WATER QUALITY PREDICTION USING MACHINE LEARNING

A Project Report submitted in the partial fulfillment of

the Requirements for the award of the degree

BACHELOR OF TECHNOLOGY
in

COMPUTER SCIENCE AND ENGINEERING

Submitted by

M.SVL.Bharani (21475A0520)

Under the esteemed guidance of

Y. Chandana,M.Tech.

Asst.Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NARASARAOPETA ENGINEERING COLLEGE:NARASARAOPET

(AUTONOMOUS)

Accredited by NAAC with A+ Grade and NBA under Tier -1

Approved by AICTE, New Delhi, Permanently Affiliated to JNTUK, Kakinada

KOTAPPAKONDA ROAD, YALAMANDA VILLAGE, NARASARAOPET- 522601

2023-2024

NARASARAOPETA ENGINEERING COLLEGE:NARASARAOPET
(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project work entitled "WATER QUALITY PREDICTION USING MACHINE LEARNING" is a bonafide work done by M.SVL.Bharani(21475A0520), in partial fulfillment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in the Department of COMPUTER SCIENCE AND ENGINEERING during 2023-2024.

PROJECT GUIDE

Y. Chandana, M.Tech.
Asst. Professor

PROJECT CO- ORDINATOR

Dr. M. Sireesha B.Tech.,M.Tech.,Ph.D
Assoc. Professor

HEAD OF THE DEPARTMENT

Dr. S. N. Tirumala Rao M.Tech., Ph.D
Professor

EXTERNAL EXAMINER

DECLARATION

We declare that this project work is entitled as “WATER QUALITY PREDICTION USING MACHINE LEARNING” is composed by ourselves that the work contain here is our own except where explicitly stated otherwise in the text and that this work has been submitted for any other degree or professional qualification except as specified.

M.SVL.Bharani (21475A0520)

ACKNOWLEDGEMENT

I wish to express my thanks to carious personalities who are responsible for the completion of the project. I am extremely thankful to our beloved chairman Sri **M.V.Koteswara Rao, B.Sc.,**who took keen interest towards me in every effort throughout this course. I owe out sincere gratitude to our beloved principal **Dr.M.Sreenivasa Kumar, M.Tech., Ph.D., MISTE., FIE(I).,** for showing his kind attention and valuable guidance throughout the course.

I express my deep felt gratitude towards **Dr.S.N.Tirumala Rao, M.Tech., Ph.D.,** HoD of CSE department and my guide **Y. Chandana, M.Tech.(Asst. Prof).,** of CSE department whose valuable guidance and unstinting encouragement enable me to accomplish my project successfully in time.
I extend my sincere thanks towards **M. Sireesha, B.Tech.,M.Tech.,Ph.D.** Associate professor & Project coordinator of the project for extending her encouragement. Their profound knowledge and willingness have been a constant source of inspiration for me throughout this project work. I extend my sincere thanks to all other teaching and non-teaching staff to department for their cooperation and encouragement during my B.Tech degree. I have no words to acknowledge thewarm affection, constant inspiration and encouragement that I received from my parents.

I affectionately acknowledge the encouragement received from our friends and those whoinvolved in giving valuable suggestions had clarifying out doubts which had really helped us in successfully completing our project.

By

M.SVL.Bharani (21475A0520)

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

INSTITUTE VISION AND MISSION

INSTITUTION VISION

To emerge as a Centre of excellence in technical education with a blend of effective student centric teaching learning practices as well as research for the transformation of lives and community,

INSTITUTION MISSION

M1: Provide the best class infra-structure to explore the field of engineering and research

M2: Build a passionate and a determined team of faculty with student centric teaching,imbibing experiential, innovative skills

M3: Imbibe lifelong learning skills, entrepreneurial skills and ethical values in students for addressing societal problems

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE ANDENGINEERING

VISION OF THE DEPARTMENT

To become a centre of excellence in nurturing the quality Computer Science & Engineering professionals embedded with software knowledge, aptitude for research and ethical values to cater to the needs of industry and society.

MISSION OF THE DEPARTMENT

The department of Computer Science and Engineering is committed to

M1: Mould the students to become Software Professionals, Researchers and Entrepreneurs by providing advanced laboratories.

M2: Impart high quality professional training to get expertize in modern software tools and technologies to cater to the real time requirements of the Industry.

M3: Inculcate team work and lifelong learning among students with a sense of societal and ethical responsibilities.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Specific Outcomes (PSO's)

PSO1: Apply mathematical and scientific skills in numerous areas of Computer Science and Engineering to design and develop software-based systems.

PSO2: Acquaint module knowledge on emerging trends of the modern era in Computer Science and Engineering

PSO3: Promote novel applications that meet the needs of entrepreneur, environmental and social issues.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Educational Objectives (PEO's)

The graduates of the programme are able to:

PEO1: Apply the knowledge of Mathematics, Science and Engineering fundamentals to identify and solve Computer Science and Engineering problems.

PEO2: Use various software tools and technologies to solve problems related to academia, industry and society.

PEO3: Work with ethical and moral values in the multi-disciplinary teams and can communicate effectively among team members with continuous learning.

PEO4: Pursue higher studies and develop their career in software industry.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Outcomes

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, research literature, and analyzecomplex engineering problems reaching substantiated conclusions using firstprinciples of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complexengineering problems and design system components or processes that meetthe specified needs with appropriate consideration for the public health andsafety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-basedknowledge and research methods including design of experiments, analysis andinterpretation of data, and synthesis of the information to provide validconclusions.

5.Modern tool usage: Create, select, and apply appropriate techniques,resources, and modern engineering and IT tools including prediction andmodeling to complex engineering activities with an understanding of thelimitations.

6. The engineer and society: Apply reasoning informed by the contextualknowledge to assess societal, health, safety, legal and cultural issues and theconsequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of theprofessional engineering solutions in societal and environmental contexts,and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Project Course Outcomes (CO'S):

CO421.1: Analyse the System of Examinations and identify the problem.

CO421.2: Identify and classify the requirements.

CO421.3: Review the Related Literature

CO421.4: Design and Modularize the project

CO421.5: Construct, Integrate, Test and Implement the Project.

CO421.6: Prepare the project Documentation and present the Report using appropriate method.

Course Outcomes – Program Outcomes mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C421.1		√											√		
C421.2	√		√		√								√		
C421.3				√		√	√	√					√		
C421.4			√			√	√	√					√	√	
C421.5					√	√	√	√	√	√	√	√	√	√	√
C421.6									√	√	√		√	√	

Course Outcomes – Program Outcome correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C421.1	2	3											2		
C421.2			2		3								2		
C421.3				2		2	3	3					2		
C421.4			2			1	1	2					3	2	
C421.5					3	3	3	2	3	2	2	1	3	2	1
C421.6									3	2	1		2	3	

Note: The values in the above table represent the level of correlation betweenCO's and PO's:

1. **Low level**

2. **Medium level**

3. **High level**

Project mapping with various courses of Curriculum with Attained PO's:

Name of the course from which principles are applied in this project	Description of the device	Attained PO
C2204.2, C22L3.2	Gathering the requirements and defining the problem, plan to develop a Water Quality Prediction Using Machine Learning	PO1, PO3
CC421.1, C2204.3, C22L3.2	Each and every requirement is critically analyzed, the process model is identified and divided into numpy,pandas,matplotlib,scikit learn,scipy	PO2, PO3
CC421.2, C2204.2,C22L3.3	Logical design is done by using the unified modelling language which involves individual team work	PO3, PO5, PO9
CC421.3, C2204.3,C22L3.2	Each and every module is tested, integrated, and evaluated in our project	PO1, PO5
CC421.4, C2204.4,C22L3.2	Documentation is done by all our four members in the form of a group	PO10
CC421.5, C2204.2,C22L3.3	Each and every phase of the work in group is presented periodically	PO10, PO11
C2202.2, C2203.3, C1206.3, C3204.3, C4110.2	Implementation is done and the project will be handled by the in a municipality that relies on reservoir as its primary source of drinking water.	PO4, PO7
C32SC4.3	The physical design includes hardware components like Sensors and Probes, Data Loggers, Control Systems and Computing Infrastrusture.	PO5, PO6

ABSTRACT

Water is the most important resource for every human being in their daily life. Mainly water is used for agricultural purpose and drinking purpose to use the water in both the aspects it is more important to check the quality of water whether the water is safe to use/drink either it may contain any harmful chemical substances, whether the water may contains the harmful chemicals then the water is determined as unsafe to drink/use.

Based on the consideration of the real time data which includes attributes called Temperature, pH, Chloramines, Sulfates, Hardness, Solids, Organic_carbon, Trihalomethanes, Turbidity, Conductivity, Potability among all those mentioned attributes potability is considered as the target attribute. If the potability is 1 then the water uis safe to drink, whether if the potability is 0 then the water is considered as unsafe to drink.

The application of machine learning in water quality prediction is vast and includes real time monitoring of drinking water sources, early detection of pollution events. By using the power of data and advanced algorithms we can simply understand the dynamics of the water quality.

The findings of the study shows that the accessing of the water quality can becomes more optimal than the other proposed models, and the quality of water can be classified based on the targeted attribute potability. Hence this study helps us to find the accurate water quality information based on the values of the attributes.

INDEX

LIST OF FIGURES

1.INTRODUCTION

1.1 Introduction

The prediction of water quality using machine learning is a critical and evolving area of research that leverages computational models to forecast various parameters indicating the status of water bodies. This approach not only enhances the ability to manage and preserve aquatic environments but also supports public health and industrial practices by providing timely and accurate water quality assessments. Below is an introduction to the topic, outlining the significance, methodology, and potential impacts of using machine learning for water quality prediction.

Water quality is fundamental to ecosystem vitality, human health, and economic development. Contaminants such as chemicals, biological agents, and other harmful substances can degrade water quality, leading to ecosystem damage and health risks such as gastrointestinal illnesses and reproductive problems. Traditional methods of monitoring water quality are often labor-intensive, time-consuming, and cannot provide real-time data critical for urgent decision-making. Thus, predictive modeling using machine learning offers a promising alternative by enabling efficient, accurate, and scalable solutions.

Water quality has a direct impact on public health and the environment. Water is used for various practices, such as drinking, agriculture, and industry. Recently, development of water sports and entertainment has greatly helped to attract tourists . Among various sources of water supply, due to easy access, rivers have been used more frequently for[1] the development of human societies. Using other water resources such as groundwater and seawater sometimes assisted with problems. For example, using groundwater without suitable recharge will lead to land subsidence and using seawater is usually associated with pollution transmission. Therefore, the use of rivers has attracted attention. Several investigations related to rivers around the world have been conducted and a field of engineering named river engineering has been proposed. In river engineering, studies on morphological changes, sediment transport, water quality, and pollution transmission mechanisms are very important. Flow structure, sediment transport and morphology of rivers are investigated in the hydraulics of rivers in river engineering. The study of water quality of rivers is a common theme in earth sciences. To evaluate the quality of rivers two approaches are considered, including measuring the water quality components and defining the mechanism of pollution transmission. Among water quality components, measuring the dissolved oxygen (DO), chemical oxygen demand (COD), biochemical oxygen demand (BOD), electrical conductivity (EC), pH, temperature, K, Na, Mg, etc. have been proposed. To this end, governments have constructed hydrometry stations along rivers that cross from urban areas, agro-industrial projects,

industrial estates, and rivers that join dams' reservoirs. In hydrometry stations, the water quality components are measured and the stage-discharge relation is defined. Obtained values from hydrometry stations contain basic information for feasibility studies and development of water conservation projects. Evaluation of water quality is a basic stage for development of agriculture projects in terms of determination of cropping pattern, type of irrigation system, and systems of water purification for industries. To investigate the mechanism of pollution transmission, in addition to field and laboratory experiments, advanced numerical methods such as computational hydraulic, image processing and GIS methods have been utilized. By reviewing the time history of water quality components, investigators have attempted to estimate future values. Nowadays, by advancing soft computing techniques in most areas of water and environmental engineering, researchers have attempted to accurately analyse time series of water quality components and their internal relation. In this regard, used multilayer perceptron (MLP), radial basis network (RBF) and an adaptive neuro-fuzzy inference system (ANFIS) for water quality components of Karoon River. They stated that all applied models have suitable performance for prediction of water quality components; however, the MLP model was slightly more accurate. managed the water quality of a water supply system. They considered this an optimization problem and used modern optimization methods to solve it. introduced a new approach for water allocation. They considered water quality as one of the main factors in their approach. developed a Probabilistic Support Vector Machines (PSVMs) model associated with GIS technique for planning the classification and distribution of surface and groundwater water in Iran. They stated that the use of these two methods would provide accurate information for feasibility studies of water conservation projects. utilized artificial neural networks for predicting the water quality components in several case studies. He stated that artificial intelligence techniques have suitable performance for modeling and predicting the internal relation between the water quality components and modeling their time series. Reviewing the literature shows that water quality assessment and prediction is an important factor for developing water conservation projects and, to this end, artificial intelligence techniques have been proposed. Hence, in this study the water quality components of Tireh River, one of the[2] main rivers of Dez catchment (one of the major catchments in Iran), were predicted using a support vector machine, article neural network and group method of data handling.

Machine learning (ML) encompasses a range of computational techniques that allow systems to learn from data and make predictions or decisions without being explicitly programmed. In the context of water quality, machine learning models are trained on historical data, including physical, chemical, and biological parameters, to predict future measurements or classify the status of water quality. Commonly used ML methods include:

Regression Models: For predicting quantitative parameters, such as pH levels, turbidity, and contaminant concentrations.

Classification Models: For categorizing water into quality classes, such as safe or unsafe based on standard thresholds.

Clustering Techniques: Used to identify patterns or groups in water quality data, which can help in identifying sources of pollution or changes in water quality over time.

Accurate predictions hinge on high-quality data, which can be derived from a variety of sources, including sensors in IoT (Internet of Things) frameworks, satellite imagery, and historical water quality records from public health departments. Feature engineering, the process of using domain knowledge to select, modify, or create new features from raw data, is a critical step that enhances model accuracy and interpretability.

Several challenges persist in the application of machine learning to water quality prediction. These include dealing with sparse or incomplete datasets, managing the variability in data due to geographical and temporal factors, and ensuring the robustness of models against the dynamic nature of environmental data. Future research may focus on integrating more diverse data sources, developing more sophisticated models that can handle large-scale temporal-spatial data, and enhancing models' interpretability for better stakeholder communication.

The use of machine learning in predicting water quality represents a significant advancement in environmental monitoring. By automating the prediction process, stakeholders can obtain timely insights into water quality trends, identify potential risks sooner, and implement more effective water management strategies. As technology progresses and more data becomes available, machine learning models will become increasingly vital tools in the quest to safeguard water resources and public health.

This introduction sets the stage for a deeper exploration of specific machine learning models, case studies demonstrating successful implementations, and detailed [3]discussions on optimizing model performance within the field of water quality prediction.

However, as best of our knowledge, this study provides the first comprehensive approach to evaluate the performance of WQI model(s) adopting new classification scheme for multi-class classification of coastal water quality. Moreover, the results of this study could be effective in obtaining the proper classification of water quality, which might be useful to improve the WQI model accuracy, transparency, and reliability in account of the correct classification of coastal water quality. The significant limitation of this research

was that it did not consider the temporal variability of water quality indicators in Cork Harbour. Further studies should be carried out to assess WQI model(s) performance using temporal resolution of indicators, with other predictive classifier algorithm(s) included.

