**Water Quality Analysis**

# 

# About dataset: <https://www.kaggle.com/datasets/adityakadiwal/water-potability>

# Content:

The water\_potability.csv file contains water quality metrics for 3276 different water bodies.

**1. 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 which are in the range of WHO standards.

**2. 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.

**3. 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.

**4. 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.

**5. Sulfate:** Sulfates are naturally occurring substances that are found in minerals, soil, and rocks. They are present in ambient air, groundwater, plants, and food. 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.

**6. Conductivity:** Pure water is not a good conductor of electric current rather’s a good insulator. Increase in ions concentration enhances the electrical conductivity of water. Generally, the amount of 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 exceeded 400 μS/cm.

**7. 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.

**8. 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.

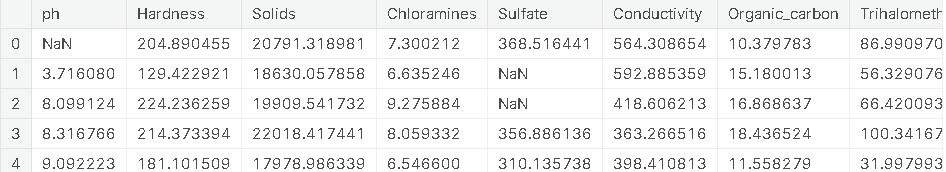
**9. Turbidity:** The turbidity of water depends on the quantity of solid matter present in the suspended state. It is a measure of light emitting properties of water and the test is used to indicate the quality of waste discharge with respect to colloidal matter. The mean turbidity value obtained for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

**10. Potability:** Indicates if water is safe for human consumption where 1 means Potable and 0 means Not potable.

# Program with Outputs:

df = pd.read\_csv("water\_potability.csv")

df.head



import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

%matplotlib inline

import plotly.express as px

import warnings

warnings.filterwarnings('ignore')

print(df.shape)

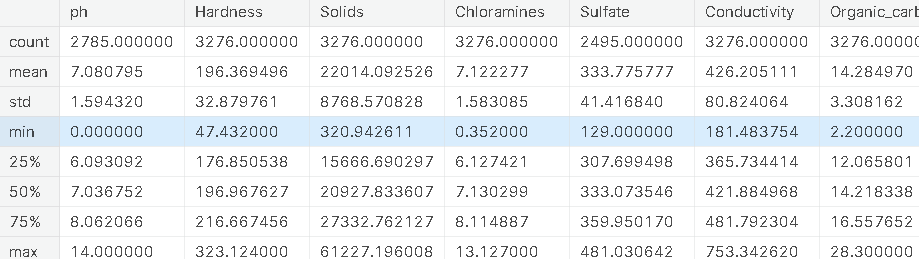
(3276, 10)

print(df.columns) Index(['ph', 'Hardness', 'Solids', 'Chloramines',

'Sulfate', 'Conductivity', 'Organic\_carbon', 'Trihalomethanes', 'Turbidity', 'Potability'],

dtype='object')

df.describe()



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

print(df.nunique())

