting Criteria**:
   - At each node, choose the parameter that maximizes information gain or minimizes impurity to split the dataset.
   - Calculate the information gain or impurity reduction for each possible split.
3. **Stopping Criteria**:
   - Stop splitting when a predefined stopping criterion is met, such as reaching a maximum depth, minimum number of samples per node, or minimum impurity threshold.
4. **Imputation**:
   - For each sample with missing values, traverse the decision tree to find the appropriate leaf node.
   - Use the majority class or mean/median value of samples in that leaf node to impute the missing value for the corresponding parameter.

Random forest classifier

We have a dataset with parameters {x_1, x_2, x_3, . . ., x_9}, where each parameter represents different water quality attributes such as pH, hardness, etc. The dataset also contains a target variable, 'potability'. However, some of the values in the parameters may be missing. We aim to use a Random Forest classifier algorithm to impute these missing values [37].

- x_i: The ith parameter where $i = 1, 2, 3, \ldots, 9$.
- potability: Target variable representing whether water is potable.

A Random Forest classifier is an ensemble learning method that constructs many decision trees during training and outputs the mode of the classes (classification) or mean prediction (regression) of the individual trees [38].

1. **Data Preparation**: Prepare the dataset by replacing missing values in parameters $x_1, x_2, x_3, \ldots, x_9$ with NaN (Not a Number) or any other placeholder value.
2. **Train the Random Forest Classifier**: Train a Random Forest classifier on the dataset with missing values. The input features are $x_1, x_2, x_3, \ldots, x_9$ and the target variable is 'potability'.
3. **Impute Missing Values**: For each missing value in parameters $x_1, x_2, x_3, \ldots, x_9$, use the trained Random Forest classifier to predict the missing value based on the values of other parameters and the target variable 'potability'.

Let Y be the set of parameters $\{x_1, x_2, x_3, \ldots, x_9\}$, and let potability be the target variable. The Random Forest classifier learns a mapping $f: X \rightarrow Y$, where X is the space of possible feature vectors. Thus, for any input feature vector x with missing values, the imputed value y' can be represented as:

$$y' = f(x).$$

Support Vector Machine (SVM) classifier

Let $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$ be the training dataset, where $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{i9})$ is the feature vector and y_i is the target variable. Using SVM, we formulate the imputation of missing values as a binary classification problem.

Let $\hat{\mathbf{x}}_i = (x_{i1}, x_{i2}, \ldots, x_{ij-1}, x_{ij+1}, \ldots, x_{i9})$ represent the feature vector of $\mathbf{x}_i$ with the j^{th} feature removed [39].

The objective function of SVM for imputation can be written as:

$$\min_{\mathbf{w},b,\xi} \frac{1}{2} \| \mathbf{w} \|^2 + C \sum_{i=1}^{n} \xi_i.$$

Subject to the constraints:

$$y_i(\mathbf{w} \cdot \hat{\mathbf{x}}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, 2, \ldots, n$$

Where:

- $\mathbf{w}$ is the weight vector,
- b is the bias term,
- $\boldsymbol{\xi}$ are slack variables,
- C is the regularization parameter,
- y_i is the class label (potability) of $\mathbf{x}_i$,
- $(\hat{\mathbf{x}}_i, y_i)$ are training samples with missing values imputed.

The decision function is given by: [40]

$$f(\mathbf{x}) = \text{sign}(\mathbf{w} \cdot \hat{\mathbf{x}} + b)$$

Once the SVM model is trained, missing values can be imputed using the decision function $f(\mathbf{x})$ for each sample $\mathbf{x}$ with missing values.

Naive Bayes classifier

Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem. It is a simple and fast algorithm that is particularly effective for text classification and spam filtering [41–43].

Let $X = \{x_1, x_2, x_3, \ldots, x_9\}$ denote the set of parameters: pH, Hardness, Solids, Chloramines, Sulfate, Conductivity, Organic_carbon, Trihalomethanes, and Turbidity, and Y denote the target variable, Potability.

For each parameter x_i where $i = 1, 2, \ldots, 9$, let C_j denote the jth class of x_i, where j ranges over all possible values of x_i.

Let $P(Y = y)$ denote the prior probability of the target variable being y (potable or non-potable).

For each parameter x_i, we calculate the conditional probability of each class C_j given the target variable Y, denoted as $P(C_j|Y)$, using the formula:

$$P(C_j|Y) = \frac{P(Y|C_j) \cdot P(C_j)}{P(Y)}$$

Where:

- $P(Y|C_j)$ is the probability of the target variable being y given the class C_j, calculated using the training data.
- $P(C_j)$ is the prior probability of class C_j, calculated using the training data.
- $P(Y)$ is the prior probability of the target variable Y, calculated using the training data.

Then, to impute missing values for a given sample $\vec{x} = \{x_1, x_2, x_3, \ldots, x_9\}$, we calculate the probability of each class C_j for each parameter x_i, using the Naive Bayes assumption that the parameters are conditionally independent given the target variable:

$$P(C_j|\vec{x}) = P(C_j) \cdot \prod_{i=1}^{9} P(x_i|C_j)$$

Finally, for each missing value x_i, we impute the most probable class C_j and assign the corresponding value of x_i for that class.

This algorithm is based on the Naive Bayes assumption and is commonly used for imputing missing values in classification problems.

Nearest Neighbors algorithm

Let $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$ be the training dataset, where each $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ represents a data point with p features, and y_i represents the target variable [44].

We define the distance function $d(\mathbf{x}_i, \mathbf{x}_j)$ to measure the similarity between two data points $\mathbf{x}_i$ and $\mathbf{x}_j$. One commonly used distance metric is the Euclidean distance:

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{k=1}^{p} (x_{ik} - x_{jk})^2}$$

Given a data point $\mathbf{x}$ with missing values, let $\mathbf{x}_{\text{miss}}$ denote the subset of features with missing values, and $\mathbf{x}_{\text{obs}}$ denote the subset of features with observed values. We aim to impute the missing values in $\mathbf{x}_{\text{miss}}$ using the k nearest neighbors of $\mathbf{x}$ in the feature space [45].

The Nearest Neighbors imputation algorithm can be summarized as follows:

1. For each data point $\mathbf{x}_i$ in the dataset, compute the distance $d(\mathbf{x}, \mathbf{x}_i)$ between $\mathbf{x}$ and $\mathbf{x}_i$.
2. Select the k data points with the smallest distances to $\mathbf{x}$, denoted as $\{(\mathbf{x}_{i_1}, y_{i_1}), (\mathbf{x}_{i_2}, y_{i_2}), \ldots, (\mathbf{x}_{i_k}, y_{i_k})\}$.
3. Impute the missing values in $\mathbf{x}_{\text{miss}}$ using the observed values in the corresponding features of the k nearest neighbors. One common approach is to take the average of the observed values:>

$$\hat{x}_j = \frac{1}{k} \sum_{l=1}^{k} x_{i_l j}$$

where $\hat{x}_j$ is the imputed value for the j-th feature in $\mathbf{x}_{\text{miss}}$.

