**AIR QUALITY ANALYSIS AND PREDICTION IN TAMILNADU**

Phase 5:

**PROBLEM STATEMENT :**

The problem is to analyze and predict air quality in Tamilnadu, with a focus on understanding the current state of air pollution and forecasting future pollution levels. This involves collecting and analyzing data from various sources to assess the air quality in different regions of Tamilnadu, identifying key factors contributing to pollution, and developing models to predict air quality trends. The goal is to provide accurate information and early warnings to help residents and authorities take preventive measures to mitigate the impact of poor air quality on health and the environment.

**PROBLEM DEFINITIONS :**

**1.Data Collection :** Gather the air quality data from various monitoring stations in Tamil Nadu, ensuring it's reliable and up-to-date.

**2.Data Preprocessing :** Clean and prepare the data for analysis. This involves handling missing values, outliers, and ensuring data consistency.

**3.Exploratory Data Analysis (EDA) :** Perform EDA to gain initial insights into the data. Create visualizations like time series plots, heatmaps, or geographical maps to identify pollution trends and hotspots.

**4.Identify High Pollution Areas :** Use your EDA findings to pinpoint areas with consistently high pollution levels. This can help focus mitigation efforts.

**5.Feature Engineering :** Create relevant features for your predictive model. You mentioned predicting RSPM/PM10 levels based on SO2 and NO2, so feature engineering would include creating appropriate variables for this prediction.

**6.Model Building :** Choose a suitable machine learning or statistical model for your prediction task. Common choices include linear regression, decision trees, or neural networks.

**DESIGN THINKING :**

**Project Objectives :** Define objectives such as analyzing air quality trends, identifying pollution hotspots, and building a predictive model for RSPM/PM10 levels.

**Analysis Approach:** Plan the steps to load, preprocess, analyze, and visualize the air quality data.

**Visualization Selection :** Determine visualization techniques (e.g., line charts, heatmaps) to effectively represent air quality trends and pollution levels.

**1. Empathize :** Begin by understanding the specific air quality challenges in Tamilnadu, considering factors like geography, industrial activity, and population density. - Conduct interviews, surveys, and gather data to empathize with the concerns and needs of local communities, environmental experts, and policymakers.

**2. Define :** Clearly define the problem by identifying the key issues related to air quality, such as sources of pollution, seasonal variations, and health impacts. - Develop a user-centered problem statement, e.g., "How might we predict and mitigate air quality issues to improve the health and well-being of Tamilnadu residents?"

**3. Ideate :** Brainstorm potential solutions with a multidisciplinary team, including environmental scientists, data analysts, and community representatives. - Generate innovative ideas, such as using IoT sensors, satellite data, machine learning models, or policy interventions, to address air quality concerns.

**4. Prototype :** Create prototypes or proof-of-concept models of the proposed solutions. For example: - Build a prototype air quality monitoring system using IoT devices. - Develop a machine learning model to predict air quality based on historical data. - Design policy recommendations for reducing pollution from key sources. - Test these prototypes in a controlled environment to evaluate their effectiveness.

**5.Test and Implement :** Deploy the prototypes in real-world settings across Tamilnadu to collect data and assess their performance. - Gather feedback from users and stakeholders to refine and iterate on the solutions. - Once the solutions are proven effective, implement them on a larger scale, integrating them into the existing infrastructures.

**Innovation**

In this phase you need to put your design into innovation to solve the problem. Explain in detail the complete steps that will be taken by you to put your design that you thought of in previous phase into transformation.

**ALGORITHM OR STEPS TO PERFORM THE AIR QUALITY ANALYSIS USING MACHINE LEARNING**

**ALGORITHM:**

**1. Data Collection :** Gather data from air quality monitoring stations, sensors, or satellite imagery. This data may include measurements of pollutants like PM2.5, PM10, ozone, nitrogen dioxide (NO2), sulfur dioxide (SO2), carbon monoxide (CO), and volatile organic compounds (VOCs).

**2. Data Preprocessing :** Clean and preprocess the collected data to remove outliers, missing values, and errors. Ensure that the data is in a consistent format for analysis.

**3. Data Visualization :** Create visual representations of the data through graphs, charts, and maps to identify patterns and trends. Visualization can help in understanding air quality variations over time and location.

