iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VI **Month of publication:** June 2022

DOI: https://doi.org/10.22214/ijraset.2022.44658

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

Water Quality Prediction using Machine Learning

Nishant Rawat[1], Mangani Daudi Kazembe[2], Pradeep Kumar Mishra[3]
[1, 2, 3]*Department of Computer Science and Engineering, Sharda University, Greater Noida*

Abstract: ***Freshwater is a critical resource for agriculture and industry's survival. Examination of water quality is a fundamental stage in the administration of freshwater assets. As indicated by the World Health Organization's yearly report, many individuals are getting sick or some are dead due to the lack of safe drinking water, especially pregnant ladies and kids. It is critical to test the quality of water prior to involving it for any reason, whether it is for animal watering, chemical spraying(Pesticides etc), or drinking water. Water quality testing is a strategy for finding clean drinking water. Accordingly, appropriate water monitoring is basic for safe, clean, and sterile water. Water testing is fundamental for looking at the legitimate working of water sources, testing the safety of drinking water, identifying disease outbreaks, and approving methodology and safeguard activities. Water quality is a proportion of a water's readiness for a specific utilize in view of physical, chemical, and biological qualities.***
Keywords: Machine learning; water monitoring; disease detection.

I. INTRODUCTION

Water is the principal source for shipping energy to each cell in the body and is additionally the regulator of all body capacities. The cerebrum contains 80% of water. Extreme drying out may prompt mental hindrances and loss of capacity to obviously think. Water is one of the most fundamental regular assets for the endurance of the whole life on this planet. In light of the nature of water, it tends to be utilized for various purposes like drinking, washing, or water system. Plants and creatures likewise rely upon water for their endurance. To put it plainly, all living organic entities need an enormous amount and great nature of water for presence. Freshwater is a fundamental asset to horticulture and industry for its essential presence. Water quality observation is a key stage in the administration of freshwater assets. As indicated by the yearly report of WHO, many individuals are kicking the bucket because of the absence of unadulterated drinking water particularly pregnant ladies and youngsters. It is critical to check the nature of water for its expected reason, whether it be animals watering, compound showering, or drinking water.

Water quality testing is a device that can be utilized to find unadulterated drinking water. Consequently, the right checking of water is incredibly much significant for protecting unadulterated, and clean water. Water testing assumes a key part in breaking down the right activity of water supplies, testing the wellbeing of drinking water, perceiving sickness flare-ups, and approving cycles and precaution measures. Water quality is the proportion of the reasonableness of water for a specific reason in view of explicit physical, substance, and organic attributes.

Testing the nature of a water body, both surface water, and groundwater, can assist us with responding to inquiries concerning whether the water is satisfactory for drinking, washing, or water system to give some examples of applications. It can utilize the consequences of water quality tests to look at the nature of water starting with one water body and then onto the next in a locale, state, or across the entire country. Microbiological quality is for the most part the main pressing concern on the grounds that irresistible infections brought about by pathogenic microorganisms, infections, helminths, and so on are the most well-known and boundless wellbeing risk connected with drinking water. Overabundance amount of certain synthetic substances in drinking water prompts well-being risk. These synthetics incorporate fluoride, arsenic, and nitrate. Safe drinking (consumable) water should be passed on to the client for drinking, food game plan, individual neatness, and washing. The water ought to satisfy the normal quality rules for making it pure at the spot of supply to the clients.

One of the greatest machine (statistical) learning algorithms for pragmatic applications is Breiman's random forests (RF). Despite its practical usefulness, random forests remained fairly obscure until recently when compared to other AI and machine learning techniques, with little use in water research, particularly hydrological applications.

As a result, the power of 'Breiman's' unique calculation and its variations in water assets, and applications remains un-utilized. Aside from the standard applications of RF-based calculations in relapse and grouping issues, as well as the calculation of significant measurements, heir utilization for decile expectation, endurance investigation, and offhand surmising, appear to be less well known among water researchers and professionals. Random forest is accepted to have a place with the class of data driven models with regards to water resources.

©IJRASET: All Rights are Reserved

Literature Survey shows a few current studies on the use of information-driven models in water assets, water asset design, and hydrology, where the random forest is lacking and a significant portion of the text is dedicated to brain organizations. Some of the most often discussed topics in literature on information-driven models are expectation (or determination), preprocessing, variable selection, splitting the dataset into preparation and testing phases, and predictive execution assessments.

II. LITERATURE SURVEY

Reference	Author Name	Paper	Theme	Area of Examination	Algorithm	Result
1.	Geetha, Gouthami et al.	Internet of Things Enabled Real Time Water Quality Monitoring System ,2017.	Water quality monitoring	Test water samples and upload data on internet for analysis	None	NA
2.	Ahmed et al.	Efficient Water Quality Prediction Using Supervised Machine Learning, 2019	Water quality levels	Use of machine learning algorithms to estimate water quality index	Gradient Boost Algorithm	Make a base for an economical ongoing water quality recognition framework.
3.	Ashwini et al.	“Intelligent Model For Predicting Water Quality”	Water quality check	Plan and foster a minimal expense framework for the ongoing observing of water quality utilizing the Internet of Things (IoT) and Machine Learning (ML)	K-Nearest Neighbour	It deliver a practical and economical solution without any human intervention
4.	Prasad et al.	“Smart Water Quality Monitoring System”, 2015	Water quality monitoring system	Upload water quality data onto the internet using IoT, and wireless sensors	None	Successfully send the alarm based on the parameter for immediate action.
5.	Mohammed et al.	“Machine Learning: Based Detection of Water Contamination in Water Distribution systems”,2018	Water contamination	Detection of water contamination using machine learning model	None	NA
6.	Singh et al.	Review on Data Mining Techniques for Prediction of Water Quality,2017	Water quality prediction and data mining	Studying various data mining techniques for prediction of water quality	Naïve Bayes, Back Propagation, KNN	NA
7.	Kumar et al.	Smart Water Monitoring System for Real-Time Water Quality and Usage Monitoring,2018	Smart Water Quantity meter and Smart Water Quality meter	Configuration Smart Water Quantity Meter to guarantee water protection by observing how much water drank by a family, and informing something very similar to the shopper and the power	None	Implement quality check meter which improve the predict rate and reduce the error.