4. Repeat steps 1-3 for each data point with missing values.

The Nearest Neighbors algorithm for imputing missing values assumes that similar data points have similar feature values. By identifying the k nearest neighbors of a data point with missing values, we leverage the information from similar data points to impute the missing values effectively.

This algorithm is handy when dealing with datasets with missing values, as it allows us to retain as much information as possible from the available data.

XGBoost: Extreme gradient boosting

XGBoost is a powerful and efficient machine-learning algorithm based on gradient gradient-boosting framework. It is designed for speed and performance, making it a popular choice for various predictive modeling tasks [46, 47].

Given Training Dataset: $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$

where $\mathbf{x}_i = (x_{i1}, x_{i2}, x_{i3}, x_{i4}, \ldots, x_{i9})$ represents the feature vector for the ith sample, and y_i represents the corresponding target variable.

Let $\mathbf{x}_i^{(j)}$ be the jth feature of the ith sample.

The objective function of the XGBoost algorithm for imputing missing values can be represented as:

$$\text{Objective Function}: \mathcal{L}(\Phi) = \sum_{i=1}^{N} \mathcal{L}(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

Where:

- $\hat{y}_i = \sum_{k=1}^{K} f_k(\mathbf{x}_i)$ is the prediction for the ith sample obtained by summing the predictions of all individual trees f_k.

- $\mathcal{L}$ is the loss function measuring the discrepancy between the true target y_i and the predicted target $\hat{y}_i$.
- Ω is the regularization term to penalize the complexity of the model.

The prediction for the ith sample by the kth tree is given by:

$$\hat{y}_i^{(k)} = \hat{y}_{i-1}^{(k)} + \gamma \cdot h_k(\mathbf{x}_i^{(j)})$$

Where:

- $\hat{y}_{i-1}^{(k)}$ is the prediction of the previous ensemble.
- γ is the learning rate.
- $h_k(\mathbf{x}_i^{(j)})$ is the contribution of the kth tree to the prediction for the ith sample.

To impute missing values, we modify the prediction for the ith sample by considering the values of the features:

$$\hat{y}_i^{(k)} = \hat{y}_{i-1}^{(k)} + \gamma \cdot h_k(\mathbf{x}_i^{(j)}) \cdot m(\mathbf{x}_i^{(j)})$$

Where:

- $m(\mathbf{x}_i^{(j)})$ is the masking function which masks the contribution of the kth tree if the value of x_{ij} is missing.

This way, XGBoost adapts to the missing values during training by learning appropriate splits and imputations for the missing data [48, 49].

Accuracy prediction

Determining the accuracy of each algorithm involves utilizing confusion matrices to assess performance. These matrices provide a comprehensive summary of model predictions compared to the true values in a tabular format. Key terms include:

- True Positive (TP): Correctly predicted positive instances.
- True Negative (TN): Correctly predicted negative instances.
- False Positive (FP): Instances falsely predicted as positive (Type I error).
- False Negative (FN): Instances falsely predicted as negative (Type II error).

The accuracy of a classification model is calculated using the formula:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Population}} = \frac{TP + TN}{TP + TN + FP + FN}$$

PCA-based enhanced classification accuracy: A dimensionality reduction approach

Dimensionality reduction, a critical preprocessing step in machine learning, is particularly beneficial when dealing with high-dimensional data. Principal Component Analysis (PCA), a popular technique, can significantly improve classification accuracy by identifying and retaining the most informative features while discarding irrelevant or redundant ones.

Let $\mathbf{X}$ be the $n \times p$ data matrix, where n is the number of samples, and p is the number of features. The PCA process can be summarized as:

1. $\mathbf{X}_{std} = \frac{\mathbf{X}-\mu}{\sigma}$ (Standardization)
2. $\mathbf{C} = \frac{1}{n-1}\mathbf{X}_{std}^T\mathbf{X}_{std}$ (Covariance matrix)
3. $\mathbf{C}\,\mathbf{v}_i = \lambda_i\mathbf{v}_i$ (Eigenvalue decomposition)
4. $\mathbf{Z} = \mathbf{X}_{std}\mathbf{V}_k$ (Projection onto top k PCs).

Here, μ is the mean vector of features, σ is the standard deviation vector of features, $\mathbf{v}_i$ are the eigenvectors, λ_i are the eigenvalues, $\mathbf{V}_k$ is the matrix of the top k eigenvectors. PCA transforms the original data into a new coordinate system, aligning the axes with the directions of maximum variance. We select the top k PCs, where k is the desired reduced dimensionality. After that, we project the original data onto the selected PCs to obtain the reduced representation.

The red and blue arrows in the Fig 6 represent the first two principal components, and the dots are the data points. PCA projects the data onto the PC1 and PC2 axes, capturing most of the variance with fewer dimensions (for instance, red is much bigger, showing that the one-dimensional line towards red captures the most information).

In this paper, we employ PCA, then ML based classification techniques. This 2-step approach is novel, to our knowledge. PCA can enhance classification accuracy by:

1. Noise Reduction: Discarding irrelevant features reduces noise, leading to a cleaner data representation.
2. Overfitting Prevention: Lower dimensionality helps prevent overfitting, improving generalization to unseen data.
3. Computational Efficiency: Fewer features lead to faster training and prediction times.
4. Feature Extraction: PCA can sometimes reveal hidden patterns or relationships in the data.

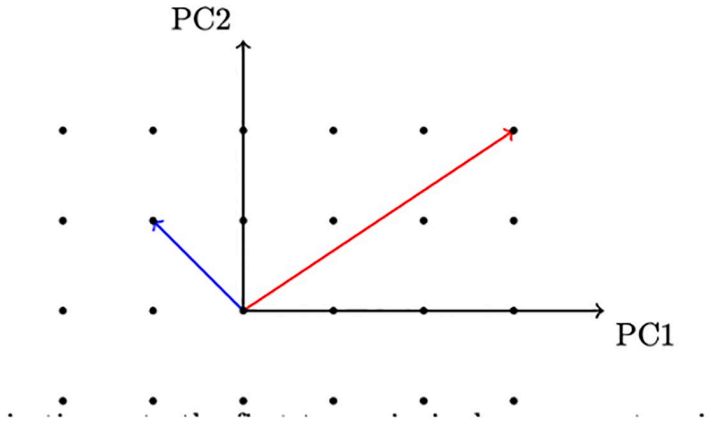

Fig 6. PCA projection onto the first two principal components using a simple illustration. The prominence of the large red arrow indicates that the one-dimensional line directed towards red encompasses the majority of the information present in the data.

