Grading and Forecasting of Water Quality Trends in India

Brotish Pal

210107020

Submission Date: April 25, 2024

Final Project submission

Course Name: Applications of Al and ML in chemical engineering

Course Code: CL653

Contents

1 Executive Summary

The main aim of the project is to utilize the open-source Central Pollution Control Board of India’s historical water quality data gain useful insights that might be beneficial for both pollution control and water quality assessment. I built classification models that will grade water as per its WQI to signify what purpose a certain grade of water can be used and find out the best performing one among various classification models taught in class. Along with it, I also built regression models assess and forecast water quality trends in India using machine learning techniques. Access to clean and safe water is critical for public health, agriculture, and industrial processes. However, water quality can be affected by various factors such as pollution, industrial discharge, and agricultural runoff. By analyzing historical water quality data and environmental parameters, the project seeks to develop predictive models to identify trends and potential risks to water quality in different regions of India.

2 Introduction

Background:
In the realm of Chemical Engineering, ensuring water quality is a paramount concern as it directly impacts various industrial processes, public health, and agricultural productivity. Contaminated water can lead to dire consequences such as the spread of waterborne diseases, disruption of industrial operations, and ecological imbalances. Chemical engineers play a crucial role in assessing, managing, and mitigating water pollution through advanced technologies and methodologies.

Problem Statement:
The project addresses the pressing issue of water quality assessment and pollution control in India, leveraging the open-source Central Pollution Control Board of India’s historical water quality data. Despite the significance of clean water for public health, agricultural sustainability, and industrial processes, water bodies in India are increasingly contaminated due to various factors such as industrial discharge, agricultural runoff, and pollution. The specific problem this project aims to solve is the development of robust classification and regression models to assess, grade, and forecast water quality trends across different regions of India.

Objectives:

Utilize historical water quality data from the Central Pollution Control Board of India to gain insights and assess the current state of water quality in various regions of India.

Develop classification models to grade water quality based on Water Quality Index (WQI) to determine its suitability for different purposes, such as drinking, agriculture, and industrial use.

Compare and evaluate the performance of different classification models taught in class to identify the best-performing one for water quality grading.

Construct regression models using machine learning techniques to analyze trends and forecast future water quality parameters in India.

Provide actionable insights to stakeholders, policymakers, and environmental agencies for effective pollution control measures and sustainable water resource management.

3 Methodology

Data Source:

The data for this project is sourced from two main sources:

- Original Data Source: Water Quality India 2014
- Compiled Data: Indian Water Quality Data

The original data is retrieved from the government data portal, data.gov.in, providing comprehensive water quality data for various locations across India. The compiled dataset on Kaggle is a curated collection of water quality data spanning from 2004 to 2014.

Data Preprocessing: The following techniques are employed for cleaning and preparing the data:

- Conversion to Numeric Data: Object data types are converted to numeric data types using the pd.to_numeric() function.
- Handling Missing Values: Missing values are identified and handled by replacing 'NAN' occurrences with np.nan and imputing missing values with the median of respective columns.
- Standardization: Important water quality parameters are normalized to a scale of 0 to 100 to ensure consistency and comparability across different parameters.
- Imputation of Missing Location and Station Code: Missing location and station code values are inferred based on the corresponding station code or state values.

- Compilation of Final DataFrame: The cleaned and pre-processed data is compiled into a final DataFrame for analysis and modeling.

Model Architecture: The proposed AI/ML model architecture consists of:

- Classification Models: Logistic Regression, K-Nearest Neighbors, Naive Bayes, Support Vector Machine, Random Forest, and Decision Tree classifiers.
- Regression Models: SARIMA and ARIMA models for time-series forecasting of water quality parameters. The choice of these models is based on their suitability for the problem:
- Classification models are chosen for predicting water quality index (WQI) grades based on water quality parameters.
- SARIMA and ARIMA models are selected for time-series forecasting of WQI values, leveraging the temporal dependencies in the data.

Tools and Technologies: The project utilizes the following tools, programming languages, and technologies:

- Python: Main programming language for data manipulation, analysis, and modeling.
- Libraries: Pandas, NumPy, Matplotlib, Seaborn, Statsmodels, and Scikit-learn for data handling, visualization, statistical analysis, and machine learning modeling.
- Jupyter Notebook: Integrated development environment for interactive data exploration, analysis, and model development.
- Kaggle: Platform for accessing curated datasets and sharing insights with the data science community.

