**PHASE 5**

|  |  |
| --- | --- |
| **Date** | **31-10-2023** |
| **Team ID** | **499** |
| **Project Name** | **6112-AIR QUALITY ASSESSMENT IN TN** |

|  |  |
| --- | --- |
| 1 | Problem Statement |
| 2 | Project Overview |
| 3 | Design Thinking Approach |
| 4 | Development Phases |
| 5 | Data Visualization using Python |
| 6 | Data Visualization using IBM Cognos |
| 7 | Environmental Improvement |
| 8 | Conclusion |

**AIR QUALITY ASSESSMENT IN TN:**

**PROBLEM STATEMENT:**

**Objective:** Develop a comprehensive and localized air quality assessment system in Tennessee, leveraging advanced monitoring technologies and data analysis techniques.

**Data:** To conduct a comprehensive air quality assessment in Tennessee, critical data includes measurements of pollutants like PM2.5, PM10, NO2, SO2, CO, and O3 from monitoring stations. Meteorological data, encompassing temperature, humidity, wind patterns, and precipitation, is vital for understanding atmospheric conditions. Geographic information system (GIS) data aids in assessing how topography and land use influence air quality. Emission inventories offer insights into pollution sources, while population density and demographic data help identify vulnerable communities. Health records, historical data, and air quality modeling outputs provide additional context. Access to policy and regulatory information ensures evaluations align with existing guidelines, while remote sensing data offers a broader perspective on pollution patterns.

**Project Overview**:

Objective: The project aims to analyze air quality data to understand pollution patterns and provide insights for environmental improvement measures.

**LITERATURE SERVEY:**

**1. CURRENT AIR QUALITY ANALYTICS AND MONITORING: A REVIEW**

The article provides an in-depth overview of contemporary air quality analytics and monitoring methods. It highlights the growing concern over air pollution and the need for precise monitoring techniques. The document acknowledges the limitations of traditional methods and advocates for a comprehensive approach that combines cutting-edge technologies. These include passive sampling, remote sensing, and biomonitoring using lichens, mosses, and tree barks. The article emphasizes the significance of standardization, collaboration, and data sharing to enhance air quality assessment accuracy and coverage. Overall, it underscores the importance of innovative strategies to address the escalating challenges of air pollution.

**2. ANALYSIS OF AIR POLLUTION DATA IN INDIA BETWEEN 2015 AND 2019**

The document focuses on leveraging data analytics techniques to analyze air pollution data in India spanning 2015 to 2019. By addressing the problem of air pollution through data-driven insights, the analysis aims to offer a comprehensive understanding of pollution dynamics, contributing sources, and their impact on health and the environment. The proposed system underscores data preprocessing, exploratory analysis, and advanced analytics as crucial components to derive meaningful conclusions. Ultimately, the analysis seeks to empower decision-makers with evidence-based strategies to curb air pollution and pave the way for a healthier and cleaner future

**3. URBAN AIR QUALITY ANALYSIS AND PREDICTION USING MACHINE LEARNING**

The study addresses the pressing concern of air pollution by proposing a machine learning-based approach for predicting air pollutant concentrations. The research focuses on utilizing data from monitoring stations and employs K-means clustering to group similar data points. These clusters are then labeled based on pollutant concentrations. Two classification techniques, Multinominal Logistic Regression, and Decision Tree algorithms, are employed to analyze the data and predict air quality levels. Through comprehensive testing and analysis, the study aims to determine the most accurate method for predicting air pollution levels. The results are promising, with the Multinominal Logistic Regression model showing higher accuracy compared to the Decision Tree model. The research contributes to effective air quality management strategies by providing a predictive tool that aids in understanding and addressing air pollution levels for healthier environments.

**4. ANALYSIS OF AIR QUALITY INDICES USING FUZZY INFERENCE SYSTEM**

The document discusses an approach to assess and monitor air quality using a fuzzy inference system in the context of urban areas, with a focus on Chennai. Air pollution is a growing concern due to industrialization and urbanization, impacting human health and the environment. The proposed solution utilizes fuzzy logic to analyze input variables like SO2, NO2, TSPM, and RSPM, categorizing them into linguistic terms representing pollution levels. These variables are processed through predefined rules to generate an Air Quality Index (AQI), indicating air quality as excellent, good, moderate, poor, or very poor. The fuzzy inference system offers real-time monitoring and suggests outdoor activities based on AQI categories. The model proves effective in assessing air quality, particularly in commercial, residential, and industrial areas. This approach presents a valuable tool for urban planning, aiding in reducing air pollution's adverse effects and promoting better environmental health.

**5. MONITORING AMBIENT AIR QUALITY STUDY IN ARIYALUR, TAMILNADU, INDIA**

The Ariyalur concrete plant threatens the local ecosystem due to concrete kiln emissions, causing gray-white vegetation within 5 km. The region faces high PM10 accumulation risk from anthropogenic sources, mainly vehicular emissions and sports-related activities alongside limestone mining. Air pollution is a global concern, worsened by motorization, congested traffic, and blocked roads. Urgent measures are required, including advanced particle monitoring and improved sustainability practices within 50 days.

**DESIGN THINKING APPROACH :**

**Empathize:**

We understand that addressing air quality issues in Tennessee is of paramount importance for the well-being and health of its residents. The challenges faced, from limited monitoring infrastructure to the complexities of attributing pollution sources, undoubtedly present significant hurdles. The diverse geography of Tennessee further compounds the issue, demanding a tailored approach for each region. Balancing regulatory compliance with enforcement and public engagement is a delicate task. Additionally, uncertainties in predictive modeling and potential health impacts underscore the need for thorough assessments. It's clear that a collaborative, multidisciplinary effort is required to tackle these challenges and pave the way for cleaner, healthier air for all Tennesseans.

**ACTIONS:**

**Acknowledgment**: Recognize the concerns and challenges faced by the affected individuals and communities regarding air quality in Tennessee.

**Validation**: Validate their experiences and feelings, acknowledging that poor air quality can have serious impacts on health and quality of life.

**Active** **Listening**: Provide a safe space for individuals to express their concerns and experiences related to air quality issues, without judgment or interruption.

**Understanding**: Seek to understand the specific impacts that poor air quality has on their daily lives, such as health issues, lifestyle adjustments, or economic burden

**Define:**

Based on our understanding of the problem and the users' needs, we will define clear objectives and success criteria for our project.

**Ideate:**

Brainstorm potential solutions and approaches to address the problem. This phase involves thinking creatively and considering various statistics and techniques for Air quality assessment in Tamil Nadu.

**Actions:**

- Explore different Mathematical Statistics like F-Test, T-test, etc..,

- Experiment with feature engineering techniques to enhance model performance.

