International Research Journal in Global Engineering and Sciences. (IRJGES)
ISSN : 2456-172X | Vol. 6, No. 2, June-August, 2021 | Pages 18-43

Water Quality Prediction using Machine Learning Algorithms

M V Jerabandi, mark.jerabandi@gmail.com
Deepa E K [1], Jessica S S[2], Rakshita R S[3], Shweta T M[4]
1, deepudee583@gmail.com 2, mahimasindhe5212@gmail.com
3, rakshitas999@gmail.com 4, shwetameti11@gmail.com
Rural Engineering College Hulkoti, Gadag Karnataka

ABSTRACT

This study investigates the performance of artificial intelligence techniques including artificial neural network (ANN), group method of data handling (GMDH) and support vector machine (SVM)for predicting water quality components of Tireh River located in the southwest of Iran. To develop the ANN and SVM, different types of transfer and kernel functions were tested, respectively. Reviewing the results of ANN and SVM indicated that both models have suitable performance for predicting water quality components. During the process of development of ANN and SVM, it was found that tansig and RBF as transfer and kernel functions have the best performance among the tested functions. Comparison of outcomes of GMDH model with other applied models shows that although this model has acceptable performance for predicting the components of water quality, its accuracy is slightly less than ANN and SVM. The evaluation of the accuracy of the applied models according to the error indexes declared that SVM was the most accurate model. Examining the results of the models showed that all of them had some over-estimation properties. By evaluating the results of the models based on the DDR index, it was found that the lowest DDR value was related to the performance of the SVM model.

1.0 Introduction

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 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.

1.1 Background

Access to safe drinking-water is essential to health, a basic human right and a component of effective policy for health protection. This is important as a health and development issue at a national, regional and local level. In some regions, it has been shown that investments in water supply and sanitation can yield a net economic benefit, since the reductions in adverse health

effects and health care costs outweigh the costs of undertaking the interventions. 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. are considered. To this end, governments have constructed hydrometer stations along rivers that cross from urban areas, agro-industrial projects, industrial estates, and rivers that join dams' reservoirs. In hydrometer stations, the water quality components are measured and the stage-discharge relation is defined. Obtained values from hydrometer 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 analyze time series of water quality components and their internal relation. In this regard, used multilayer perceptron (MLP), radial basis network (RBF) and an adaptive neuron-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.

1.2 Machine learning

Machine learning is the study of computer algorithms that can improve automatically through experience and by the use of data.[1] It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so.[2] Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision, where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks.

Machine learning programs can perform tasks without being explicitly programmed to do so. It involves computers learning from data provided so that they carry out certain tasks. For simple tasks assigned to computers, it is possible to program algorithms telling the machine how to execute all steps required to solve the problem at hand; on the computer's part, no learning is needed. For more advanced tasks, it can be challenging for a human to manually create the needed algorithms. In practice, it can turn out to be more effective to help the machine develop its own algorithm, rather than having human programmers specify every needed step.

1.3 Machine learning Algorithm

Broadly, there are 3 types of Machine Learning Algorithms viz. Supervised Learning, Unsupervised Learning and Reinforcement Learning.

- **Supervised Learning:** This algorithm consists of a target / outcome variable (or dependent variable) which is to be predicted from a given set of predictors (independent variables). Using these set of variables, we generate a function that map inputs to desired outputs. The training process continues until the model achieves a desired level of accuracy on the training data. Examples of Supervised Learning: Regression, Decision Tree, Random Forest, KNN, Logistic Regression etc.

- **Unsupervised Learning**: In this algorithm, we do not have any target or outcome variable to predict / estimate. It is used for clustering population in different groups, which is widely used for segmenting customers in different groups for specific intervention. Examples of Unsupervised Learning: Apriori algorithm, K means.

- **Reinforcement Learning:** Using this algorithm, the machine is trained to make specific decisions. It works this way: the machine is exposed to an environment where it trains itself continually using trial and error. This machine learns from past experience and tries to capture the best possible knowledge to make accurate business decisions. Example of Reinforcement Learning: Markov Decision Process.

1.4 Literature Survey

In computer science, water quality prediction using machine learning play an important role. Wide variety of research papers are available in this area of machine learning. As part of literature survey, many research papers and journals are referred. Few of the papers referred are briefly discussed in this section.

Saber Kouadri, Ahmed Elbeltagi, Abu Reza Md., Towfqul Islam, Samir Kateb (ref 7) Year, :2021a paper entitled “Performance of machine learning methods in predicting water quality index based on irregular data set: application on Illizi region (Algerian southeast)”, presented 8 artificial intelligence algorithms e.g., multi linear regression (MLR), random forest (RF), M5P tree (M5P), random subspace (RSS), additive regression (AR), artificial neural network (ANN), support vector regression (SVR), and locally weighted linear regression (LWLR), were employed to generate WQI prediction in Illizi region, southeast Algeria. Using the best subset regression, 12 different input combinations were developed and the strategy of work was based on two scenarios. The first scenario aims to reduce the time consumption in WQI computation, where all parameters were used as inputs. The second scenario intends to show the water quality variation in the critical cases when the necessary analyses are unavailable.

Iran Atefeh Nouraki1 & Mohammad Alavi1 & Mona Golabi1 & Mohammad Albaji. Year :2021 (ref3), the machine learning models have been successfully implemented for modeling total dissolved solids (TDS), sodium absorption ratio (SAR) and total hardness (TH) content in aquatic ecosystems with insufficient data. Multiple linear regression (MLR), M5P model tree, support vector regression (SVR) and random forest regression (RFR) models were used to predict in the paper titled ""Prediction of water quality parameters using machine learning models: a case study of the Karun River".

"Coastal Water Quality Prediction Based on Machine Learning with Feature Interpretation and Spatio–temporal Analysis", Luka Grbcic, Goran MausaYear:2021 (ref8), presented routine monitoring data of Escherichia Coli and enterococci across 15 public beaches in the city of Rijeka, Croatia, were used to build machine learning models for predicting their levels based on environmental parameters as well as to investigate their dynamics and relationships with environmental stressors. Gradient Boosting algorithms (Catboost, Xgboost), Random Forests, Support Vector Regression and Artificial Neural Networks were trained with routine monitoring measurements from all sampling sites and used to predict E. Coli and enterococci values based on environmental features. The evaluation of stability and generalizability with 10-fold cross validation analysis of the machine learning models is proposed.

Md. Saikat Islam Khan ad, Nazrul Islam bd, Jia Uddin c, Sifatul Islam, Mostofa Kamal NasirYear:2021 (Ref11),"Water quality prediction and classification based on principal component regression and gradient boosting classifier approach", proposed the water quality prediction model utilizing the principal component regression technique. Firstly, the water quality index (WQI) is calculated using the weighted arithmetic index method. Secondly, the principal component analysis (PCA) is applied to the dataset, and the most dominant WQI parameters have been extracted. Thirdly, to predict the WQI, different regression algorithms are used to the PCA output. Finally, the Gradient Boosting Classifier is utilized to classify the water quality status.

Advanced artificial intelligence (AI) algorithms are developed to predict water quality index (WQI) and water quality classification (WQC). For the WQI prediction, artificial neural network models, namely nonlinear autoregressive neural network (NARNET) and long short-term memory (LSTM) deep learning algorithm, have been developed. In addition, three machine learning algorithms, namely, support vector machine (SVM), K-nearest neighbour (K-NN), and Naive Bayes, have been used for the WQC forecasting. These are explained in a paper titled "Water Quality Prediction Using Artificial Intelligence Algorithms", Theyazn H. H Mohammed Al-Yaari, Hasan Alkahtani, and MashaelMaashi Year:2020 (Ref 2).

Umair Ahmed, Rafia Mumtaz, Hirra Anwar, Asad A. Shah, Rabia Irfan and José García-Nieto Year:2019(Ref1), has published a paper "Efficient Water Quality Prediction Using Supervised Machine Learning", the proposed methodology employs four input parameters, namely, temperature, turbidity, pH and total dissolved solids. Of all the employed algorithms, gradient boosting, with a learning rate of 0.1 and polynomial regression, with a degree of 2, predict the WQI most efficiently, having a mean absolute error (MAE) of 1.9642 and 2.7273, respectively. Whereas multi-layer perceptron (MLP), with a configuration of (3, 7), classifies the WQC most efficiently, with an accuracy of 0.8507.

Ali NajahAhmeda, FaridahBinti Othman, Year:2019 (Ref 5), "Machine learning methods for better water quality prediction ". The study proposes the use of enhanced Wavelet De-noising Techniques using Neuro-Fuzzy Inference Systems (WDT-ANFIS) according to historical water-quality parametric data. The effectiveness of each model was examined in order to predict key parameters that could be affected as a result of urbanization surrounding rivers. This area of research accords with the available secondary data for each water quality parameter of Johor River. The parameters comprise ammoniacal nitrogen (AN), suspended solid (SS), and pH.

"Water quality prediction using machine learning methods", Amir Hamzeh Haghiabi, Ali Heider Nasrolahi and Abbas Parsaie. Year:2018, proposed method investigates the performance of artificial intelligence techniques including artificial neural network (ANN), group method of data handling (GMDH) and support vector machine (SVM) for predicting water quality components of Tireh River located in the southwest of Iran. To develop the ANN and SVM, different types of transfer and kernel functions were tested, respectively. Reviewing the results of ANN and SVM indicated that both models have suitable performance for predicting water quality components.

Xiu Li, Jingdong Song, year:2015," "A New ANN-Markov Chain Methodology for Water Quality Prediction", have discussed artificial neural network and Markov chain approach are used to develop a new hybrid methodology for predicting the biochemical oxygen demand which is the main indicator of water quality. ANN produces the primary values and then the results are modified by three regression methods using the Markov transitional probability matrices respectively.