https://doi.org/10.1371/journal.pwat.0000259.g006

Results

In this study, we applied machine learning techniques to the Water Quality and Potability dataset to evaluate their performance in predicting water potability based on water quality attributes. Our results demonstrate varying degrees of accuracy across the different classifiers. The Support Vector Machine (SVM) classifier achieved the highest accuracy score of 69%, followed closely by k-Nearest Neighbors (k-NN) and Gaussian Naive Bayes, both achieving an accuracy of 67.65%. Random Forest also performed similarly with an accuracy score of 66.9%. However, Decision Tree and Logistic Regression classifiers exhibited comparatively lower accuracy scores of 58.697% and 62.66%, respectively. Interestingly, the XGBoost classifier yielded an accuracy score of 64.69%, showing competitive performance with other methods. These results highlight the importance of selecting an appropriate machine learning algorithm tailored to the dataset's characteristics. Additionally, further exploration into ensemble methods or parameter tuning could potentially enhance the predictive performance of these classifiers. Overall, these findings contribute to advancing water quality assessment methodologies and provide valuable insights for decision-makers in ensuring safe and potable water resources. All the values and methods are mentioned in the Table 1 After applying seven machine learning algorithms to the data following missing value imputations, we found that the Support Vector Machine (SVM) algorithm outperforms the others. A bar diagram illustrating their accuracy scores is presented in Fig 7. Since SVM achieved the highest accuracy score of 69%, we created an SVM partition plot focusing on two features to provide an overall overview. Despite utilizing all nine features to attain the accuracy score, visualizing them in a nine-dimensional space is impractical, as demonstrated in Fig 8.

We extended our analysis by applying machine learning techniques to a reduced dataset obtained through Principal Component Analysis (PCA), specifically focusing on the first six dimensions. This dimensionality reduction technique aims to capture the most significant patterns in the original dataset while reducing computational complexity. Our results show notable improvements in the accuracy scores of several classifiers when applied to the PCA-transformed dataset. Notably, Logistic Regression classification exhibited a remarkable increase in accuracy, achieving an impressive score of 99.89%. Similarly, XGBoost demonstrated a substantial improvement, with an accuracy score of 99.28%. Other classifiers, including k-Nearest Neighbors, Gaussian Naive Bayes, and Random Forest, maintained high accuracy scores of 98.8%, indicating robust performance even with reduced dimensions. Support Vector Machine (SVM) and Random Forest classifiers also displayed impressive improvements, with accuracy scores of 99% and 98.88%, respectively. These findings underscore the efficacy of PCA in enhancing the predictive performance of specific classifiers, particularly

Table 1. By using the seven different machine learning algorithms on the data after missing value imputations, we get that the Support Vector Machine algorithm gives us the best result among them per the accuracy score. All of the accuracy scores are mentioned in the table.

Machine Learning algorithm used	Accuracy Score
Logistic Regression	62.66
Decision Tree Classifier	62.25
Random Forest Classifier	66.9
Support Vector Machine	69
Gaussian Naive Bias	67.65
Nearest Neighbors	67.655
XGBoost	64.69

https://doi.org/10.1371/journal.pwat.0000259.t001

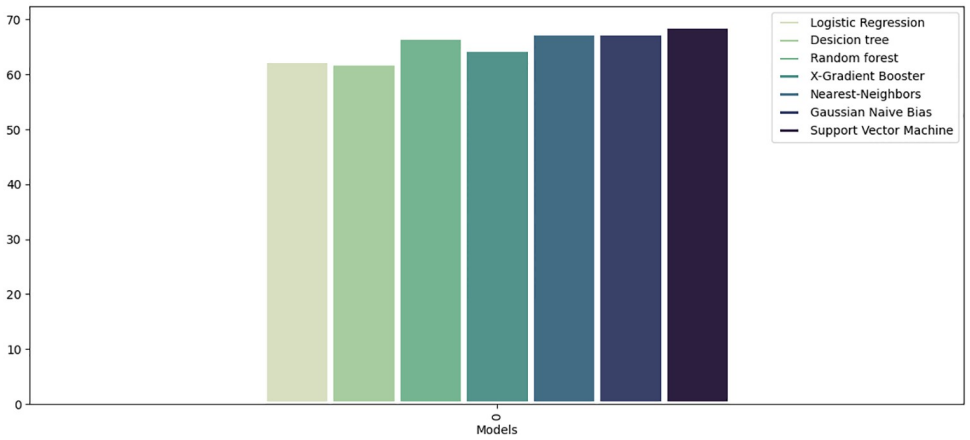

Fig 7. By using the seven different machine learning algorithms on the data after missing value imputations, we get that the Support Vector Machine algorithm is giving us the best result among them. A pictorial bar diagram concerning their accuracy score is given in this plot.

