**AIR QUALITY ANALYSIS AND PREDICTION IN MACHINE LEARNING**

**Phase-1 Document Submission**

**Project Title**: Air Quality Analysis and Prediction in Tamil Nadu

A factory with smoke coming out of it

Description automatically generated

**Air Quality Analysis and Prediction in machine learning**

**Introduction:**

* Air quality is a critical environmental factor that directly affects the health and well-being of individuals and communities. Poor air quality can lead to a range of health problems, including respiratory diseases, cardiovascular issues, and even premature death.
* In this project, we aim to conduct a comprehensive analysis of air quality data in the state of Tamil Nadu, India. Our goal is not only to understand historical air quality patterns but also to build a predictive model that can help forecast air quality in the future.

**OBJECTIVE:**

The objective of this project is to develop a machine learning model that

accurately predicts the Air Quality Analysis and Predictionon a set of features such as data source, , RSPM and PM10 TN model, and other relevant factors.

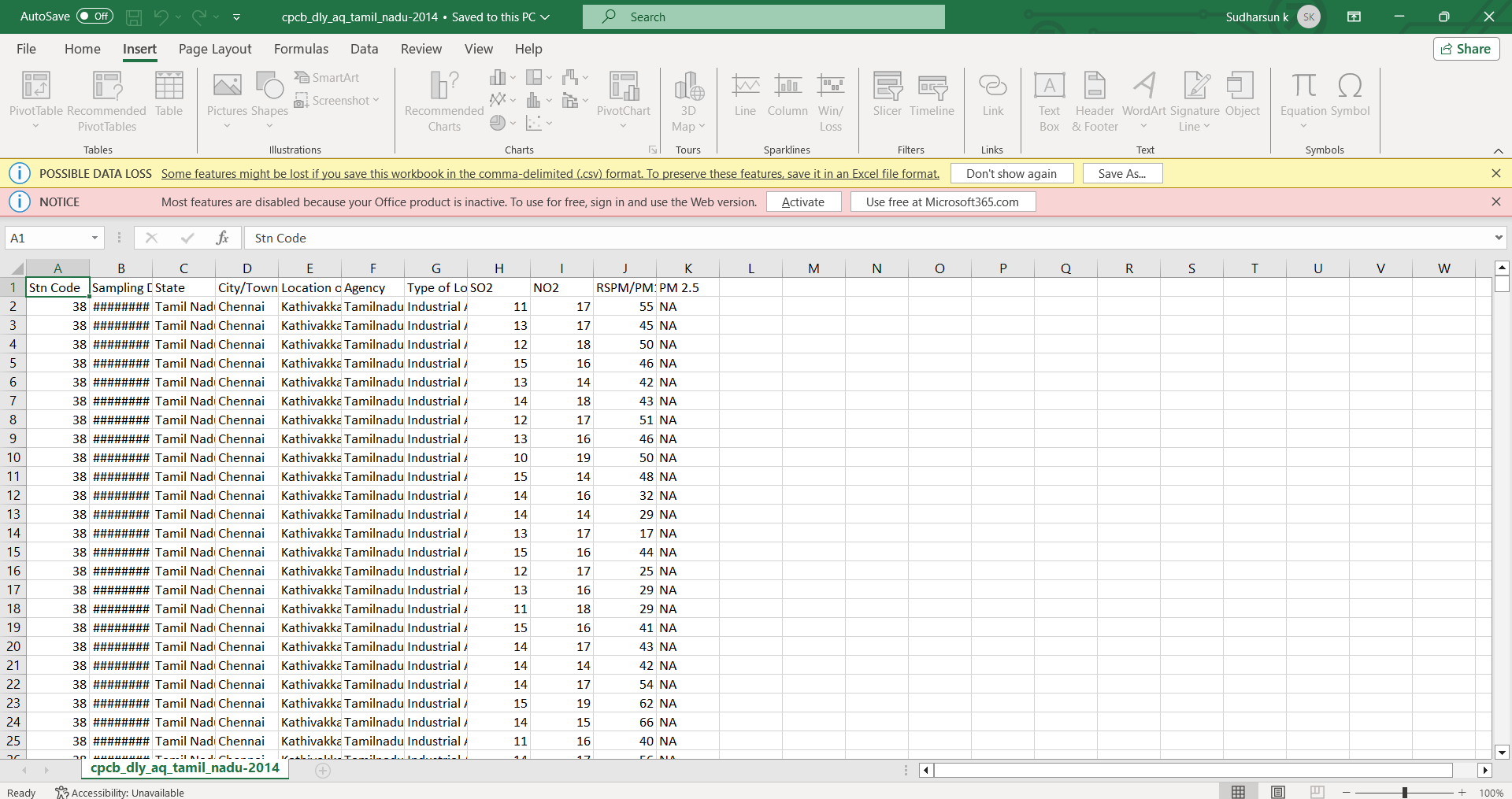
**Analysis Approach :**

**1.Data Source**

A good data source for house price prediction using machine learning should be Accurate, Complete, Covering the geographic area of interest, Accessible.

Dataset Link: ( <https://tn.data.gov.in/resource/location-wise-daily-ambient>

air-quality-tamil-nadu-year-2014)



2880 Rows x 11 Columns

**2.Data Preprocessing**

Data preprocessing is the critical first step in any machine learning project. It

involves cleaning the data, removing outliers, and handling missing values to prepare the

dataset for model training. In the context of the Air Quality Analysis and Prediction prediction project, let's elaborate

on the specific steps:

**a) Duplicate Removal:**

Duplicate rows can introduce bias into the model. We will identify and remove