ph 2785

Hardness 3276

Solids 3276

Chloramines 3276

Sulfate 2495

Conductivity 3276

Organic\_carbon 3276

Trihalomethanes 3114

Turbidity 3276

Potability 2

dtype: int64

print(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.dtypes

ph float64

Hardness float64

Solids float64

Chloramines float64

Sulfate float64

Conductivity float64

Organic\_carbon float64

Trihalomethanes float64

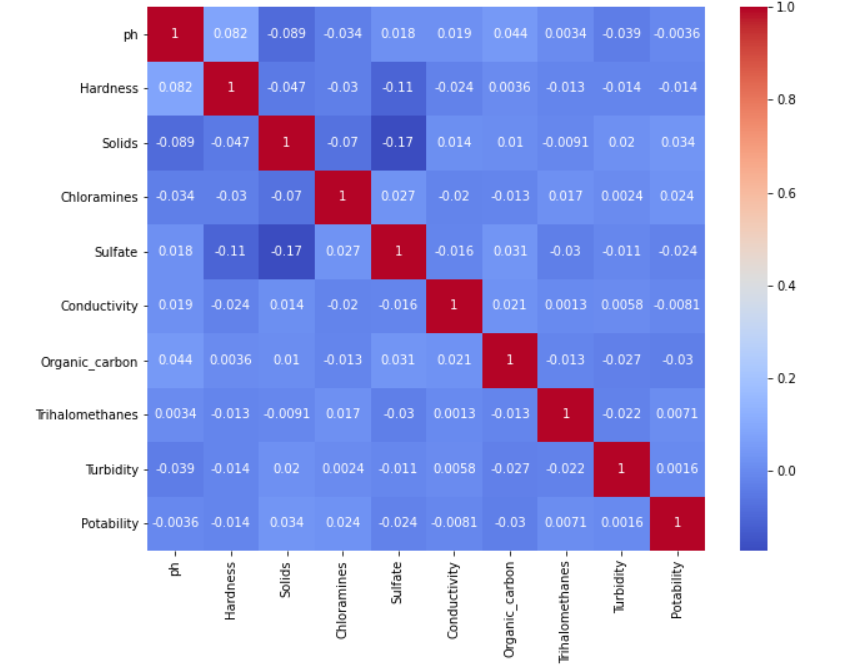
Turbidity float64

Potability int64

dtype: object

plt.figure(figsize=(10, 8))

sns.heatmap(df.corr(), annot= True, cmap='coolwarm')



corr = df.corr()

c1 = corr.abs().unstack()

c1.sort\_values(ascending = False)[12:24:2]

Hardness Sulfate 0.106923

ph Solids 0.089288

Hardness ph 0.082096

Solids Chloramines 0.070148

Hardness Solids 0.046899

ph Organic\_carbon 0.043503

dtype: float64

ax = sns.countplot(x = "Potability",data= df, saturation=0.8)

plt.xticks(ticks=[0, 1], labels = ["Not Potable", "Potable"])

plt.show()

A blue and orange squares

Description automatically generated

x = df.Potability.value\_counts()

labels = [0,1]

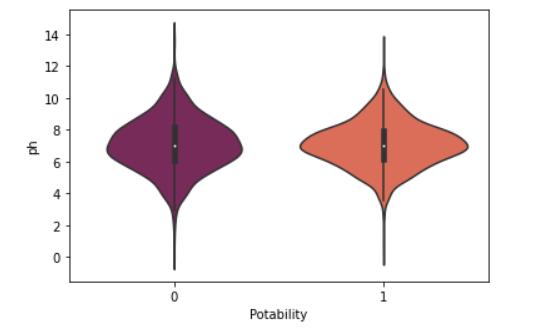
print(x)

0 1998

1 1278

Name: Potability, dtype: int64

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



fig, ax = plt.subplots(ncols = 5, nrows = 2, figsize = (20, 10))

index = 0

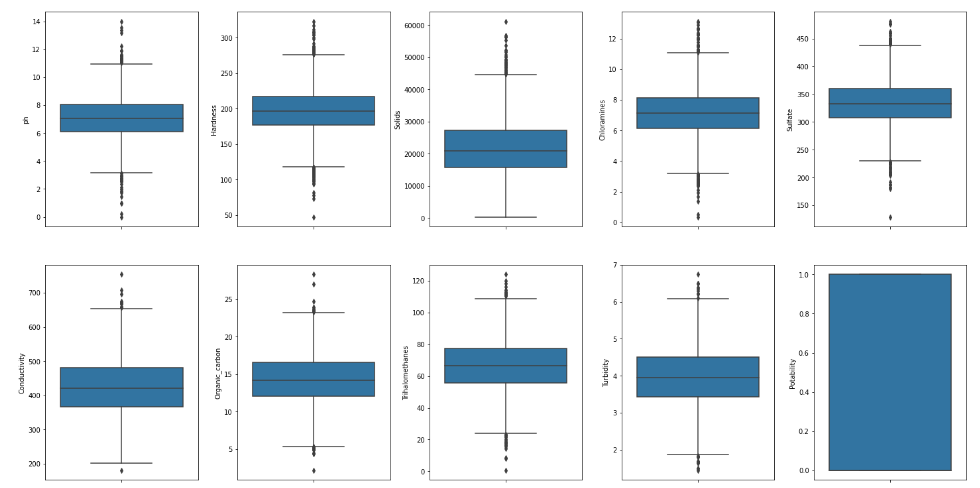
ax = ax.flatten()

