# Phase 5: Project Documentation and Submission

Project 9: Air Quality Analysis and Prediction in Tamil Nadu

**PROBLEM STATEMENT:**

The objective of this project is to employ data science techniques to analyze air quality in various regions of Tamil Nadu, India. This analysis aims to provide insights into air pollution levels, identify key contributors, and develop predictive models for air quality. The project will use historical and real-time data sources to create actionable recommendations for improving air quality in the region.

**DESIGN THINKING:**

Gather relevant data sources. In Tamil Nadu, this might include air quality monitoring stations, weather data, satellite imagery, and pollution sources data.

**PHASE DEVELOPMENT**:

PHASE 1:

* In this phase the design thinking and the project definition of air quality analysis and prediction was developed.

PHASE 2:

* This phase includes the innovative thinking of this project and it contains the detailed description of the given dataset.

PHASE 3:

* With The brief explanation of above two phases the project is being developed here by loading the dataset.

PHASE 4:

* By proceeding ,the further development is made with data processing ,data cleaning ,EDA and also the data visualisation.

PHASE 5:

* Here comes the final completion which we have done .this phase has complete documentation and the scripts related to this project have been attached.

**Dataset collection:**

Gather historical and real-time air quality data from government agencies, environmental sensors, and satellite sources. This data should identifying pollution hotspots, and building a predictive model for RSPM/PM10 levels and cover various pollutants such NO2, SO2  .

**Pollutant Analysis:**

Analyze the concentrations of major air pollutants including SO2,NO2,O3.

**Geographic Variation :**

Identify regions with highest levels of air pollution and understand the factors contributing to this variation.

**Building Predictive Model :**

The dataset will be using from <https://tn.data.gov.in/resource/location-wise-daily-ambient-air-quality-tamil-nadu-year-2014>

* First and foremost, import the necessary python libraries like matplotlib, scikit, seaborn etc.
* Explore the data set. Python has several [functions](https://365datascience.com/tutorials/python-tutorials/python-functions/) that will help you with your explorations.
* Feature the selections that contribute more to dataset.
* Select only those that have the strongest relationship with the predicted variable.
* To build model, train and split the dataset and the testing is done using regression techniques.

The dataset is being collected from <https://www.kaggle.com/datasets>. Here we go to look up the details of the respected Dataset.

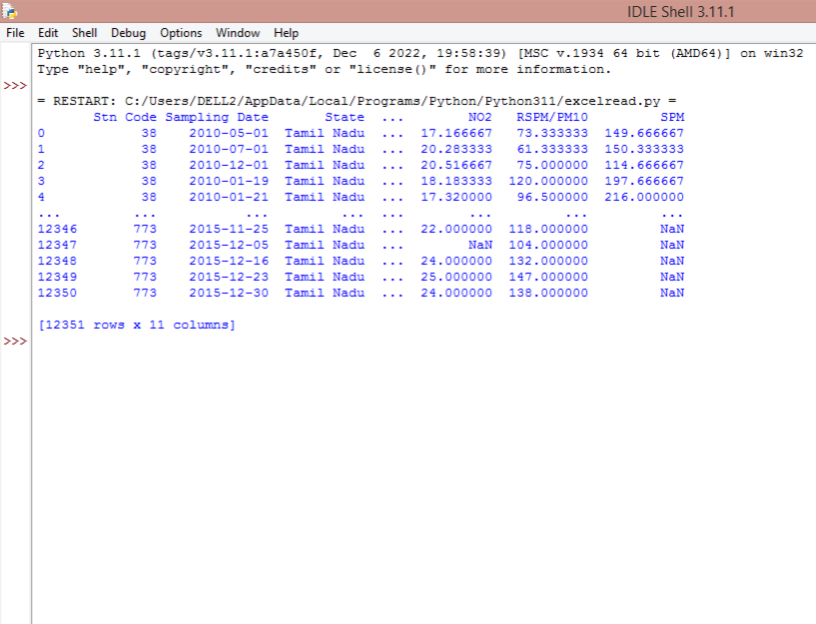
* The dataset is taken and it is descript as Air-Quality-data-Tamil-Nadu.
* The dataset contains the trends of air quality from the year 2010 to 2015 from various districts in Tamil Nadu .