Dr. Shobha G., year: 2014(ref 10), paper titled "Water Quality Prediction Using Data Mining techniques: A Survey", is published. It was focused on Data mining and its importance of continues growing in business and in learning organization over coming decades. Data mining methods may be classified by the function they perform or by their class of applications. Using this approach, four major categories of processing algorithms and rule approaches emerge: 1) Classification, 2) Association, 3) Sequence and 4) Cluster. This paper explores various data mining techniques like Artificial Neural Network, Back propagation, MLP, GRNN, Decision Tree etc. used in prediction of water quality.

1.5 Motivation and Problem Definition

Water is the most significant resource of life, crucial for supporting the life of most existing creatures and human beings. Living organisms need water with enough quality to continue their lives. There are certain limits of pollutions that water species can tolerate. Exceeding these limits affects the existence of these creatures and threatens their lives. Most ambient water bodies such as rivers, lakes, and streams have specific quality standards that indicate their quality. Moreover, water specifications for other applications/usages possess their standards. For example, irrigation water must be neither too saline nor contain toxic materials that can be transferred to plants or soil and thus destroying the ecosystems. Water quality for industrial uses also requires different properties based on the specific industrial processes.

The rapid industrial development has prompted the decay of water quality at a disturbing rate. Furthermore, infrastructures, with the absence of public awareness, and less hygienic qualities, significantly affect the quality of drinking water [1]. In fact, the consequences of polluted drinking water are so dangerous and can badly affect health, the environment, and infrastructures.

Problem Definition

Considering the motivation, the problem statement is coined as "Water quality prediction using machine learning and comparative study of different machine learning algorithms".

1.6 Objectives Fulfilled

Objectives of the proposed system are,

i. Collect the dataset required to predict water quality and identify the python libraries to be imported.
ii. Implement a method for data cleansing.
iii. Design the machine learning models namely Decision Tree, Random Forest at linear Regression.
iv. Visualization of predicted result.

1.7 Scope and Limitations

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 the development of human societies.

Limitations

Considering 4 to 5 components of water do not lead to proper prediction of water quality through machine learning models. To achieve better performence it is required to consider more number of water components and train the models with huge dataset which further required high configured computing device.

2.0 Methodology

"Water Quality Prediction ", is the work carried out in this project. This chapter helps to understand the methodology behind Prediction of water quality and implementation of the system. Also briefly explains about the hardware component and Software used for prediction.

2.1. The Proposed Methodology

The proposed methodology for the prediction of water quality is depicted in below figure 2.1. At the initial stage user will gather the dataset required to predict the quality of the water. Once

the data has been loaded, system will pre-process the data and extraction of features will be done. After the features required for the prediction have been extracted, system will compare the features with model and prediction will be given as final result in the end. Graphical visualization being made to compare the performance parameters of all Machine Learning algorithms considered in our work.

2.1.1. About Dataset

Access to safe drinking-water is essential to health, a basic human right and a component of effective policy for health protection. This is important as a health and development issue at a national, regional and local level. In some regions, it has been shown that investments in water supply and sanitation can yield a net economic benefit, since the reductions in adverse health effects and health care costs outweigh the costs of undertaking the interventions. The dataset is available on kaggle and data world. And the parameters considered are as follows,

♣ pH value

PH is an important parameter in evaluating the acid–base balance of water. It is also the indicator of acidic or alkaline condition of water status. WHO has recommended maximum permissible limit of pH from 6.5 to 8.5 The current investigation ranges were 6.52–6.83.

♣ Hardness

Hardness is mainly caused by calcium and magnesium salts. These salts are dissolved from geologic deposits through which water travels. The length of time water is in contact with hardness producing material helps determine how much hardness there is in raw water. Hardness was originally defined as the capacity of water to precipitate soap caused by Calcium and Magnesium.

♣ Solids (Total dissolved solids – TDS)

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 un-wanted taste and diluted color in appearance of water. This is the important parameter for the use of water. The water with high TDS value indicates that water is highly mineralized. Desirable limit for TDS is 500 mg/l and maximum limit is 1000 mg/l which prescribed for drinking purpose.

♣ Chloramines

Chlorine and chloramine are the major disinfectants used in public water systems. Chloramines are most commonly formed when ammonia is added to chlorine to treat drinking water. Chlorine levels up to 4 milligrams per liter (mg/L or 4 parts per million (ppm)) are considered safe in drinking water.

♣ Sulfate

Sulfates are naturally occurring substances that are found in minerals, soil, and rocks. They are present in ambient air, groundwater, plants, and food. The principal commercial use of sulfate is in the chemical industry. 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, although much higher concentrations (1000 mg/L) are found in some geographic locations.

♣ Conductivity

Pure water is not a good conductor of electric current rather a good insulator. Increase in ions concentration enhances the electrical conductivity of water. Generally, the dissolved solids in water determines the electrical conductivity. Electrical conductivity (EC) actually measures the

ionic process of a solution that enables it to transmit current. According to WHO standards, EC value should not exceded 400 µS/cm.

♣ Organic carbon

Total Organic Carbon (TOC) in source waters comes from decaying natural organic matter (NOM) as well as synthetic sources. TOC is a measure of the total amount of carbon in organic compounds in pure water. According to US EPA < 2 mg/L as TOC in treated / drinking water, and < 4 mg/Lit in source water which is use for treatment.

♣ Trihalomethanes

THMs are chemicals which may be found in water treated with chlorine. The concentration of THMs in drinking water varies according to the level of organic material in the water, the amount of chlorine required to treat the water, and the temperature of the water that is being treated. THM levels up to 80 ppm is considered safe in drinking water.

♣ Turbidity

The turbidity of water depends on the quantity of solid matter present in the suspended state. It is a measure of light emitting properties of water and the test is used to indicate the quality of waste discharge with respect to colloidal matter. The mean turbidity value obtained for Wondo Genet Campus (0.98 NTU) is lower than the WHO recommended value of 5.00 NTU.

♣ Potability

Indicates if water is safe for human consumption where 1 means Potable and 0 means Not potable (0) Water is not safe to drink and (1) Water is safe to drink.

2.1.2. Data Exploration

The collected dataset is explored to see the distribution of water measurement components and the unwanted data is cleaned.

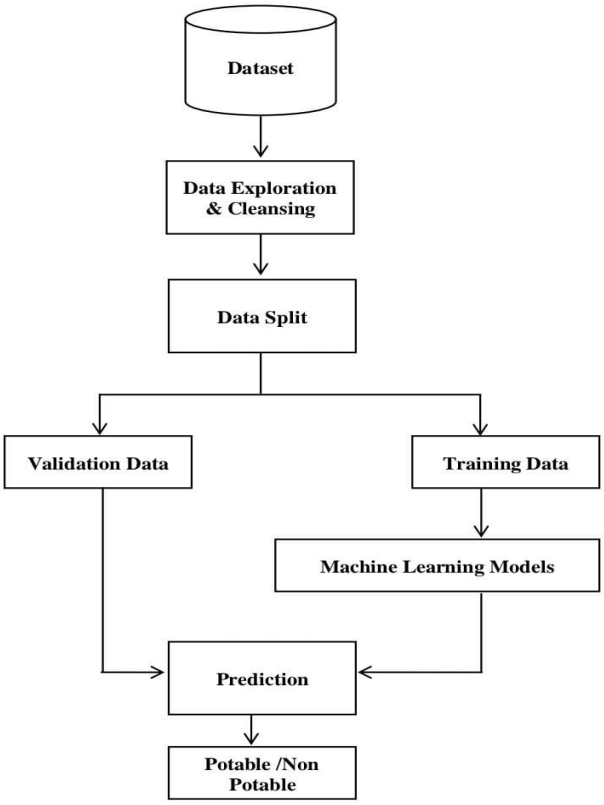


Fig. 2.1 BLOCK DAIGRAM OF THE PROPOSED METHODOLOGY

The first part of any data analysis or predictive modeling task is an initial exploration of the data. Even if you collected the data yourself and you already have a list of questions in mind that you want to answer, it is important to explore the data before doing any serious analysis, since oddities in the data can cause bugs and muddle your results. Before exploring deeper questions, you have to answer many simpler ones about the form and quality of data. That said, it is important to go into your initial data exploration with a big picture question in mind since the goal of your analysis should inform how you prepare the data.

The first step in exploratory analysis is reading in the data and then exploring the variables. It is important to get a sense of how many variables and cases there are, the data types of the variables and the range of values they take on.

The second step, it's a good idea to start off by checking the dimensions of your data set with df.shape and the variable data types of df.dtypes.

In the third step, After getting a sense of the data's structure, it is a good idea to look at a statistical summary of the variables with df.describe():

Although describe gives a concise overview of each variable, it does not necessarily give us enough information to determine what each variable means. Certain variables like "Age" and "Fare" are self-explanatory, while others like "SibSp" and "Parch" are not. Whoever collects or provides data for download should also provide a list of variable descriptions. In this case, Kaggle provides a list of descriptions on the data download page:

After looking at the data for the first time, you should ask yourself a few questions:

- Do I need all of the variables?
- Should I transform any variables?
- Are there NA values, outliers or other strange values?
- Should I create new variables?

- **Do I need all of the variables?**

Getting rid of unnecessary variables is a good first step when dealing with any data set, since dropping variables reduces complexity and can make computation on the data faster. Whether you should get rid of a variable or not will depend on size of the data set and the goal of your analysis. With a data set as small as the Titanic data, there's no real need to drop variables from a computing perspective (we have plenty of memory and processing power to deal with such a small data set) but it can still be helpful to drop variables that will only distract from your goal.

This data set is provided in conjunction with a predictive modeling competition where the goal is to use the training data to predict whether person is affected by Water quality or not.

