**WATER QUALITY ANALYSIS**

**BATCH MEMBER**

**712521121028:SIVADHARMAN.R**

**Phase 4 Submission Document**

**Project Title: Water Quality Analysis**

**Phase3: *Development part 2***

**Topic: Continue analysing the water quality by feature engineering, model training and evaluation**



**WATER QUALITY ANALYSIS**

**Introduction:**

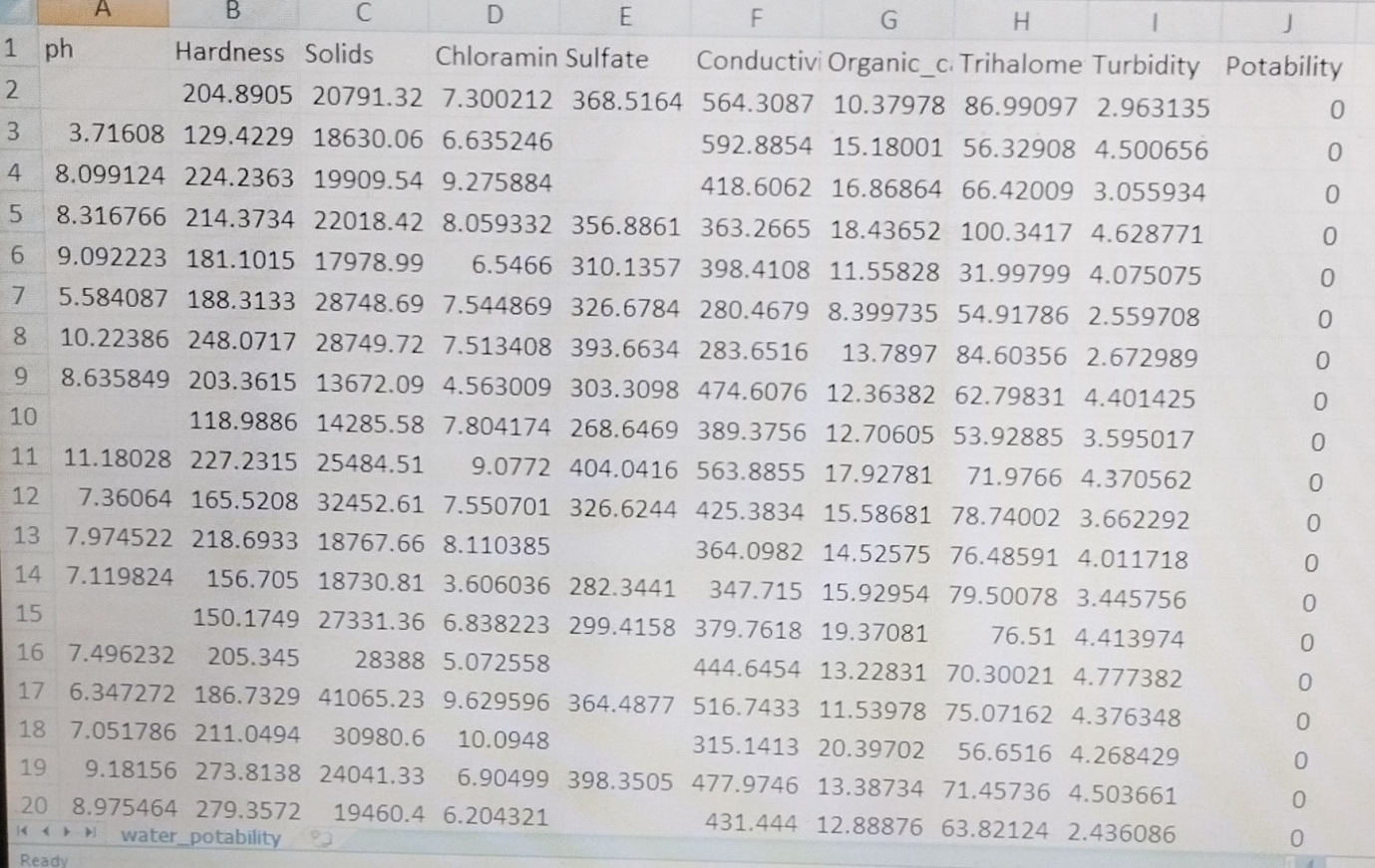
1. To measure concentration of the constituents in quantity for characterisation of water for different uses .

2.Of the various parameters in potable water few are objectionable even when present in very small quantity.

3. Others if only present in unusual quantities as to relegate the water from the potable to the unusable class.

4. The analyst familiar with water quality characterisation will often select parameters to be measured based on experience and intuition.