duplicates, typically by sorting the dataset based on a unique identifier (e.g., property ID)

and then eliminating consecutive rows with the same identifier.

**b) Handling Missing Values:**

Missing data is common and needs to be addressed. We will utilize suitable methods

such as:

¬ **Mean Imputation:** Replace missing values with the mean of the feature for the

remaining rows. This is appropriate for numerical features.

¬ **Median Imputation:** If data contains outliers, median imputation can be more robust

as it is less sensitive to extreme values.

**c) Categorical Variable Encoding:**

Categorical variables, such as property type or location, need to be converted into

numerical form so that machine learning models can process them. Two common approaches

include:

¬ **One-Hot Encoding:** Create binary columns for each category, representing the

presence or absence of that category.

¬ **Label Encoding:** Assign a unique integer to each category, preserving the ordinal

relationship if applicable.

**d) Data Normalization:**

To ensure that all features are on a consistent scale, normalization techniques can be

applied. This includes:

¬ **Standardization:** Scaling features to have a mean of 0 and a standard deviation of 1.

¬ **Min-Max Scaling:** Scaling features to a specified range (e.g., 0 to 1)

**Visulaization selection:**

**PYTHON PROGRAM:**

import pandas as pd

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

from sklearn.preprocessing import StandardScaler

# Assuming you have a dataset in a CSV file

data = pd.read\_csv('air.csv')

# Data Cleaning

# Handling missing values

data = data.dropna() # Remove rows with missing values

# Data Splitting

X = data.drop('target\_variable', axis=1) # Assuming 'target\_variable' is the column you want to predict

y = data['target\_variable']