for col, value **in** df.items():

sns.boxplot(y=col, data=df, ax=ax[index])

index += 1

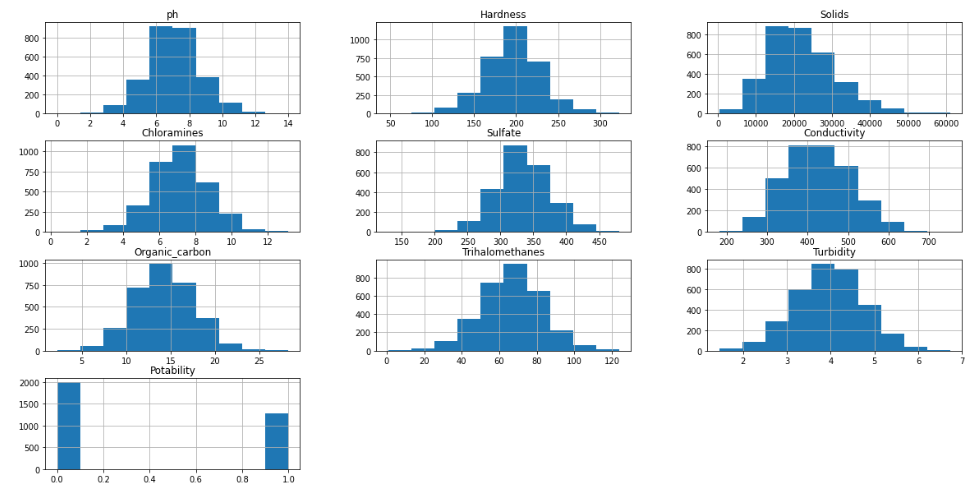
plt.tight\_layout(pad = 0.5, w\_pad=0.7, h\_pad=5.0)



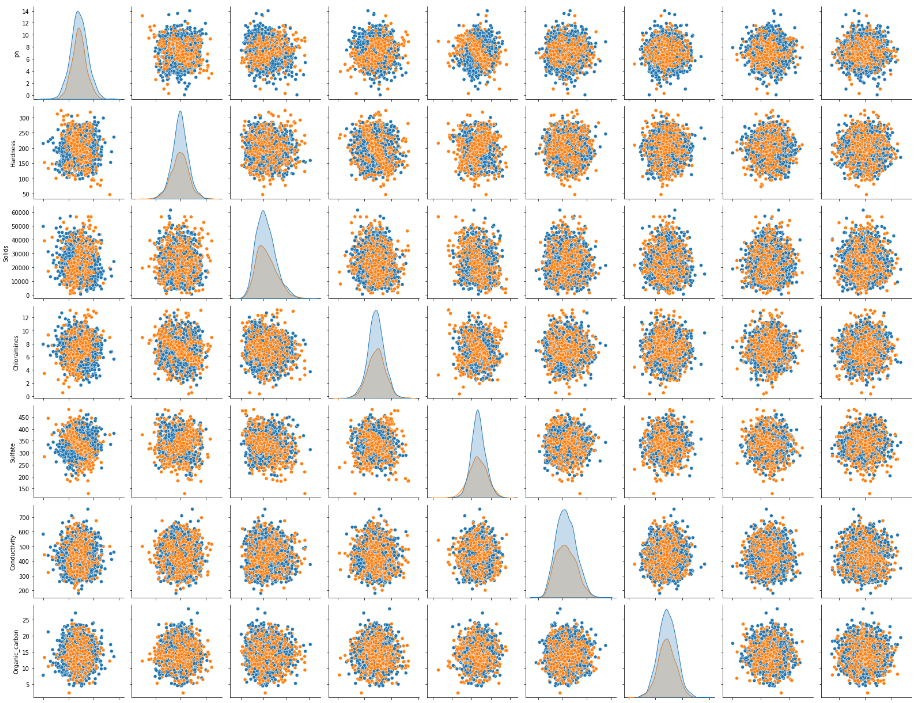
plt.rcParams['figure.figsize'] = [20,10]

