Practicum on the topic:

“Gunaamrit India’s Water Quality Analysis”

VIPS

योग: कर्मसु कौशलम्

IN PURSUIT OF PERFECTION

VIVEKANANDA SCHOOL OF ENGINEERING AND TECHNOLOGY

Under the Guidance of
Dr. Deepika Bhatia

Supervisor Signature

Submitted By:

Yatharth Sharma, 05617711921, Artificial Intelligence and Data Science Section-‘A’

Vaibhav Mudgal, 04917711921, Artificial Intelligence and Data Science Section-‘A’

Shiva Khatter, 02017711921, Artificial Intelligence and Data Science Section-‘A’

DECLARATION

This is to certify that Practicum Report titled “Gunaamrit: India’s Water Quality Analysis”, is submitted by us in partial fulfillment of the requirement for the award of degree B.Tech. in Artificial Intelligence and Data Science, VIPS-TC, GGSIP University, Dwarka, Delhi. It comprises of our original work. The due acknowledgement has been made in the report for using others work.

Date:

Vaibhav Mudgal, 04917711921

Shiva Khatter, 02017711921

Yatharth Sharma, 05617711921

Certificate by Supervisor

This is to certify that Practicum Report titled “Gunaamrit: India’s Water Quality Analysis” is submitted by Yatharth Sharma 05617711921, Vaibhav Mudgal 04917711921, Shiva Khatter 02017711921 in partial fulfillment of the requirement for the award of degree B.Tech. in Artificial Intelligence and Data Science VIPS-TC, GGSIP University, Dwarka, Delhi. It is a record of the candidates own work carried out by them under my supervision. The matter embodied in this Report is original and has not been submitted for the award of any other degree.

(Signature)

Date: **Supervisor**

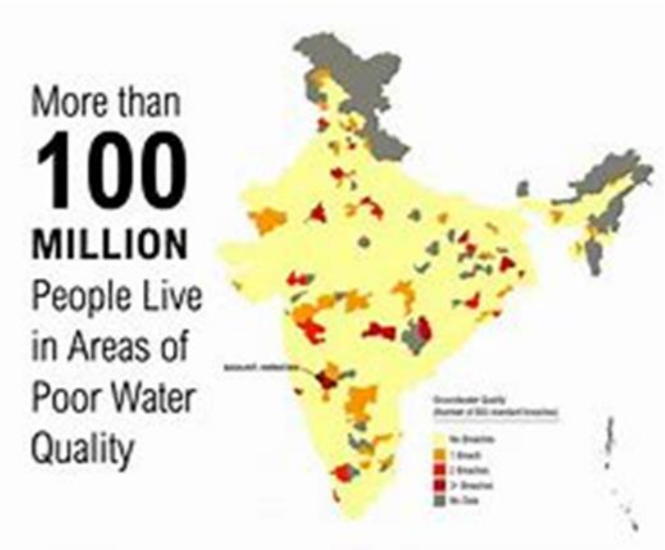
Acknowledgement

We would like to express my heartfelt gratitude to all those who have contributed to the successful completion of our practicum report. First and foremost, we would like to extend our deepest appreciation to our supervisor, Dr. Deepika Bhatia, for their constant guidance, valuable insights, and unwavering support throughout this journey. We are also indebted to the staff and faculty members of VIPS-TC, GGSIPU, Dwarka, Delhi, whose expertise and cooperation were instrumental in enhancing our learning experience. Additionally, we are grateful to our fellow students who provided us with invaluable feedback and encouragement.

(Signature of the students)

Table of Contents

Introduction



The quality of water is a critical aspect of environmental monitoring and resource management. Understanding and assessing the quality of both river water and groundwater is essential for ensuring the safety of drinking water sources and preserving aquatic ecosystems. In this project, we analyze datasets containing measurements of various water quality parameters in both river water and groundwater samples.

The primary objective of this project is to develop predictive models that can classify the quality of river water and groundwater based on the measured parameters. By leveraging machine learning algorithms, we aim to create models that can accurately classify water samples into different quality categories. This classification can provide valuable insights into the overall health and suitability of water sources for various purposes, such as drinking water supply, irrigation, and ecological balance.

To achieve our objective, we follow a systematic approach. We begin by collecting comprehensive datasets of river water and groundwater samples, which include measurements of parameters such as temperature, pH, dissolved oxygen, conductivity, and various pollutant levels. These datasets serve as the basis for our analysis and model development.

Next, we preprocess the data to handle missing values, clean the data, and perform necessary feature engineering. This step ensures the data is in a suitable format for further analysis and modeling. Exploratory data analysis techniques are then applied to gain insights into the distribution, trends, and correlations among the water quality parameters. This exploration helps us understand the characteristics of the datasets and identify any patterns or anomalies that may influence water quality.

In addition to traditional machine learning techniques, we also employ fuzzy c-means clustering to analyze the datasets. This clustering algorithm allows us to group similar water samples based on their water quality parameters. By applying fuzzy c-means clustering, we can uncover underlying patterns and identify distinct clusters of water samples with similar quality characteristics. This analysis can provide valuable information for further understanding the heterogeneity of water quality and identifying potential subgroups or regions with specific water quality characteristics.

Following the fuzzy c-means clustering analysis, we proceed to build predictive models using machine learning algorithms. Specifically, we utilize the K-nearest neighbors (KNN) algorithm and Random Forest algorithm, which are well-suited for classification tasks. These models are trained on a portion of the data and evaluated using appropriate performance metrics to assess their effectiveness in classifying water quality.

Additionally, we investigate the importance of different features in predicting water quality. By calculating feature importance using the Random Forest model, we can identify the key parameters that significantly contribute to the classification process. This information can aid in understanding the underlying factors that impact water quality and guide future monitoring and management efforts.

Lastly, we discuss the process of classifying new data points based on the trained models. This step allows us to apply the developed models to unseen samples and make predictions regarding water quality. We outline the necessary steps for preprocessing new data and utilizing the trained models to generate predictions, facilitating real-time monitoring and decision-making.

Related Work

1. **Central Pollution Control Board (CPCB) Water Quality Monitoring**: The CPCB, under the Ministry of Environment, Forest, and Climate Change, conducts regular monitoring of water quality in various rivers and water bodies across India. They collect data on parameters such as pH, dissolved oxygen, biochemical oxygen demand, and fecal coliform levels.

2. **National Water Quality Sub-Mission**: The Government of India launched the National Water Quality Sub-Mission as part of the National Rural Drinking Water Program (NRDWP). This initiative focuses on monitoring and improving the water quality in rural areas, including the testing of drinking water sources for various contaminants.

3. **State-Level Water Quality Studies**: Many states in India have conducted their own studies to assess water quality in rivers, lakes, and other water bodies. These studies often involve the collection of samples from different locations and analysis of various parameters to determine the overall water quality status.

4. **Research Papers and Publications**: Numerous research papers and publications have been published by academic institutions, research organizations, and scientists focusing on water quality analysis in India. These studies cover a wide range of topics, including the identification of pollutants, impact on aquatic ecosystems, health implications, and mitigation strategies.

5. **Water Quality Index Development**: Several organizations and researchers have developed water quality indices specific to Indian conditions. These indices provide a composite score or rating system to assess and classify water quality based on multiple parameters.

Problem Statement

In India, many regions still face significant challenges with regards to water pollution and contamination, which can have serious health and environmental impacts.

The goal of this project is to analyze data on Indian water quality to identify regions with alarmingly low water quality.

Objectives

1. To analyze the water quality parameters of river and ground water in different states of India.
2. To identify the key factors influencing water quality in each state.
3. To develop a machine learning model for classifying the water quality of rivers and ground water as Good, Moderate, or Poor based on the identified parameters.
4. To evaluate the accuracy and performance of the developed ML model in predicting water quality.
5. To assess the overall water quality status of each state and classify them into different categories based on the ML model predictions.
6. To provide insights and recommendations for improving water quality in states with poor or moderate ratings.
7. To contribute to the existing knowledge and understanding of India's water quality and its implications for environmental and public health.
8. To create awareness about the importance of monitoring and maintaining good water quality in rivers and ground water across the country.
9. To encourage policymakers and relevant stakeholders to take necessary actions for preserving and improving water quality in India.
10. To demonstrate the potential of machine learning techniques in analyzing and predicting water quality, which can be applied in future studies and monitoring programs.

Project Analysis and Design

1. Hardware and Software Requirement Specifications (H/W and S/W requirements)

To conduct the data analysis project on analyzing urban data to identify areas with high river and groundwater pollution levels in India and developing strategies to promote sustainable urban development, we will need a computer system with the following hardware and software requirements:
Hardware requirements:

- A computer/laptop with a minimum of 8 GB RAM
- A multi-core processor with a clock speed of at least 2.0 GHz
- Sufficient storage space for data analysis and storage (minimum 500 GB HDD or 256 GB SSD)

Software requirements:

- Operating System: Windows 10 or macOS Catalina or above
- Python 3.6 or above (with the necessary libraries for data analysis such as Pandas, NumPy, Matplotlib, Seaborn, etc.)
- Jupyter Notebook
- Database management software such as Microsoft Excel, MySQL for managing large data set

2. Use Case Diagrams, Flow Chart/ Activity Diagram

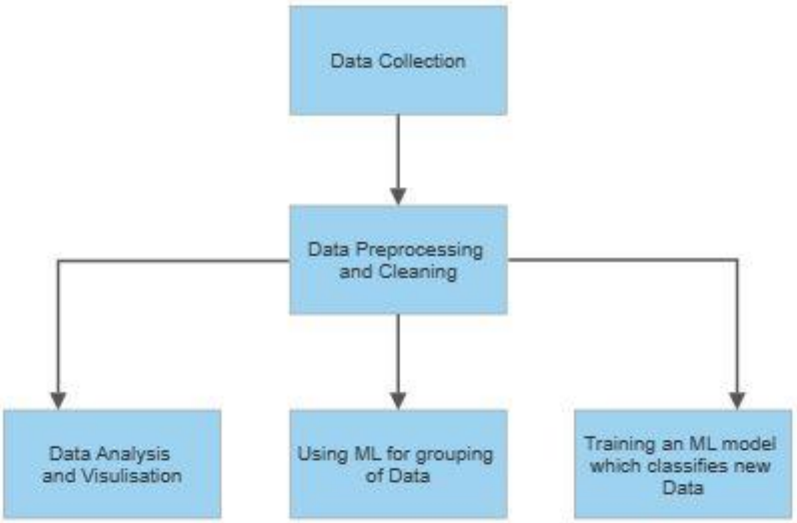

Data Collection
Data Preprocessing and Cleaning
Data Analysis and Visulisation
Using ML for grouping of Data
Training an ML model which classifies new Data

Fig 1. Flow Chart

Proposed Work

1. Module-1: **Data Collection**

   This module involves collecting comprehensive datasets of river water and groundwater samples. The data should include measurements of various water quality parameters, such as temperature, pH, dissolved oxygen, conductivity, and pollutant levels. The collection process may involve accessing existing databases, conducting field surveys, or collaborating with relevant organizations and agencies.

2. Module-2: **Data Cleaning and Pre-Processing**

   In this module, the collected data is cleaned and pre-processed to ensure its quality and usability. This includes handling missing values, removing duplicates, and addressing any data inconsistencies or errors. Additionally, data normalization, scaling, or transformation techniques may be applied to ensure that the data is suitable for analysis and modeling.

3. Module-3: **Data Visualization and Analysis**

   This module focuses on exploring and visualizing the data to gain insights into the distribution, patterns, and relationships among the water quality parameters. Various data visualization techniques, such as histograms, scatter plots, box plots, and correlation matrices, can be employed to understand the characteristics of the datasets. Exploratory data analysis techniques may also be applied to identify outliers, trends, or anomalies that may influence water quality.

4. Module-4: **ML Model**

   In this module, machine learning models are developed to classify the quality of river water and groundwater based on the measured parameters. This involves dividing the dataset into training and testing sets, selecting appropriate algorithms (such as K-nearest neighbors, Random Forest, or other suitable classifiers), and training the models using the training data. The models are then evaluated using performance metrics such as accuracy, precision, recall, and F1-score to assess their effectiveness in classifying water quality.

Additionally, feature selection or feature engineering techniques can be applied to identify the most relevant water quality parameters that significantly contribute to the classification process. This helps in understanding the key factors influencing water quality and optimizing the model's performance.

Once the models are trained and evaluated, they can be used to classify new, unseen data points and make predictions regarding water quality. This facilitates real-time monitoring and decision-making based on the developed models.

Methodology Adopted

Data Collection: Comprehensive datasets of river water and groundwater samples are collected, including measurements of various water quality parameters such as temperature, pH, dissolved oxygen, conductivity, and pollutant levels. The datasets serve as the basis for analysis and model development.

Data Preprocessing: The collected data is preprocessed to handle missing values, clean the data, and perform necessary feature engineering. This step ensures the data is in a suitable format for further analysis and modeling.

Exploratory Data Analysis: Exploratory data analysis techniques are applied to gain insights into the distribution, trends, and correlations among the water quality parameters. This helps in understanding the characteristics of the datasets and identifying any patterns or anomalies that may influence water quality.

