Water Quality Prediction Using Machine Learning

Dr. Sanjeev Singh[1], Dr. Dilkeshwar Pandey[2] Shashwat Singh[3], Anurag Shrivastava[3], Pankaj Kumar[3], Prajwal Upman[3]

[1] Supervisor, Department of Civil Engineering, KIET Group of Institutions, Delhi-NCR, Ghaziabad.

[2]Supervisor, Department of Computer Science & Engineering, KIET Group of Institutions, Delhi-NCR, Ghaziabad.

[3]Undergraduate Scholars, Department of Civil Engineering, KIET Group of Institutions, Delhi-NCR, Ghaziabad

doi: 10.48047/ecb/2023.12.si6.138

1. *Introduction*

Water is a resource that is crucial for the survival of most living creatures, including humans, and is vital for sustaining life. To ensure the survival of these organisms, it is crucial to make sure that the water quality is enough. There are limits to the amount of pollution aquatic species can tolerate; exceeding these limits might negatively impact their survival and endanger their very existence. There are certain standards of quality that may be used to gauge the caliber of different ambient water sources including rivers, lakes, and streams. Additionally, diverse applications call for water that satisfies certain requirements, such as irrigation water, which must not be overly salty and be free of toxins that are damaging to soil and plants to maintain the ecosystems' natural balance. Additionally, certain qualities are required due to the nature of the water used in industrial processes to meet their unique requirements. Although certain freshwater resources, such as ground and surface water, are naturally occurring and inexpensive, pollution of these resources may also be brought on by industrial activity and human activities. Because it provides information that is crucial for making decisions on the management and conservation of water, the difficulty of forecasting water quality is significant in the field of environmental science. The time-consuming and resource-intensive quality prediction has been carried out using empirical models or simulations based on physical principles, both of which may take a lot of resources. On the other hand, the process of developing precise and successful models for forecasting water quality has gained more attention considering recent advances in machine learning. This work explores how machine learning algorithms may be used to anticipate several aquatic systems' water quality indicators, including pH, dissolved oxygen concentration, and turbidity. Along with a review of the pertinent research on this subject, a discussion of the drawbacks and restrictions of using machine learning for water quality prediction, and an analysis of the relevant literature, we provide case examples illuminating the effectiveness of machine learning models in predicting water quality parameters. Our results suggest that machine learning has the potential to change the field of water quality prediction, enabling more precise and effective management of water resources. The potential for machine learning to change the field of water quality prediction might make this achievable.

2 *Literature Review*

A growing area of research focuses on the use of various machine learning algorithms to predict the

water quality index (WQI) and water quality class (WQC) based on various water quality parameters such as turbidity, total suspended solids (TSS), and dissolved oxygen (DO). This field of study is referred to as "water quality prediction using

machine learning." Researchers have used many datasets originating from a range of geographies and sources of water to train and assess various algorithms. The study "Machine Learning-Based Ensemble Prediction of Water-Quality Variables" is one example. Using data from three separate water bodies in the Midwest, "Using Feature-Level and Decision-Level Fusion with Proximal Remote Sensing" illustrated the effectiveness of machine learning regression approaches and decision-level fusion for forecasting water-quality characteristics. The title of the study was "Machine Learning-Based Ensemble Prediction of Water-Quality Variables." In a similar vein, the paper "Emulating process-based water quality modeling in water source reservoirs using Machine Learning" used data gathered from Norway's Brusdalsvatnet Lake to show that a Machine Learning (ML) model, more specifically the Long Short-Term Memory (LSTM), can be a viable replacement for process-based hydrodynamic and water quality models when it comes to the management of water sources. The water quality of the lake was simulated using these models. A dataset kept by the Central Pollution Control Board of India (CPCB) was used in the research paper titled "Water Quality Prediction Using Machine Learning" to test how effectively different machine learning algorithms could predict water quality.

The paper "Performance of machine learning methods in predicting water quality index based on the irregular data set: application on Illizi Region (Algerian Southeast)" also assessed eight artificial intelligence algorithms to produce WQI prediction in the Illizi region, southeast Algeria, using data from the Directorate of Water Resources (DRE) of the State of Illizi. The State of Illizi's Directorate of Water Resources provided the information.

3 ***METHODOLOGIES***

The following is a rundown of each process that goes into the production of our model:

3.1 Problem Identification: The identification of the issue statement is the task at hand in this stage. The challenge at hand is the prediction of water quality using machine learning.