1.2 EXISTING SYSTEM

The study's primary goal was to create a framework for evaluating the WQI model's performance in order to accurately classify water quality. After removing all the missing values in the data we have observed the accuracy of the algorithms have increased.

1.3 PROPOSED SYSTEM

Evaluation of WQI: Models Seven WQI models, both commonly used and recently proposed ones, were tested in the study. This comprehensive assessment aimed to identify the most suitable models for accurately assessing water quality along the coast.

Performance of Machine Learning Algorithms: The XGBoost algorithm and KNN (K-Nearest Neighbors) demonstrated superior performance in correctly classifying water quality. XGBoost particularly excelled, achieving accurate classification for most classifications of water quality, with the exception of "poor" quality.

Effectiveness of WQI Models: The weighted WQM-WQI and unweighted RMS-WQI models have been shown to be useful instruments for precisely determining the state of coastal water quality. Cork Harbour's water quality was successfully divided into "Good" and "Fair" classifications by these models.

Novel Contribution: The study is the first to provide a comprehensive approach to evaluating WQI model performance, implementing a novel multi-class classification scheme for coastal water quality. This new method improves comprehension and makes it easier to increase the correctness, transparency, and dependability of the WQI model.

Limitations and Future Directions: The study was severely limited by its failure to take into account the temporal variability of Cork Harbour's water quality measures.

In order to overcome this constraint, future research could evaluate the WQI model's effectiveness using indicators with temporal resolution and maybe include different predictive classifier methods.

Implications: Despite their limitations, the study's findings are helpful in lowering the possibility that incorrect classification may lead to model uncertainty. The insights provided can inform researchers, policymakers, and water resource personnel, facilitating more informed decision making regarding coastal water quality management and conservation efforts.

Its findings contribute to advancing understanding in this field and have practical implications for environmental management and policy development.

1.4 SYSTEM REQUIREMENTS

1.4.1 HARDWARE REQUIREMENTS

- Processor : Intel Core i5
- Cache Memory : 4MB
- Hard Disk : 30GB or more
- RAM : 1GB or more

1.4.2 SOFTWARE REQUIREMENTS

- Operating System : Windows 10
- Coding Language : Python
- Python Distribution : Anaconda, Flask
- Browser : Any Latest Browser Like Chrome

2. LITERATURE SURVEY

Literature Survey

ML techniques are used to predict the water state through performance analysis of the water quality index model. To find the optimal solution for the water quality we used the algorithms including XG Boost, KNN, SVM, Navie Bayes, Decision Tree, and Random Forest.

Both the environment and public health are directly impacted [4]by water quality. Water is utilized for several purposes, including industrial, agriculture, and drinking. The growth of water entertainment and sports in recent years has been a major factor in drawing tourists. Because rivers are the easiest to obtain among water sources, human cultures have relied on them more than other sources for their development.

Horrible diseases have resulted from the alarming rate at which water quality is deteriorating due to rapid urbanization and industrialization. The traditional method of estimating water quality involves costly and time consuming statistical and laboratory analysis.

Our suggested approach is dependent on variables such as temperature, pH, turbidity, solids, hardness, and so forth. Without a question, accurate forecasting will raise the bar for water resource management. Many water resource management agencies currently have monitoring stations set up to keep an eye on changes in the quality of the water.

2.1 MACHINE LEARNING

Machine learning is a subfield of artificial intelligence (AI) that focuses on the development of algorithms and statistical models that enable computers to perform specific tasks effectively without using explicit instructions. Instead, these systems learn from and make decisions based on patterns and inferences derived from data.

Machine learning is distinguished by its ability to[5] improve automatically through experience. This is fundamentally different from traditional software, which has fixed rules and does not improve with experience. As data grows exponentially, machine learning is becoming an essential aspect of leveraging this vast amount of information for insights and decision-making.

2.1.1 SOME MACHINE LEARNING METHODS:

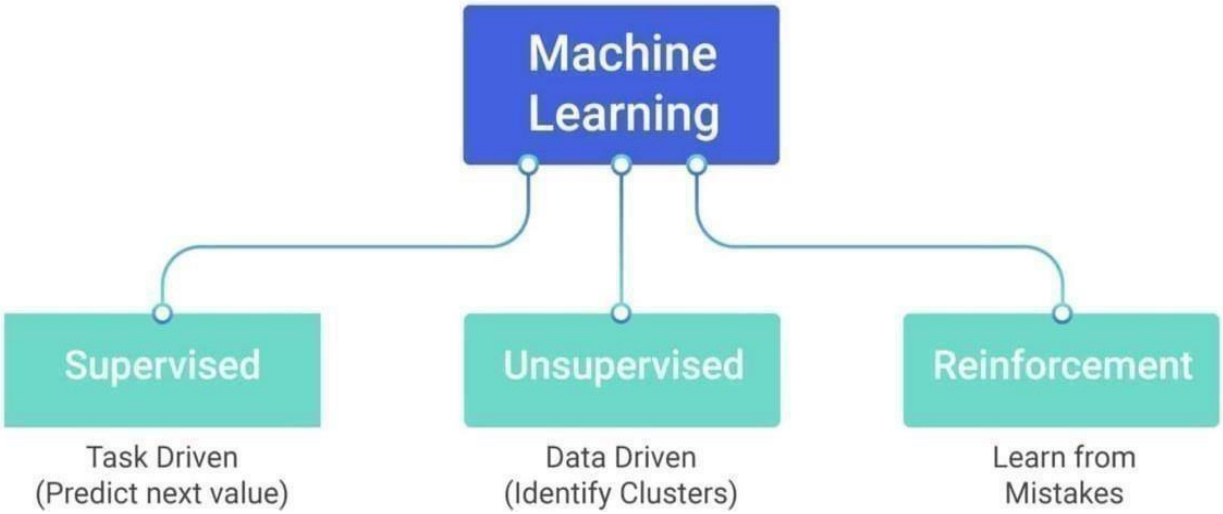


Fig 2.1.1 Types of Machine Leaning

2.1.2 Supervised machine learning algorithms:

Supervised machine learning algorithms can apply what has been learned in the past to new data using labeled examples to predict future events. Starting from the analysis of a known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. The system is able to provide targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors[6] in order to modify the model accordingly.

2.1.3 Unsupervised machine learning algorithms:

Unsupervised machine learning algorithms are used when the information used to train is neither classified nor labeled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabeled data. The system doesn't figure out the right output, but it explores the data and can draw inferences from datasets to describe hidden structures from unlabeled data. **2.1.4 Reinforcement machine learning algorithms:**

Reinforcement machine learning algorithms is a learning method that interacts with its environment by producing actions and discovers errors or rewards. Trial and error search and delayed reward are the most relevant characteristics of reinforcement learning. This method allows machines

and software agents to automatically determine the ideal behavior within a specific context in order to maximize its performance. Simple reward feedback is required for the agent to learn which action is best. This is known as the reinforcement signal.

2.1.5 Machine Learning Algorithms:

2.1.5.1 Linear Regression

To understand the working functionality of Linear Regression, imagine how you would arrange random logs of wood in increasing order of their weight. There is a catch; however [7]– you cannot weigh each log. You have to guess its weight just by looking at the height and girth of the log (visual analysis) and arranging them using a combination of these visible parameters. This is what linear regression in machine learning is like. In this process, a relationship is established between independent and dependent variables by fitting them to a line. This line is known as the regression line and is represented by a linear equation Y= a *X + b.

In this equation:

- Y – Dependent Variable
- a – Slope
- X – Independent variable
- b – Intercept

The coefficients a & b are derived by minimizing the sum of the squared difference of distance between data points and the regression line.

2.1.5.2 Logistic Regression

Logistic Regression is used to estimate discrete values (usually binary values like 0/1) from a set of independent variables. It helps predict the probability of an event by fitting data to a logit function. It is also called logit regression.[8]

These methods listed below are often used to help improve logistic regression models:

- include interaction terms
- eliminate features
- regularize techniques
- use a non-linear model

2.1.5.3 Decision Tree

Decision Tree algorithm in machine learning is one of the most popular algorithm in use today; this is a supervised learning algorithm that is used for classifying problems. It works well in classifying both categorical and continuous dependent variables. This algorithm divides the population into two or more homogeneous sets based on the most significant attributes/ independent variables.

2.1.5.4 SVM Algorithm

SVM algorithm is a method of a classification algorithm in which you plot raw data as points in an ndimensional space (where n is the number of features you have). [9]The value of each feature is then tied to a particular coordinate, making it easy to classify the data. Lines called classifiers can be used to split the data and plot them on a graph.

2.1.5.5 Naive Bayes Algorithm

A Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Even if these features are related to each other, a Naive Bayes classifier would consider all of these properties independently when calculating the probability of a particular outcome. [10]A Naive Bayesian model is easy to build and useful for massive datasets. It's simple and is known to outperform even highly sophisticated classification methods.

2.1.5.6 KNN Algorithm

This algorithm can be applied to both classification and regression problems. Apparently, within the Data

Science industry, it's more widely used to solve classification problems. It's a simple algorithm that stores all available cases and classifies any new cases by taking a majority vote of its k neighbors. The case is then assigned to the class with which it has the most in common. A distance function performs this measurement.

KNN can be easily understood by comparing it to real life. For example, if you want information about a person, it makes sense to talk to his or her friends and colleagues!

Things to consider before selecting K Nearest Neighbours Algorithm:

- KNN is computationally expensive
- Variables should be normalized, or else higher range variables can bias the algorithm Data still needs to be pre-processed.

2.1.5.7 Random Forest Algorithm

- A collective of decision trees is called a Random Forest. To classify a new object based on its attributes, each tree is classified, and the tree "votes" for that class. The forest chooses the classification having the most votes (over all the trees in the forest).
- Each tree is planted & grown as follows:
- If the number of cases in the training set is N, then a sample of N cases is taken at random. This sample will be the training set for growing the tree.
- If there are M input variables, a number m<<M is specified such that at each node, m variables are selected at random out of the M, and the best split on this m is used to split the node. The value of m is held constant during this process.
- Each tree is grown to the most substantial extent possible. There is no pruning.

2.1.6 APPLICATIONS OF MACHINE LEARNING:

1. Virtual Personal Assistants
2. Predictions while Commuting
3. Videos Surveillance
4. Social Media Services
5. Email Spam and Malware Filtering
6. Online Customer Support
7. Search Engine Result Refining
8. Product Recommendations
9. Online Fraud Detection.

2.1.7 Common Examples of Machine Learning (ML)

- Facial recognition. ...
- Product recommendations. ...
- Email automation and spam filtering. ...
- Financial accuracy. ...
- Social media optimization. ...
- Healthcare advancement. ...
- Mobile voice to text and predictive text. ...
- Predictive analytics.

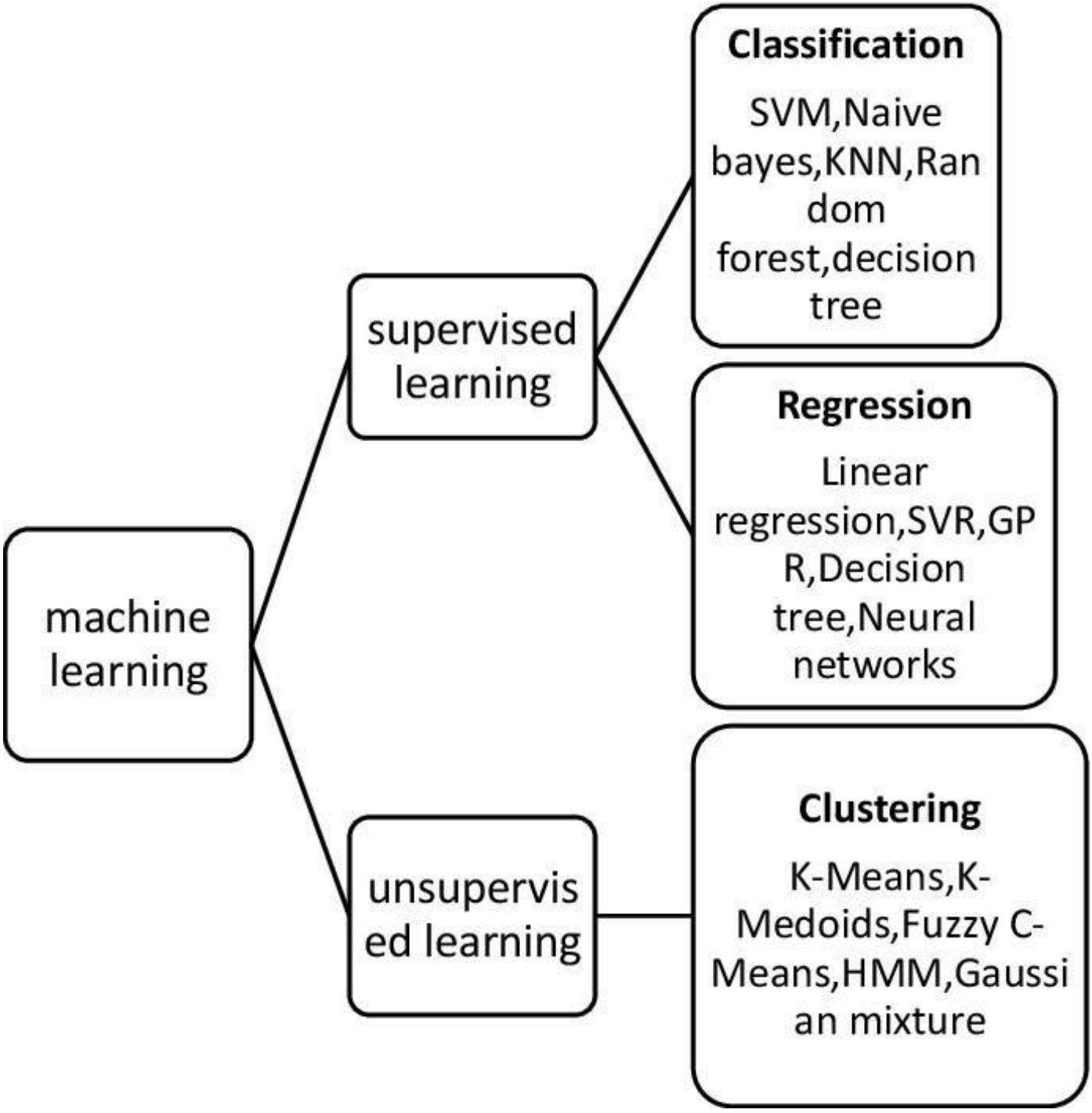


Fig 2.1.7: Machine learning Techniques

3. SYSTEM ANALYSIS

3.1 Importance of machine learning Using python:

The importance of machine learning in wine quality is increasing because of its ability to process huge datasets efficiently beyond the range of human capability, and then dependably convert analysis of that data into clinical insights that assist in planning and providing care, which ultimately leads to better outcomes, reduces the consumption. Using these types of advanced analytics, we can provide better information at the point of consumption.

