**WATER QUALITY ANALYSIS**

**PHASE -5**

Projects Objective

Access to safe drinking water is essential for good health, the objective of water quality analysis is to assess and ensure the safety, cleanliness, and suitability of water for various purposes, including human consumption, agriculture, industrial processes, and ecosystem maintenance. In this project we will be analysing a given water sample by finding its pH, Hardness, Solids, Cholramines, Sulfate, conductivity etc and check whether the water is fit for drinking

Design Thinking

In the phase of design thinking we will understand the motive of the project and make a plan on how to proceed further, in this project we collected datas for this project and we understood this project using visualization strategies like correlation analysis , parameter distribution etc and we also created a predictive model using machine learning algorithm.

Data Preprocessing

Data processing is used to transform the given data into a format that is most effective before using the dataset, in data processing we remove all the null values and all the outliners.

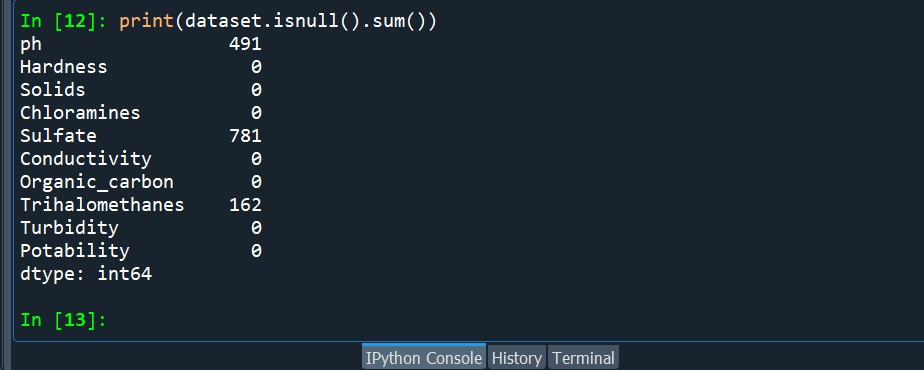
In this part we removed all the null values and outliners form the given data set

To remove the null values

dataset.columns

dataset.describe

print(dataset.isnull().sum())

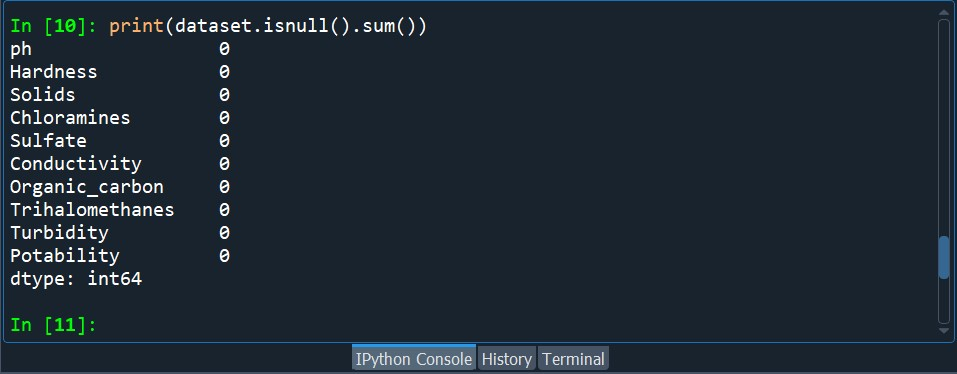


dataset.ph=dataset.ph.fillna(“unknown”)

dataset.Sulfate=dataset.Sulfate.fillna(“unknown”)

dataset.Trihalomethanes=dataset.Trihalomethanes.fillna(“unknown”)

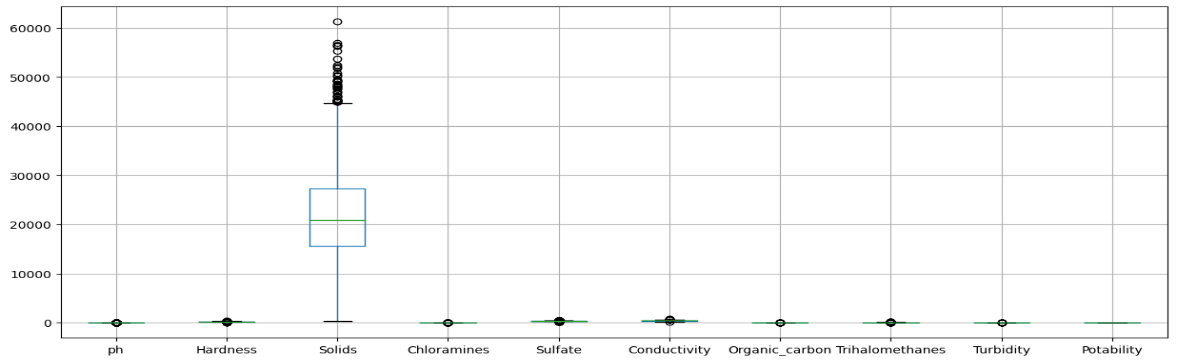
print(dataset.isnull().sum())