8.	Koditala et al.	Water QualityMonitoring System using IoT and Machine Learning, in Proceedings of the IEEE International Conference on Research in Intelligent and Computing in Engineering, pp.1-5, 2018	Water quality monitoring	Use of emerging technologies like IoT, machine learning and cloud computing to replace traditional water quality monitoring techniques	None (but designed some)	Used several sensor to determine the quality of the water which are inexpensive giving a inexpensive solution.
9.	Haghiabi et al.	Water quality prediction using machine learning methods, 2018	Water quality monitoring	Examine execution of artificial intelligence strategies remembering artificial neural network for anticipating water quality parts	Firefly Algorithm	NA
10.	Gollapalli et al.	Ensemble Machine Learning Model to Predict the Waterborne Syndrome, 2022	Maintain hygienic access to clean water	Use of machine learning model extract data on hygienic conditions and water quality	Naïve Bayes	Address the challenges associated with waterborne disease in low income nation.

III. RANDOM FOREST

Breiman's random forest (RF) notion, as well as related ideas and results, are covered in this section. To summarise, Breiman's RF calculation is distinguished from earlier RF executions by the use of characterization and relapse trees as base students; see Section 3.1 below. For convenience and without sacrificing unanimity, we will now accept the RF boundary documentation used in the randomForest R package, which is directly linked to Breiman's original work.

A. How Random Forest Works

A few works and reading materials, for example, Breiman, Biau, and Scornet, as well as James et al., Hastie et al., Kuhn and Johnson, have detailed introductions to RF calculations. A random forest is an AI computation that combines the concepts of classification and regression trees, as well as bagging and randomization. These principles are presented in Segment 3.1.1-Section 3.1.3, and Section 3.1.4 discusses how and why they are connected.

1) Supervised Learning

Supervised learning is one of the most generally practised parts of machine learning(ML) that utilizations named preparing information to assist models with making precise expectations. The preparation information here fills in as a boss and an educator for the machines, subsequently the name. A comparative system is instrumental in settling genuine difficulties, for example, picture classification, spam separating, risk evaluation, and so on. A supervised learning algo generally has an objective or result variable (or ward variable), which is recognized from a gave set of indicators (free factors). The calculation utilizes this arrangement of factors to make a capacity that guides contributions to wanted yields. This preparing system is repeated however long it takes for the model to accomplish an elevated degree of precision. Under the umbrella of supervised learning fall: Classification, Regression and Forecasting. Supervised learning frequently require non-trivial dataset sizes to advance dependably from ground truth perceptions. Models might require a large number of information and result guides to gain from to really perform. Bigger datasets, including more prominent quantities of noteworthy models from which to learn, empower the calculations to integrate an assortment of edge cases and produce models that handle these edge cases exquisitely. Contingent upon the business main concern, numerous long stretches of information are important to represent irregularity.

2) Classification and Regression Trees

In classification tasks, the AI program should reach a determination from observed values and decide to what classification new observations belong. For instance, while filtering messages as 'spam' or 'not spam', the program should check out at existing observational information and channel the messages appropriately. In regression tasks, the AI program should assess - and comprehend - the connections among variables. Regression analysis centers around one ward variable and a progression of other changing factors - making it especially valuable for expectation and determination.

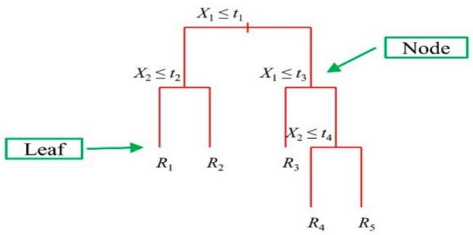


Fig. 1:- Decision tree example. "Xj denotes predictor variables. The tree has four internal nodes and five leaves (terminal nodes). Xj ≤ tk and Xj > tk correspond to the left and right branches of each internal split, respectively. Ri denotes the mean of the observations at leaf I".

The decision criteria for node portions in regression trees are tuned/advanced by streamlining the number of squared deviations, while the Gini record is streamlined in order. It's worth noting that, on the whole, tree-based computations (counting CARTs) are quite raucous, with big differences in the splitting rules and the tree's size having been identified.

3) Bagging

Bagging (truncation for bootstrap aggregation) is a Breiman-proposed outfit learning approach. It generates a bootstrap test from the initial data and then uses the test to generate a model (e.g., a CART). The system is ntree times rehashed. It is the normal to Bag's forecast of the ntree prepared models' expectations. Sacking reduces the discrepancy between expected and actual effort in this way, but it still requires unprejudiced models to function well.

4) Random Forests

Random Forest is bagging of CARTs with some extra level of randomization. Stowing of
Trucks are expected to reduce their precariousness (see Section 3.1.2). Further, randomization is utilized to decrease the relationship between the trees and, thusly, lessen the fluctuation of the expectations (i.e., the normal of the trees). Randomization is led by haphazardly choosing to try indicator factors as a contender for parting. Forecast in relapse is performed by averaging the expectations of each tree, while in the order it is performed by acquiring the larger part class vote from the singular tree class votes. A possibility for boundary tuning of arbitrary backwoods is to use out-of-sack (OOB) blunders. Out-of-pack tests (around 1/3 of the preparation set, see Biau and Scornet) are the examples staying in the wake of bootstrapping the preparation set. The previously mentioned strategy looks like the notable k-overlay cross-approval.

IV. DECISION TREE

Decision tree is the essential design of numerous group learning techniques, called order tree when utilized for characterization and relapse tree when utilized for relapse. In contrast to other order strategies that consolidate a bunch of highlights in a solitary choice advance to perform grouping, choice trees depend on a multi-stage or various leveled choice plan or tree construction and comprise of nodes and coordinated edges. Decision tree comprises of two sorts of nodes: inner nodes and leaf nodes. Component and trait information addressed with "inside node" and the "leaf node" addresses a class. In particular, a "leaf node" address consequence of a choice from the root node to this leaf node, and an inside node addresses the information characterization test performed by then, i.e., the element or on the other hand trait to which the test information has a place. Every node of the decision tree structure makes a paired choice, and the examples contained in the node in view of the consequence of the quality test are separated into sub-node (the root node contains the full arrangement of tests), and the way from the root node to each leaf node relates to a grouping of choice tests. This handling is generally performed by dropping down the tree until a leaf node is reached. In the choice tree approach, the attributes of the information (i.e., other water boundaries) are indicator factors, while the class to be planned (TN) is alluded to as the objective variable. The tree-like construction of a decision tree is displayed in Fig 2.