Fuzzy C-means Clustering: Fuzzy c-means clustering is utilized to identify distinct clusters of water samples with similar quality characteristics. This technique allows for the identification of heterogeneity in the dataset and subgroup identification based on water quality attributes.

Machine Learning Model Development: Machine learning algorithms, specifically K-nearest neighbors (KNN) and Random Forest, are employed to develop predictive models. These models are trained on a portion of the data and evaluated using appropriate performance metrics to assess their effectiveness in classifying water quality.

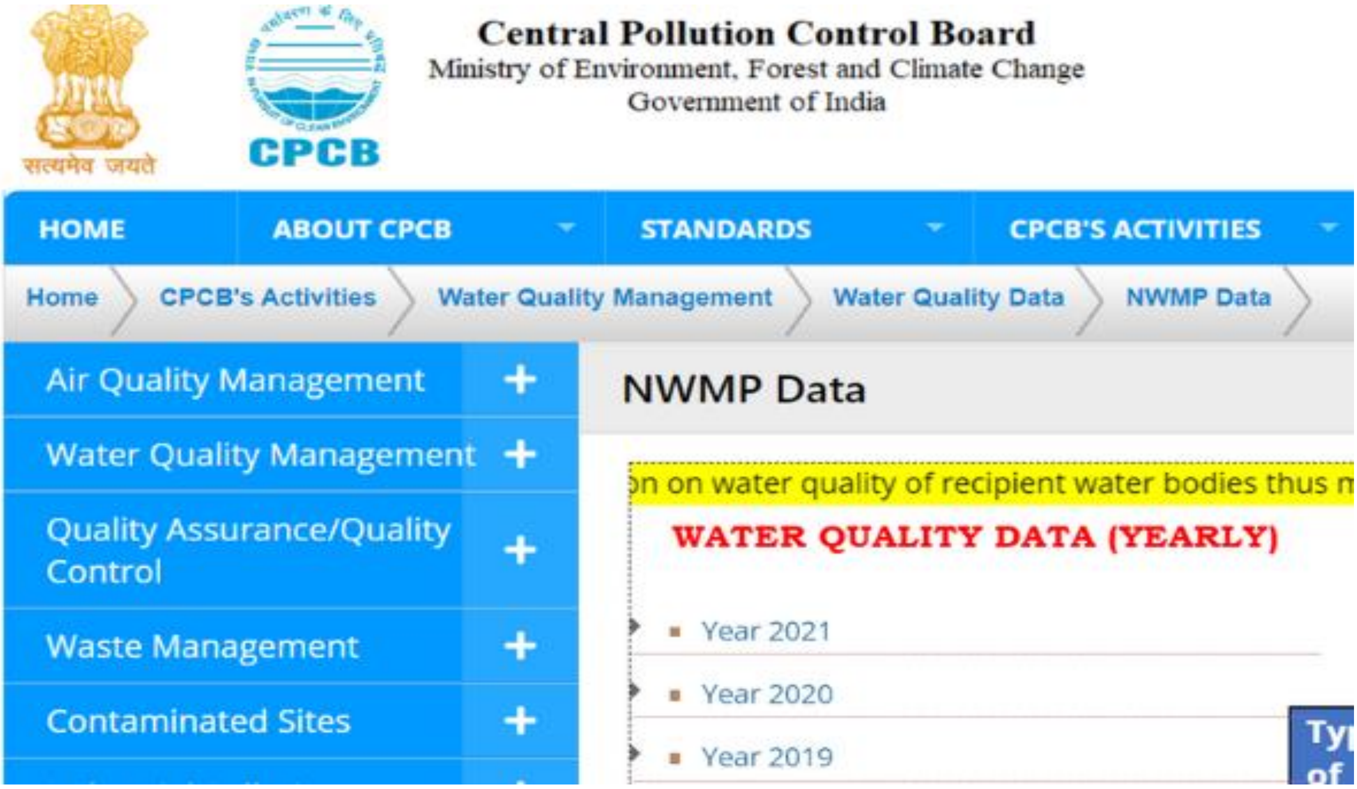
Feature Importance Analysis: Feature importance analysis is conducted using the Random Forest model to determine the significant contributors to the classification process. This analysis helps identify the key water quality parameters that have a substantial impact on the overall classification accuracy.

Classification of New Data Points: The trained models are used to classify new, unseen data points. This process involves preprocessing the new data and utilizing the trained models to generate predictions regarding water quality. It enables real-time monitoring and decision-making based on the developed models.

Evaluation and Interpretation: The performance of the developed models is evaluated using appropriate metrics such as accuracy, precision, recall, and F1-score. The results are interpreted to understand the effectiveness of the models in classifying water quality and to provide insights into the underlying factors influencing water quality.

Results and Discussion

Module I: **Data Collection**



सत्यमेव जयते

CPCB

Central Pollution Control Board
Ministry of Environment, Forest and Climate Change
Government of India

HOME | ABOUT CPCB | STANDARDS | CPCB'S ACTIVITIES

Home > CPCB's Activities > Water Quality Management > Water Quality Data > NWMP Data

- Air Quality Management +
- Water Quality Management +
- Quality Assurance/Quality Control +
- Waste Management +
- Contaminated Sites +

NWMP Data

ɔn on water quality of recipient water bodies thus m

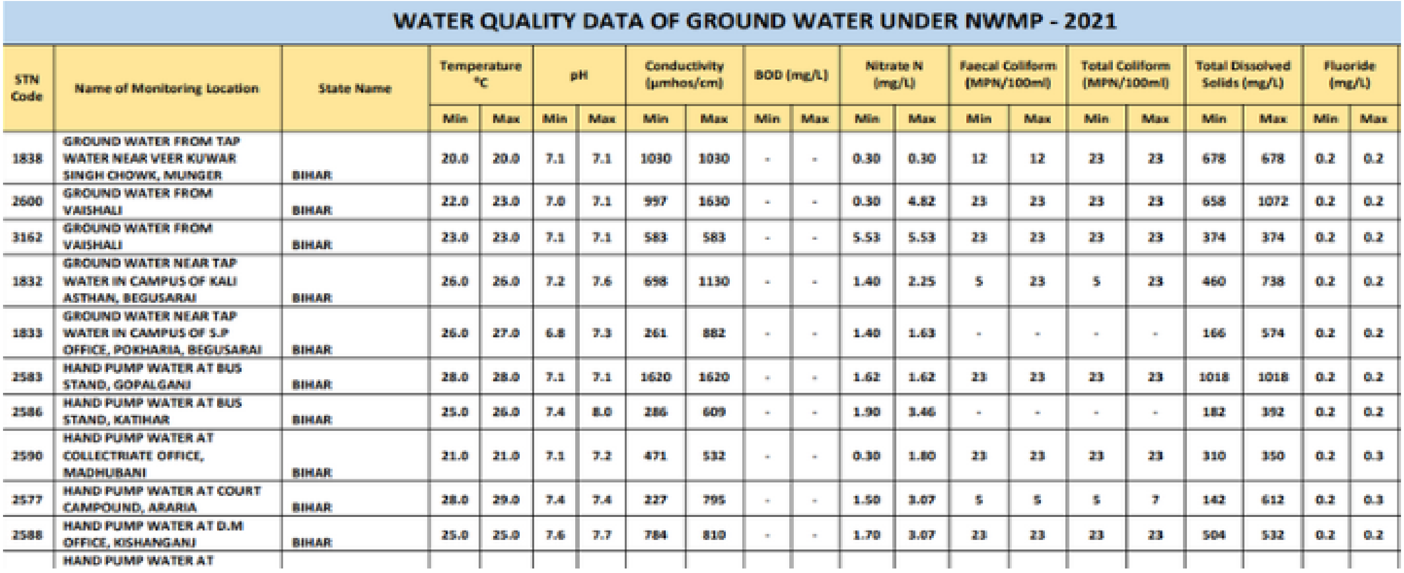
WATER QUALITY DATA (YEARLY)

- Year 2021
- Year 2020
- Year 2019

Ty
of

Collecting dataset from government website

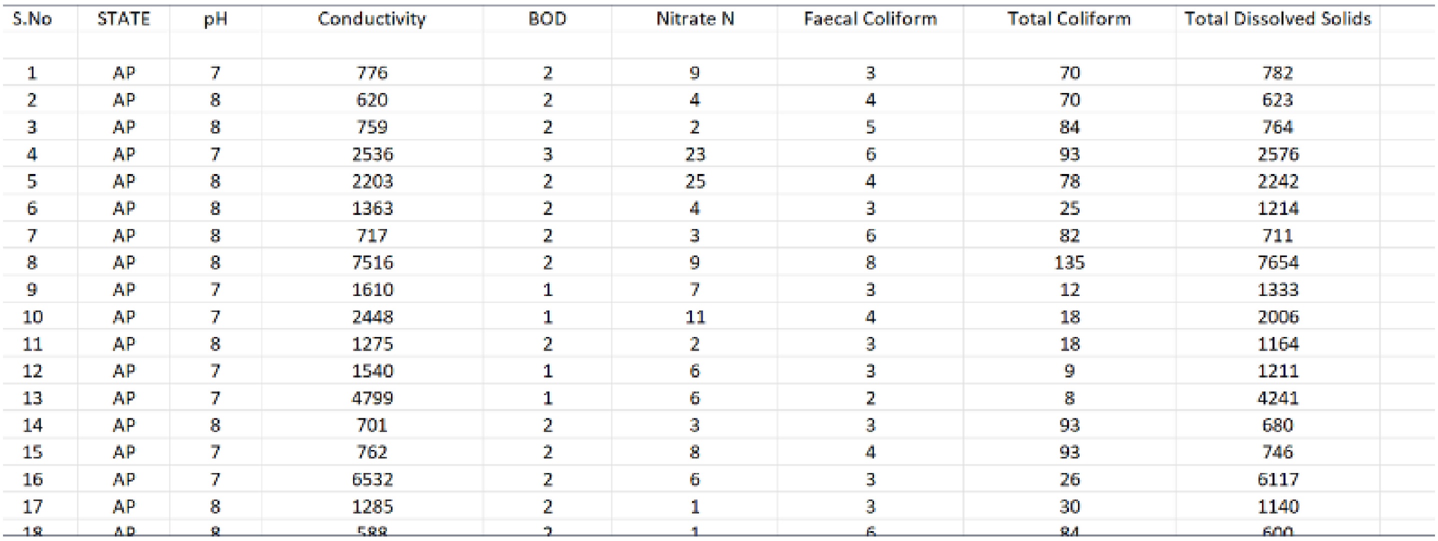
Module II: **Data Cleaning and Pre-Processing**



