DATA ANALTYICS WITH COGNOS

PROJECT: AIR QUALITY ANLYSIS TAMIL NADU

PHASE 2 PROJECT

TEAM MEMBERS:

1. S. BALAKANDAN – 820421205012 – balakandan50@gmail.com
2. R. BHARATH RAJ – 820421205015 – bharathrajravi2004@gmail.com
3. V. ARUN KUMAR – 820421205008 – arurarun0088@gmail.com
4. M. MUKILAN – 820421205042 – mukilmuruga07@gmail.com
5. T. LOGESH – 820421205039 – logeshselvan6@gmail.com

Table of Contents:

1. Introduction:

Recently, much has been discussed about air pollution and its consequences on the environment. These discussion always gain prominence when some of their consequences haunt the world and leave us wondering what will be of future generation. Air quality is a critical concern for public health and environmental well-being. Tamil Nadu have PM 2.5 of 28.2 microgram/m3 levels. This Project emphasis by analysing and pre-processing the air quality dataset which is essential for informed decision-making and effective pollution control measures.

2. Problem Statement:

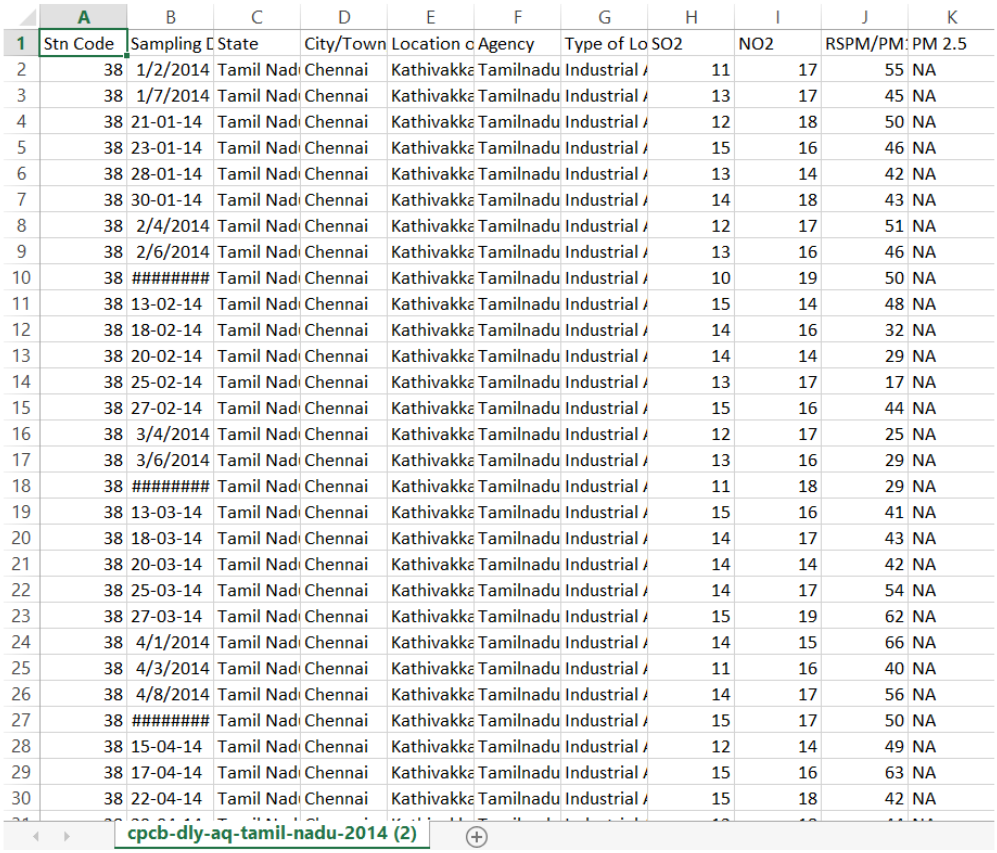
The objective is to build the project by loading and pre-processing the dataset. Begin the analysis by Loading and pre-processing the Air Quality Dataset. Load the dataset using Python and Data Manipulation Libraries.

3. Data Collection:

The dataset containing location-wise daily ambient air quality records for Tamil Nadu in the year 2014 has been obtained from the below datalink.

