AIR QUALITY ANALYSIS AND PREDICTION IN TAMIL NADU

Phase 2:

Short Explanation about the question:
Air Quality Analysis and Prediction in Tamil Nadu

The question involves analyzing and predicting air quality in the state of Tamil Nadu. Specifically, it focuses on understanding the relationship between air quality parameters, such as sulfur dioxide (SO2) and nitrogen dioxide (NO2), and particulate matter (RSPM/PM10), which can impact air quality and human health.

Where you got the dataset and its detail:

The dataset is obtained from Kaggle. The dataset can be found at this link: [Air Quality Data Set](https://www.kaggle.com/datasets/fedesoriano/air-quality-data-set). It contains information about air quality measurements in Tamil Nadu in 2014.

Details about columns which I used in the dataset - SO_2, NO_2, and RSPM/PM10:

SO_2 (Sulfur Dioxide):SO_2 is a gaseous air pollutant produced by the burning of fossil fuels, particularly in industrial processes. It is a key contributor to air pollution and can have adverse health effects.

NO_2(Nitrogen Dioxide): NO_2 is another gaseous air pollutant, often associated with vehicle emissions and industrial processes. Like SO_2, it can also impact air quality and health.

RSPM/PM10 (Particulate Matter): Particulate matter refers to tiny solid particles or liquid droplets in the air. PM10 specifically refers to particles with a diameter of 10 micrometers or smaller. These particles can originate from various sources and affect air quality and respiratory health.

Details of libraries to be used - referred Jupyter notebook file

We used several libraries in Jupyter notebook, including **'pandas'** for data manipulation, **'nump'** for numerical operations**, 'matplotlib.pyplot'** for data visualization, and **'sklearn'** (scikit-learn) for machine learning tools.

How to train and test - referred Jupyter notebook file:

In Jupyter notebook, these steps are followed:

- Split the dataset into a training set and a testing set using a random mask.
- Created a linear regression model using scikit-learn's ‘linear_model.LinearRegression()’.
- Trained the model on the training data using 'SO_2' and 'NO_2' as features and 'RSPMorPM10' as the target variable.
- Made predictions on the test data using the trained model.
- Evaluated the model's performance by calculating the Mean Squared Error (MSE) and the Variance score (R^2) to check how well the model predicts 'RSPMorPM10' based on 'SO_2' and 'NO_2'.

Rest of the explanation:

The rest of the analysis involves data preprocessing, including renaming columns and handling missing values, data visualization to understand the relationship between 'SO_2' and 'RSPMorPM10', and building and evaluating a linear regression model for predicting 'RSPMorPM10'.

What metrics used for the accuracy check (metric used to explain the results of the coefficients received)

- For accuracy check,the Mean Squared Error (MSE) and the Variance score (R^2) are used

➢ **Mean Squared Error (MSE):**This metric measures the average squared difference between the predicted and actual values. A lower MSE indicates a better-performing model. In the analysis, the MSE was approximately 720.36, suggesting the model's predictions were not very close to the actual values on average.

➢ **Variance score (R^2)**: This score represents the proportion of the variance in the dependent variable ('RSPMorPM10') that is predictable from the independent variables ('SO_2' and 'NO_2'). A higher R^2 score (close to 1) indicates a better fit of the model to the data. In the analysis, the R^2score was approximately 0.17, indicating that the model explains only a small portion of the variance in 'RSPMorPM10'.