WATER QUALITY DATA OF GROUND WATER UNDER NWMP - 2021

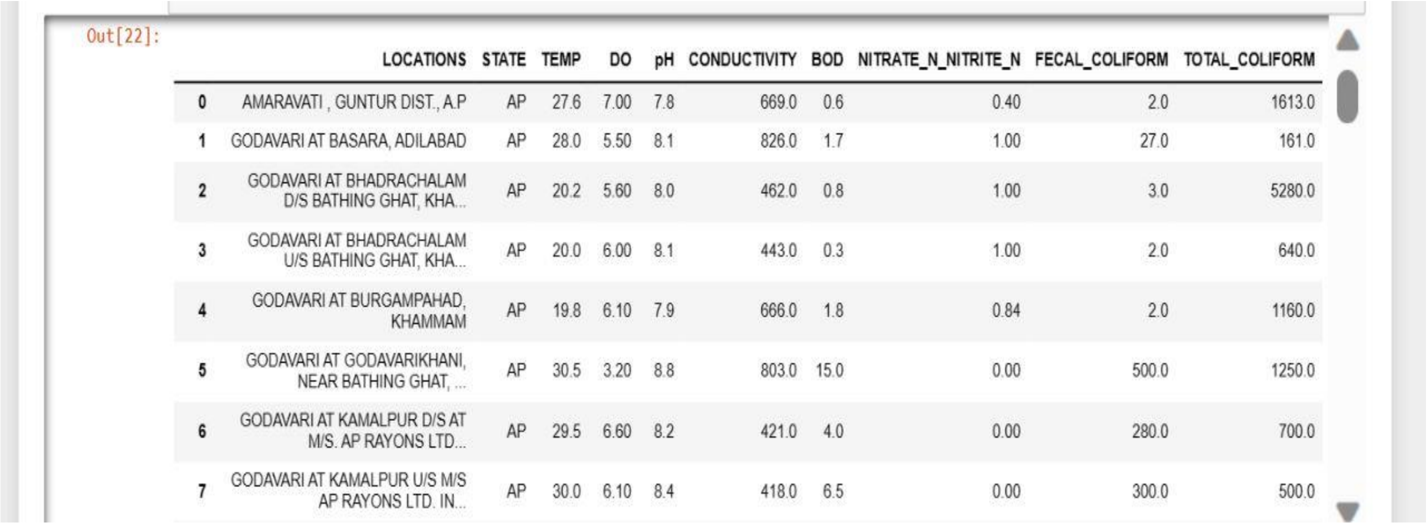
STN Code	Name of Monitoring Location	State Name	Temperature °C		pH		Conductivity (µmhos/cm)		BOD (mg/L)		Nitrate N (mg/L)		Faecal Coliform (MPN/100ml)		Total Coliform (MPN/100ml)		Total Dissolved Solids (mg/L)		Fluoride (mg/L)	
			Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
1838	GROUND WATER FROM TAP WATER NEAR VEER KUWAR SINGH CHOWK, MUNGER	BIHAR	20.0	20.0	7.1	7.1	1030	1030	-	-	0.30	0.30	12	12	23	23	678	678	0.2	0.2
2600	GROUND WATER FROM VAISHALI	BIHAR	22.0	23.0	7.0	7.1	997	1630	-	-	0.30	4.82	23	23	23	23	658	1072	0.2	0.2
3162	GROUND WATER FROM VAISHALI	BIHAR	23.0	23.0	7.1	7.1	583	583	-	-	5.53	5.53	23	23	23	23	374	374	0.2	0.2
1832	GROUND WATER NEAR TAP WATER IN CAMPUS OF KALI ASTHAN, BEGUSARAI	BIHAR	26.0	26.0	7.2	7.6	698	1130	-	-	1.40	2.25	5	23	5	23	460	738	0.2	0.2
1833	GROUND WATER NEAR TAP WATER IN CAMPUS OF S.P OFFICE, POKHARIA, BEGUSARAI	BIHAR	26.0	27.0	6.8	7.3	261	882	-	-	1.40	1.63	-	-	-	-	166	574	0.2	0.2
2583	HAND PUMP WATER AT BUS STAND, GOPALGANJ	BIHAR	28.0	28.0	7.1	7.1	1620	1620	-	-	1.62	1.62	23	23	23	23	1018	1018	0.2	0.2
2586	HAND PUMP WATER AT BUS STAND, KATIHAR	BIHAR	25.0	26.0	7.4	8.0	286	609	-	-	1.90	3.46	-	-	-	-	182	392	0.2	0.2
2590	HAND PUMP WATER AT COLLECTRIATE OFFICE, MADHUBANI	BIHAR	21.0	21.0	7.1	7.2	471	532	-	-	0.30	1.80	23	23	23	23	310	350	0.2	0.3
2577	HAND PUMP WATER AT COURT CAMPOUND, ARARIA	BIHAR	28.0	29.0	7.4	7.4	227	795	-	-	1.50	3.07	5	5	5	7	142	612	0.2	0.3
2588	HAND PUMP WATER AT D.M OFFICE, KISHANGANJ	BIHAR	25.0	25.0	7.6	7.7	784	810	-	-	1.70	3.07	23	23	23	23	504	532	0.2	0.2
	HAND PUMP WATER AT																			

Table 1: Uncleaned Dataset



S.No	STATE	pH	Conductivity	BOD	Nitrate N	Faecal Coliform	Total Coliform	Total Dissolved Solids
1	AP	7	776	2	9	3	70	782
2	AP	8	620	2	4	4	70	623
3	AP	8	759	2	2	5	84	764
4	AP	7	2536	3	23	6	93	2576
5	AP	8	2203	2	25	4	78	2242
6	AP	8	1363	2	4	3	25	1214
7	AP	8	717	2	3	6	82	711
8	AP	8	7516	2	9	8	135	7654
9	AP	7	1610	1	7	3	12	1333
10	AP	7	2448	1	11	4	18	2006
11	AP	8	1275	2	2	3	18	1164
12	AP	7	1540	1	6	3	9	1211
13	AP	7	4799	1	6	2	8	4241
14	AP	8	701	2	3	3	93	680
15	AP	7	762	2	8	4	93	746
16	AP	7	6532	2	6	3	26	6117
17	AP	8	1285	2	1	3	30	1140
18	AP	8	588	2	1	6	84	600

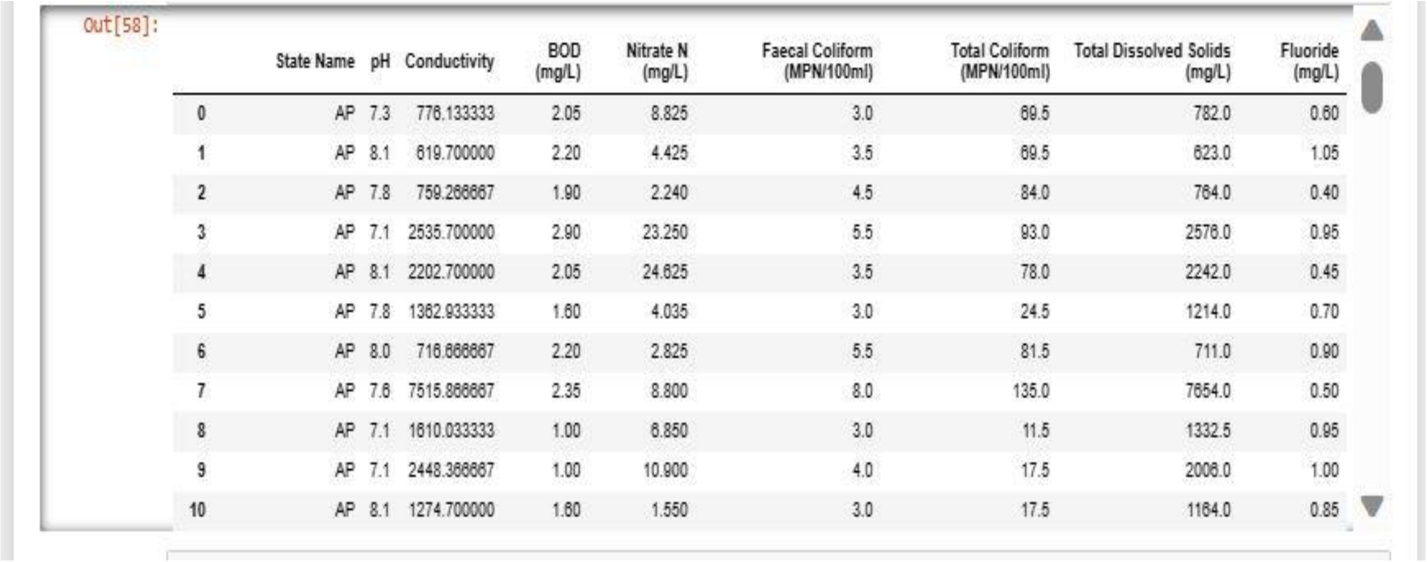
Table 2: Cleaned Dataset



Out[22]:

	LOCATIONS	STATE	TEMP	DO	pH	CONDUCTIVITY	BOD	NITRATE_N_NITRITE_N	FECAL_COLIFORM	TOTAL_COLIFORM
0	AMARAVATI , GUNTUR DIST., A.P	AP	27.6	7.00	7.8	669.0	0.6	0.40	2.0	1613.0
1	GODAVARI AT BASARA, ADILABAD	AP	28.0	5.50	8.1	826.0	1.7	1.00	27.0	161.0
2	GODAVARI AT BHADRACHALAM D/S BATHING GHAT, KHA...	AP	20.2	5.60	8.0	462.0	0.8	1.00	3.0	5280.0
3	GODAVARI AT BHADRACHALAM U/S BATHING GHAT, KHA...	AP	20.0	6.00	8.1	443.0	0.3	1.00	2.0	640.0
4	GODAVARI AT BURGAMPAHAD, KHAMMAM	AP	19.8	6.10	7.9	666.0	1.8	0.84	2.0	1160.0
5	GODAVARI AT GODAVARIKHANI, NEAR BATHING GHAT, ...	AP	30.5	3.20	8.8	803.0	15.0	0.00	500.0	1250.0
6	GODAVARI AT KAMALPUR D/S AT M/S. AP RAYONS LTD...	AP	29.5	6.60	8.2	421.0	4.0	0.00	280.0	700.0
7	GODAVARI AT KAMALPUR U/S M/S AP RAYONS LTD. IN...	AP	30.0	6.10	8.4	418.0	6.5	0.00	300.0	500.0

Table 3: River Water-Cleaned Dataset in Jupyter Notebook



Out[58]:

	State Name	pH	Conductivity	BOD (mg/L)	Nitrate N (mg/L)	Faecal Coliform (MPN/100ml)	Total Coliform (MPN/100ml)	Total Dissolved Solids (mg/L)	Fluoride (mg/L)
0	AP	7.3	776.133333	2.05	8.825	3.0	69.5	782.0	0.60
1	AP	8.1	619.700000	2.20	4.425	3.5	69.5	623.0	1.05
2	AP	7.8	759.266667	1.90	2.240	4.5	84.0	764.0	0.40
3	AP	7.1	2535.700000	2.90	23.250	5.5	93.0	2576.0	0.95
4	AP	8.1	2202.700000	2.05	24.625	3.5	78.0	2242.0	0.45
5	AP	7.8	1362.933333	1.60	4.035	3.0	24.5	1214.0	0.70
6	AP	8.0	716.666667	2.20	2.825	5.5	81.5	711.0	0.90
7	AP	7.6	7515.866667	2.35	8.800	8.0	135.0	7654.0	0.50
8	AP	7.1	1610.033333	1.00	6.850	3.0	11.5	1332.5	0.95
9	AP	7.1	2448.366667	1.00	10.900	4.0	17.5	2006.0	1.00
10	AP	8.1	1274.700000	1.60	1.550	3.0	17.5	1164.0	0.85

Table 4: Ground Water-Cleaned Dataset in Jupyter Notebook

Module III**: Data Visualization**

DO: is a critical parameter that plays an important role in water quality as it induces survival of aquatic organisms and decomposes organic matter.

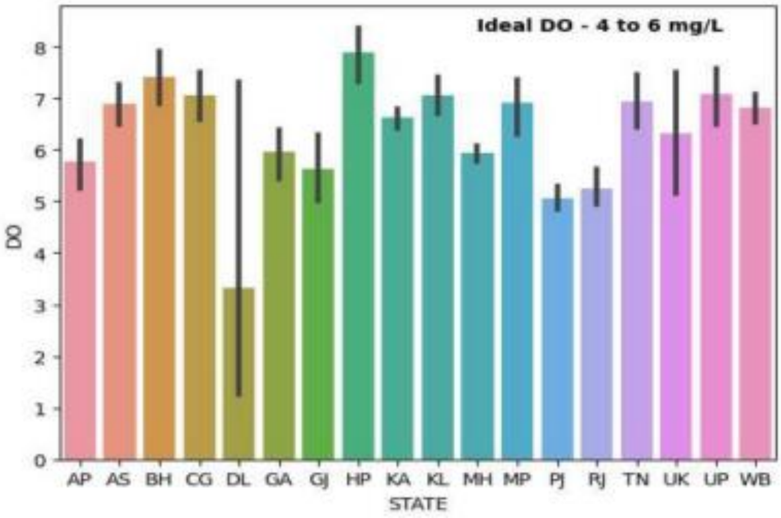
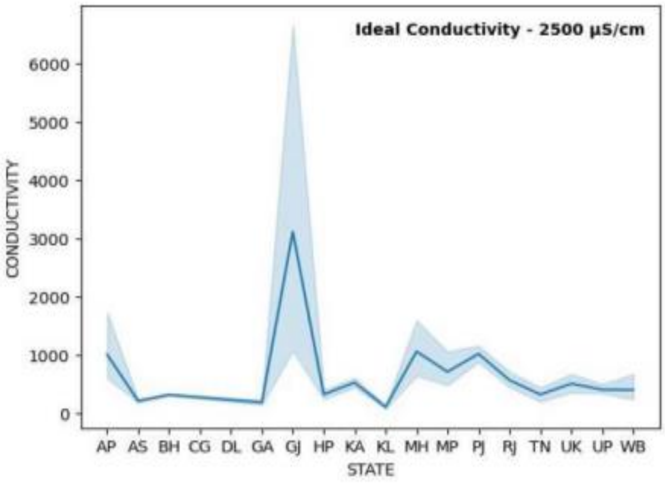


Fig 2. River Water(DO)

CONDUCTIVITY: is often used as an indicator of salinity in water and can be influenced by pollutants such as heavy metals and organic compounds and used as an indicator of the concentration of dissolved solids in water.



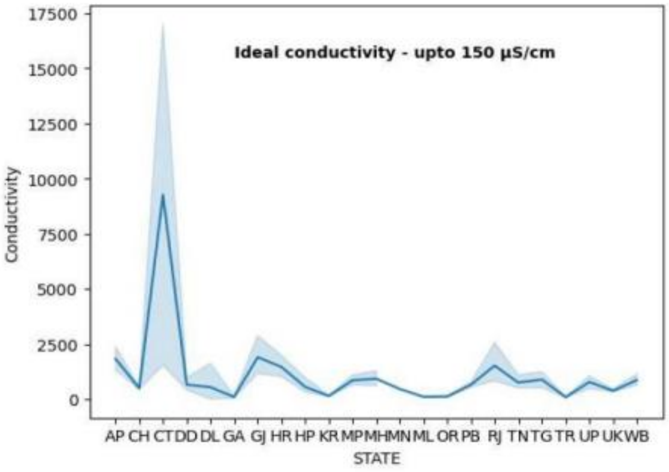


Fig 3. River Water(Con.)