https://doi.org/10.1371/journal.pwat.0000259.g007

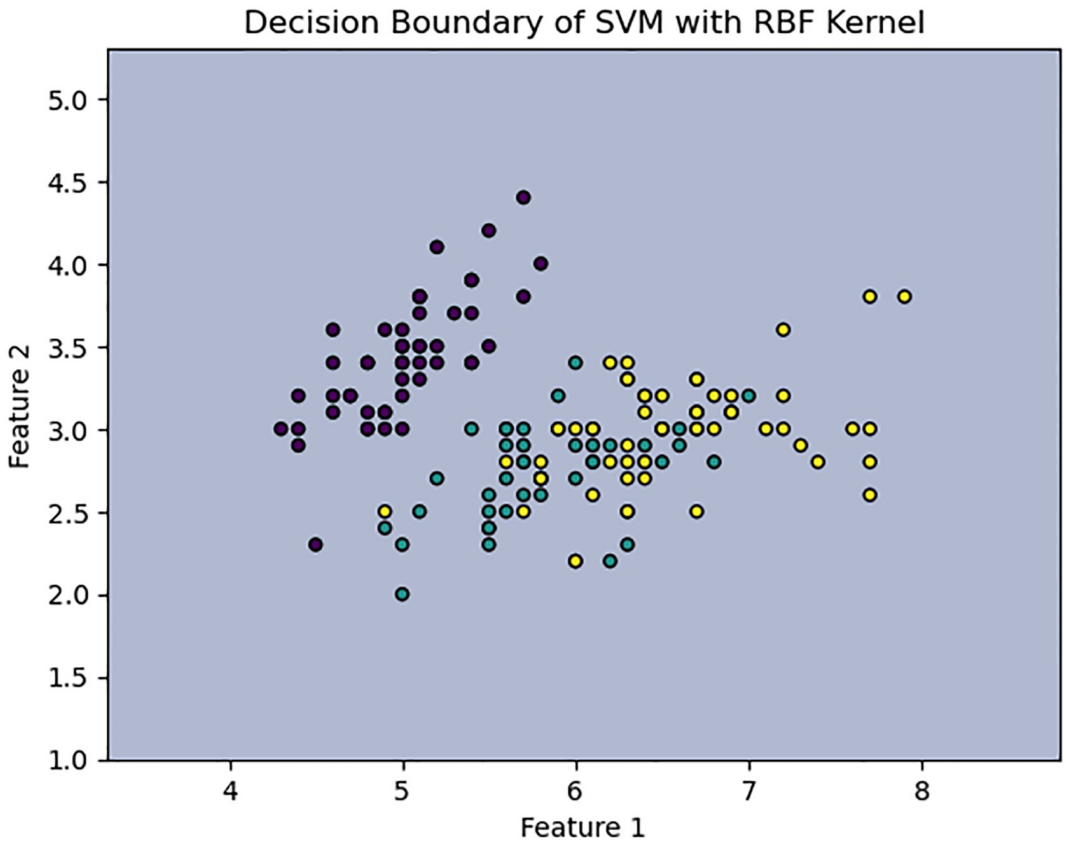

Fig 8. As we get the best algorithm Support Vector Machine with the accuracy score of 69%, we plotted the SVM partition plot concerning two features to give an overall overview. Though we used all of the nine features to get the accuracy score, it is not possible to plot nine features concerning nine dimensions.

https://doi.org/10.1371/journal.pwat.0000259.g008

Table 2. By using the seven different machine learning algorithms on the Principal Components after using PCA, we get that the Logistics Algorithm gives us the best result among them per the accuracy score. All of the accuracy scores are mentioned in the table.

Machine Learning algorithm used	Accuracy Score
Logistic Regression	99.89
Decision Tree Classifier	96.74
Random Forest Classifier	98.88
Support Vector Machine	99
Gaussian Naive Bias	98.80
Nearest Neighbors	98.88
XGBoost	99.28

https://doi.org/10.1371/journal.pwat.0000259.t002

Decision Tree and XGBoost while maintaining competitive accuracy across other methods. Our results suggest that leveraging dimensionality reduction techniques can significantly contribute to more efficient and accurate water potability prediction models, thus facilitating informed decision-making in water management and public health initiatives. The PCA picture and the accuracy scores are mentioned in Table 2.

Using the seven different machine learning algorithms on the Principal Components after using PCA shows us that the Logistic Algorithm gives us the best result. A pictorial bar diagram concerning their accuracy score is provided in the Fig 9.

In the Fig 10, we have plotted the Eigenvalues of each dimension after performing the Principal Component Analysis; here, the blue bar diagram shows the explained variance ratio of each dimension, and the Orange line shows the cumulative explained variance of 6 dimensions where it is cumulatively carrying nearly 65% of the information.

In the Fig 11, we plotted the PCA concerning the first three dimensions where our main factor is water potability(o or 1); here, 1 is defined as yellow, and 0 is defined as purple. From this Fig 11, we can see that three dimensions are not enough to get a suitable data partition.

So after interpreting the Figs 10 and 11, we choose six optimal dimensions.

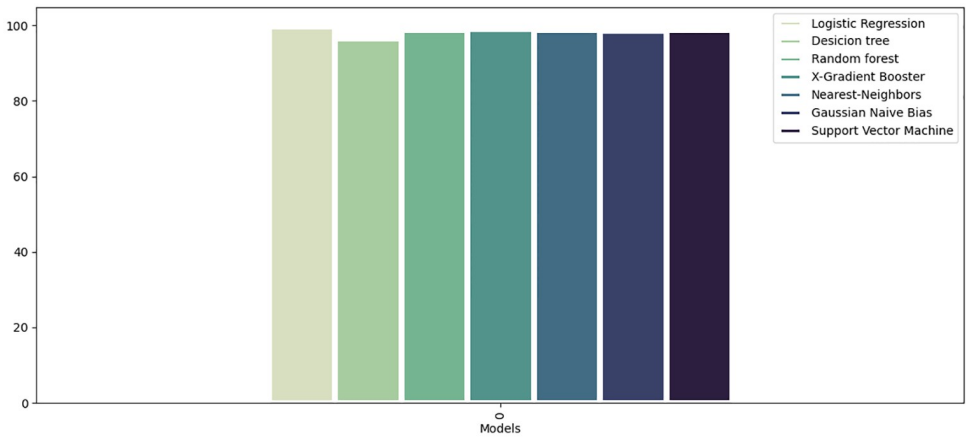

Fig 9. Using the seven different machine learning algorithms on the Principal Components after using PCA, we get that the Logistic Algorithm gives us the best result. A pictorial bar diagram concerning their accuracy score is given in this plot.