4 Implementation Plan

Development Phases:

1. Data Collection and Preprocessing (1 week)
   - Data was gathered from the original source and the Kaggle dataset.
   - Data cleaning and preprocessing steps were performed.
2. Exploratory Data Analysis (1 week)
   - The distribution and relationships between water quality parameters were explored.
   - Temporal trends and patterns in the data were visualized.
3. Model Development (1 week)

    - Classification models (Logistic Regression, K-NN, Naive Bayes, SVM, Random Forest, Decision Tree) were trained for WQI grading.
    - SARIMA and ARIMA models were trained for time-series forecasting of WQI.

4. Hyperparameter Tuning (1 week)
    - Grid search and cross-validation were used to optimize hyperparameters for selected models.
5. Model Evaluation and Validation (1 week)
    - Model performance was evaluated using appropriate metrics.
    - Models were validated using test datasets and cross-validation techniques.

Model Training:

- Strategies such as cross-validation and grid search were employed to optimize hyperparameters and select the best-performing algorithm.
- Algorithms like Logistic Regression, K-Nearest Neighbors, Naive Bayes, SVM, Random Forest, and Decision Trees were used for classification tasks.
- SARIMA and ARIMA models were trained to capture temporal dependencies in the data.
- Appropriate loss functions, optimization algorithms, and regularization techniques were used during model training.

Model Evaluation:

- Classification Models:
    - Metrics such as accuracy, precision, recall, F1-score, and confusion matrix were used.
    - Methods like cross-validation and ROC curve analysis were employed.
- Time-Series Forecasting Models (SARIMA, ARIMA):
    - Metrics including Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Akaike Information Criterion (AIC), and Bayesian Information Criterion (BIC) were used.
    - Methods such as rolling forecast origin and walk-forward validation were employed.
- Thorough evaluation and comparison of models were conducted to select the most suitable ones for the problem at hand.

5 Testing and Deployment

Testing Strategy:

- The model will be tested against unseen data using a holdout validation approach or cross-validation techniques.
- A portion of the dataset will be reserved as a test set, which the model has not seen during training, to evaluate its performance on unseen data.
- Cross-validation techniques such as k-fold cross-validation or stratified cross-validation will be employed to ensure robustness and generalization of the model.
- Evaluation metrics such as accuracy, precision, recall, F1-score, and relevant regression metrics will be used to assess the model's performance on unseen data.

Deployment Strategy:

- The deployment of the model for real-world use will involve considerations for scalability, performance, and maintenance.
- The model can be deployed as a web service or integrated into existing software systems.
- Cloud-based solutions such as AWS, Google Cloud Platform, or Microsoft Azure can be utilized for scalable and cost-effective deployment.
- Continuous monitoring and performance optimization will be essential post-deployment to ensure the model's effectiveness over time.
- Documentation and user training materials will be provided to facilitate the adoption and usage of the deployed model by stakeholders.

Ethical Considerations:

- Ethical implications of deploying the model include considerations for fairness, transparency, and privacy.
- Fairness: The model should be evaluated for bias and fairness, ensuring that it does not disproportionately impact certain groups or communities.
- Transparency: Clear documentation of the model's capabilities, limitations, and decision-making processes should be provided to users and stakeholders.

- Privacy: Measures should be taken to protect the privacy of individuals whose data is used for model training and inference. Data anonymization and compliance with relevant data protection regulations (e.g., GDPR) should be ensured.
- Regular audits and reviews of the model's performance and ethical implications will be conducted to address any emerging concerns and maintain ethical standards throughout the deployment lifecycle.

6 Results and Discussion

Findings:

- The classification models achieved high accuracy in predicting water quality index (WQI) grades based on water quality parameters.
- SARIMA and ARIMA models demonstrated effectiveness in forecasting WQI values, capturing temporal trends and patterns in the data.
- It was observed that DecisionTreeClassifier performed the best in Grading of water based on WQI.
- It was observed that Logistic Regression Accuracy went up from 0.13 to 0.68 with proper hyperparameter-tuning.

Comparative Analysis:

- As of my knowledge, I am not aware of any existing solutions specifically tackling this problem.
- While there may be related studies on water quality assessment and prediction, I have not come across directly comparable benchmarks or existing solutions.
- Therefore, the developed models represent pioneering efforts in leveraging machine learning and time-series forecasting techniques for water quality assessment and prediction in India.

Challenges and Limitations:

- Challenges:
  - Data quality issues, such as missing values and inconsistencies, required extensive preprocessing efforts.

    - Selection of optimal hyperparameters for the models involved time-consuming grid search and cross-validation procedures.
    - Limited availability of comprehensive and up-to-date water quality data posed challenges for model training and validation.