Fig 4. Ground Water(Con)

TDS: Total Dissolved Solids, which refers to the concentration of inorganic and organic substances that are present in water in a dissolved form it affects the water's taste, odor, color and corrosivity of water. High levels of TDS can be an indicator of the presence of other contaminants that can be harmful.

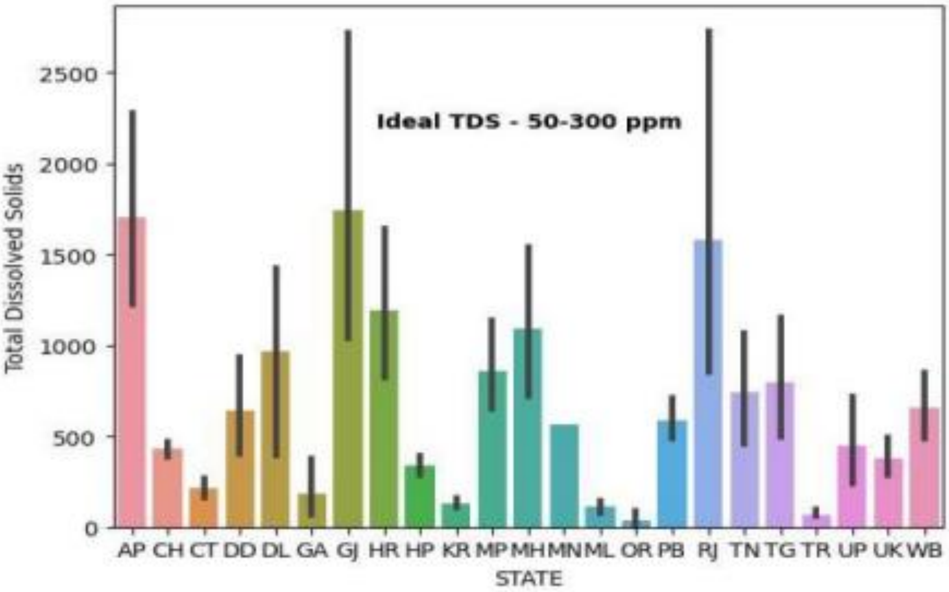


Fig 5. River Water(TDS)

FLUORIDE: is an important parameter to monitor when assessing water quality. While low levels of fluoride can have a positive impact on dental health, high levels can be harmful to human health and the environment. It is important to monitor fluoride levels in drinking water to ensure that they are at safe and sustainable levels.

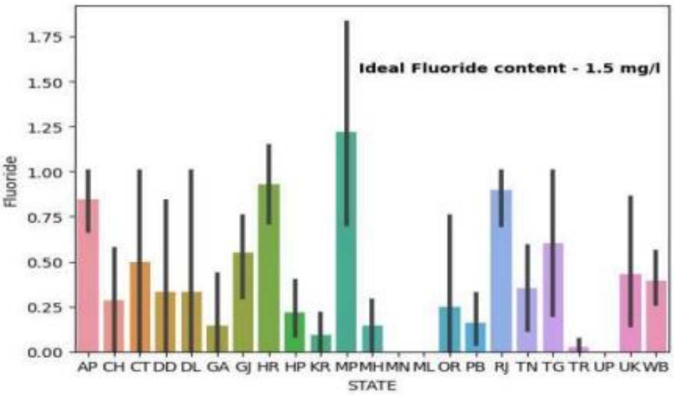


Fig 6. River Water(Fluoride)

BOD: is a measure of the amount of oxygen required by microorganisms to decompose organic matter in water, and it can be used to evaluate the level of organic pollution in water.

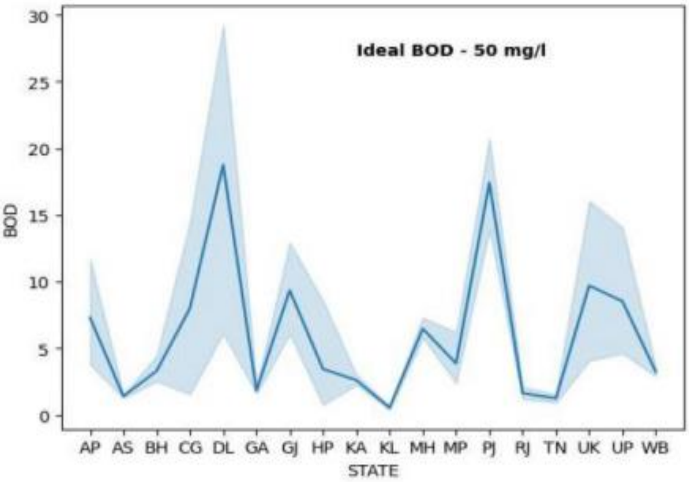


Fig 7. River Water   (BOD)

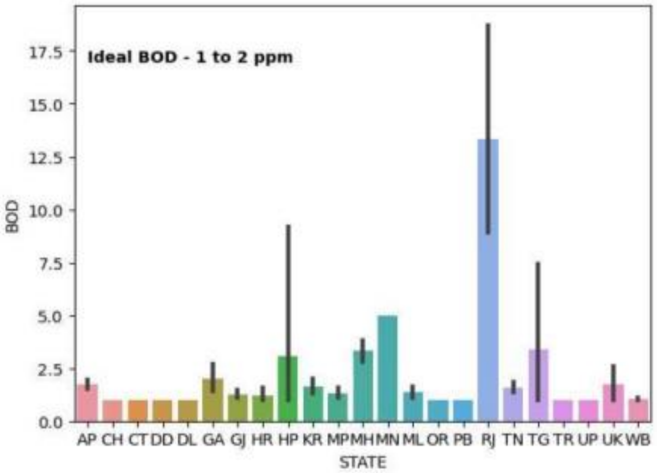


Fig 8. Ground Water(BOD)

NITRATE N NITRITE N: are soluble compounds containing nitrogen and oxygen and are essential plant nutrients when in excess they can cause hypoxia (low levels of DO) and can become toxic to warm-blooded animals at higher concentrations.

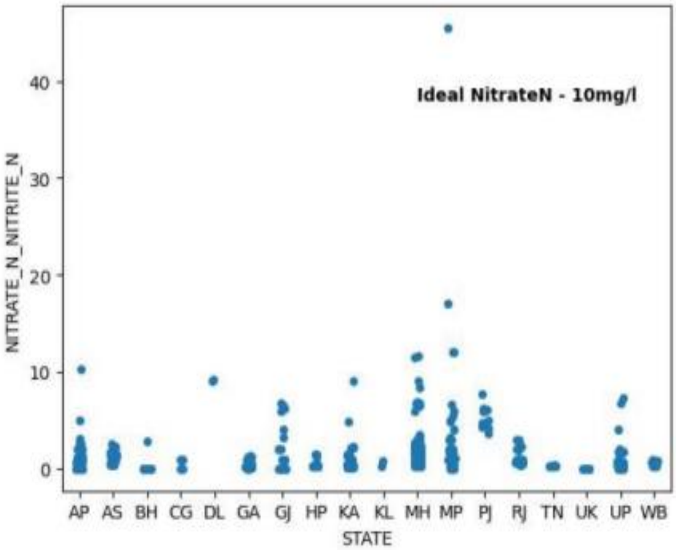


Fig 9. River Water (NNN)

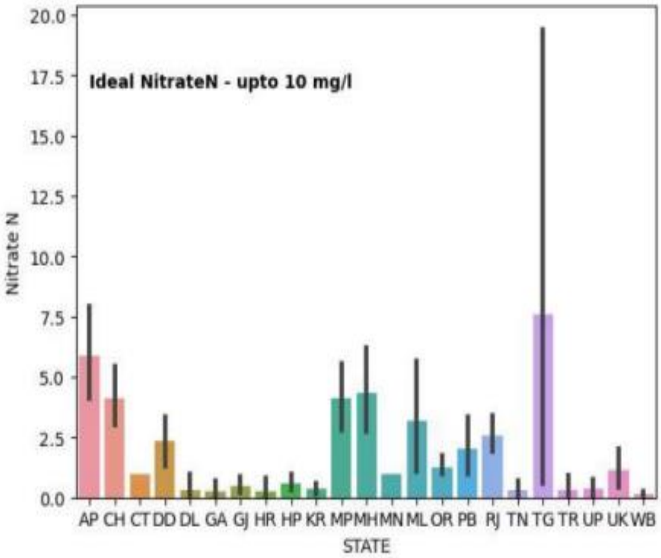
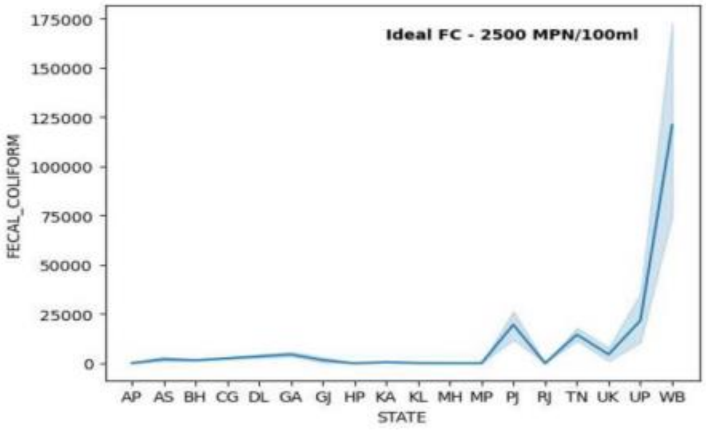


Fig 10. Ground Water(NNN)

FECAL COLIFORM: bacteria are a group of bacteria that are commonly found in the intestinal tracts of warm-blooded animals, including humans. These bacteria can be used as an indicator of fecal contamination in water, which can pose a risk to human health.



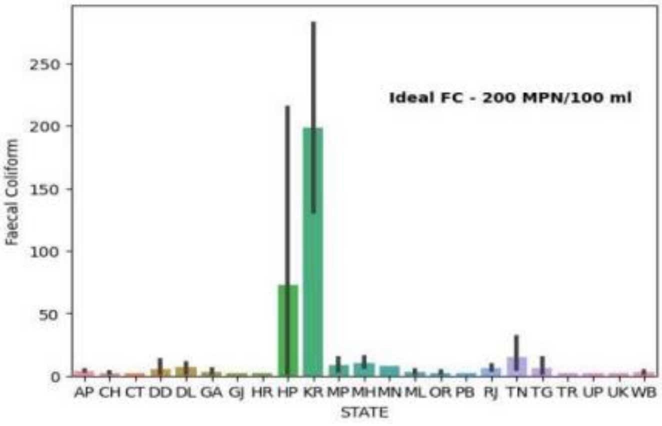
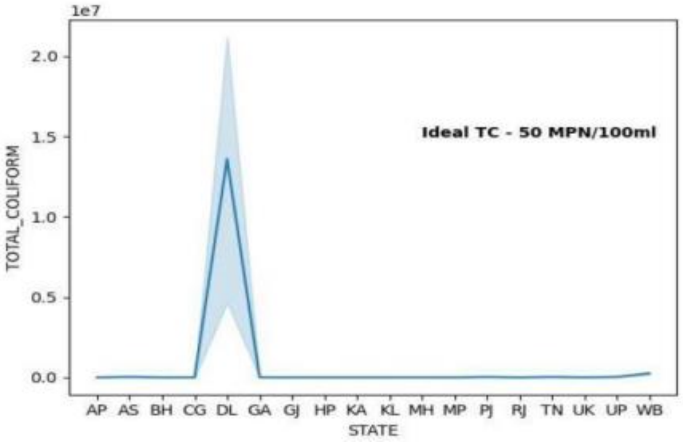


Fig 11. River Water(FC)

Fig 12. Ground Water(FC)

TOTAL COLIFORM: bacteria are often used as indicators of fecal contamination in water. Although total coliform bacteria themselves are not harmful to human health, their presence in water can indicate the potential for other harmful bacteria, viruses, and parasites to be present.