Dataset Link: https://tn.data.gov.in/resource/location-wise-daily-ambient-air-quality-tamil-nadu-year-2014

Dataset:



	A	B	C	D	E	F	G	H	I	J	K
1	Stn Code	Sampling D	State	City/Town	Location o	Agency	Type of Lo	SO2	NO2	RSPM/PM:	PM 2.5
2	38	1/2/2014	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	11	17	55	NA
3	38	1/7/2014	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	13	17	45	NA
4	38	21-01-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	12	18	50	NA
5	38	23-01-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	15	16	46	NA
6	38	28-01-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	13	14	42	NA
7	38	30-01-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	14	18	43	NA
8	38	2/4/2014	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	12	17	51	NA
9	38	2/6/2014	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	13	16	46	NA
10	38	########	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	10	19	50	NA
11	38	13-02-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	15	14	48	NA
12	38	18-02-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	14	16	32	NA
13	38	20-02-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	14	14	29	NA
14	38	25-02-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	13	17	17	NA
15	38	27-02-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	15	16	44	NA
16	38	3/4/2014	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	12	17	25	NA
17	38	3/6/2014	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	13	16	29	NA
18	38	########	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	11	18	29	NA
19	38	13-03-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	15	16	41	NA
20	38	18-03-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	14	17	43	NA
21	38	20-03-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	14	14	42	NA
22	38	25-03-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	14	17	54	NA
23	38	27-03-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	15	19	62	NA
24	38	4/1/2014	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	14	15	66	NA
25	38	4/3/2014	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	11	16	40	NA
26	38	4/8/2014	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	14	17	56	NA
27	38	########	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	15	17	50	NA
28	38	15-04-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	12	14	49	NA
29	38	17-04-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	15	16	63	NA
30	38	22-04-14	Tamil Nadi	Chennai	Kathivakka	Tamilnadu	Industrial A	15	18	42	NA

cpcb-dly-aq-tamil-nadu-2014 (2)

4. Data Cleansing:

Data cleaning is one of the important parts of machine learning. It plays a significant part in building a model.

Data pre-processing is a crucial step in preparing the dataset for machine learning.

Data Cleansing involves:

- Identifying and removing any missing, duplicate or irrelevant data.
- Handling missing data.
- Removing outliers.
- Scaling the data.

Using Methods Such as:

- dropna()
- drop()
- drop_duplicates()
- scale()
- get_dummies()

5. Feature Engineering:

Feature engineering involves creating new features or modifying existing ones to improve model performance. In this context, it may involve generating lag features, aggregating data over time intervals, or incorporating weather data if available to capture external factors influencing air quality.

6. Data Manipulation Library Selection:

Select appropriate data manipulation libraries for the task. We will be using:

- Pandas
- Tensorflow
- SciPy
- NumPy
- Scikit-learn
- Seaborn
- Pytorch and etc..

7. Model Training and Evaluation:

Split the dataset into training and testing sets. Train the selected models on the training data and evaluate their performance using suitable metrics (e.g., Mean Absolute Error, R-squared and Accuracy). To Build Model such as Linear Regression, Decision Tree Regressor and SVR.

8. Program:

- First, import the required Python libraries. As given below.

```
# In[1]: import pandas as pd
         from sklearn.linear_model import LinearRegression
         from sklearn.linear_model import DecisionTreeRegressor
         from sklearn.linear_model import SVR
         import tensorflow as tf
         import tensorflow_datasets as tfds
         from sklearn.model_selection import train_test_split
         from sklearn.metrics import mean_squared_error,r2_score
         from sklearn.metrics import accuracy_score
         import matplotlib.pyplot as plt
         from sklearn import datasets
         import numpy as np
```

- To import the database into the jupyter notebook, you can use the following Python code given below.

```
# In[2]:data =pd.read_csv(r"C:\Users\Administrator\Downloads\airquality.csv")


# In[3]:print(data.head())
```

 # OP [3]:

```
   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

                    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

                                    Agency Type of Location   SO2   NO2  \
0  Tamilnadu State Pollution Control Board  Industrial Area  11.0  17.0
1  Tamilnadu State Pollution Control Board  Industrial Area  13.0  17.0
2  Tamilnadu State Pollution Control Board  Industrial Area  12.0  18.0
3  Tamilnadu State Pollution Control Board  Industrial Area  15.0  16.0
4  Tamilnadu State Pollution Control Board  Industrial Area  13.0  14.0

   RSPM/PM10  PM 2.5
0       55.0     NaN
1       45.0     NaN
2       50.0     NaN
3       46.0     NaN
4       42.0     NaN
```