- **Should I Transform Any Variables?**

When you first load a data set, some of the variables may be encoded as data types that don't fit well with what the data really is or what it means.

- **Are there NA Values, Outliers or Other Strange Values?**

Data sets are often littered with missing data, extreme data points called outliers and other strange values. Missing values, outliers and strange values can negatively affect statistical tests and models and may even cause certain functions to fail.

Detecting missing values is the easy part: it is far more difficult to decide how to handle them. In cases where you have a lot of data and only a few missing values, it might make sense to simply delete records with missing values present. On the other hand, if you have more than a handful of missing values, removing records with missing values could cause you to get rid of a lot of data. Missing values in categorical data are not particularly troubling because you can simply treat NA as an additional category. Missing values in numeric variables are more

troublesome, since you can't just treat a missing value as number. It is not a good idea to throw all those records away.

Here are a few ways we could deal with them:

♣ Replace the null values with 0s.

♣ Replace the null values with some central value like the mean or median.

♣ Impute some other value.

♣ Split the data set into two parts: one set with where records have an Age value and another set where age is null.

• Should I Create New Variables?

The variables present when you load a data set aren't always the most useful variables for analysis. Creating new variables that are derivations or combinations existing ones is a common step to take before jumping into an analysis or modelling task.

For example, imagine you are analyzing web site auctions where one of the data fields is a text description of the item being sold. A raw block of text is difficult to use in any sort of analysis, but you could create new variables from it such as a variable storing the length of the description or variables indicating the presence of certain keywords. Creating a new variable can be as simple as taking one variable and adding, multiplying or dividing by another.

2.1.3. Data Cleaning

Data cleaning is one of the important parts of machine learning. It plays a significant part in building a model. It surely isn't the fanciest part of machine learning and at the same time, there aren't any hidden tricks or secrets to uncover. However, the success or failure of a project relies on proper data cleaning. Professional data scientists usually invest a very large portion of their time in this step because of the belief that "Better data beats fancier algorithms".

If we have a well-cleaned dataset, there are chances that we can get achieve good results with simple algorithms also, which can prove very beneficial at times especially in terms of computation when the dataset size is large. Obviously, different types of data will require different types of cleaning. However, this systematic approach can always serve as a good starting point.

Steps involved in Data Cleaning:

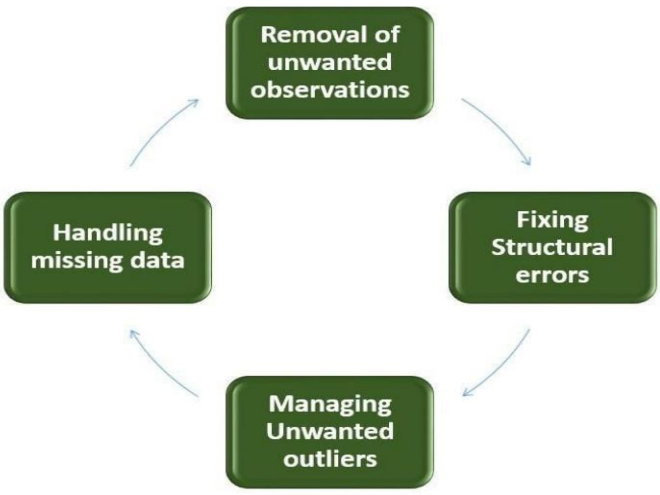


Fig. 2.2 CYCLE OF DATA CLEANING PROCESS

♣ **Removal of unwanted observations**

This includes deleting duplicate/ redundant or irrelevant values from your dataset. Duplicate observations most frequently arise during data collection and irrelevant observations are those that don't actually fit the specific problem that you're trying to solve.

♣ Redundant observations alter the efficiency by a great extent as the data repeats and may add towards the correct side or towards the incorrect side, thereby producing unfaithful results.

♣ Irrelevant observations are any type of data that is of no use to us and can be removed directly.

♣ **Fixing Structural errors**

The errors that arise during measurement, transfer of data, or other similar situations are called structural errors. Structural errors include typos in the name of features, the same attribute with a different name, mislabeled classes, separate classes that should really be the same, or inconsistent capitalization.

For example, the model will treat America and America as different classes or values, though they represent the same value or red, yellow, and red yellow as different classes or attributes, though one class can be included in the other two classes. So, these are some structural errors that make our model inefficient and give poor quality results.

♣ **Managing Unwanted outliers**

Outliers can cause problems with certain types of models. For example, linear regression models are less robust to outliers than decision tree models. Generally, we should not remove outliers until we have a legitimate reason to remove them. Sometimes, removing them improves performance, sometimes not. So, one must have a good reason to remove the outlier, such as suspicious measurements that are unlikely to be part of real data.

♣ **Handling missing data**

Missing data is a deceptively tricky issue in machine learning. We cannot just ignore or remove the missing observation. They must be handled carefully as they can be an indication of something important. The two most common ways to deal with missing data are:

♣ **Dropping observations with missing values.**

The fact that the value was missing may be informative in itself. Plus, in the real world, you often need to make predictions on new data even if some of the features are missing!

♣ **Imputing the missing values from past observations.**

Again, "missingness" is almost always informative in itself, and you should tell your algorithm if a value was missing. Even if you build a model to impute your values, you're not adding any real information. You're just reinforcing the patterns already provided by other features.

Missing data is like missing a puzzle piece. If you drop it, that's like pretending the puzzle slot isn't there. If you impute it, that's like trying to squeeze in a piece from somewhere else in the puzzle.

So, missing data is always an informative and an indication of something important. And we must be aware of our algorithm of missing data by flagging it. By using this technique of flagging and filling, you are essentially allowing the algorithm to estimate the optimal constant for missingness, instead of just filling it in with the mean.

2.1.4. Data Split

The cleansed data is split into two categories in the ratio of 80:20 for training and validation respectively.

2.1.5. Machine learning models

Three machine learning models namely logistic regression, decision tree and XGB are trained to see the performance on collected data and the input data is assessed to predict whether the water is potable or not.

♣ Logistic regression

Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. The nature of target or dependent variable is dichotomous, which means there would be only two possible classes.

In simple words, the dependent variable is binary in nature having data coded as either 1 (stands for success/yes) or 0 (stands for failure/no). Mathematically, a logistic regression model predicts P(Y=1) as a function of X. It is one of the simplest ML algorithms that can be used for various classification problems such as spam detection, Diabetes prediction, cancer detection etc.

♣ Decision Tree

It is a type of supervised learning algorithm that is mostly used for classification problems. Surprisingly, it works for both categorical and continuous dependent variables. In this algorithm, we split the population into two or more homogeneous sets. This is done based on most significant attributes/ independent variables to make as distinct groups as possible.

♣ XG Boost

Another classic gradient boosting algorithm that's known to be the decisive choice between winning and losing in some Kaggle competitions.

The XG Boost has an immensely high predictive power which makes it the best choice for accuracy in events as it possesses both linear model and the tree learning algorithm, making the algorithm almost 10x faster than existing gradient booster techniques.

The support includes various objective functions, including regression, classification and ranking. One of the most interesting things about the XG Boost is that it is also called a regularized boosting technique.

This helps to reduce over fit modeling and has a massive support for a range of languages such as Scala, Java, R, Python, Julia and C++.

Supports distributed and widespread training on many machines that encompass GCE, AWS,0 Azure and Yarn clusters. XG Boost can also be integrated with Spark, Flink and other cloud dataflow systems with a built in cross validation at each iteration of the boosting process.

3.0 System implementation

The chapter 2 tells step by step approach for Water Quality Prediction. This chapter deals with the implementation of Water quality Prediction.

3.1. Data Set

The proposed methodology for the prediction of water quality begins with gathering the required dataset as shown in figure 2.1. The dataset is downloaded from the kaggle repository(https://www.kaggle.com/datasets/adityakadiwal/water-potability), the downloaded file water_potability.csv contains water quality metrics for 3276 different water bodies. Following are further details about the file contents.

TABLE 3.1 WATERMETRICS UNITS AND MEANING

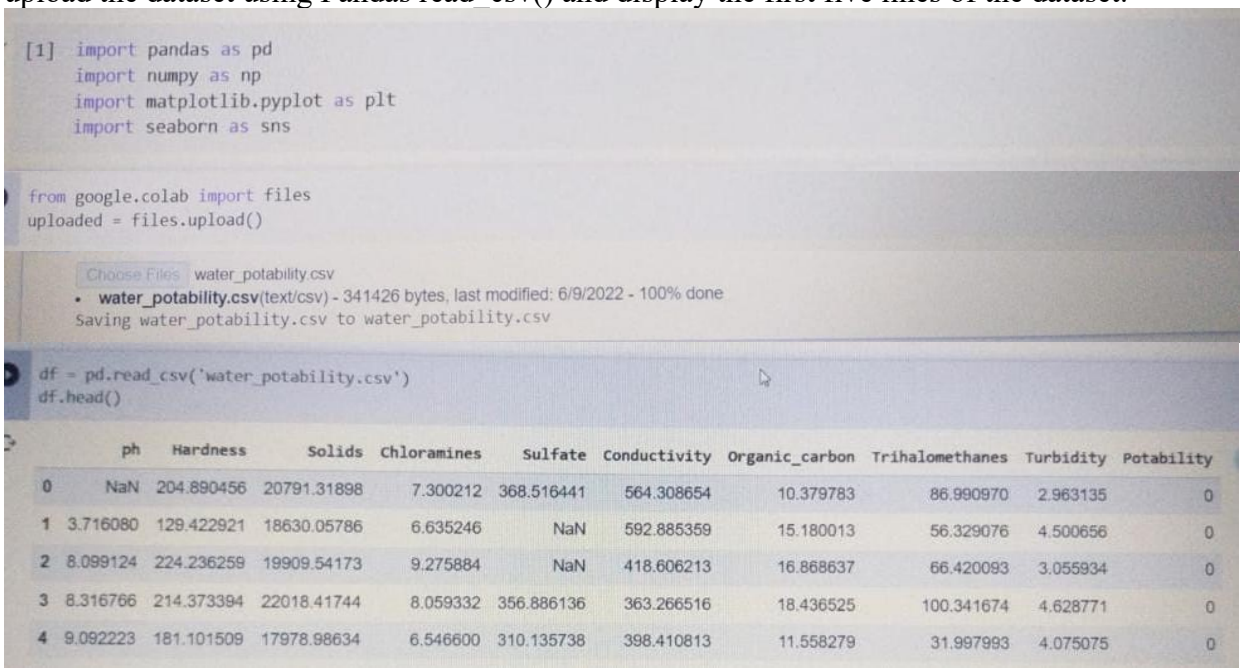
Sl. No.	Unit	Meaning
1.	ppm	parts per million
2.	µg/L	microgram per litre
3.	mg/L	milligram per litre