**4. Descriptive Statistics:** Calculate basic statistics such as mean, median, standard deviation, and percentiles to describe the central tendencies and variations in pollutant concentrations.

**5. Time-Series Analysis :** Analyze data over time to identify seasonal, daily, or hourly patterns. This can help in understanding how air quality changes throughout the day or year**.**

**6. Spatial Analysis :**

Examine the spatial distribution of pollutants to identify areas with higher or lower air quality. Geographic Information Systems (GIS) tools are often used for this purpose.

**7. Correlation Analysis :** Investigate relationships between different pollutants and meteorological variables. For example, you can assess how temperature, wind speed, and humidity correlate with pollutant levels**.**

**8. Regression Analysis :** Use regression models to predict air quality based on factors like emissions, weather conditions, and geographical features.

**9. Health Impact Assessment :** Assess the potential health effects of the observed air quality by linking it to relevant health data, such as hospital admissions or respiratory conditions.

**10. Compliance Assessment :** Compare air quality data against established air quality standards and regulations to determine whether specific areas meet the required air quality levels

**PERFORMING AIR QUALITY ANALYSIS USING MACHINE LEARNING:**

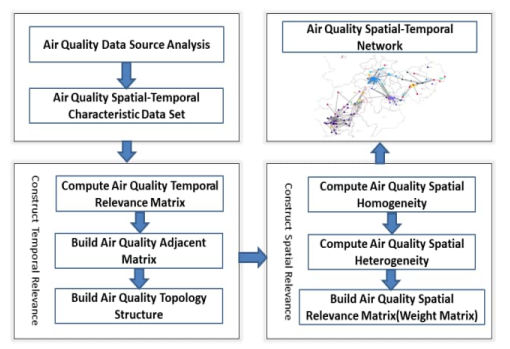
**1. Data Collection :** Gather air quality data from reliable sources, such as government monitoring stations, sensors, or publicly available datasets. This data should include information on pollutant concentrations (e.g., PM2.5, PM10, NO2, CO, SO2), meteorological conditions (temperature, humidity, wind speed, etc.), and geographical information.

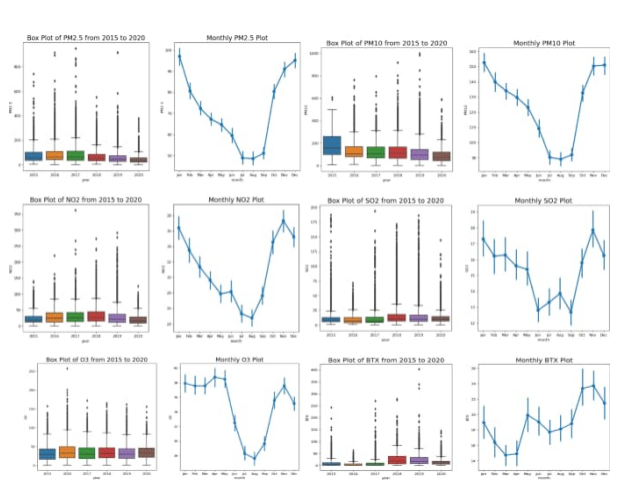
**2. Data Preprocessing** : Clean and prepare the data for analysis. This includes handling missing values, outliers, and converting data into a suitable format. It's essential to ensure data quality and consistency.

**3. Exploratory Data Analysis (EDA) :** Visualize the data to identify trends, patterns, and anomalies. EDA helps in understanding the distribution of pollutants, their correlations with meteorological factors, and any temporal or spatial variations.

**4. Feature Engineering** : Create new features or variables that can enhance the predictive power of your models. For example, you can calculate rolling averages, time lags, or use geographic features like proximity to pollution sources.

**5. Model Selection** : Choose appropriate data science models for air quality prediction. Common choices include regression models, time series models, machine learning algorithms like Random Forest, Gradient Boosting, or deep learning models (e.g., neural networks).

**FLOW CHART**

**EXPECTED OUTPUT:**

**Introduction**

Air quality can be defined as the measurement of quality of the air we breathe and the concentrations of the pollutants in the air that can cause various health issues. Air quality can be used for various purposes such as the communication of air quality with the public, to plan strategies that can be used to reduce air pollution, and to monitor short term and long term trends. Air quality can be measured using various machine learning algorithms.