- Data Cleansing and Transformation are done by following Python Code.
  - drop() - To drop an entire Column.
  - dropna() - To drop a the NaN Values.
  - drop_duplicates() - To drop the Duplicates in the Dataset.
  - scale() - In cases where all the columns have a significant difference in their scales, are needed to be modified in such a way that all those values fall into the same scale.
  - get_dummies() - indicate whether each row in the original dataset belongs to a particular category or not.

```
# In[3]: df=data.drop(['PM 2.5'],axis=1)
# In[4]: df.head()

# OP[4]:
```

	Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM/PM10
0	38	01-02-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	11.0	17.0	55.0
1	38	01-07-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	17.0	45.0
2	38	21-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	12.0	18.0	50.0
3	38	23-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	15.0	16.0	46.0
4	38	28-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	14.0	42.0

```
# In[5]: newf=df.dropna()
          newd=newf.drop_duplicates()

# In[6]: newf.head()
         newf.scale()
         newf.get_dummies()

# OP[6]:
```

Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM/PM10	
0	38	01-02-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	11.0	17.0	55.0
1	38	01-07-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	17.0	45.0
2	38	21-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	12.0	18.0	50.0
3	38	23-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	15.0	16.0	46.0
4	38	28-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	14.0	42.0

- Split the data for Training and Testing and train the model for Linear Regression.

```
# In[7]: x = newf[['NO2','SO2']]


# In[8]: y = newf['RSPM/PM10']
```

```
# In[9]:x_train,x_test,y_train,y_test =
         train_test_split(x,y,test_size=0.3,random_state=0)


# In[8]: model1 = LinearRegression()
         model2 = DecisionTreeRegressor()
         model3 = SVR()
```

OP[8]: ☑ LinearRegression

```
LinearRegression()
```

```
# In[9]: model1.fit(x_train,y_train)     #Linear Regression
         model2.fit(x_train,y_train)     #Decision Tree Regressor
         model3.fit(x_train,y_train)     #Support Vector Regression


# In[10]: y_pred1 = model1.predict(y_test)
          y_pred2 = model2.predict(y_test)
          y_pred3 = model3.predict(y_test)
```

- Evaluate the Model using Mean Squared Error and R2 score.

```
# In[11]: mse_linear = mean_squared_error(y_test,y_pred1)
          mse_DTree = mean_squared_error(y_test,y_pred2)
          mse_SVR = mean_squared_error(y_test,y_pred3)



# In[12]: r2_linear = r2_score(y_test,y_pred1)
          r2_DTree = r2_score(y_test,y_pred2)
          r2_SVR = r2_score(y_test,y_pred3)

# In[13]: accuracy_linear = accuracy_score(y_pred1 , y_test)
          accuracy_DTree = accuracy_score(y_pred2 , y_test)
          accuracy_SVR = accuracy_score(y_pred3 , y_test)


# In[14]: print("Mean Squared Error:",mse_linear)
          print("R-squared:",r2_linear)
          print("Accuracy",accuracy_linear)

# OP[14]: Mean Squared Error: 908.4528649741137
          R-squared: 0.19877081345863346
```

- Constructing a Time series plot for each Air Pollution:

```
# time series plot for each air pollutant
fig = go.Figure()

for pollutant in ['co', 'no', 'no2', 'o3', 'so2', 'pm2_5', 'pm10', 'nh3']:
    fig.add_trace(go.Scatter(x=data['date'], y=data[pollutant], mode='lines',
                             name=pollutant))

fig.update_layout(title='Time Series Analysis of Air Pollutants in Delhi',
                  xaxis_title='Date', yaxis_title='Concentration (µg/m³)')
fig.show()
```

- Calculating Air Quality Index using RSPM/PM10 and Categorizing Air Quality Index :

```
# Define AQI breakpoints and corresponding AQI values
aqi_breakpoints = [
    (0, 12.0, 50), (12.1, 35.4, 100), (35.5, 55.4, 150),
    (55.5, 150.4, 200), (150.5, 250.4, 300), (250.5, 350.4, 400),
    (350.5, 500.4, 500)
]

def calculate_aqi(pollutant_name, concentration):
    for low, high, aqi in aqi_breakpoints:
        if low <= concentration <= high:
            return aqi
    return None

def calculate_overall_aqi(row):
    aqi_values = []
    pollutants = ['co', 'no', 'no2', 'o3', 'so2', 'pm2_5', 'pm10', 'nh3']
    for pollutant in pollutants:
        aqi = calculate_aqi(pollutant, row[pollutant])
        if aqi is not None:
            aqi_values.append(aqi)
    return max(aqi_values)

# Calculate AQI for each row
data['AQI'] = data.apply(calculate_overall_aqi, axis=1)

# Define AQI categories
aqi_categories = [
```

```
    (0, 50, 'Good'), (51, 100, 'Moderate'), (101, 150, 'Unhealthy for
Sensitive Groups'),
    (151, 200, 'Unhealthy'), (201, 300, 'Very Unhealthy'), (301, 500,
'Hazardous')
]

def categorize_aqi(aqi_value):
    for low, high, category in aqi_categories:
        if low <= aqi_value <= high:
            return category
    return None

# Categorize AQI
data['AQI Category'] = data['AQI'].apply(categorize_aqi)
print(data.head())
```