-Visualize the parameter like SO2 and NO2

**Prototype**

Create a Dashboard for the Air Quality Assessment in TamilNadu using IBM Cognos

**DEVELOPMENT PHASES** :

**Planning and Research**: Define project objectives, gather historical air quality data, and plan the data collection process.

**Data Preprocessing**: Air quality data from monitoring stations, satellites, and other sources. Clean, preprocess, and structure the data.

**Analysis and Visualization**:

Analyze the data to identify pollution trends, sources, and hotspots. Visualize the findings using tools like Python for custom visualizations and platforms like IBM Cognos for comprehensive reporting.

**Insights and Recommendations**: Interpret the findings and provide actionable recommendations for environmental improvement.

**Implementation**: Implement the recommended environmental measures, which could include regulatory changes, pollution control technologies, or public awareness campaigns.

**Testing and Validation**: Monitor air quality after implementation to ensure that the measures have a positive impact.

**Analysis Objectives**:

Pollution Source Identification: Identify major sources of air pollution (e.g., industrial emissions, traffic) and assess their contributions.

**Temporal and Spatial Analysis**: Analyze air quality variations over time and across different geographical areas.

**Python Code Integration**:

# Data Uploding

import pandas as pd  
x=pd.read\_csv("/content/cpcb\_dly\_aq\_tamil\_nadu-2014.csv")

x

Stn Code Sampling Date State City/Town/Village/Area \  
0 38 01-02-14 Tamil Nadu Chennai   
1 38 01-07-14 Tamil Nadu Chennai   
2 38 21-01-14 Tamil Nadu Chennai   
3 38 23-01-14 Tamil Nadu Chennai   
4 38 28-01-14 Tamil Nadu Chennai   
... ... ... ... ...   
2874 773 12-03-14 Tamil Nadu Trichy   
2875 773 12-10-14 Tamil Nadu Trichy   
2876 773 17-12-14 Tamil Nadu Trichy   
2877 773 24-12-14 Tamil Nadu Trichy   
2878 773 31-12-14 Tamil Nadu Trichy   
  
 Location of Monitoring Station \  
0 Kathivakkam, Municipal Kalyana Mandapam, Chennai   
1 Kathivakkam, Municipal Kalyana Mandapam, Chennai   
2 Kathivakkam, Municipal Kalyana Mandapam, Chennai   
3 Kathivakkam, Municipal Kalyana Mandapam, Chennai   
4 Kathivakkam, Municipal Kalyana Mandapam, Chennai   
... ...   
2874 Central Bus Stand, Trichy   
2875 Central Bus Stand, Trichy   
2876 Central Bus Stand, Trichy   
2877 Central Bus Stand, Trichy   
2878 Central Bus Stand, Trichy   
  
 Agency \  
0 Tamilnadu State Pollution Control Board   
1 Tamilnadu State Pollution Control Board   
2 Tamilnadu State Pollution Control Board   
3 Tamilnadu State Pollution Control Board   
4 Tamilnadu State Pollution Control Board   
... ...   
2874 Tamilnadu State Pollution Control Board   
2875 Tamilnadu State Pollution Control Board   
2876 Tamilnadu State Pollution Control Board   
2877 Tamilnadu State Pollution Control Board   
2878 Tamilnadu State Pollution Control Board   
  
 Type of Location SO2 NO2 RSPM/PM10 PM 2.5   
0 Industrial Area 11.0 17.0 55.0 NaN   
1 Industrial Area 13.0 17.0 45.0 NaN   
2 Industrial Area 12.0 18.0 50.0 NaN   
3 Industrial Area 15.0 16.0 46.0 NaN   
4 Industrial Area 13.0 14.0 42.0 NaN   
... ... ... ... ... ...   
2874 Residential, Rural and other Areas 15.0 18.0 102.0 NaN   
2875 Residential, Rural and other Areas 12.0 14.0 91.0 NaN   
2876 Residential, Rural and other Areas 19.0 22.0 100.0 NaN   
2877 Residential, Rural and other Areas 15.0 17.0 95.0 NaN   
2878 Residential, Rural and other Areas 14.0 16.0 94.0 NaN   
  
[2879 rows x 11 columns]

# Data Pre-Processing

*Checking for null values*

x.isnull().sum()

Stn Code 0  
Sampling Date 0  
State 0  
City/Town/Village/Area 0  
Location of Monitoring Station 0  
Agency 0  
Type of Location 0  
SO2 11  
NO2 13  
RSPM/PM10 4  
PM 2.5 2879  
dtype: int64

x["SO2"]

0 11.0  
1 13.0  
2 12.0  
3 15.0  
4 13.0  
 ...   
2874 15.0  
2875 12.0  
2876 19.0  
2877 15.0  
2878 14.0  
Name: SO2, Length: 2879, dtype: float64

import numpy as np

*Replacing null value with mean*

x["SO2"]=x["SO2"].fillna(x["SO2"].mean())

x["SO2"]

0 11.0  
1 13.0  
2 12.0  
3 15.0  
4 13.0  
 ...   
2874 15.0  
2875 12.0  
2876 19.0  
2877 15.0  
2878 14.0  
Name: SO2, Length: 2879, dtype: float64

x["NO2"]

0 17.0  
1 17.0  
2 18.0  
3 16.0  
4 14.0  
 ...   
2874 18.0  
2875 14.0  
2876 22.0  
2877 17.0  
2878 16.0  
Name: NO2, Length: 2879, dtype: float64

x["NO2"]=x["NO2"].fillna(x["NO2"].mean())

x.isnull().sum()

Stn Code 0  
Sampling Date 0  
State 0  
City/Town/Village/Area 0  
Location of Monitoring Station 0  
Agency 0  
Type of Location 0  
SO2 0  
NO2 0  
RSPM/PM10 4  
PM 2.5 2879  
dtype: int64

x["RSPM/PM10"]

0 55.0  
1 45.0  
2 50.0  
3 46.0  
4 42.0  
 ...   
2874 102.0  
2875 91.0  
2876 100.0  
2877 95.0  
2878 94.0  
Name: RSPM/PM10, Length: 2879, dtype: float64

x["RSPM/PM10"]=x["RSPM/PM10"].fillna(x["RSPM/PM10"].mean())

x.drop("PM 2.5",axis=1,inplace=True)

*Standard scalling*

from sklearn.preprocessing import StandardScaler  
  
scaler = StandardScaler()  
x[['SO2', 'NO2', 'RSPM/PM10']] = scaler.fit\_transform(x[['SO2', 'NO2', 'RSPM/PM10']])