**GIVEN DATA SET:**

** Necessary step to follow :**

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import plotly.plotly as py

%matplotlib inline

plt.rcParams['figure.figsize'] = (10, 7)

import warnings

warnings.filterwarnings('ignore')

import os

print(os.listdir("../input"))

['data.csv']

data=pd.read\_csv('../input/data.csv',encoding = "ISO-8859-1")

data.head()

**Output :**

****

**Location Monitoring Station :**

data.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 435742 entries, 0 to 435741

Data columns (total 13 columns):

stn\_code 291665 non-null object

sampling\_date 435739 non-null object

state 435742 non-null object

location 435739 non-null object

agency 286261 non-null object

type 430349 non-null object

so2 401096 non-null float64

no2 419509 non-null float64

rspm 395520 non-null float64

spm 198355 non-null float64

location\_monitoring\_station 408251 non-null object

pm2\_5 9314 non-null float64

date 435735 non-null object

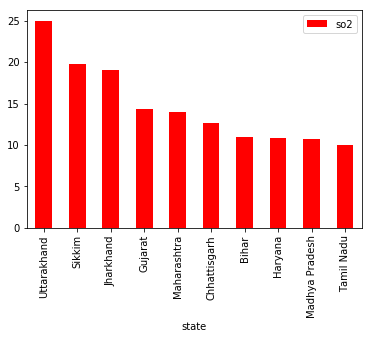
dtypes: float64(5), object(8)

memory usage: 43.2+ MB

**Measuring Places :**

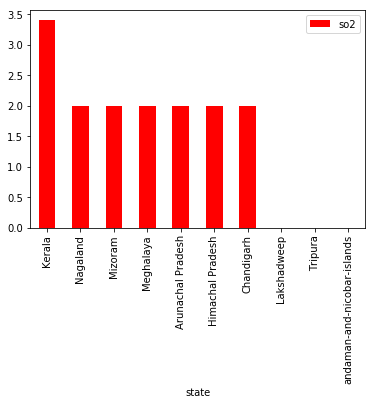
data[['so2','state']].groupby(["state"]).median().sort\_values(by='so2',ascending=False).head(10).plot.bar(color='r')

plt.show()



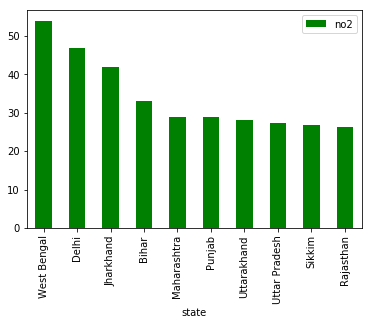
data[['so2','state']].groupby(["state"]).median().sort\_values(by='so2',ascending=False).tail(10).plot.bar(color='r')

plt.show()



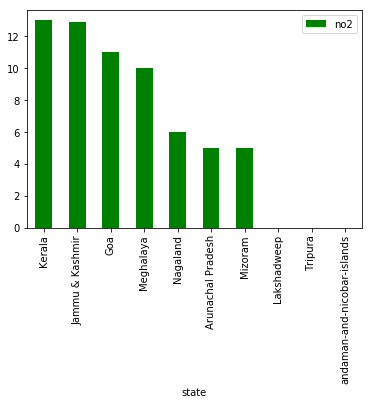
data[['no2','state']].groupby(["state"]).median().sort\_values(by='no2',ascending=False).head(10).plot.bar(color='g')

plt.show()



data[['no2','state']].groupby(["state"]).median().sort\_values(by='no2',ascending=False).tail(10).plot.bar(color='g')

plt.show()



data[['rspm','state']].groupby(["state"]).median().sort\_values(by='rspm',ascending=False).head(10).plot.bar(color='b')

plt.show()

