Perform the air quality analysis and create visualizations.

Calculate average SO2, NO2, and RSPM/PM10 levels across different monitoring stations,cities, or areas. Identify pollution trends and areas with high pollution levels.

Create visualizations using data visualization libraries .

Introduction:

Introduce the purpose and significance of the air quality analysis. Explain the importance of monitoring SO2, NO2, and RSPM/PM10 levels in various locations. Mention the potential health and environmental impacts of air pollution.

Process:

1. \*\*Data Collection:\*\*

- Acquire air quality data from reliable sources such as government agencies, environmental organizations, or research institutions.

- Gather data for SO2, NO2, and RSPM/PM10 levels across different monitoring stations, cities, or areas.

2. \*\*Data Preprocessing:\*\*

- Handle missing or erroneous data points.

- Convert data into a suitable format for analysis (e.g., CSV, Excel).

- Ensure data quality and consistency.

3. \*\*Data Analysis:\*\*

- Calculate average SO2, NO2, and RSPM/PM10 levels across different categories, such as monitoring stations, cities, or areas.

- Identify trends and patterns in the data to understand pollution variations over time and space.

- Use statistical methods to analyze the data, including means, medians, and standard deviations.

Python program:

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

data = pd.read\_csv('air\_quality\_data.csv')

data.dropna(inplace=True)

avg\_so2 = data.groupby('City')['SO2'].mean()

avg\_no2 = data.groupby('City')['NO2'].mean()

avg\_rspm\_pm10 = data.groupby('City')['RSPM/PM10'].mean()

plt.figure(figsize=(12, 6))

plt.subplot(2, 2, 1)

sns.barplot(x=avg\_so2.index, y=avg\_so2.values)

plt.title('Average SO2 Levels by City')

plt.subplot(2, 2, 2)

sns.barplot(x=avg\_no2.index, y=avg\_no2.values)

plt.title('Average NO2 Levels by City')

plt.subplot(2, 2, 3)

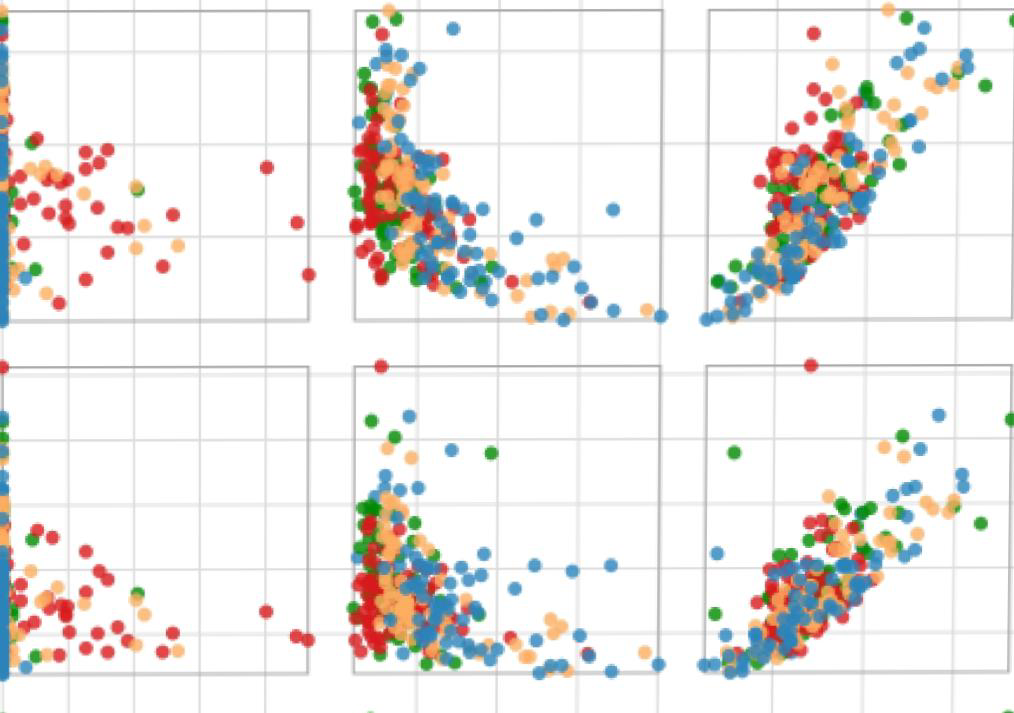
sns.barplot(x=avg\_rspm\_pm10.index, y=avg\_rspm\_pm10.values)