- Air quality index over time:

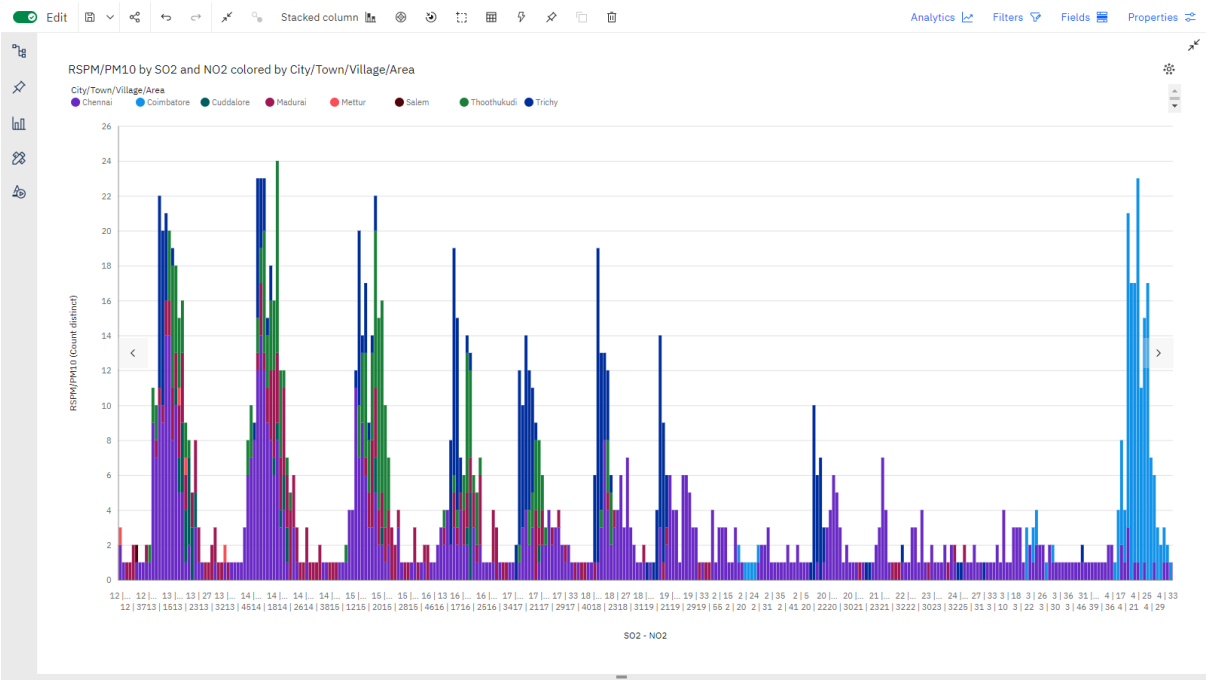
```
# AQI over time
fig = px.bar(data, x=" Sampling Date", y="AQI",
             title="AQI of Tamil Nadu in January")
fig.update_xaxes(title=" Sampling Date")
fig.update_yaxes(title="AQI")
fig.show()
```