*Feature Engineering*

x['Sampling Date'] = pd.to\_datetime(x['Sampling Date'], format='%d-%m-%y')

x['Day'] = x['Sampling Date'].dt.day  
x['Month'] = x['Sampling Date'].dt.month  
x['Year'] = x['Sampling Date'].dt.year

# Data Visualization

import matplotlib.pyplot as plt

import numpy as np  
from google.colab import autoviz  
  
def violin\_plot(df, value\_colname, facet\_colname, figscale=1, mpl\_palette\_name='Dark2', \*\*kwargs):  
 from matplotlib import pyplot as plt  
 import seaborn as sns  
 figsize = (12 \* figscale, 1.2 \* figscale \* len(df[facet\_colname].unique()))  
 plt.figure(figsize=figsize)  
 sns.violinplot(df, x=value\_colname, y=facet\_colname, palette=mpl\_palette\_name, \*\*kwargs)  
 sns.despine(top=True, right=True, bottom=True, left=True)  
 return autoviz.MplChart.from\_current\_mpl\_state()  
  
chart = violin\_plot(x, \*['SO2', 'City/Town/Village/Area'], \*\*{'inner': 'box'})  
chart

<google.colab.\_quickchart\_lib.MplChart at 0x7d046c4f8be0>

import numpy as np  
from google.colab import autoviz  
  
def heatmap(df, x\_colname, y\_colname, figscale=1, mpl\_palette\_name='viridis'):  
 from matplotlib import pyplot as plt  
 import seaborn as sns  
 import pandas as pd  
 plt.subplots(figsize=(8 \* figscale, 8 \* figscale))  
 df\_2dhist = pd.DataFrame({  
 x\_label: grp[y\_colname].value\_counts()  
 for x\_label, grp in df.groupby(x\_colname)  
 })  
 sns.heatmap(df\_2dhist, cmap=mpl\_palette\_name)  
 plt.xlabel(x\_colname)  
 plt.ylabel(y\_colname)  
 return autoviz.MplChart.from\_current\_mpl\_state()  
  
chart = heatmap(x, \*['Agency', 'Type of Location'], \*\*{})  
chart

<google.colab.\_quickchart\_lib.MplChart at 0x7d046c4da1a0>

import numpy as np  
from google.colab import autoviz  
  
def value\_plot(df, y, figscale=1):  
 from matplotlib import pyplot as plt  
 df[y].plot(kind='line', figsize=(8 \* figscale, 4 \* figscale), title=y)  
 plt.gca().spines[['top', 'right']].set\_visible(False)  
 plt.tight\_layout()  
 return autoviz.MplChart.from\_current\_mpl\_state()  
  
chart = value\_plot(x, \*['SO2'], \*\*{})  
chart

<google.colab.\_quickchart\_lib.MplChart at 0x7d046c432dd0>

import numpy as np  
from google.colab import autoviz  
  
def value\_plot(df, y, figscale=1):  
 from matplotlib import pyplot as plt  
 df[y].plot(kind='line', figsize=(8 \* figscale, 4 \* figscale), title=y)  
 plt.gca().spines[['top', 'right']].set\_visible(False)  
 plt.tight\_layout()  
 return autoviz.MplChart.from\_current\_mpl\_state()  
  
chart = value\_plot(x, \*['NO2'], \*\*{})  
chart

<google.colab.\_quickchart\_lib.MplChart at 0x7d046f31a1a0>

import numpy as np  
from google.colab import autoviz  
  
def scatter\_plot(df, x\_colname, y\_colname, figscale=1, alpha=.8):  
 from matplotlib import pyplot as plt  
 plt.figure(figsize=(6 \* figscale, 6 \* figscale))  
 df.plot(kind='scatter', x=x\_colname, y=y\_colname, s=(32 \* figscale), alpha=alpha)  
 plt.gca().spines[['top', 'right',]].set\_visible(False)  
 plt.tight\_layout()  
 return autoviz.MplChart.from\_current\_mpl\_state()  
  
chart = scatter\_plot(x, \*['RSPM/PM10', 'Day'], \*\*{})  
chart

<google.colab.\_quickchart\_lib.MplChart at 0x7d046c4a3f40>

<Figure size 600x600 with 0 Axes>

import numpy as np  
from google.colab import autoviz  
  
def categorical\_histogram(df, colname, figscale=1, mpl\_palette\_name='Dark2'):  
 from matplotlib import pyplot as plt  
 import seaborn as sns  
 df.groupby(colname).size().plot(kind='barh', color=sns.palettes.mpl\_palette(mpl\_palette\_name), figsize=(8\*figscale, 4.8\*figscale))  
 plt.gca().spines[['top', 'right',]].set\_visible(False)  
 return autoviz.MplChart.from\_current\_mpl\_state()  
  
chart = categorical\_histogram(x, \*['City/Town/Village/Area'], \*\*{})  
chart

<google.colab.\_quickchart\_lib.MplChart at 0x7d046c370d00>

import numpy as np  
from google.colab import autoviz  
  
def histogram(df, colname, num\_bins=20, figscale=1):  
 from matplotlib import pyplot as plt  
 df[colname].plot(kind='hist', bins=num\_bins, title=colname, figsize=(8\*figscale, 4\*figscale))  
 plt.gca().spines[['top', 'right',]].set\_visible(False)  
 plt.tight\_layout()  
 return autoviz.MplChart.from\_current\_mpl\_state()  
  
chart = histogram(x, \*['RSPM/PM10'], \*\*{})  
chart

<google.colab.\_quickchart\_lib.MplChart at 0x7d046c5af130>

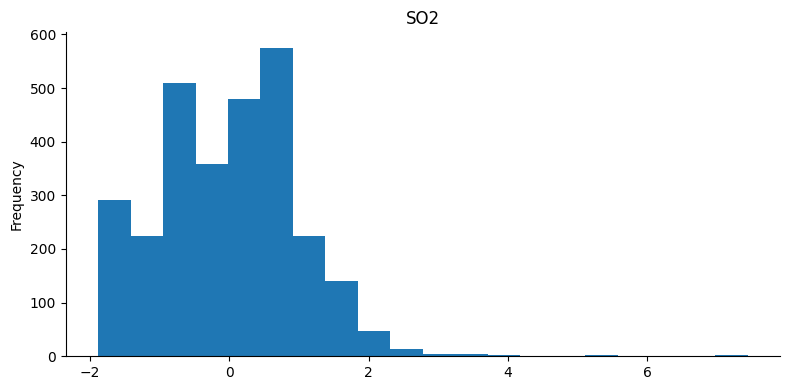
import numpy as np  
from google.colab import autoviz  
  