Given dataset:

****

3277 rows\*7 columns

**Overview of the process:**

1. Sample Collection:

* Select sampling sites that represent the area of interest.
* Use clean containers and equipment to collect water samples.
* Ensure proper labelling and documentation of sample location and time.

2. Preservation:

* Some samples may require immediate analysis, while others may need preservation to prevent changes in their properties. Common preservation methods include cooling, acidification, or adding specific chemicals.

3.Pre-processing:

* Filtration: Remove particulate matter using filters to obtain a clear liquid sample.
* Homogenization: Mix the sample to ensure a representative composition.

Physical Parameters Analysis:

* Temperature: Measure water temperature using a thermometer or a sensor.
* pH: Determine the acidity or alkalinity of the water with a pH meter.
* Turbidity: Assess water cloudiness or clarity using a turbid meter.

Chemical Parameters Analysis:

* Dissolved Oxygen (DO): Measure the concentration of oxygen in the water.
* Conductivity: Assess water's ability to conduct electrical current.
* Nutrients (e.g., nitrates, phosphates): Analyze the levels of essential nutrients.
* Heavy Metals: Test for the presence of potentially toxic elements like lead, mercury, and cadmium.
* Organic Contaminants: Detect organic compounds, such as pesticides or hydrocarbons.

Biological Parameters Analysis:

* Bacterial Analysis: Assess the presence of coli form bacteria or pathogens.
* Algal and Phytoplankton Analysis: Identify and quantify microscopic organisms.
* Macro invertebrates: Study aquatic insects, worms, and other macro invertebrates as indicators of water quality.

Data Interpretation:

* Compare the results to water quality standards, guidelines, or regulatory limits.
* Assess the implications for human health, aquatic ecosystems, and intended water uses.

Reporting:

* Prepare a detailed report summarizing the findings, methods used, and conclusions.
* Share the results with relevant authorities, stakeholders, or the public.

**Procedure:**

This test uses liquid reagent and a colour comparator. Dissolved Oxygen (DO): A measure of the amount of oxygen dissolved in water. This test uses a two-step procedure. In the first step, the sample is "fixed"; in the second, it is titrated to determine the level of DO in parts per million (ppm).

**Feature selection:**

class WaterQualityAnalyzer:

def \_init\_(self):

self.pH = 7.0 # Placeholder pH value

self.dissolved\_oxygen = 6.0 # Placeholder dissolved oxygen value

self.turbidity = 2.0 # Placeholder turbidity value

self.temperature = 25.0 # Placeholder temperature value

def collect\_data(self):

# In a real application, you would collect data from sensors.

# For this example, we'll use random values.

import random

self.pH = random.uniform(6.5, 8.5)

self.dissolved\_oxygen = random.uniform(4.0, 10.0)

self.turbidity = random.uniform(0.0, 5.0)

self.temperature = random.uniform(20.0, 30.0)

def analyze(self):

# Perform water quality analysis based on collected data.

if 6.5 <= self.pH <= 8.5:

pH\_status = "Optimal"

else:

pH\_status = "Suboptimal"

if 5.0 <= self.dissolved\_oxygen <= 9.0:

oxygen\_status = "Optimal"

else:

oxygen\_status = "Suboptimal"

if self.turbidity <= 2.0:

turbidity\_status = "Optimal"

else:

turbidity\_status = "Suboptimal"

return {

"pH": self.pH,

"pH\_status": pH\_status,

"Dissolved Oxygen (mg/L)": self.dissolved\_oxygen,

"Dissolved Oxygen\_status": oxygen\_status,

"Turbidity (NTU)": self.turbidity,

"Turbidity\_status": turbidity\_status,

"Temperature (°C)": self.temperature,

}

if \_name\_ == "\_main\_":

analyzer = WaterQualityAnalyzer()

analyzer.collect\_data()

result = analyzer.analyze()

print ("Water Quality Analysis Results:")

For key, value in result.items():

print (f"{key}: {value}")

**Model training:**

1. **Data Collection**:

Gather a comprehensive dataset of water quality measurements. This dataset should include various parameters such as pH, turbidity, dissolved oxygen, temperature, nutrient levels, and pollutants. Make sure the data is representative of the conditions you want to analyze.

2. **Data Pre-processing**:

Clean the data by handling missing values, outliers, and inconsistencies. Normalize or standardize the data to ensure that all parameters have the same scale, which is crucial for many machine learning algorithms.

Feature Selection: Identify the most relevant features (parameters) for your analysis. Some parameters may have more influence on water quality than others. Feature selection techniques can help you choose the most important variables.