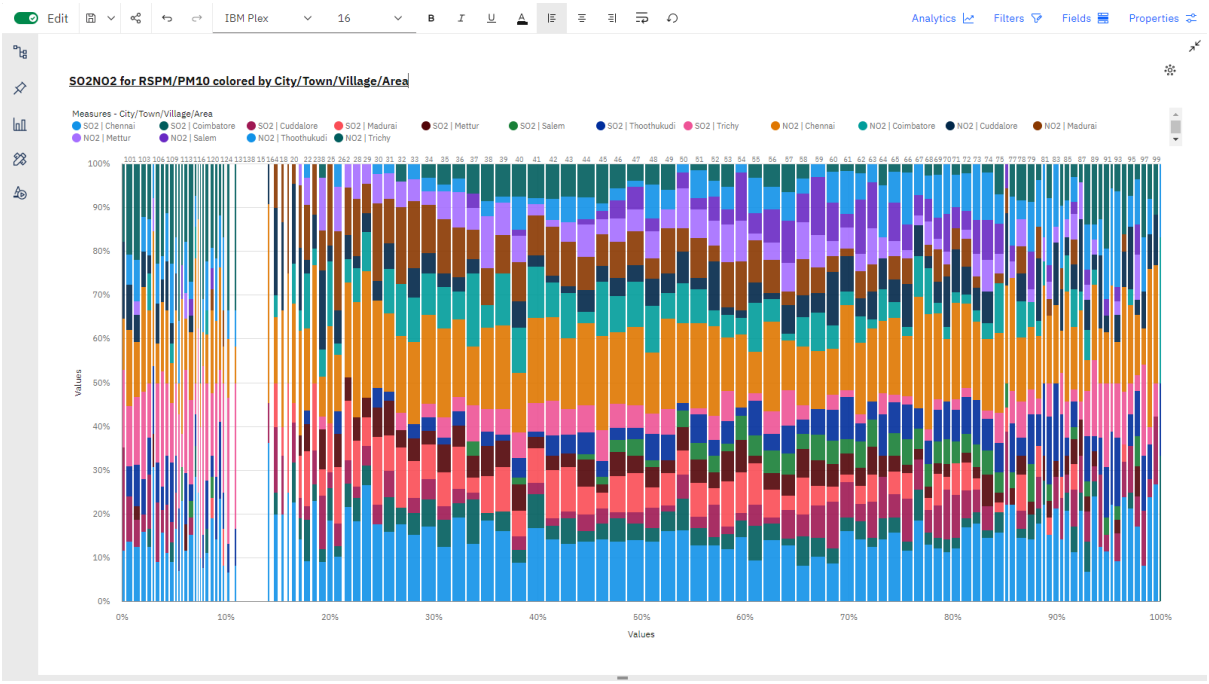
- Air quality index distribution over time:

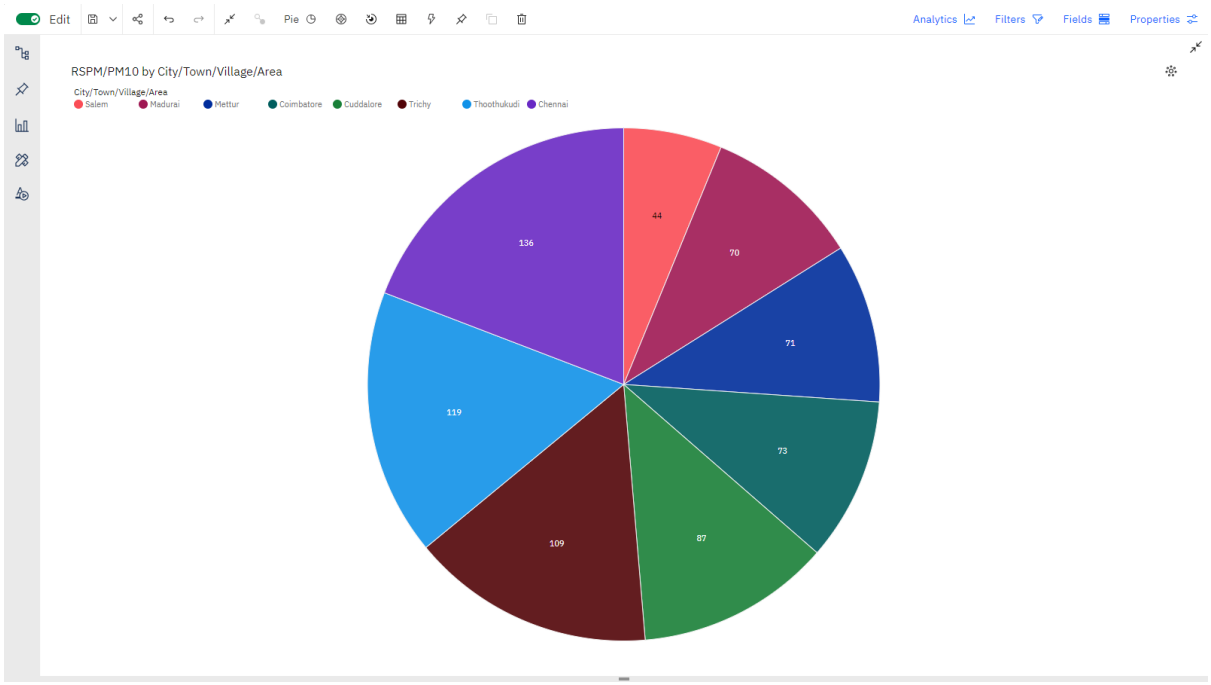
```
# AQI Category Distribution Over Time
fig = px.histogram(data, x=" Sampling Date ",
                   color="AQI Category",
                   title="AQI Category Distribution Over Time")
fig.update_xaxes(title=" Sampling Date ")
fig.update_yaxes(title="Count")
fig.show()
```