3.2 Data Extraction*:* The process of extracting data includes collecting information from a single source or numerous sources so that it may be analyzed, stored, or processed at a different place. In the context of the situation at hand, we received our data from the website.

https://www.kaggle.com/datasets/adityakadiwal/waterpotabiliy

3.3 Data Preprocessing: Processing the data plays an important part in improving the quality of the analysis done on the data. The term "data processing" refers to the act of gathering and manipulating various components of data to create information that is both usable and relevant.

3.3.1 Dealing with missing values: Multiple strategies are available to replace missing values in data. Using means as a strategy to handle missing values in numeric columns is the one that sees the most usage. In cases when there are outliers among the data, however, means might not be the best choice. As a result, addressing outliers is necessary before using the mean replacement approach.

3.3.2 ***Water Quality Index (WQI):*** **"**The Water Quality Index (WQI)" is an all-encompassing measurement of water quality that considers a variety of factors. In the past, the WQI calculation has relied on the usage of nine different factors. Formula (1) is typically utilized in practice when attempting to ascertain the WQI.

$$\boldsymbol{WQI} = \frac{\sum_{i=1}^{N} \boldsymbol{qi} \times \boldsymbol{wi}}{\sum_{i=1}^{N} \boldsymbol{wi}} \qquad \textbf{(1)}$$

3.3.3 ***Data Visualization: The*** process of displaying data in a visual form, with the goal of making it easier to see patterns, correlations, and trends within the data (Fig. 2), is referred to as data visualization. matrix, we can identify patterns and establish dependent features by making use of features that are readily available.

3.3.4 ***Correlation Analysis:*** By evaluating the correlation coefficients, a correlation matrix may be a helpful tool for determining the probable correlations that exist between several different parameters. A table containing all the potential value pairings is shown. Through the examination of the heatmap that was produced by the correlation The connection between all the traits is

presented in the study's Figure 3, and it shows that the link between them is not very strong at all. As a result, there is no requirement to get rid of any of the characteristics contained in the dataset.

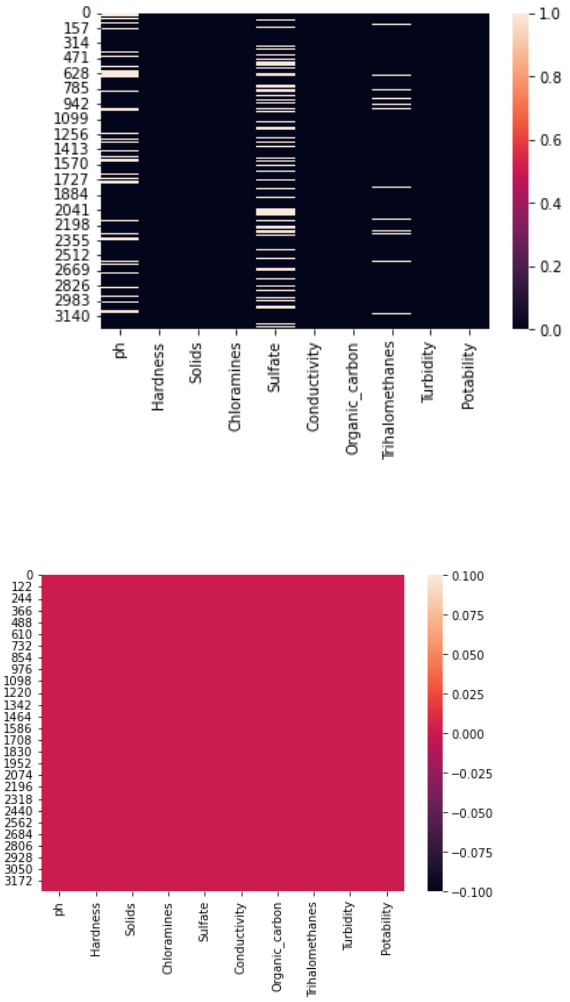


Fig 1 Heatmap before and after removing missing values.

