Project 2: Air Q Assessment TN

Objective:

Our goal is to incorporate Machine Learning algorithms to improve the accuracy of the predictive model by continuously monitoring and training the ML Model for Air Quality Assessment in Tamil Nadu.

Project Definition:

The project aims to analyze and visualize air quality data from monitoring stations in Tamil Nadu. The objective is to gain insights into air pollution trends, identify areas with high pollution levels, and develop a predictive model to estimate RSPM/PM10 levels based on SO2 and NO2 levels. This project involves defining objectives, designing the analysis approach, selecting visualization techniques, and creating a predictive model using Python and relevant libraries.

Phase 2: Innovation

Consider incorporating machine learning algorithms to improve the accuracy of the predictive model.

1. Data Preparation:
   - Download and clean the air quality data from the provided dataset link.
   - Handle missing values, if any, and perform necessary data preprocessing.
   - Create a dataset that includes features (SO2 and NO2 levels) and the target variable (RSPM/PM10 levels).

2. Feature Engineering:

   - Consider adding additional features that may impact air quality, such as weather conditions, geographical coordinates, or time of day.
   - Transform or engineer features to make them more informative for the predictive model.

3. Data Splitting:

   - Split your dataset into training, validation, and test sets. This ensures that you have a separate dataset for model training, validation, and final evaluation.
4. Select Machine Learning Algorithm:

- Choose a machine learning algorithm suitable for your regression task. Linear regression, decision trees, random forests, or gradient boosting are good options to start with.

5. Model Training and Tuning:
   - Train the selected machine learning model on the training dataset.
   - Tune the model's hyperparameters to optimize its performance. You can use techniques like grid search or random search for hyperparameter tuning.
6. Model Evaluation:
   - Evaluate the model's performance using appropriate regression metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), or R-squared (R2).
   - Use the validation dataset to assess the model's accuracy and generalization capability.
7. Visualization:
   - Create visualizations to present your findings and model performance.
   - Use libraries like Matplotlib or Seaborn to create informative plots and graphs.
8. Model Interpretation:
   - Depending on the chosen algorithm, consider methods for model interpretation and feature importance analysis. This helps understand which features have the most significant impact on air quality.
9. Predictive Model Deployment (Optional):
   - If applicable, deploy the predictive model in a real-time or batch processing environment, allowing for real-time predictions or batch predictions for future data.
10. Documentation and Reporting:
    - Document the entire process, including data preprocessing, model selection, hyperparameter tuning, and evaluation.
    - Create a detailed report or presentation summarizing your findings, insights, and the performance of the predictive model.
11. Phase 2 Innovation:
    - Explore advanced machine learning techniques or alternative models to further enhance predictive accuracy.
    - Consider incorporating time series forecasting methods if the dataset contains temporal information.
    - Remember to continuously monitor and update your predictive model as new data becomes available to ensure it remains accurate and relevant for air quality assessment in Tamil Nadu.