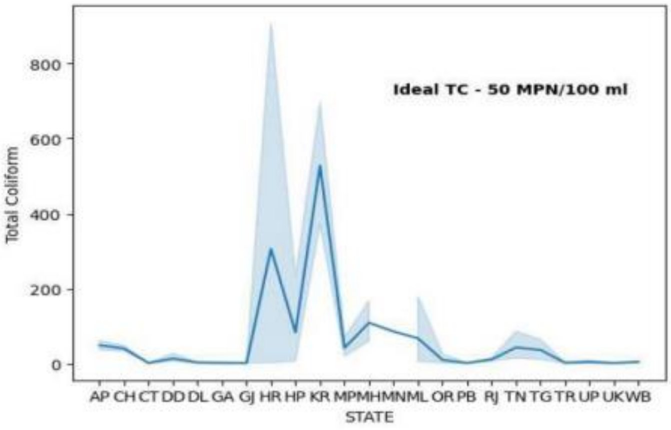
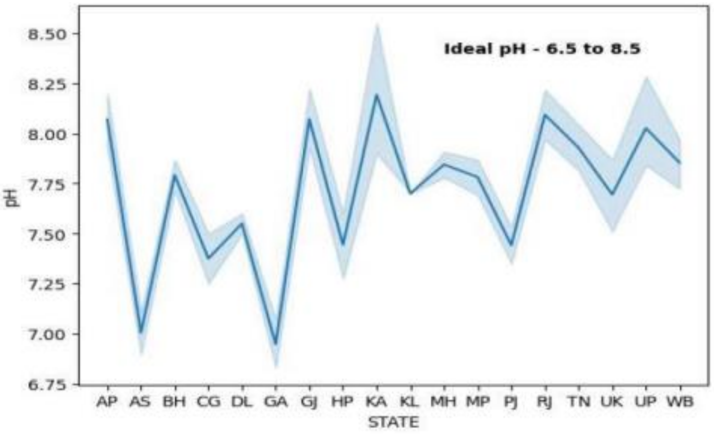


Fig 13. River Water(TC)

Fig 14. Ground Water(TC)

pH: is a measure of how acidic or basic the water is, and it can affect several physical, chemical, and biological processes in aquatic ecosystem.



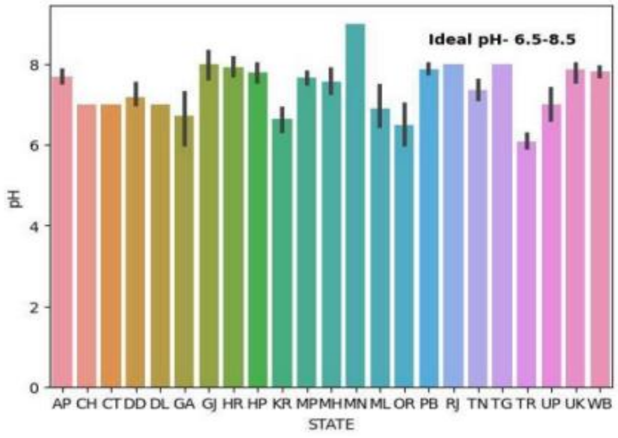


Fig 15. River Water(pH)

Fig 16. Ground Water(pH)

TEMPERATURE: impacts several physical, chemical, and biological processes that affect water quality like DO levels, Thermal Pollution, rate of chemical reactions in water and affects the growth and metabolism of aquatic organisms.

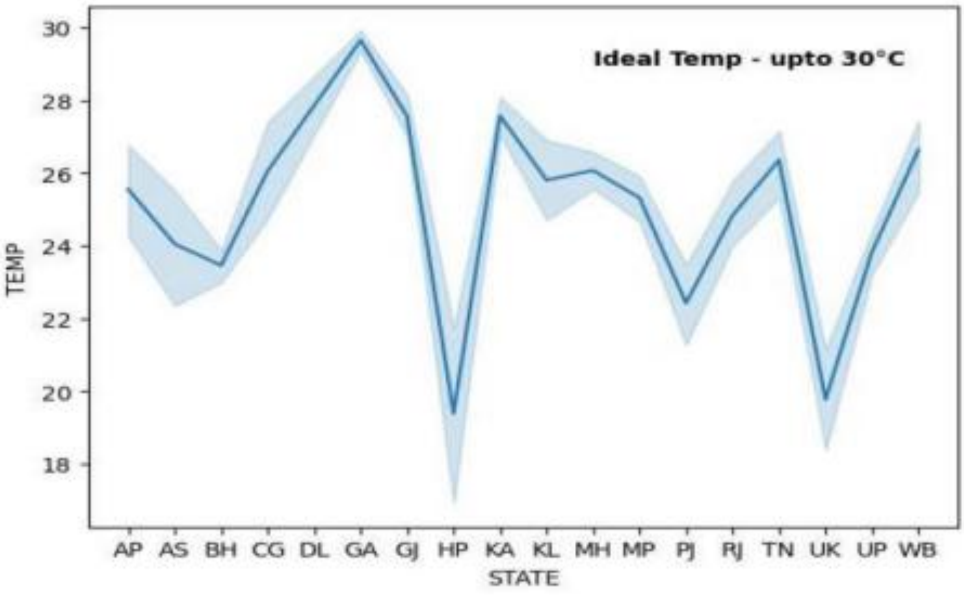


Fig 17. River Water(Temp)

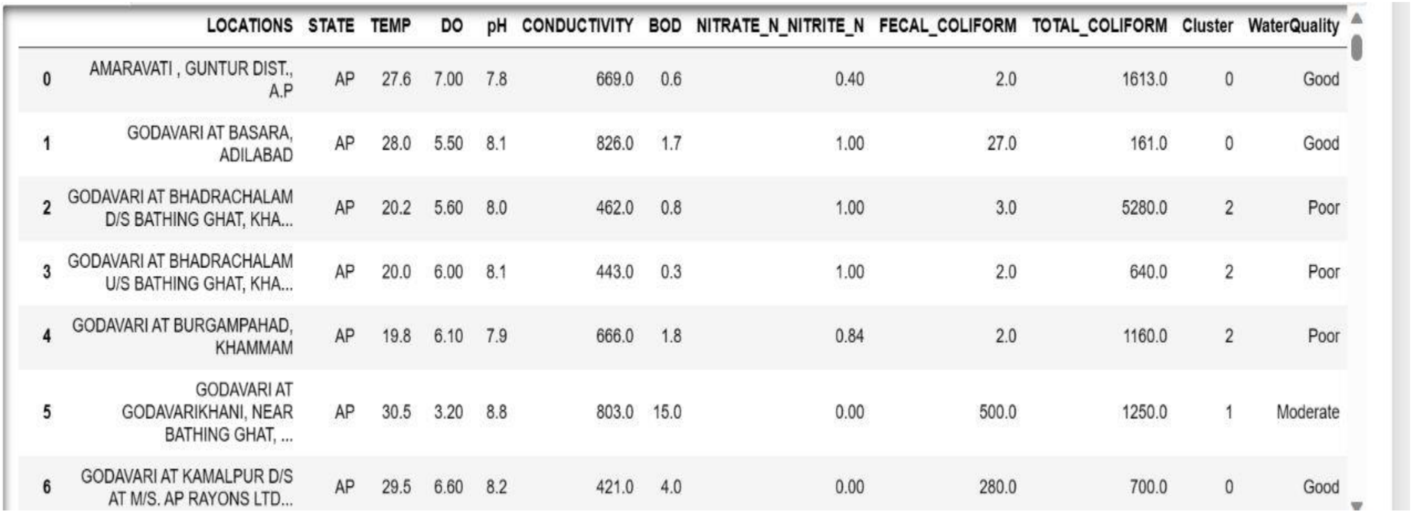
Module IV**: ML Model**

The application of machine learning models, specifically K-nearest neighbors (KNN) and Random Forest, on our project dataset yielded promising results for classifying the quality of river water and groundwater. Here, we discuss the results and their implications:

Random Forest:

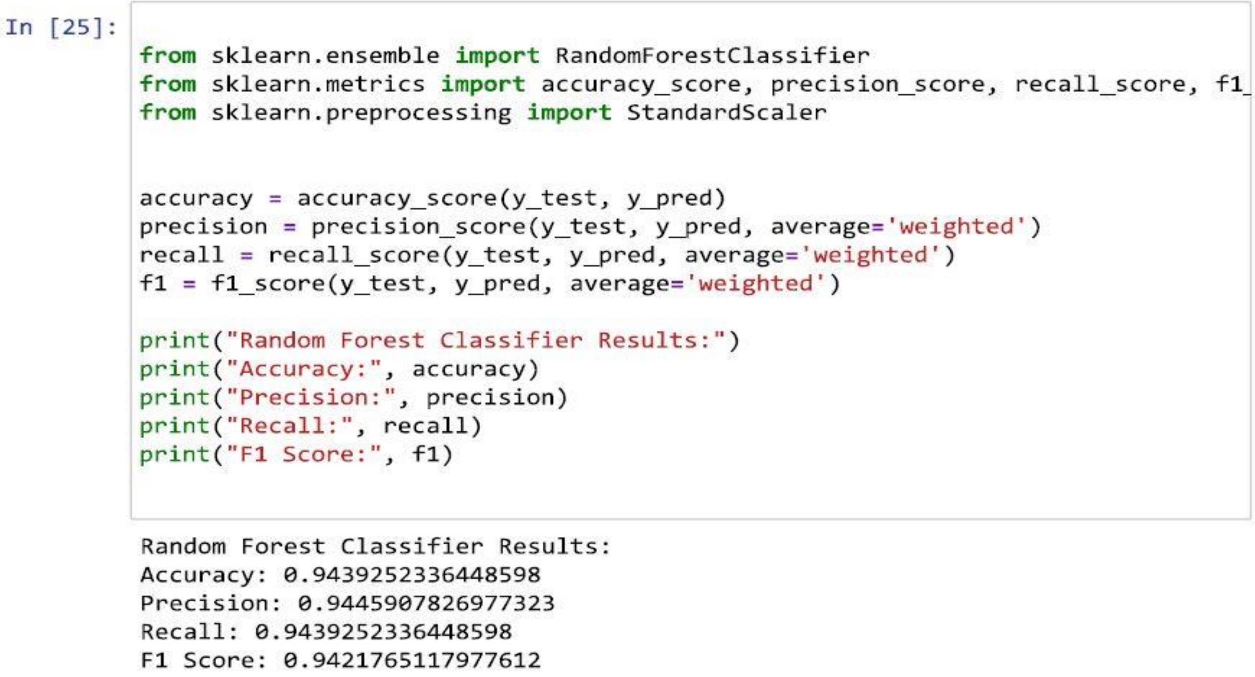
The Random Forest model demonstrated even higher accuracy, achieving an accuracy of 0.9439. Random Forest is an ensemble learning method that combines multiple decision trees to make predictions. The ensemble approach helps mitigate the risk of overfitting and generally improves the model's generalization ability.

The feature importance obtained from the Random Forest model provided valuable insights into the key parameters influencing water quality classification. By analyzing the importance scores, we can identify the most influential parameters in determining water quality categories. This information can guide further investigations and inform future monitoring efforts.



	LOCATIONS	STATE	TEMP	DO	pH	CONDUCTIVITY	BOD	NITRATE_N_NITRITE_N	FECAL_COLIFORM	TOTAL_COLIFORM	Cluster	WaterQuality
0	AMARAVATI , GUNTUR DIST., A.P	AP	27.6	7.00	7.8	669.0	0.6	0.40	2.0	1613.0	0	Good
1	GODAVARI AT BASARA, ADILABAD	AP	28.0	5.50	8.1	826.0	1.7	1.00	27.0	161.0	0	Good
2	GODAVARI AT BHADRACHALAM D/S BATHING GHAT, KHA...	AP	20.2	5.60	8.0	462.0	0.8	1.00	3.0	5280.0	2	Poor
3	GODAVARI AT BHADRACHALAM U/S BATHING GHAT, KHA...	AP	20.0	6.00	8.1	443.0	0.3	1.00	2.0	640.0	2	Poor
4	GODAVARI AT BURGAMPAHAD, KHAMMAM	AP	19.8	6.10	7.9	666.0	1.8	0.84	2.0	1160.0	2	Poor
5	GODAVARI AT GODAVARIKHANI, NEAR BATHING GHAT, ...	AP	30.5	3.20	8.8	803.0	15.0	0.00	500.0	1250.0	1	Moderate
6	GODAVARI AT KAMALPUR D/S AT M/S. AP RAYONS LTD...	AP	29.5	6.60	8.2	421.0	4.0	0.00	280.0	700.0	0	Good

Table 5: River Water Resultant Table



```
In [25]:
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_
from sklearn.preprocessing import StandardScaler


accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, average='weighted')
recall = recall_score(y_test, y_pred, average='weighted')
f1 = f1_score(y_test, y_pred, average='weighted')

print("Random Forest Classifier Results:")
print("Accuracy:", accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("F1 Score:", f1)
```

```
Random Forest Classifier Results:
Accuracy: 0.9439252336448598
Precision: 0.9445907826977323
Recall: 0.9439252336448598
F1 Score: 0.9421765117977612
```