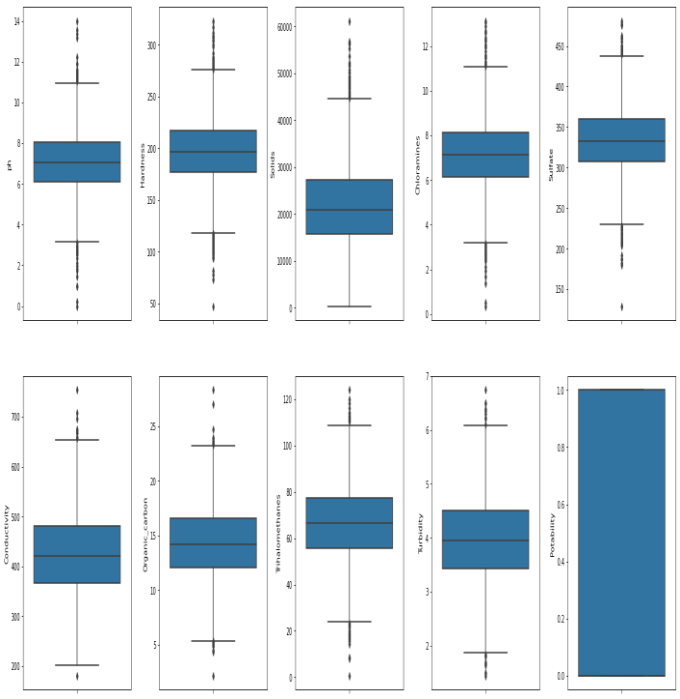
The heatmap that may be found below (Fig. 4) displays the correlation that exists between the various characteristics.

3.3.5 ***Data Splitting:*** Before analyzing the performance of the machine learning model, the data must be divided into a training set and a testing set. It was decided to divide the dataset into two subsets, with 33% of the data being used for testing and 67% being used for training. The goal is to develop a connection between the independent and dependent parameters for the model to provide predictions or draw conclusions. The effectiveness of the machine learning algorithm is then determined using the outcomes of the tests. Thanks to data partitioning, it can assess the model's performance before using it to simulate real-world scenarios by computing accuracy measures.

4 ***Prediction of Water Potability using ML algorithms.***

4.1 ***Algorithm:*** In order to accomplish this goal, machine learning strategies were utilized in the estimation of the water's potability. Both regression and classification were accomplished using algorithms. During our investigation, we utilized several algorithms.

4.1.1 ***Logistic Regression:*** The purpose of this regression model is to arrive at an estimate of the likelihood of a specific outcome by analyzing the values of the variables that are considered independent. Logical regression, as opposed to linear regression, models the logarithm of the probabilities of the outcome variable. Linear regression is used for analyzing data with continuous dependent variables. This transformation makes it possible to describe the dependent variable as a function of the independent variables while preserving a constrained range of values that is limited to the range between 0 and 1.



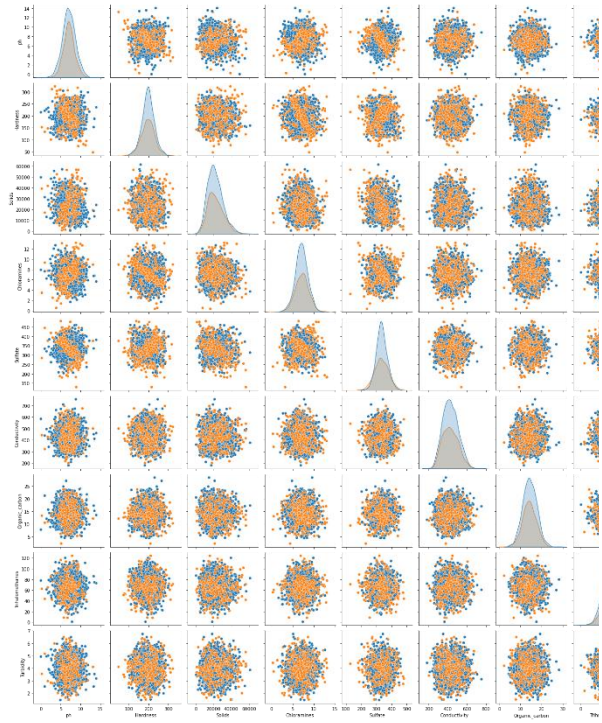


Fig 2 Visualizing data and checking for outliers.

As shown in (2), the sigmoid function is utilized in the process of doing analysis in logistic regression.