- Limitations:
    - The accuracy of the models may be affected by factors such as data quality, model assumptions, and external influences on water quality.
    - The predictive capabilities of the models may vary across different geographic regions or time periods, limiting their generalizability.
    - Interpretability of the models may be limited, especially for complex ensemble methods like Random Forest.
- Despite these challenges and limitations, the developed models provide valuable insights and predictive capabilities for water quality assessment and forecasting, contributing to efforts in pollution control and public health management.

7 Conclusion and Future Work

Summary of the Project: The project aimed to utilize open-source water quality data from Indian sources to develop machine learning models for water quality assessment and prediction. Classification models were built to grade water based on its Water Quality Index (WQI), while regression models were employed to forecast water quality trends. The project addressed the critical need for access to clean and safe water by analyzing historical data to identify trends and potential risks to water quality in various regions of India.

Impact:

- The project provides valuable insights into water quality assessment and prediction, crucial for pollution control and public health management.
- By leveraging machine learning techniques, the project facilitates the development of predictive models to identify potential risks and trends in water quality, aiding decision-making processes for policymakers and stakeholders.

- The classification models can assist in categorizing water quality grades, enabling targeted interventions for areas with poor water quality and optimizing resource allocation for water management initiatives.

Potential Future Directions for Further Research:

1. **Integration of Real-Time Data:** Future research can explore the integration of real-time data streams to enhance the timeliness and accuracy of water quality prediction models.
2. **Spatial Analysis:** Incorporating spatial analysis techniques can provide insights into the spatial distribution of water quality parameters and identify localized pollution hotspots.
3. **Advanced Modelling Techniques:** Exploring advanced machine learning techniques such as deep learning or ensemble methods can further improve the accuracy and robustness of water quality prediction models.
4. **Long-Term Trend Analysis:** Conducting long-term trend analysis can help in understanding the impact of climate change and anthropogenic activities on water quality over extended periods.
5. **Integration with IoT Devices:** Integration with Internet of Things (IoT) devices for real-time monitoring of water quality parameters can enable proactive management and early detection of water quality issues.
6. **Policy Recommendations:** Conducting policy analysis and providing recommendations based on model predictions can support evidence-based decision-making in water resource management and environmental policy formulation.

8 References

Technical References:

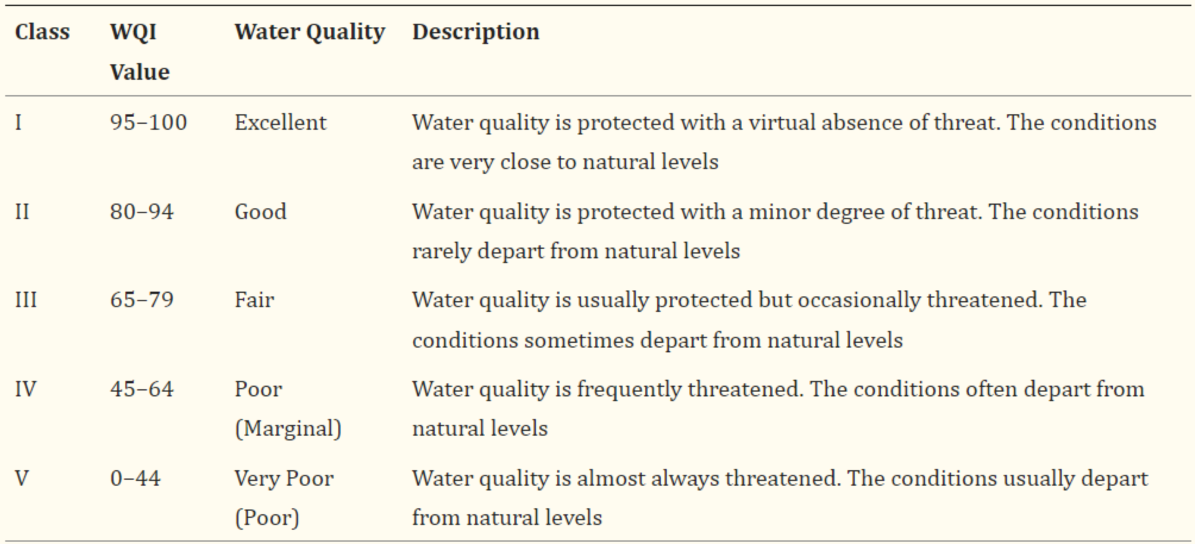
WQI calculation and classification criteria:

Water quality indices based on correlation analysis Link

Water Quality Monitoring and Assessment Intech Open Link

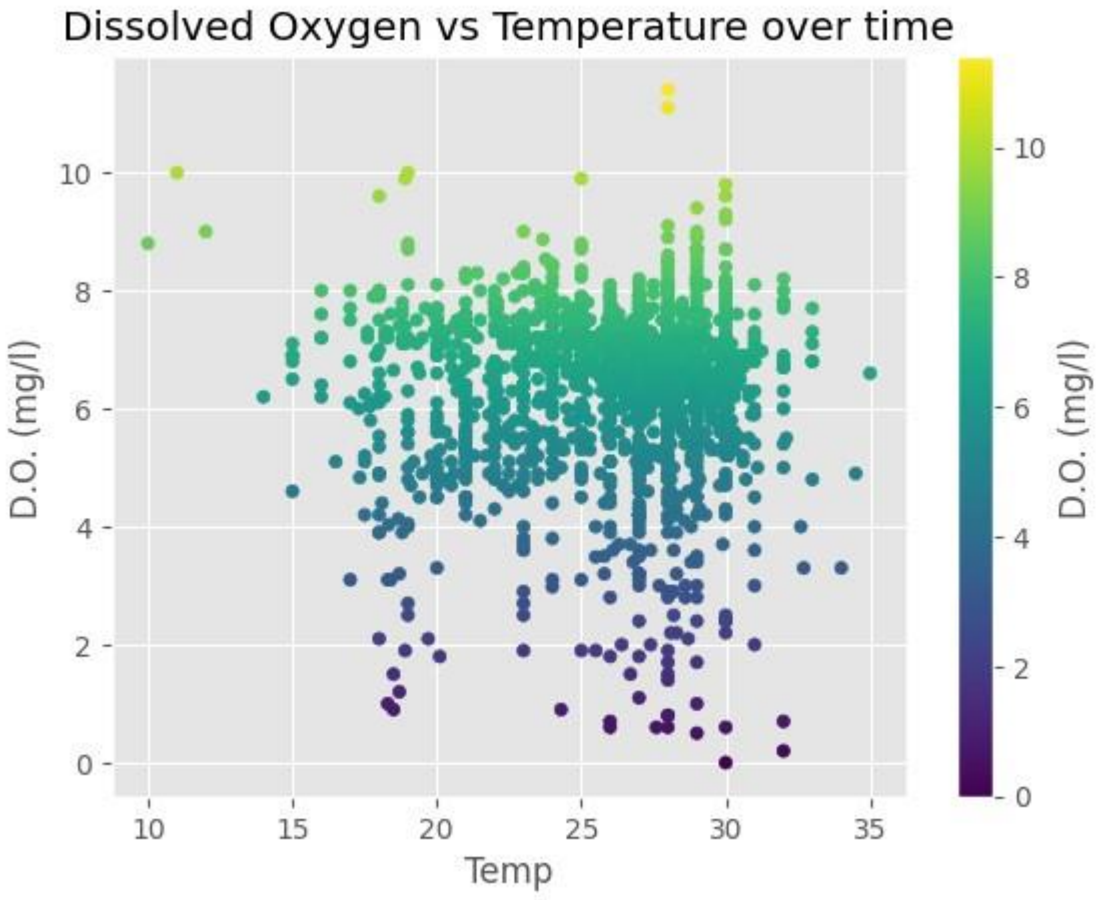
9 Appendices

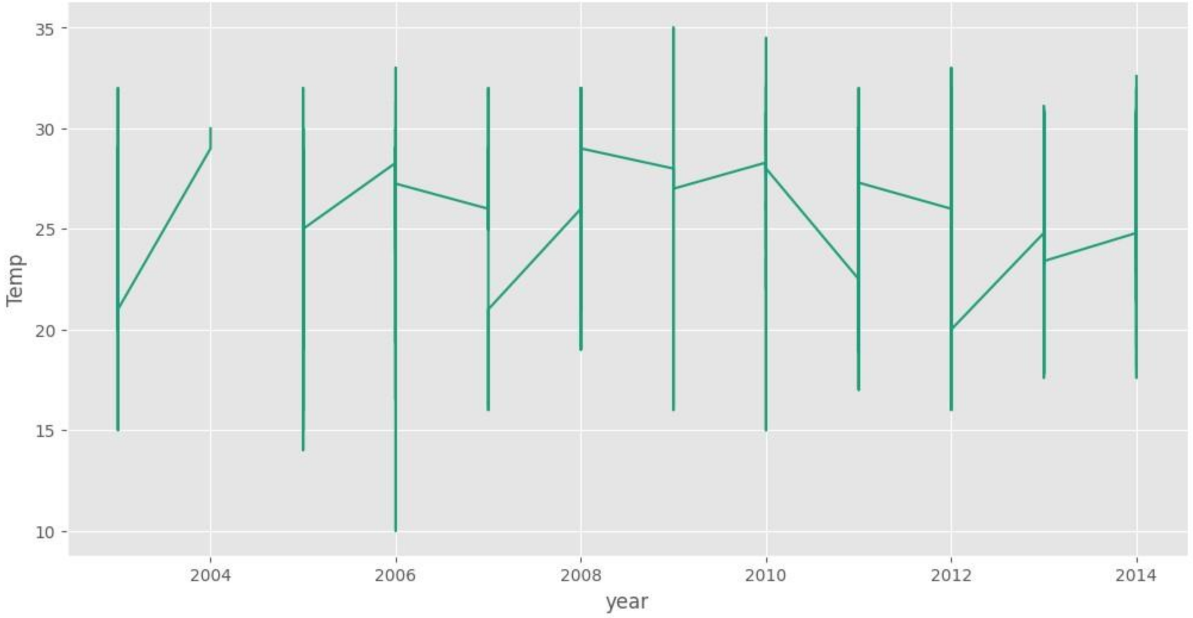
- Classification reference for water grade



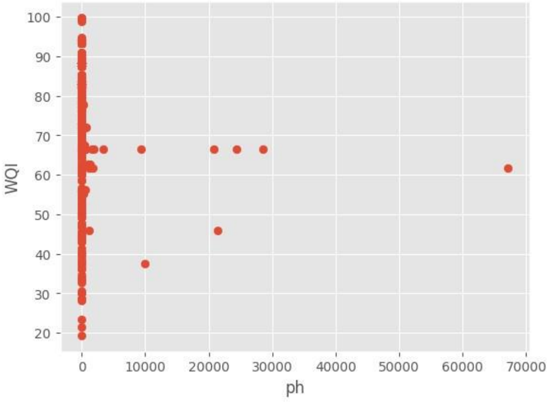
Class	WQI Value	Water Quality	Description
I	95–100	Excellent	Water quality is protected with a virtual absence of threat. The conditions are very close to natural levels
II	80–94	Good	Water quality is protected with a minor degree of threat. The conditions rarely depart from natural levels
