IBM NAAN MUDHALVAN - PHASE 3

DOMAIN – DATA ANALYTICS

Air Quality Analysis in Tamil Nadu

Introduction:

In Tamil Nadu, the "Air Quality Analysis" project seeks to analyze and visualize data from air quality monitoring stations. The primary goal is to uncover air pollution trends, pinpoint pollution hotspots, and construct a predictive model for RSPM/PM10 estimation based on SO2 and NO2 levels. By leveraging Python and pertinent libraries, this project will contribute to informed decision-making and environmental well-being in the region.

Problem statement:

The air quality in Tamil Nadu is a growing concern, with deteriorating levels of air pollution posing a significant threat to the health and well-being of its residents. This problem statement aims to address the pressing issues related to air quality in Tamil Nadu by analyzing the factors contributing to air pollution, assessing its impact on public health and the environment, and proposing effective strategies and policies to mitigate and improve air quality.

PreProcessing Steps:

- **Data Collection and Integration:** Gather air quality data from monitoring stations in Tamil Nadu, ensuring that data from various sources and stations are integrated into a unified dataset.
- **Data Cleaning:** Identify and handle missing values, outliers, and inconsistencies in the dataset to ensure data integrity and accuracy.
- **Feature Selection:** Determine which features (e.g., SO2, NO2, RSPM/PM10) are relevant to the project objectives and remove unnecessary variables.
- **Data Transformation:** Normalize or scale the data as needed to bring it to a consistent and comparable format. This may involve transforming units, aggregating data over time intervals, or spatial scales.
- **Data Validation**: Check the data for consistency and accuracy, ensuring that it aligns with the project's objectives and that it is ready for subsequent analysis and modeling.

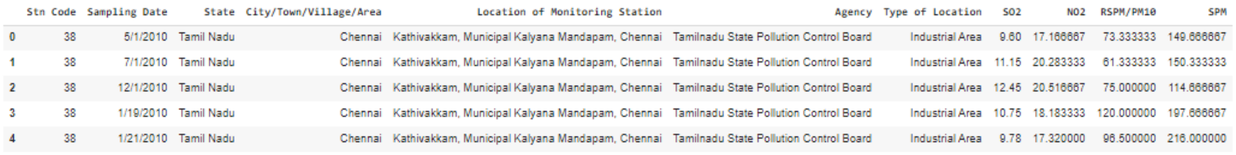
Analysis and Visualization:

The project's analysis and visualization phase will involve processing extensive air quality data to identify pollution trends and high-risk areas. Data preprocessing, statistical analysis, and machine learning techniques will be employed to uncover patterns. Interactive visualizations, such as maps and graphs, will be created to make the insights accessible. Python and data visualization libraries will be used to present findings effectively, aiding in data-driven decision-making for air quality improvement.

```
# Importing the necessary python libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
# Importing the csv file
df = pd.read_csv('Air_Quality.csv')
df.head()
```

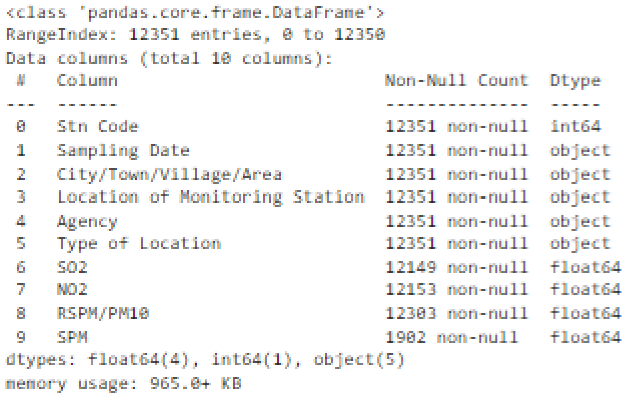
Output:



	Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM/PM10	SPM
0	38	5/1/2010	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	9.60	17.166667	73.333333	149.666667
1	38	7/1/2010	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	11.15	20.283333	61.333333	150.333333
2	38	12/1/2010	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	12.45	20.516667	75.000000	114.666667
3	38	1/19/2010	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	10.75	18.183333	120.000000	197.666667
4	38	1/21/2010	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	9.78	17.320000	96.500000	216.000000

```
df.info()
```

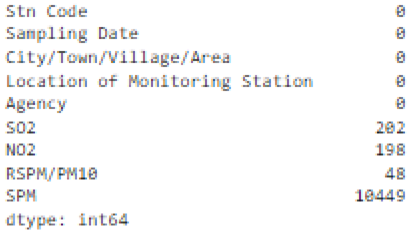
Output:



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12351 entries, 0 to 12350
Data columns (total 10 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   Stn Code                        12351 non-null  int64  
 1   Sampling Date                   12351 non-null  object 
 2   City/Town/Village/Area          12351 non-null  object 
 3   Location of Monitoring Station  12351 non-null  object 
 4   Agency                          12351 non-null  object 
 5   Type of Location                12351 non-null  object 
 6   SO2                             12149 non-null  float64
 7   NO2                             12153 non-null  float64
 8   RSPM/PM10                       12303 non-null  float64
 9   SPM                             1902 non-null   float64
dtypes: float64(4), int64(1), object(5)
memory usage: 965.0+ KB
```

```
# Check for null values
df.isna().sum()
#pd.isnull(df).sum()
```

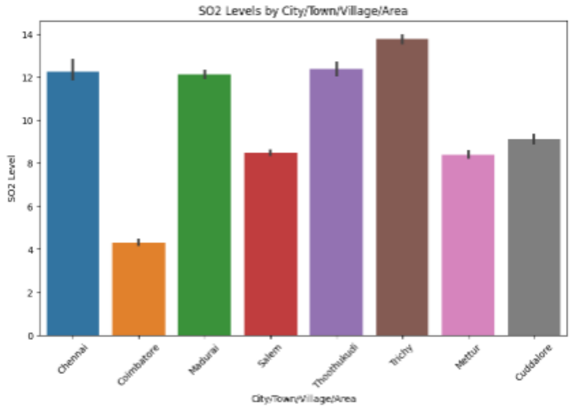
Output:



```
Stn Code                              0
Sampling Date                         0
City/Town/Village/Area                0
Location of Monitoring Station        0
Agency                                0
SO2                                 202
NO2                                 198
RSPM/PM10                            48
SPM                               10449
dtype: int64
```

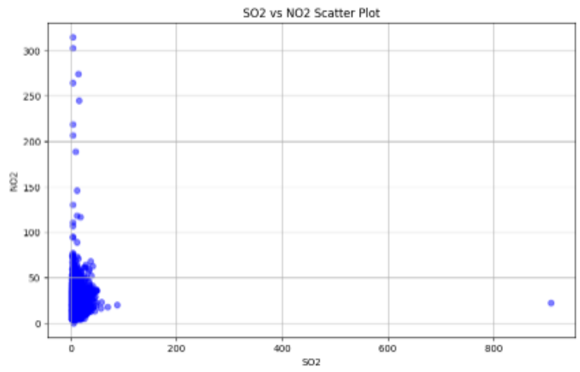
```python
# Create the line chart
dat = pd.DataFrame(df)
plt.figure(figsize=(10, 6))
sns.barplot(x='City/Town/Village/Area', y='SO2', data=dat)
plt.title('SO2 Levels by City/Town/Village/Area')
plt.xticks(rotation=45)
plt.xlabel('City/Town/Village/Area')
plt.ylabel('SO2 Level')
plt.show()
```

Output:



```python
# Scatter plot for 'SO2' vs 'NO2'
plt.figure(figsize=(10, 6))
plt.scatter(df['SO2'], df['NO2'], c='blue', alpha=0.5)
plt.title('SO2 vs NO2 Scatter Plot')
plt.xlabel('SO2')
plt.ylabel('NO2')
plt.grid(True)  # Add gridlines (optional)
plt.show()
```

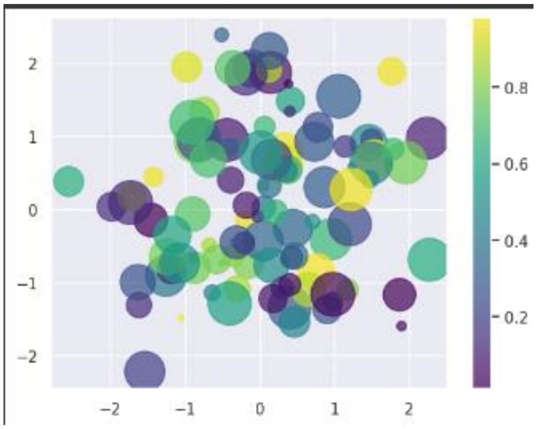
Output:



```
#Creating a Scatter Plot with Variable Colors and Sizes.
import numpy as np
import matplotlib.pyplot as plt
rng=np.random.RandomState(0)
x=rng.randn(100)
y=rng.randn(100)
colors=rng.rand(100)
sizes=1000*rng.rand(100)

plt.scatter(x,y,c=colors,s=sizes,alpha=0.7,cmap='viridis')
plt.colorbar()
plt.show()
```

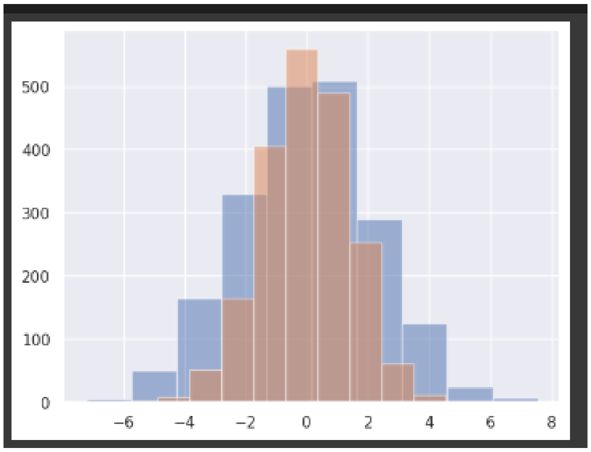
Output:



```
#Visualization of Univariate Distributions with Histograms
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()

data= np.random.multivariate_normal([0,0], [[5, 2], [2, 2]], size=2000)
data= pd.DataFrame(data, columns=['x','y'])
plt.hist(data["x"], alpha=0.5)
plt.hist(data["y"], alpha=0.5)
plt.show()
```

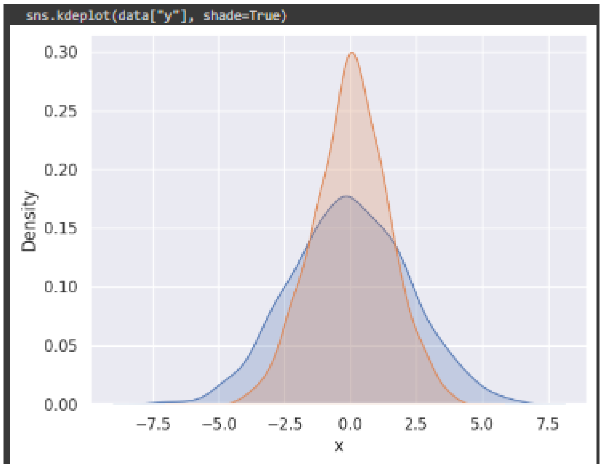
Output:



```
#Visualization of Bivariate Distributions with Kernel Density Estimation (KDE) Plots
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()

data= np.random.multivariate_normal([0,0], [[5, 2], [2, 2]], size=2000)
data= pd.DataFrame(data, columns=['x','y'])
sns.kdeplot(data["x"], shade=True)
sns.kdeplot(data["y"], shade=True)
plt.show()
```

Output:

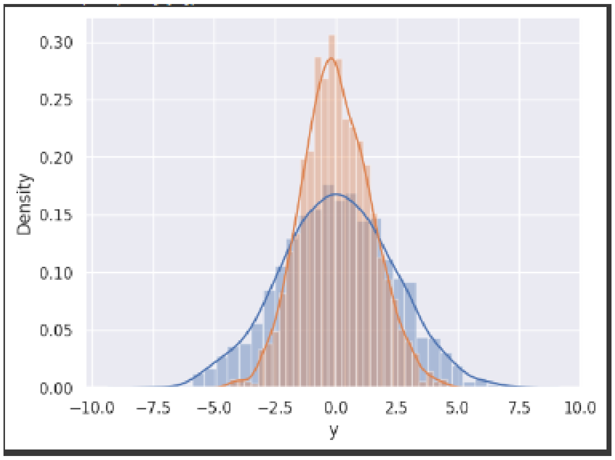


```
sns.kdeplot(data["y"], shade=True)
```

```
#Visualization of Bivariate Distributions with Probability Density Plots
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()

data= np.random.multivariate_normal([0,0], [[5, 2], [2, 2]], size=2000)
data= pd.DataFrame(data, columns=['x','y'])
sns.distplot(data["x"])
sns.distplot(data["y"])
plt.show()
```

Output:



Conclusion:

The preprocessing phase of the "Air Quality Analysis in Tamil Nadu" project is a crucial preparatory stage. By collecting, cleaning, and transforming the air quality dataset, we ensure that the data is of high quality and well-suited for analysis and modeling. This process is fundamental to our objective of gaining insights into air pollution trends, identifying pollution hotspots, and constructing a predictive model for RSPM/PM10 estimation based on SO2 and NO2 levels. With clean and consistent data, we are better equipped to make informed decisions and take steps toward improving air quality in Tamil Nadu.