3**. Model Selection**:

Choose an appropriate machine learning or statistical model for your analysis. Common models for water quality analysis include regression models (e.g., linear regression), decision trees, random forests, or deep learning models like neural networks.

4**. Training the Model**:

Split your dataset into a training set and a testing set to train and evaluate your model. Train the model using the training data, and tune hyper parameters to improve performance.

5. **Model Evaluation**:

Use metrics like Mean Absolute Error, Root Mean Squared Error, or others depending on the specific goals of your analysis to evaluate the model's performance. Cross-validation can help assess how well the model generalizes to new data.

6**. Interpretability**:

If your analysis requires understanding the factors influencing water quality, consider using interpretable models or techniques to explain the model's predictions. Explainable AI can be valuable in this context.

7**. Deployment**:

Once you have a trained and validated model, deploy it for real-world applications. This might involve integrating it into a monitoring system, a mobile app, or a web platform to provide real-time or periodic water quality predictions or alerts.

8. **Continuous Improvement**:

Regularly update and retrain the model as new data becomes available to ensure that it remains accurate and relevant.

Compliance and Regulations: Ensure that your model and its predictions comply with local and national water quality regulations and standards.

Remember that the success of your model depends on the quality of your data, the choice of the appropriate features, and the selection of a suitable machine learning algorithm. It's also essential to involve domain experts who understand water quality to guide the process and interpret the results effectively.

**Program Output:**

pH: 7.230141217189239

pH\_status: Optimal

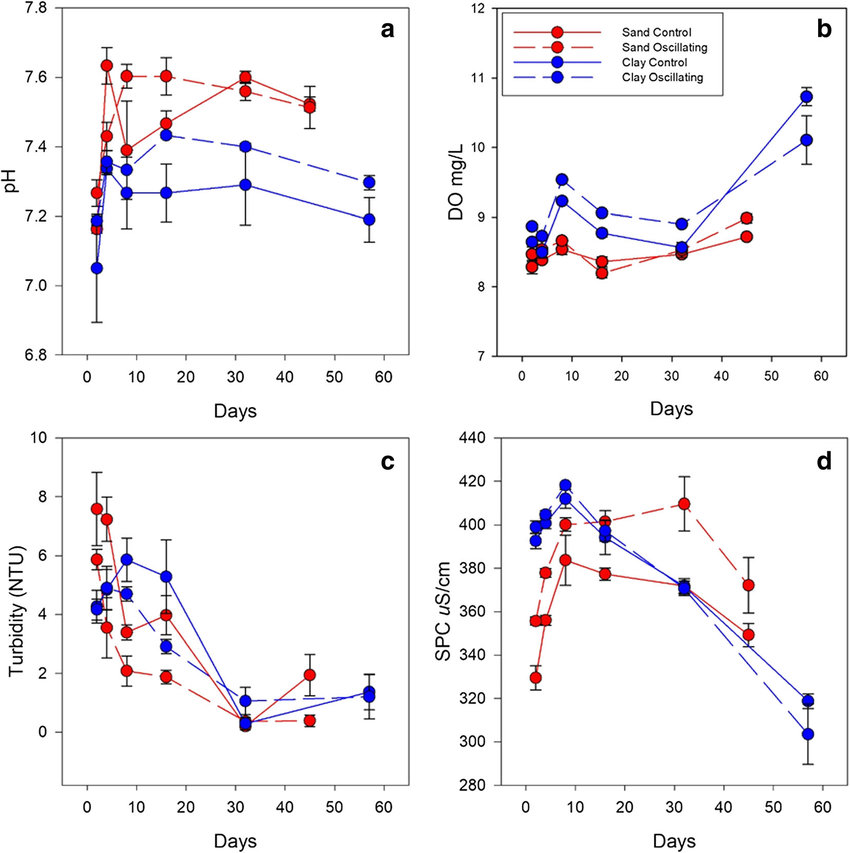
Dissolved Oxygen (mg/L): 8.27849742613547

Dissolved Oxygen status: Optimal

Turbidity (NTU): 3.421739774856105

Turbidity status: Suboptimal

Temperature (°C): 26.589718561447264"



**Feature Engineering:**

Feature engineering is a crucial aspect of water quality analysis, enabling the extraction of valuable insights from raw data. This process involves the creation of relevant features to better understand and model water quality parameters. Various techniques can be employed, including temporal features like hourly or seasonal averages, statistical measures to capture data distribution, and the calculation of lag variables or moving averages to identify trends and autocorrelations. Interaction features that explore relationships between different parameters, categorical encoding for location data and the incorporation of domain-specific information such as weather or land use can all enhance the analysis. Seasonal decomposition, frequency analysis, and time series features.