This shows the various columns which is responsible for the air quality.

**CODING:**

|  |
| --- |
| import pandas as pd  df=pd.read\_excel("C:\\Users\\DELL2\\Downloads\\Air\_Quality.xlsx")  print(df) |

OUTPUT :



**Data Cleaning :**

* Data cleansing, also referred to as data cleaning or data scrubbing, is the process of fixing incorrect, incomplete, duplicate or otherwise erroneous data in a data set.
* The drop() method removes the specified row or column.
* The ‘agency’ column is removed from the dataset as its contains duplicate columns.

|  |
| --- |
| df.drop(columns='Agency',inplace=True)  print(df) |

OUTPUT :

|  |
| --- |
| = RESTART: C:/Users/DELL2/AppData/Local/Programs/Python/Python311/excelread.py =  Stn Code Sampling Date State ... NO2 RSPM/PM10 SPM  0 38 2010-05-01 Tamil Nadu ... 17.166667 73.333333 149.666667  1 38 2010-07-01 Tamil Nadu ... 20.283333 61.333333 150.333333  2 38 2010-12-01 Tamil Nadu ... 20.516667 75.000000 114.666667  3 38 2010-01-19 Tamil Nadu ... 18.183333 120.000000 197.666667  4 38 2010-01-21 Tamil Nadu ... 17.320000 96.500000 216.000000  [5 rows x 11 columns]  Stn Code Sampling Date State ... NO2 RSPM/PM10 SPM  0 38 2010-05-01 Tamil Nadu ... 17.166667 73.333333 149.666667  1 38 2010-07-01 Tamil Nadu ... 20.283333 61.333333 150.333333  2 38 2010-12-01 Tamil Nadu ... 20.516667 75.000000 114.666667  3 38 2010-01-19 Tamil Nadu ... 18.183333 120.000000 197.666667  4 38 2010-01-21 Tamil Nadu ... 17.320000 96.500000 216.000000  ... ... ... ... ... ... ... ...  12346 773 2015-11-25 Tamil Nadu ... 22.000000 118.000000 NaN  12347 773 2015-12-05 Tamil Nadu ... NaN 104.000000 NaN  12348 773 2015-12-16 Tamil Nadu ... 24.000000 132.000000 NaN  12349 773 2015-12-23 Tamil Nadu ... 25.000000 147.000000 NaN  12350 773 2015-12-30 Tamil Nadu ... 24.000000 138.000000 NaN  [12351 rows x 10 columns] |

**Data Preprocessing:**

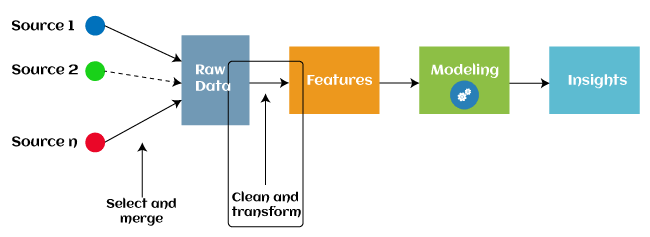
Clean and preprocess the collected data to handle missing values,outliers, and ensure data consistency..

**Feature Engineering :**

* **Feature engineering is the pre-processing step of machine learning, which extracts features from raw data.**
* It helps to represent an underlying problem to predictive models in a better way, which as a result, improve the accuracy of the model for unseen data.