Now to find the outliners

*data.boxplot(figsize=(15,6))*

*plt.show()*

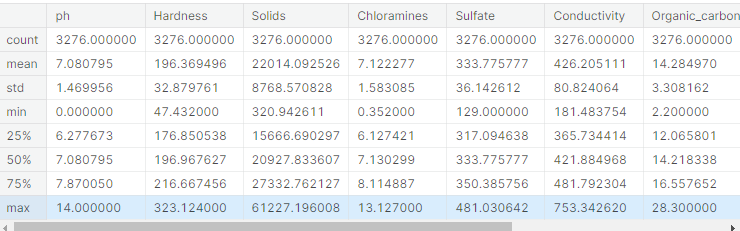
*o/p*

Exploratory Data Analysis(EDA)

Exploratory Data Analysis (EDA) refers to the method of studying and exploring record sets to apprehend their predominant traits, discover patterns, locate outliers, and identify relationships between variables. EDA is normally carried out as a preliminary step before undertaking extra formal statistical analyses or modeling.

data.describe()

o/p:



**Checking if we need to do Dimensonility Reduction**

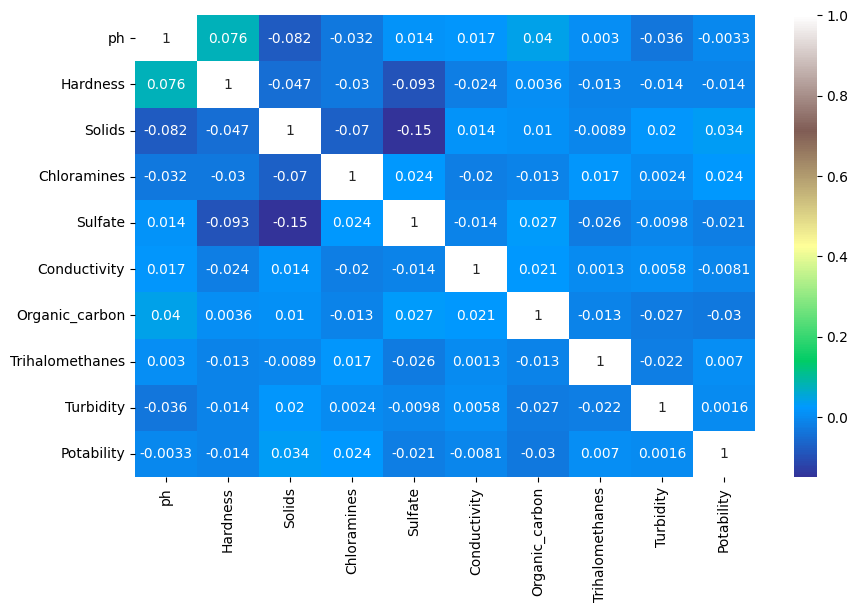
we are trying the reduce the dimension to which are correlating for that we are looking for the similarity of the features with this chart because less feature is making easy to the predict but we have so small similarity of the features and we cant use the remove feature

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

*fig= plt.gcf()*

*fig.set\_size\_inches(10,6)*

*plt.show()*

**

Data Visualization

In data visualization we created histograms and scatter plots to make the given data set more understandable

data['Potability'].value\_counts()

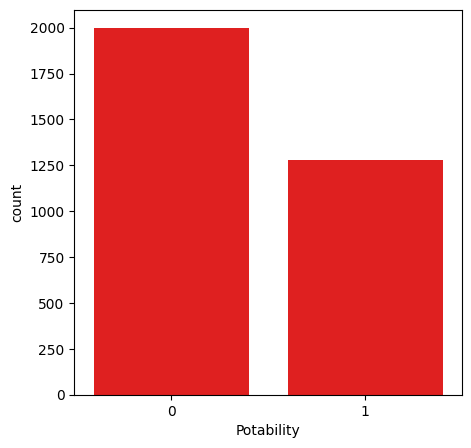
o/p Potability

0 1998

1 1278

plt.figure(figsize=(5,5))

sns.countplot( x=data["Potability"], color="red")

plt.show()