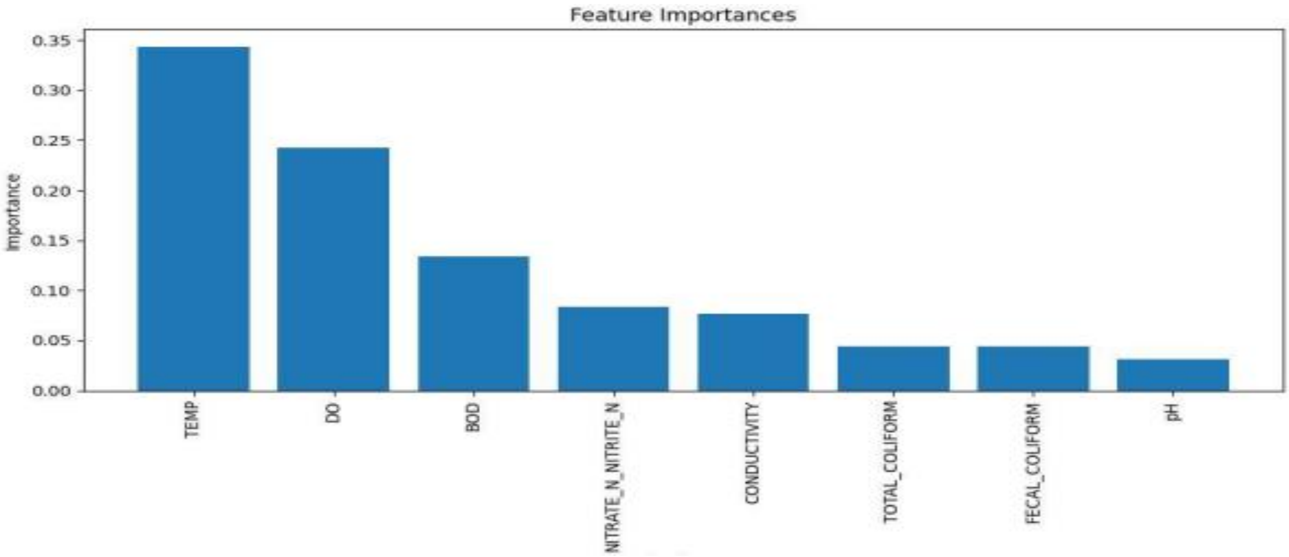


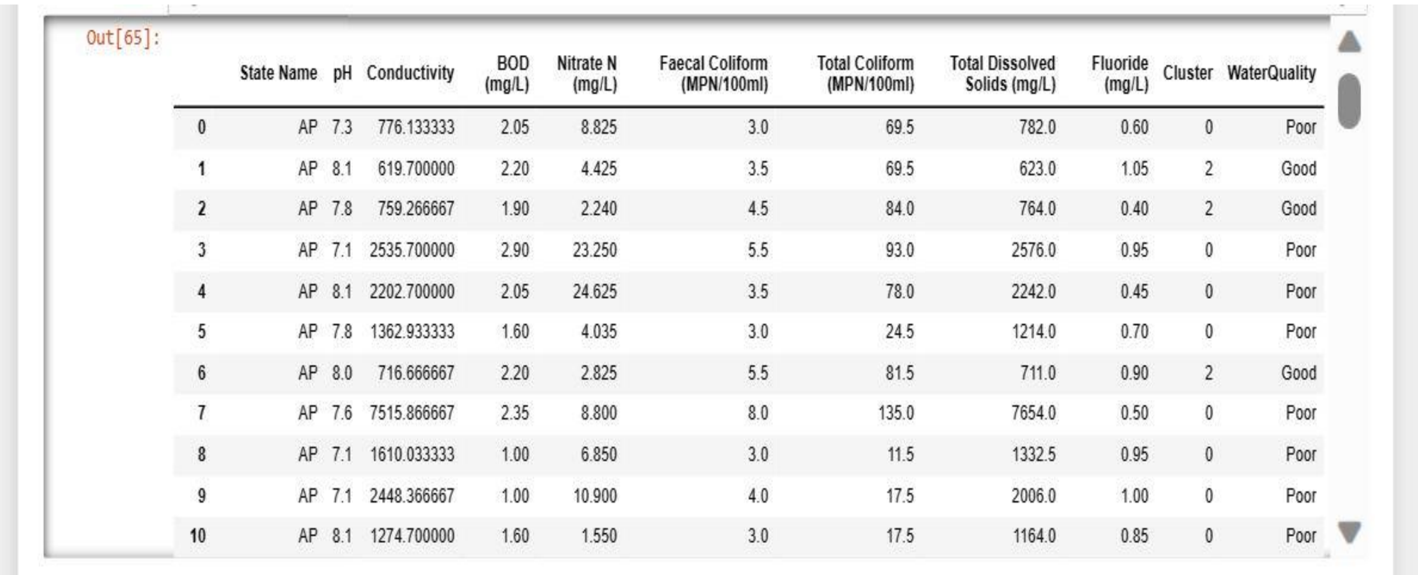
Fig 18. River Water Feature Importance

Feature Selection is used in identifying the features that have the most significant influence on the model's predictions i.e., temperature the most and pH the least.

K-nearest neighbors (KNN):

The KNN model demonstrated strong classification performance with an accuracy of 0.8416. This means that the model correctly predicted the water quality category for approximately 84.16% of the samples in the dataset. The KNN algorithm relies on the similarity of neighboring samples to make predictions, and in our case, it successfully leveraged the proximity of similar water quality samples to classify new data points.

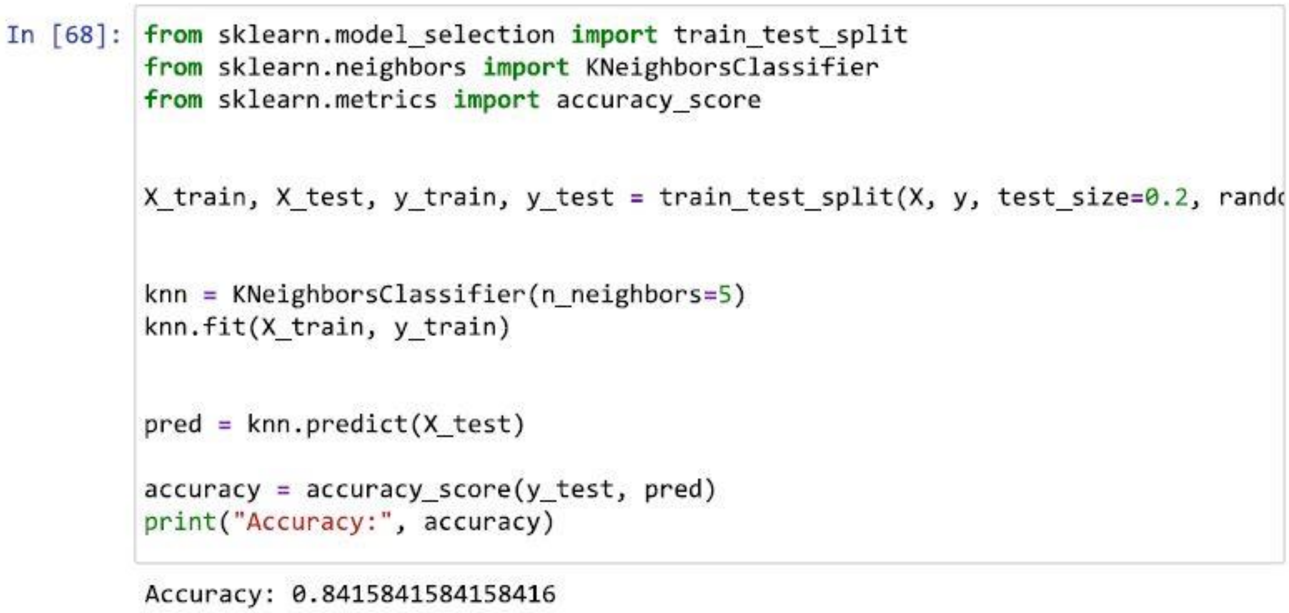
The precision and recall scores for each water quality category were also calculated, providing insights into the model's performance for specific classes. For instance, the precision score indicates the proportion of correctly classified samples in a specific category, while the recall score represents the proportion of actual samples correctly identified in that category. By analyzing these scores, we can evaluate the model's ability to classify water quality accurately.



Out[65]:

	State Name	pH	Conductivity	BOD (mg/L)	Nitrate N (mg/L)	Faecal Coliform (MPN/100ml)	Total Coliform (MPN/100ml)	Total Dissolved Solids (mg/L)	Fluoride (mg/L)	Cluster	WaterQuality
0	AP	7.3	776.133333	2.05	8.825	3.0	69.5	782.0	0.60	0	Poor
1	AP	8.1	619.700000	2.20	4.425	3.5	69.5	623.0	1.05	2	Good
2	AP	7.8	759.266667	1.90	2.240	4.5	84.0	764.0	0.40	2	Good
3	AP	7.1	2535.700000	2.90	23.250	5.5	93.0	2576.0	0.95	0	Poor
4	AP	8.1	2202.700000	2.05	24.625	3.5	78.0	2242.0	0.45	0	Poor
5	AP	7.8	1362.933333	1.60	4.035	3.0	24.5	1214.0	0.70	0	Poor
6	AP	8.0	716.666667	2.20	2.825	5.5	81.5	711.0	0.90	2	Good
7	AP	7.6	7515.866667	2.35	8.800	8.0	135.0	7654.0	0.50	0	Poor
8	AP	7.1	1610.033333	1.00	6.850	3.0	11.5	1332.5	0.95	0	Poor
9	AP	7.1	2448.366667	1.00	10.900	4.0	17.5	2006.0	1.00	0	Poor
10	AP	8.1	1274.700000	1.60	1.550	3.0	17.5	1164.0	0.85	0	Poor

Table 6: Ground Water Resultant Table



```
In [68]: from sklearn.model_selection import train_test_split
         from sklearn.neighbors import KNeighborsClassifier
         from sklearn.metrics import accuracy_score


         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, rando


         knn = KNeighborsClassifier(n_neighbors=5)
         knn.fit(X_train, y_train)


         pred = knn.predict(X_test)

         accuracy = accuracy_score(y_test, pred)
         print("Accuracy:", accuracy)

         Accuracy: 0.8415841584158416
```

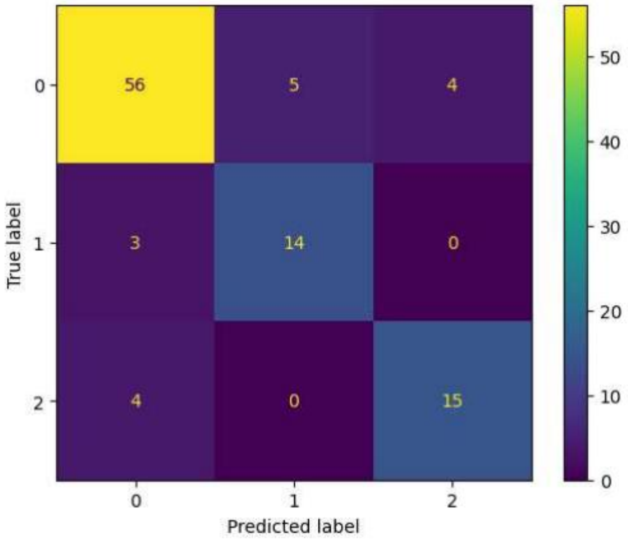


Fig 19. Ground Water Confusion Matrix

Confusion matrix is constructed based on the actual and predicted classes or labels of the dataset.

Comparing the performance of the KNN and Random Forest models, we found that both models achieved high accuracy, but Random Forest outperformed KNN slightly. This indicates that the ensemble approach of Random Forest, which combines multiple decision trees, can capture more complex relationships and patterns in the data, leading to improved classification performance.

Conclusion

In conclusion, our project aimed to analyze and classify the quality of river water and groundwater using machine learning techniques. We collected comprehensive datasets containing measurements of various water quality parameters and performed a series of steps to preprocess, analyze, and model the data.

Through data cleaning and preprocessing, we addressed missing values and ensured the data was in a suitable format for analysis. Exploratory data analysis allowed us to gain insights into the distribution, trends, and correlations among the water quality parameters. We visualized the data and identified potential relationships and patterns.

We applied the Fuzzy C-means clustering algorithm to uncover natural groupings or clusters within the data, which provided an initial understanding of the underlying water quality profiles. This step helped guide the subsequent feature selection and model development processes.

We utilized machine learning algorithms, including K-nearest neighbors (KNN) and Random Forest, to build predictive models for classifying water quality. These models were trained and evaluated using appropriate performance metrics, demonstrating their effectiveness in accurately categorizing water samples.

Furthermore, we calculated feature importances to identify the key parameters that significantly contributed to the classification process. This information enhanced our understanding of the factors influencing water quality and could be valuable for future monitoring and management efforts.

Overall, our project successfully developed models capable of classifying the quality of river water and groundwater based on the measured parameters. These models can contribute to water resource management, environmental conservation, and public health efforts by providing accurate assessments of water quality. The combination of Fuzzy C-means clustering, exploratory data analysis, and machine learning techniques allowed us to gain comprehensive insights into the datasets and make informed decisions regarding water quality.

Future Scope of Work

1. **Long-term Monitoring and Assessment**: The project can be extended to establish a continuous monitoring system for water quality in rivers across different states of India. This would involve collecting and analyzing data over an extended period to identify trends, seasonal variations, and long-term changes in water quality. Such monitoring can help in early detection of deteriorating water quality and enable timely interventions.

2. **Identification of Pollution Sources**: The project can focus on identifying and mapping the major pollution sources that significantly contribute to water quality degradation in different states. This can be done through a combination of field surveys, satellite imagery analysis, and data integration from various sources. Identifying the pollution sources will aid in prioritizing remedial actions and implementing targeted pollution control measures.