TABLE 3.2. COLUMN DESCRIPTION

Sl. No.	Column	Description
1.	Ph	pH of 1. water (0 to 14).
2.	Hardness	Capacity of water to precipitate soap in mg/L.
3.	Solids	Total dissolved solids in ppm.
4.	Chloramines	Amount of Chloramines in ppm
5.	Sulfate	Amount of Sulfates dissolved in mg/L.
6.	Conductivity	Electrical conductivity of water in µS/cm.
7.	Organic carbon	Amount of organic carbon in ppm.
8.	Trihalomethanes	Amount of Trihalomethanes in µg/L.
9.	Turbidity	Measure of light emitting property of water in NTU.
10.	Potability:	Indicates if water is safe for human consumption. Potable -1 and Not potable -0

3.2. Data exploration and cleansing

Import all the required libraries which are used to train the model or visualize the data. Then upload the dataset using Pandas read_csv() and display the first five lines of the dataset.



```
[1] import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt
    import seaborn as sns

from google.colab import files
uploaded = files.upload()
```

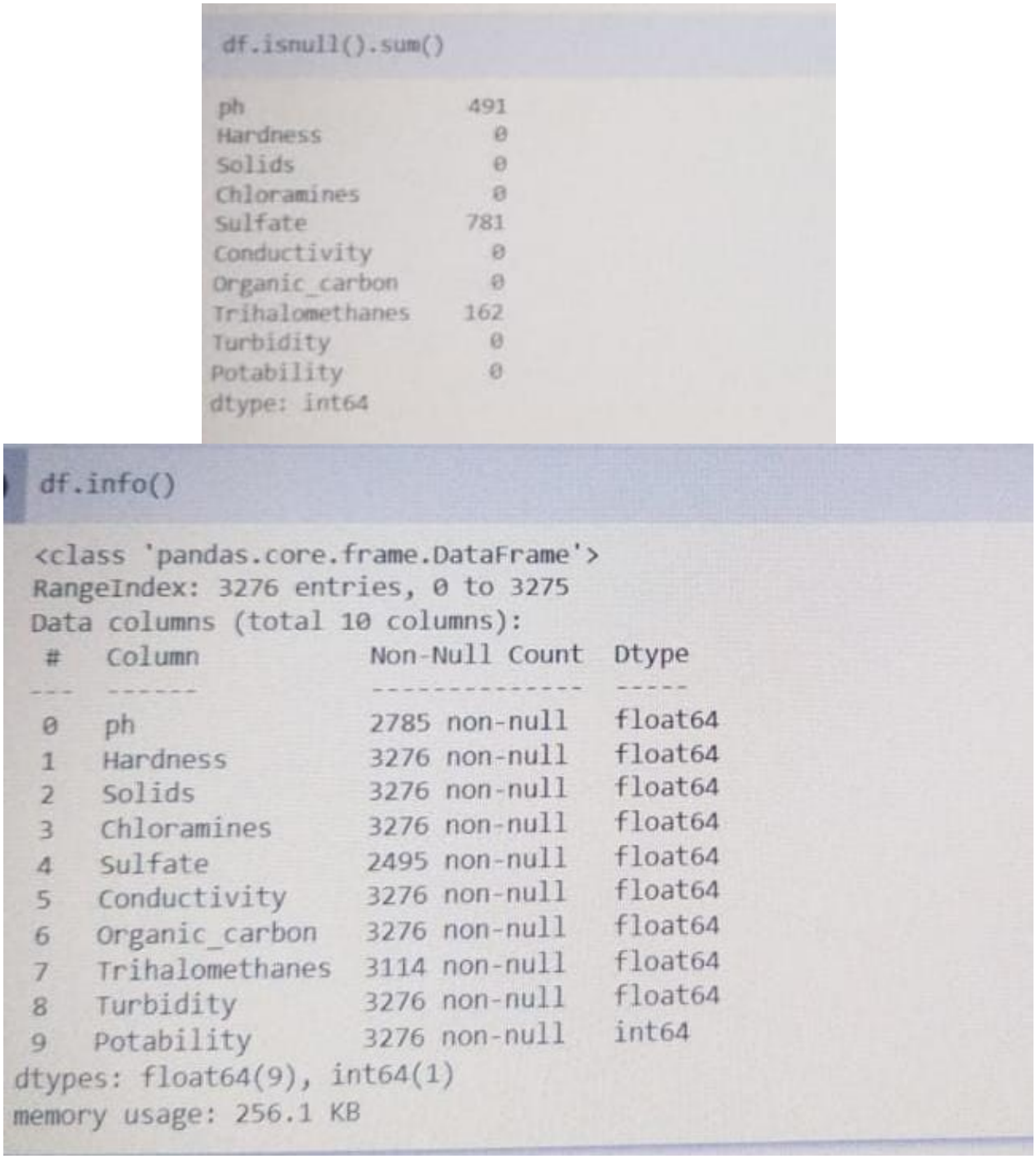
Choose Files water_potability.csv
- water_potability.csv(text/csv) - 341426 bytes, last modified: 6/9/2022 - 100% done

Saving water_potability.csv to water_potability.csv

```
df = pd.read_csv('water_potability.csv')
df.head()
```

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	NaN	204.890456	20791.31898	7.300212	368.516441	564.308654	10.379783	86.990970	2.963135	0
1	3.716080	129.422921	18630.05786	6.635246	NaN	592.885359	15.180013	56.329076	4.500656	0
2	8.099124	224.236259	19909.54173	9.275884	NaN	418.606213	16.868637	66.420093	3.055934	0
3	8.316766	214.373394	22018.41744	8.059332	356.886136	363.266516	18.436525	100.341674	4.628771	0
4	9.092223	181.101509	17978.98634	6.546600	310.135738	398.410813	11.558279	31.997993	4.075075	0

Then perform Exploratory Data Analysis. In EDA, First check the shape of the data set. Then check that there are Null values or not and you can see in the below image that pH, Sulfate, Trihalomethanes contain NULL values. Then check the information of the data set.

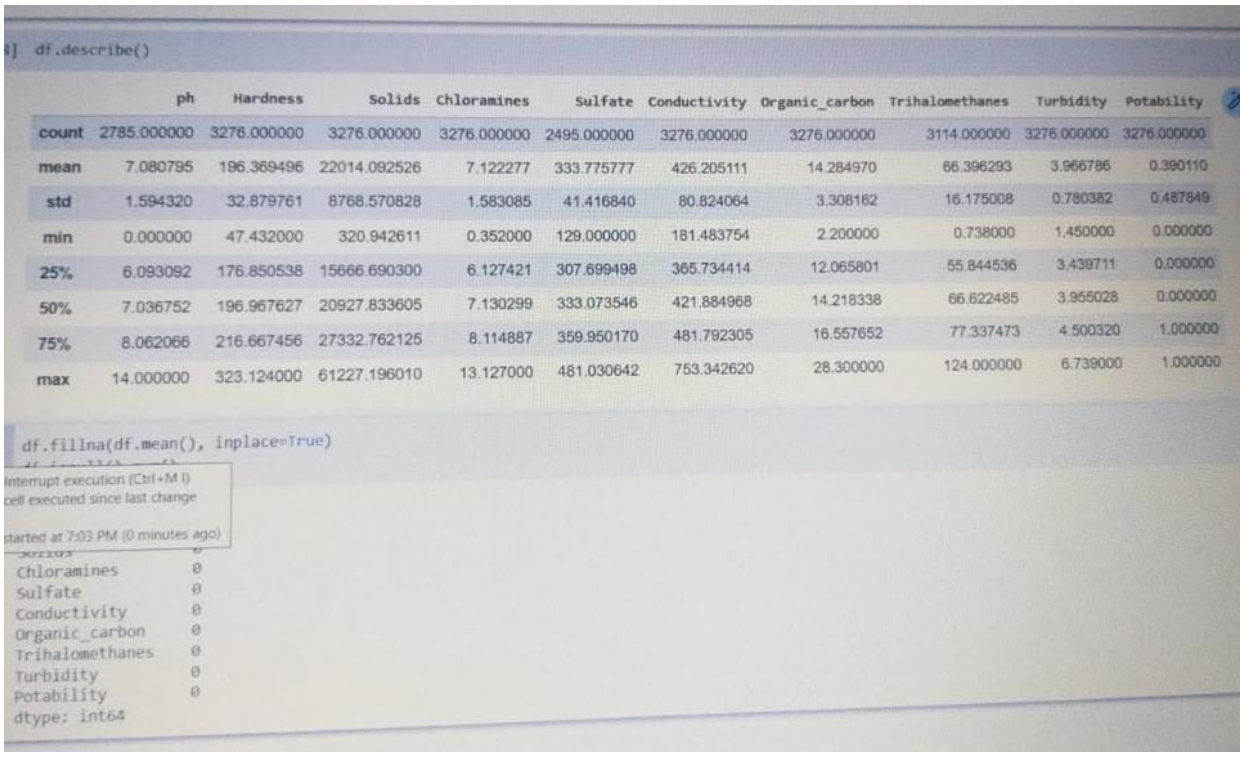


```
df.isnull().sum()

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

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3276 entries, 0 to 3275
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   ph               2785 non-null   float64
 1   Hardness         3276 non-null   float64
 2   Solids           3276 non-null   float64
 3   Chloramines      3276 non-null   float64
 4   Sulfate          2495 non-null   float64
 5   Conductivity     3276 non-null   float64
 6   Organic_carbon   3276 non-null   float64
 7   Trihalomethanes  3114 non-null   float64
 8   Turbidity        3276 non-null   float64
 9   Potability       3276 non-null   int64
dtypes: float64(9), int64(1)
memory usage: 256.1 KB
```