https://doi.org/10.1371/journal.pwat.0000259.g009

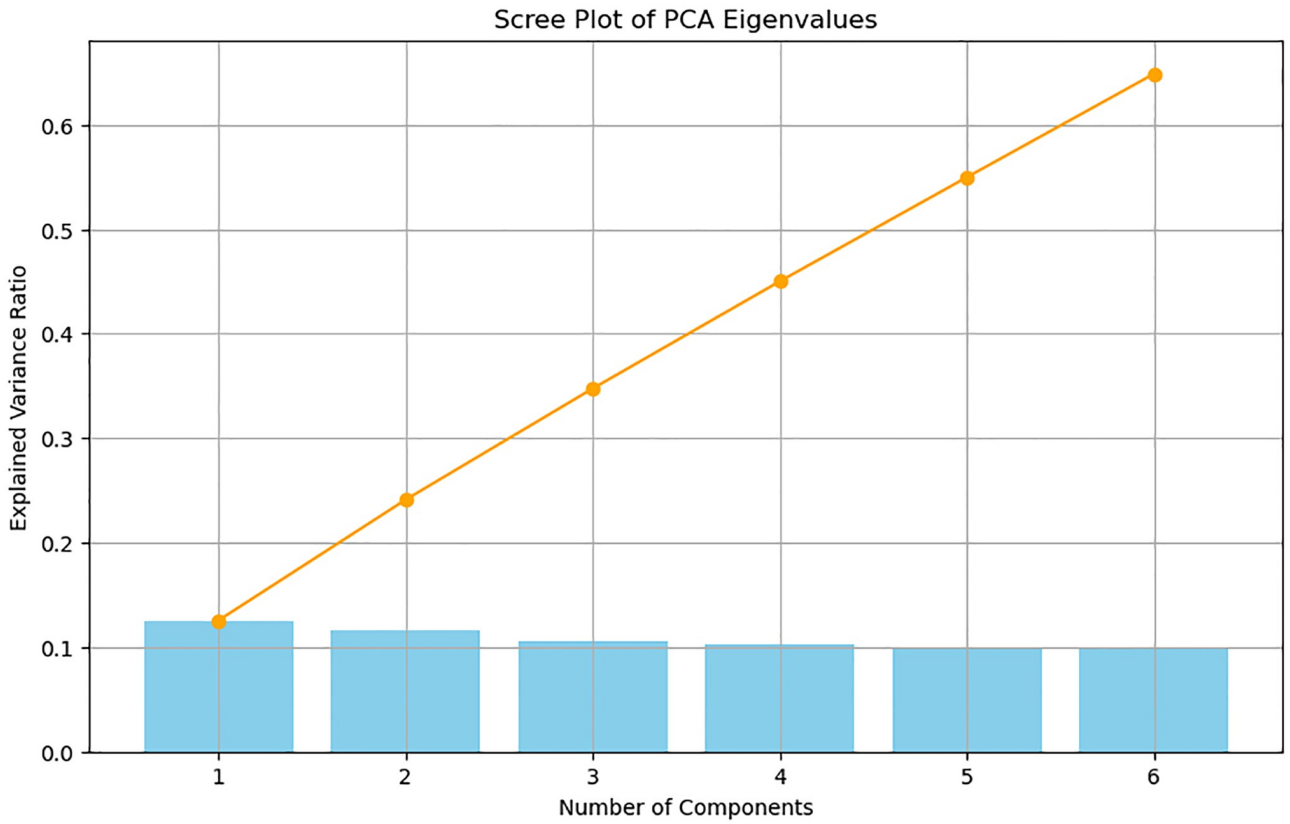

Fig 10. Plotted the Eigenvalues of each dimension after performing the Principal Component Analysis; here, the blue bar diagram shows the explained variance ratio of each dimension, and the Orange line shows the cumulative explained variance of 6 dimensions.

https://doi.org/10.1371/journal.pwat.0000259.g010

After identifying the Support Vector Machine as the top-performing algorithm with an accuracy score of 99.89%, we generated a Logistic Regression partition plot focusing on two features to provide a comprehensive overview. While all nine features were utilized to achieve the accuracy score, visualizing them in a nine-dimensional space is impractical, as depicted in Fig 12.

In the Fig 13, feature importances of logistic regression based on the first six dimensions or components of PCA are plotted. The plot displays the importance percentages of each feature, representing their contribution to the logistic regression model's decision-making process. Where all the feature importance of the dimensions are given in percentages shown in the picture, from the picture, we can see that Dim 6 is carrying most of the information (near about 42.34%) and then Dim 3(19.27%) and Dim 2(12.49%) and then the rest of the dimensions.

Discussion

Our analysis observed varying degrees of accuracy among machine learning techniques applied to the original Water Quality and Potability dataset and the reduced dataset obtained through Principal Component Analysis (PCA). Initially, on the original dataset, the Support Vector Machine exhibited the highest accuracy of 69%, followed closely by Gaussian Naive bias and the Nearest Neighbors algorithm with an accuracy of 67.65%.

The suboptimal accuracy, as mentioned above, of the machine learning algorithms, after being applied directly to the water quality dataset, can be attributed to several factors. High dimensionality can lead to the curse of dimensionality, where the performance of models

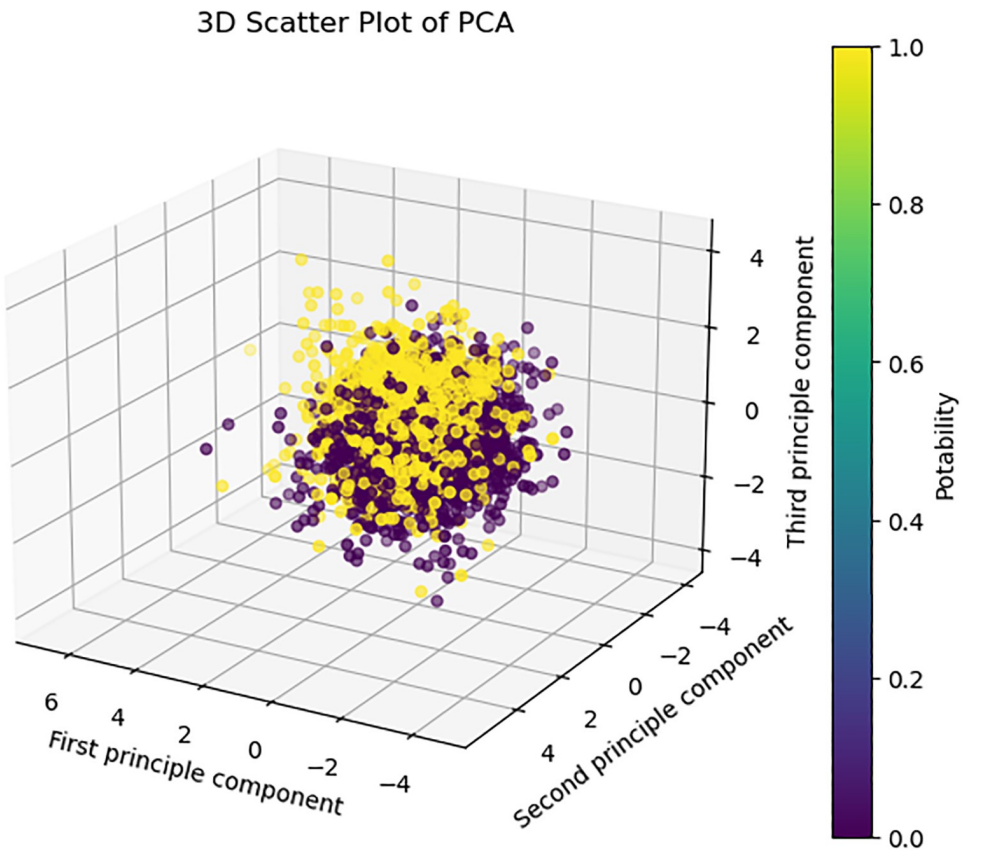

Fig 11. Plotted the PCA concerning the first three dimensions where our main factor is water potability(o or 1); 1 is yellow, and 0 is purple.

https://doi.org/10.1371/journal.pwat.0000259.g011

deteriorates as the number of features increases. To overcome this issue, we used the Principal Component analysis.

Several challenges and limitations were encountered during the PCA process. For overcoming the challenges e ensured the dataset was properly standardized, as PCA requires data to be normalized to function correctly. We Handled missing values using the EM algorithm for better prediction, as imputation methods can affect the variance structure of the data, potentially distorting PCA results. Additionally, we have selected the optimal number of principal components to retain was complex, maintaining the balance between explaining sufficient variance and avoiding overfitting.