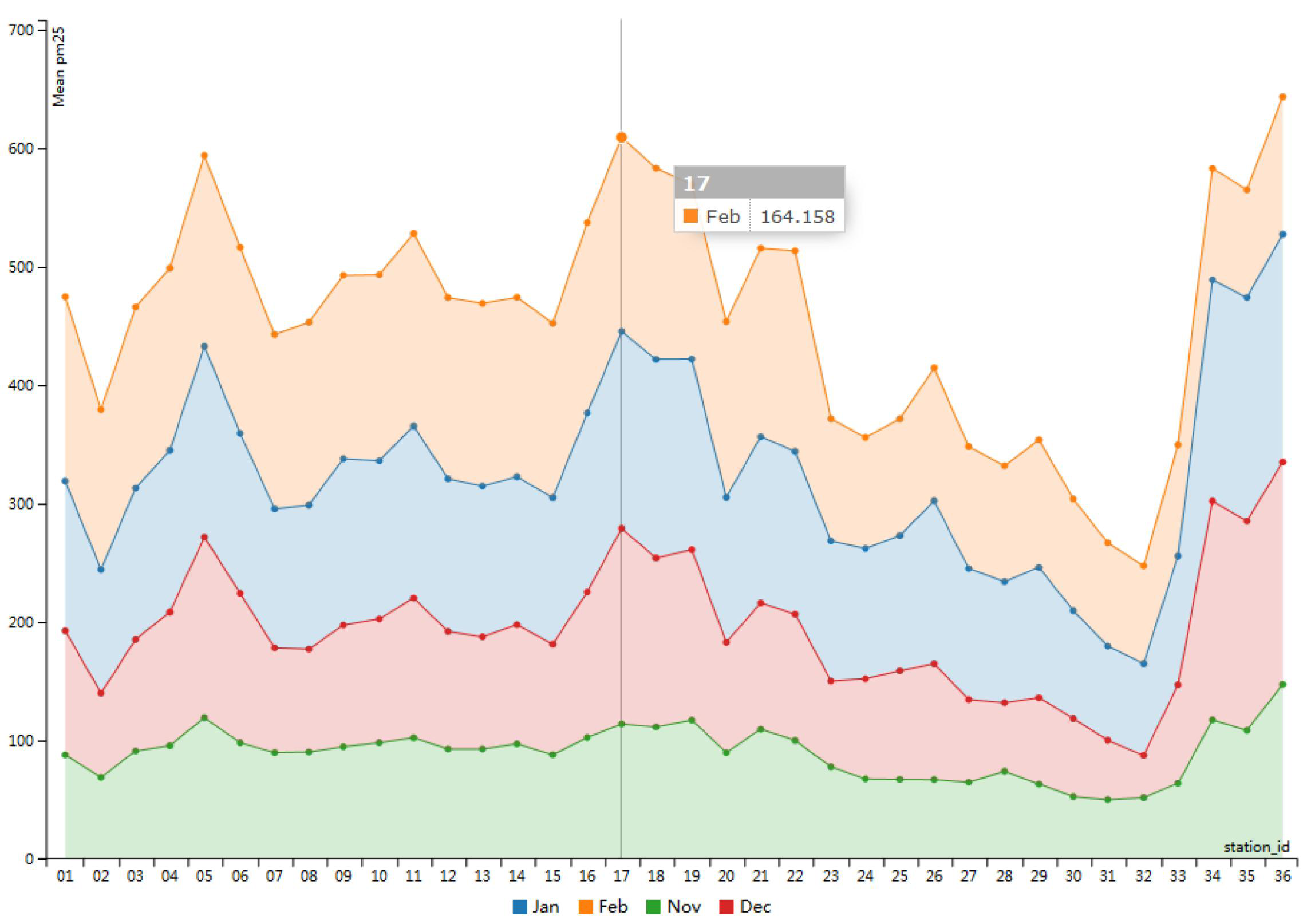
plt.title('Average RSPM/PM10 Levels by City')

plt.tight\_layout()

plt.show()

Output:





Algorithm:

1. necessary Python libraries (Pandas, NumPy, Matplotlib/Seaborn).

2. Load the air quality data into a Pandas DataFrame.

3. Data Preprocessing:

- Handle missing values.

- Filter and categorize data as needed.

4. Calculate the average SO2, NO2, and RSPM/PM10 levels across different categories (stations, cities, areas).

5. Data Visualization:

- Create visualizations using Matplotlib or Seaborn to present the results. Possible visualizations include bar charts, line plots, and heatmaps to show pollution trends.

6. Interpret the visualizations:

- Explain the insights gained from the visualizations. Identify areas with high pollution levels and trends in pollutant concentrations.

Conclusion:

Summarize the findings of the air quality analysis and the significance of the results. Discuss the implications for public health and the environment. Highlight any actionable recommendations for addressing pollution in specific areas. Emphasize the importance of continued monitoring and data-driven decision-making to improve air quality.