**Steps involved in feature engineering are :**

* Selecting, extracting, and transforming the most relevant features from the available data
* Manipulating the data set to improve machine learning model training
* Creating new variables that aren't in the training set
* Remodelling raw data into a format that successfully represents the underlying patterns within the data
* Ensuring the model is flexible in the variety of data it can ingest
* Ensuring variables are on the same scale
* Making the model easier to understand
* Improving accuracy
* Avoiding computational errors



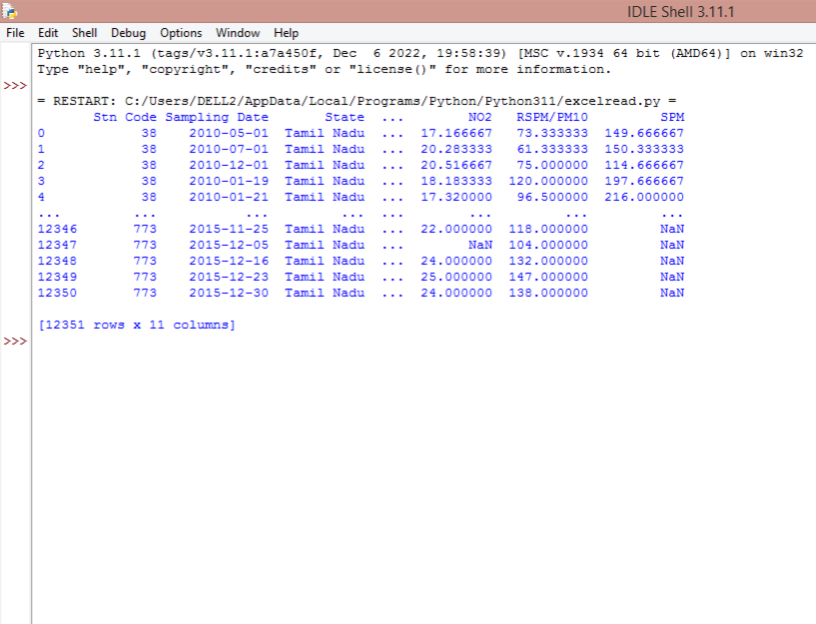
**Exploratory data analysis:**

* EDA is a phenomenon under data analysis used for gaining a better understanding of data aspects like main features of data, variables and relationships that hold between them ,identifying which variables are important for our problem.
* It uses data manipulation techniques and several statistical tools to describe and understand the relationship between variables and how these can impact business.

**CODING:**

|  |
| --- |
| import pandas as pd  df=pd.read\_excel("C:\\Users\\DELL2\\Downloads\\Air\_Quality.xlsx")  print(df)  #printing head  print(df.head()) |

**OUTPUT :**



**Renaming the columns :**

**CODING:**

|  |
| --- |
| df.rename(columns={'City/Town/Village/Area':'City'})  #RENAMING COLUMN  d=df.rename(columns={'City/Town/Village/Area':'City'})  s=df.columns  print(s)  #TITLE OF COLUMNS  print(d.columns) |

OUTPUT :

|  |
| --- |
| Index(['Stn Code', 'Sampling Date', 'State', 'City','Location of Monitoring Station', 'Agency', 'Type of Location', 'SO2', 'NO2', 'RSPM/PM10', 'SPM'],dtype='object') |

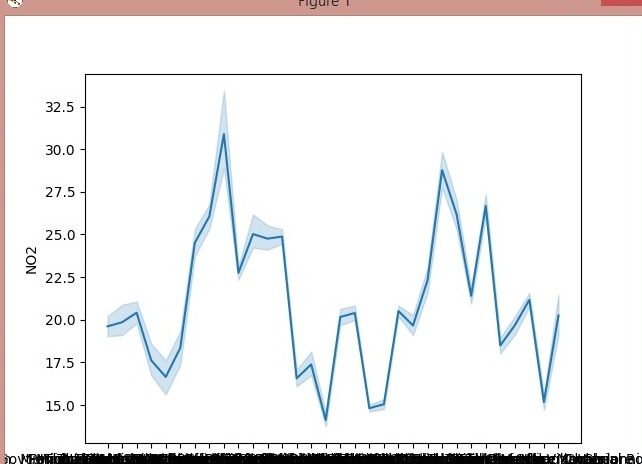
**INNOVATIVE TECHNIQUE:**

**Data Visualization :**

**Line chart:**

import matplotlib.pyplot as plt  
import seaborn as sns  
  
sns.lineplot(x='Location of Monitoring Station', y='NO2', data=df)  
plt.show()