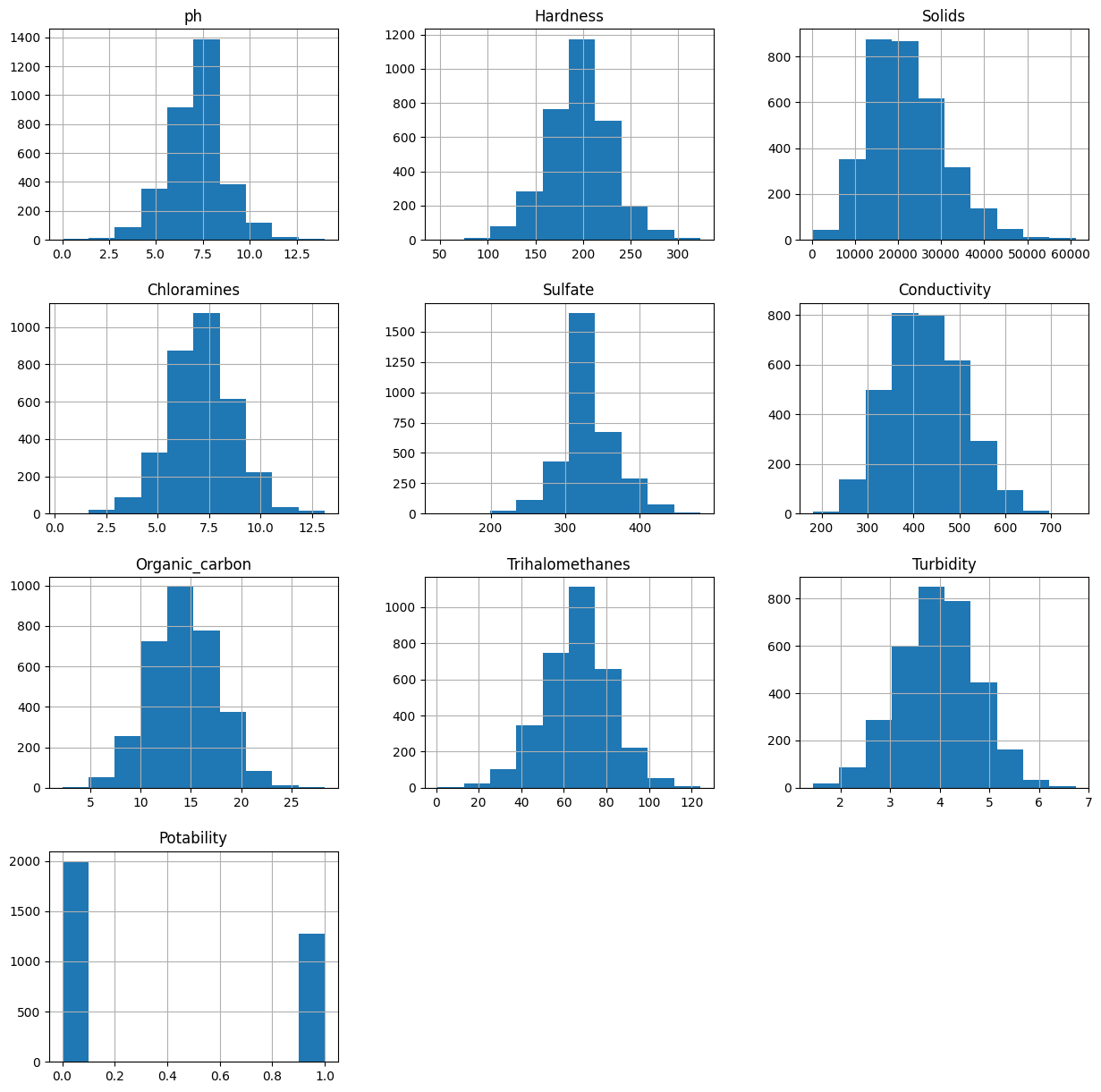
o/p

# Histogram

We also plotted hisogram for each column to make them easily understandable

data.hist(figsize=(15,15))

plt.show()



