AIR QUALITY ANALYSIS AND PREDICTION

Introduction:

This project aims to comprehensively analyze air quality in Tamil Nadu, addressing key aspects such as average pollutant concentrations, trend identification, and predictive modeling. By calculating the average levels of SO_2, NO_2, and RSPM/PM10 across diverse monitoring stations and regions, this analysis establishes a foundational understanding of typical pollution scenarios. Leveraging data visualization techniques like trend plots and daily average air quality visualizations, the study delves into temporal and spatial patterns, shedding light on the evolution of air pollution. The incorporation of predictive modeling enhances the project's utility by allowing users to estimate pollution levels based on input data, fostering an interactive and proactive approach to air quality assessment.

Project Objectives:

The primary objectives of this project are to analyze air quality data in Tamil Nadu, specifically focusing on the levels of SO_2, NO_2, and RSPM/PM10. The project aims to:

1. Calculate Average Pollutant Levels: Compute the average concentrations of SO_2, NO_2, and RSPM/PM10 across different monitoring stations, cities, and areas in Tamil Nadu.

2. Identify Pollution Trends: Explore temporal trends in air pollution, particularly focusing on long-term patterns and variations in RSPM/PM10 levels.

3. Create Predictive Models: Develop a linear regression model to predict RSPM/PM10 levels based on SO_2 and NO_2 concentrations.

4. Visualize Data: Utilize data visualization libraries such as Matplotlib and Seaborn to create informative plots and visualizations for effective communication of insights.

Analysis Approach:

The analysis is conducted in several phases:

1. Data Loading and Preprocessing:

Load the air quality data and preprocess it to ensure that it is in a suitable format for analysis. This involves handling missing values, converting data types, and filtering data relevant to Tamil Nadu.

2. Average Pollution Levels Analysis:

Calculate the mean concentrations of SO_2, NO_2, and RSPM/PM10 for different areas, providing an overview of typical pollution levels.

3. Trend Analysis:

Identify temporal trends in air pollution by visualizing the concentration levels of SO_2, NO_2, and RSPM/PM10 over time. This includes both overall trends and specific focus on RSPM/PM10.

4. Predictive Modeling:

Train a linear regression model using the concentrations of SO_2 and NO_2 to predict RSPM/PM10 levels. Evaluate the model's performance and provide a tool for predicting pollutant levels based on user input.

5. Visualization Techniques:

Utilize Matplotlib and Seaborn for creating visualizations, including line plots, box plots, and a heatmap. These visualizations enhance the understanding of pollution trends and spatial distribution.

Code Implementation:

The project is implemented using Python, and the relevant code snippets have been provided in the documentation. The code includes:

- Loading and preprocessing the data.
- Calculating average pollution levels.
- Visualizing trends over time.
- Training a linear regression model for prediction.
- User input for predicting pollutant levels.
- Creating visualizations using Matplotlib and Seaborn.