Fig. 3.3 DATASET INFORMATION

Now describe the dataset which shows the minimum value, maximum value, mean value, count, standard deviation, etc. Then finally we handle the missing values. We filled the missing values in our feature using a mean value of each feature which means we filled the mean value to handle missing data. Then again check that there are null values present or not.



```
df.describe()
```

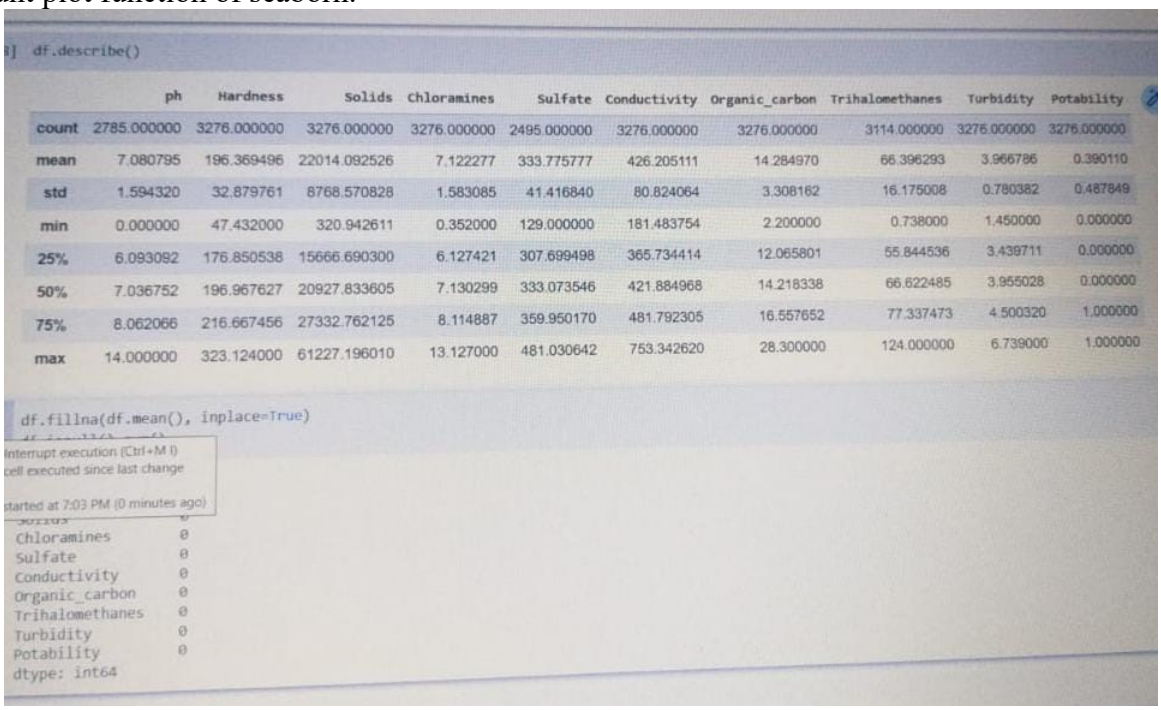
	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
count	2785.000000	3276.000000	3276.000000	3276.000000	2495.000000	3276.000000	3276.000000	3114.000000	3276.000000	3276.000000
mean	7.080795	196.369496	22014.092526	7.122277	333.775777	426.205111	14.284970	66.396293	3.966786	0.390110
std	1.594320	32.879761	8768.570828	1.583085	41.416840	80.824064	3.308162	16.175008	0.780382	0.487849
min	0.000000	47.432000	320.942611	0.352000	129.000000	181.483754	2.200000	0.738000	1.450000	0.000000
25%	6.093092	176.850538	15666.690300	6.127421	307.699498	365.734414	12.065801	55.844536	3.439711	0.000000
50%	7.036752	196.967627	20927.833605	7.130299	333.073546	421.884968	14.218338	66.622485	3.955028	0.000000
75%	8.062066	216.667456	27332.762125	8.114887	359.950170	481.792305	16.557652	77.337473	4.500320	1.000000
max	14.000000	323.124000	61227.196010	13.127000	481.030642	753.342620	28.300000	124.000000	6.739000	1.000000

```
df.fillna(df.mean(), inplace=True)
```

```
Interrupt execution (Ctrl+M I)
cell executed since last change

started at 7:03 PM (0 minutes ago)

Chloramines      0
Sulfate          0
Conductivity     0
Organic_carbon   0
Trihalomethanes  0
Turbidity        0
Potability       0
dtype: int64
```

Fig. 3.4 FILLING OF MISSING/NULL VALUE

Check the value counts of our target feature potability. Then visualize the portability using a count plot function of seaborn.



```
df.describe()
```

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
count	2785.000000	3276.000000	3276.000000	3276.000000	2495.000000	3276.000000	3276.000000	3114.000000	3276.000000	3276.000000
mean	7.080795	196.369496	22014.092526	7.122277	333.775777	426.205111	14.284970	66.396293	3.966786	0.390110
std	1.594320	32.879761	8768.570828	1.583085	41.416840	80.824064	3.308162	16.175008	0.780382	0.487849
min	0.000000	47.432000	320.942611	0.352000	129.000000	181.483754	2.200000	0.738000	1.450000	0.000000
25%	6.093092	176.850538	15666.690300	6.127421	307.699498	365.734414	12.065801	55.844536	3.439711	0.000000
50%	7.036752	196.967627	20927.833605	7.130299	333.073546	421.884968	14.218338	66.622485	3.955028	0.000000
75%	8.062066	216.667456	27332.762125	8.114887	359.950170	481.792305	16.557652	77.337473	4.500320	1.000000
max	14.000000	323.124000	61227.196010	13.127000	481.030642	753.342620	28.300000	124.000000	6.739000	1.000000

```
df.fillna(df.mean(), inplace=True)
```

```
Interrupt execution (Ctrl+M I)
cell executed since last change

started at 7:03 PM (0 minutes ago)

Chloramines      0
Sulfate          0
Conductivity     0
Organic_carbon   0
Trihalomethanes  0
Turbidity        0
Potability       0
dtype: int64
```

Fig. 3.5 POTABILITY COUNT

Now visualize the pH value using a dist plot function to check that it contains a normal distribution or not. So, you can see that it is a normal distribution.

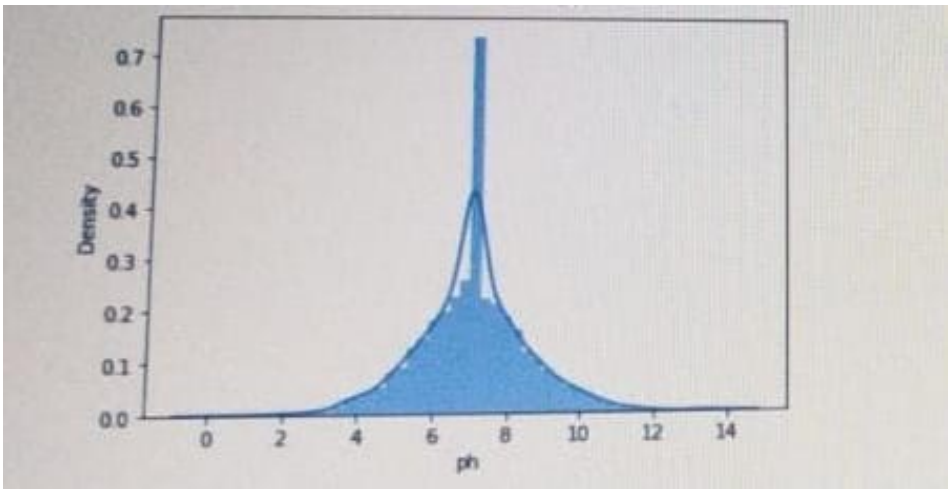


Fig. 3.6 PH VALUE DISTRIBUTION USING DISPLOT FUNTION

Visualize all the features of the data set as shown in figure 3.7.