def histogram(df, colname, num\_bins=20, figscale=1):  
 from matplotlib import pyplot as plt  
 df[colname].plot(kind='hist', bins=num\_bins, title=colname, figsize=(8\*figscale, 4\*figscale))  
 plt.gca().spines[['top', 'right',]].set\_visible(False)  
 plt.tight\_layout()  
 return autoviz.MplChart.from\_current\_mpl\_state()  
  
chart = histogram(x, \*['NO2'], \*\*{})  
chart

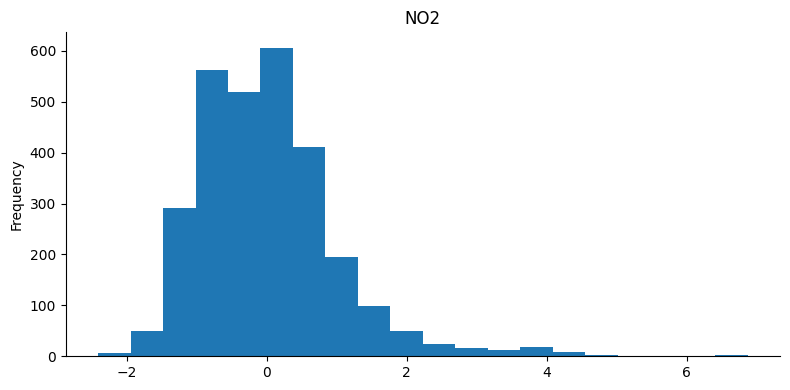
<google.colab.\_quickchart\_lib.MplChart at 0x7d046c433ca0>

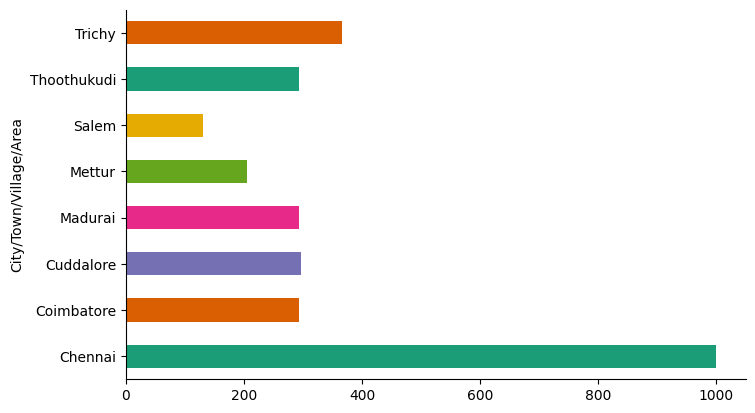
import numpy as np  
from google.colab import autoviz  
import matplotlib.pyplot as plt  
  
def histogram(df, colname, num\_bins=20, figscale=1):  
 from matplotlib import pyplot as plt  
 df[colname].plot(kind='hist', bins=num\_bins, title=colname, figsize=(8\*figscale, 4\*figscale))  
 plt.gca().spines[['top', 'right',]].set\_visible(False)  
 plt.tight\_layout()  
 return autoviz.MplChart.from\_current\_mpl\_state()  
  
chart = histogram(x, \*['SO2'], \*\*{})  
chart

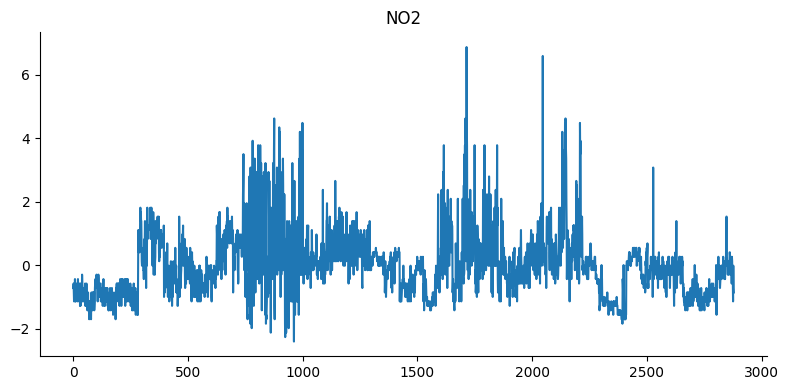
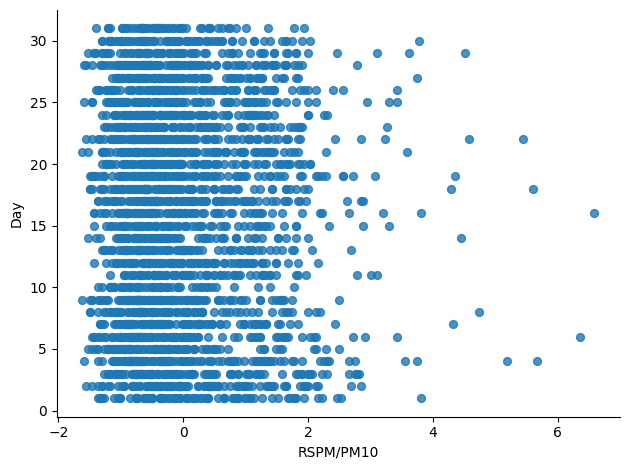
<google.colab.\_quickchart\_lib.MplChart at 0x7d046e6e0e80>