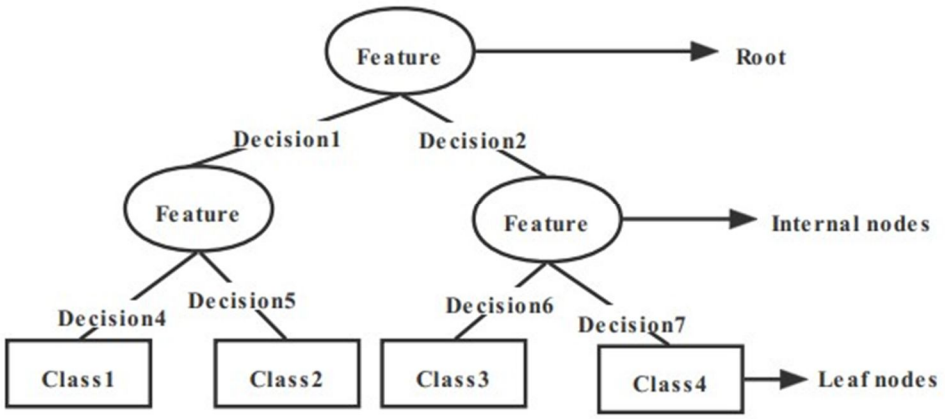


Fig2:- Decision tree structure

V. PROCESS FLOW

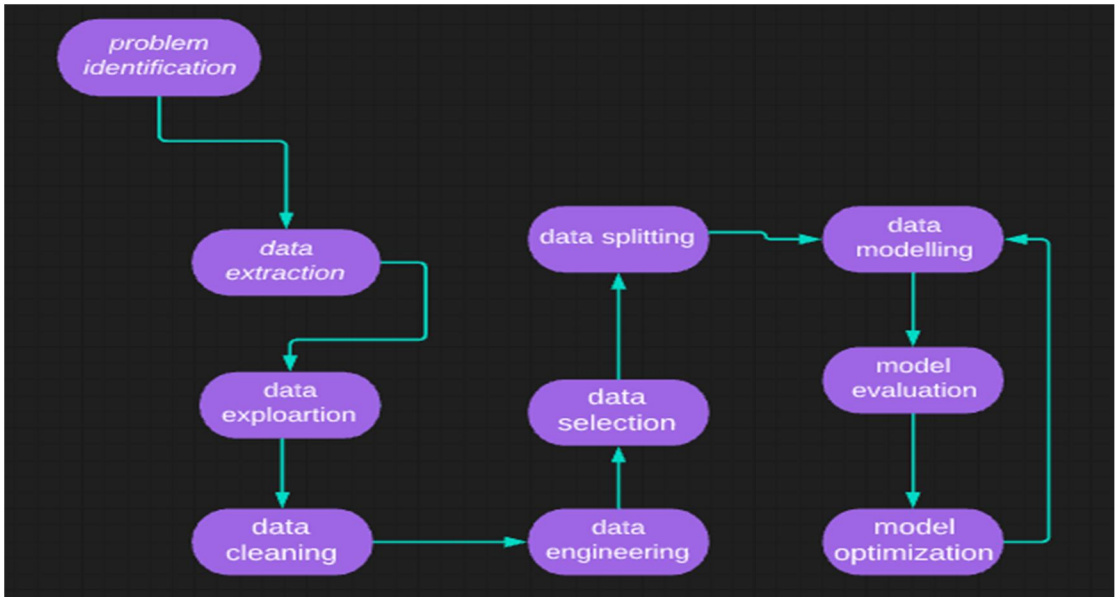


Fig.- Process Flow of the Model

There are basically 10 steps for making our model predict the water quality of the water samples. Those steps are:-

A. Problem Identification

In this step, we identify the problem which is solved by our model. So the problem to be solved by our model is water quality prediction using a dataset.

B. Data Extraction:-

In this, we extract the data from the internet to train our data and predict the water quality. So for that, we take the CPCB(Central Pollution Control Board India) dataset which contains 3277 instances of 13 different wellsprings which are collected between 2014 to 2020.

C. Data Exploration:-

In this step, we analyze the data visually by comparing some parameters of water with the WHO standards of water. It gives a slight overview of the data.

D. Data Cleaning

In this step, we clean that data like if there are some missing values in it so we replace them with mean and remove noise from the data.

E. Data Engineering

In this step, we ensure that the data is quality data so that the prediction accuracy increases.

F. Data Selection

In this step, we select the data types and source of the data. The essential goal of data selection is deciding fitting data type, source, and instrument that permit agents to respond to explore questions sufficiently

G. Data Splitting

In this step, we divide the dataset into smaller subsets for easing the complexity. Normally, with a two-section split, one section is utilized to assess or test the information and the other to prepare the model.

H. Data Modeling

In this step, we create a graph of the dataset for visual representation of data for better understanding. A Data Model is this theoretical model that permits the further structure of conceptual models and to set connections between data.

I. Model Evaluation

Model Evaluation is a fundamental piece of the model improvement process. In this step, we evaluate our model and check how well our model do in the future.

VI. PARAMETERS

Parameters are the basic and most important component of a model. Based on the parameter model prediction and it shows the skill of the model over the data. Similarly, random forest algorithms also have some parameters to predict water quality. We take 10 parameter to predict the water quality. Those parameters are :-

1) Ph
2) Solid
3) Cholride
4) Conductance
5) Sulphate
6) Hardness as CaCO3
7) Trihalomethanes
8) Turbidity
9) Organic Carbon
10) Potability

To predict whether the water is drinkable or not WHO provides some standard values for these parameters of water which is as follows,

Table:- WHO standard for parameters

Parameter	WHO limit
ph	6
Solid	500ppm
Cholride	200mg/l
Conductance	2000 µS/cm Fecal Col
Sulfate	500mg/l
Hardness as CaCO3	500mg/l
Trihalomethanes	0.5ppb
Turbidity	1NTU
Organic Carbon	2mg/l
Potability	1 µg/l

VII. WATER QUALITY INDEX

WQI is the correlation of the sum with an erratic or logical norm or with a pre-determined base. In this way, the WQI observed and announced natural status and patterns on guidelines quantitatively. A water quality list is a way to sum up a lot of water quality information into straightforward terms (e.g., great) for answering to the board and the general population in a predictable way. Notwithstanding the nonattendance of a universally acknowledged composite file of water quality, a few nations have utilized and are involving collected water quality information in the improvement of water quality lists.

To calculate the water quality index(WQI) conventionally we take 10 features of water to reflect the quality of water like ph, chloride, conductance, etc. In this paper, we use all 10 parameters to calculate the WQI of the water. The general formula to calculate the water quality index is given below,

$$\mathrm{WQI} = \frac{\sum q_{value} \times w_factor}{\sum w_factor}$$

where q is the boundary of that parameter, w_factor is the weight of that parameter.