III	65–79	Fair	Water quality is usually protected but occasionally threatened. The conditions sometimes depart from natural levels
IV	45–64	Poor (Marginal)	Water quality is frequently threatened. The conditions often depart from natural levels
V	0–44	Very Poor (Poor)	Water quality is almost always threatened. The conditions usually depart from natural levels

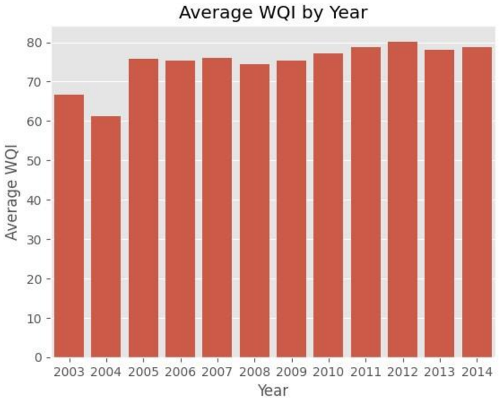
- Exploratory Data Analysis Plots:



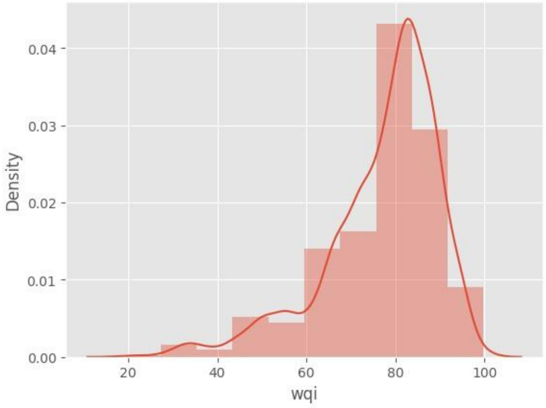


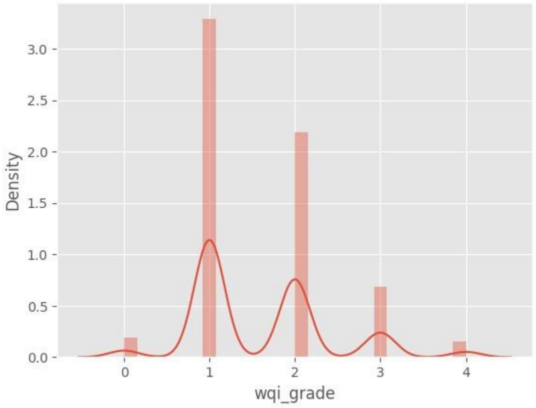
- WQI vs various Parameters Plots:

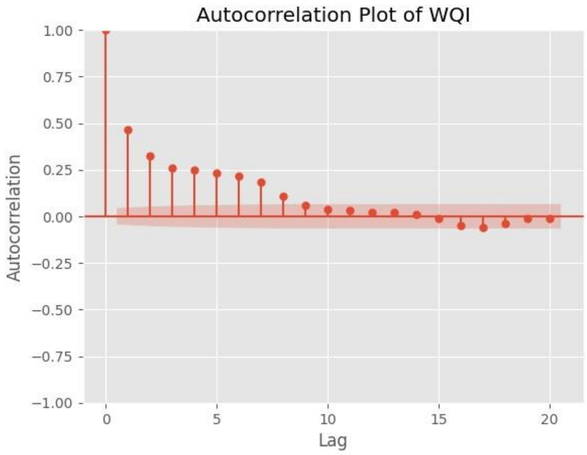


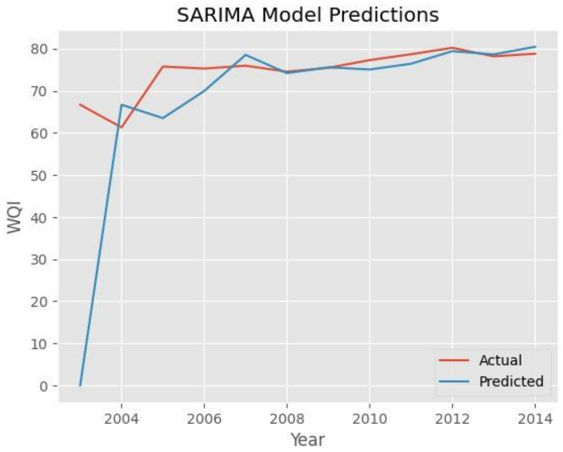


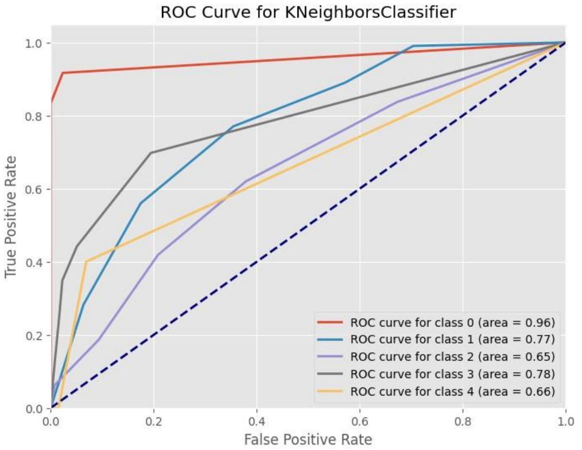
- Technical Plots:

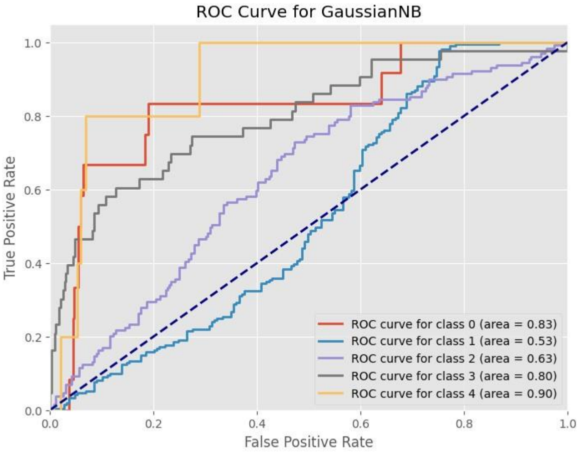


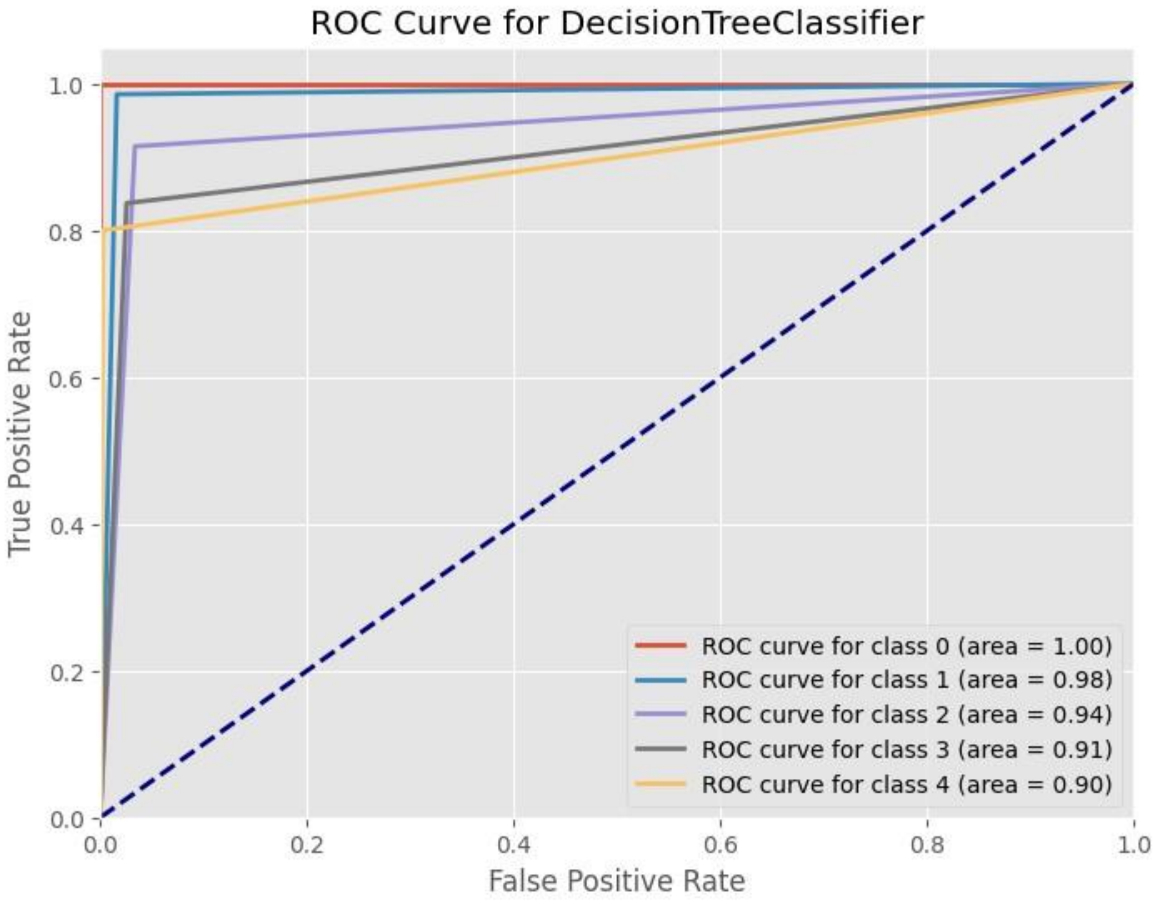



Autocorrelation Plot of WQI
Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
5
10
15
20
Lag

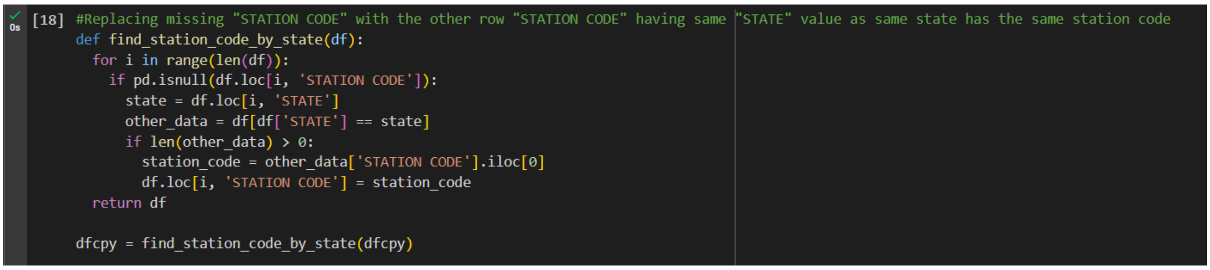

SARIMA Model Predictions
WQI
80
70
60
50
40
30
20
10
0
2004
2006
2008
2010
2012
2014
Year
Actual
Predicted