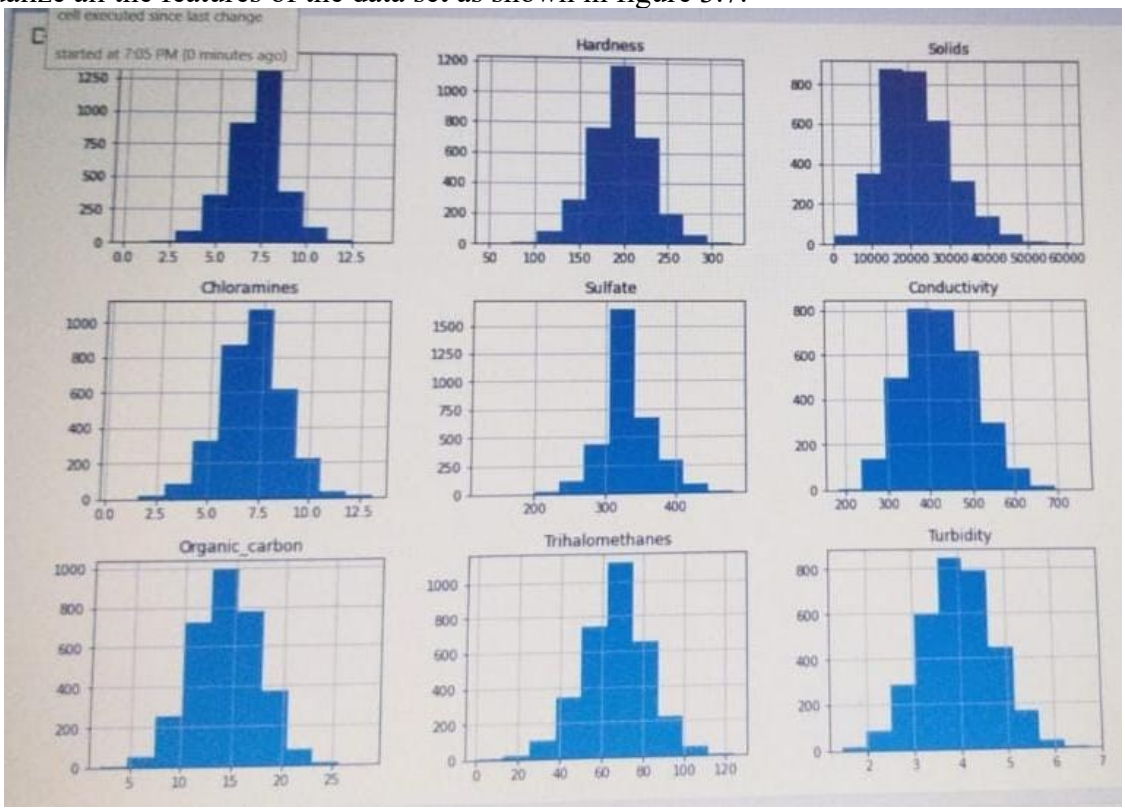
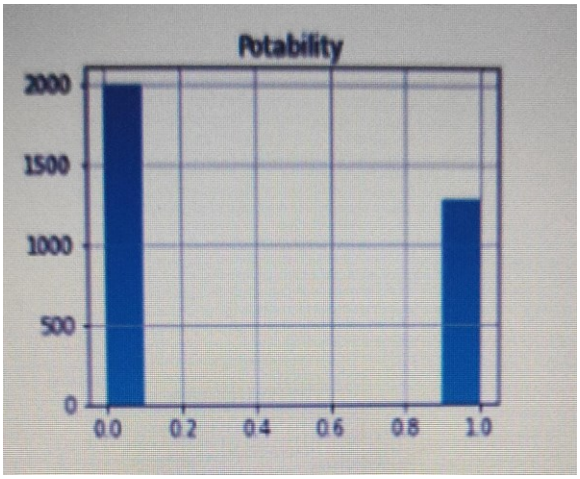


Fig. 3.7 VISUALIZATION OF WATER METRICS



Visualize the correlation of all the features using a heat map function of seaborn. but you can see in the heat map as shown in figure 3.8, that there is no correlation between any feature; it means that we can't reduce the dimension.

Now see the outlier using a box plot function as in figure 3.9. So, you can see that the Solid feature contains outliers but we can't remove the outliers from it because if we remove the outliers from the Solid feature. So, water will be safe to drink every time.

It contains an outlier to make the water impure which means it will tell us that water is safe or not. Solid may be high to make the water unsafe to drink.

Fig. 3.8 HEATMAP FOR WATER METRICS

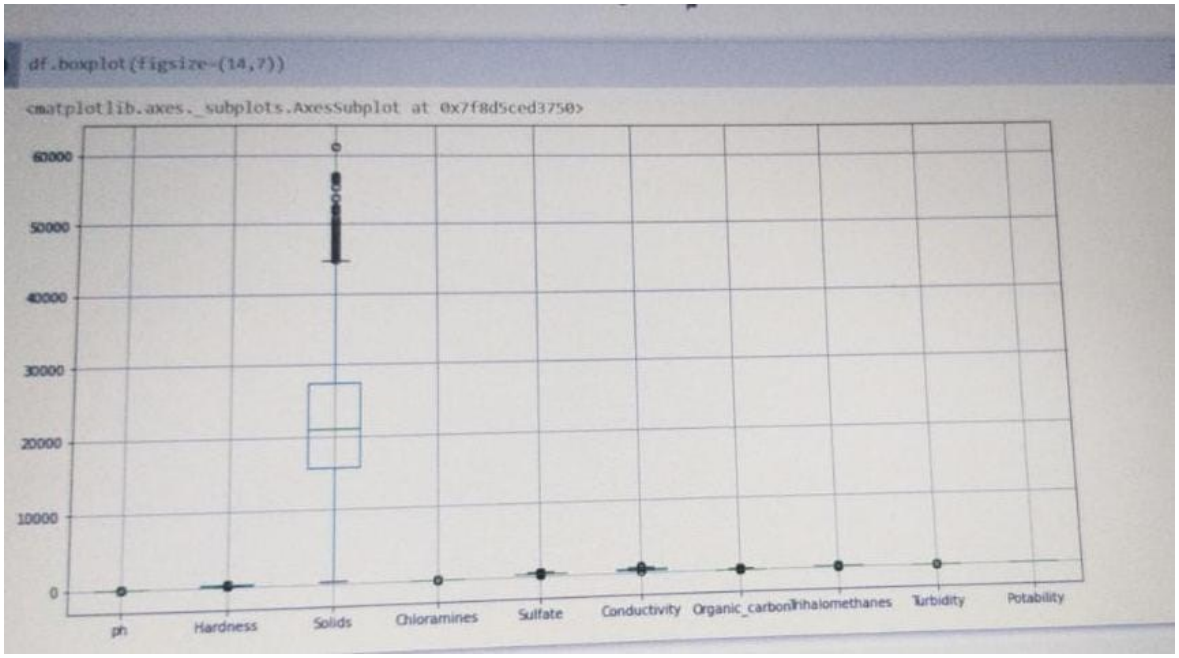


Fig. 3.9 OUTLIER

3.3. Data Split

Now it's time to prepare the data set. Divide the data into the independent and dependent features.

All are independent features except Potability because Potability is our dependent feature. Split the data set into the training and testing using the train test split function which returns four data sets.

```
X = df .drop ('Portability', axis=1)
Y= df['Potability']
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split (X,Y, test_size = 0.2, random_state=101, shuffle=True)
```

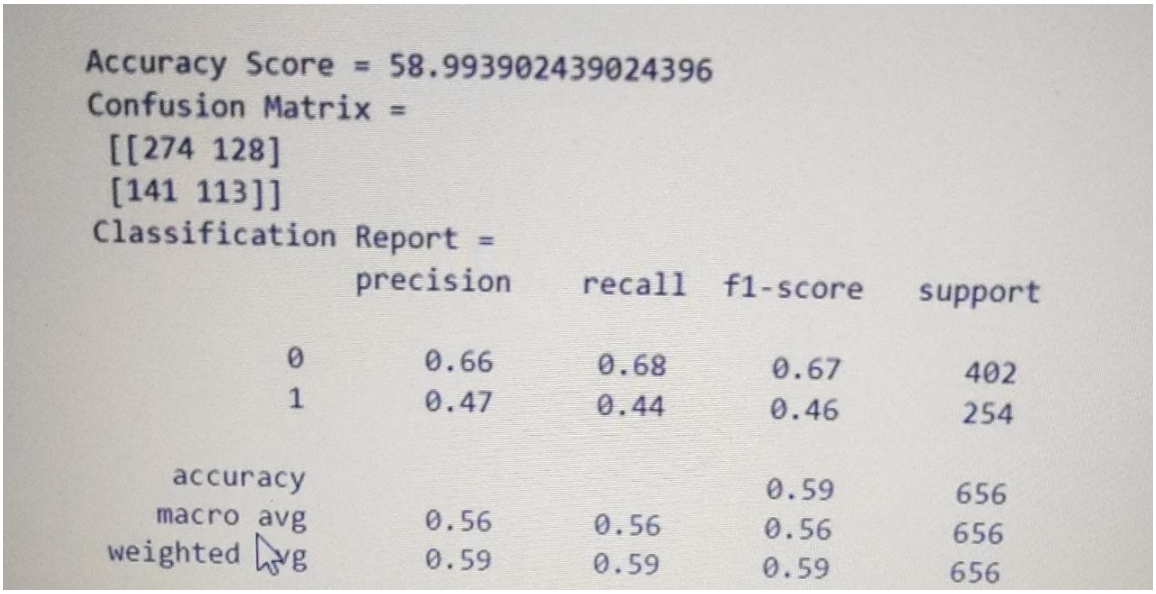
3.4. Machine Learning models

In this section different machine learning models are trained and compared the performance of each model.

3.4.1. Decision Tree Classifier

Now define the decision tree classifier model and train the model using the data set (X_train, Y_train). Then test the model using the test data set (X_test).

Now it's time to evaluate the model using the accuracy score, confusion matrix and classification report. Evaluation techniques take two parameters; one is the actual data and the other one is a predicted data. And You can see that overall accuracy is 59%.



```
Accuracy Score = 58.993902439024396
Confusion Matrix =
 [[274 128]
 [141 113]]
Classification Report =
              precision    recall  f1-score   support

           0       0.66      0.68      0.67       402
           1       0.47      0.44      0.46       254

    accuracy                           0.59       656
   macro avg       0.56      0.56      0.56       656
weighted avg       0.59      0.59      0.59       656
```

Fig 3.10 OUTPUT OF DECISION TREE CLASSIFIER

3.4.2. Random Forest Classifier

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is controlled with the max samples parameter if bootstrap=True (default), otherwise the whole dataset is used to build each tree.