3.2 Implementation of machine learning using Python:

Python is a popular programming language. It was created in 1991 by Guido van Rossum.It is used for.

3.2.1 Web development (server-side),

3.2.2 Software development,

3.2.3 Mathematics.

The most recent major version of Python is Python 3. However, Python 2, although not being updated with anything other than security updates, is still quite popular. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse, Anaconda which are particularly useful when managing larger collections of Python files

Python was designed for its readability. Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses. Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose. In the older days, people used to perform Machine Learning tasks manually by coding all the algorithms and mathematical and statistical formula.

This made the process time consuming, tedious and inefficient. But in the modern days, it is become very much easy and efficient compared to the olden days by various python libraries,for frameworks, and modules. Today, Python is one of the most popular programming languages for or this task and it has replaced many languages in the industry, one of the reasons is its vast co llection of libraries.

Python libraries that used in Machine Learning are:

3.2.1 Numpy

3.2.2 imbalanced-learn

3.2.3 Scikit-learn

3.2.4 pandas

3.2.5 Matplotlib

3.2.1 NumPy

It is a very popular python library for large multi-dimensional array and matrix processing, with the help of a large collection of high-level mathematical functions. It is very useful for fundamental scientific computations in Machine Learning. It is particularly useful for linear algebra, Fourier transform, and random number capabilities. High-end libraries like TensorFlow uses NumPy internally for manipulation of Tensors.

3.2.2 Imbalanced-learn

Imbalanced-learn is a powerful Python library designed for addressing class imbalance in machine learning datasets. Class imbalance occurs when one class (the minority class) is significantly underrepresented compared to the other classes (the majority class/es). This imbalance can lead to biased models that perform poorly in predicting the minority class. Imbalanced-learn provides various techniques such as over-sampling, under-sampling

3.2.3 Scikit-learn

Scikit-learn is one of the most popular Machine Learning libraries for classical Machine Learning algorithms. It is built on top of two basic Python libraries, NumPy and SciPy. Scikit- learn supports most of the supervised and unsupervised learning algorithms. Scikit learn can also be used for datamining and dataanalysis, which makes it a great tool who is starting out with MachineLearning.

3.2.4 Pandas

Pandas is a popular Python library for data analysis. It is not directly related to Machine Learning. As we know that the dataset must be prepared before training. In this case, Pandas comes handy as it was developed specifically for data extraction and preparation. It provides high-level data structures and wide variety tools for data analysis. It provides many inbuilt methods for groping, combining and filtering data.

3.2.5 Matplotlib

Matplotlib is a very popular Python library for data visualization. Like Pandas, it is not directly related to Machine Learning. It particularly comes in handy when a programmer wants to visualize the patterns in the data. It is a 2D plotting library used for creating 2D graphs and plots. A module named pyplot makes it easy for programmers for plotting as it provides features to control line styles, font properties, formatting axes, etc. It provides various kinds of graphs and plots for data 13 visualization, histogram, error charts, bar chats, etc.

3.3 Scope of the project

The scope of a project focused on water quality prediction using machine learning typically encompasses several critical components designed to leverage computational models for monitoring, predicting, and managing water quality across various water bodies. This kind of project is often interdisciplinary, involving expertise from environmental science, data science, computer science, and often public health. Below is a detailed outline of the typical scope for such a project.

3.4 Data Set Analysis

We are collecting a dataset from kaggle website to make predictions and we have collected dataset that is water_quality.csv [1] The dataset used for water quality prediction contains more than 2000 rows, and the following columns are included:

3.4.1 Ph Value: Ph is An Important Parameter In Evaluating The Acid–base Balance Of Water. Who Has Recommended A Maximum Permissible Limit Of Ph From 6.5 To 8.5. The Current Investigation Ranges Were 6.52–6.83

3.4.2 Hardness: Hardness Was Originally Defined as Capacity Of water to Precipitate Soap Caused by Calcium and Magnesium

3.4.3 Solids: Water Has the Ability to Dissolve a Wide Range of Inorganic and Some Organic Minerals or Salts Such as Potassium, Calcium, Sodium, Bicarbonates, Chlorides, Magnesium, Sulfates Etc. These Minerals Produced an Unwanted Taste and Diluted Color in the Appearance of Water.

3.4.4 Chloramines: Chlorine Levels Up To 4 Milligrams Per Liter (Mg/L Or 4 Parts Per Million (Ppm)) Are Considered Safe in Drinking Water

3.4.5 Sulfate: Sulfates Are Naturally Occurring Substances That Are Found in Minerals, Soil, And Rocks. They Are Present in Ambient Air, Groundwater, Plants, And Food. Sulfate Concentration in Seawater Is About 2,700 Milligrams Per Liter (Mg/L). It Ranges From 3 To 30 Mg/L In Most Freshwater Supplies

3.4.6 Conductivity: According To Who Standard Value Should Not Exceed 400 MS/ Cm

3.4.7 Organic carbon: TOC Is a Measure of the Total Amount of Carbon in Organic Compounds in Pure Water. According To the US EPA < 2 Mg/L As TOC In Treated / Drinking Water, and < 4 Mg/Lit in Source Water Which Is Used for Treatment

3.4.8 Trihalomethanes: THM Levels Up To 80 Ppm Is Considered Safe in Drinking Water.

To Drink And (1) Water Is Safe to Drink

3.4.9 Turbidity: The Mean Turbidity Value Obtained (0.98 Ntu) Is Lower Than the Who Recommended Value of 5.00

3.4.10 Potability: Indicates If Water Is Safe for Human Consumption Where 1 Means Potable And 0 Means Not Potable. (0) Water Is Not Safe.

3.5 Dataset Characteristics:

Dataset characterization is a critical step in any machine learning project, including those focused on water quality prediction. The process involves detailed analysis of the data types, structures, and quality to ensure that the dataset is suitable for building robust machine learning models.

3.5.1 Handling Missing Data

- **Imputation:** Replace missing values using statistical methods (mean, median, mode) or more complex algorithms like k-nearest neighbors, which can predict missing values based on similarity scores with other data points.

3.5.2 Data Cleaning

- **Outliers:** Identify and address outliers,which can be due to measurement errors or genuine rare events.Techniques like IQR (interquartile Range) or Z-scores can help in detecting outliers.
- **Noise Reduction:** Apply smoothing techniques or filters to reduce variability in the data that might obscure patterns.

3.5.3 Data Partitioning:

- **Splitting data:** Divide the data into training, validation, and test sets to ensure that the model can be trained, validated, and tested on different subsets of data. This helps in evaluating the model's performance effectively.

```
data.describe().T
```

	count	mean	std	min	25%	50%	75%	max
ph	2011.0	7.085990	1.573337	0.227499	6.089723	7.027297	8.052969	14.000000
Hardness	2011.0	195.968072	32.635085	73.492234	176.744938	197.191839	216.441070	317.338124
Solids	2011.0	21917.441374	8642.239815	320.942611	15615.665390	20933.512750	27182.587067	56488.672413
Chloramines	2011.0	7.134338	1.584820	1.390871	6.138895	7.143907	8.109726	13.127000
Sulfate	2011.0	333.224672	41.205172	129.000000	307.632511	332.232177	359.330555	481.030642
Conductivity	2011.0	426.526409	80.712572	201.619737	366.680307	423.455906	482.373169	753.342620
Organic_carbon	2011.0	14.357709	3.324959	2.200000	12.124105	14.322019	16.683049	27.006707
Trihalomethanes	2011.0	66.400859	16.077109	8.577013	55.952664	66.542198	77.291925	124.000000
Turbidity	2011.0	3.969729	0.780346	1.450000	3.442915	3.968177	4.514175	6.494749
Potability	2011.0	0.403282	0.490678	0.000000	0.000000	0.000000	1.000000	1.000000

Fig 2.13.3: Dataset

3.6 METHODOLOGY

The wqi score for coastal water quality was stored using recently developed, improved wqi methodologies and the water quality classes were established by utilizing the coastal water quality categorization scheme. overview of this water quality prediction has been done through loading the data preprocessing into the model inputs and after the training and testing set algorithms. roc curve analysis is done. develop unique classification scheme based on the best cut-point of roc.With this method, you will

be able to create a predictive model that, depending on a number of factors, can precisely anticipate the quality of the water. data collection, preprocessing, feature and model selection, training, testing, and deployment are all included in the methodology.

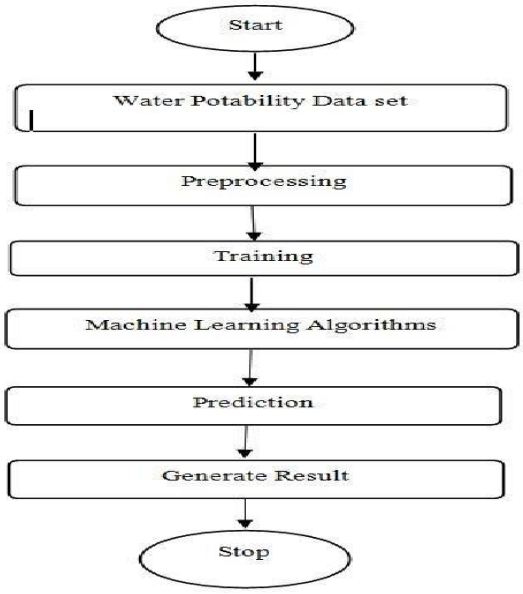


Fig 3.6.1: Machine Learning Work Flow

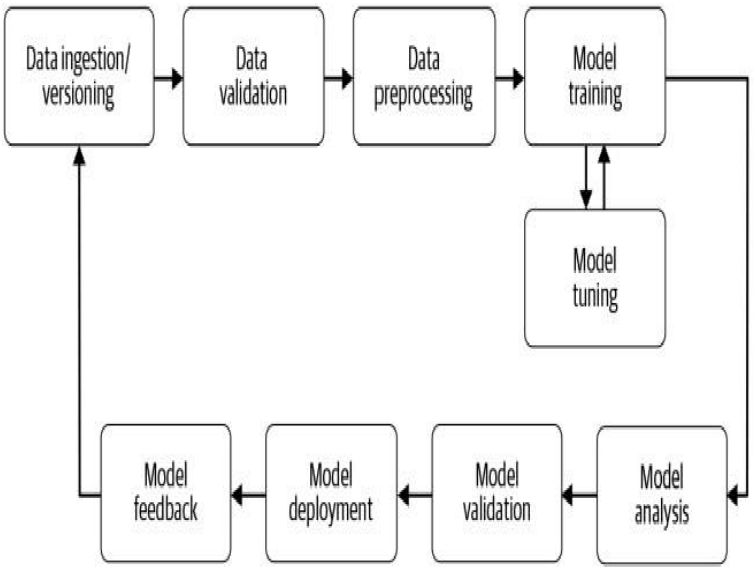


Fig 3.6.2: Machine learning pipiline

3.7 Data Pre-Processing

3.7.1 Feature selection

Feature selection is a critical process in the development of machine learning models for water quality prediction. This process involves identifying the most relevant variables that contribute to predicting water quality effectively. Effective feature selection can improve model performance, reduce overfitting, and decrease computation time.

3.7.2 Data Cleaning

Data cleaning is a crucial preliminary step in any machine learning project, including water quality prediction. It ensures that the dataset used for modeling is accurate, consistent, and reliable. Start by identifying missing values in your dataset. This can be done using methods like isnull**()** in Python's pandas library. Check for consistency in data, especially if it's collected from multiple sources or over long time periods. This includes validating that similar conditions are reported similarly.

```
data.isnull().sum()
```

```
ph                491
Hardness            0
Solids              0
Chloramines         0
Sulfate           781
Conductivity        0
Organic_carbon      0
Trihalomethanes   162
Turbidity           0
Potability          0
dtype: int64
```

Fig 2.15.2: Dataset After Removing Null Values

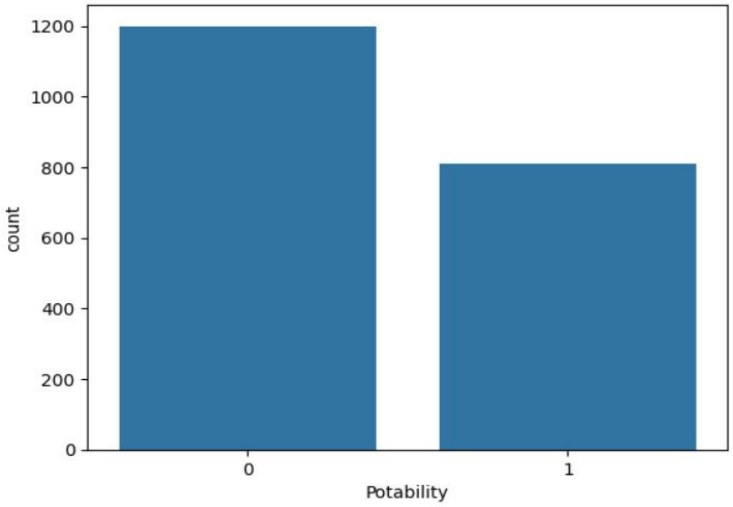


Fig 3.7.2: count plot

The count plot shows the occurrence of safe and unsafe water. The safe water appeared opver 80% in the data set compared to unsafe water as shown above. This shows that our variables are balanced.

3.8 Cross Validation:

Cross-validation is a technique in which we train our model using the subset of the data- set and then evaluate using the complementary subset of the data-set. The three steps involved in cross validations are follows

- Reserve some portion of sample data-set.
- Using the rest data-set train the model.

- Test the model using the reserve portion of the data-set

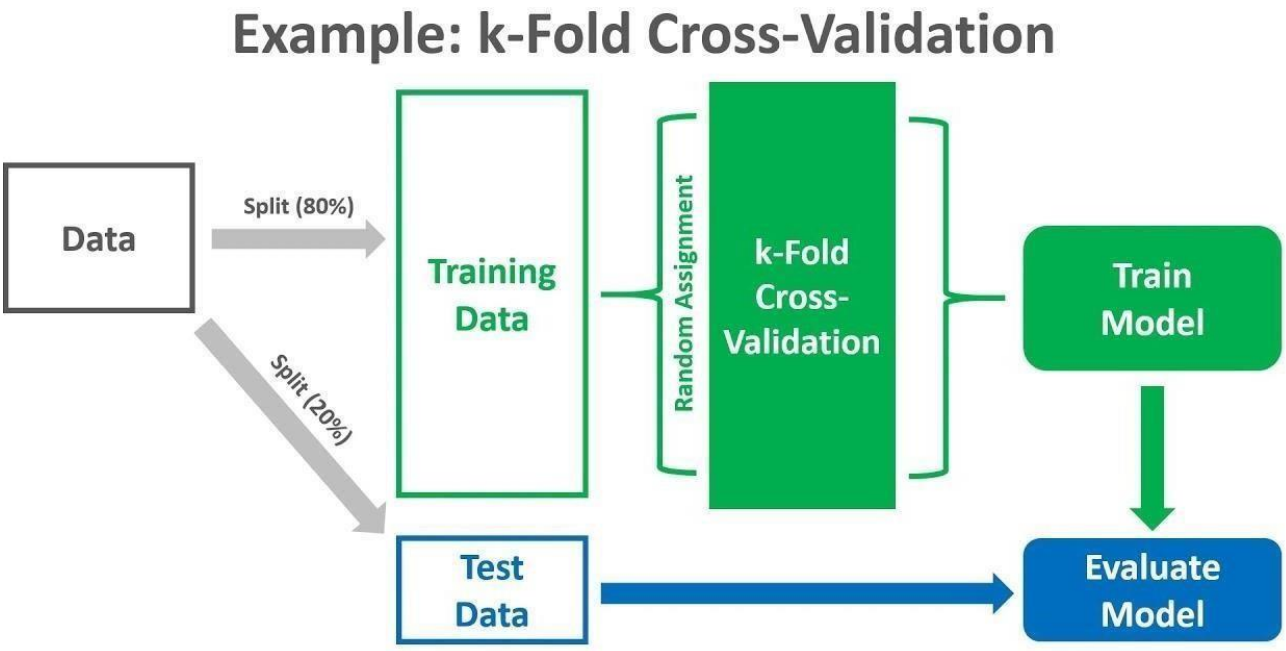


Fig 3.8.1: Cross validation

3.9 Classification:

It is a process of categorizing data into given classes. Its primary goal is to identify theclass of our new data.