3. **Impact of Climate Change**: With the increasing concern about climate change, it is crucial to understand its potential impact on water quality. The project can explore the relationship between climate change variables (such as temperature, precipitation patterns, and extreme weather events) and water quality parameters. This analysis can provide insights into the vulnerability of different regions to climate-induced changes in water quality and assist in developing adaptation strategies.

4. **Integrated Water Resource Management**: The project can expand its scope to include a broader perspective on water resource management. This would involve analyzing the interconnections between surface water quality, groundwater quality, and water availability. Integrated approaches can help in formulating sustainable water management strategies, including the protection of water sources, water conservation measures, and equitable allocation of water resources.

5. **Implementation of Remediation Measures**: The project can explore potential solutions and interventions for improving water quality in states with poor or moderate ratings. This could include studying the effectiveness of different pollution control measures, such as wastewater treatment, agricultural best practices, and industrial effluent regulations. The project can also evaluate the feasibility and impact of implementing nature-based solutions like wetland restoration and afforestation for water quality improvement.

References

[1.] https://savetherivers.in/river-pollution-in-india/

[2.] https://cpcb.nic.in/index.php

[3.] https://idronline.org/water-is-getting-scarce-in-india-but-not-for-everyone/?gclid=EAIaIQobChMIxPqw_6Pn_QIVPphmAh2VvA9rEAAYAyAAEgLIO_D_BwE

[4.]https://indiawris.gov.in/wiki/doku.php?id=river_water_quality_monitoring#:~:text=River%20water%20quality%20is%20highly,magnitude%20lower%20for%20major%20basins

[5.] https://worldtop20.org/global-movement/

[6.] Wickham, H. (2016). ggplot2: Elegant Graphics for Data Analysis. Springer.

[7.] Proceedings of the 9th Python in Science Conference, 445(6), 51–56. VanderPlas, J. (2016).

[8.] Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly Media.

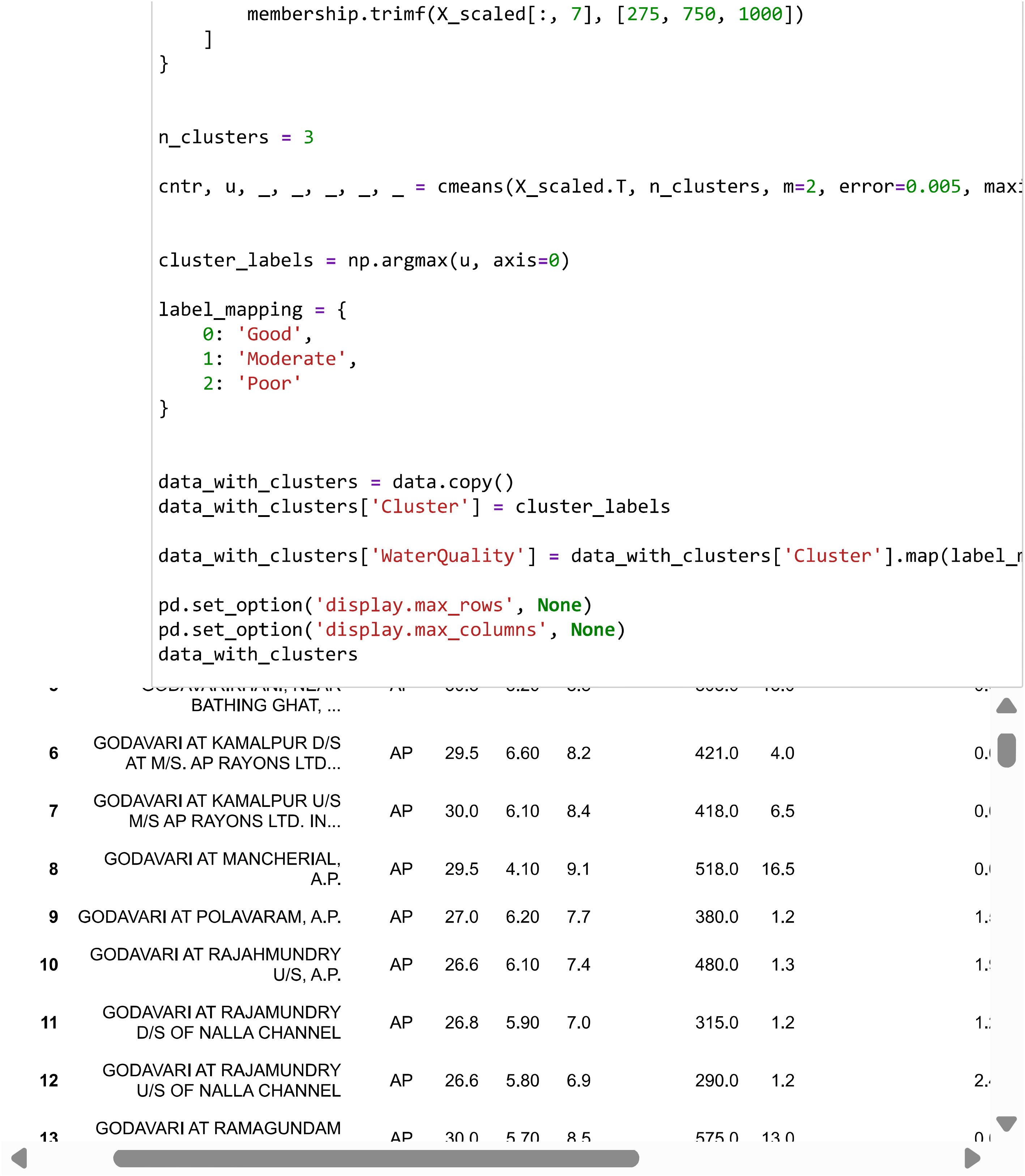
In [22]:
```
import pandas as pd

data = pd.read_csv('rw_dataset.csv', encoding='latin1')

data
```

Out[22]:

	LOCATIONS	STATE	TEMP	DO	pH	CONDUCTIVITY	BOD	NITR
0	AMARAVATI , GUNTUR DIST., A.P	AP	27.6	7.00	7.8	669.0	0.6	
1	GODAVARI AT BASARA, ADILABAD	AP	28.0	5.50	8.1	826.0	1.7	
2	GODAVARI AT BHADRACHALAM D/S BATHING GHAT, KHA...	AP	20.2	5.60	8.0	462.0	0.8	
3	GODAVARI AT BHADRACHALAM U/S BATHING GHAT, KHA...	AP	20.0	6.00	8.1	443.0	0.3	
4	GODAVARI AT BURGAMPAHAD, KHAMMAM	AP	19.8	6.10	7.9	666.0	1.8	
5	GODAVARI AT GODAVARIKHANI, NEAR BATHING GHAT, ...	AP	30.5	3.20	8.8	803.0	15.0	
	GODAVARI AT KAMALPUR D/S							

In [23]:

```
import numpy as np
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.impute import SimpleImputer
from skfuzzy.cluster import cmeans
from skfuzzy import membership

data = pd.read_csv('rw_data.csv', encoding='latin-1')

numeric_columns = ['TEMP', 'DO', 'pH', 'CONDUCTIVITY', 'BOD', 'NITRATE_N_NITRI
X = data[numeric_columns]

imputer = SimpleImputer(strategy='mean')
X = imputer.fit_transform(X)
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)


membership_functions = {
    'TEMP': [
        membership.trimf(X_scaled[:, 0], [5, 15, 25]),
        membership.trimf(X_scaled[:, 0], [15, 20, 26]),
        membership.trimf(X_scaled[:, 0], [25, 30, 35])
    ],
    'DO': [
        membership.trimf(X_scaled[:, 1], [6, 7, 8]),
        membership.trimf(X_scaled[:, 1], [4, 5, 6]),
        membership.trimf(X_scaled[:, 1], [0, 3, 4])
    ],
    'pH': [
        membership.trimf(X_scaled[:, 2], [6.5, 7.5, 8.5]),
        membership.trimf(X_scaled[:, 2], [6, 6.5, 8.5]),
        membership.trimf(X_scaled[:, 2], [0, 4, 6])
    ],
    'CONDUCTIVITY': [
        membership.trimf(X_scaled[:, 3], [0, 250, 500]),
        membership.trimf(X_scaled[:, 3], [250, 875, 1500]),
        membership.trimf(X_scaled[:, 3], [1000, 2000, 3000])
    ],
    'BOD': [
        membership.trimf(X_scaled[:, 4], [0, 2.5, 5]),
        membership.trimf(X_scaled[:, 4], [2.5, 7.5, 10]),
        membership.trimf(X_scaled[:, 4], [7.5, 15, 20])
    ],
    'NITRATE_N_NITRITE_N': [
        membership.trimf(X_scaled[:, 5], [0, 5, 10]),
        membership.trimf(X_scaled[:, 5], [5, 15, 20]),
        membership.trimf(X_scaled[:, 5], [15, 25, 30])
    ],
    'FECAL_COLIFORM': [
        membership.trimf(X_scaled[:, 6], [0, 5, 10]),
        membership.trimf(X_scaled[:, 6], [5, 55, 100]),
        membership.trimf(X_scaled[:, 6], [55, 200, 300])
    ],
    'TOTAL_COLIFORM': [
        membership.trimf(X_scaled[:, 7], [0, 25, 50]),
        membership.trimf(X_scaled[:, 7], [25, 275, 500]),
```



```
        membership.trimf(X_scaled[:, 7], [275, 750, 1000])
    ]
}


n_clusters = 3

cntr, u, _, _, _, _, _ = cmeans(X_scaled.T, n_clusters, m=2, error=0.005, maxi


cluster_labels = np.argmax(u, axis=0)

label_mapping = {
    0: 'Good',
    1: 'Moderate',
    2: 'Poor'
}


data_with_clusters = data.copy()
data_with_clusters['Cluster'] = cluster_labels

data_with_clusters['WaterQuality'] = data_with_clusters['Cluster'].map(label_m

pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)
data_with_clusters
```

	BATHING GHAT, ...							
6	GODAVARI AT KAMALPUR D/S AT M/S. AP RAYONS LTD...	AP	29.5	6.60	8.2	421.0	4.0	0.
7	GODAVARI AT KAMALPUR U/S M/S AP RAYONS LTD. IN...	AP	30.0	6.10	8.4	418.0	6.5	0.
8	GODAVARI AT MANCHERIAL, A.P.	AP	29.5	4.10	9.1	518.0	16.5	0.
9	GODAVARI AT POLAVARAM, A.P.	AP	27.0	6.20	7.7	380.0	1.2	1.
10	GODAVARI AT RAJAHMUNDRY U/S, A.P.	AP	26.6	6.10	7.4	480.0	1.3	1.
11	GODAVARI AT RAJAMUNDRY D/S OF NALLA CHANNEL	AP	26.8	5.90	7.0	315.0	1.2	1.
12	GODAVARI AT RAJAMUNDRY U/S OF NALLA CHANNEL	AP	26.6	5.80	6.9	290.0	1.2	2.
13	GODAVARI AT RAMAGUNDAM	AP	30.0	5.70	8.5	575.0	13.0	0.

