AIR QUALITY ANAYLSIS IN TAMIL NADU

Problem Statement:

- The state of Tamil Nadu faces significant challenges related to air quality, characterized by the presence of air pollutants that exceed acceptable levels, posing health risks to its residents and impacting the environment.

Problem Solutions:

Data collection and Monitoring:

- Enhance air quality monitoring infrastructure.
- Collect and analyze data to identify pollution sources and hotspots.

Emission Reduction Policies:

- Implement and enforce stricter emissions standards for industries and vehicles.
- Promote the use of cleaner fuels and technologies.

Promote Public Transportation:

- Invest in efficient and affordable public transportation systems to reduce the number of vehicles on the road.

Green Energy Sources:

- Encourage the adoption of renewable energy sources to reduce reliance on fossil fuels.

Reforestation and Green Spaces:

- Increase green spaces, parks, and tree planting to absorb pollutants and improve air quality.

Awareness and Education:

- Educate the public about the importance of reducing pollution and adopting sustainable practices.

Regulate Construction:

- Implement construction regulations to reduce dust and emissions from building activities.
- Encourage sustainable urban planning and green building practices.

Industrial Compliance:

- Ensure industries comply with pollution control regulations.
- Promote cleaner production processes.

Community Engagement:

- Engage communities in air quality improvement efforts.
- Encourage citizen initiatives like carpooling and reducing waste.

Government Support:

- Allocate funds for air quality improvement projects.
- Collaborate with other regions and countries to address cross-border pollution issues.

Healthcare and Awareness:

- Improve healthcare infrastructure to handle pollution-related illnesses.
- Raise awareness about the health risks of poor air quality.

Research and Innovation:

- Invest in research for innovative pollution control technologies.
- Support startups and businesses working on air quality solutions.

Data Visualization:

- Utilize data visualization to represent air quality data effectively.
- The library used for data visualization is Matplotlib.

Innovation:

Improving air quality analysis in Tamil Nadu, like in many other regions, is crucial for public health and environmental protection. Here are some innovative ideas and technologies that could be implemented to enhance air quality analysis in Tamil Nadu:

Distributed Air Quality Monitoring:

Deploy a network of low-cost, portable air quality sensors throughout Tamil Nadu. These sensors can provide real-time data and be connected to a centralized database for easy access and analysis by researchers and the public.

Mobile Apps for Air Quality:

Develop mobile applications that allow residents to access real-time air quality data specificto their location. These apps can also provide health advisories and tips on how to reduce exposure to poor air quality.

Satellite Imagery and Remote Sensing:

Utilize satellite imagery and remote sensing technology to monitor air quality from space. This can provide a broader perspective on the region's air quality, especially in remote or rural areas.

Predictive Modeling:

Implement advanced predictive modeling techniques to forecast air quality conditions.Machine learning algorithms can use historical data, weather patterns, and other variables to provideaccurate short-term and long-term air quality predictions.

Air Quality Index (AQI) Improvements:

Enhance the existing AQI system with more pollutants, sub-indices for different areas within Tamil Nadu, and an easy-to-understand color-coded scale. This can help the public better understand and respond to air quality information.

Public Awareness Campaigns:

Launch public awareness campaigns to educate residents about the health impacts of poor air quality and how they can contribute to reducing pollution through individual actions.

Collaboration with Academic Institutions:

Partner with universities and research institutions to conduct in-depth studies on air quality in Tamil Nadu. This collaboration can lead to the development of new monitoring technologies and research-driven policy recommendations.

Green Infrastructure:

Promote green infrastructure projects, such as urban forests, green roofs, and vertical gardens, to help absorb pollutants and improve air quality in urban areas.Electric and Green Transport Initiatives:

Encourage the adoption of electric vehicles and the development of efficient public transportation systems to reduce emissions from the transportation sector, a significant contributor to air pollution.

Air Quality Data Visualization:

Create user-friendly data visualization tools that make air quality data easily understandablefor policymakers, researchers, and the general public. These tools can include interactive maps, charts, and dashboards.

Collaboration with Industry:

Work closely with industries to implement cleaner production technologies and ensure compliance with emission standards. This can reduce industrial pollution in the region.

Air Quality Drones:

Employ drones equipped with air quality sensors to monitor hard-to-reach or pollution-prone areas, such as industrial zones or construction sites.

Community Engagement:

Engage local communities in air quality monitoring and decision-making processes. Empower citizens to take action to improve air quality in their neighbour hoods.

Early Warning Systems:

Develop early warning systems that can notify residents and authorities when air quality levels become hazardous, allowing them to take protective measures.

By combining these innovative approaches and technologies, Tamil Nadu can significantly improve its air quality analysis, leading to better-informed decisions and ultimately a healthier environment forits residents.

Development Part 1:

Introduction:

Analysing air quality in Tamil Nadu is crucial for both environmental and public health reasons. The state of Tamil Nadu, located in southern India, has experienced rapid industrialization and urbanization in recent years, which has led to various air quality challenges. To undertake a comprehensive air quality analysis for development, it is important to consider several key aspects and steps.

1. Data Collection:

Monitoring Stations:

Establish a network of air quality monitoring stations across Tamil Nadu. These stations should be strategically located in urban, industrial, and rural areas to capture a representative sample of air quality conditions.

Parameters:

Measure various air quality parameters, including particulate matter (PM2.5 and PM10), nitrogen dioxide (NO_2), sulfur dioxide (SO_2), carbon monoxide (CO), ozone (O_3), and volatile organic compounds (VOCs).

Meteorological Data:

Collect meteorological data, such as temperature, humidity, wind speed, and wind direction, as these factors can influence air quality.

Historical Data:

Gather historical air quality data to establish trends and identify areas with chronic air quality problems.

2. Data Analysis:

Air Quality Index (AQI):

Calculate the AQI for different locations in Tamil Nadu to provide a clear and understandable representation of air quality to the public.

Identify Hotspots:

Identify areas with consistently poor air quality, such as major cities or industrial zones, and pinpoint the key pollutants responsible.

Seasonal Trends:

Analyze seasonal variations in air quality, as well as the factors contributing to these variations, such as agricultural burning, weather conditions, or industrial activity.

3. Pollution Sources:

Industrial Emissions:

Examine emissions from industrial facilities, such as factories and power plants, and assess compliance with emission standards.

Vehicle Emissions:

Evaluate the impact of vehicular emissions on air quality, considering the prevalence of different types of vehicles and fuel types.

Agricultural Practices:

Investigate the role of agriculture in air quality, including the use of pesticides and burning of crop residues.

Waste Management:

Assess waste disposal practices and their impact on air quality, especially in urban areas.

4. Health Impact Assessment:

Collaborate with healthcare institutions to study the health effects of poor air quality on the population of Tamil Nadu.

Identify vulnerable groups, such as children, the elderly, and individuals with pre-existing respiratory conditions, and assess their exposure and health outcomes.

5. Policy and Regulation:

Review existing air quality regulations and policies in Tamil Nadu to identify gaps or areas for improvement.