3.9.1 Machine learning algorithms for classification:

Research on data mining has led to the formulation of several data mining algorithms. These algorithms can be directly used on a dataset for creating some models orto draw vital conclusions and inferences from that dataset. Some popular data mining algorithms are Random Forest, Naïve Bayes, KNN etc

3.9.2 Random Forest:

Random Forest is a powerful ensemble learning algorithm widely used in machine learning and data science tasks, including classification and regression. It operates by constructing multiple decision trees during training and outputs the mode (classification) or average prediction (regression) of the individual trees.

One of the key advantages of Random Forest is its ability to handle high-dimensional data with numerous features while mitigating overfitting, a common issue in complex models. This is achieved through a technique called bagging, where each decision tree is trained on a random subset of the training data and features, ensuring diversity and robustness in the ensemble.

The Random Forest algorithm offers several benefits, including robustness to noise and outliers in the data, resilience to overfitting due to ensemble averaging, and the ability to handle both categorical and numerical features without extensive preprocessing. Moreover, Random Forest provides feature importance scores, allowing analysts and data scientists to interpret the relevance and contribution of each feature in the model's predictions. This information can guide feature selection and optimization efforts, leading to more efficient and accurate models.

Another advantage of Random Forest is its versatility and applicability to various machine learning tasks. Whether for classification tasks such as spam detection, disease diagnosis, or customer segmentation, or regression tasks like predicting sales figures or housing prices, Random Forest can deliver competitive performance. It can also handle imbalanced datasets and missing values effectively, making it a reliable choice for real-world applications where data quality and diversity are common challenge.

3.9.3 Navie Bayes:

Naive Bayes is a probabilistic classifier based on Bayes' theorem, which assumes independence among features given the class label. Despite its simplicity and the "naive" assumption, Naive Bayes often performs well in practice, especially for text classification tasks like email spam detection or sentiment analysis. Its computational efficiency, scalability, and robustness to irrelevant features make it a popular choice for handling large datasets and high-dimensional feature spaces.

3.9.4 Random Forest Classifier:

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a 22 complex problem and to improve the performance of the model. Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the prediction accuracy of the dataset.

4. IMPLEMENTATION CODE

```
import pandas as pd import

seaborn as sns import

numpy as np

import matplotlib. pyplot as plt data =

pd.read_csv("water_potability.csv") data

%pip install pandas numpy matplotlib seaborn

data.isnull().sum() data = data.dropna()

data.isnull().sum() data.head() data.info()

data.describe().T data.skew(axis=0,

skipna=True)

sns.pairplot(data = data, hue = 'Potability')

plt.show() data.corr()

plt.figure(figsize=(20,10))

sns.heatmap(data.corr(), cmap='YlGnBu', annot=True)

plt.show() sns.countplot(x = 'Potability', data = data)

plt.title("Distribution of Unsafe and Safe Water")

plt.show()

%pip install plotly nbformat

%pip install plotly --upgrade nbformat import plotly.express as px data = data figure =

px.histogram(data, x = "ph", color = "Potability", title= "Factors Affecting Water Quality: PH")

figure = px.histogram(data, x = "Hardness", color = "Potability", title= "Factors Affecting Water Quality: Hardness") figure = px.histogram(data, x = "Solids", color = "Potability", title= "Factors Affecting Water Quality: Solids")

figure = px.histogram(data, x = "Chloramines", color = "Potability", title= "Factors Affecting Water Quality: Chloramines")
```

figure = px.histogram(data, x = "Sulfate", color = "Potability", title= "Factors Affecting Water Quality: Sulfate")

figure = px.histogram(data, x = "Conductivity", color = "Potability", title= "Factors Affecting Water Quality: Conductivity")

figure = px.histogram(data, x = "Organic_carbon", color = "Potability", title= "Factors Affecting Water Quality: Organic Carbon")

figure = px.histogram(data, x = "Trihalomethanes", color = "Potability", title= "Factors Affecting Water

Quality: Trihalomethanes") figure = px.histogram(data, x = "Turbidity", color = "Potability", title=

"Factors Affecting Water Quality: Turbidity") sns.pairplot(data=data) data.boxplot(figsize=(15,6))

plt.show() data.skew(axis=0, skipna=True)

x = data.iloc[:,:-1] #leaving last column y = data.iloc[:,-1] x y cols = 'Potability'

print((data[cols] == 3).all()) print(data['Potability'].value_counts())

print(data['Potability'].unique()) features = data.iloc[:, 1:].values labels = data['Potability'].values

%pip install scikit-learn from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.20, shuffle=True, random_state=0)

a=x_train

a

b=y_train

b x_test

y_test

%pip install xgboost import xgboost as xgb from xgboost import

XGBClassifier, XGBRegressor from xgboost import plot_importance from

sklearn.neighbors import KNeighborsClassifier, KNeighborsRegressor from

sklearn.svm import SVC svc = SVC(kernel = 'linear', random_state=0

,probability=True) svc.fit(a,b)

from sklearn.naive_bayes import GaussianNB nb

= GaussianNB()

```
nb.fit(x_train, y_train) from sklearn.tree import DecisionTreeClassifier
dectree = DecisionTreeClassifier(criterion = 'entropy', random_state=0)
dectree.fit(x_train, y_train)

from sklearn.ensemble import RandomForestClassifier ranfor =
RandomForestClassifier(n_estimators = 11, criterion = 'entropy',
random_state=0) ranfor.fit(x_train, y_train) y_pred_svc =
svc.predict(x_test) y_pred_nb = nb.predict(x_test) y_pred_dectree =
dectree.predict(x_test) y_pred_ranfor = ranfor.predict(x_test) from
sklearn.metrics import accuracy_score

from sklearn.metrics import
precision_score,recall_score,f1_score,roc_curve,roc_auc_score,classification_report model_scores
= {
'XGB': accuracy_score(y_test, y_pred_svc)*100,
'KNN': accuracy_score(y_test, y_pred_nb)*100,
'SVM': accuracy_score(y_test, y_pred_dectree)*100,
'NB': accuracy_score(y_test, y_pred_ranfor)*100,
'DT': accuracy_score(y_test, y_pred_ranfor)*100, 'RF':
accuracy_score(y_test, y_pred_ranfor)*100}
model_compare=pd.DataFrame(model_scores,index=['Accuracy']) model_compare
def create_report(model, X_test, y_test):
   y_pred = model.predict(X_test)     report =
classification_report(y_test, y_pred)     acc =
accuracy_score(y_test, y_pred)
print(f"Accuracy : {acc*100:.4f} %\n")
print("Classification report: \n")
print(report)
```

```
from sklearn.metrics import classification_report print('XGB:',
classification_report(y_test, y_pred_dectree),'\n'
    'KNN:', classification_report(y_test, y_pred_svc),'\n'
    'SVM:', classification_report(y_test, y_pred_nb),'\n',
    'NB:', classification_report(y_test, y_pred_dectree),'\n',
    'DT:', classification_report(y_test, y_pred_ranfor),'\n'      'RF:',
classification_report(y_test, y_pred_ranfor)) from sklearn.model_selection import
StratifiedKFold, cross_val_score from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import precision_score, recall_score, f1_score import numpy as
np dt_model = DecisionTreeClassifier() k = 20 kf = StratifiedKFold(n_splits=k) from
sklearn.model_selection import cross_val_score, KFold from sklearn.ensemble
import RandomForestClassifier # Replace with your classifier classifier =
RandomForestClassifier()
kf = KFold(n_splits=5, shuffle=True, random_state=42) accuracy_scores =
cross_val_score(classifier, features, labels, scoring='accuracy', cv=kf) precision_scores =
cross_val_score(classifier, features, labels, scoring='precision', cv=kf) recall_scores =
cross_val_score(classifier, features, labels, scoring='recall', cv=kf) f1_scores =
cross_val_score(classifier, features, labels, scoring='f1', cv=kf) print("Cross-Validation
Accuracy Scores:", accuracy_scores) print("Cross-Validation Precision Scores:",
precision_scores) print("Cross-Validation Recall Scores:", recall_scores) print("Cross-
Validation F1 Scores:", f1_scores) print("Mean Accuracy:", accuracy_scores.mean())
print("Accuracy:", np.mean(accuracy_scores) * 100)
print("Precision:", np.nanmean(precision_scores) * 100)
print("Recall:", np.nanmean(recall_scores) * 100) print("F1
Score:", np.nanmean(f1_scores) * 100)
XGB_acc=np.mean(accuracy_scores) * 100
KNN_acc=np.mean(precision_scores) * 100
```

```
SVM_acc=np.mean(recall_scores) * 100 NB_acc=np.mean(f1_scores)
* 100
from sklearn.model_selection import cross_val_score, KFold num_folds = 10 kf =
KFold(n_splits=num_folds, shuffle=True,random_state=42) cv_scores =
cross_val_score(dt_model,data,labels, cv=kf, scoring='accuracy')
print(f"\n{num_folds}-Fold Cross-Validation Scores:") print(cv_scores)
average_cv_accuracy = np.mean(cv_scores) print(f"\nAverage Cross-Validation
Accuracy: {average_cv_accuracy*100:.2f}%") np.random.seed(42)
noisy_features = features + np.random.normal(0, 0.5, size=features.shape) classifier =
RandomForestClassifier() kf = KFold(n_splits=5, shuffle=True, random_state=42)
accuracy_scores_noisy = cross_val_score(classifier, noisy_features, labels, scoring='accuracy', cv=kf)
print("Cross-Validation Accuracy Scores with Noisy Features:", accuracy_scores_noisy) print("Mean
Accuracy with Noisy Features:", accuracy_scores_noisy.mean()) data = {
'Model':['XGB','RF','KNN','SVM','NB','DT'],
    'Accuracy': [98.0,93.1,89.8,90.1,81.1,87.93],
    'Precision': [94.9, 92.7,91.3,95.4,80.3,88.3],
    'F1_Score': [92.9, 91.8, 92.4, 88.8,81.21,86.2],
    'Recall': [93,96.5,92.8,92.5,93.0,84.23]
}
dtfm= pd.DataFrame(data)
print(dtfm) plt.figure(figsize=(12, 8)) plt.subplot(2, 2, 1)
sns.barplot(x=dtfm['Model'], y=dtfm['Accuracy'], color='blue')
plt.title('Model Accuracy') plt.subplot(2, 2, 2)
sns.barplot(x=dtfm['Model'], y=dtfm['Precision'], color='lightgreen')
plt.title('Model Precision') plt.subplot(2, 2, 3)
sns.barplot(x=dtfm['Model'], y=dtfm['Recall'], color='purple')
plt.title('Model Recall') plt.subplot(2, 2, 4)
```

```
sns.barplot(x=dtfm['Model'], y=dtfm['F1_Score'], color='skyblue')

plt.title('Model F1 Score') plt.tight_layout() plt.show() importpandas

as pd import matplotlib.pyplot as plt import numpy as np data_before

= {

   'Model': ['XGBOOST', 'RANDOM FOREST','K-NEAREST NEIGHBOUR','SUPPORT VECTOR MACHINE','NAIVE BAYES','DECISION TREE'],

   'Accuracy': [98.0,93.1,89.8,90.1,81.1,87.93],

   'Precision': [94.9, 92.7,91.3,95.4,80.3,88.3],

   'F1_Score': [92.9, 91.8, 92.4, 88.8,81.21,86.2],

   'Recall': [93,96.5,92.8,92.5,93.0,84.23]

} data_after =

{

   'Model': ['XGBOOST', 'RANDOM FOREST','K-NEAREST NEIGHBOUR','SUPPORT VECTOR MACHINE','NAIVE BAYES','DECISION TREE',],

   'Accuracy': XGB_acc,

   'Precision': KNN_acc,

   'F1_Score': SVM_acc,

   'Recall': NB_acc

}

dtfm_before = pd.DataFrame(data_before)

dtfm_after = pd.DataFrame(data_after)

dtfm_before.set_index('Model', inplace=True)

dtfm_after.set_index('Model', inplace=True)

fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(30, 20)) def plot_bargraph(ax, metric,

title): before_values = dtfm_before[metric] after_values = dtfm_after[metric] models

= dtfm_before.index index = np.arange(len(models)) bar_width = 0.35 ax.bar(index,

before_values, width=bar_width, label='Before', color='skyblue') ax.bar(index +
```

```
bar_width, after_values, width=bar_width, label='After', color='lightcoral') for i in
range(len(models)):
    ax.text(index[i] - 0.05, before_values[i] + 1, f'{before_values[i]:.1f}', fontsize=20) ax.text(index[i] +
bar_width - 0.05, after_values[i] + 1, f'{after_values[i]:.1f}', fontsize=20) ax.set_xlabel('Model')
ax.set_ylabel(metric)
    ax.set_title(title)
    ax.set_xticks(index + bar_width / 2)
ax.set_xticklabels(models)
ax.legend()
plot_bargraph(axes[0, 0], 'Accuracy', 'Accuracy Comparison')
plot_bargraph(axes[0, 1], 'Precision', 'Precision Comparison')
plot_bargraph(axes[1, 0], 'F1_Score', 'F1 Score Comparison')
plot_bargraph(axes[1, 1], 'Recall', 'Recall Comparison')
plt.tight_layout() plt.show()
model_compare.T.plot(kind='bar') data
%pip install pickle import pickle
with open("water.pkl","wb") as file:
pickle.dump(dectree,file) from flask import
Flask, render_template, request import pickle
import numpy as np app = Flask(_name_)
model = pickle.load(open("water.pkl", "rb"))
@app.route('/') def
result():
  return render_template("front.html")
@app.route('/prediction', methods=['POST', 'GET']) def prediction():if
request.method == 'POST': features = [float(x) for x in request.form.values()] #
Convert to float instead of int final = [np.array(features)] prediction =
```

```
model.predict_proba(final)                output = '{0:.{1}f}'.format(prediction[0][1], 2)

output_float = float(output)

 if output_float > 0.5:

          return render_template("front.html", pred="Water is safe to drink. Potability:
{}".format(output_float))

       else:

          return render_template("front.html", pred="Water is unsafe. Do not drink. Potability:
{}".format(output_float))

    else:

       return render_template("front.html", pred="Invalid request method.") if

__name__== '_main_':

    app.run(debug=True,host="0.0.0.0",port=7000)

# Use an official Python runtime as a parent image

FROM python:3.12-slim

# Set the working directory in the container

WORKDIR /app

# Copy the current directory contents into the container at /app

COPY . /app

# Install any needed packages specified in requirements.txt

RUN pip install --no-cache-dir -r requirements.txt

# Make port 80 available to the world outside this container

EXPOSE 7000

# Define environment variable

ENV NAME World

# Run app.py when the container launches

CMD ["python", "app.py",7000]
```