**OUTPUT:**



**Barchart:**

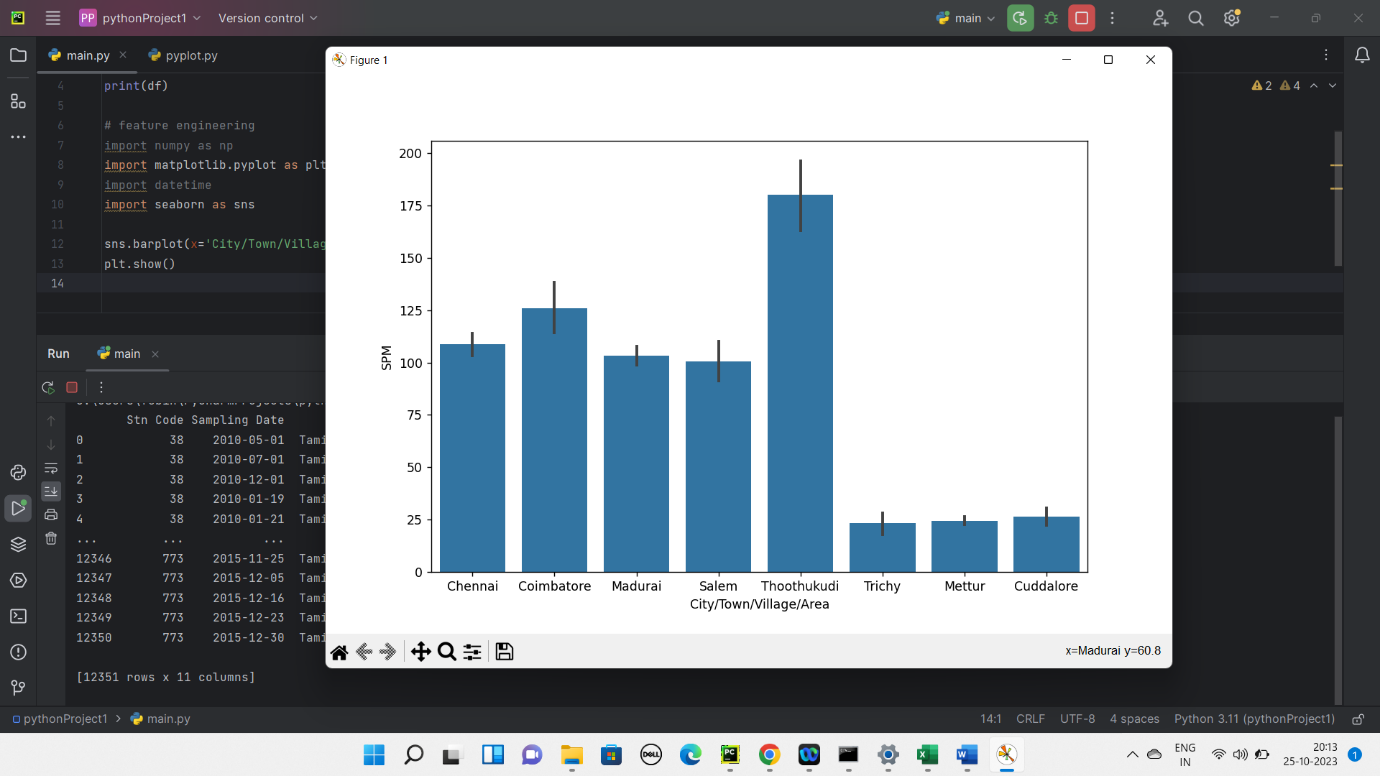
import matplotlib.pyplot as plt

import seaborn as sns

sns.barplot(x='City/Town/Village/Area', y='SPM', data=df)

plt.show()