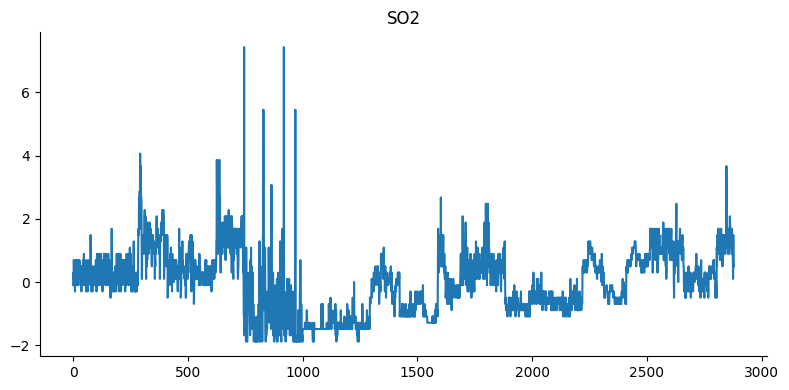
**DATA VISUALIZATION USING PYTHON :**

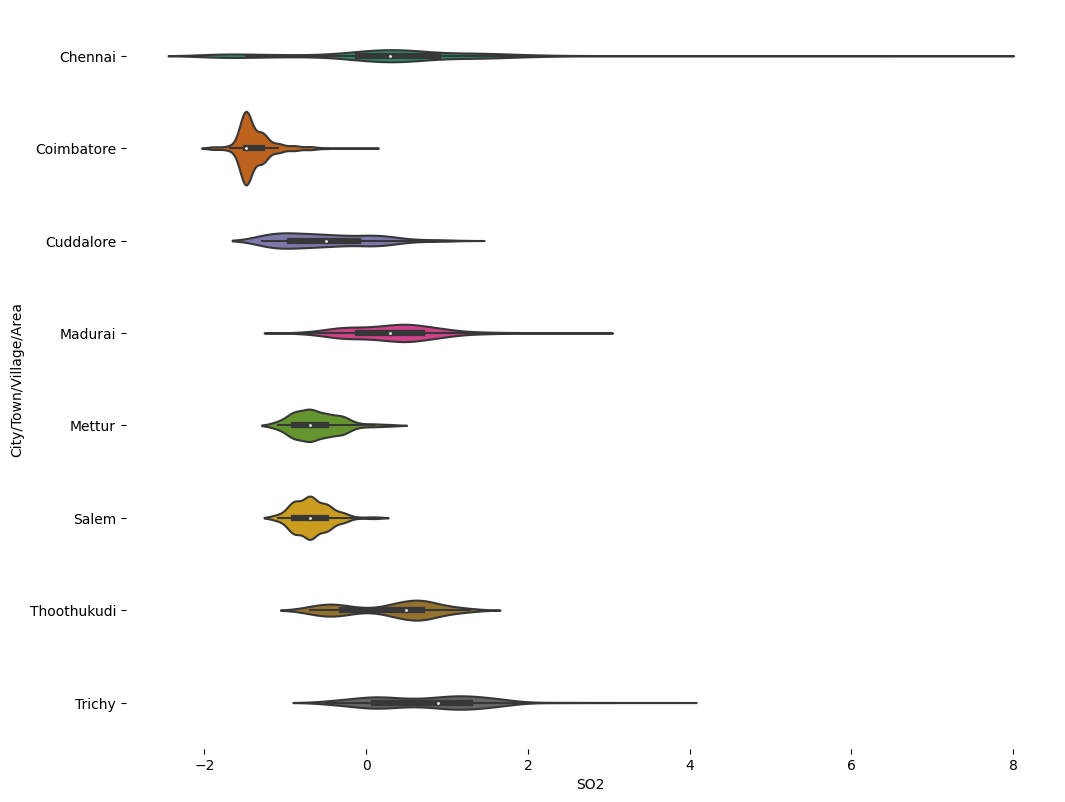


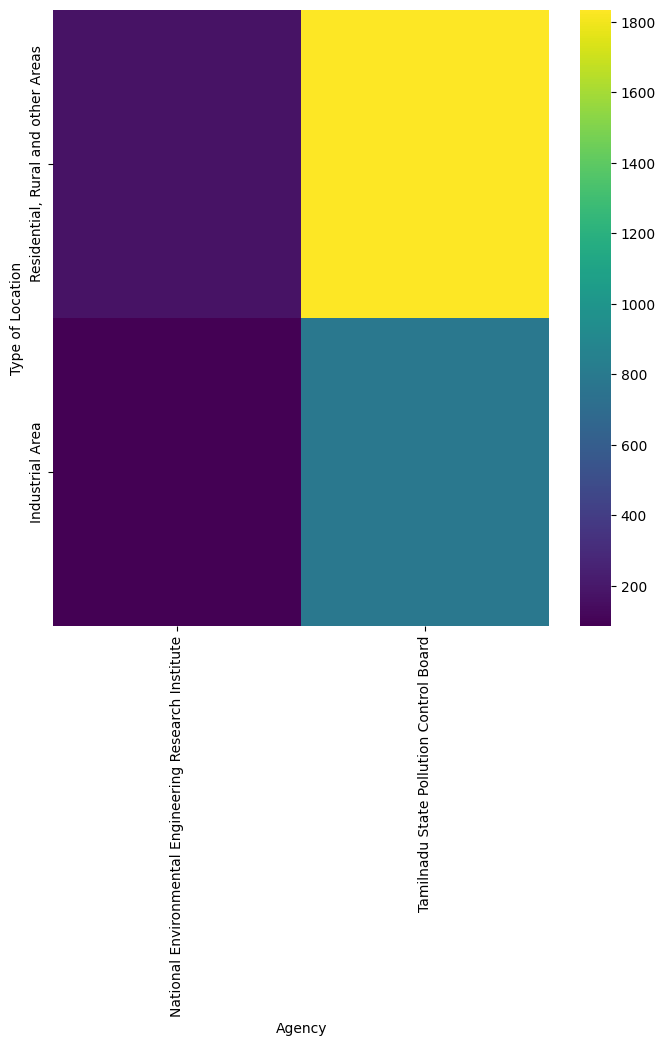