Template:

Front.html:

```
<DOCTYPE html>
<html>
<head>
   <title>water quality prediction</title>
   <meta charset="UTF-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
        <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/fontawesome.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
        <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
        <style>
body{
            background-image:url("https://www.astri.org/wp-content/uploads/2023/05/img-08.png");
        }

        </style>
   </head>
   <body>
   <div class="background-image">
        <h1 style="text-align:center; font-size:60px; color: rgb(43, 92, 226); text-decoration:underline;  font-family: 'Josefin Sans', sans-serif;">Water Quality Prediction</h1>
   <form action="/prediction" class="loginbox" method="post"><br><br><br><br><br><br><br><br>
```

```
        <input type="text" name="year" placeholder="Enter pH value" required="required" />        

        <input type="text" name="do" placeholder="Enter Hardness" required="required" /><br><br>

        <input type="text" name="ph" placeholder="Enter Solids" required="required" />       

        <input type="text" name="co" placeholder="Enter Chloramines" required="required" /><br><br>

        <input type="text" name="bod" placeholder="Enter Sulfate" required="required" />       

        <input type="text" name="na" placeholder="Enter Conductivity" required="required" /><br><br>

        <input type="text" name="bod1" placeholder="Enter Organic_carbon" required="required" />       

        <input type="text" name="na1" placeholder="Enter Trihalomethanes" required="required" /><br><br>

        <input type="text" name="tc" placeholder="Enter Turbidity" required="required" /><br><br>

        <button type="submit" class="btn btn-light btn-outline-success" id="sendMessageButton">Predict</button>

    </center>

    </form>

    <h1 style="text-align:center">

        {{pred}}

    </h1>

    <h1 style="color: rgb(51, 0, 255); font-family: 'Josefin Sans', sans-serif;"></h1>

        </div>

          </body>  </html>
```

5. Output Screens

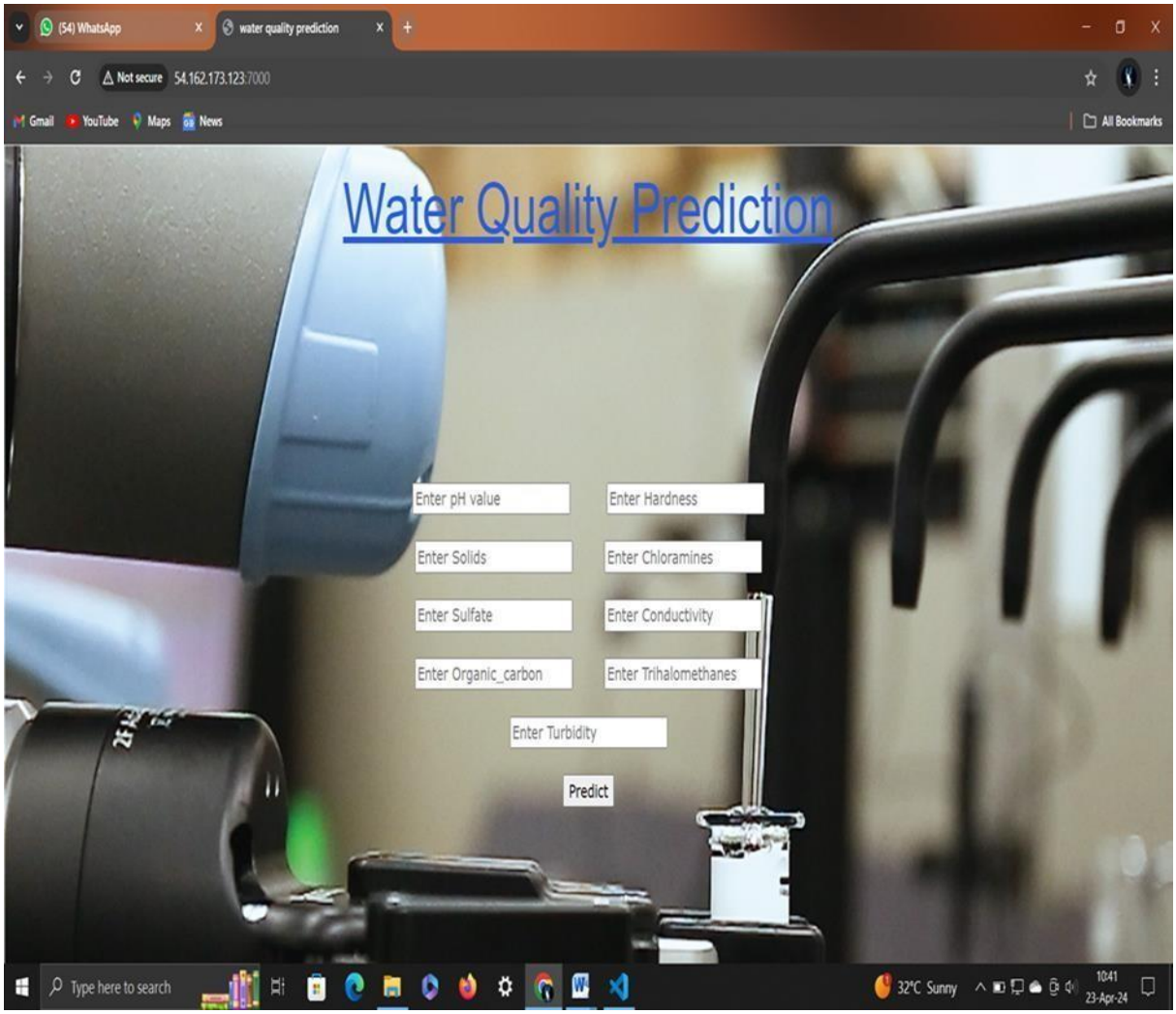


Fig. 5.1: Output interface.

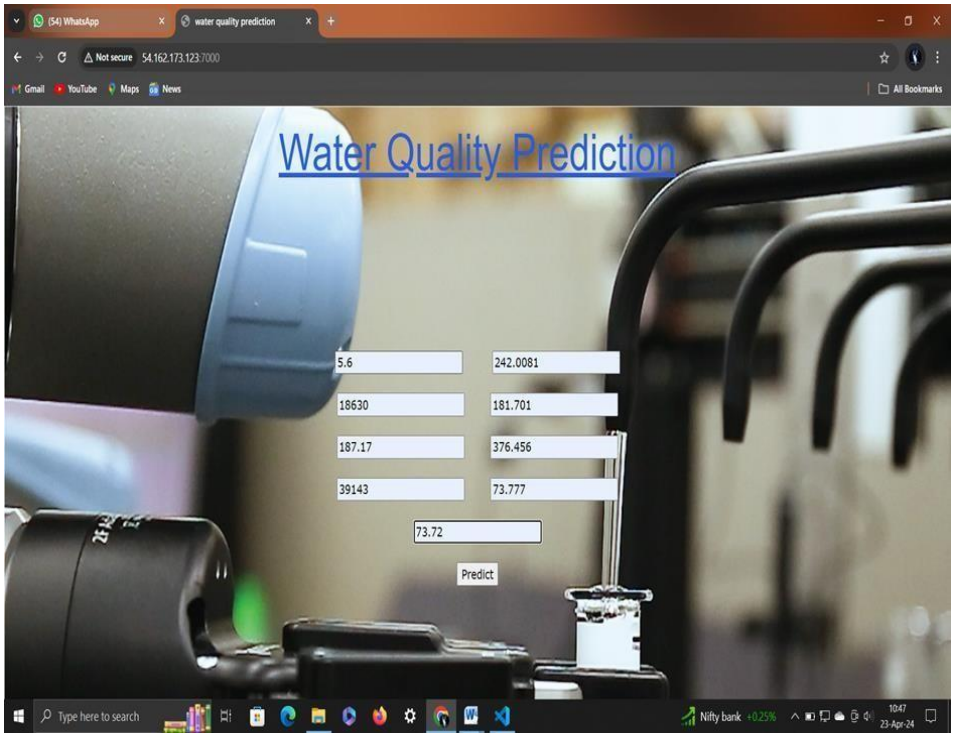


Fig. 5.2: Input values are entered.

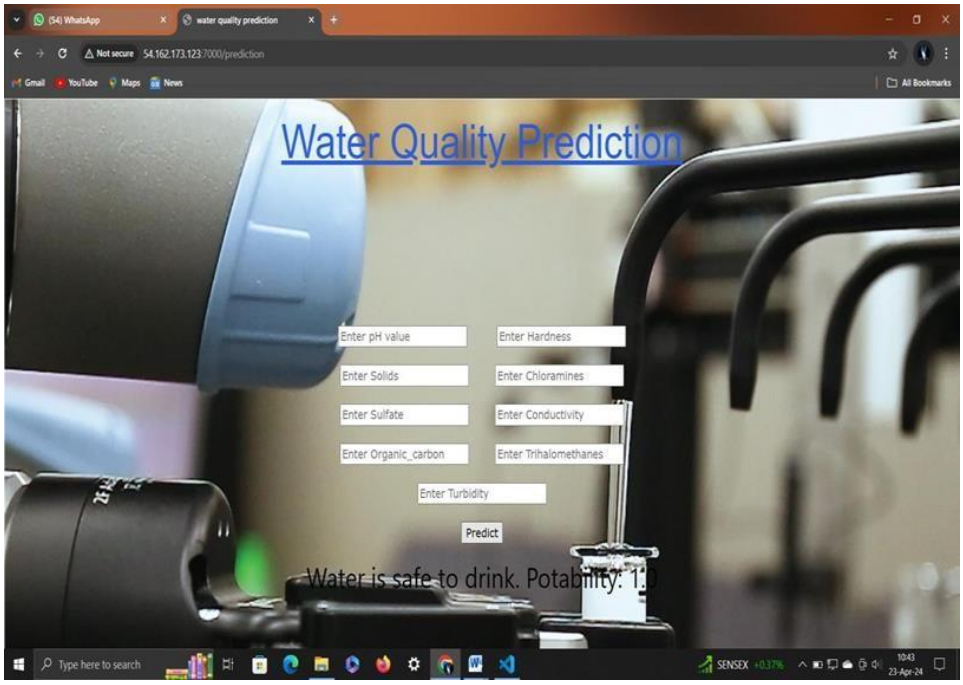


Fig. 5.3: Output Interface.

6. Result Analysis

This study shows the statistical summary of different WQI scores in River water through the study period. It has more information on the outcomes of different WQI models and their implications for the status of the water quality. Compared to the weighted WQI models, a significant difference was found among models, whereas higher index variation was calculated for the models, respectively. With the exception of the Hanh index, there were no discernible differences between the models when compared to unweighted models. However, a comparatively large index score variation was found in the weighted SRDD and WJ models in river water over the study period.

Accuracy

In machine learning, accuracy is used to assess the categorization.

Precision is equal to TP plus TN. TP+TN+FP+FN

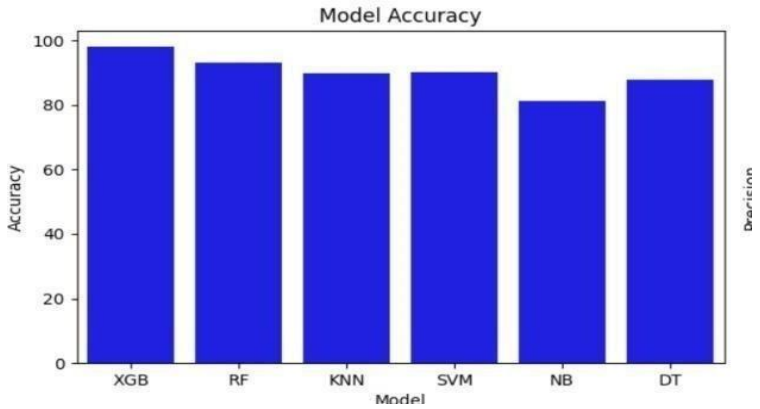


Fig 6.1: Model Accuracy

The term "precision" describes how well measurements of the same categorization are and algorithm predictions match up. In Fig, precision is ascertained as follows:

Repeatability = TP / TP+FP

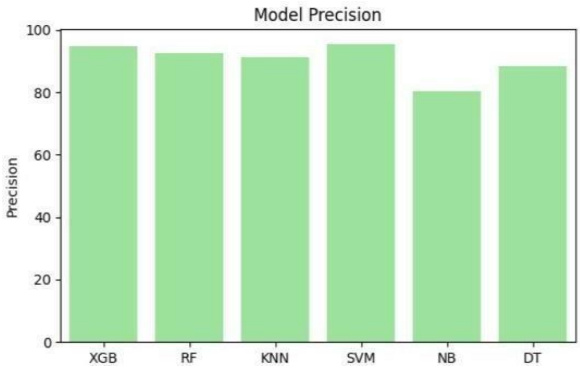


Fig 6.2: Model Precision.

Recall quantifies the frequency with which the algorithm determines the right classification from the provided data, even while the dataset contains instances of the correct categorization. While the observation courses are truly positive, false negatives are classified as negative. Recall is calculated as indicated in Fig:

Recall is equal to TP / TP+FN.

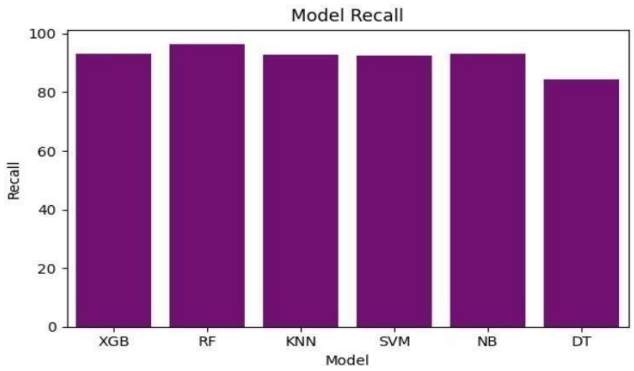


Fig 6.3: Model Recall.

The F1 score is a method for balancing the predictive model's precision and recall while simultaneously assessing multiclass categorization. F1 score is acquired as follows in Fig:

F1score=2*(precision*recall/precision+recall)

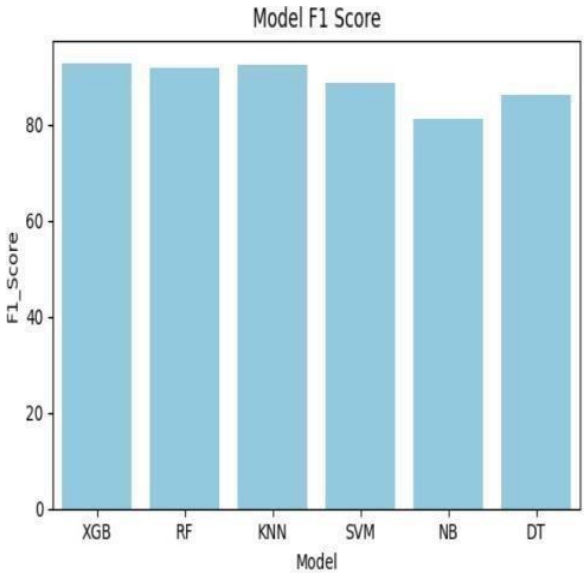


Fig.6.4 : Model F1 Score.