The default values for the parameters controlling the size of the trees (e.g max_depth, min_samples_leaf, etc.) lead to fully grown and un pruned trees which can potentially be very large on some data sets. To reduce memory consumption, the complexity and size of the trees should be controlled by setting those parameter values.

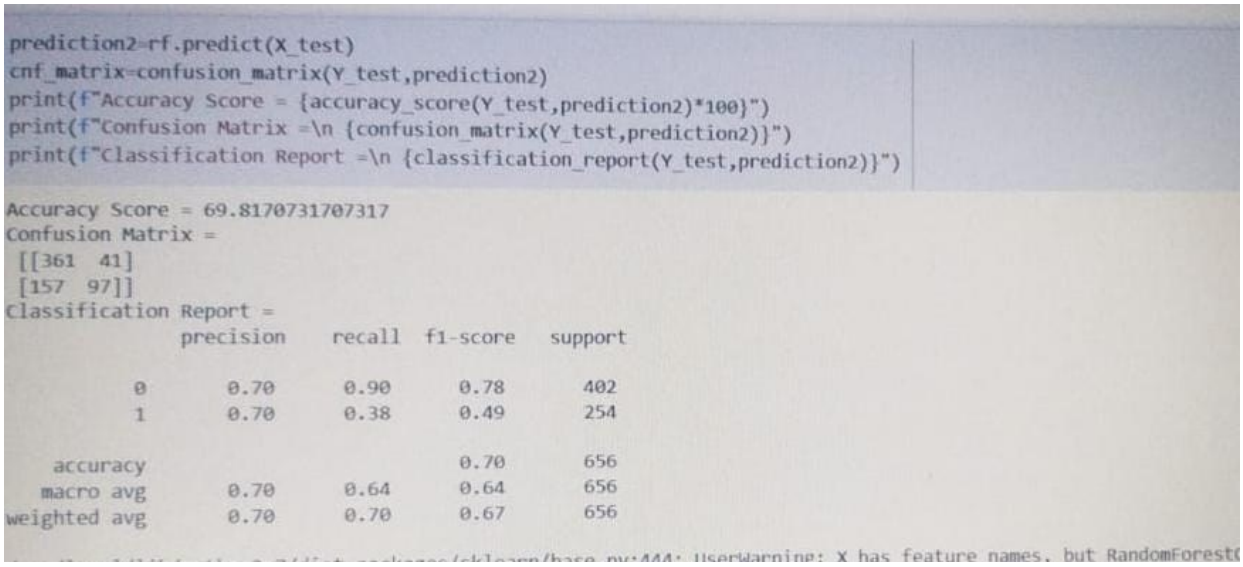
The features are always randomly permuted at each split. Therefore, the be split may vary, even with the same training data, max_features=n_features and bootstrap=False, if the improvement of the criterion is identical for several splits enumerated during the search of the best split. To obtain a deterministic behaviour during fitting, random state has to be fixed.

Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. Below are two assumptions for a better Random Forest classifier:

- There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result.
- The predictions from each tree must have very low correlations.

Now define the random forest classifier model and train the model using the data set (X_train, Y_train). Then test the model using the test data set (X_test).

Now it's time to evaluate the model using the accuracy score, confusion matrix and classification report. Evaluation techniques take two parameters; one is the actual data and the other one is a predicted data. And can see that overall accuracy is 69%.



```
prediction2=rf.predict(X_test)
cnf_matrix=confusion_matrix(Y_test,prediction2)
print(f"Accuracy Score = {accuracy_score(Y_test,prediction2)*100}")
print(f"Confusion Matrix =\n {confusion_matrix(Y_test,prediction2)}")
print(f"Classification Report =\n {classification_report(Y_test,prediction2)}")
```

```
Accuracy Score = 69.8170731707317
Confusion Matrix =
 [[361  41]
 [157  97]]
Classification Report =
              precision    recall  f1-score   support

           0       0.70      0.90      0.78       402
           1       0.70      0.38      0.49       254

    accuracy                           0.70       656
   macro avg       0.70      0.64      0.64       656
weighted avg       0.70      0.70      0.67       656
```

Fig. 3.11 OUTPUT OF RANDOM FOREST CLASSIFIER

3.4.3. Logistic regression

Logistic regression is one of the most popular Machine Learning Algorithm that comes under Supervised Learning techniques. It can be used for Classification as well as for Regression problems, but mainly used for Classification problems.

Logistic regression is used to predict the categorical dependent variable with the help of independent variables. The output of Logistic Regression problem can be only between the 0 and 1.

Logistic regression can be used where the probabilities between two classes is required. Such as whether it will rain today or not, either 0 or 1, true or false etc. Logistic regression is based on the concept of Maximum Likelihood estimation. According to this estimation, the observed data should be most probable. In logistic regression, we pass the weighted sum of inputs through an activation function that can map values in between 0 and 1. Such activation function is known as sigmoid function and the curve obtained is called as sigmoid curve or S-curve.

Now define the logistic regression model and train the model using the data set (X_train, Y_train). Then test the model using the test data set (X_test).

Now it's time to evaluate the model using the accuracy score, confusion matrix and classification report. Evaluation techniques take two parameters; one is the actual data and the other one is a predicted data. And can see that overall accuracy is 61%.

3.4.4. K-Nearest Neighbour

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.

K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset. KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.
Now define the K-Nearest Neighbour model and train the model using the data set(X_train, Y_train).Then test the model using the test data set (X_test).
Now it's time to evaluate the model using the accuracy score, confusion matrix and classification report. Evaluation techniques take two parameters; one is the actual data and the other one is a predicted data. And can see that overall accuracy is 59%.

4.0 Result and discussion

In this section, prior to discussing the results, we will describe different measures used to assess the accuracy of the applied machine learning algorithms.

4.1. Accuracy Measures

R Squared Error (RSE): R squared error (RSE), also known as the coefficient of determination, and often denoted as, determines the goodness of fit of the model. It particularly explains the amount of variance of the dependent variable that is explainable through the independent variable, as shown in Equation (8). Higher RSE values mean that the independent variables largely explain the variance of the dependent variable [34].

$$\text{RSE or } R^2 = 1 - \frac{\text{Explained variation}}{\text{Total variation}}$$

For classification, we used the following measures:

- **Accuracy**

Accuracy is the correct number of predictions made by the model over all the observed values. Accuracy is measured by Equation (9), where TP refers to true positive, TN refers to true negative, FP refers to false positive and FN refers to false negative [11,13].

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

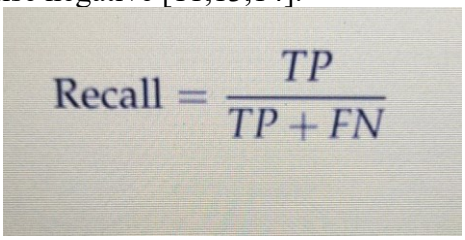
- **Precision:**

Precision is the proportion of correctly classified instances of a particular positive class out of the total classified instances of that class. Precision is calculated with the formula shown in Equation (10), where TP refers to true positive and FP refers to false positive [7,35,36].

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

• **Recall**

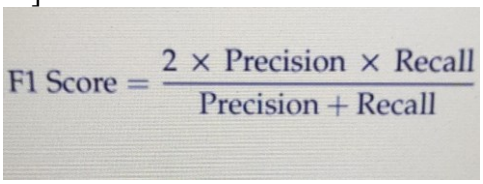
Recall is the proportion of instances of a particular positive class that were actually classified correctly. Recall is calculated with the formula shown in Equation, where TP refers to true positive and FN refers to false negative [11,13,14].



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

• **F1Score**

As precision and recall, individually, do not cover all aspects of the accuracy, we took their harmonic mean to reflect the F1 score, as shown in Equation, which covers both aspects and reflects the overall accuracy measure better. It ranges between 0 and 1. The higher the score, the better the accuracy. [11,13,14].



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

4.2. Correlation Analysis

To find the dependent variables and to predict hard-to-estimate variables through easily attainable parameters, we performed correlation analysis to extract the possible relationships between the parameters. We used the most commonly used and effective correlation method, known as the Pearson correlation. We applied the Pearson correlation on the raw values of the parameters listed in Table 4.1 and applied it after normalizing the values through q-value normalization as explained in the subsequent section. As the correlation chart in Table 4 indicates:

• Alkalinity (Alk) is highly correlated with hardness (CaCO3) and calcium (Ca).
• Hardness is highly correlated with alkalinity and calcium, and loosely correlated with pH.
• Conductance is highly correlated with total dissolved solids, chlorides and fecal coliform count, and loosely correlated with calcium and temperature.
• Calcium is highly correlated with alkalinity and hardness, while loosely correlated with TDS, chlorides, conductance and pH.
• TDS is highly correlated with conductance, chlorides and fecal coliform, and loosely correlated with calcium and temperature.
• Chlorides are highly correlated with conductance and TDS, and loosely correlated with temperature, calcium and fecal coliform.
• Fecal coliform is correlated with conductance and TDS, and loosely correlated with chlorides.

Now that we have listed the correlation analysis observations, we find that our predicting parameter WQI is correlated with seven parameters, namely temperature, turbidity, pH, hardness as CaCO3, conductance, total dissolved solids and fecal coliform count. We have to choose the minimal number of parameters to predict the WQI, in order to lower the cost of the system. The three parameters whose sensors are easily available, cost the lowest and contribute distinctly to the WQI are temperature, turbidity and pH, which deems them naturally selected. The other convenient parameter is total dissolved solids, whose sensor is also easily available

and is correlated with conductance and fecal coliform count, which means selecting TDS would allow us to discard the other two parameters. We leave the remaining inconvenient parameter, hardness as CaCO3, out because it is not highly correlated comparatively and is not easy to acquire.