Python code :

```
import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LinearRegression

from sklearn.metrics import mean_squared_error, r2_score

# Load air quality data

data = pd.read_csv('cpcb_dly_aq_tamil_nadu-2014.csv')

# Data Preprocessing

data['Sampling Date'] = pd.to_datetime(data['Sampling Date'])

tn_data = data[data['State'] == 'Tamil Nadu']

# Display basic statistics

print(data.describe())

# Check for missing values

print(data.isnull().sum())
```

```
# Check unique values in categorical columns
print(data['State'].unique())
print(data['City/Town/Village/Area'].unique())
# Plot trends in RSPM/PM10 pollution levels over time
plt.figure(figsize=(22, 12))
plt.plot(data.index, data['RSPM/PM10'], marker='o', linestyle='-', color='b', label='RSPM/PM10')
plt.xlabel('Date')
plt.ylabel('RSPM/PM10 Levels')
plt.title('Trends in RSPM/PM10 Pollution Levels Over Time')
plt.grid(True)
plt.legend()
plt.show()
# Group data by date and calculate mean values
daily_mean = data.groupby('Sampling Date').mean()
# Plot daily average air quality
plt.figure(figsize=(12, 6))
plt.plot(daily_mean.index, daily_mean['SO2'], label='Mean SO2')
plt.plot(daily_mean.index, daily_mean['NO2'], label='Mean NO2')
plt.plot(daily_mean.index, daily_mean['RSPM/PM10'], label='Mean RSPM/PM10')
plt.xlabel('Sampling Date')
plt.ylabel('Mean Concentration')
plt.title('Daily Average Air Quality in Tamil Nadu')
plt.legend()
plt.grid(True)
plt.show()
# Calculate daily average SO2 and NO2 concentrations
daily_mean = data.groupby('Sampling Date')[['SO2', 'NO2']].mean()
# Plot daily average SO2 and NO2 concentrations
plt.figure(figsize=(12, 6))
plt.plot(daily_mean.index, daily_mean['SO2'], label='Mean SO2 Concentration')
plt.plot(daily_mean.index, daily_mean['NO2'], label='Mean NO2 Concentration')
```

```
plt.xlabel('Sampling Date')
plt.ylabel('Mean Concentration (µg/m³)')
plt.title('Daily Average SO2 and NO2 Concentrations in Tamil Nadu')
plt.legend()
plt.grid(True)
plt.show()
# Display statistics and box plots for SO2 and NO2 concentrations
so2_stats = data['SO2'].describe()
no2_stats = data['NO2'].describe()
plt.figure(figsize=(12, 6))
plt.subplot(1, 2, 1)
data.boxplot(column='SO2')
plt.title('SO2 Concentration Box Plot')
plt.subplot(1, 2, 2)
data.boxplot(column='NO2')
plt.title('NO2 Concentration Box Plot')
plt.tight_layout()
plt.show()
# Visualize trends in air pollution over time using seaborn
plt.figure(figsize=(12, 6))
for pollutant in ['SO2', 'NO2', 'RSPM/PM10']:
    sns.lineplot(data=tn_data, x='Sampling Date', y=pollutant, label=pollutant)
plt.xlabel('Year')
plt.ylabel('Concentration (µg/m³)')
plt.title('Air Pollution Trends in Tamil Nadu')
plt.legend()
plt.grid(True)
plt.show()
# Train a linear regression model to predict RSPM/PM10 levels
selected_columns = ['SO2', 'NO2', 'RSPM/PM10']
tn_data = tn_data[selected_columns].dropna()
```

```
X = tn_data[['SO2', 'NO2']]
y = tn_data['RSPM/PM10']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f'Mean Squared Error: {mse}')
print(f'R-squared (Coefficient of Determination): {r2}')
# User Input for Prediction
user_so2 = float(input("Enter SO2 concentration (µg/m³): "))
user_no2 = float(input("Enter NO2 concentration (µg/m³): "))
# Predict RSPM/PM10 level based on user input
user_input = pd.DataFrame({'SO2': [user_so2], 'NO2': [user_no2]})
predicted_rspm_pm10 = model.predict(user_input[['SO2', 'NO2']])
print(f'Predicted RSPM/PM10 Level: {predicted_rspm_pm10[0]:.2f}')
# Visualization of Actual vs. Predicted
plt.figure(figsize=(8, 8))
plt.scatter(X_test['NO2'], y_test, color='blue', label='Actual', alpha=0.5)
plt.scatter(user_input['NO2'], predicted_rspm_pm10, color='red', marker='X', label='User Input')
plt.xlabel('NO2 Concentration')
plt.ylabel('RSPM/PM10 Levels')
plt.title('Actual vs. Predicted RSPM/PM10 Levels')
plt.legend()
plt.grid(True)
plt.show()
# Visualize pollutant levels by location and time using a heatmap
df = pd.read_csv('cpcb_dly_aq_tamil_nadu-2014.csv')
data_heatmap = df.pivot_table(index='Location of Monitoring Station', values='RSPM/PM10')
plt.figure(figsize=(12, 8))
```

```
sns.heatmap(data_heatmap, cmap='YlGnBu', annot=True, fmt=".1f")

plt.xlabel('Time')

plt.ylabel('Location')

plt.title('Air Quality Heatmap')

plt.show()

# Calculate average pollution levels for each area

grouped = data.groupby('City/Town/Village/Area')[['SO2', 'NO2', 'RSPM/PM10']].mean()

print(grouped)
```