On the basis of WQI value it is determined that the quality of the water is drinkable or not.

VIII. WORKING

A. *First we Import all the Libraries*

Input

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
plt.style.use('fivethirtyeight')
plt.style.use('dark_background')
import seaborn as sns
color = sns.color_palette()
import plotly.express          as ex
import plotly.graph_objs       as go
import plotly.offline          as pyo
import scipy.stats             as stats
import pymc3                   as pm
import theano.tensor           as tt
from matplotlib.colors import ListedColormap
from scipy.stats import norm, boxcox
from sklearn.metrics import confusion_matrix, classification_report, accuracy_score
from collections import Counter
from scipy import stats
from tqdm import tqdm_notebook

from sklearn import metrics
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import mean_squared_error, mean_absolute_error, confusion_matrix, r2_score, accuracy_score
from sklearn.model_selection import (GridSearchCV, KFold, train_test_split, cross_val_score)

from imblearn.over_sampling import SMOTE
from collections import Counter

from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import GaussianNB
from sklearn.ensemble import RandomForestClassifier, ExtraTreesClassifier
from sklearn import svm
from xgboost.sklearn import XGBClassifier
from sklearn.tree import DecisionTreeClassifier
```

B. *Import the Dataset*

Input

```
path = "/content/drive/MyDrive/water_potability.csv"
df = pd.read_csv(path)
```

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

C. Counting Rows and Columns in the dataset

Input

```
df.shape
```

Output:-

(3276, 10)

D. Statistical Analysis

Input:-

```
df.describe().T.style.background_gradient(subset=['mean','std','50%','count'], cmap='PuBu')
```

Output:-

	count	mean	std	min	25%	50%	75%	max
ph	2785.000000	7.080795	1.594320	0.000000	6.093092	7.036752	8.062066	14.000000
Hardness	3276.000000	196.369496	32.879761	47.432000	176.850538	196.967627	216.667456	323.124000
Solids	3276.000000	22014.092526	8768.570828	320.942611	15666.690297	20927.833607	27332.762127	61227.196008
Chloramines	3276.000000	7.122277	1.583085	0.352000	6.127421	7.130299	8.114887	13.127000
Sulfate	2495.000000	333.775777	41.416840	129.000000	307.699498	333.073546	359.950170	481.030642
Conductivity	3276.000000	426.205111	80.824064	181.483754	365.734414	421.884968	481.792304	753.342620
Organic_carbon	3276.000000	14.284970	3.308162	2.200000	12.065801	14.218338	16.557652	28.300000
Trihalomethanes	3114.000000	66.396293	16.175008	0.738000	55.844536	66.622485	77.337473	124.000000
Turbidity	3276.000000	3.966786	0.780382	1.450000	3.439711	3.955028	4.500320	6.739000

From the above table, we can see that the count of each feature is not sthe ame. so there must be some null values. Feature Solids has a high mean and standard deviation compared to other feature. so the distribution must be high. However, the above description is for the overall population. Let's try the same for 2 samples based on the Portability feature.

E. Checking for the portablity of the water

Input

```
df[df['Potability']==1].describe().T.style.background_gradient(subset=['mean','std','50%','count'], cmap='PuBu')
```

Output

	count	mean	std	min	25%	50%	75%	max
ph	1101.000000	7.073783	1.448048	0.227499	6.179312	7.036752	7.933068	13.175402
Hardness	1278.000000	195.800744	35.547041	47.432000	174.330531	196.632907	218.003420	323.124000
Solids	1278.000000	22383.991018	9101.010208	728.750830	15668.985035	21199.386614	27973.236446	56488.672413
Chloramines	1278.000000	7.169338	1.702988	0.352000	6.094134	7.215163	8.199261	13.127000
Sulfate	985.000000	332.566990	47.692818	129.000000	300.763772	331.838167	365.941346	481.030642
Conductivity	1278.000000	425.383800	82.048446	201.619737	360.939023	420.712729	484.155911	695.369528
Organic_carbon	1278.000000	14.160893	3.263907	2.200000	12.033897	14.162809	16.356245	23.604298
Trihalomethanes	1223.000000	66.539684	16.327419	8.175876	56.014249	66.678214	77.380975	124.000000
Turbidity	1278.000000	3.968328	0.780842	1.492207	3.430909	3.958576	4.509569	6.494249

©IJRASET: All Rights are Reserved

Input

```
df[df['Potability']==0].describe().T.style.background_gradient(subset=['mean','std','50%','count'], cmap='RdBu')
```

Output

	count	mean	std	min	25%	50%	75%	max
ph	1684.000000	7.085378	1.683499	0.000000	6.037723	7.035456	8.155510	14.000000
Hardness	1998.000000	196.733292	31.057540	98.452931	177.823265	197.123423	216.120687	304.235912
Solids	1998.000000	21777.490788	8543.068788	320.942611	15663.057382	20809.618280	27006.249009	61227.196008
Chloramines	1998.000000	7.092175	1.501045	1.683993	6.155640	7.090334	8.066462	12.653362
Sulfate	1510.000000	334.564290	36.745549	203.444521	311.264006	333.389426	356.853897	460.107069
Conductivity	1998.000000	426.730454	80.047317	181.483754	368.498530	422.229331	480.677198	753.342620
Organic_carbon	1998.000000	14.364335	3.334554	4.371899	12.101057	14.293508	16.649485	28.300000
Trihalomethanes	1891.000000	66.303555	16.079320	0.738000	55.706530	66.542198	77.277704	120.030077
Turbidity	1998.000000	3.965800	0.780282	1.450000	3.444062	3.948076	4.496106	6.739000

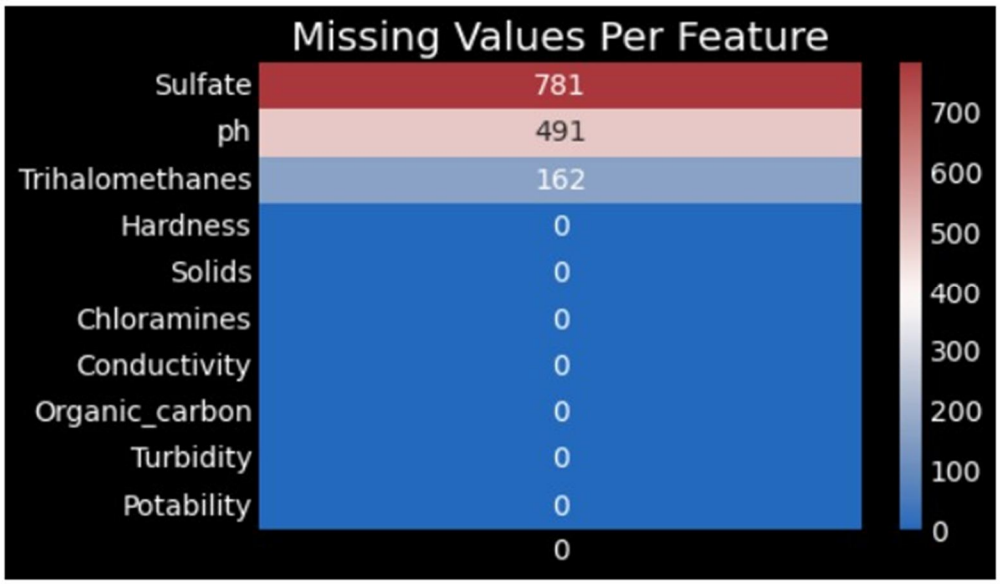
Mean and std of almost all features are similar for both samples. there are few differnces in Solids feature. Further analysis using hypothetical testing could help us to identify the significance

F. Checking For Missing Values

Input