Principal Component Analysis (PCA) was chosen as the dimensionality reduction technique due to its ability to maximize variance and retain the most significant features of the data. In our analysis, we have a total of 9 parameters. PCA reduces the dimensions to 6, and also, based on the parameter variance and eigenvalues, we have used the first 6 dimensions, which carry nearly about 66% of the information of our dataset, which is sufficient is also shown in the Fig 10. Now, the new dataset has 6 parameters, Dim 1 to Dim 6, and the target variable water potability. The findings of this study regarding the impact of dimensionality reduction on predictive accuracy are generalizable to other datasets and real-world applications involving similar machine-learning tasks.

After applying PCA and reducing the dataset to six dimensions, we found a general decrease in accuracy across all classifiers. Logistic regression achieved the highest accuracy of

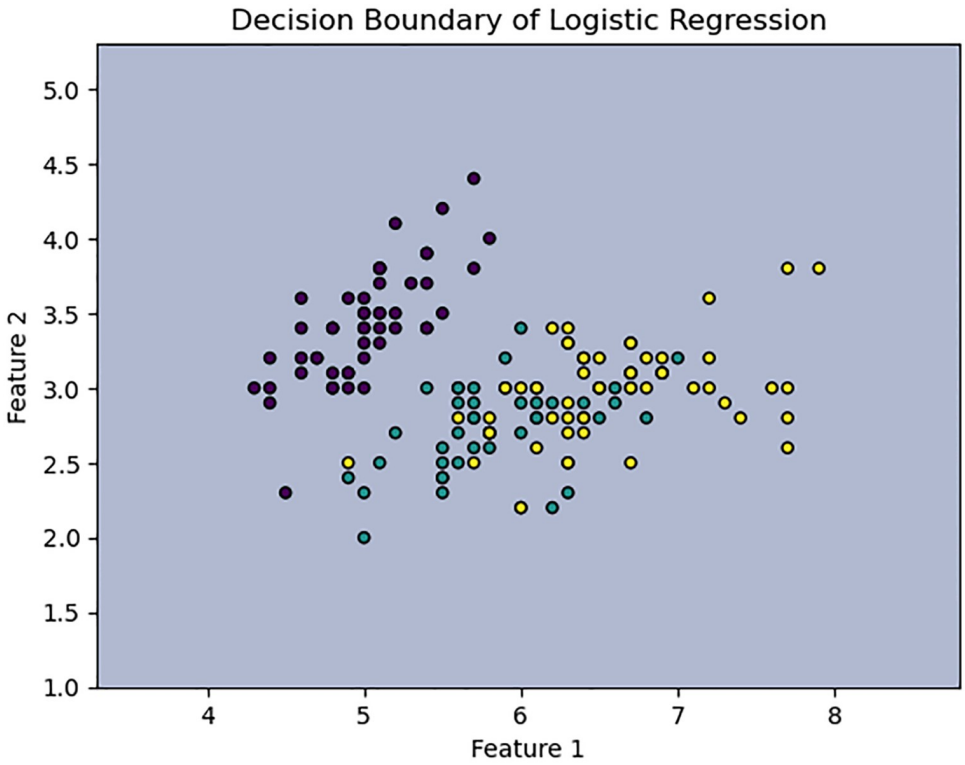

Fig 12. Although a Support Vector Machine (SVM) algorithm achieved the highest accuracy score of 99.89%, we utilized a Logistic Regression partition plot with two features to provide a general overview. While all nine features contributed to the final accuracy score, visualizing a decision boundary in nine dimensions is not feasible.

https://doi.org/10.1371/journal.pwat.0000259.g012

99.89%, followed by Support Vector Machine, XGBoost, and Nearest Neighbour, with accuracy scores of 99%, 99.28%, and 98.88%.

To ensure model reliability and robustness, seven classification algorithms were initially applied directly to the original dataset to establish a baseline. Following this, Principal Component Analysis (PCA) was used to reduce dimensionality to six principal components, simplifying the data while retaining crucial variance. After PCA, models were re-evaluated, resulting in significantly increased accuracy across all classifiers, nearing 100%. This approach validated the efficacy of dimensionality reduction and emphasized the importance of method selection for improved predictive accuracy in water potability assessment.

The accuracy of all classifiers increased significantly after applying PCA primarily due to the dimensionality reduction that PCA provides. By transforming the original high-dimensional data into a lower-dimensional space, PCA captures the most important variance in the data, thereby eliminating noise and redundant features. This simplification makes it easier for classifiers to detect patterns and relationships within the data, improving their performance. Additionally, reducing the number of dimensions helps to mitigate the curse of dimensionality, where too many features can lead to overfitting and poor generalization for new data.

Ethical implications and potential biases in machine learning for water potability prediction

Our findings suggest that while PCA can help reduce computational complexity and potentially improve the performance of specific classifiers, it may not always lead to better accuracy