$$g(z) = \frac{1}{1 + e\hat{} - x} \qquad (2)$$

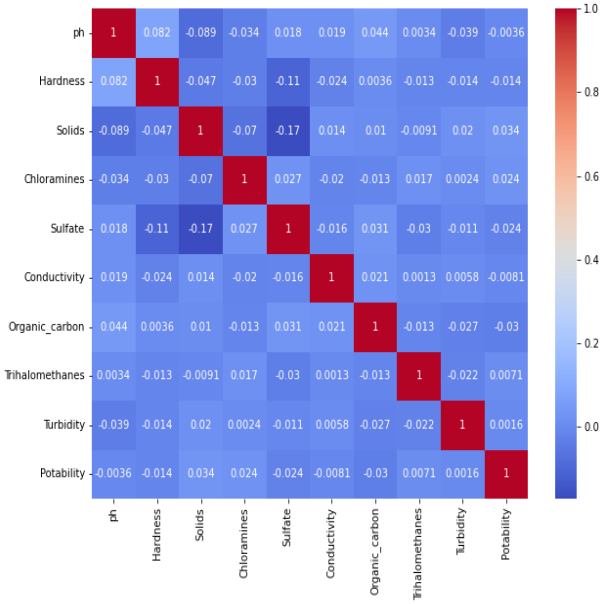


Fig 3 Correlation Heatmap

4.1.2 ***Support Vector Machine:*** Using, Support Vector Machines (SVM) classifies data, performs regression analysis, and identifies outliers. Support vector machines (SVM) are used to find a hyperplane that successfully splits the data into several groups. The path or plane that optimizes the distance between the two classes is known as a hyperplane. The margin is the distance between each class's closest-to-the-hyperplane data points and the hyperplane itself.

4.1.3 ***Decision Tree Classifier:*** It is a type of supervised learning algorithm that finds widespread use in the field of machine learning for the purpose of addressing classification issues. It constructs a tree-like representation of the decisions and the various outcomes associated with each option. The tree is organized such that each node inside the tree represents a feature, and each branch organizes the values associated with that feature. The classes or categories that the input instances are a part of are represented by the tree's leaves.

4.1.4 ***Random Forest Classifier:*** This Classifier constructs a series of decision trees by employing a random subset of the training data and a random subset of the input characteristics at each node of the tree. This allows the tree to predict new data more accurately. This randomization helps to prevent overfitting, which in turn improves the model's performance in terms of its ability to generalize. Each decision tree in the forest is trained on its own, and the final forecast is determined either by taking the average of all the trees' predictions or by deciding which predictions received the most votes.

4.1.5 ***XGBoost Classifier:*** Extreme Gradient Boosting, often known as XGBoost, is the name of a scalable machine learning program that is distributed over multiple servers. It employs the gradient-boosted decision tree (GBDT) algorithm. Regression, classification, and ranking are some of the issues it can address. It is the most popular machine learning framework. Additionally, parallel tree boosting is supported.

4.1.6 ***AdaBoost Classifier:*** The boosting approach known as "Adaptive Boosting," which is shortened as "AdaBoost," is utilized in machine learning as part of an Ensemble Method. The word "Adaptive

Boosting" is sometimes abbreviated as "AdaBoost." It is given the moniker "Adaptive Boosting" since the weights are reallocated to each instance, with bigger weights being added to examples that were wrongly classified. The reason for this is that the overall accuracy of the classification may be improved.

4.1.7 ***K Neighbors:*** K-Nearest Neighbor is a machine learning algorithm that is one of the most basic since it is based on the idea of supervised learning, which is a learning method. The K-NN approach operates under the presumption that the new case or data is comparable to the cases that already exist. A new instance is assigned to the categories that are the most comparable to the categories that are already available to the user. The k-NN algorithm is responsible for remembering all the data that is available and determining how to classify incoming data points based on the degree to which they are like the data that has come before. This means that if new data is collected, it will be easily capable of being classified into an appropriate suite category by making use of the K-NN approach.

4.2 ***Measure:***The following list outlines the criteria that served as the basis for the evaluation of the model's performance and can be found here.

4.2.1 ***Precision:*** refers to the proportion of occurrences inside a classifier that have been successfully categorized, as a comparison to the total number of contexts that have been interpreted. Equation (3) is applied to determine TP, which stands for "positive class," whereas FP refers to the amount of precision connected with false alarms. Both concepts are related to accuracy.

$$Precision = \frac{TP}{TP+FP}$$

(3)

4.2.2 ***Accuracy:*** this is the statistic that requires the least amount of explanation because it reflects the proportion of instances that have been correctly categorized in comparison to the total number of examples that are contained in the dataset. Accuracy is calculated by dividing the total number of occurrences in the dataset by the number of true positives and true negatives in the dataset (true positives plus true negatives plus false positives plus false negatives) (Equation 4).

$$\boldsymbol{Accuracy = \frac{TP + TN}{TP + FP + TN + FN}}$$

(4)

4.2.3 ***Recall:*** It can also be referred to as the true positive rate or sensitivity. It determines the proportion of real positives in the dataset that are actually accurate positives by measuring the percentage of true positives. It is determined by taking TP and dividing that number by (TP + FN) (Equation 5). When we wish to identify all positive cases while minimizing the possibility of false negatives, recall is helpful.