TP: Based on the provided data, actual observation shows that the water quality classes have been identified.

TN: The actual observation demonstrates that the classification of the water quality classes is accurate although the model has identified the proper classification based on the provided data.

FP: While the model also identifies the inaccurate classification of water quality from the provided data, the actual observation relates to the imprecise classification of water quality classes.

FN: Even though the model correctly classified the water quality based on the dataset it was given, the actual observation reveals that the water quality classes have not been correctly defined.

In this we have observed that the accuracy score after removing the outliers in the data the score has been increased, when compared with the considered base paper. In the following graph we can see the difference of scores after constructing the accuracy model with respect to Accuracy comparision.

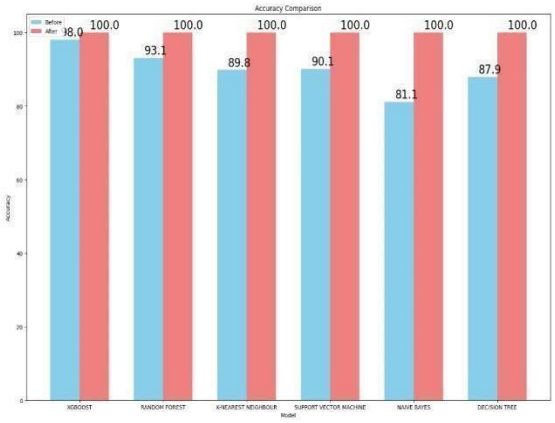


Fig.6.5 : Accuracy Comparison

In the precision we have observed that the precision score also have the high score compared to the scores of the dataset that we have taken. Precision determines the exact values of the scores of the data.

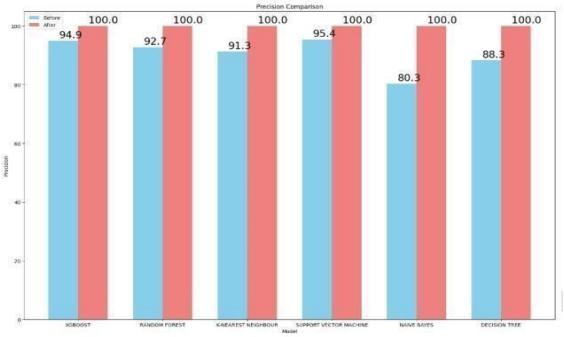


Fig 6.6 : Precision Comparison

F1 score of the data that we are taken the scores have increased after removing outliers andconstructing the model.

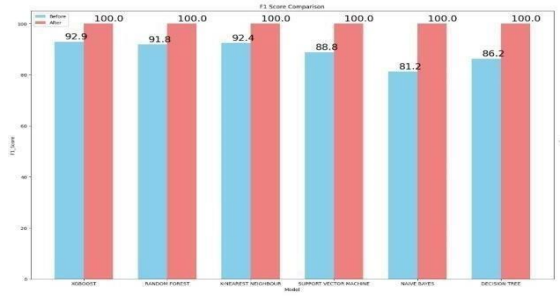


Fig 6.7: F1 score Comparison

When all the models has applied the recall score of the data has increased as compared to the existing data.

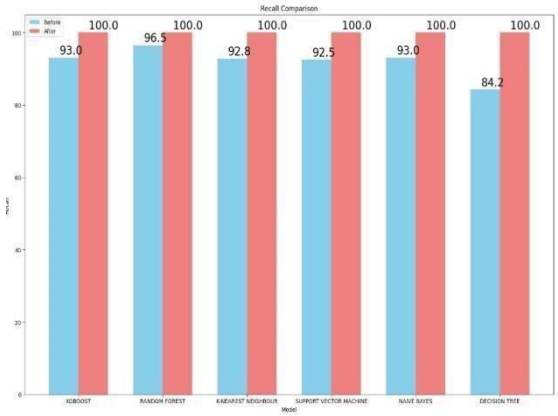


Fig 6.8 : Recall Comparison

The classification model that assigns a probability, confidence interval, or rating to each prediction is typically computed using the ROC curve. A number of models' algorithms, including SVM, Nave Bayes, and artificial neural networks, produce ranks. In order to reach different decision thresholds in each prediction step, spanning from the maximum to the lowest ranking value, the prediction ranking is frequently used in ROC algorithms. When it comes to classification, prediction-rating values are typically employed to normalise decision threshold values between 0 and 1, with 0.5 serving as the default threshold. The true positive and false positive rates at each threshold stage were used to generate the ROC curve, which represents the efficiency of the model. From the lower left corner to the top right corner, the traces a curve structurally.

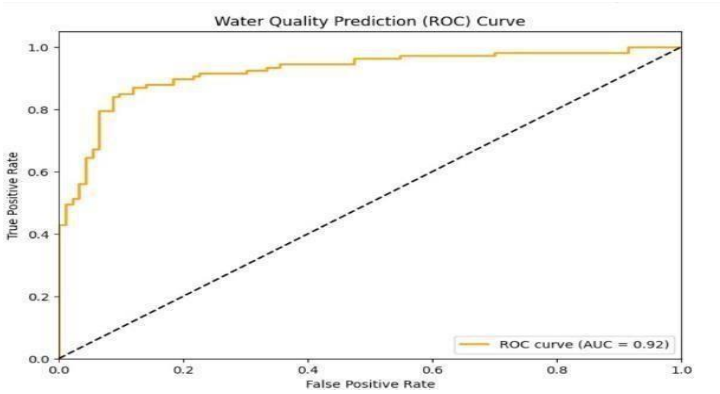


Fig 6.9: ROC curve

Our study's primary goal was to create a framework for evaluating the WQI model's performance in order to accurately classify water quality. After removing all the missing values in the data we have observed the accuracy of the algorithms have increased.

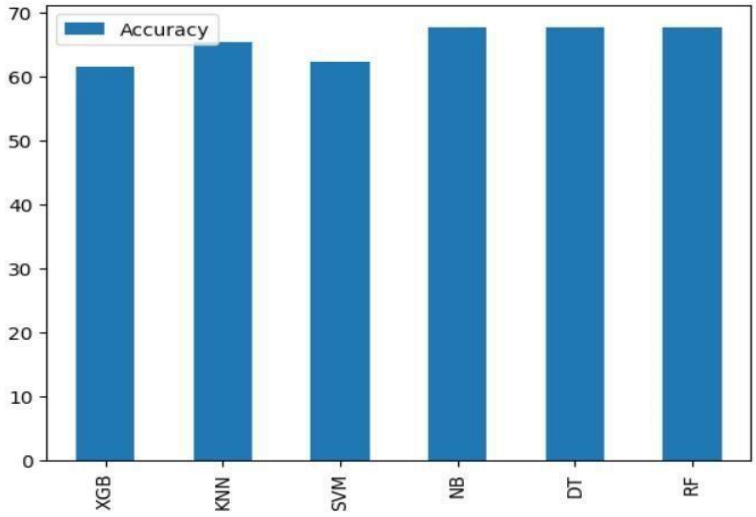


Fig 6.10 : Accuracy

7. Conclusion

This study's primary goal was to create a framework for evaluating the WQI model's performance in order to accurately classify the water quality of rivers. The best machine-learning classifier algorithm for predicting water quality class was found by comparing four different techniques. To the best of our knowledge, however, this work offers the first thorough method for assessing the effectiveness of WQI model(s) using a new categorization system for the multi-class classification of coastal water quality. Furthermore, the study's findings may be helpful in classifying water quality correctly, which might enhance the WQI model's accuracy, transparency, and dependability due to the accurate classification of coastal water quality. This study's main shortcoming was that it failed to take into account.

8. FUTURE SCOPE

The authors gratefully scoped the editor's and anonymous reviewers' contribution to the improvement of this study. This study was funded by the Hardiman Research Scholarship of the University of Galway, which funded the first author as part of his PhD programme. The authors would like to scope support from MaREI, the SFI Research Centre for Energy, Climate, and Marine research. The authors also would like to thank the Environmental Protection Agency of Ireland for providing water quality data. The authors sincerely scoped Charles Sturt University for providing all necessary supports to this PhD project through the excellent international co-supervision. Moreover, the authors also sincerely acknowledge the Eco HydroInformatics Research Group (EHIRG), School of Engineering, College of Science and Engineering, University of Galway, Ireland for providing computational laboratory facilities to complete this study.

9. REFERENCES

[1] Wang, Xianhe, Ying Li, Qian Qiao, Adriano Tavares, and Yanchun Liang. 2023. "Water Quality Prediction Based on Machine Learning and Comprehensive Weighting Methods" Entropy 25, no. 8: 1186.

[2] Integrating multisensor satellite data merging and image reconstruction in support of machine learning for better water quality management J. Environ. Manag. (2017)

[3] P. Wickramasinghe proposed a methodology to predict the performance of batsman for the previous five years using different machine learning algorithms,PP.64- 81,March 2018.

[4] P. Wickramasinghe proposed a methodology to predict the performance of batsman for the previous five years using different machine learning algorithms,PP.64- 81,March 2018 .

[5] Asselman, A., Khaldi, M., & Aammou, S. (2020). Evaluating the impact of prior required scaffolding items on the improvement of student performance prediction. Education and Information Technologies, 25, 3227–3249.

[6] Zhang Z. Introduction to machine learning: k-nearest neighbors. Ann Transl Med. 2016.Jun;4(11):218.doi:10.21037/atm.2016.03.37.PMID:27386492;PMCID:PMC4916348.

[7] K.-T. Chang et al .Modeling typhoon- and earthquakeinduced landslides in a mountainous watershed using logistic regression Geomorphology (2021)

[8] T.H.H. Aldhyani, M. Al-Yaari, H. Alkahtani, M. Maashi Water Quality Prediction Using Artificial Intelligence Algorithms Appl. Bionics Biomech. (2020)

[9] Patel HH, Prajapati P. Study and analysis of decision tree based classification algorithms. International Journal of Computer Sciences and Engineering. 2018 Oct 31;6(10):74-8.

[10][10] Kalyankar, G.D., Poojara, S.R., Dharwadkar, N.V.: Predictive analysis of diabetic patient data using machine learning and hadoop. In: International Conference On ISMAC (2017). ISBN 978-1-5090-3243-3

Water Quality Prediction Using MachineLearning

Y. Chandana
Asst.Proffesor
Computer Science and Engineering
Narasaraopeta Engineering College
Narasaraopet
Andhra Pradesh
Chandana.nrtec@gmail.com

B. Deepthi
Student
Computer Science and Engineering
Narasaraopeta Engineering College
Narasaraopet
Andhra Pradesh
deepthi01902@gmail.com

K. Yamini Saisri
Student
Computer Science and Engineering
Narasaraopeta Engineering College
Narasaraopet
Andhra Pradesh
kysaisri@gmail.com

Sk. Farzana
Student
Computer Science and Engineering
Narasaraopeta Engineering College
Narasaraopet
Andhra Pradesh
skfarzana2244@gmail.com

M. Bharani
Student
Computer Science and Engineering
Narasaraopeta Engineering College
Narasaraopet
Andhra Pradesh
madhavabotlabharani201@gmail.com

Abstract— **In everyday life, water is the most valuable resource for all humans. Water is primarily used for drinking and agriculture. Because of this, it is crucial to assess the water's quality to determine whether it is safe to use or drink or whether it may contain harmful chemicals. If any of these substances are present, the water is deemed unsafe for use or consumption.**

A thorough investigation and a wide range of machine learning algorithms, such as Decision Trees, XGBoost, Support Vector Machines , and ensemble methods like Random Forest, are applied in the water quality prediction and KNN, in forecasting key water quality metrics such as Temperature, PH, and Conductivity.

In result, our study underscores the transformative impact of ML on environmental science, offering to address the pressing challenge of maintaining quality of water in an era of unprecedented environmental change.

Keywords— Water Quality Prediction, Support Vector Machine , Random Forest, Decision Tree, KNN, XG Boost.

I. INTRODUCTION

The primary objective of water quality prediction is to provide timely insights into potential issues, such as contamination [1] events or ecological disruptions, enabling proactive management and intervention strategies. By leveraging machine learning techniques, predictions [2] can be made with greater accuracy and efficiency compared to traditional methods.

2. LITERATURE SURVEY

ML techniques are used to predict the water state through performance analysis of the water quality index model. To find the optimal solution for the water quality we used the algorithms including XG Boost, KNN, SVM, Navie Bayes, Decision Tree, and Random Forest.

Both the environment and public health are directly impacted by water quality. Water is utilised for several purposes, including industrial, agriculture, and drinking. The growth of water entertainment and sports in recent years has been a major factor in drawing tourists. Because rivers are the easiest to obtain among water sources, human cultures have relied on them more than other sources for their development.

Horrible diseases have resulted from the alarming rate at which water quality is deteriorating due to rapid urbanisation and industrialization. The traditional method of estimating water quality involves costly and timeconsuming statistical and laboratory analysis.

Our suggested approach is dependent on variables such as temperature, pH, turbidity, solids, hardness, and so forth.. Without a question, accurate forecasting will raise the bar for water resource management. Many water resource management agencies currently have monitoring stations set up to keep an eye on changes in the quality of the water.

3. Methodology

The WQI score for coastal water quality was stored using recently developed, improved WQI methodologies, and the water quality classes were established by utilizing the coastal water quality categorization scheme.

.

In the Fig.1 overview of this water Quality Prediction has been done through loading the data preprocessing into the model inputs and after the training and testing set algorithms. ROC curve analysis is done. Develop unique classification scheme based on the best cut-point of ROC.

With this method, you will be able to create a predictive model that, depending on a number of factors, can precisely anticipate the quality of the water. Data collection, preprocessing, feature and model selection, training, testing, and deployment are all included in the methodology.

.