**OUTPUT:**



**Scatter Chart:**

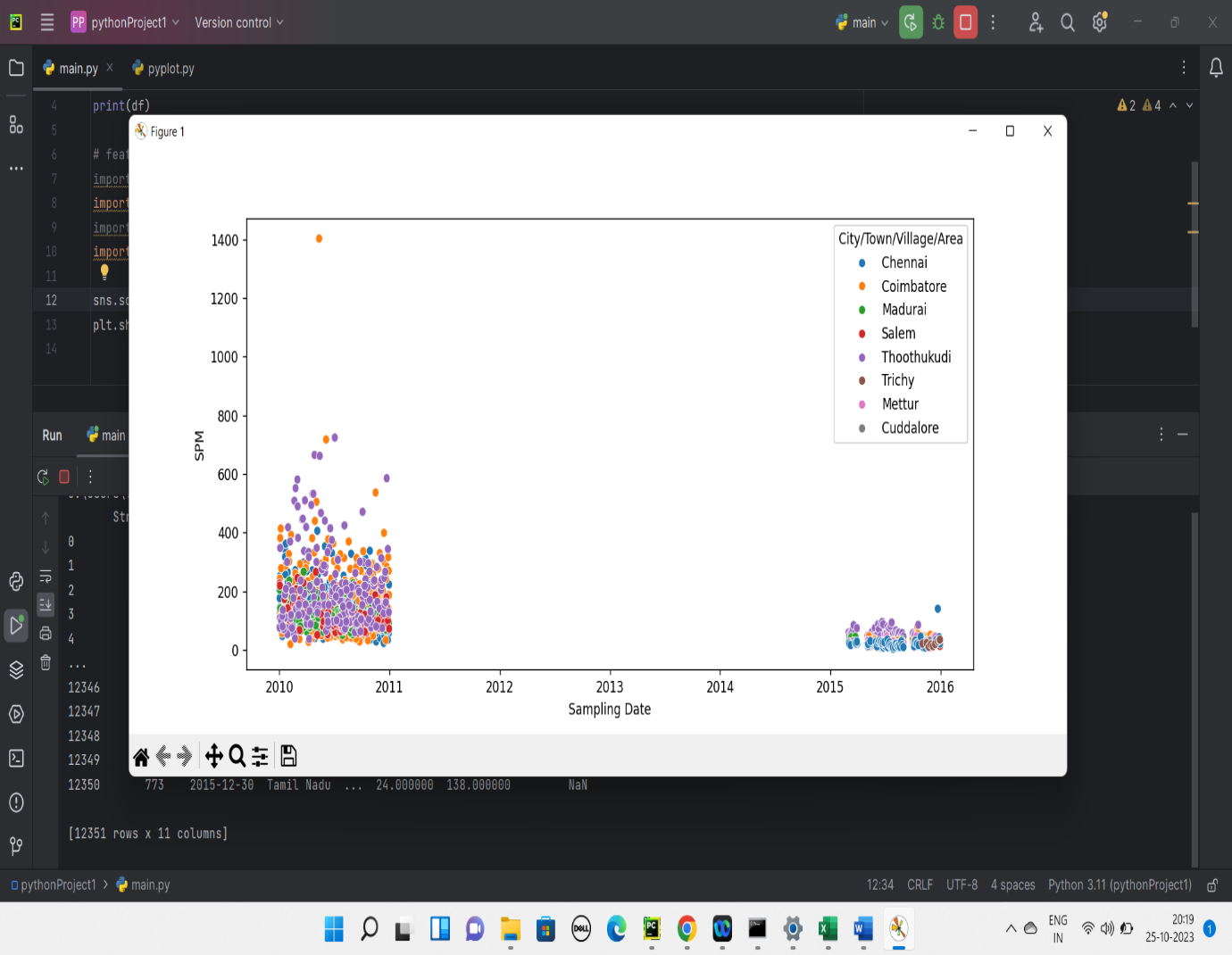
import matplotlib.pyplot as plt

import seaborn as sns

sns.scatterplot(x='Sampling Date', y='SPM', data=df, hue='City/Town/Village/Area')

plt.show()