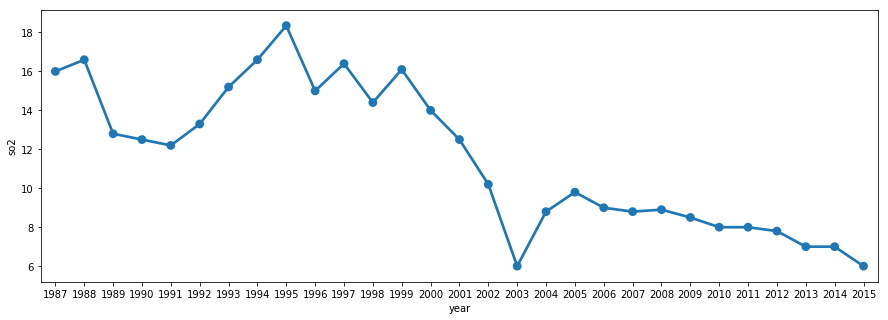
**SO2 ANALYSIS :**

df = data[['so2','year','state']].groupby(["year"]).median().reset\_index().sort\_values(by='year',ascending=False)

f,ax=plt.subplots(figsize=(15,5))

sns.pointplot(x='year', y='so2', data=df)

<matplotlib.axes.\_subplots.AxesSubplot at 0x7f7a975db208>

****

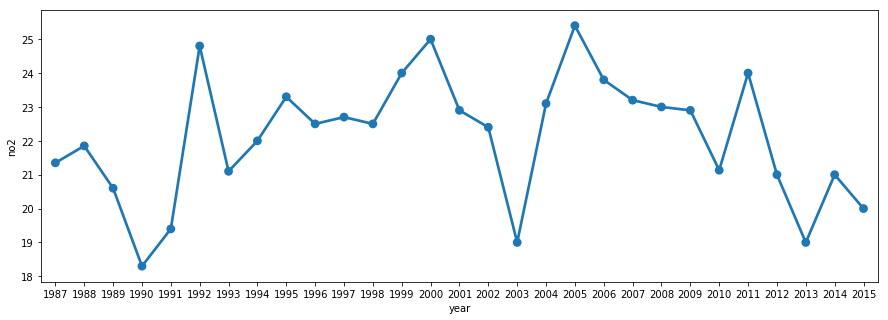
**NO2 ANALYSIS :**

df = data[['no2','year','state']].groupby(["year"]).median().reset\_index().sort\_values(by='year',ascending=False)

f,ax=plt.subplots(figsize=(15,5))

sns.pointplot(x='year', y='no2', data=df)

<matplotlib.axes.\_subplots.AxesSubplot at 0x7f7a90c69d68>

****

**COMMON DATAS FOR AIR QUALITY ANALYSIS :**

When conducting an air quality analysis, it's essential to gather various data to assess the levels of pollutants and their impact on the environment and human health. Here are some common types of data and parameters to consider:

**1. Particulate Matter (PM):**

- PM2.5 (particles with a diameter of 2.5 micrometers or smaller)

- PM10 (particles with a diameter of 10 micrometers or smaller)

**2. Gaseous Pollutants:**

- Nitrogen Dioxide (NO2)

- Sulfur Dioxide (SO2)

- Carbon Monoxide (CO)

- Ozone (O3)

- Volatile Organic Compounds (VOCs)

**3. Meteorological Data:**

- Temperature

- Relative Humidity

- Wind speed and direction

- Atmospheric pressure

- Precipitation

**4. Geographic Data:**

- GPS coordinates or location information

- Topography and land use data

- Elevation

**5. Time Data:**

- Time and date of measurements

- Data collection frequency (e.g., hourly, daily, or continuous monitoring)

**6. Source Emissions Data:**

- Information on local industrial and traffic sources

- Emission inventories

**7. Health Data:**

- Information on health outcomes and population demographics in the area

- Hospital admissions and emergency room visits related to air quality

**8. Air Quality Index (AQI) Data:**

- Calculated AQI values based on pollutant concentrations

**9. Historical Data:**

- Long-term trends and historical records of air quality

- Seasonal variations

**10. Air Quality Monitoring Equipment Data:**

- Calibration and maintenance records

- Sensor types and specifications

**Necessary steps to follow**:

**Algorithm Selection:**

Choose a suitable machine learning algorithm based on the nature of your project. Common choices include decision trees, support vector machines, neural networks, and more.

**Linear Regression:**

Use for regression problems when you want to predict a continuous target variable.

**Logistic Regression:**

Suitable for binary classification problems (e.g., spam detection).

Z = spam[‘EmailText’]

y = spam[“Label”]

z\_train, z\_test,y\_train, y\_test = train\_test\_split(z,y,)

**Day level**

To get AQI at day level, the AQI values are averaged over the hours of the day.