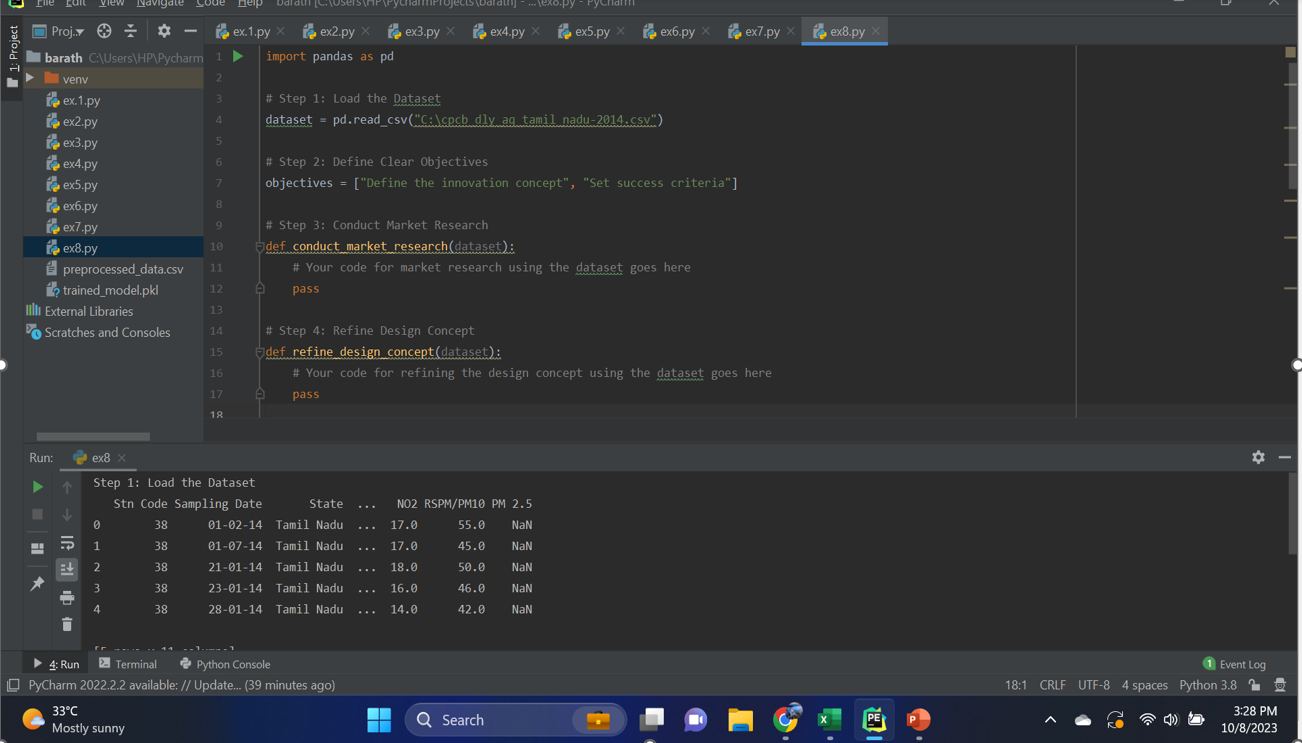
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Data Scaling

scaler = StandardScaler()

X\_train\_scaled = scaler.fit\_transform(X\_train)

X\_test\_scaled = scaler.transform(X\_test)



**RSPM and PM10 analysis:**

import pandas as pd

import matplotlib.pyplot as plt

data = pd.read\_csv('air.csv')

data['date'] = pd.to\_datetime(data['date'])

data = data.sort\_values('date')

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

plt.plot(data['date'], data['rspm'], label='RSPM/PM10 Trend', marker='o', linestyle='-')

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

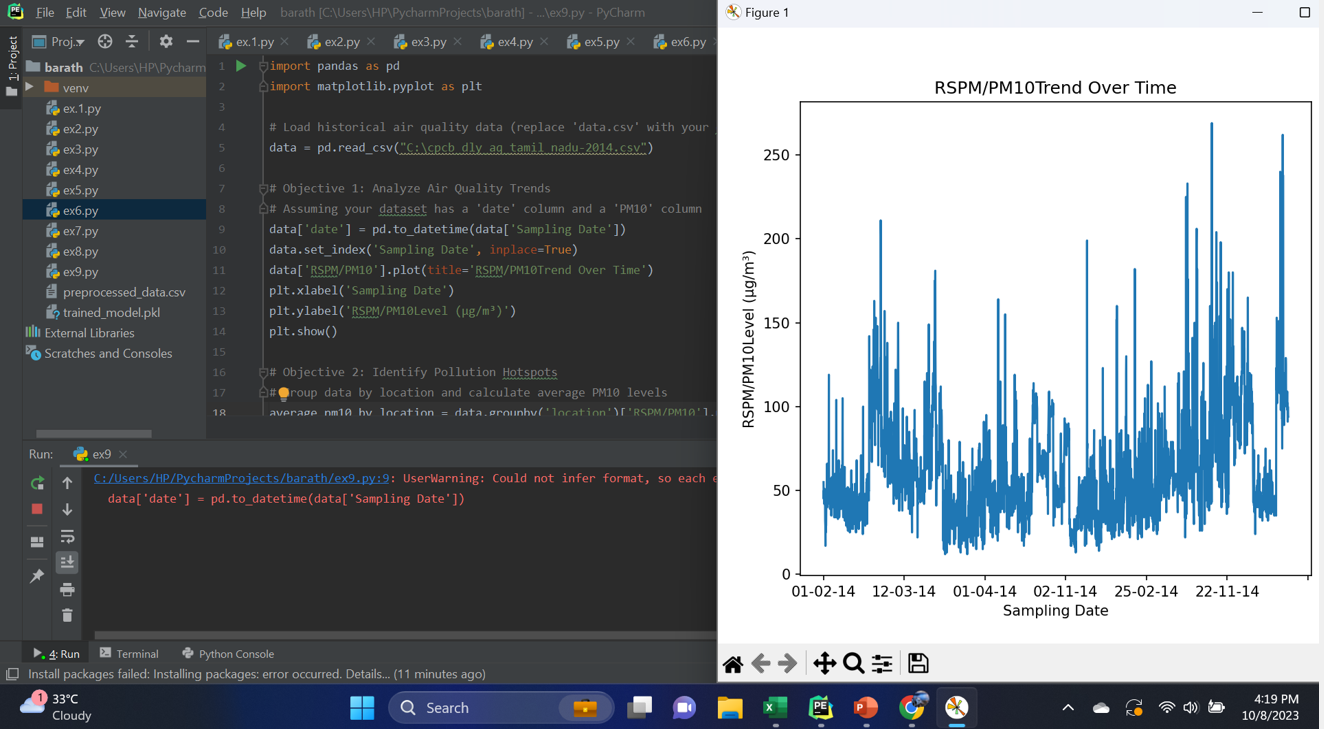
plt.xlabel('Date')