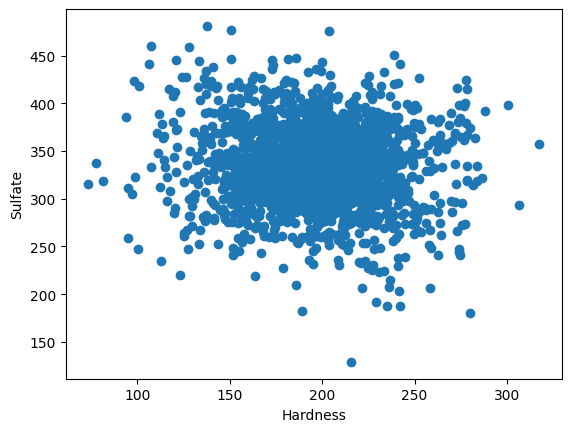
# Scatter Plot

gp = plt.scatter(ks['Hardness'],ks['Sulfate'])

plt.xlabel('Hardness')

plt.ylabel('Sulfate')

plt.show(gp)



sns.scatterplot(x=data['ph'], y=data['Potability'])

plt.show()

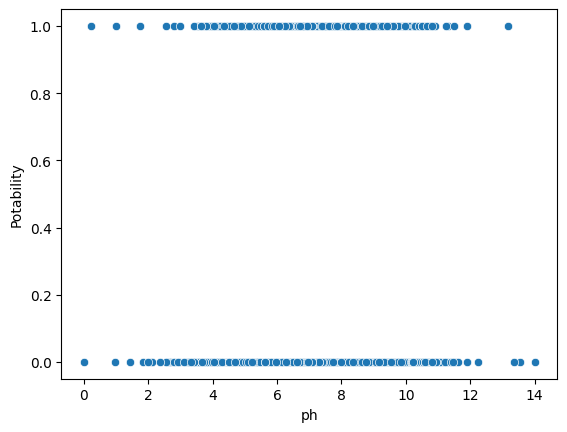


fig = px.scatter(data,x ="ph",y="Hardness",color= "Potability",template="plotly\_dark")

fig.show()

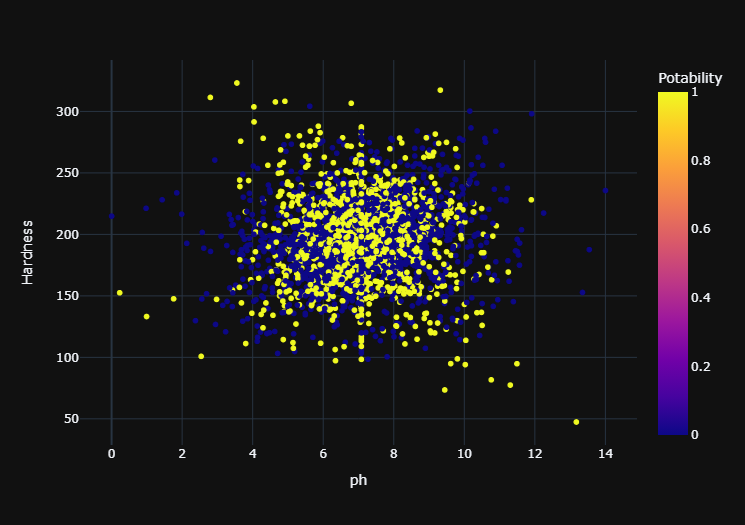
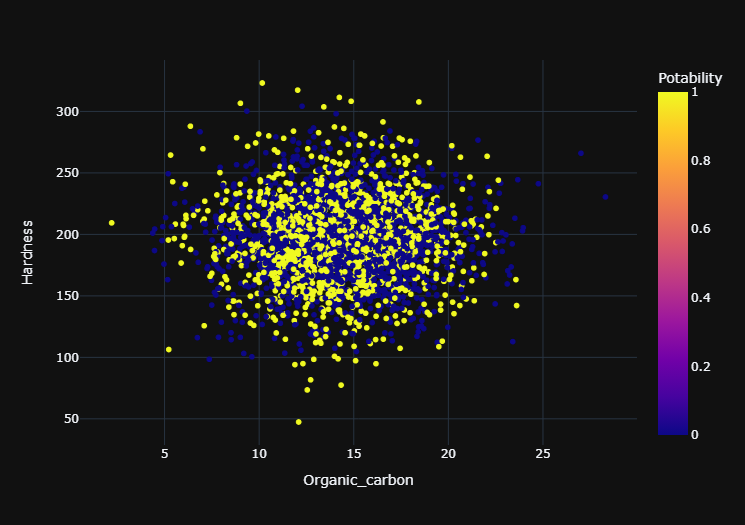


fig = px.scatter(data,x ="Organic\_carbon",y="Hardness",color= "Potability",template="plotly\_dark")

fig.show()