```
plt.title('Missing Values Per Feature')
nans = df.isna().sum().sort_values(ascending=False).to_frame()
sns.heatmap(nans,annot=True,fmt='d',cmap='vlag')
```

Output



As we can see that only sulfate, ph and trihalomethanes has the mssing values so we replace them with the population mean.

G. Filling the Missing Values

Input

```
phMean_0 = df[df['Potability'] == 0]['ph'].mean(skipna=True)
df.loc[(df['Potability'] == 0) & (df['ph'].isna()), 'ph'] = phMean_0
phMean_1 = df[df['Potability'] == 1]['ph'].mean(skipna=True)
df.loc[(df['Potability'] == 1) & (df['ph'].isna()), 'ph'] = phMean_1

##################################### Imputing 'Sulfate' value #####################################

SulfateMean_0 = df[df['Potability'] == 0]['Sulfate'].mean(skipna=True)
df.loc[(df['Potability'] == 0) & (df['Sulfate'].isna()), 'Sulfate'] = SulfateMean_0
SulfateMean_1 = df[df['Potability'] == 1]['Sulfate'].mean(skipna=True)
df.loc[(df['Potability'] == 1) & (df['Sulfate'].isna()), 'Sulfate'] = SulfateMean_1

################################## Imputing 'Trihalomethanes' value ##################################

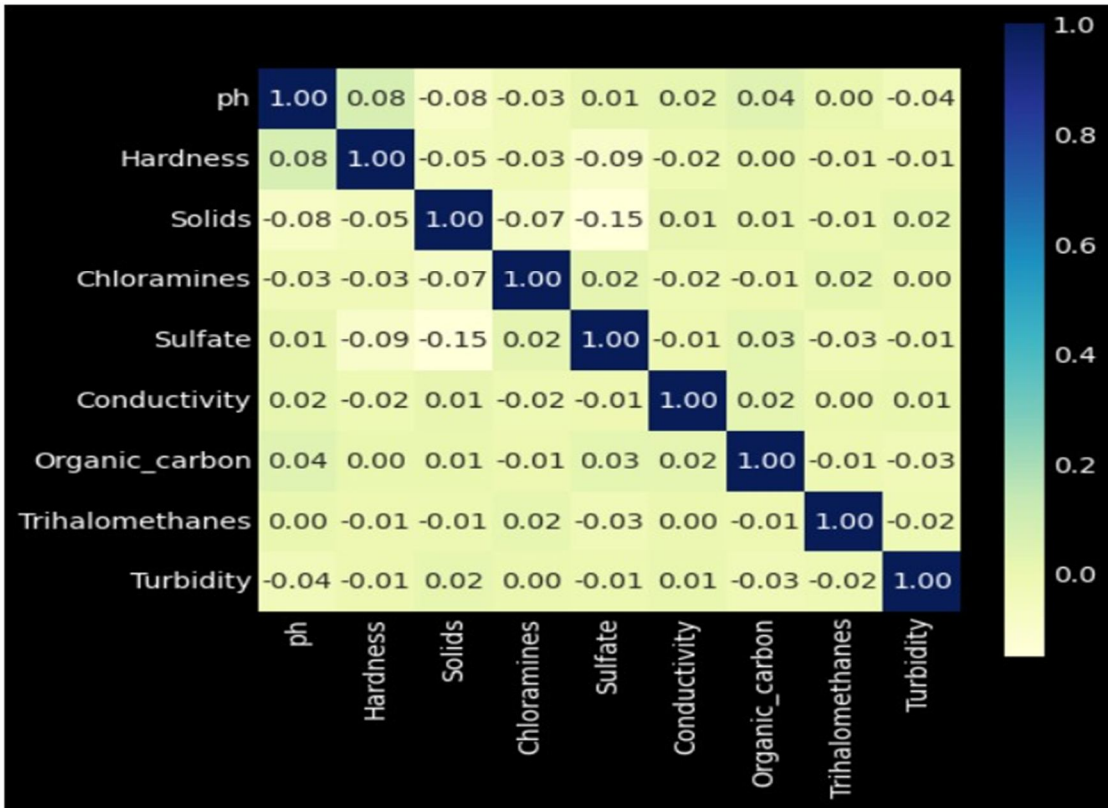
TrihalomethanesMean_0 = df[df['Potability'] == 0]['Trihalomethanes'].mean(skipna=True)
df.loc[(df['Potability'] == 0) & (df['Trihalomethanes'].isna()), 'Trihalomethanes'] = TrihalomethanesMean_0
TrihalomethanesMean_1 = df[df['Potability'] == 1]['Trihalomethanes'].mean(skipna=True)
df.loc[(df['Potability'] == 1) & (df['Trihalomethanes'].isna()), 'Trihalomethanes'] = TrihalomethanesMean_1
```

H. Exploratory Data Analysis

Input

