INSTRUCTIONS

Data Collection:

Find a reliable source for air quality data in Tamilnadu.

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

Data Loading:

* Once we have the dataset, we can load it into your Python environment.
* We can use libraries like Pandas to read data from various formats (CSV, Excel, etc.).

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import folium

from sklearn.preprocessing import LabelEncoder

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import classification\_report, confusion\_matrix, accuracy\_score

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import accuracy\_score

# Load the dataset

df = pd.read\_csv('C://air.csv')

print(df.head())

Data Cleaning:

* Data may have missing values or inconsistencies.
* We should clean the data by handling missing values, removing duplicates, and ensuring data types are correct.

# Handle missing values

df['SO2']=df['SO2'].fillna(0).astype('str').astype('float')

df['NO2']=df['NO2'].fillna(0).astype('str').astype('float')

df['RSPM/PM10']=df['RSPM/PM10'].fillna(0).astype('str').astype('float')

df['PM 2.5']=df['PM 2.5'].fillna(0).astype('str').astype('float')

df.drop(['Stn Code','Agency'],axis=1,inplace=True)

df=df.rename(index=str,columns={'Sampling Date':'year'})

print(df.info())

Data Analysis:

* Perform basic statistical analysis to understand the dataset.

# Calculate mean, median, etc.

mean\_so2 = df['SO2'].mean()

median\_so2 = df['SO2'].median()

std\_dev\_so2 = df['SO2'].std()

print(f"Mean SO2 Level: {mean\_so2}")

print(f"Median SO2 Level: {median\_so2}")

print(f"Standard Deviation SO2 Level: {std\_dev\_so2}")

Data Visualization:

* Create visualizations to better understand the air quality trends.
* Matplotlib and Seaborn are popular libraries for this.

#plot time series data for so2

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

plt.subplot(3, 1, 1)

plt.plot(df['SO2'], label='SO2 Levels', color='blue')

plt.title('SO2 Levels Over Time')

plt.legend()

# Plot time series data for NO2

plt.subplot(3, 1, 2)

plt.plot(df['NO2'], label='NO2 Levels', color='green')

plt.title('NO2 Levels Over Time')

plt.legend()

# Plot time series data for RSPM/PM10

plt.subplot(3, 1, 3)

plt.plot(df['RSPM/PM10'], label='RSPM/PM10 Levels', color='red')

plt.title('RSPM/PM10 Levels Over Time')

plt.legend()

plt.tight\_layout()

plt.show()

# Box Plot of rspm

plt.figure(figsize=(10, 6))

sns.boxplot(x='City/Town/Village/Area', y='RSPM/PM10', data=df)

plt.xlabel('City/Town/Village/Area')

plt.ylabel('RSPM/PM10 Levels')

plt.title('RSPM/PM10 Levels Across Cities (Box Plot)')

plt.xticks(rotation=45)

plt.show

# Heatmap for rspm

pivot\_table = df.pivot\_table(index='City/Town/Village/Area', columns='year', values='RSPM/PM10', aggfunc='mean')

plt.figure(figsize=(10, 6))

sns.heatmap(pivot\_table, cmap='YlGnBu', annot=True)

plt.xlabel('year')

plt.ylabel('City/Town/Village/Area')

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