Develop or update regulations to control emissions from various sources, and enforce strict compliance measures.

6. Public Awareness:

Launch public awareness campaigns to educate residents about the health risks associated with poor air quality and ways to protect themselves.

Provide real-time air quality information through websites, apps, and public displays.

7. Mitigation Strategies:

Implement pollution control technologies in industries and encourage the use of cleaner fuels.

Promote sustainable urban planning, public transportation, and green spaces to reduce vehicle emissions and enhance air quality.

Encourage agricultural practices that minimize burning and promote sustainable waste management.

8. International Cooperation:

Collaborate with neighboring states and countries to address transboundary air pollution issues, especially during cross-border events like crop burning.

Input:

```
import os import glob import time
import numpy as np import pandas as pd
import matplotlib.pyplot as plt import seaborn as sns sns.set_theme()
from sklearn.ensemble import ( RandomForestRegressor, GradientBoostingRegressor,
AdaBoostRegressor, HistGradientBoostingRegressor

)
from sklearn.metrics import ( r2_score, mean_squared_error, mean_absolute_error,
mean_absolute_percentage_error
)
from sklearn.model_selection import ( cross_val_score,
TimeSeriesSplit, RandomizedSearchCV
)


import xgboost as xgb
from IPython.display import clear_output


N_JOBS = -1
RANDOM_STATE = 18
DATASET_SRC = '/content/cpcb_dly_aq_tamil_nadu-2014.csv' df_states =
pd.read_csv('/content/cpcb_dly_aq_tamil_nadu-2014.csv')
df_states.drop(columns=['Agency', 'Type of Location', 'SO2'], inplace=True) df_states.head()

df_states.head()
unique_states = df_states['State'].unique() unique_states
def combine_state_df(State): '''
Combine all state files into a single dataframe and attaching the city information.
Parameters


state_name (str): The name of the state Return


df (DataFrame): The combined dataframe from all files of a specific state

'''
state_code = df_states[df_states['State'] == state_name]['file_name'].iloc[0][:2] state_files =
glob.glob('/content/cpcb_dly_aq_tamil_nadu-2014.csv') print(f'Combining a total of {len(state_files)}
files...\n')
combined_df = []
for state_file in state_files:
file_name = state_file.split(f'{DATASET_SRC}/')[1][0:-4] file_df = pd.read_csv(state_file)
file_df['City'] = df_states[df_states['file_name'] == file_name]['City'].values[0] file_df['city'] =
file_df['City'].astype('string')
combined_df.append(file_df) return pd.concat(combined_df)
df_states.info()


def create_dt_index(dataframe):
dataframe = dataframe.drop(columns='To Date')
dataframe['From Date'] = pd.to_datetime(dataframe['From Date']) dataframe =
dataframe.rename(columns={'From Date': 'datetime'}) return dataframe.set_index('datetime')

df_states.head(2)
```

```
df_states.tail(2)
def plot_feature_similarities(dataframe, feature_groups, columns=2): rows =
int((len(feature_groups)/columns)//1)
fig, axes = plt.subplots(rows, columns, figsize=(13, 4*rows)) fig.tight_layout(pad=3.0)

row_num = 0
col_num = 0
for pos, group in enumerate(feature_groups): # Move to new row
if pos % columns == 0 and pos != 0: row_num += 1
col_num = 0

for feature in feature_groups[group]:
df_feature = dataframe[dataframe[feature].notnull()][feature]
df_feature = df_feature.groupby([df_feature.index.year]).mean(numeric_only=True)
sns.lineplot(data=df_feature, label=feature, ax=axes[row_num, col_num])
axes[row_num, col_num].set_title(group) axes[row_num, col_num].set(xlabel=None) col_num += 1

plt.plot()


df_states.isnull().sum().sort_values(ascending=False)


df_states = df_states.dropna(how='all')
df_states = df_states.dropna(how='all', axis='columns')


def get_null_info(dataframe): null_vals = dataframe.isnull().sum()

df_null_vals = pd.concat({'Null Count': null_vals,
'Percent Missing (%)': round(null_vals * 100 / len(dataframe), 2)}, axis=1) return
df_null_vals.sort_values(by=['Null Count'], ascending=False)

df_null_info = get_null_info(df_states)


plt.figure(figsize=(8, 10))
sns.barplot(data=df_null_info, x='Percent Missing (%)', y=df_null_info.index, orient='h',
color='steelblue')
plt.show()


def get_overall_ds_info(): features = {} total_records = 0

for i, state_name in enumerate(unique_states): clear_output(wait=False)

print(f"Processing state of {state_name} ({i+1}/{len(unique_states)})")


temp_df = combine_state_df(state_name) # Get combined state dataframe temp_df =
create_dt_index(temp_df)          # Create datetime index
temp_df = temp_df.dropna(how='all')     # Drop empty rows


comparisons = get_null_info(temp_df)


total_records += df.shape[0]
```

```
for feature in comparisons.index: if feature in features:
features[feature] += comparisons.loc[[feature]]['Null Count'].values[0] else:
features[feature] = comparisons.loc[[feature]]['Null Count'].values[0]


ds_null_info = pd.DataFrame.from_dict(features, orient='index', columns=['Null Count'])
ds_null_info['Percent Missing (%)'] = round(ds_null_info['Null Count'] * 100 / total_records, 2)
ds_null_info = ds_null_info.sort_values(by=['Null Count'], ascending=False) return ds_null_info

threshold = 0.6
df_states = df_states.dropna(thresh=df_states.shape[0]*threshold, axis=1) get_null_info(df_states)

get_null_info(df)def plot_features_by_group(features, df_states): for feature in features:
fig, ax = plt.subplots(1, 1, figsize=(12, 4)) fig.suptitle(feature)

labels = []
for i, (group, group_df) in enumerate(slice_groups.items()):
data_slice = group_df[group_df.columns.intersection(pollutants[feature])]


# Keep only the NOx feature, as it combines both NO (Nitrogen Oxide) and NO2 (Nitrogen

Dioxide)
if feature == "Nitrogen Compounds":
data_slice = data_slice.drop(['NO (ug/m3)', 'NO2 (ug/m3)'], axis=1)


data_slice.plot(kind="line", ax=ax) for column in data_slice.columns:
labels.append(f'{column} [{group}]') ax.set(xlabel=None)
ax.legend(labels) plt.plot()
get_null_info(df_states)
df = df_states.interpolate(method='pad') df = df_states.fillna(df_states.mean()) df.info()
df_states.plot.bar()
```

Output:

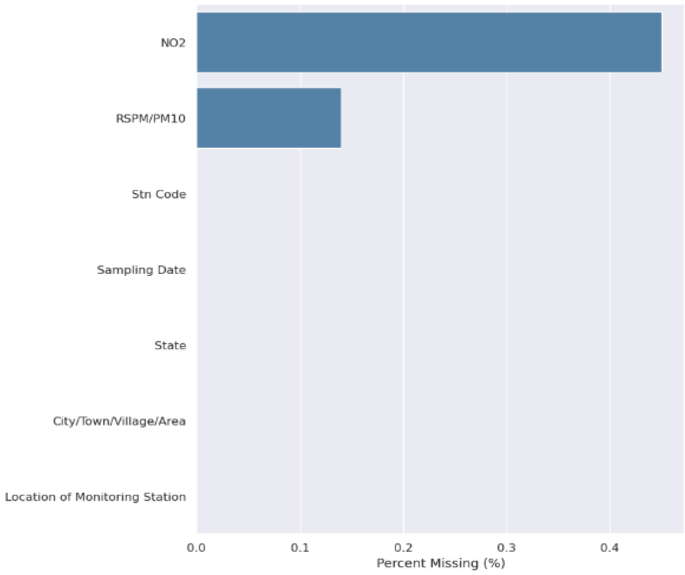
	Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	NO2	RSPM/PM10	PM 2.5
0	38	01-02-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	17.0	55.0	NaN
1	38	01-07-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	17.0	45.0	NaN
2	38	21-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	18.0	50.0	NaN
3	38	23-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	16.0	46.0	NaN
4	38	28-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	14.0	42.0	NaN

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2879 entries, 0 to 2878
Data columns (total 8 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Stn Code                        2879 non-null   int64
 1   Sampling Date                   2879 non-null   object
 2   State                           2879 non-null   object
 3   City/Town/Village/Area          2879 non-null   object
 4   Location of Monitoring Station  2879 non-null   object
 5   NO2                             2866 non-null   float64
 6   RSPM/PM10                       2875 non-null   float64
 7   PM 2.5                          0 non-null      float64
dtypes: float64(3), int64(1), object(4)
memory usage: 180.1+ KB
```

	Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	NO2	RSPM/PM10	PM 2.5
0	38	01-02-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	17.0	55.0	NaN
1	38	01-07-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	17.0	45.0	NaN

	Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	NO2	RSPM/PM10	PM 2.5
2877	773	24-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	17.0	95.0	NaN
2878	773	31-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	16.0	94.0	NaN

```
PM 2.5                            2879
NO2                                 13
RSPM/PM10                            4
Stn Code                             0
Sampling Date                        0
State                                0
City/Town/Village/Area               0
Location of Monitoring Station       0
dtype: int64
```

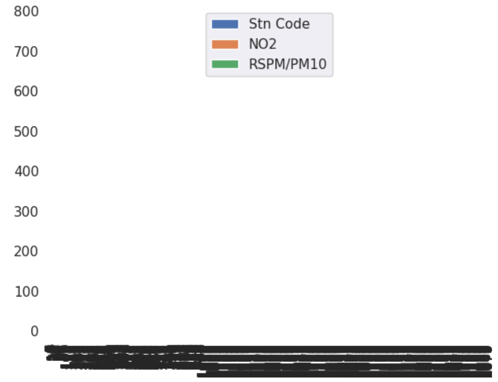


	Null Count	Percent Missing (%)
NO2	13	0.45
RSPM/PM10	4	0.14
Stn Code	0	0.00
Sampling Date	0	0.00
State	0	0.00
City/Town/Village/Area	0	0.00
Location of Monitoring Station	0	0.00

	Null Count	Percent Missing (%)
NO2	13	0.45
RSPM/PM10	4	0.14
Stn Code	0	0.00
Sampling Date	0	0.00
State	0	0.00
City/Town/Village/Area	0	0.00
Location of Monitoring Station	0	0.00

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2879 entries, 0 to 2878
Data columns (total 7 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Stn Code                        2879 non-null   int64
 1   Sampling Date                   2879 non-null   object
 2   State                           2879 non-null   object
 3   City/Town/Village/Area          2879 non-null   object
 4   Location of Monitoring Station  2879 non-null   object
 5   NO2                             2879 non-null   float64
 6   RSPM/PM10                       2879 non-null   float64
dtypes: float64(2), int64(1), object(4)
memory usage: 157.6+ KB
<ipython-input-110-4697573c63a4>:2: FutureWarning: The default value of numeric_only in DataFrame.mean is deprecated. In a future version, it will defaul
  df = df_states.fillna(df_states.mean())
```

```
<Axes: >
```




```
[8] df_states.head()
```

	Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	NO2	RSPM/PM10	PM 2.5
0	38	01-02-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	17.0	55.0	NaN
1	38	01-07-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	17.0	45.0	NaN
2	38	21-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	18.0	50.0	NaN
3	38	23-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	16.0	46.0	NaN
4	38	28-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	14.0	42.0	NaN

```
df_states.tail()
```

	Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	NO2	RSPM/PM10	PM 2.5
2874	773	12-03-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	18.0	102.0	NaN
2875	773	12-10-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	14.0	91.0	NaN
2876	773	17-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	22.0	100.0	NaN
2877	773	24-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	17.0	95.0	NaN
2878	773	31-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	16.0	94.0	NaN

Development Part 2:

Key Factors Affecting Air Quality in Tamil Nadu:

Industrial Emissions:

Tamil Nadu is home to a significant number of industries, including automobile manufacturing,textile, and petrochemicals.These industries often emit pollutants like particulate matter, sulfur dioxide (SO_2), and nitrogen oxides (NO_x), which can severely affect air quality.

Urbanization:

Rapid urbanization and population growth in cities like Chennai, Coimbatore, and Madurai have led to increased vehicular emissions, construction dust, and solid waste generation, all of which contribute to poor air quality.

Agricultural Activities:

In rural areas, agriculture is a significant contributor to air pollution due to the use of fertilizers, pesticides, and crop residue burning. The latter is a major concern during certain seasons.

Natural Factors:

Weather conditions, such as temperature inversions, can trap pollutants close to the ground, exacerbating air quality issues.
The state's proximity to the Bay of Bengal also influences air quality, with sea breezes sometimes helping disperse pollutants.

Steps Taken for Air Quality Improvement:

Regulatory Measures:

The Tamil Nadu Pollution Control Board (TNPCB) enforces air quality standards and regulations. It has implemented measures to control industrial emissions and monitor compliance.

Promotion of Green Transport:

Encouraging the use of public transportation, electric vehicles, and cycling can help reduce vehicular emissions. Initiatives such as the Chennai Metro have been introduced to ease traffic congestion.

Waste Management:

Proper waste management and recycling can reduce the release of air pollutants from landfills and open burning. Tamil Nadu has been promoting the "Swachh Bharat" campaign to address waste issues.

Crop Residue Management:

To combat crop residue burning, the government can promote alternatives like the use of crop residue for bioenergy or organic manure, and provide incentives to farmers for adopting these practices.

Greenery and Urban Planning:

Increasing green cover in urban areas can help absorb pollutants and improve air quality. Urban planning should prioritize green spaces and efficient land use.

Air Quality Monitoring:

Expanding and improving air quality monitoring networks across the state is essential for accurate data collection and timely action against pollution sources.