$$\boldsymbol{Recall = \frac{TP}{TP+FN}}$$

(5)

4.2.4 ***F1 Score:*** this is the optimal balance between accuracy and accessibility. It strikes a compromise between precision and recall, making it a useful statistic in situations in which consideration should be given to both aspects. It is determined in the manner depicted in Equation 6. The range of possible F1 scores is from 0 (worst) to 1 (best).

$$\boldsymbol{F1\ Score = \frac{2\times Precision\times Recall}{Precision\times Recall}}$$

(6)

4.2.5 ***Results for Algorithms:*** In order to construct the regression and classifier model based on the dataset, we made use of all the methods that were discussed before. The hyperparameter tweaking approach was utilized throughout the assessment of the model.

	Model	Accuracy score
1	SVM	0.688540
2	XGBoost	0.670980
3	KNeighbours	0.653420
4	Decision Tree	0.645102
5	AdaBoost	0.634011

6	Logistic Regression	0.628466
7	Random Forest	0.628466

Table1 comparison of different classifiers

4.4 *Hyperparameter Tunning*

"Hyperparameter tuning" is the process of identifying, for a certain machine learning model, the optimal combination of hyperparameters to improve the performance of the model. This process is referred to as "hyperparameter tuning." One example of a hyperparameter is the learning rate. Other examples include the batch size, the number of hidden layers, and the number of neurons in each hidden layer. These model parameters can't be taught during the training phase; thus, they must be given beforehand for training to get underway. There are several methods available for tuning hyperparameters. Some examples of these methods are manual tuning, grid search, random search, and Bayesian optimization.

4.4.1 *GridSearchCV:* GridSearchCV functions by first having the user provide a grid of potential hyperparameters, and then searching through each combination of those hyperparameters in the grid in an exhaustive manner. GridSearchCV conducts a cross-validation test on the data used for training to evaluate the performance of the model for each possible combination of hyperparameters. The hyperparameters that lead to the best performance are the ones that are ultimately selected to be the optimum hyperparameters.

4.4.2 *RandomizedSearchCV:* works by first describing a probability distribution for each hyperparameter, and then, constructing a set of hyperparameter combinations, randomly sampling from each of those distributions. To assess how well a model works, RandomizedSearchCV uses a process called cross-validation on the data used for training it. This is done for each possible combination. The hyperparameters that lead to the best performance are the ones that are selected to be the optimum ones.

4.4.3 *Bayesian optimization:* in Bayesian optimization, you make use of probabilistic models to direct your search for the optimal combination of hyperparameters, and this helps to ensure that you find the best possible results. This approach is more efficient than random search and grid search because it can discover interesting hyperparameters early in the search process and focus on investigating the most promising parts of the search space. Random search and grid search both focus on searching across the whole search space from beginning to end.

4.4.4 *Results of Hyperparameter Tunning:* After conducting hyperparameter tweaking, the accuracy of classifiers such as RF increases in terms of precision and top scores, but it falls in other scenarios, as shown in Table (4). This contrasts with other situations, in which accuracy worsens.

Model	**Accuracy before Hyperparameter tunning**	**Accuracy after Hyperparameter tunning**	
		Best Score	**Test Score**
SVC	0.688	0.605	0.628
xgboost	0.670	0.649	0.667
KNN	0.653	0.637	0.637
DT	0.645	0.632	0.63
Adaboost	0.634	0.637	0.64
Logestic Regression	0.628	0.605	0.6

Table 2 Results of Hyperparameter Tuning

5 *Results*

In this study, the capacity of five distinct machine learning algorithms to predict the separate components of a dataset containing information about water quality was evaluated, examined, and compared. To achieve this objective, variables from the most well-known datasets, such as pH, hardness, solids, electrical conductivity (EC), and turbidity, were gathered for collection. According to the findings, the models that were applied had a performance level that was sufficient for predicting water quality measurements (Table 3). Yet RF and

XGB, on the other hand, have the greatest levels of performance.