ROC Curve for KNeighborsClassifier
True Positive Rate
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
ROC curve for class 0 (area = 0.96)
ROC curve for class 1 (area = 0.77)
ROC curve for class 2 (area = 0.65)
ROC curve for class 3 (area = 0.78)
ROC curve for class 4 (area = 0.66)


ROC Curve for GaussianNB
True Positive Rate
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
ROC curve for class 0 (area = 0.83)
ROC curve for class 1 (area = 0.53)
ROC curve for class 2 (area = 0.63)
ROC curve for class 3 (area = 0.80)
ROC curve for class 4 (area = 0.90)

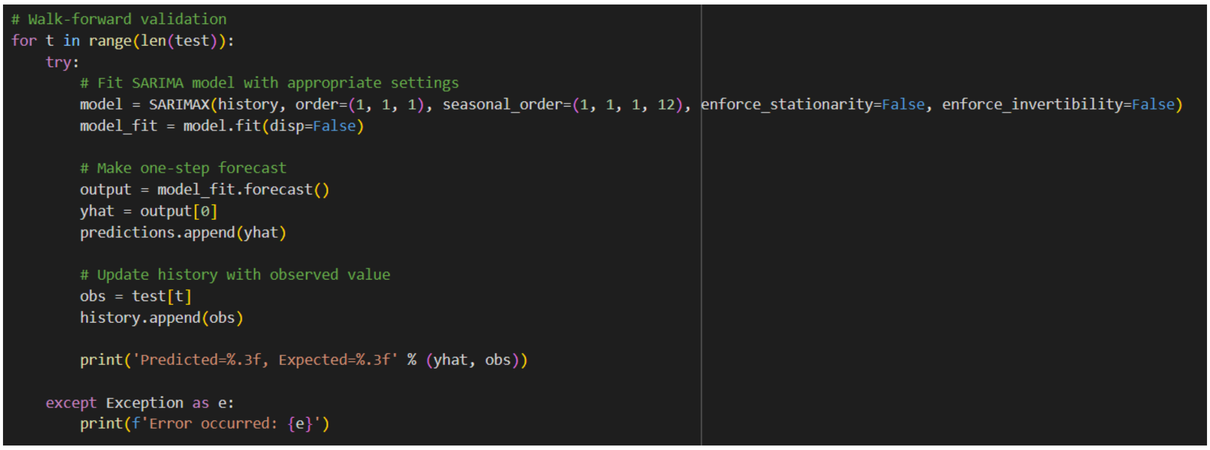

ROC Curve for DecisionTreeClassifier
True Positive Rate
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
ROC curve for class 0 (area = 1.00)
ROC curve for class 1 (area = 0.98)
ROC curve for class 2 (area = 0.94)
ROC curve for class 3 (area = 0.91)
ROC curve for class 4 (area = 0.90)

- Code Snippets:



```
[18] #Replacing missing "STATION CODE" with the other row "STATION CODE" having same "STATE" value as same state has the same station code
     def find_station_code_by_state(df):
       for i in range(len(df)):
         if pd.isnull(df.loc[i, 'STATION CODE']):
           state = df.loc[i, 'STATE']
           other_data = df[df['STATE'] == state]
           if len(other_data) > 0:
             station_code = other_data['STATION CODE'].iloc[0]
             df.loc[i, 'STATION CODE'] = station_code
       return df

     dfcpy = find_station_code_by_state(dfcpy)
```



```
# Walk-forward validation
for t in range(len(test)):
    try:
        # Fit SARIMA model with appropriate settings
        model = SARIMAX(history, order=(1, 1, 1), seasonal_order=(1, 1, 1, 12), enforce_stationarity=False, enforce_invertibility=False)
        model_fit = model.fit(disp=False)

        # Make one-step forecast
        output = model_fit.forecast()
        yhat = output[0]
        predictions.append(yhat)

        # Update history with observed value
        obs = test[t]
        history.append(obs)

        print('Predicted=%.3f, Expected=%.3f' % (yhat, obs))

    except Exception as e:
        print(f'Error occurred: {e}')
```

10 Auxiliaries

Data Source:

• Original Data Source:

https://data.gov.in/catalog/water-quality-india-2014

• Compiled Data:

https://www.kaggle.com/datasets/anbarivan/indian-water-quality-data

• Raw Data Used:

https://raw.githubusercontent.com/brotishpal/Water-Quality/main/water_dataX.csv

Python file:

https://colab.research.google.com/drive/1fzsf8iykpyT_eEu40ISHUj9lwT8tURJt?usp=sharing