Public Awareness:

Awareness campaigns to educate the public about the importance of clean air and ways to reduce pollution are crucial. Citizen engagement is a valuable tool in the fight against air pollution.

Policy Integration:

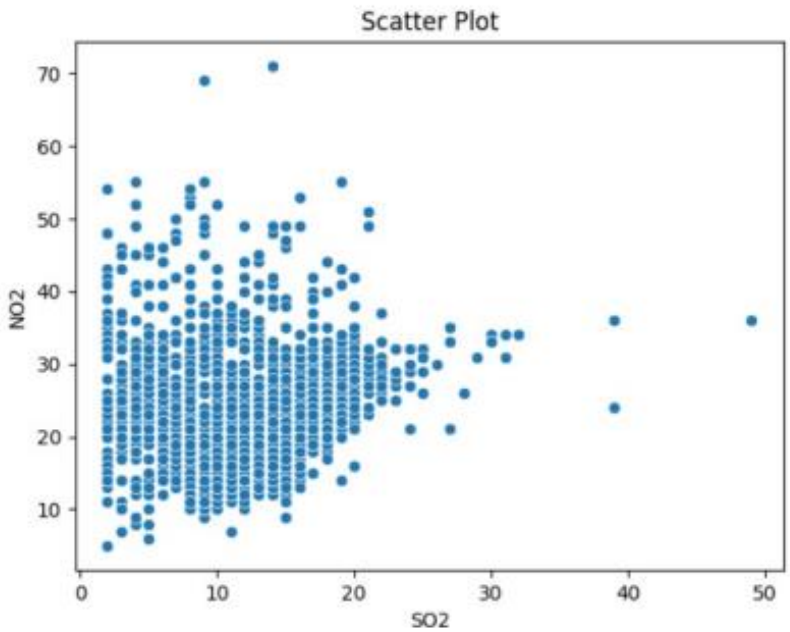
Coordinating air quality management with other environmental policies, such as water resource management and land-use planning, can lead to more comprehensive solutions.

Input & Output:

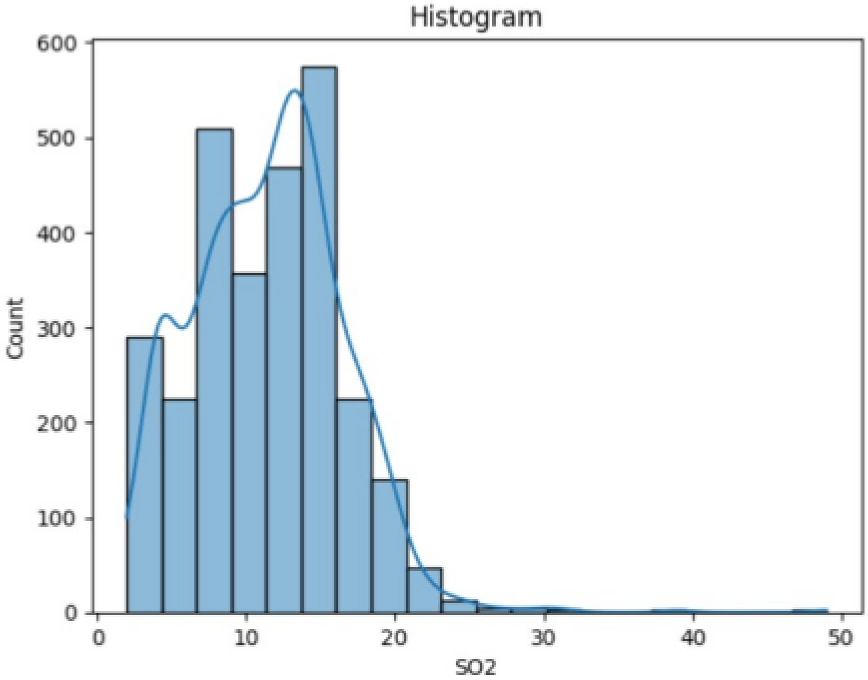
```
[1] import pandas as pd
    import matplotlib.pyplot as plt
    import seaborn as sns
```

```
[2] file_path = "/content/cpcb_dly_aq_tamil_nadu-2014.csv"
    df = pd.read_csv(file_path)
```

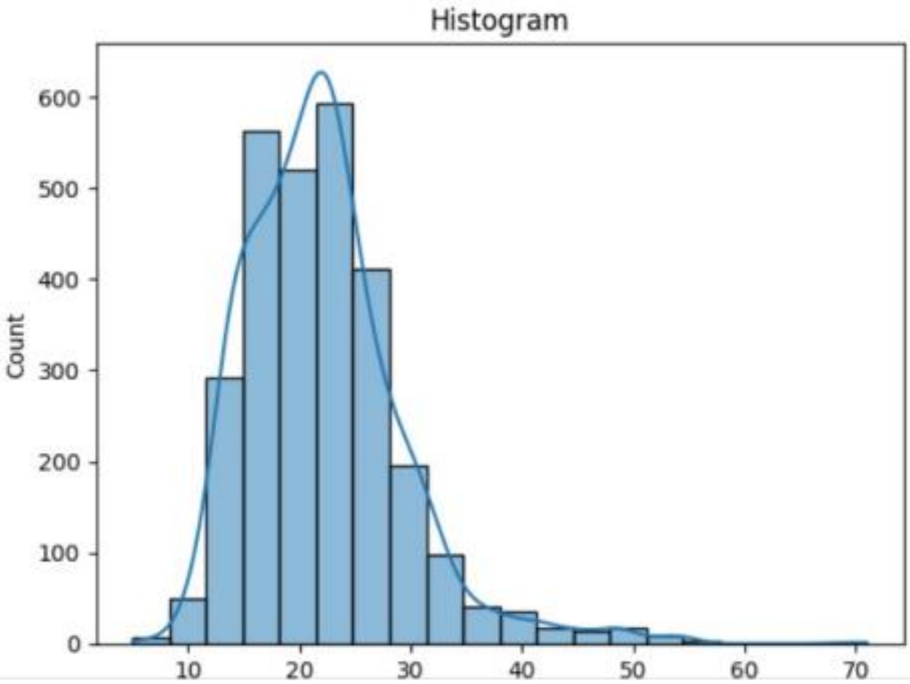
```
[3] sns.scatterplot(data=df, x='SO2', y='NO2')
    plt.title('Scatter Plot')
    plt.show()
```