```
Corrmat = df.corr()
plt.subplots(figsize=(7,7))
sns.heatmap(Corrmat, cmap="YlGnBu", square = True, annot=True, fmt='.2f')
plt.show()
```

Output

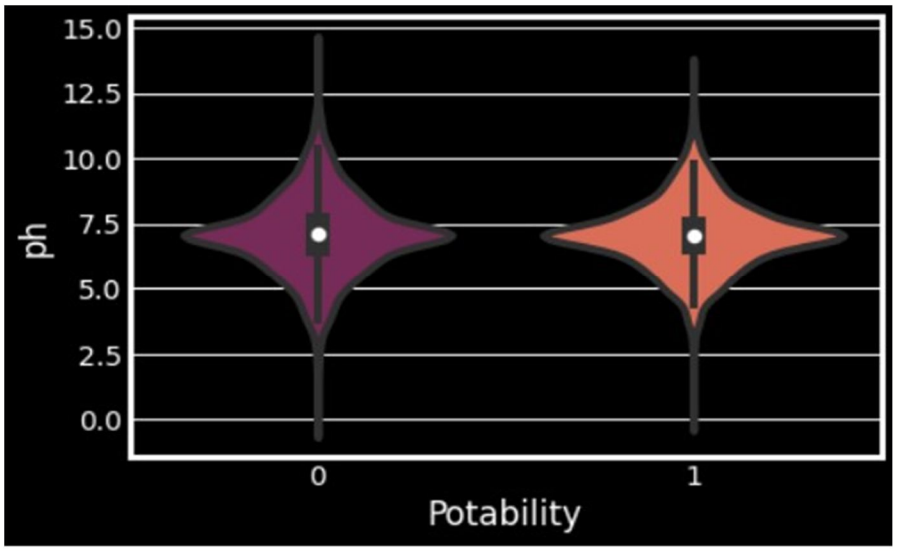


I. Plotting a violinplot between potability and ph

Input

```
sns.violinplot(x='Potability', y='ph', data=df, palette='rocket')
```

Output



J. Boxplot and density distribution of different features by Potability

Input:-

```
print('Boxplot and density distribution of different features by Potability\n')

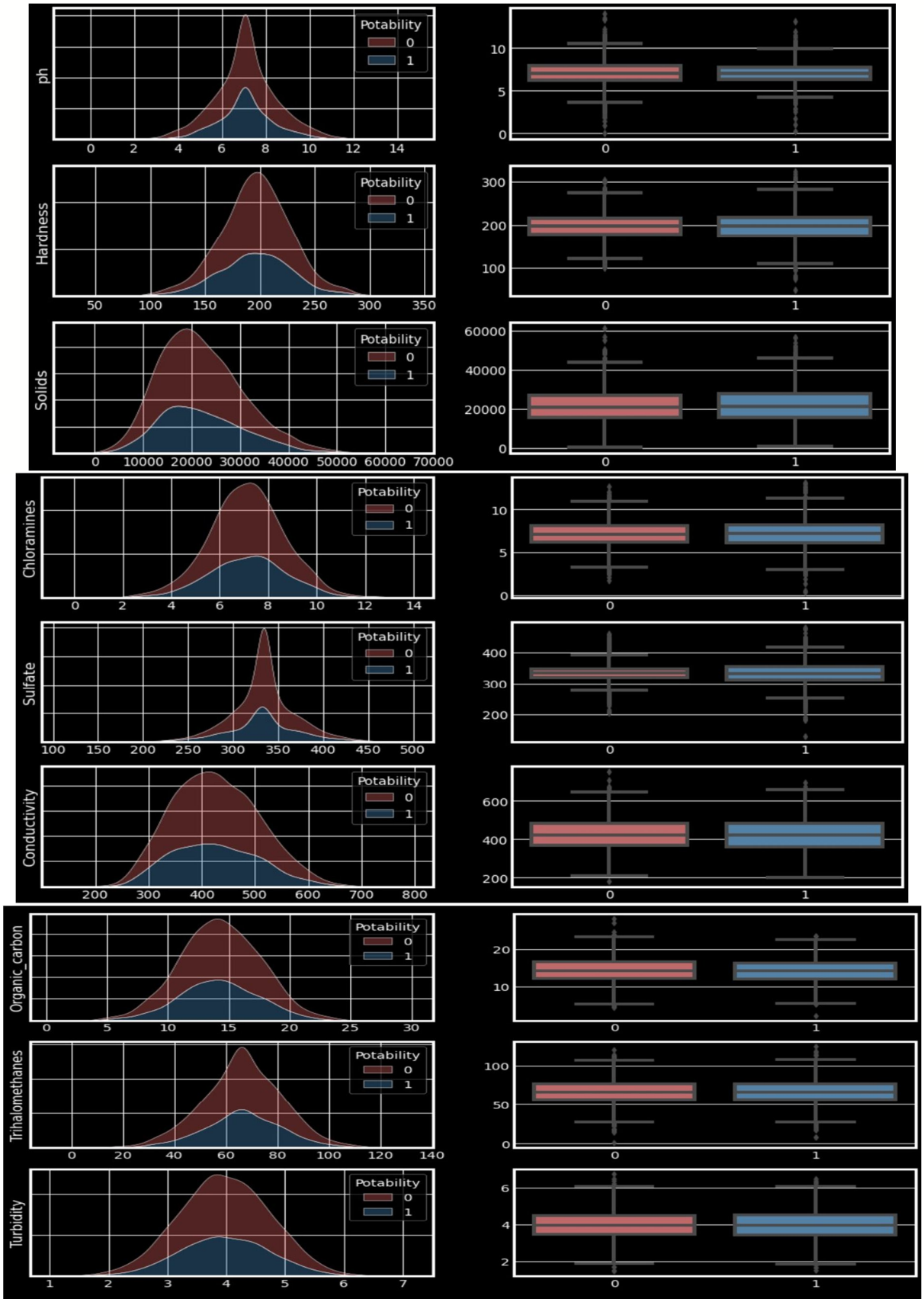
fig, ax = plt.subplots(ncols=2, nrows=9, figsize=(14, 28))

features = list(df.columns.drop('Potability'))
i=0
for cols in features:
    sns.kdeplot(df[cols], fill=True, alpha=0.4, hue = df.Potability,
                palette=('indianred', 'steelblue'), multiple='stack', ax=ax[i,0])

    sns.boxplot(data= df, y=cols, x='Potability', ax=ax[i, 1],
               palette=('indianred', 'steelblue'))
    ax[i,0].set_xlabel(' ')
    ax[i,1].set_xlabel(' ')
    ax[i,1].set_ylabel(' ')
    ax[i,1].xaxis.set_tick_params(labelsize=14)
    ax[i,0].tick_params(left=False, labelleft=False)
    ax[i,0].set_ylabel(cols, fontsize=16)
    i=i+1

plt.show()
```

Output


ph
Hardness
Solids
Chloramines
Sulfate
Conductivity
Organic_carbon
Trihalomethanes
Turbidity
Potability
0
1

K. Balancing the data by SMOTE

Input