To conclude the correlation analysis, we selected four parameters for the prediction of WQI, namely, temperature, turbidity, pH and total dissolved solids. We initially just considered the first three parameters, given their low cost, and if needed, TDS will be included later to analyze its contribution to the accuracy.

TABLE 4.1 CORRELATION ANALYSIS OF WATER PARAMETER

	Temp	Turb	pH	Alk	$CaCO_3$	Cond	Ca	TDS	Cl	NO_2	FC	WQI
Temp	1.000	0.103	0.005	−0.193	−0.288	0.266	−0.150	0.274	0.293	−0.154	0.194	**−0.467**
Turb	0.103	1.000	−0.0886	−0.093	−0.146	0.048	−0.122	0.042	0.037	0.0002	0.037	**−0.354**
pH	0.005	−0.088	1.000	−0.177	−0.278	−0.065	−0.236	−0.060	−0.149	0.167	0.054	**−0.431**
Alk	−0.193	−0.092	−0.177	1.000	0.462	0.011	0.444	0.012	0.061	0.046	0.013	0.223
$CaCO_3$	−0.288	−0.146	−0.278	0.462	1.000	0.068	0.637	0.060	0.135	0.078	0.016	0.360
Cond	0.266	0.048	−0.064	0.011	0.068	1.000	0.225	0.973	0.780	0.100	0.456	−0.370
Ca	−0.150	−0.122	−0.236	0.444	0.637	0.225	1.000	0.219	0.262	0.124	0.113	0.188
TDS	0.273	0.041	−0.060	0.012	0.060	0.974	0.219	1.000	0.765	0.095	0.454	**−0.381**
Cl	0.292	0.037	−0.149	0.061	0.135	0.780	0.262	0.765	1.000	0.036	0.353	−0.274
NO_2	−0.154	0.0002	0.167	0.046	0.078	0.100	0.124	0.095	0.036	1.000	0.193	−0.209
FC	0.194	0.037	0.053	0.012	0.016	0.456	0.113	0.454	0.353	0.193	1.000	−0.421
WQI	−0.467	−0.354	−0.431	0.223	0.360	−0.370	0.188	−0.381	−0.274	−0.209	−0.421	1.000

4.3. Results and Discussion

For validating the developed model, the dataset has been divided into 80% training and 20% testing subsets. The table 4.2. Shows the performance evaluation for different machine learning algorithms viz. decision tree, random forest, logistic regression and K-nearest neighbor classifier, which were experimented on 9 water quality metrics.

TABLE 4.2. RESULT OF DIFFERENT MACHINE LEARNING ALGORITHMS

Metric	DT	RF	LR	KNN
Accuracy	0.589939	0.699878	0.612805	0.596037
F1-Score	0.454361	0.472362	0.000000	0.204204
Recall	0.440945	0.370079	0.000000	0.133858
Precision	0.468619	0.652778	0.000000	0.430380
R2-Score	-0.728209	-0.349160	-0.631841	-0.702511

From the results it is observed that among all random forest machine learning algorithm yields good accuracy. Since the data set which is available has limited number of records resulting to remarkable accuracy. As we increase the size of the dataset positively we can get good result.

4.4. Test Cases

Testing forms an integral part of any software development project. Testing helps in ensuring that the final product is by and large, free of defects and it meets the desired requirements. Proper testing in the development phase helps in identifying the critical errors in the design and implementation of various functionalities thereby ensuring product reliability. Even though it is a bit time-consuming and a costly process at first, it helps in the long run of software development.

Although machine learning systems are not traditional software systems, not testing them properly for their intended purposes can lead to a huge impact in the real world. This is because machine learning systems reflect the biases of the real world. Not accounting or testing for them will inevitably have lasting and sometimes irreversible impacts.

In traditional software systems, code is written for having a desired behavior as the outcome. Testing them involves testing the logic behind the actual behavior and how it compares with the expected behavior. In machine learning systems, however, data and desired behavior are the inputs and the models learn the logic as the outcome of the training and optimization processes. In this case, testing involves validating the consistency of the model's logic and our desired behavior.

Due to the process of models learning the logic, there are some notable obstacles in the way of testing Machine Learning systems. They are:

• Indeterminate outcomes: on retraining, it's highly possible that the model parameters vary significantly.

• Generalization: it's a huge task for Machine Learning models to predict sensible outcomes for data not encountered in their training.

• Coverage: there is no set method of determining test coverage for a Machine Learning model.

• Interpretability: most ML models are black boxes and don't have a comprehensible logic for a certain decision made during prediction.

These issues lead to a lower understanding of the scenarios in which models fail and the reason for that behavior; not to mention, making it more difficult for developers to improve their behaviors.

4.4.1. Uploading of dataset

TABLE 4.3. GIVES TEST CASE IN UPLOADING DATASET

Sl. No	Design step	Expected result	Actual result
1	Option to browse the file to be uploaded.	Opens file browser	File browser is opened successfully
2	Select the file and submit	file selection & submit	as expected
3	After submission file to be loaded into colab workspace	load file successfully into workspace	as expected

4.4.2. Data preprocessing

TABLE 4.4. SHOWS TEST CASE FOR DATA PREPROCESSING

Sl. No	Design step	Expected result	Actual result
1	find the null values in uploaded file	show count of null values	as expected
2	find the duplicate values in uploaded file	show count of null values	as expected
3	find drop redundant data in the uploaded file	removes redundant column	as expected
4	fill null values by statistical values	Null values are filled by selected statistical value	as expected

5.0 Conclusion and Future work

Water is one of the most essential resources for survival and its quality is determined through WQI. Conventionally, to test water quality, one has to go through expensive and cumbersome lab analysis. This project explored an alternative method of machine learning to predict water quality using minimal and easily available water quality parameters. The data used to conduct the study were acquired from kaggle and contained 3276 samples. A set of representative supervised machine learning algorithms were employed to estimate water quality. This showed that random forest outperformed other algorithms by predicting water quality most efficiently.

In future works, we propose integrating the findings of this project in a large-scale IoT-based online monitoring system using only the sensors of the required parameters. The tested algorithms would predict the water quality immediately based on the real-time data fed from the IoT system It would identify poor quality water before it is released for consumption and alert concerned authorities. It will hopefully result in curtailment of people consuming poor quality water and consequently deescalate harrowing diseases like typhoid and diarrhea. In this regard, the application of a prescriptive analysis from the expected values would lead to future facilities to support decision and policy makers.

References:

[a] Saber Kouadri, Ahmed Elbeltagi, Abu Reza Md., Towfqul Islam, Samir Kateb, "Performance of machine learning methods in predicting water quality index based on irregular data set: application on Illizi region (Algerian southeast)", Applied Water Science (2021) 11:190 https://doi.org/10.1007/s13201-021-01528-9.

[b] Iran Atefeh Nouraki1 & Mohammad Alavi1 & Mona Golabi1 & Mohammad Albaji, "Prediction of water quality parameters using machine learning models: a case study of the Karun River", Environmental Science and Pollution Research https://doi.org/10.1007/s11356-021-14560-8., 2001.

[c] Luka Grbcic, Goran Mausa, "Coastal Water Quality Prediction Based on Machine Learning with Feature Interpretation and Spatio–temporal Analysis", arXiv:2107.03230v2 [stat.AP] 9 Jul 2021.

[d] Md. Saikat Islam Khan, Nazrul Islam, Jia Uddin c, Sifatul Islam, Mostofa Kamal Nasir, "Water quality prediction and classification based on principal component regression and gradient boosting classifier approach", Journal of King Saud University – Computer and Information Sciences, 2021, https://doi.org/10.1016/j.jksuci.2021.06.003.

[e] Umair Ahmed, RafiaMumtaz, Hirra Anwar, Asad A. Shah, Rabia Irfan and José García-Nieto, "Efficient Water Quality Prediction Using Supervised Machin learning", Water 2019, 11, 2210; doi:10.3390/w11112210 www.mdpi.com/journal/ Water.

[f] Ali NajahAhmeda, FaridahBinti Othman, "Machine learning methods for better water quality prediction ", Journal of Hydrology journal homepage: www.elsevier.com/locate/jhydrol, https://doi.org/10.1016/j.jhydrol.2019.124084.

[g] AmerHamzehHaghiabi, Ali HeiderNasrolahi and Abbas Parsaie, "Water quality prediction using machine learning methods", Water Quality Research Journal | 2018.

[h] Xiu Li, Jingdong Song, "A New ANN-Markov Chain Methodology for Water Quality Prediction", National Natural Science Foundation of China (NSFC, Project No.:71171121161033005) and National "863"High Technology Research and Development Program of China. (863 Project No.: 2012AA09A408). 978-1-4799-1959-8/15/$31.00 @2015 IEEE.

[i] Shoba G, Dr. Shobha G, "Water Quality Prediction Using Data Mining techniques: A Survey", IJECS Volume 3. Issue 6 June, 2014 Page No.6299- 6306.
[j] Shafi, U.; Mumtaz, R.; Anwar, H.; Qamar, A.M.; Khurshid, H. Surface Water Pollution Detection using Internet of Things. In Proceedings of the 2018 15th International Conference on Smart Cities: Improving Quality of Life Using ICT &IoT (HONET-ICT), Islamabad, Pakistan, 8–10 October 2018; pp. 92– 96.
[k] Menard, S. Coefficients of determination for multiple logistic regression analysis. Am. Stat. 2000, 54, 17–24.
[l] Goutte, C.; Gaussier, E. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In Proceedings of the European Conference on Information Retrieval, Santiago de Compostela, Spain, 21–23 March 2005; pp. 345–359.