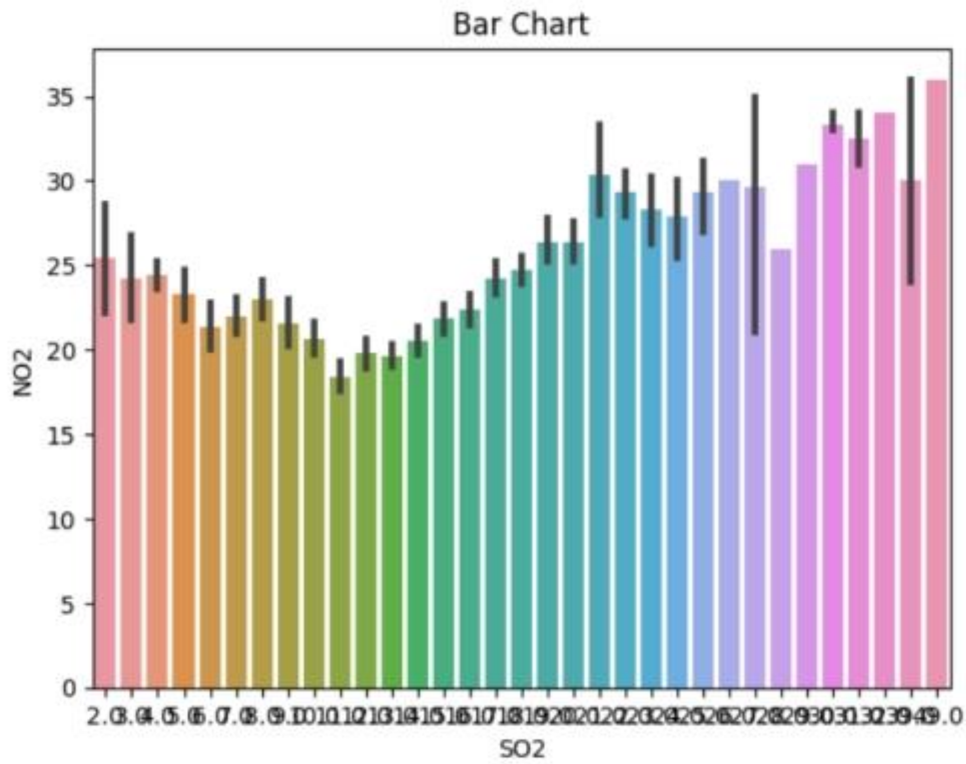
```
[4] sns.histplot(data=df, x='SO2', bins=20, kde=True)
    plt.title('Histogram')
    plt.show()
```




```
[5] sns.histplot(data=df, x='NO2', bins=20, kde=True)
    plt.title('Histogram')
    plt.show()
```



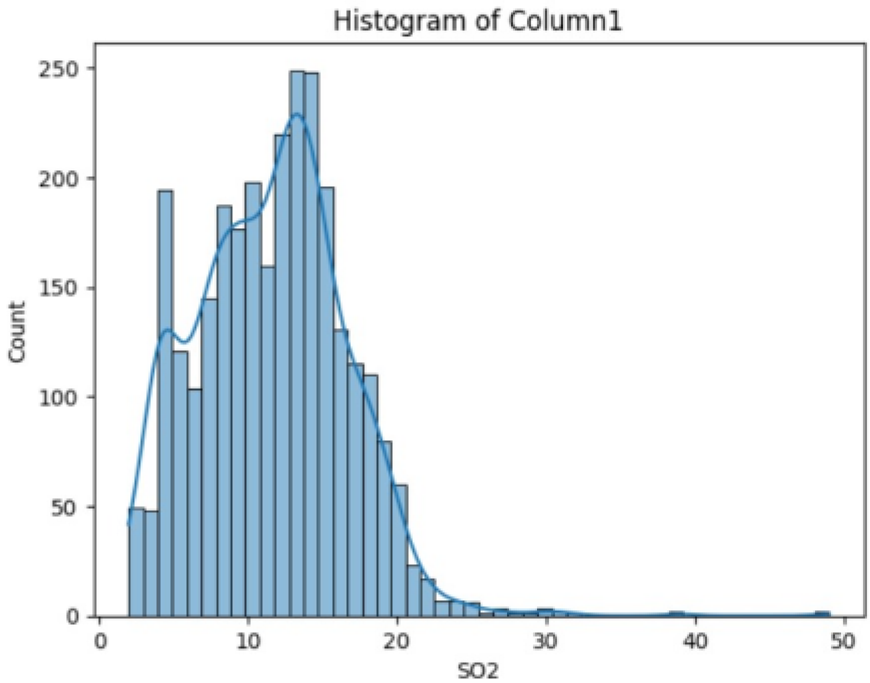
```python
sns.barplot(data=df, x='SO2', y='NO2')
plt.title('Bar Chart')
plt.show()
```



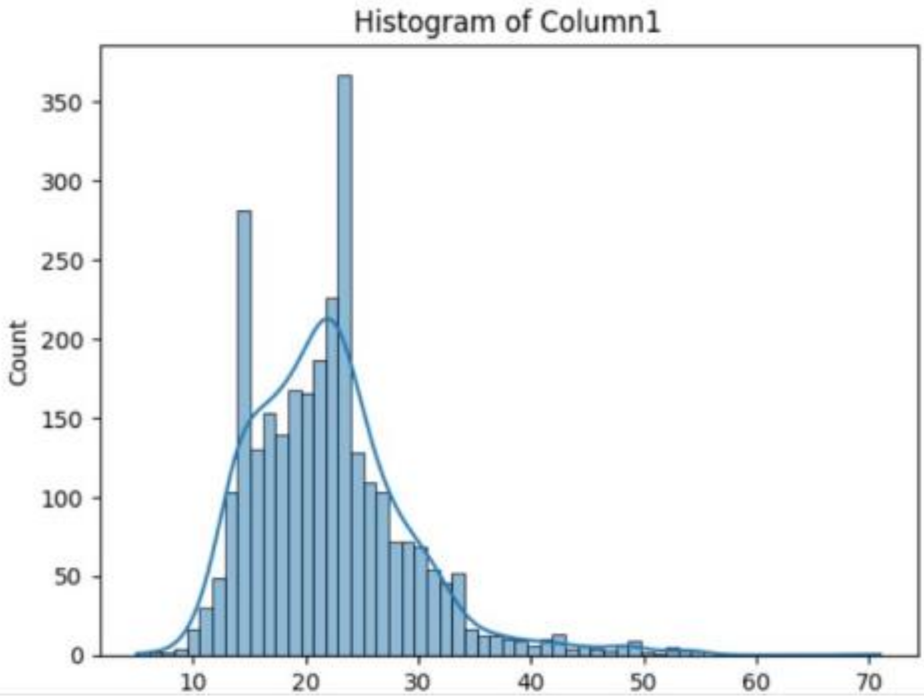

```python
grouped = df.groupby(['State', 'City/Town/Village/Area', ])
averages = grouped[['SO2', 'NO2', 'RSPM/PM10']].mean()
averages = averages.reset_index()
print(averages)
```

```
       State City/Town/Village/Area        SO2        NO2  RSPM/PM10
0  Tamil Nadu                Chennai  13.014042  22.088442  58.998000
1  Tamil Nadu             Coimbatore   4.541096  25.325342  49.217241
2  Tamil Nadu              Cuddalore   8.965986  19.710884  61.881757
3  Tamil Nadu                Madurai  13.319728  25.768707  45.724490
4  Tamil Nadu                 Mettur   8.429268  23.185366  52.721951
5  Tamil Nadu                  Salem   8.114504  28.664122  62.954198
6  Tamil Nadu            Thoothukudi  12.989691  18.512027  83.458904
7  Tamil Nadu                 Trichy  15.293956  18.695055  85.054496
```