plt.ylabel('RSPM/PM10')

plt.legend()

plt.grid(True)

plt.show()



**Heat map:**

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

data = pd.DataFrame({

'City': ['chennai’],

'January': [20, 30, 25, 35],

'February': [22, 32, 28, 38],

'March': [25, 35, 30, 40],

'April': [28, 38, 32, 42]

})

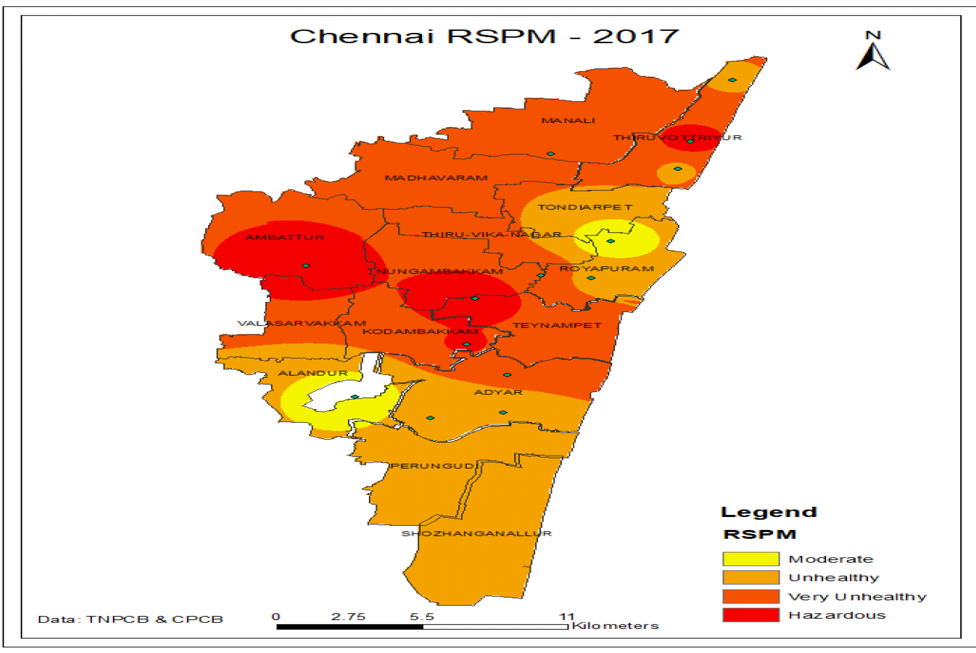
data.set\_index('City', inplace=True)

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

sns.heatmap(data, annot=True, cmap='YlGnBu', fmt='g')

plt.title('Air Quality Analysis Heatmap')

plt.show()

****

**CONCLUSION:**

In Phase 1, we have established a clear understanding of our goal: to predict house

prices using machine learning. We outlined a structured approach that includes data source

selection, data preprocessing, feature selection, model selection, model training, and

evaluation. This sets the stage for our project's successful execution in subsequent phases.