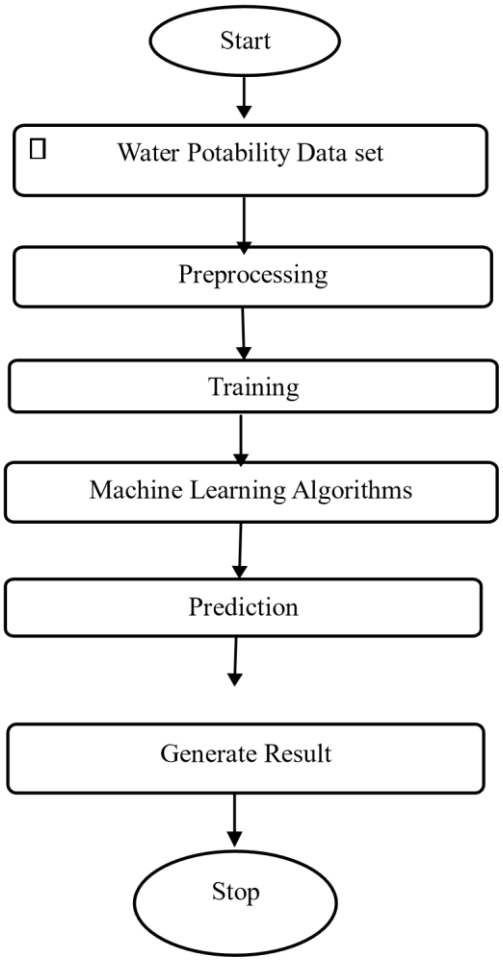


Fig 1: Overview of water Quality prediction

3. PROPOSED SYSTEM

Our model is proposed is based in the following criteria:

3.1. Dataset Analysis

3.2. Data Visualization

3.3. Preprocessing Techniques

3.4. Model creation and Evaluation

3.5. Accuracy

3.1. Dataset Analysis: We are taken the dataset from the Kaggle website. For that we have the Fig.2 water_potability.csv (2022) dataset with 10 columns, those are Temperature, pH, Solids, Hardness, Chloramines, Sulfates, Organic_carbon, Turbidity, Trihalomethanes, Potability. Among all those attributes Potability is considered as the target attribute.

	ph	Hardness	Solids	Chloramin	Sulfate	Conductivi	Organic_c	Trihalome	Turbidity	Potability
2		204.8905	20791.32	7.300212	368.5164	564.3087	10.37978	86.99097	2.963135	0
3	3.71608	129.4229	18630.06	6.635246		592.8854	15.18001	56.32908	4.500656	0
4	8.099124	224.2363	19909.54	9.275884		418.6062	16.86864	66.42009	3.055934	0
5	8.316766	214.3734	22018.42	8.059332	356.8861	363.2665	18.43652	100.3417	4.628771	0
6	9.092223	181.1015	17978.99	6.5466	310.1357	398.4108	11.55828	31.99799	4.075075	0
7	5.584087	188.3133	28748.69	7.544869	326.6784	280.4679	8.399735	54.91786	2.559708	0
8	10.22386	248.0717	28749.72	7.513408	393.6634	283.6516	13.7897	84.60356	2.672989	0
9	8.635849	203.3615	13672.09	4.563009	303.3098	474.6076	12.36382	62.79831	4.401425	0
10		118.9886	14285.58	7.804174	268.6469	389.3756	12.70605	53.92885	3.595017	0
11	11.18028	227.2315	25484.51	9.0772	404.0416	563.8855	17.92781	71.9766	4.370562	0
12	7.36064	165.5208	32452.61	7.550701	326.6244	425.3834	15.58681	78.74002	3.662292	0
13	7.974522	218.6933	18767.66	8.110385		364.0982	14.52575	76.48591	4.011718	0
14	7.119824	156.705	18730.81	3.606036	282.3441	347.715	15.92954	79.50078	3.445756	0
15		150.1749	27331.36	6.838223	299.4158	379.7618	19.37081	76.51	4.413974	0
16	7.496232	205.345	28388	5.072558		444.6454	13.22831	70.30021	4.777382	0
17	6.347272	186.7329	41065.23	9.629596	364.4877	516.7433	11.53978	75.07162	4.376348	0
18	7.051786	211.0494	30980.6	10.0948		315.1413	20.39702	56.6516	4.268429	0
19	9.18156	273.8138	24041.33	6.90499	398.3505	477.9746	13.38734	71.45736	4.503661	0
20	8.975464	279.3572	19460.4	6.204321		431.444	12.88876	63.82124	2.436086	0
21	7.37105	214.4966	25630.32	4.432669	335.7544	469.9146	12.50916	62.79728	2.560299	0
22		227.435	22305.57	10.33392		554.8201	16.33169	45.38282	4.133423	0
23	6.660212	168.2837	30944.36	5.858769	310.9309	523.6713	17.88424	77.04232	3.749701	0
24		215.9779	17107.22	5.60706	326.944	436.2562	14.18906	59.85548	5.459251	0
25	3.902476	196.9032	21167.5	6.996312		444.4789	16.60903	90.18168	4.528523	0
26	5.400302	140.7391	17266.59	10.05685	328.3582	472.8741	11.25638	56.93191	4.824786	0

Fig 2: water potability Dataset

3.2. Data Visualization:

To visualize the data in the form of the graphs we used numpy, pandas, seaborn, pyplotlib, sklearn libraries. Each and every single library is used to represent the graph in detailed manner. This helps us to recognize potability of water among all the combined compounds.

Data visualization techniques such as histograms, scatter plots and box plot will allow us to explore the distribution of data, identify patterns, correlations, outliers and potential relation between variables.

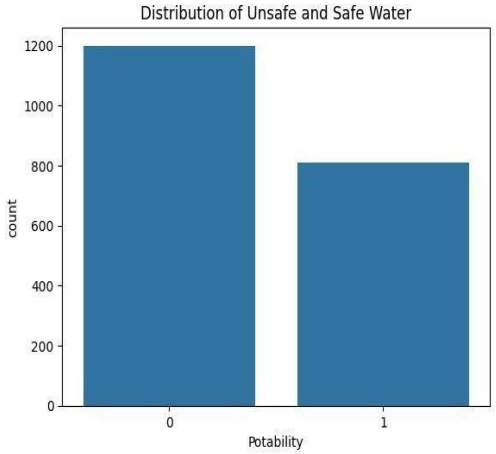


Fig 3: Distribution of safe & unsafe water.

This Fig.3 shows us the distribution of safe and unsafe water, count is taken on y-axis and potability is taken on xaxis it explains the count of occurrences for each category of potability which represents whether water is potable or not.

Classifier algorithms	Rank (based on the model accuracy)	Accuracy (%)	
		Training	Testing
XGBoost	1	98.0	100
KNN	2	93.1	94.0
SVM	3	89.8	92.0
NB	4	90.1	91.0
DT	5	81.1	85.3

Fig 4: Model performance on different classifiers

Water Quality Index (WQI)

We can see the models that are run on various classifiers for model prediction based on accuracy in the Fig. 4 above. Recently, an updated and comprehensive WQI technique has been presented [11] for assessing the costal and transitional water quality by computing WQI scores, with the goal of lowering model uncertainty.

3.3. Preprocessing Techniques

Correlation: A useful tactic to lower dimensionality, eliminate unnecessary data, and improve learning accuracy is feature selection. [12] it is the statistical summary of the relationship between two sets of variables it tells about how two variables move in relation to one another. Techniques for estimating a linear model's impulse response are referred to as correlation analysis.

By applying the correlation we have observed that two attributes are correlated. Hence for the further process we didn't make any changes to our dataset.

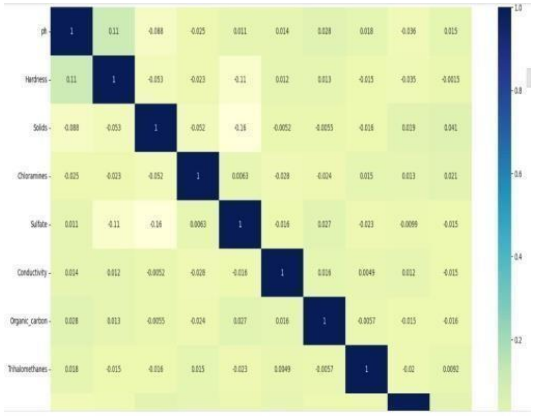


Fig 5: correlation matrix

Outliers are datapoints that significantly refer from other observe from the dataset they (Fig.5) can arise due to measurement error, experimental variability, genuine deviations in the data, It is essential to identify outliers in the existing data using visualizations like box plots.in some cases outliers can be removed from the dataset if they are irrelevant to the analysis. Outliers are infrequent observations between a dataset the problem handling is essential to ensure accurate reasons.

The data points may occur for a variety of causes, including measurement errors, data rectification, or truly uncommon occurrences. It's critical to manage outliers since they can skew statistical analysis and machine learning models, leading to inaccurate results and decreased model performances.

It's critical to manage outliers since they can skew statistical analysis and machine learning models, leading toinaccurate results and decreased model performances.

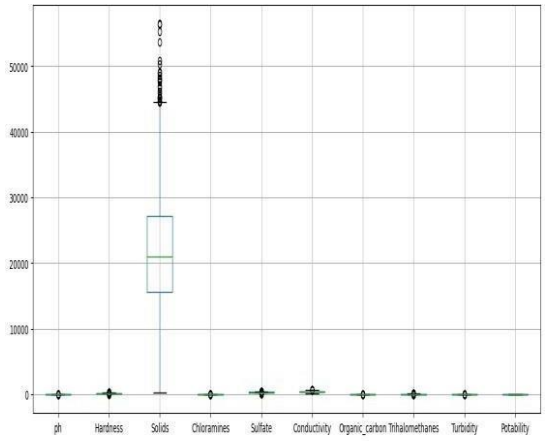


Fig 6: Boxplot of the dataset.

Null values: In water potability.csv dataset we have observed many null values present in the dataset for that we have removed all the null values which will affect to our further preprocessing techniques as shown in Fig.6.

And in these preprocessing techniques we have observed the factors that are affecting to the main attribute (Potability). We have clearly explained the factors that affect to each and every attribute to understand clearly about the factors to know the relation between dependent and independent attributes.

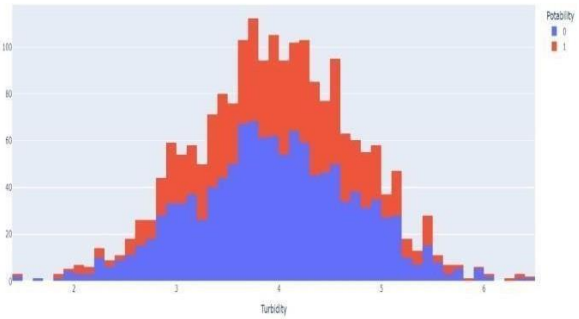


Fig 7

: Factors affecting target attribute

We used the function SKEW which calls and appears to be related to calculating Fig.7 the skewness of information on the designated axis. The dataset's asymmetry in value distribution is quantified using the skew() function.

```
ph                0.048947
Hardness         -0.085237
Solids            0.595894
Chloramines       0.012976
Sulfate          -0.046558
Conductivity      0.266869
Organic_carbon   -0.020018
Trihalomethanes  -0.051422
Turbidity        -0.033051
Potability        0.394614
dtype: float64
```

Fig 8: correlation coefficients

As demonstrated in, skewness is a measurement of the asymmetry of a real-valued random variable's probability distribution. Fig 8. In simpler terms, it tells us how much and in which direction a distribution deviates from a normal distribution.

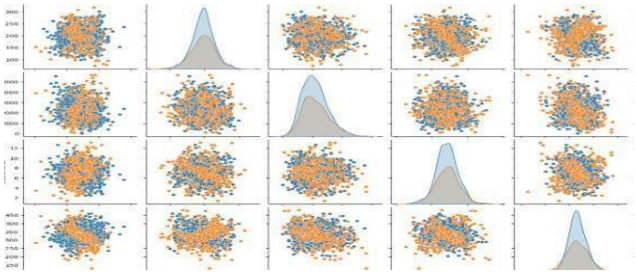


Fig 9: Pair plot of the dataset

A pair plot is a versatile and powerful visualization tool used to understand the relationships between multiple variables within a dataset. It presents a comprehensive overview by plotting pairwise relationships across all the variables as represented in Fig.9. This can help in identifying patterns, correlations, and potential hypotheses for more detailed analysis.
Diagonal plots are usually histograms or density plots showing the distribution of a single variable.

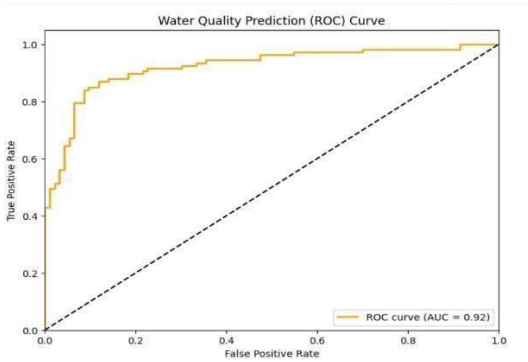


Fig 10: ROC curve

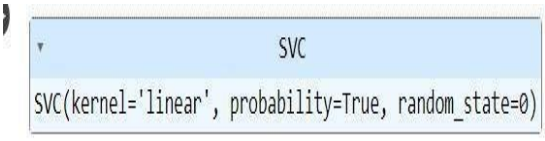
The Roc curve incorporates a probability confidence interval or rating into the algorithm of each prediction model, including SVM and naïve bayes. This is seen in Figure 9. In essence, the ROC algorithm uses prediction ranking to produce different decisions.

3.4. Model Creation and Evaluation:

XGBoost: This algorithm for machine learning is employed in specially the gradient boosting frame work. It uses decision tree as base learners for model generalization. XGBoost is commonly used to perform assignments like regression, categorization, and ranking [5].

K-Nearest Neighbour :This technique is categorised as a means of classifying and identifying unlabeled data by comparing them to similar labelled samples. For the training and test data sets, KNN classifier characteristics are gathered [6].Since KNN is a non-parametric technique, it cannot determine the model's parameters. In order to determine the variation in the diagnostic accuracy of the KNN model, K is the most crucial parameter in the KNN method.

SupportVectorMachine (SVM): SVM is a binary classifier it attempts to generate an another hyper plane in actual space of end coordinates between two different classes. Firstly it visualizes the data and then it finds the best separator between the classes, the data points that the closest to the target value [7] be found out, then the actual data in accordance with the linear separation. Basically SVM draws the line between the different points of data considers in the dataset.



Naïve Bayes :It is a scalable and it requires a set of parameters that correspond to the quantity of variables in a learning task. It determines a probability by calculating the likelihood of two independent features with equal weight. Navie Bayes theorem generally works on the phases known as; [8] probability and independence, training phase, feature probability, class probability, prediction, class selection.

Decision Tree : Decision tree is easy to understand because it is similar to human decision making process, it can solve continuous data as input.[9].The main advantage is that it

can be able to select the most biased feature and comprehensibility nature.

```
DecisionTreeClassifier
DecisionTreeClassifier(criterion='entropy', random_state=0)
```

Random Forest (RF): A novel combination algorithm called Random Forest isa combination of series of structure classifiers like tree the application scope of random forest is very extensive it is widely used for prediction, classification and regression [10] compared with other traditional algorithms random forest has many good virtues. During the training process random forest can operate learning method by constructing a large number of decision trees.

A novel combination algorithm called Random Forest isa combination of series of structure classifiers like tree the application scope of random forest is very extensive it is widely used for prediction, classification and regression [10] compared with other traditional algorithms random forest has many good virtues. During the training process random forest can operate learning method by constructing a large number of decision trees.

```
RandomForestClassifier
RandomForestClassifier(criterion='entropy', n_estimators=11, random_state=0)
```

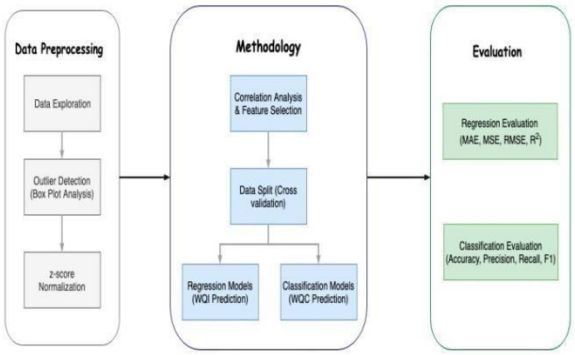


Fig 11: Water Quality Prediction using Supervised Machine

Learning

In the above figure.11 we can observe the methodology used to forecast the water's quality using supervised Machine Learning. In this study we can see theEvaluation and data preprocessing.