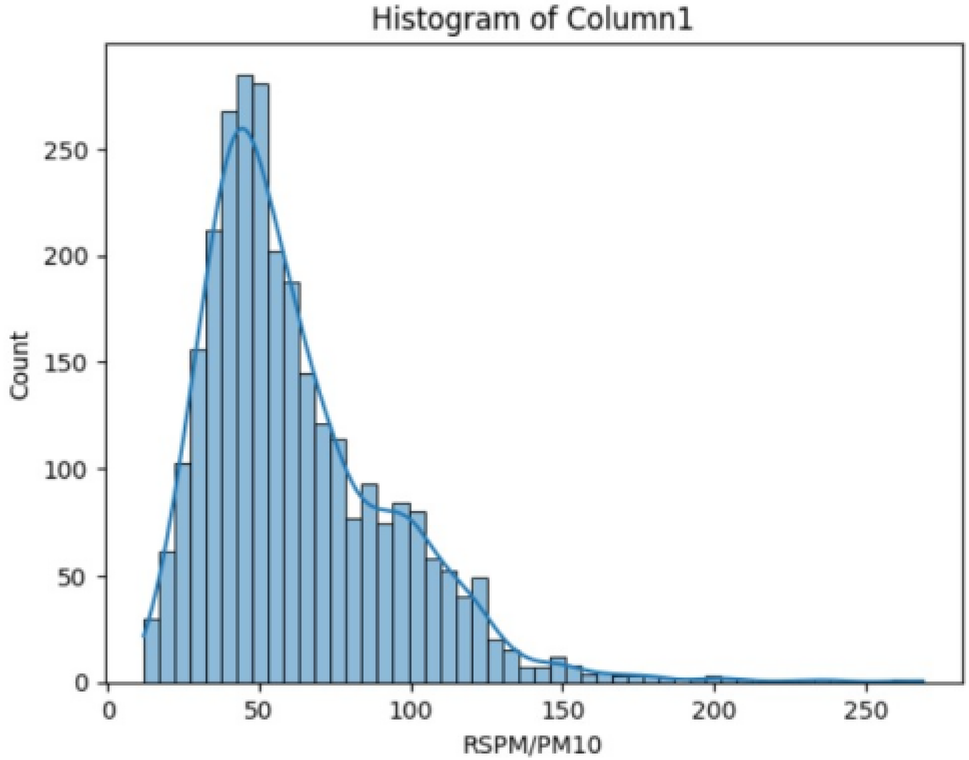
```
sns.histplot(data=df, x='SO2', kde=True)
plt.title('Histogram of Column1')
plt.show()
```



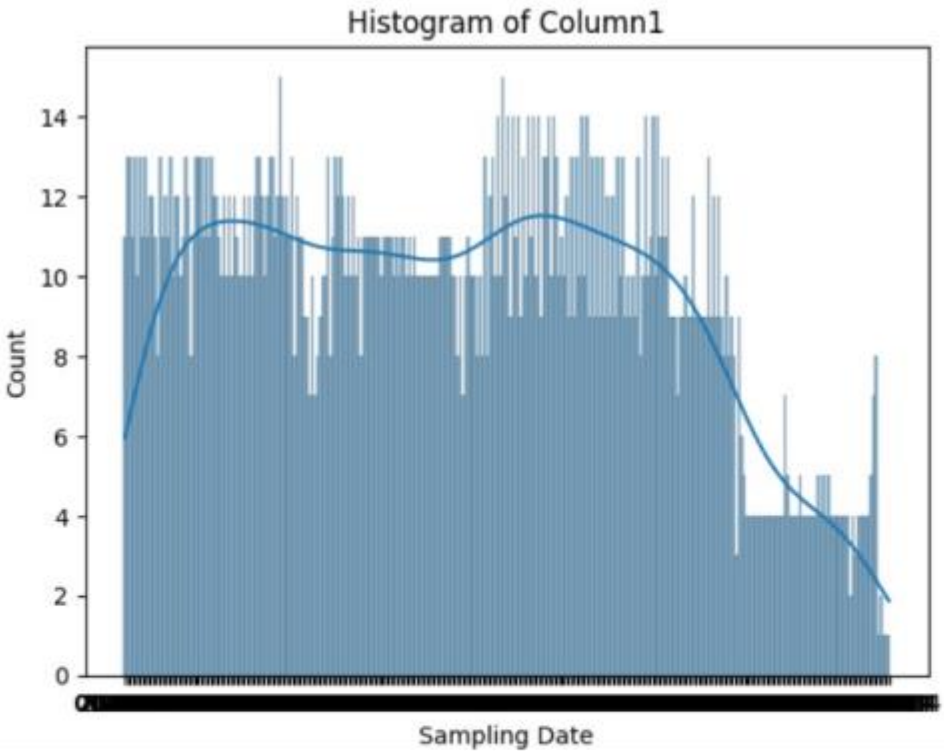

```
[10] sns.histplot(data=df, x='NO2', kde=True)
     plt.title('Histogram of Column1')
     plt.show()
```



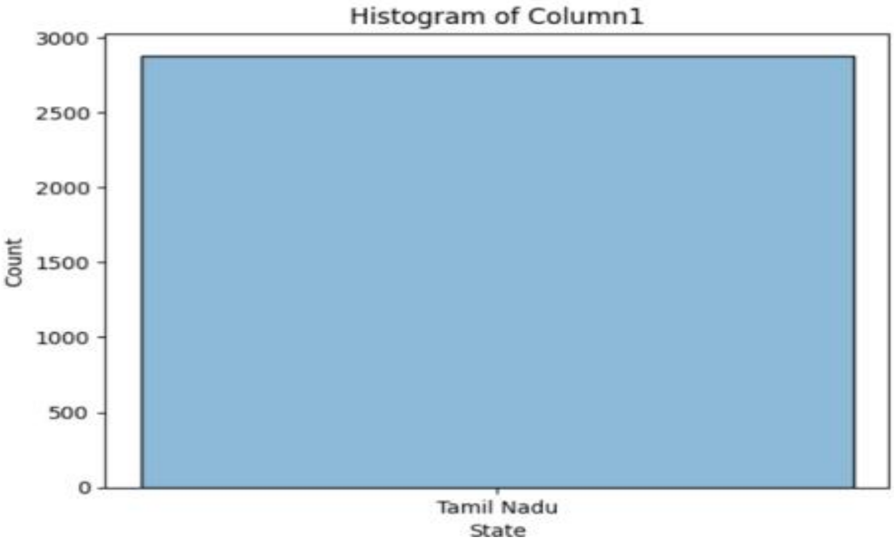
```
sns.histplot(data=df, x='RSPM/PM10', kde=True)
plt.title('Histogram of Column1')
plt.show()
```




```
sns.histplot(data=df, x='Sampling Date', kde=True)
plt.title('Histogram of Column1')
plt.show()
```