# **Predictive Model Training**

Predictive modeling is the process of using known results to create a statistical model that can be used for predictive analysis

## Y= data['Potability'] # target variable is 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, shuffle=True,random\_state=101)

### we splited the data to make a prediction on that train data

X\_train

## X\_test

## Y\_train

o/p

748 1

2279 0

1960 1

1491 1

2991 0

..

599 0

1599 1

1361 0

1547 1

863 0

Y\_test

o/p

2541 0

2605 0

330 1

515 0

400 1

..

482 0

2970 0

50 0

839 0

374 1

Random Forest

The random forest algorithm is made up of a collection of decision trees, and each tree in the ensemble is comprised of a data sample drawn from a training set with replacement, called the bootstrap sample

from sklearn.tree import DecisionTreeClassifier

dt = DecisionTreeClassifier(criterion= 'entropy', min\_samples\_split= 3,)

dt.fit(X\_train,Y\_train)

Y\_test

o/p

2541 0

2605 0

330 1

515 0

400 1

..

482 0

2970 0

50 0

839 0

374 1

Y\_prediction=dt.predict(X\_test)

from sklearn.metrics import accuracy\_score, confusion\_matrix

accuracy\_score(Y\_prediction,Y\_test)\*100

o/p:59.756097560975604

confusion\_matrix(Y\_prediction,Y\_test)

o/p array([[262, 124],

[140, 130]])

from sklearn.model\_selection import GridSearchCV

from sklearn.model\_selection import RepeatedStratifiedKFold

dt= DecisionTreeClassifier()

criterion = ["gini","entropy"]

splitter = ['best','random ']

min\_samples\_split=range (1,10)

parameters = dict(criterion=criterion,splitter= splitter, min\_samples\_split= min\_samples\_split)

cv= RepeatedStratifiedKFold(n\_splits = 5,random\_state=101)

grid\_search\_cv\_dt= GridSearchCV(estimator=dt, param\_grid=parameters,scoring='accuracy',cv=cv)

grid\_search\_cv\_dt.fit(X\_train,Y\_train)

o/p:

One or more of the test scores are non-finite:

[ nan nan 0.57664122 nan 0.5798855 nan

0.58038168 nan 0.57916031 nan 0.58041985 nan

0.58118321 nan 0.58053435 nan 0.58049618 nan

nan nan 0.58519084 nan 0.58431298 nan

0.58473282 nan 0.58461832 nan 0.58278626 nan

0.58534351 nan 0.58305344 nan 0.58793893 nan]

print(grid\_search\_cv\_dt.best\_params\_)

prediction\_grid=grid\_search\_cv\_dt.predict(X\_test)

accuracy\_score(Y\_test,prediction\_grid)\*100

o/p 59.756097560975604

How the insights from the analysis can help assess water quality and determine potability?

Water quality analysis provides crucial insights that are fundamental for ensuring safe drinking water. By understanding the composition and contaminants present, we can compare the data with local, national, or international standards (e.g., EPA standards in the U.S.). If the levels of contaminants exceed the permissible limits, the water is considered non-potable.

ConclusionTop of Form

In this project we explained the objective of water quality analysis ,what we understood about it and to proceed the project further we made a design plan .And to make the data set more effective , we preprocessed the data by removing all the null values and finding the outliners.

We have also used data visualisation and predictive models to enhance our understanding about water potability.

By leveraging visualization libraries such as Matplotlib and Seaborn, we have created informative histograms, scatter plots, and correlation matrices, which have illuminated the relationships between various water quality parameters. These visualizations have provided valuable insights into the data, helping us identify patterns and trends that might not be immediately apparent from raw numbers alone.

Moreover, the development of a predictive model, utilizing techniques like Logistic Regression and Random Forest, has taken our analysis a step further. We have not only assessed the current water quality but also sought to predict water potability based on the collected data. This predictive model has the potential to be a valuable tool for assessing the safety of water sources in real-time and making informed decisions about water treatment and distribution.