Example Outputs:

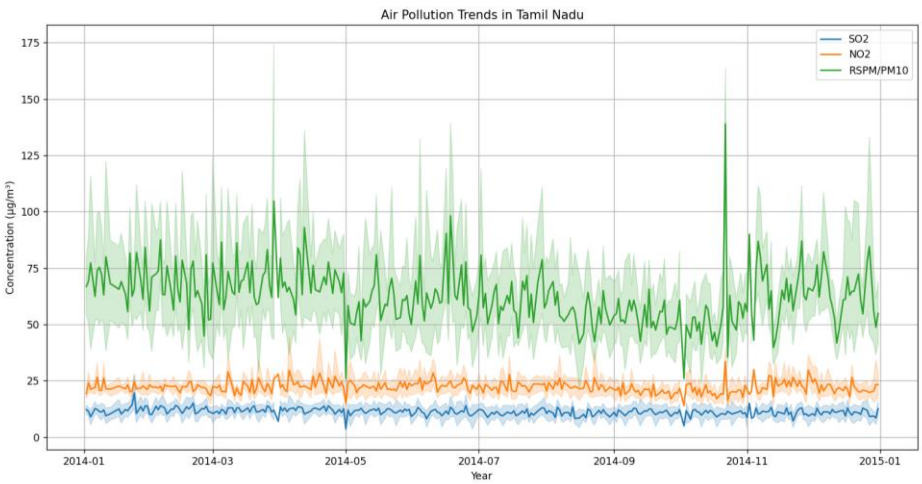
1. Average Pollution Levels:

- Mean concentrations of SO2, NO2, and RSPM/PM10 for different areas.

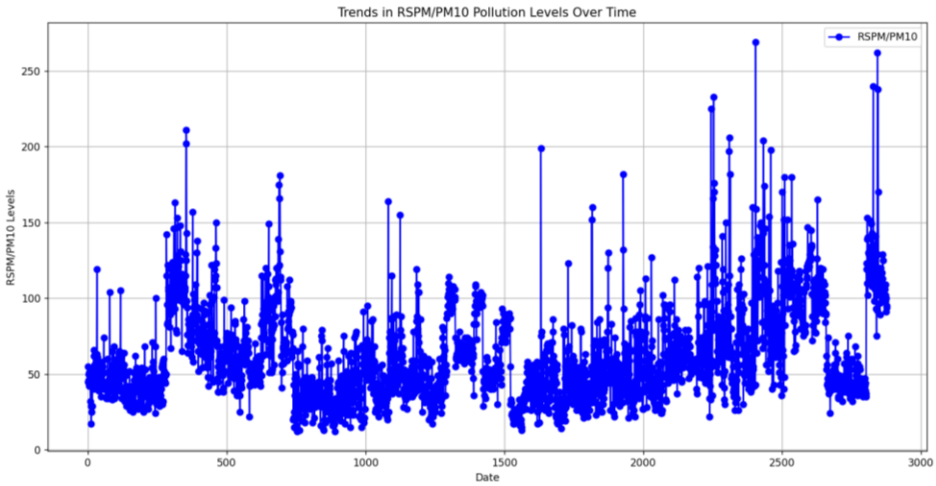
```
                               SO2        NO2  RSPM/PM10
City/Town/Village/Area
Chennai                  13.014042  22.088442  58.998000
Coimbatore                4.541096  25.325342  49.217241
Cuddalore                 8.965986  19.710884  61.881757
Madurai                  13.319728  25.768707  45.724490
Mettur                    8.429268  23.185366  52.721951
Salem                     8.114504  28.664122  62.954198
Thoothukudi              12.989691  18.512027  83.458904
Trichy                   15.293956  18.695055  85.054496
```

2. Trend Analysis:

- Line plots showcasing trends in SO2, NO2, and RSPM/PM10 concentrations over the years.