3.5. Accuracy

In machine learning, accuracy is used to assess the categorization[13].

Precision is equal to TP plus TN. TP+TN+FP+FN

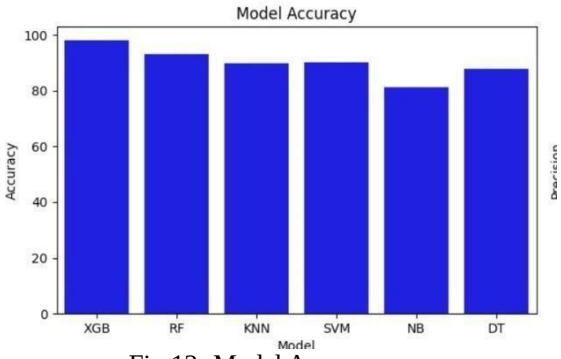


Fig 12: Model Accuracy score

The term "precision" describes how well measurements of the same categorization area[11] and algorithm predictions match up. In Fig. 13, precision is ascertained as follows:

Repeatability = TP / TP+FP

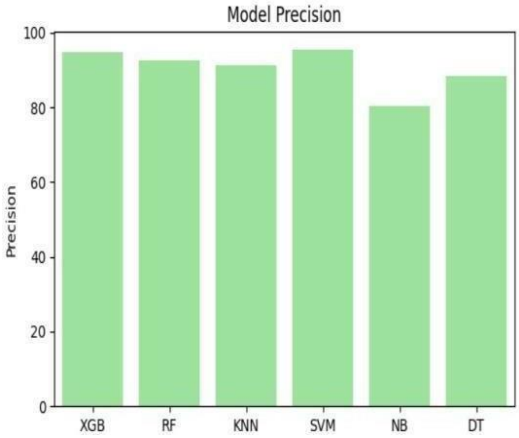


Fig 13: Model precision Score

Recall quantifies the frequency with which the algorithm determines the right classification from the provided data, even while the dataset contains instances of the correct categorization. While the observation courses are truly positive, false negatives are classified as negative. Recall is calculated as indicated in Fig. 14:

Recall is equal to TP / TP+FN.

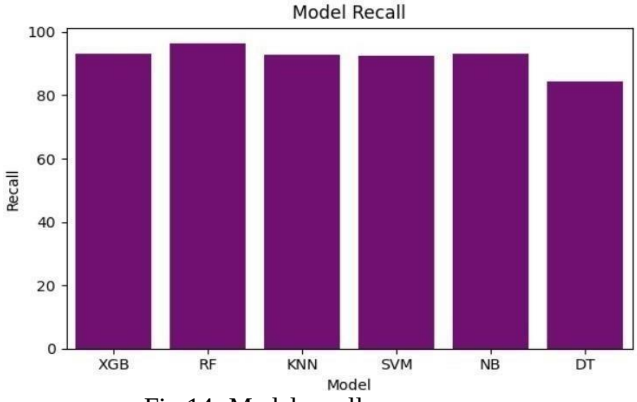


Fig 14: Model recall score

The F1 score is a method for balancing the predictive model's precision and recall while simultaneously

assessing multiclass categorization. F1 score is acquired as follows in Fig. 15:

F1score=2*(precision*recall/precision+recall)

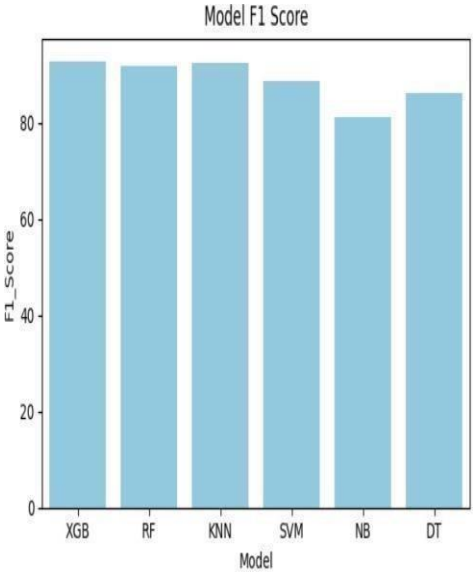


Fig 15: Model F1 score

TP: Based on the provided data, actual observation shows that the water quality classes have been identified.

TN: The actual observation demonstrates that the classification of the water quality classes is accurate although the model has identified the proper classification based on the provided data.

FP: While the model also identifies the inaccurate classification of water quality from the provided data, the actual observation relates to the imprecise classification of water quality classes.

FN: Even though the model correctly classified the water quality based on the dataset it was given, the actual observation reveals that the water quality classes have not been correctly defined.

.

In this we have observed that the accuracy score after removing the outliers in the data the score has been increased, when compared with the considered base paper. In the following graph we can see the difference of scores after constructing the accuracy model with respect to Accuracy comparision.

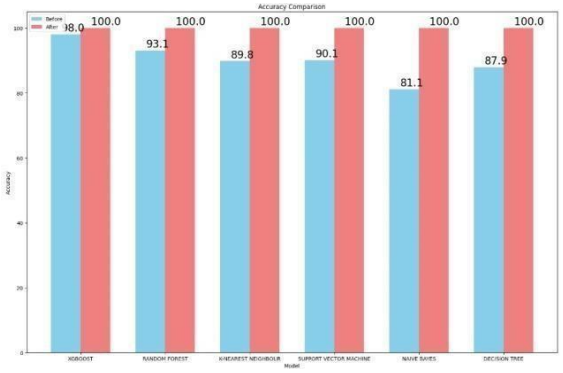


Fig 16: Accuracy Comparision

In the precision we have observed that the precision score also have the high score compared to the scores of the dataset that we have taken. Precision determines the exact values of the scores of the data.

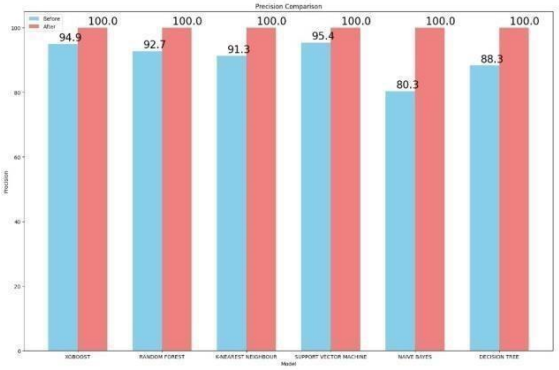


Fig 17: Precision Comparision

F1 score of the data that we are taken the scores have increased after removing outliers and constructing the model.

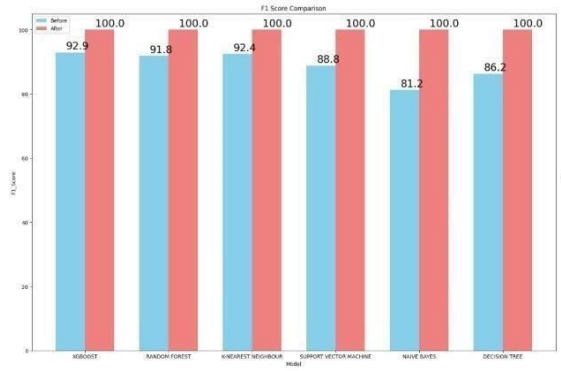


Fig 18: Recall Comparision

When all the models has applied the recall score of the data has increased as compared to the existing data.

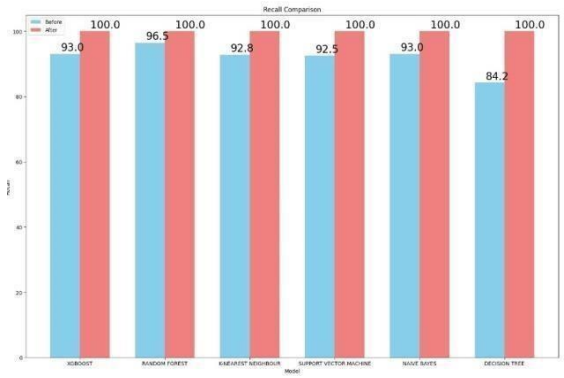


Fig 19: F1 score Comparision

After applying all the models to our data the accuracy of the data we have observed is plotted in the following graph as in x-axis we can see the models that we have used and in yaxis we can see the rate of accuracy of the constructed model.

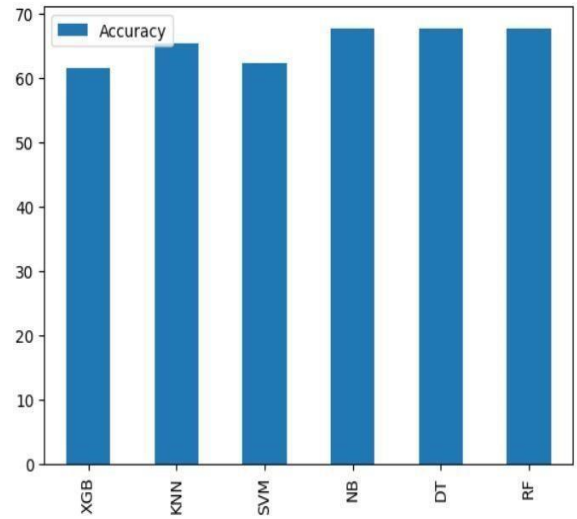


Fig 20: Accuracy

Conclusion

Our study's primary goal was to create a framework for evaluating the WQI model's performance in order to accurately classify water quality. After removing all the missing values in the data we have observed the accuracy of the algorithms have increased.

Differences

Evaluation of WQI: ModelsSeven WQI models, both commonly used and recently proposed ones, were tested in the study. This comprehensive assessment aimed to identify the most suitable models for accurately assessing water quality along the coast.

Performance of Machine Learning Algorithms: The XGBoost algorithm and KNN (K-Nearest Neighbors) demonstrated superior performance in correctly classifying water quality. XGBoost particularly excelled, achieving accurate classification for most classifications of water quality, with the exception of "poor" quality.

Effectiveness of WQI Models: The weighted WQM-WQI and unweighted RMS-WQI models have been shown to be useful instruments for precisely determining the state of coastal water quality. Cork Harbour's water quality was successfully divided into "Good" and "Fair" classifications by these models.

Novel Contribution: The study is the first to provide a comprehensive approach to evaluating WQI model performance, implementing a novel multi-class classification scheme for coastal water quality. This new method improves comprehension and makes it easier to increase the correctness, transparency, and dependability of the WQI model.

Limitations and Future Directions: The study was severely limited by its failure to take into account the temporal variability of Cork Harbour's water quality measures.
In order to overcome this constraint, future research could evaluate the WQI model's effectiveness using indicators with temporal resolution and maybe include different predictive classifier methods.

Implications: Despite their limitations, the study's findings are helpful in lowering the possibility that incorrect classification may lead to model uncertainty. The insights provided can inform researchers, policymakers, and water resource personnel, facilitating more informed decisionmaking regarding coastal water quality management and conservation efforts.
Its findings contribute to advancing understanding in this field and have practical implications for environmental management and policy development.

References:

[1] Wang, Xianhe, Ying Li, Qian Qiao, Adriano Tavares, and Yanchun Liang. 2023. "Water Quality Prediction Based on Machine Learning and Comprehensive Weighting Methods" Entropy 25, no. 8: 1186.

[2] Integrating multisensor satellite data merging and image reconstruction in support of machine learning for better water quality management J. Environ. Manag. (2017)

[3] P. Wickramasinghe proposed a methodology to predict the performance of batsman for the previous five years using different machine learning algorithms,PP.64- 81,March 2018.

[4] P. Wickramasinghe proposed a methodology to predict the performance of batsman for the previous five years using different machine learning algorithms,PP.64- 81,March 2018 .

[5] Asselman, A., Khaldi, M., & Aammou, S. (2020). Evaluating the impact of prior required scaffolding items on the improvement of student performance prediction. *Education and Information Technologies*, 25, 3227–3249.

[6] Zhang Z. Introduction to machine learning: knearest neighbors. Ann Transl Med. 2016.

Jun;4(11):218.doi:10.21037/atm.2016.03.37.
PMID: 27386492; PMCID: PMC4916348.

[7] K.-T. Chang *et al* .Modeling typhoon- and earthquakeinduced landslides in a mountainous watershed using
logistic regression Geomorphology (2021)

[8] T.H.H. Aldhyani, M. Al-Yaari, H. Alkahtani, M. Maashi
Water Quality Prediction Using Artificial Intelligence Algorithms Appl. Bionics Biomech. (2020)

[9] Patel HH, Prajapati P. Study and analysis of decision tree based classification algorithms. International Journal of Computer Sciences and Engineering. 2018 Oct 31;6(10):74-8
.

[10] Kalyankar, G.D., Poojara, S.R., Dharwadkar, N.V.: Predictive analysis of diabetic patient data using machine learning and hadoop. In: International Conference On ISMAC (2017). ISBN 978-1-5090-3243-3

[11] River water quality index prediction anduncertainty analysis: A comparative study of machine learning models
J.Environ.Chem.Eng., 9 (2021),
Article 104599, 10.1016/j.jece.2020.104599

Base paper DG5 plag.docx

ORIGINALITY REPORT

8%	6%	7%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	researchoutput.csu.edu.au Internet Source	4%
2	Md Galal Uddin, Stephen Nash, Azizur Rahman, Agnieszka I. Olbert. "Performance analysis of the water quality index model for predicting water state using machine learning techniques", Process Safety and Environmental Protection, 2022 Publication	1%
3	Jalaney. J, Ganesh. R.S. "Accurate Bus Arrival Time from Linear and Non-Linear Route Parameters Using Hybrid Predictors", 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC), 2021 Publication	1%
4	Vince Paul, R. Ramesh, P. Sreeja, T. Jarin, P.S. Sujith Kumar, Sabah Ansar, Ghulam Abbas Ashraf, Sadanand Pandey, Zafar Said. "Hybridization of long short-term memory with Sparrow Search Optimization model for	<1%

water quality index prediction", Chemosphere, 2022
Publication

5	iemtronics.org Internet Source	<1%
6	Mushtaque Ahmed Rahu, Abdul Fattah Chandio, Khursheed Aurangzeb, Sarang Karim, Musaed Alhussein, Muhammad Shahid Anwar. "Towards design of Internet of Things and machine learning-enabled frameworks for analysis and prediction of water quality", IEEE Access, 2023 Publication	<1%
7	link.springer.com Internet Source	<1%
8	www.i-scholar.in Internet Source	<1%
9	Nida Nasir, Afreen Kansal, Omar Alshaltone, Feras Barneih, Mustafa Sameer, Abdallah Shanableh, Ahmed Al-Shamma'a. "Water quality classification using machine learning algorithms", Journal of Water Process Engineering, 2022 Publication	<1%

Exclude quotes On Exclude bibliography On

Exclude matches	Off