Df\_station\_hour = df

df\_station\_day = pd.read\_csv(PATH\_STATION\_DAY)

df\_station\_day = df\_station\_day.merge(df.groupby([“StationId”, “Date”])[“AQI\_calculated”].mean().reset\_index(), on = [“StationId”, “Date”])

df\_station\_day.AQI\_calculated = round(df\_station\_day.AQI\_calculated)

**City level**

To get AQI at city level, the AQI values are averaged over stations of the city.

Df\_city\_hour = pd.read\_csv(PATH\_CITY\_HOUR)

df\_city\_day = pd.read\_csv(PATH\_CITY\_DAY)

df\_city\_hour[“Date”] = pd.to\_datetime(df\_city\_hour.Datetime).dt.date.astype(str)

df\_city\_hour = df\_city\_hour.merge(df.groupby([“City”, “Datetime”])[“AQI\_calculated”].mean().reset\_index(), on = [“City”, “Datetime”]) df\_city\_hour.AQI\_calculated = round(df\_city\_hour.AQI\_calculated)

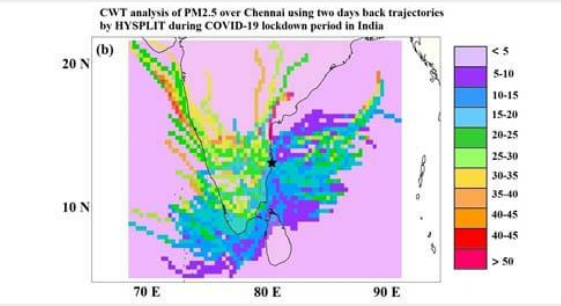
df\_city\_day = df\_city\_day.merge(df\_city\_hour.groupby([“City”, “Date”])[“AQI\_calculated”].mean().reset\_index(), on = [“City”, “Date”])

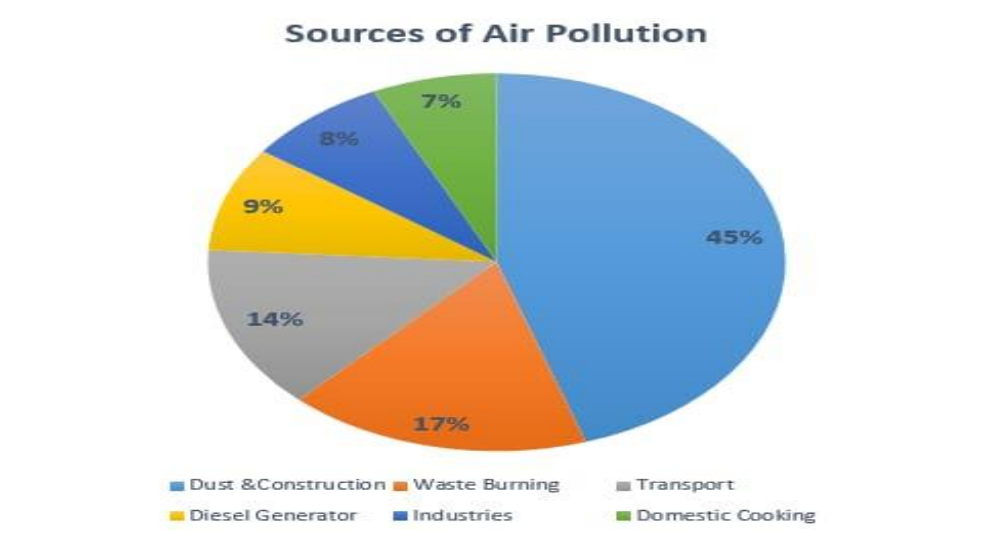
df\_city\_day.AQI\_calculated = round(df\_city\_day.AQI\_calculated)

**Data Visualization**

datacount =sns.countplot(x =“location”,data = tn);

datacount.set\_xticklabels(datacount.get\_xticklabels(), rotation=90);





**Conclusion:**

Air pollution is a global problem; researchers from all around the world are working to discover a solution. To accurately forecast the AQI, machine learning techniques were investigated. The present study assessed the performance of the three best data mining models (SVR, RFR, and CR) for predicting the accurate AQI data in some of India’s most populous and polluted cities. The synthetic minority oversampling technique (SMOTE) was used to equalize the class data to get better and consistent results.