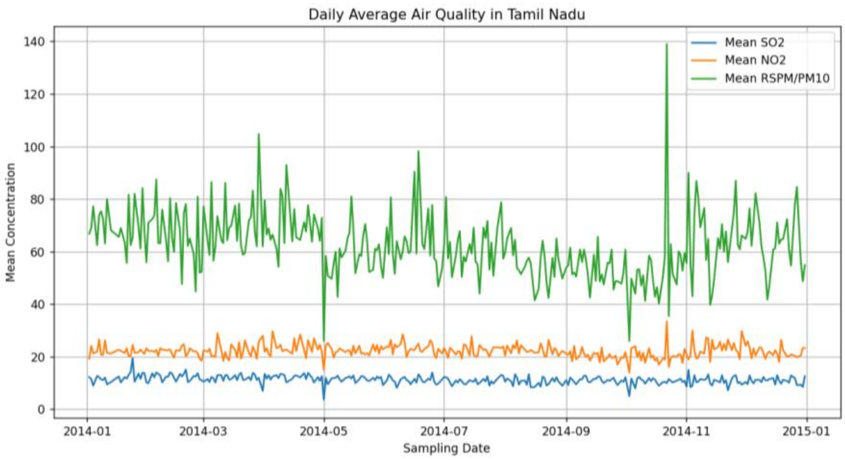


- Specific focus on RSPM/PM10 trends over time.



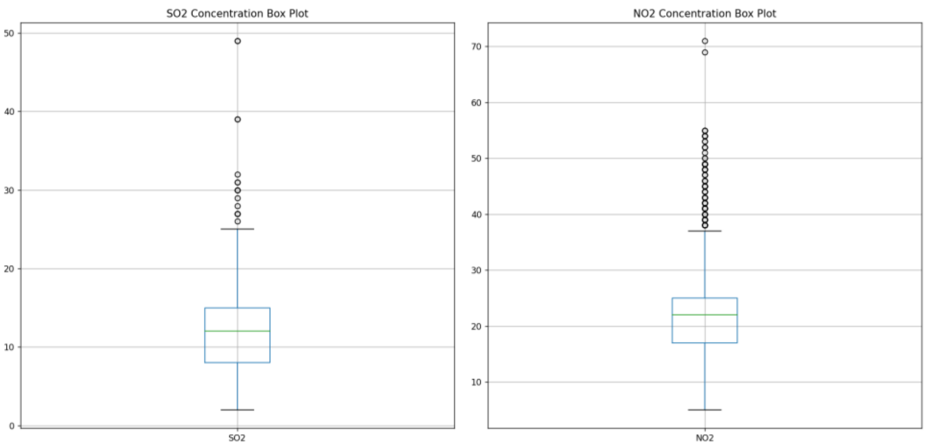
3. Daily Average Air Quality:

— Line plot illustrating the daily average concentrations of SO2, NO2, and RSPM/PM10, providing insights into day-to-day variations.



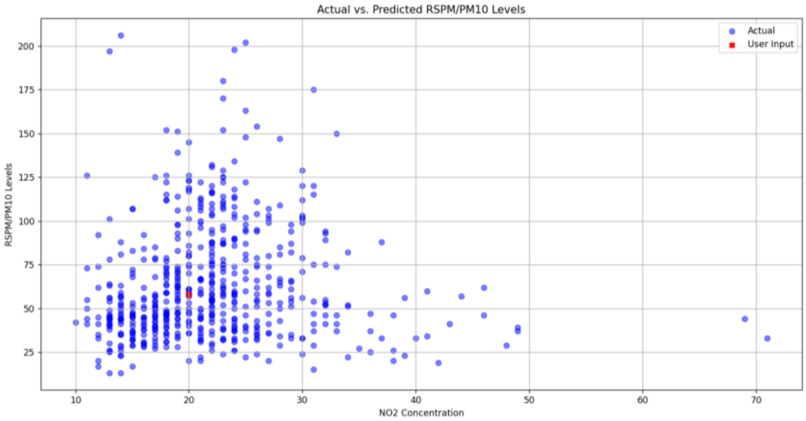
4. Box Plots and Descriptive Statistics:

— Descriptive statistics for SO2 and NO2 concentrations.
— Box plots depicting the distribution of SO2 and NO2 concentrations.



5. Predictive Modeling:

- Actual vs. predicted RSPM/PM10 levels visualization.



- Evaluation metrics such as Mean Squared Error and R-squared.

```
Mean Squared Error: 835.4788249190386
R-squared (Coefficient of Determination): 0.20658507746336507
```

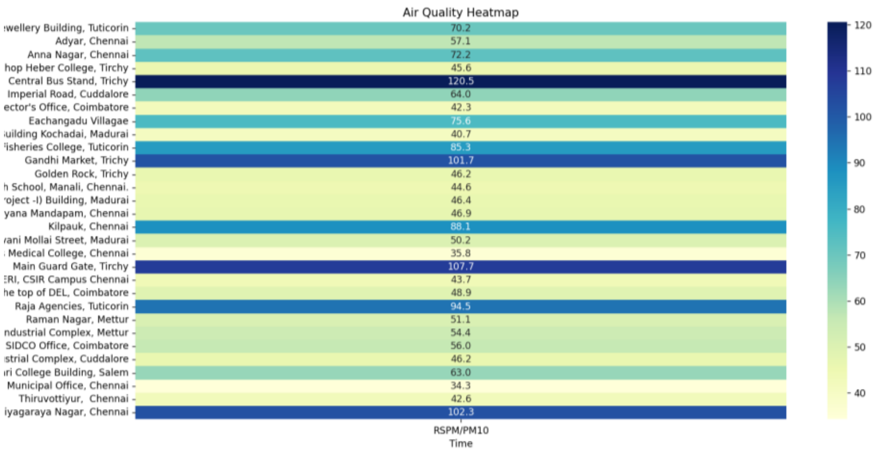
6. User Input and Prediction:

- User input for SO2 and NO2 concentrations.
- Predicted RSPM/PM10 level based on user input.

```
Enter SO2 concentration (µg/m³): 10
Enter NO2 concentration (µg/m³): 20
Predicted RSPM/PM10 Level: 57.79
```

7. Visualization Techniques:

- Heatmap illustrating pollutant levels by location and time.



Insights:

The analysis provides valuable insights into air pollution trends and pollution levels in Tamil Nadu:

- Identifying Pollution Hotspots: Average pollution levels help identify regions with consistently high concentrations of pollutants.

- Temporal Trends: Trend analysis reveals how pollution levels have evolved over time, contributing to a deeper understanding of the factors influencing air quality.

- Prediction Tool: The predictive model allows for estimating RSPM/PM10 levels based on user input, providing a practical tool for decision-makers.

- Spatial Distribution: The heatmap visualizes pollution levels across monitoring stations, offering insights into spatial patterns.

This comprehensive analysis contributes to informed decision-making for environmental management and public health initiatives in Tamil Nadu.

Conclusion:

In conclusion, this multifaceted analysis provides valuable insights into air pollution trends and pollutant levels in Tamil Nadu. The examination of average concentrations, trend patterns, and predictive modeling collectively contributes to a nuanced understanding of the state's air quality landscape. Daily average air quality visualizations and box plots offer detailed perspectives on short-term variations and concentration distributions. The user interactivity aspect allows for broader applicability, engaging stakeholders in the assessment process. These findings collectively empower decision-makers, environmental agencies, and the public with actionable information, facilitating targeted interventions and policies to address and mitigate air pollution in Tamil Nadu for the betterment of public health and the environment.