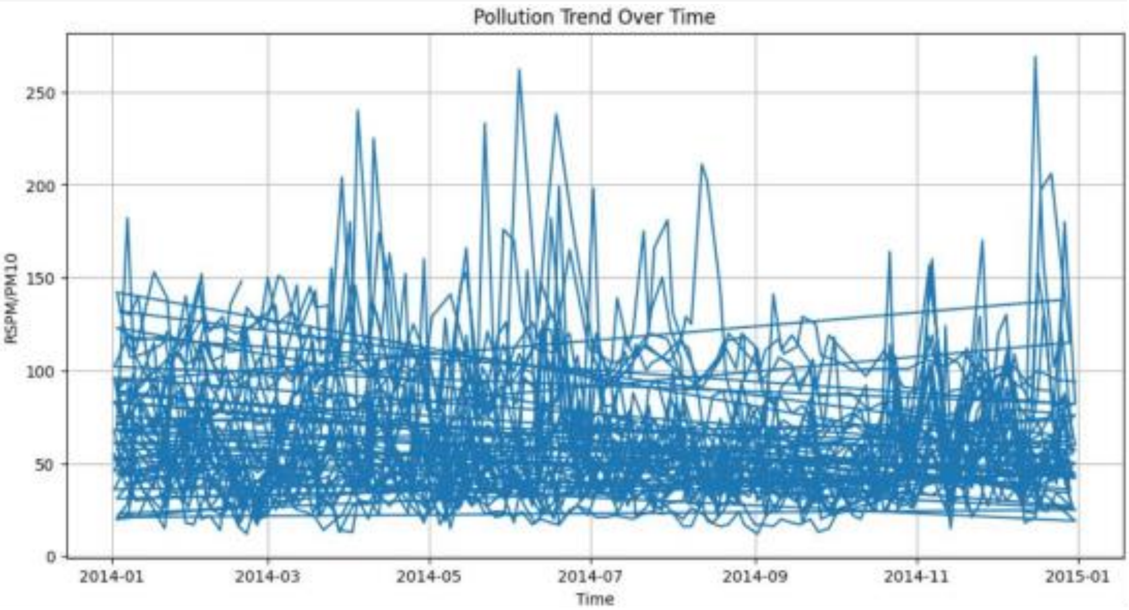


```
sns.histplot(data=df, x='State', kde=True)
plt.title('Histogram of Column1')
plt.show()
```




```
[15] df['Sampling Date'] = pd.to_datetime(df['Sampling Date'])
```

```
plt.figure(figsize=(12, 6))
plt.plot(df['Sampling Date'], df['RSPM/PM10'])
plt.title('Pollution Trend Over Time')
plt.xlabel('Time')
plt.ylabel('RSPM/PM10')
plt.grid()
plt.show()
```



how the analysis provides insights into air pollution trends and pollution levels in Tamil Nadu?

To analyze air pollution trends and pollution levels in Tamil Nadu, you would typically employ a combination of data collection, measurement, and analysis techniques. Here's a step-by-step explanation of how this can be done:

Data Collection:

Gather data from various sources, including government agencies, environmental organizations, and research institutions. This data may include air quality monitoring data, weather data, emissions data, and satellite imagery.

Air Quality Monitoring Stations:

Utilize data from air quality monitoring stations strategically placed throughout Tamil Nadu. These stations measure key air pollutants, such as particulate matter (PM2.5 and PM10), ground-level ozone (O_3), sulfur dioxide (SO_2), nitrogen dioxide (NO_2), and carbon monoxide (CO).

Satellite Imagery:

Incorporate satellite imagery to monitor air quality on a broader scale. Satellites can provide data on aerosol levels, which can help assess pollution trends and sources across the region.

Weather Data:

Integrate weather data, including temperature, humidity, wind speed, and wind direction, as meteorological conditions can significantly influence air pollution levels.

Emissions Data:

Access data on industrial emissions, transportation emissions, and other pollution sources. This data may be collected from government records, industry reports, and environmental assessments.

Data Analysis:

Perform statistical and spatial analysis to identify trends and patterns in air pollution levels. Analyze the data over time, considering daily, monthly, and seasonal variations.

Identification of Pollution Sources:

Identify major pollution sources and hotspots using emission data and dispersion modeling. This helps pinpoint areas where pollution levels are consistently high.

Correlation with Health Data:

Correlate air pollution data with health statistics, such as respiratory illness rates, to understand the impact of air quality on public health in Tamil Nadu.

Trend Analysis:

Examine long-term data to identify trends in air quality. Are pollution levels increasing or decreasing over time? Are there any significant variations in different regions of Tamil Nadu?

Seasonal Variations:

Analyze seasonal variations in air pollution. Determine if pollution levels are higher during certain months or weather conditions and establish possible causes.

Policy and Regulatory Assessment:

Evaluate the effectiveness of existing air quality regulations and policies in Tamil Nadu. Are pollution levels complying with regulatory standards, and are the standards stringent enough to protect public health?

Public Awareness and Mitigation:

Share the findings with the public, government agencies, and stakeholders to raise awareness about air pollution issues. Propose strategies for pollution mitigation, such as reducing emissions from industrial sources and promoting cleaner transportation.

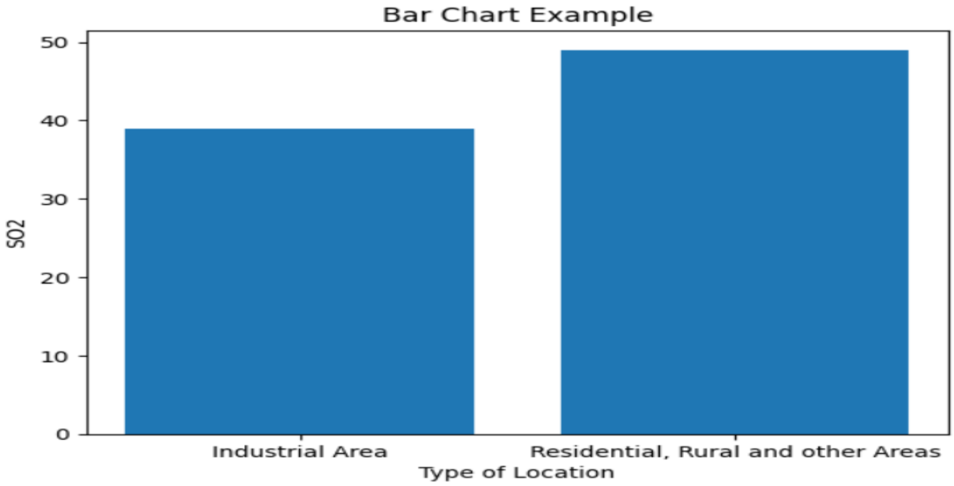
Recommendations:

Provide recommendations for improving air quality, such as implementing stricter emission standards, expanding green spaces, and encouraging the use of public transportation or electric vehicles.