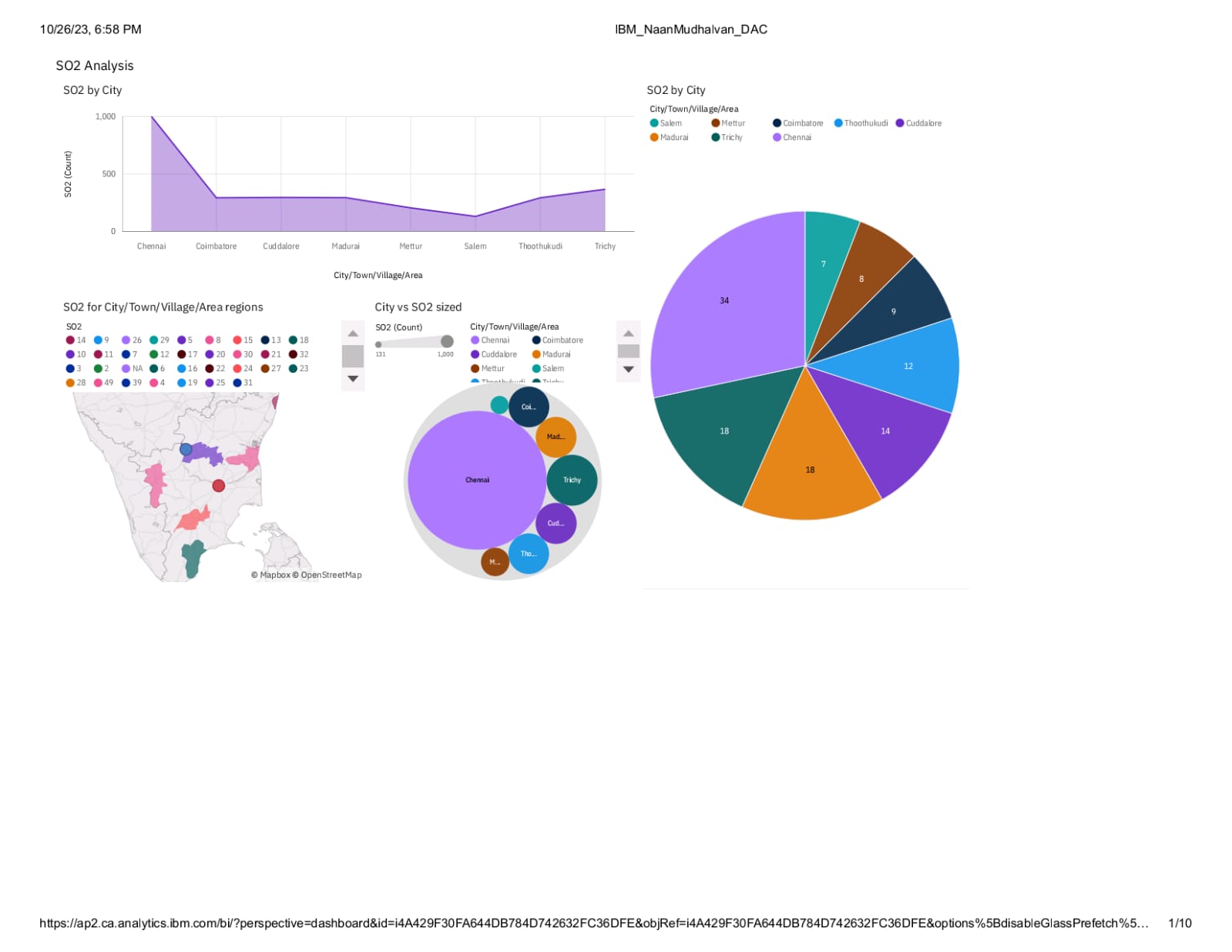






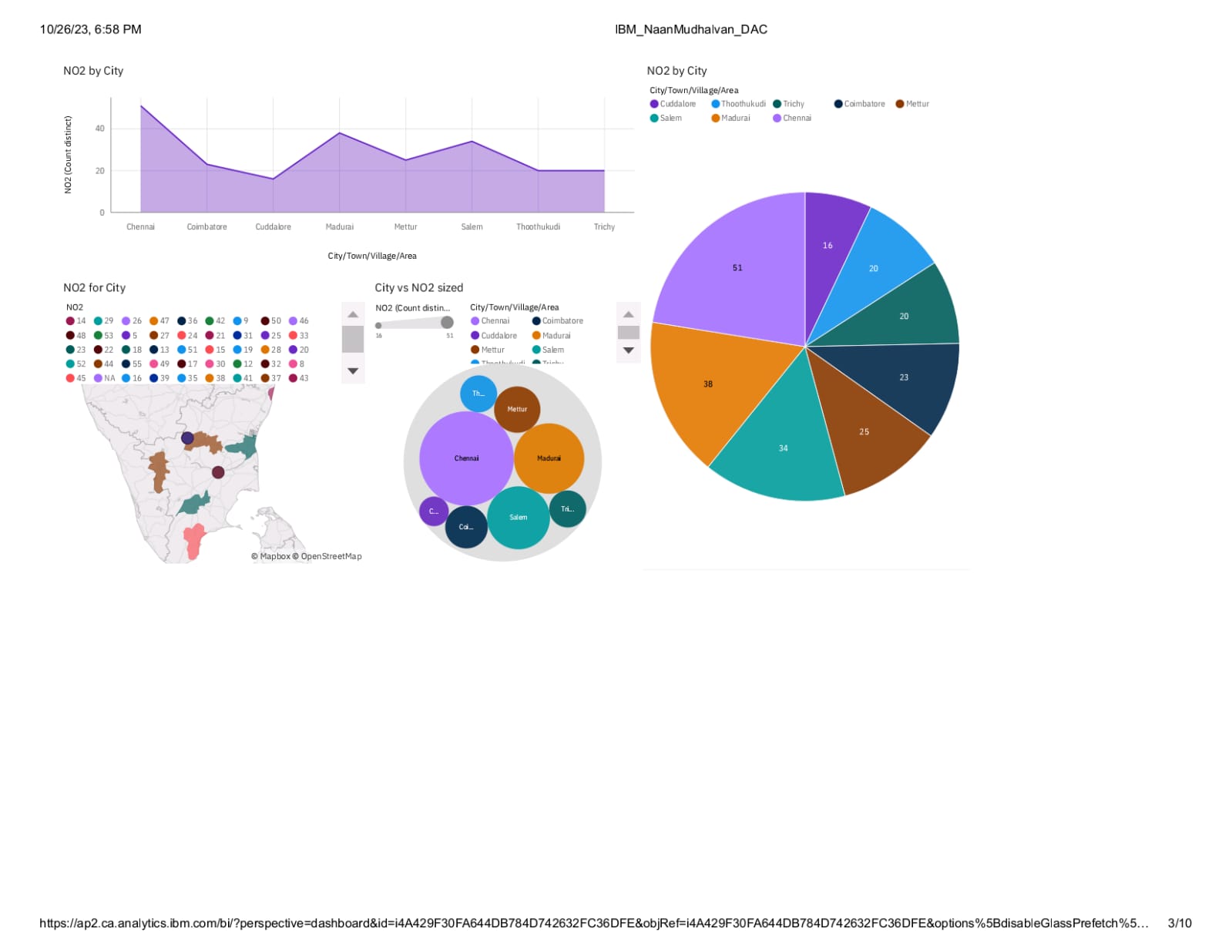
**DATA VISUALIZATION USING IBM COGNOS :**