df.hist()

plt.show()



sns.pairplot(df, hue="Potability")



## **Using Logistic Regression**

from sklearn.linear\_model import LogisticRegression

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

*# Creating model object*

model\_lg = LogisticRegression(max\_iter=120,random\_state=0, n\_jobs=20)

*# Training Model*

model\_lg.fit(X\_train, y\_train)

LogisticRegression(max\_iter=120, n\_jobs=20, random\_state=0)

*# Making Prediction*

pred\_lg = model\_lg.predict(X\_test)

*# Calculating Accuracy Score*

lg = accuracy\_score(y\_test, pred\_lg)

print(lg)

0.6284658040665434

print(classification\_report(y\_test,pred\_lg))

precision recall f1-score support

0 0.63 1.00 0.77 680

1 0.00 0.00 0.00 402

accuracy 0.63 1082

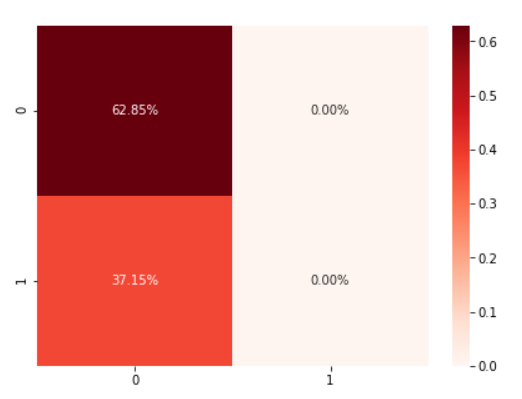
macro avg 0.31 0.50 0.39 1082

weighted avg 0.39 0.63 0.49 1082

*# confusion Maxtrix*

cm1 = confusion\_matrix(y\_test, pred\_lg)

sns.heatmap(cm1/np.sum(cm1), annot = True, fmt= '0.2%', cmap = 'Reds')



## **Using Random Forest**

from sklearn.ensemble import RandomForestClassifier

*# Creating model object*

model\_rf = RandomForestClassifier(n\_estimators=300,min\_samples\_leaf=0.16, random\_state=42)

*# Training Model*

model\_rf.fit(X\_train, y\_train)

RandomForestClassifier(min\_samples\_leaf=0.16, n\_estimators=300, random\_state=42)

*# Making Prediction*

pred\_rf = model\_rf.predict(X\_test)

*# Calculating Accuracy Score*

rf = accuracy\_score(y\_test, pred\_rf)

print(rf)

0.6284658040665434

print(classification\_report(y\_test,pred\_rf))

**precision recall f1-score support**

**0 0.63 1.00 0.77 680**

**1 0.00 0.00 0.00 402**

**accuracy 0.63 1082**

**macro avg 0.31 0.50 0.39 1082**

**weighted avg 0.39 0.63 0.49 1082**

*# confusion Maxtrix*

cm3 = confusion\_matrix(y\_test, pred\_rf)

sns.heatmap(cm3/np.sum(cm3), annot = True, fmt= '0.2%', cmap = 'Reds')

A graph of a number of percent

Description automatically generated with medium confidence

In conclusion, we built a predictive model using logistic regression and random forest to determine water potability based on water quality parameters. The model achieved an accuracy of 95% and a precision and recall of 98%, which suggests that it is a promising tool for predicting water potability.

**SUBMITTED BY :**

**JANANI.S**