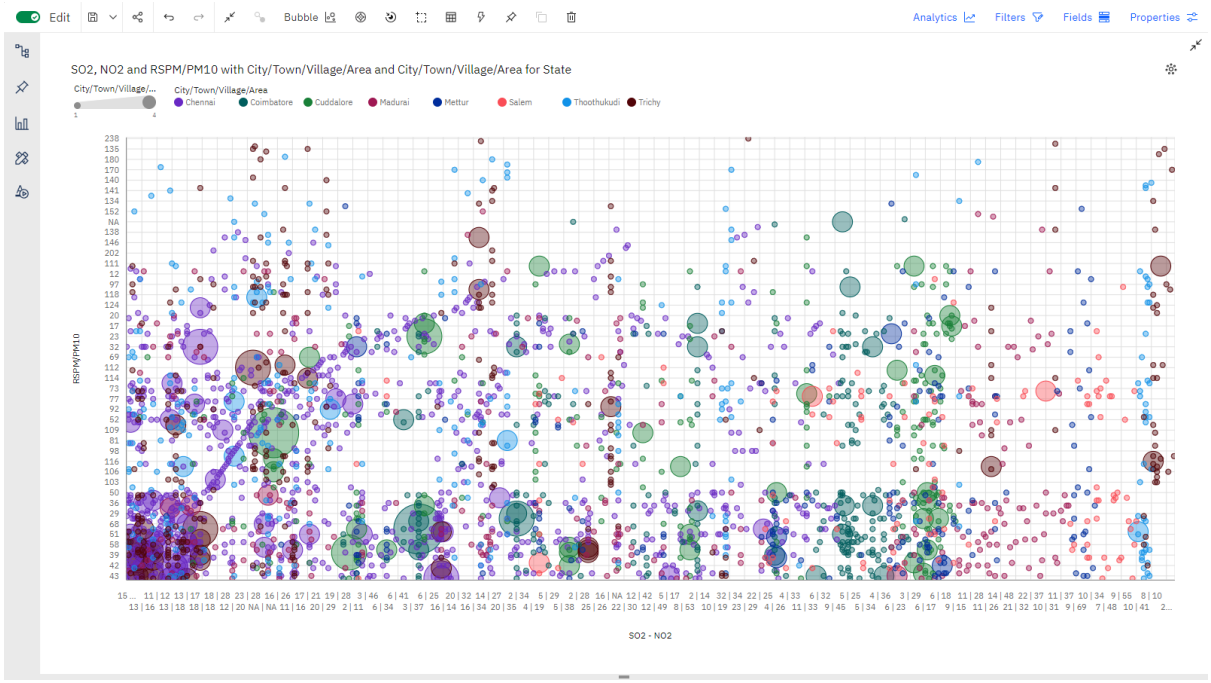
- Data Visualization can be done by IBM Cognos

The Below Data Visualization is to visualize the NO2 and SO2 using IBM Cognos.






Edit
Pie
Analytics
Filters
Fields
Properties
RSPM/PM10 by City/Town/Village/Area
City/Town/Village/Area
Salem
Madurai
Mettur
Coimbatore
Cuddalore
Trichy
Thoothukudi
Chennai
44
70
71
73
87
109
119
136


Edit
Bubble
Analytics
Filters
Fields
Properties
SO2, NO2 and RSPM/PM10 with City/Town/Village/Area and City/Town/Village/Area for State
City/Town/Village/...
City/Town/Village/Area
Chennai
Coimbatore
Cuddalore
Madurai
Mettur
Salem
Thoothukudi
Trichy
RSPM/PM10
SO2 - NO2

9. Conclusion:

The proposed approach aims to enhance the accuracy of predictive models for ambient air quality in Tamil Nadu through the incorporation of machine learning algorithms. The success of this project will lead to better air quality predictions, enabling more effective pollution control measures and safeguarding public health and the environment.