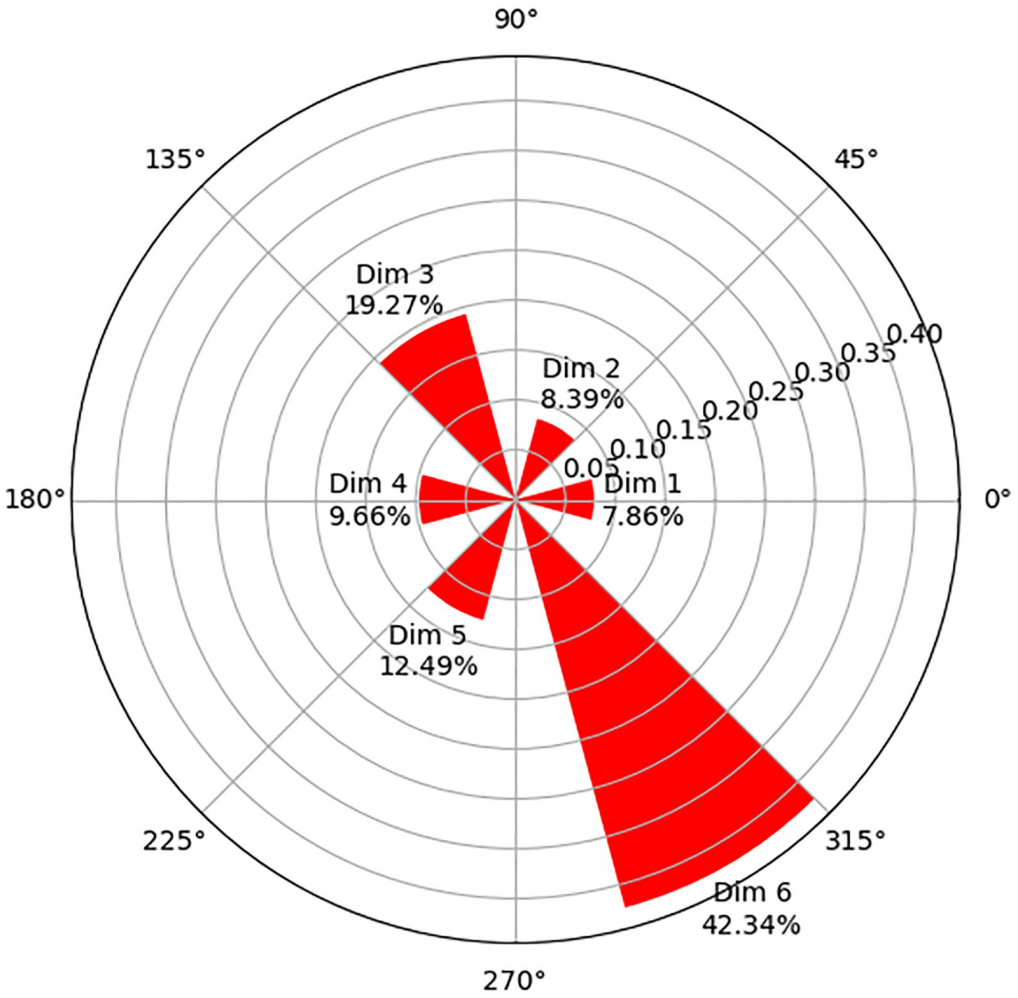

Fig 13. Feature importances of logistic regression based on PCA's first six dimensions or components. The plot displays the importance percentages of each feature, representing their contribution to the logistic regression model's decision-making process. The picture shows all the features and importance of the dimensions in percentages.

https://doi.org/10.1371/journal.pwat.0000259.g013

scores. The choice of machine learning algorithm and its compatibility with the dataset characteristics remain crucial factors in achieving accurate predictions.

Machine learning for water potability prediction involves several ethical considerations and potential biases. A significant concern is the fairness of the models, as algorithms can perpetuate biases present in training data. This can lead to inaccurate predictions for underrepresented regions, impacting equity in water safety[PMID: 33562175].

Transparency is another issue. Many machine learning models are black-box systems, making it difficult for users to understand how decisions are made. This can reduce trust and hinder the acceptance of these models[PMID: 37993801].

Data privacy and security are critical, given that water quality datasets may contain sensitive information. Ensuring this data is protected is essential for maintaining public trust and preventing misuse.

Lastly, accountability must be clear. It is vital to determine who is responsible if a model's predictions lead to public health risks.

Addressing these ethical implications and biases is crucial for the responsible use of machine learning in water potability prediction.

Conclusion

In conclusion, our study underscores the importance of carefully selecting appropriate machine learning techniques and preprocessing methods for water potability prediction tasks. Further research could explore additional dimensionality reduction techniques or ensemble methods to enhance predictive performance while minimizing computational burden. Ultimately, our findings advance water quality assessment methodologies and support informed decision-making in ensuring safe and potable water resources.

Future works

Based on the findings of this study, future research directions or follow-up studies could explore the application of machine learning techniques in predicting water potability based on water quality attributes in different geographical locations or under varying environmental conditions. Additionally, further investigation could delve into optimizing the PCA process, exploring different numbers of principal components to achieve the highest predictive accuracy while maintaining computational efficiency. Moreover, researchers could examine the robustness of the predictive models developed in this study by validating them with independent datasets or real-world water quality data collected from diverse sources. Finally, there is potential for exploring the integration of other advanced machine learning algorithms or ensemble methods to further enhance predictive performance and generalize the models across different water quality scenarios.

Author Contributions

Conceptualization: Debashis Chatterjee, Prithwish Ghosh.

Data curation: Prithwish Ghosh.

Formal analysis: Prithwish Ghosh.

Funding acquisition: Prithwish Ghosh.

Investigation: Debashis Chatterjee, Prithwish Ghosh.

Methodology: Debashis Chatterjee, Prithwish Ghosh.

Project administration: Prithwish Ghosh.

Resources: Prithwish Ghosh.

Software: Prithwish Ghosh.

Supervision: Debashis Chatterjee, Amlan Banerjee, Shiladri Shekhar Das.

Validation: Debashis Chatterjee, Prithwish Ghosh, Amlan Banerjee.

Visualization: Prithwish Ghosh.

Writing – original draft: Debashis Chatterjee, Prithwish Ghosh.

References

1. Organization WH. Guidelines for drinking-water quality; 2023. Available from: https://www.who.int/teams/environment-climate-change-and-health/water-sanitation-and-health/water-safety-and-quality/drinking-water-quality-guidelines.