Model Name	Class label	Classification Report Precision	Recall	F1score	Accuracy
SVM	Not Potable	0.69	0.82	0.75	0.60
	Potable	0.55	0.37	0.44	
XGBoost	Not Potable	0.68	0.89	0.77	0.64
	Potable	0.61	0.31	0.41	
KNeighbours	Not Potable	0.69	0.82	0.75	0.63
	Potable	0.55	0.37	0.44	
Decision Tree	Not Potable	0.66	0.90	0.76	0.63
	Potable	0.56	0.22	0.32	
AdaBoost	Not Potable	0.63	0.99	0.77	0.62
	Potable	0.62	0.04	0.07	
Logistic Regression	Not Potable	0.63	1.00	0.77	0.60
	Potable	0.00	0.00	0.00	
Random Forest	Not Potable	0.63	1.00	0.77	0.67
	Potable	0.00	0.00	0.00	

Table 3 Classification report for different ML Algorithm

References

[1]https://www.kaggle.com/datasets/adityakadiwal/water-potabilit

[2] Patel, J., Amipara, C., Ahanger, T. A., Ladhva, K., Gupta, R. K., Alsaab, H. O., Althobaiti, Y. S., & Ratna, R. (2022). A Machine Learning-Based Water Potability Prediction Model by Using Synthetic Minority Oversampling Technique and Explainable AI. *Computational Intelligence and Neuroscience*, *2022*, 1–15. https://doi.org/10.1155/2022/9283293

[3] Aldhyani, T. H. H., Al-Yaari, M., Alkahtani, H., & Maashi, M. S. (2020). Water Quality Prediction Using Artificial Intelligence Algorithms. *Applied Bionics and Biomechanics*, *2020*, 1–12. https://doi.org/10.1155/2020/6659314

[4] Mohammed, H., Tornyeviadzi, H. M., & Seidu, R. (2022). Emulating process-based water quality modelling in water source reservoirs using machine learning. *Journal of Hydrology*, *609*, 127675. https://doi.org/10.1016/j.jhydrol.2022.127675

[5] Peterson, K., Sidike, P., Sidike, P., Hasenmueller, E. A., Sloan, J. M., & Knouft, J. H. (2019). Machine Learning-Based Ensemble

[6] Fu, Zhao, "Water Quality Prediction Based on Machine Learning Techniques" (2020). UNLV Theses, Dissertations, Professional Papers, and Capstones.

Prediction of Water-quality Variables Using Feature-level and Decision-level Fusion with Proximal Remote Sensing. *Photogrammetric Engineering and Remote Sensing*, *85*(4), 269–280. https://doi.org/10.14358/pers.85.4.269

[9] Kouadri, S., Elbeltagi, A., Islam, A. R. M. T., & Kateb, S. (2021). Performance of machine learning methods in predicting water quality index based on irregular data set: application on Illizi region (Algerian southeast). *Applied WaterScience*,*11*(12). https://doi.org/10.1007/s13201-021-01528-9

[10] Ahmed, U., Mumtaz, R., Anwar, H., Shah, A. A., Irfan, R., & García-Nieto, J. (2019). Efficient Water Quality Prediction Using Supervised Machine Learning. *Water*, *11*(11), 2210. https://doi.org/10.3390/w11112210

[11] Wang, R., Kim, J., & Li, M. (2021). Predicting stream water quality under different urban development pattern scenarios with an interpretable machine learning approach. Science of the Total Environment, 761, 144057. https://doi.org/10.1016/j.scitotenv.2020.144057

3994.
[7] Haghiabi, A. H., Nasrolahi, A., & Parsaie, A. (2018). Water quality prediction using machine learning methods. *Water Quality Research Journal of Canada*, *53*(1), 3–13. https://doi.org/10.2166/wqrj.2018.025
[8] Ahmed, U., Mumtaz, R., Anwar, H., Shah, A. A., Irfan, R., & García-Nieto, J. (2019). Efficient Water Quality Prediction Using Supervised Machine Learning. *Water*, *11*(11), 2210. https://doi.org/10.3390/w11112210

[12] Kumar, B., Mukherjee, I., & Singh, U. K. (2016). Water Quality Status of Indian Major Rivers with Reference to Agriculture and Drinking Purposes. *ResearchGate*. https://www.researchgate.net/publication/305493312_Water_Quality_Status_of_Indian_Major_Rivers_with_Reference_to_Agriculture_and_Drinking_Purposes
[13] Nishant Rawat, Mangani Daudi Kazembe, Pradeep Kumar Mishra (2022). Water Quality Prediction using Machine Learning. ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
[14] Sai Sreeja Kurra, Sambangi Geethika Naidu, Sravani Chowdala, Sree Chithra Yellanki, Dr. B. Esther Sunanda (2022). WATER QUALITY PREDICTION USING MACHINE LEARNING. e-ISSN: 2582-5208Impact Factor- 6.752