* ­­­­**SO2 Analysis**



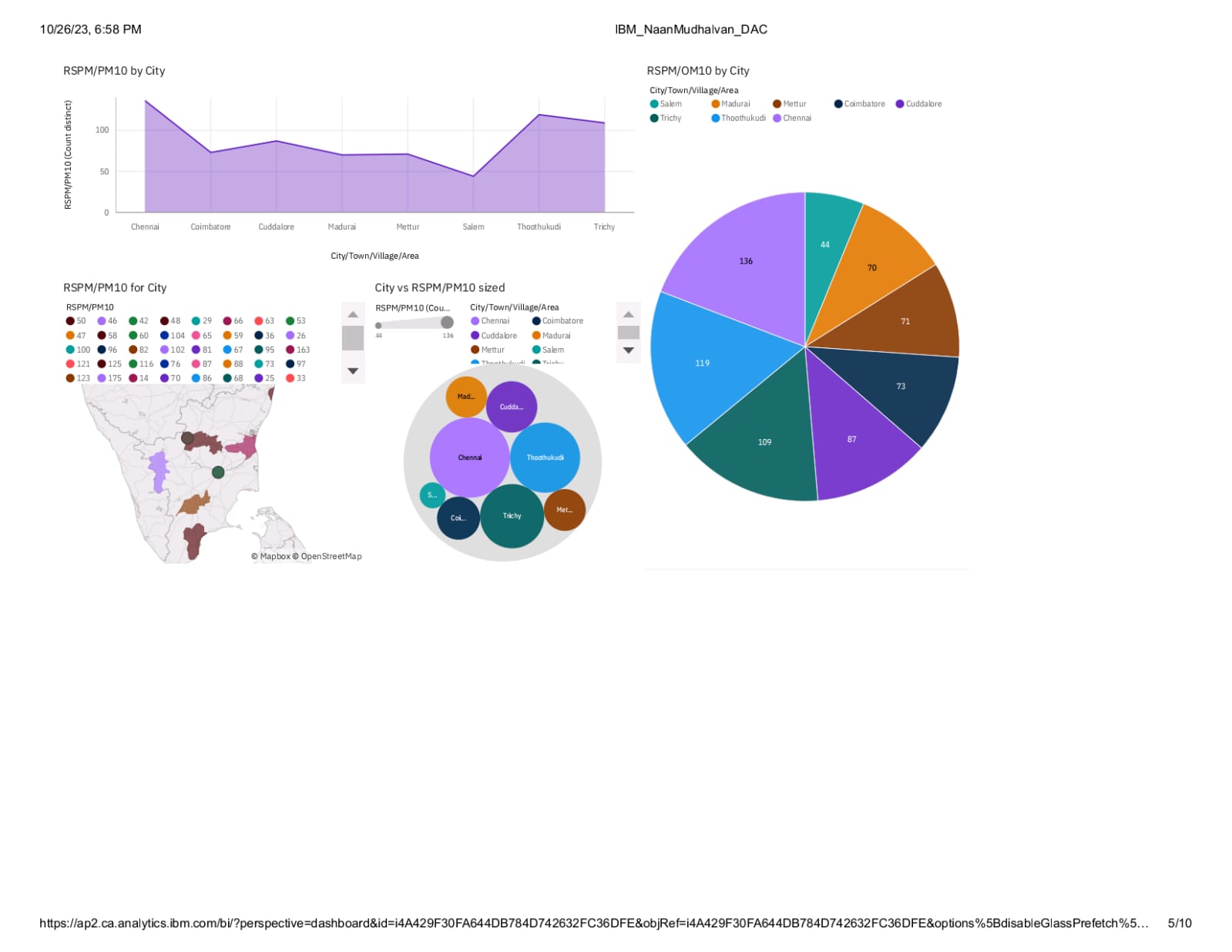
**REPORT :**

* The total number of results for SO2, across all City/Town/Village/Area, is nearly three thousand.
* The count is unusually high when the City/Town/Village/Area is Chennai.
* Salem has a SO2 of 131 for Stn Code 309
* Chennai is the most frequently occurring category of City/Town/Village/Area with a count of 1000 items with SO2 values (34.7 % of the total).
* The total number of results for SO2, across all City/Town/Village/Area, is nearly three thousand
* **NO2 Analysis**



**REPORT:**

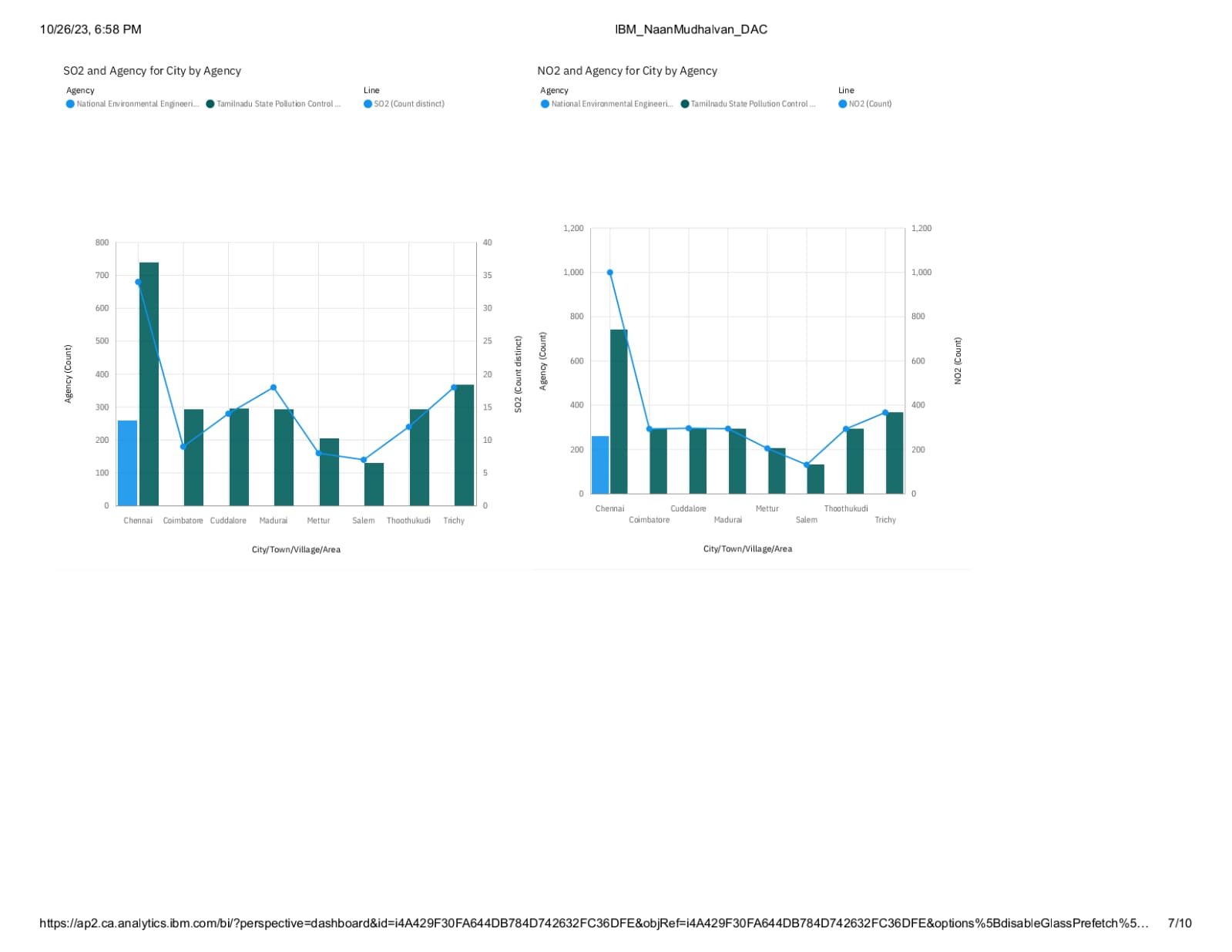
* City/Town/Village/Area Chennai has the highest NO2 Stn Code 161
* Chennai is the most frequently occurring category of City/Town/Village/Area with a count of 1000 items with NO2 values (34.7 % of the total).
* The total number of results for NO2, across all City/Town/Village/Area, is nearly three thousand
* The total number of results for NO2, across all City/Town/Village/Area, is nearly three thousand.
* **RSPM/PM10 Analysis**