```
X = df.drop('Potability', axis = 1).copy()
y = df['Potability'].copy()

############################## Train-Test split ############################
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.25)

########################## Synthetic OverSampling ##########################
print('Balancing the data by SMOTE - Oversampling of Minority level\n')
smt = SMOTE()
counter = Counter(y_train)
print('Before SMOTE', counter)
X_train, y_train = smt.fit_resample(X_train, y_train)
counter = Counter(y_train)
print('\nAfter SMOTE', counter)

################################## Scaling #################################
ssc = StandardScaler()

X_train = ssc.fit_transform(X_train)
X_test = ssc.transform(X_test)

modelAccuracy = list()
```

L. Modeling and Prediction

Input

```
model = [LogisticRegression(), DecisionTreeClassifier(), GaussianNB(), RandomForestClassifier(),
        svm.LinearSVC(), XGBClassifier()]
trainAccuracy = list()
testAccuracy = list()
kfold = KFold(n_splits=10, random_state=7, shuffle=True)

for mdl in model:
    trainResult = cross_val_score(mdl, X_train, y_train, scoring='accuracy', cv=kfold)
    trainAccuracy.append(trainResult.mean())
    mdl.fit(X_train, y_train)
    y_pred = mdl.predict(X_test)
    testResult = metrics.accuracy_score(y_test, y_pred)
    testAccuracy.append(testResult)
```

M. Random Forest Classifier

Input

```
print('Random Forest Classifier\n')
Rfc = RandomForestClassifier()
Rfc.fit(X_train, y_train)

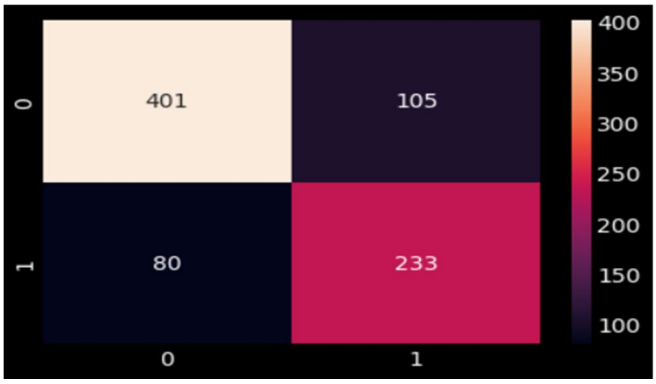
y_Rfc = Rfc.predict(X_test)
print(metrics.classification_report(y_test, y_Rfc))
print(modelAccuracy.append(metrics.accuracy_score(y_test, y_Rfc)))