2. for Disease Control C, Prevention. Waterborne Diseases; 2023. Available from: https://www.cdc.gov/healthywater/surveillance/burden/index.html.
3. Bank W. The Socioeconomic Benefits of Water Security; 2023. Available from: https://blogs.worldbank.org/water/why-water-security-our-most-urgent-challenge-today.
4. Programme UNE. Environmental Integrity and Water Quality; 2023. Available from: https://www.unep.org/topics/fresh-water/about-fresh-water.
5. Prasad A, Singh S, Mall MM. Drinking water potability prediction using machine learning approaches: a case study of Indian rivers. Environmental Monitoring and Assessment. 2023; 195(2):43. https://doi.org/10.1007/s10661-022-09995-5
6. Islam DMA, Sultana S, Kabir MS. Machine Learning for Water Quality Monitoring and Prediction: A Review. Water Resources Management. 2020; 34(15):4577–4607. https://doi.org/10.1007/s11269-020-02702-9
7. Nations U. Drinking Water Scarcity; 2023. Available from: https://www.un.org/en/global-issues/water.
8. Agency UEP. Climate Change and Water Quality; 2023. Available from: https://www.epa.gov/arc-x/climate-impacts-water-quality.
9. Shannon MAea. The Need for Decentralized Water Treatment Systems in Developing Countries. Desalination. 2019; 452:113–120. https://doi.org/10.1016/j.desal.2018.12.030
10. Wu Jea. Rapid Water Quality Assessment Using Machine Learning for Disaster Response. Hydrology and Earth System Sciences; 26(10):3331–33.
11. Zhu M, Wang J, Yang X, Zhang Y, Zhang L, Ren H, et al. A review of the application of machine learning in water quality evaluation. Eco-Environment & Health. 2022;. https://doi.org/10.1016/j.eehl.2022.06.001 PMID: 38075524
12. Ghosh H, Tusher MA, Rahat IS, Khasim S, Mohanty SN. Water Quality Assessment Through Predictive Machine Learning. In: International Conference on Intelligent Computing and Networking. Springer; 2023. p. 77–88.
13. Wang X, Li Y, Qiao Q, Tavares A, Liang Y. Water quality prediction based on machine learning and comprehensive weighting methods. Entropy. 2023; 25(8):1186. https://doi.org/10.3390/e25081186 PMID: 37628216
14. Mondal A, Dubey SS. Machine Learning-based Water Potability Prediction: Model Evaluation, and Hyperparameter Optimization;.
15. Nasir N, Kansal A, Alshaltone O, Barneih F, Sameer M, Shanableh A, et al. Water quality classification using machine learning algorithms. Journal of Water Process Engineering. 2022; 48:102920. https://doi.org/10.1016/j.jwpe.2022.102920
16. Bedi S, Samal A, Ray C, Snow D. Comparative evaluation of machine learning models for groundwater quality assessment. Environmental Monitoring and Assessment. 2020; 192:1–23. https://doi.org/10.1007/s10661-020-08695-3 PMID: 33219864
17. Huang R, Ma C, Ma J, Huangfu X, He Q. Machine learning in natural and engineered water systems. Water Research. 2021; 205:117666. https://doi.org/10.1016/j.watres.2021.117666 PMID: 34560616
18. Bui DT, Khosravi K, Tiefenbacher J, Nguyen H, Kazakis N. Improving prediction of water quality indices using novel hybrid machine-learning algorithms. Science of the Total Environment. 2020; 721:137612. https://doi.org/10.1016/j.scitotenv.2020.137612 PMID: 32169637
19. Dilmi S, Ladjal M. A novel approach for water quality classification based on the integration of deep learning and feature extraction techniques. Chemometrics and Intelligent Laboratory Systems. 2021; 214:104329. https://doi.org/10.1016/j.chemolab.2021.104329
20. Nair J, Vijayamohanan MS. A Machine Learning-Based Water Potability Prediction Model by Using Synthetic Minority Oversampling Technique and Explainable AI. Computational Intelligence and Neuroscience. 2022; 2022:9283293. https://doi.org/10.1155/2022/9283293
21. Rani A, Malik MA, Khan SA. The Quality of Drinkable Water using Machine Learning Techniques. International Journal of Advanced Research. 2021; 9(1):576–581. https://doi.org/10.14314/ijar.v9i1.13172
22. Li Yea. Water quality prediction using machine learning models based on grid search method. In: Advances in Intelligent Systems and Computing. vol. 1818. Springer, Singapore; 2023. p. 45–55.
23. Uddin K, Khan MA, Ryu JH. Water Potability Prediction Using Machine Learning. Sustainability. 2019; 11(8):2238. https://doi.org/10.3390/su11082238
24. Habib G, Qureshi S. Compressed lightweight deep learning models for resource-constrained Internet of things devices in the healthcare sector. Expert Systems. 2023; p. e13269. https://doi.org/10.1111/exsy.13269
25. Data;. https://www.kaggle.com/datasets/uom190346a/water-quality-and-potability.

26. Water, sanitation and hygiene (WASH);. https://www.who.int/health-topics/water-sanitation-and-hygiene-wash#tab=tab_1.
27. Patel S, Shah K, Vaghela S, Aglodiya M, Bhattad R. Water Potability Prediction Using Machine Learning; 2023.
28. Drinking water;. https://environment.ec.europa.eu/topics/water/drinking-water_en#:~:text=The%20recast%20Drinking%20Water%20Directive,into%20force%20in%20January%202021.
29. Ng SK, Krishnan T, McLachlan GJ. The EM algorithm. Handbook of computational statistics: concepts and methods. 2012; p. 139–172. https://doi.org/10.1007/978-3-642-21551-3_6
30. Moon TK. The expectation-maximization algorithm. IEEE Signal processing magazine. 1996; 13(6):47–60. https://doi.org/10.1109/79.543975
31. Wold S, Esbensen K, Geladi P. Principal component analysis; 1987.
32. Labrín C, Urdinez F. Principal component analysis; 2020.
33. Greenacre M, Groenen PJ, Hastie T, dEnza AI, Markos A, Tuzhilina E. Principal component analysis; 2022.
34. Wright RE. Logistic regression.; 1995.
35. Sperandei S. Understanding logistic regression analysis. Biochemia medica. 2014; 24(1):12–18. https://doi.org/10.11613/BM.2014.003 PMID: 24627710
36. Hosmer DW Jr, Lemeshow S, Sturdivant RX. Applied logistic regression. vol. 398. John Wiley & Sons; 2013.
37. Belgiu M, Drăgut L. Random forest in remote sensing: A review of applications and future directions. ISPRS journal of photogrammetry and remote sensing. 2016; 114:24–31. https://doi.org/10.1016/j.isprsjprs.2016.01.011
38. Pal M. Random forest classifier for remote sensing classification. International journal of remote sensing. 2005; 26(1):217–222. https://doi.org/10.1080/01431160412331269698
39. Hearst MA, Dumais ST, Osuna E, Platt J, Scholkopf B. Support vector machines. IEEE Intelligent Systems and their applications. 1998; 13(4):18–28. https://doi.org/10.1109/5254.708428
40. Suthaharan S, Suthaharan S. Support vector machine. Machine learning models and algorithms for big data classification: thinking with examples for effective learning. 2016; p. 207–235. https://doi.org/10.1007/978-1-4899-7641-3_9
41. Sharmila B, Nagapadma R. Intrusion detection system using naive bayes algorithm. In: 2019 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE). IEEE; 2019. p. 1–4.
42. Chen S, Webb GI, Liu L, Ma X. A novel selective naive Bayes algorithm. Knowledge-Based Systems. 2020; 192:105361. https://doi.org/10.1016/j.knosys.2019.105361
43. Berrar D. Bayes theorem and naive Bayes classifier. Encyclopedia of bioinformatics and computational biology: ABC of bioinformatics. 2018; 403:412.
44. Friedman JH, Baskett F, Shustek LJ. An algorithm for finding nearest neighbors. IEEE Transactions on computers. 1975; 100(10):1000–1006. https://doi.org/10.1109/T-C.1975.224110
45. Taunk K, De S, Verma S, Swetapadma A. A brief review of nearest neighbor algorithm for learning and classification. In: 2019 international conference on intelligent computing and control systems (ICCS). IEEE; 2019. p. 1255–1260.
46. Chen T, Guestrin C. Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining; 2016. p. 785–794.
47. Mitchell R, Frank E. Accelerating the XGBoost algorithm using GPU computing. PeerJ Computer Science. 2017; 3:e127. https://doi.org/10.7717/peerj-cs.127
48. Ogunleye A, Wang QG. XGBoost model for chronic kidney disease diagnosis. IEEE/ACM transactions on computational biology and bioinformatics. 2019; 17(6):2131–2140. https://doi.org/10.1109/TCBB.2019.2911071
49. Ghosh P. Breast Cancer Wisconsin (Diagnostic) Prediction;.