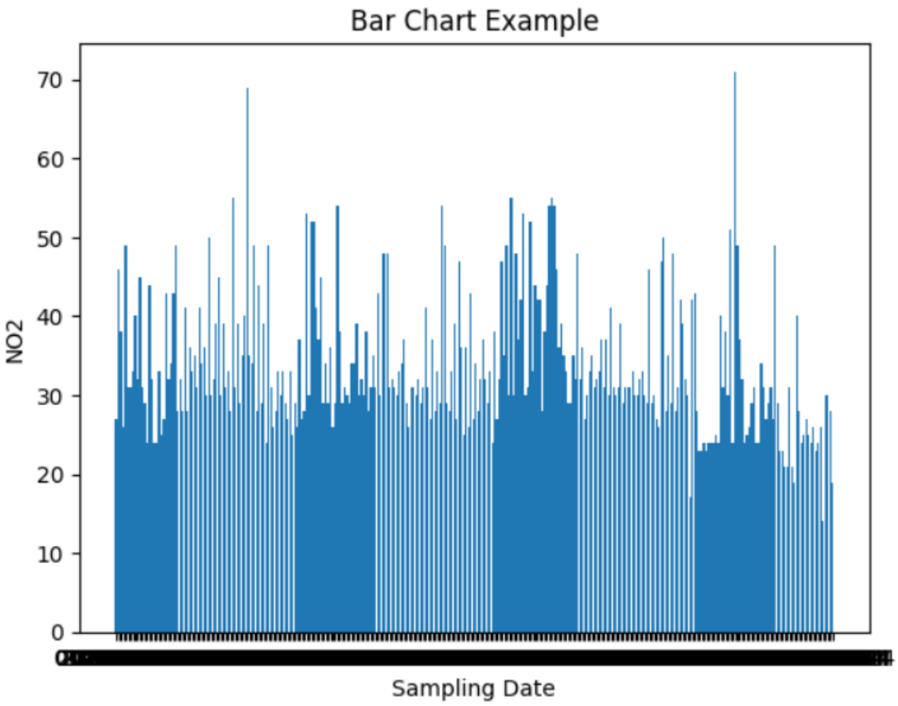
In summary, analyzing air pollution trends and pollution levels in Tamil Nadu involves a multidisciplinary approach that combines data from various sources, performs in-depth analysis, and provides insights into the state of air quality and its implications for public health and the environment. This information can inform policymaking and public awareness efforts to combat air pollution in the region.

Input & Output:

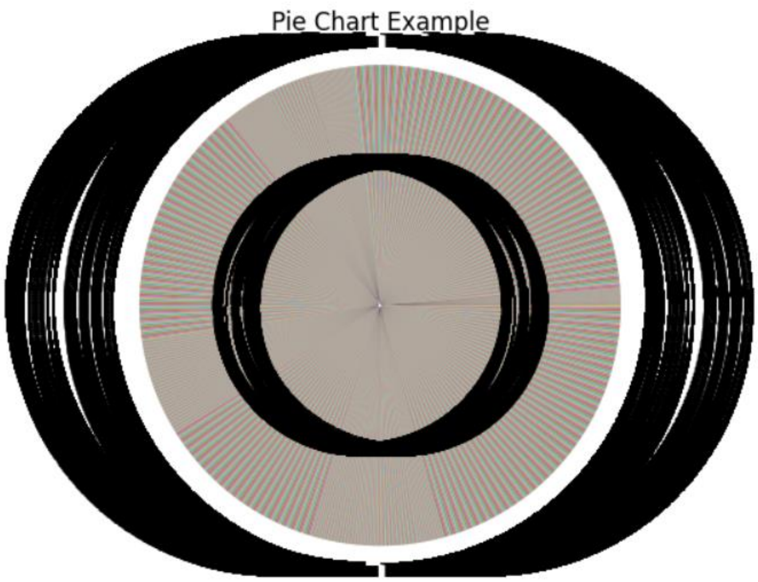
```
df = pd.read_csv('/content/cpcb_dly_aq_tamil_nadu-2014.csv')
plt.bar(df['Type of Location'], df['SO2'])
plt.xlabel('Type of Location')
plt.ylabel('SO2')
plt.title('Bar Chart Example')
plt.show()
```



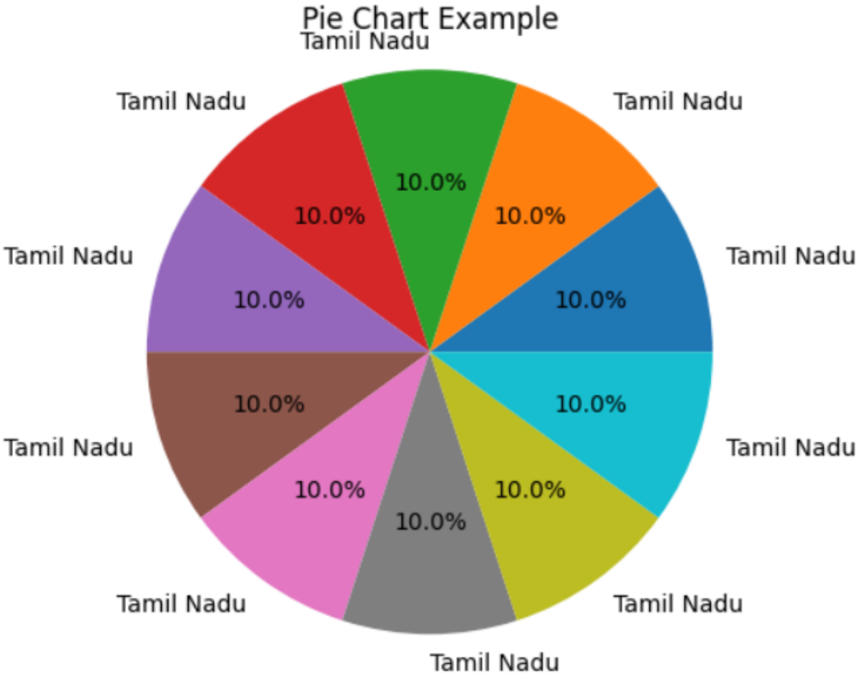
```
plt.bar(df['Sampling Date'], df['NO2'])
plt.xlabel('Sampling Date')
plt.ylabel('NO2')
plt.title('Bar Chart Example')
plt.show()
```




```
labels = df['State']
sizes = df['Stn Code']
plt.pie(sizes, labels=labels, autopct='%1.1f%%')
plt.axis('equal')
plt.title('Pie Chart Example')
plt.show()
```



```
labels = df['State'][:10]
sizes = df['Stn Code'][:10]
plt.pie(sizes, labels=labels, autopct='%1.1f%%')
plt.axis('equal')
plt.title('Pie Chart Example')
plt.show()
```




```
labels = df['State'][:20]
sizes = df['Stn Code'][:20]
lt.pie(sizes, labels=labels, autopct='%1.1f%%')
plt.axis('equal')
plt.title('Pie Chart Example')
plt.show()
```