sns.heatmap(confusion_matrix(y_test, y_Rfc), annot=True, fmt='d')
plt.show()
```

Output

Random Forest Classifier

	precision	recall	f1-score	support
0	0.83	0.79	0.81	506
1	0.69	0.74	0.72	313
accuracy			0.77	819
macro avg	0.76	0.77	0.76	819
weighted avg	0.78	0.77	0.78	819



N. Conclusion of the Program

By running the program we find that random forest classifier is best suited for this dataset and so its better to use this classifier than the other models for better prediction and accuracy.

IX. CONCLUSION AND FUTURE WORK

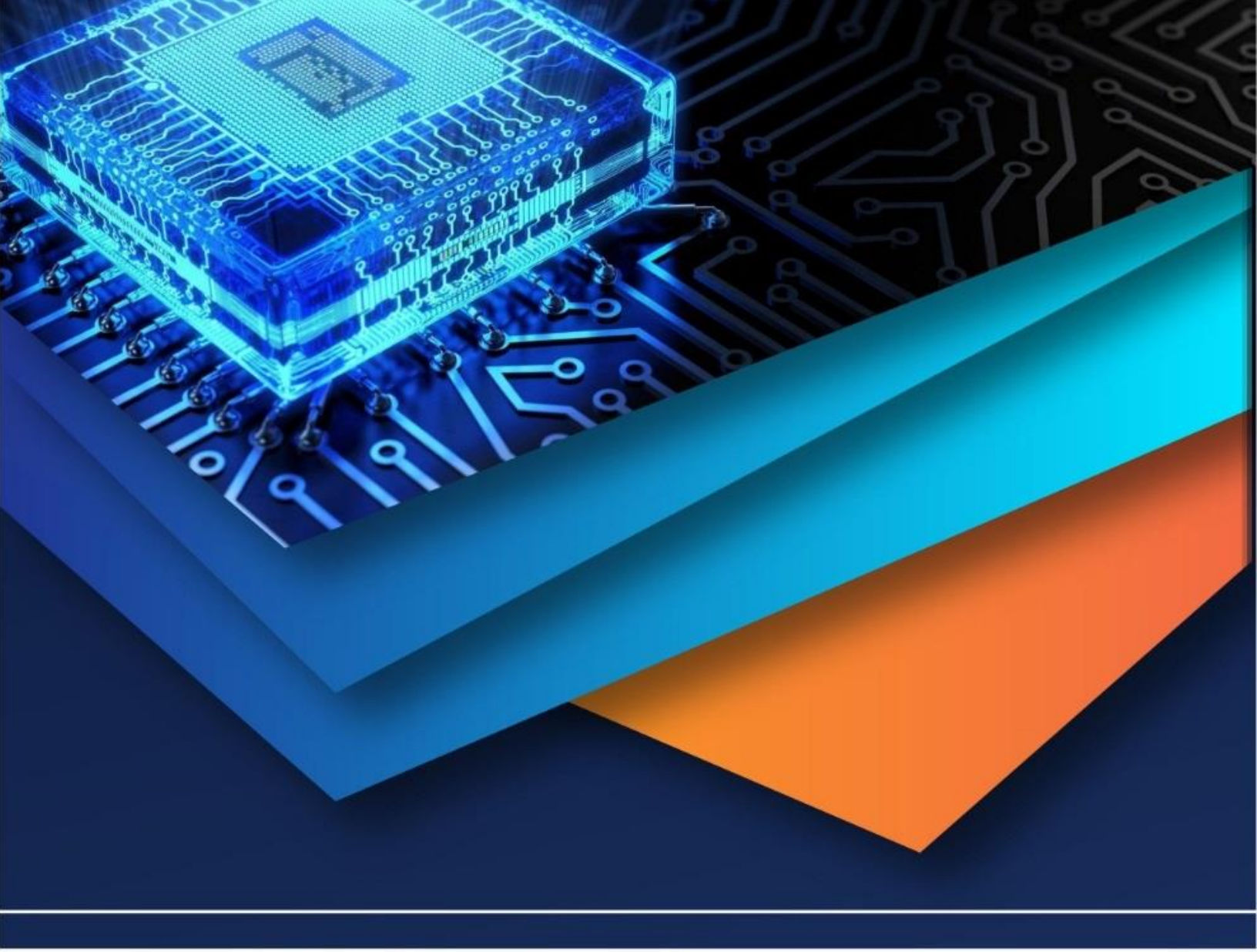
As we all know the importance of water for the human body. So knowing the Quality of the water is very much necessary because if we drink water without knowing that it is safe for drinking we could get sick. There are plenty of water-borne diseases like Cholera, Typhoid, Giardia, E. Coli, Hepatitis A, and so on. These types of diseases happen if we drink non-drinkable water. So knowing the quality of the water is the most important thing. But the main problem lies here. For testing the water quality we have to conduct lab tests on the water which is costly and time-consuming as well. So, in this paper, we propose an alternative approach using artificial intelligence to predict water quality. This method uses a significant and easily available water quality index which is set by the WHO(World Health Organisation). The data taken in this paper is taken from the PCPB India which includes 3277 examples of the distinct wellspring. In this paper, WQI(Water Quality Index) is calculated using AI techniques.

So in future work, we can integrate this with IoT based framework to study large datasets and to expand our study to a larger scale. By using that it can predict the water quality fast and more accurately than any other IoT framework. That IoT framework system uses some limits for the sensor to check the parameters like ph, Temperature, Turbidity, and so on. And further after reading this parameter pass these readings to the Arduino microcontroller and ZigBee handset for further prediction. This proposed IoT system would recognize the correct quality of water and raise some alarm to the concerned professional for further action. It prevents people from drinking undrinkable water and reduces the danger of having water-borne diseases like typhoid to happen. So by this way, we can use this model for future workshops on water quality and raise awareness

REFERENCES

[1] S. Geetha, S. Gouthami. Internet of Things Enabled Real Time Water Quality Monitoring System ,2017.
[2] Umair Ahmed, Rafia Mumtaz, Hirra Anwar, Asad A. Shah, Rabia Irfan and José García-Nieto. Efficient Water Quality Prediction Using Supervised Machine Learning, 2019.
[3] Ashwini K, D. Diviya, J.Janice Vedha, M. Deva Priya.Intelligent Model For Predicting Water Quality.
[4] A.N.Prasad, K. A. Mamun, F. R. Islam, H. Haqva.Smart Water Quality Monitoring System, 2015
[5] Hadi Mohammed, Ibrahim A. Hameed, Razak Seidu.Machine Learning: Based Detection of Water Contamination in Water Distribution systems,2018
[6] Priya Singh,Pankaj Deep Kaur.Review on Data Mining Techniques for Prediction of Water Quality,2017

[7] Manish Kumar Jha,Rajni Kumari Sah,M.S.Rashmitha,Rupam Sinha,B.Sujatha.Smart Water Monitoring System for Real-Time Water Quality and Usage Monitoring,2018
[8] Water QualityMonitoring System using IoT and Machine Learning, in Proceedings of the IEEE International Conference onResearch in Intelligent and Computing in Engineering, pp.1-5, 2018
[9] Amir Hamzeh Haghiabi, Ali Heidar Nasrolahi, Abbas Parsaie. Water quality prediction using machine learning methods, 2018
[10] A Gollapalli, Mohammed; Ensemble Machine Learning Model to Predict the Waterborne Syndrome, 2022.
[11] Breiman, L. Random forests. Mach. Learn. 2001
[12] Biau, G.Ã.Š.; Scornet, E. A random forest guided tour. TEST 2016, 2
[13] Liakos, K.; Busato, P.; Moshou, D.; Pearson, S.; Bochtis, D. Machine learning in agriculture: A Review. Sensors 2018
[14] Gislason, P.O.; Benediktsson, J.A.; Sveinsson, J.R. Random forests for land cover classification. Pattern Recognit. Lett. 2006
[15] Maxwell, A.E.; Warner, T.A.; Fang, F. Implementation of machine-learning classification in remote sensing: An applied review. Int. J. Remote Sens. 2018
[16] Chen, X.; Ishwaran, H. Random forests for genomic data analysis. Genomics 2012
[17] Boulesteix, A.L.; Janitza, S.; Kruppa, J.; König, I.R. Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. Wiley Interdiscip. Rev. Data Min. Knowl. Discov. 2012
[18] Solomatine, D.P.; Ostfeld, A. Data-driven modelling: Some past experiences and new approaches. J. Hydroinformatics 2008
[19] Gazzaz, N.M.; Yusoff, M.K.; Aris, A.Z.; Juahir, H.; Ramli, M.F. Artificial neural network modeling of the water quality index for Kinta River (Malaysia) using water quality variables as predictors. Mar. Pollut. Bull. 2012
[20] Sakizadeh, M. Artificial intelligence for the prediction of water quality index in groundwater systems. Model. Earth Syst. Environ. 2016
[21] Abyaneh, H.Z. Evaluation of multivariate linear regression and artificial neural networks in prediction of water quality parameters. J. Environ. Health Sci. Eng. 2014
[22] Kangabam, R.D.; Bhoominathan, S.D.; Kanagaraj, S.; Govindaraju, M. Development of a water quality index (WQI) for the Loktak Lake in India. Appl. Water Sci. 2017
[23] Srivastava, G.; Kumar, P. Water quality index with missing parameters. Int. J. Res. Eng. Technol. 2013,
[24] Jayalakshmi, T.; Santhakumaran, A. Statistical normalization and back propagation for classification. Int. J. Comput. Theory Eng. 2011
[25] Liaw, A.; Wiener, M. Classification and regression by randomForest. R News 2002
[26] Günther, F.; Fritsch, S. Neuralnet: Training of neural networks. R J. 2010
[27] Singh, J.; Yadav, P.; Pal, A.K.; Mishra, V. Water pollutants: Origin and status. In Sensors in Water Pollutants Monitoring: Role of Material; Springer: Berlin/Heidelberg, Germany, 2020
[28] Jiang, J.; Tang, S.; Han, D.; Fu, G.; Solomatine, D.; Zheng, Y. A comprehensive review on the design and optimization of surface water quality monitoring networks. Environ. Model. Softw. 2020
[29] Park, J.; Kim, K.T.; Lee, W.H. Recent advances in information and communications technology (ICT) and sensor technology for monitoring water quality. Water 2020



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)