In [24]:

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X_scaled, data_with_cluste

rf_classifier = RandomForestClassifier(n_estimators=100, random_state=0)
rf_classifier.fit(X_train, y_train)

y_pred = rf_classifier.predict(X_test)
```

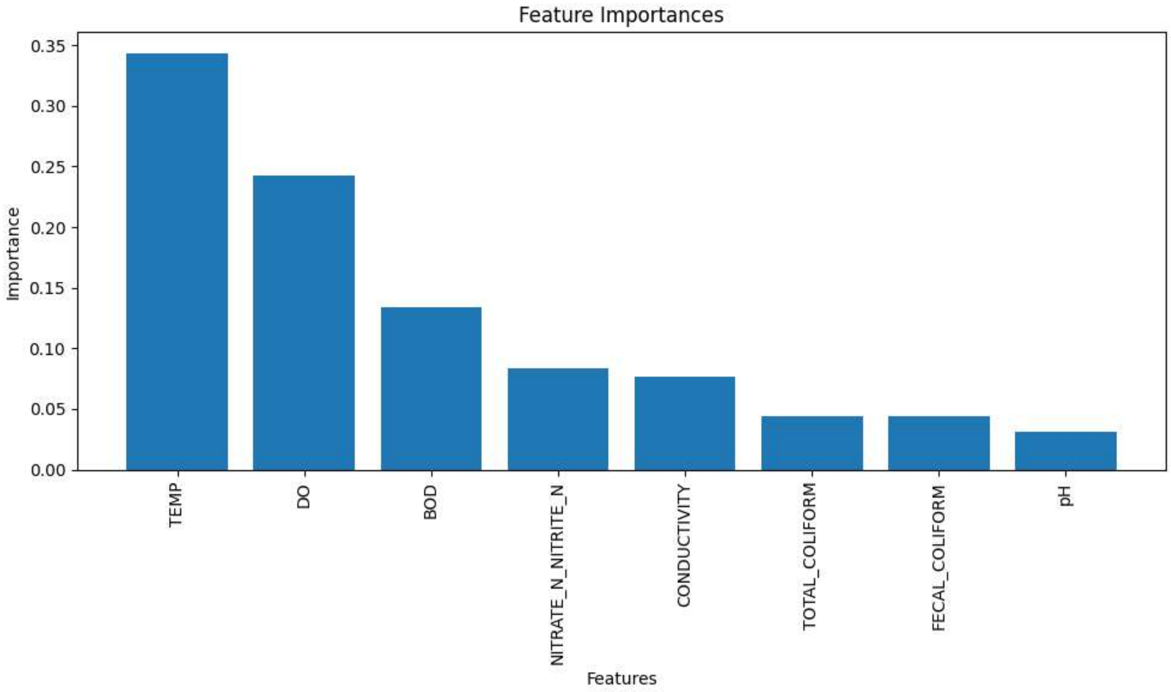
In [25]:

```

from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_
from sklearn.preprocessing import StandardScaler


accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, average='weighted')
recall = recall_score(y_test, y_pred, average='weighted')
f1 = f1_score(y_test, y_pred, average='weighted')

print("Random Forest Classifier Results:")
print("Accuracy:", accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("F1 Score:", f1)
```

```
Random Forest Classifier Results:
Accuracy: 0.9439252336448598
Precision: 0.9445907826977323
Recall: 0.9439252336448598
F1 Score: 0.9421765117977612
```

In [26]:

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.ensemble import RandomForestClassifier

rf_model = RandomForestClassifier()

rf_model.fit(X_scaled, cluster_labels)

importances = rf_model.feature_importances_

numeric_columns = ['TEMP', 'DO', 'pH', 'CONDUCTIVITY', 'BOD', 'NITRATE_N_NITRI

sorted_indices = np.argsort(importances)[::-1]
sorted_features = [numeric_columns[i] for i in sorted_indices]
sorted_importances = importances[sorted_indices]

plt.figure(figsize=(10, 6))
plt.bar(range(len(sorted_importances)), sorted_importances, align='center')
plt.xticks(range(len(sorted_importances)), sorted_features, rotation='vertical
plt.xlabel('Features')
plt.ylabel('Importance')
plt.title('Feature Importances')
plt.tight_layout()
plt.show()
```



In [55]:

```
import pandas as pd
import numpy as np
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler

data = pd.read_csv('gw_data.csv')
```

In [56]:

```
data = data.replace("#div/0!", np.nan)
```

In [57]:

```
rows_with_nan = data.isnull().any(axis=1)
```

In [58]:

```
data_without_nan = data.drop(data[rows_with_nan].index)
data
```

Out[58]:

	State Name	pH	Conductivity	BOD (mg/L)	Nitrate N (mg/L)	Faecal Coliform (MPN/100ml)	Total Coliform (MPN/100ml)	Total Dissolved Solids (mg/L)	Fluorid (mg/L
0	AP	7.3	776.133333	2.05	8.825	3.0	69.5	782.0	0.6
1	AP	8.1	619.700000	2.20	4.425	3.5	69.5	623.0	1.0
2	AP	7.8	759.266667	1.90	2.240	4.5	84.0	764.0	0.4
3	AP	7.1	2535.700000	2.90	23.250	5.5	93.0	2576.0	0.9
4	AP	8.1	2202.700000	2.05	24.625	3.5	78.0	2242.0	0.4
5	AP	7.8	1362.933333	1.60	4.035	3.0	24.5	1214.0	0.7
6	AP	8.0	716.666667	2.20	2.825	5.5	81.5	711.0	0.9
7	AP	7.6	7515.866667	2.35	8.800	8.0	135.0	7654.0	0.5
8	AP	7.1	1610.033333	1.00	6.850	3.0	11.5	1332.5	0.9

In [59]:

```
import pandas as pd
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import StandardScaler

data = pd.read_csv('gw_data.csv')
```

In [60]:

```
nan_counts = data.isnull().sum()
```

In [61]:

```
nan_counts
```

Out[61]:

```
State Name                         0
pH                                 0
Conductivity                       0
BOD (mg/L)                         0
Nitrate N (mg/L)                   0
Faecal Coliform (MPN/100ml)        0
Total Coliform (MPN/100ml)         0
Total Dissolved Solids (mg/L)      0
Fluoride (mg/L)                    0
dtype: int64
```

In [62]:

```
import pandas as pd

data = pd.read_csv('gw_data.csv')

data.dropna(inplace=True)
```

In [65]:

```
import numpy as np
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.impute import SimpleImputer
from skfuzzy.cluster import cmeans
from skfuzzy import membership

data = pd.read_csv('gw_data.csv')

numeric_columns = ['pH', 'Conductivity', 'BOD (mg/L)', 'Nitrate N (mg/L)', 'Fa
X = data[numeric_columns]

imputer = SimpleImputer(strategy='mean')
X = imputer.fit_transform(X)
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)


def membership_functions(x):

    pH_low = membership.trimf(x, [0, 0, 7])
    pH_medium = membership.trimf(x, [6, 7, 8])
    pH_high = membership.trimf(x, [7, 14, 14])

    conductivity_low = membership.trimf(x, [0, 0, 250])
    conductivity_medium = membership.trimf(x, [200, 400, 600])
    conductivity_high = membership.trimf(x, [500, 3000, 3000])

    bod_low = membership.trimf(x, [0, 0, 2])
    bod_medium = membership.trimf(x, [1, 3, 5])
    bod_high = membership.trimf(x, [4, 20, 20])

    nitrate_low = membership.trimf(x, [0, 0, 10])
    nitrate_medium = membership.trimf(x, [5, 10, 15])
    nitrate_high = membership.trimf(x, [10, 50, 50])

    faecal_coliform_low = membership.trimf(x, [0, 0, 500])
    faecal_coliform_medium = membership.trimf(x, [400, 800, 1000])
    faecal_coliform_high = membership.trimf(x, [800, 2000, 2000])

    total_coliform_low = membership.trimf(x, [0, 0, 1000])
    total_coliform_medium = membership.trimf(x, [500, 1500, 2000])
    total_coliform_high = membership.trimf(x, [1500, 4000, 4000])

    tds_low = membership.trimf(x, [0, 0, 500])
    tds_medium = membership.trimf(x, [400, 600, 1000])
    tds_high = membership.trimf(x, [800, 2000, 2000])

    fluoride_low = membership.trimf(x, [0, 0, 1])
    fluoride_medium = membership.trimf(x, [0.5, 1, 1.5])
    fluoride_high = membership.trimf(x, [1, 4, 4])

    return {
                'pH_low': pH_low,
        'pH_medium': pH_medium,
        'pH_high': pH_high,
        'conductivity_low': conductivity_low,
        'conductivity_medium': conductivity_medium,
        'conductivity_high': conductivity_high,
        'bod_low': bod_low,
        'bod_medium': bod_medium,
        'bod_high': bod_high,
        'nitrate_low': nitrate_low,
        'nitrate_medium': nitrate_medium,
        'nitrate_high': nitrate_high,
        'faecal_coliform_low': faecal_coliform_low,
        'faecal_coliform_medium': faecal_coliform_medium,
        'faecal_coliform_high': faecal_coliform_high,
        'total_coliform_low': total_coliform_low,
        'total_coliform_medium': total_coliform_medium,
        'total_coliform_high': total_coliform_high,
        'tds_low': tds_low,
        'tds_medium': tds_medium,
        'tds_high': tds_high,
        'fluoride_low': fluoride_low,
        'fluoride_medium': fluoride_medium,
        'fluoride_high': fluoride_high
    }

n_clusters = 3
cntr, u, _, _, _, _, _ = cmeans(X_scaled.T, n_clusters, m=2, error=0.005, maxi


cluster_labels = np.argmax(u, axis=0)

data_with_clusters = data.copy()
data_with_clusters['Cluster'] = cluster_labels

label_mapping = {
    0: 'Poor',
    1: 'Moderate',
    2: 'Good'
}

data_with_clusters['WaterQuality'] = data_with_clusters['Cluster'].map(label_m

pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)
data_with_clusters
```

```
5   AP  7.8  1362.933333  1.60   4.035   3.0   24.5   1214.0  0.7
6   AP  8.0   716.666667  2.20   2.825   5.5   81.5    711.0  0.9
7   AP  7.6  7515.866667  2.35   8.800   8.0  135.0   7654.0  0.5
8   AP  7.1  1610.033333  1.00   6.850   3.0   11.5   1332.5  0.9
9   AP  7.1  2448.366667  1.00  10.900   4.0   17.5   2006.0  1.0
10  AP  8.1  1274.700000  1.60   1.550   3.0   17.5   1164.0  0.8
11  AP  7.3  1540.100000  1.00   6.000   3.0    9.0   1211.0  0.5
12  AP  7.3  4798.766667  1.00   6.000   2.0    8.0   4240.5  1.0
13  AP  7.6   701.200000  1.50   3.220   3.0   93.0    680.0  0.7
14  AP  7.1   762.366667  2.40   8.000   4.0   93.0    746.0  0.8
15  AP  7.3  6532.433333  2.10   6.305   3.0   25.5   6117.0  1.5
16  AP  7.8  1284.933333  1.60   0.500   3.0   30.0   1140.0  0.3
```

In [66]:
```python
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
```

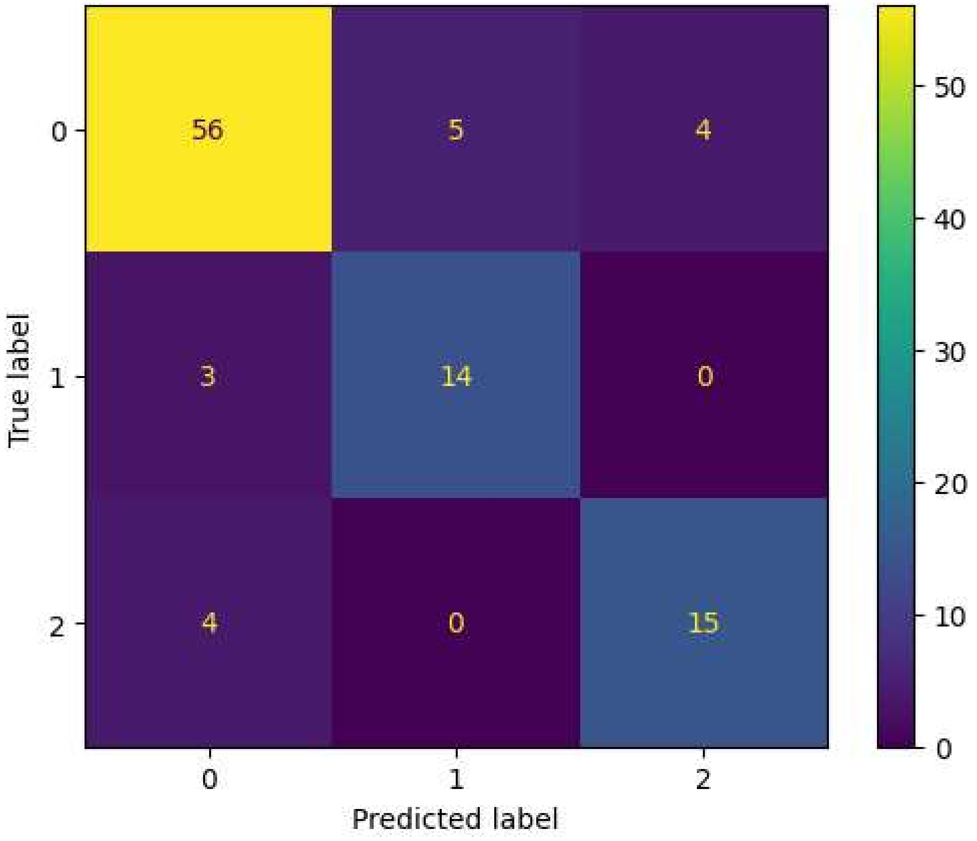
In [67]:
```python
from sklearn.preprocessing import LabelEncoder

X = data[numeric_columns]
y = data_with_clusters['WaterQuality']


imputer = SimpleImputer(strategy='mean')
X = imputer.fit_transform(X)
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)


label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)
```

```
In [68]: from sklearn.model_selection import train_test_split
         from sklearn.neighbors import KNeighborsClassifier
         from sklearn.metrics import accuracy_score


         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, rando


         knn = KNeighborsClassifier(n_neighbors=5)
         knn.fit(X_train, y_train)


         pred = knn.predict(X_test)

         accuracy = accuracy_score(y_test, pred)
         print("Accuracy:", accuracy)
```

```
Accuracy: 0.8415841584158416
```

```
In [69]: from sklearn.metrics import ConfusionMatrixDisplay
         from sklearn.metrics import confusion_matrix
```

```
In [70]: import  matplotlib.pyplot as plt
         cm = confusion_matrix(y_test, pred)
         cm_disp = ConfusionMatrixDisplay(cm)
         cm_disp.plot()
         plt.show()
```


True label
Predicted label