**REPORT:**

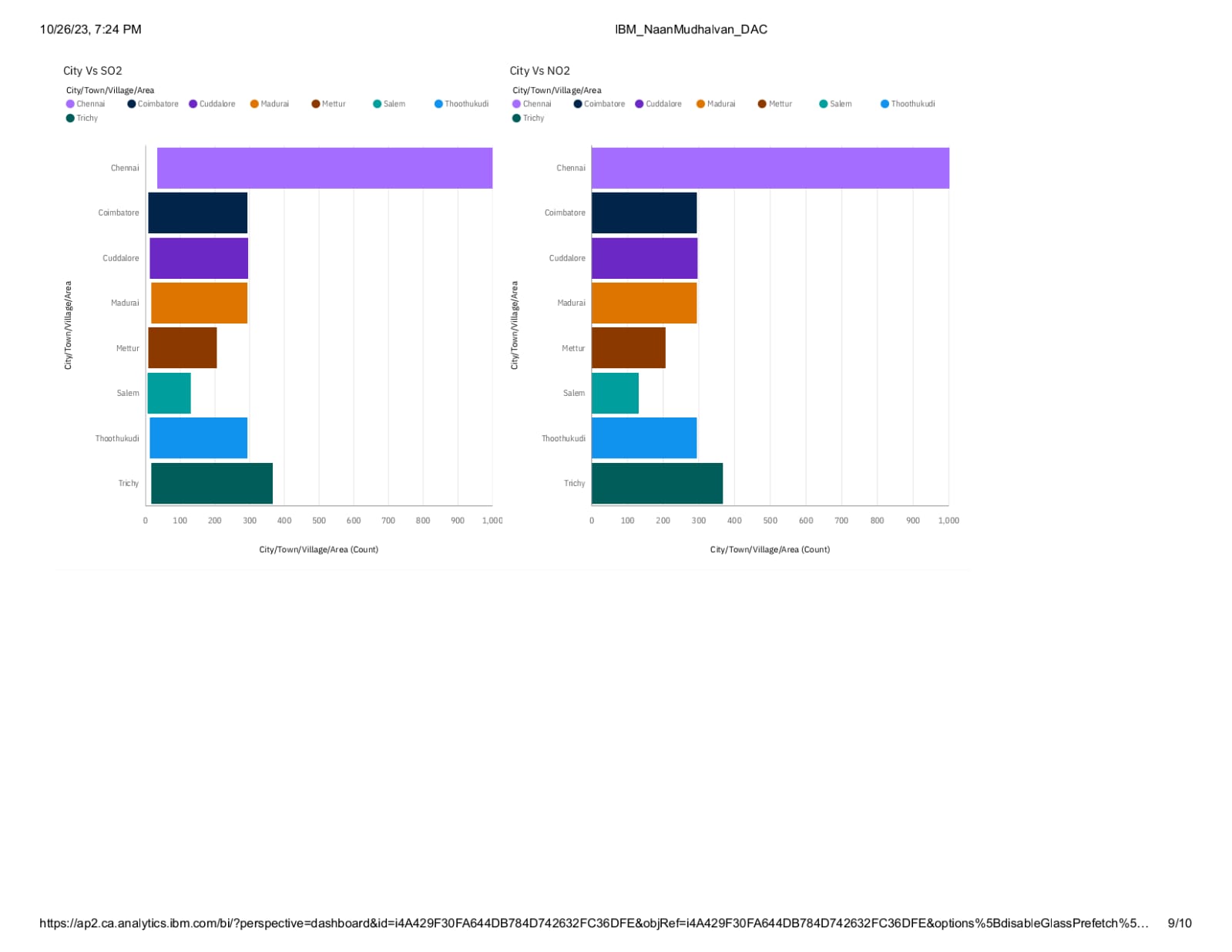
* City/Town/Village/Area Chennai has the highest RSPM/PM10 due to Stn Code 766.
* Thoothukudi has an RSPM/PM10 of 71 for Stn Code 239.
* RSPM/PM10, across all City/Town/Village/Area, is nearly three thousand.
* City/Town/Village/Area Chennai has the highest RSPM/PM10 due to Stn Code 766
* City/Town/Village/Area with a count of 1000 items with RSPM/PM10 values (34.7 % of the total).

**SO2 and NO2 Analysis**



**REPORT:**

* Tamilnadu State Pollution Control Board is the most frequently occurring category of Agency with a count of 2619 items with Agency values (91 % of the total).
* City/Town/Village/Area Chennai has the highest NO2 due to Stn Code 764.
* Salem has a NO2 of 131 for Stn Code 309.
* Tamil Nadu State Pollution Control Board is the most frequently occurring category of Agency with a count of 2619 items with Agency values (91 % of the total).
* **City vs SO2 vs NO2**



**ENVIRONMENTAL IMPROVEMENT** :

**Policy Recommendations**: Provide evidence-based recommendations for policy changes to reduce pollution levels.

**Technological Interventions**: Suggest pollution control technologies and practices for industries and transportation.

**Public Awareness Campaigns**: Develop campaigns to educate the public about the importance of reducing individual contributions to air pollution.

**Continuous Monitoring**: Implement a monitoring system to track the effectiveness of the proposed measures and adjust strategies as needed.

**CONCLUSION**:

In conclusion, the Air Quality Analysis project offers critical insights into pollution patterns and sources, providing a foundation for informed environmental interventions. Through rigorous data collection and analysis, we identified temporal and spatial variations in air quality, highlighting areas in need of targeted improvements. The integration of IBM Cognos and Python facilitated clear, interactive data visualization, enabling stakeholders to comprehend complex trends and make informed decisions.

Our recommendations encompass a multi-pronged approach, including policy adjustments, technological advancements, and public awareness campaigns. These measures aim to reduce pollution levels and enhance overall air quality, benefiting public health and the environment. Implementation will require concerted efforts from government agencies, industries, and the community.

Ongoing monitoring and evaluation are essential to gauge the effectiveness of interventions and adapt strategies as needed. By remaining responsive to changing environmental conditions, we can work towards a sustainable future with improved air quality, ensuring the well-being of communities for generations to come.