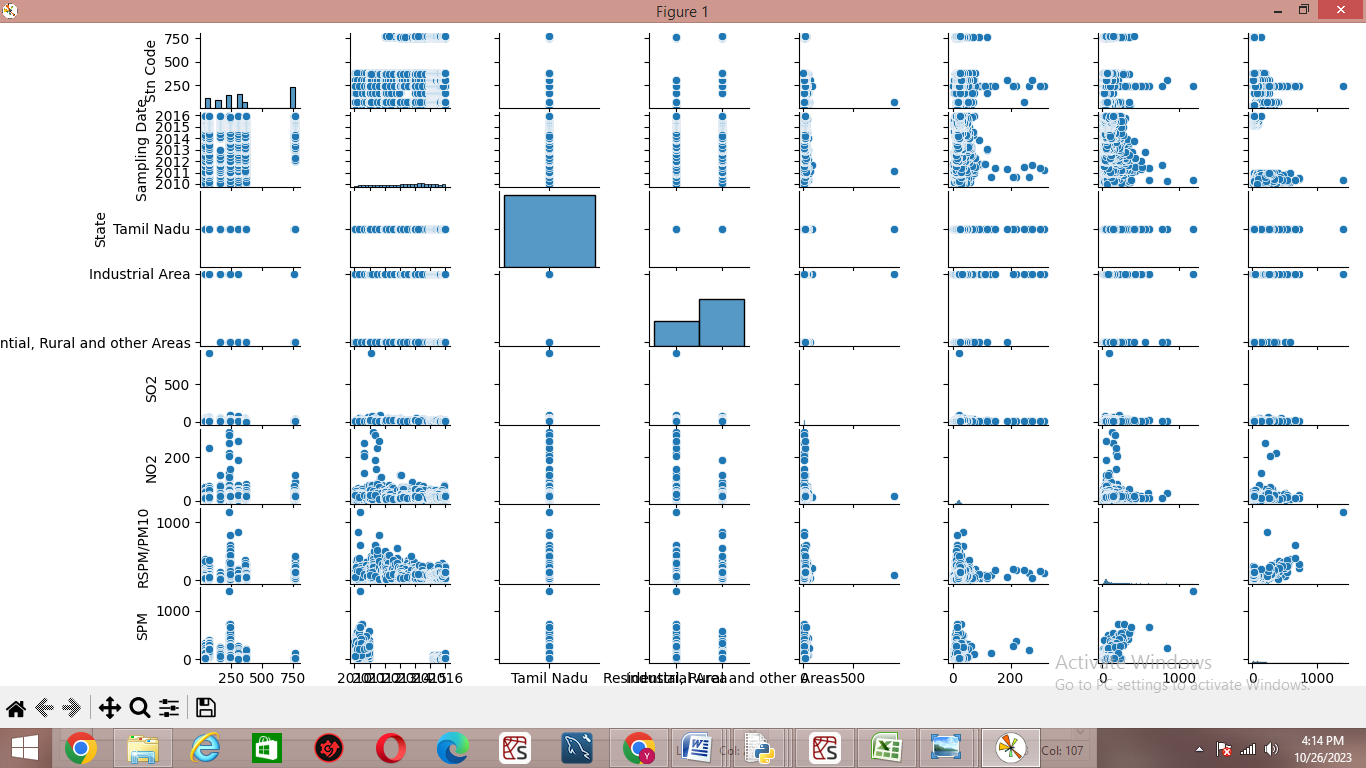
**Output:**



**Multi values chart:**

|  |
| --- |
| import pandas as pd  import matplotlib.pyplot as plt  import seaborn as sns  sns.pairplot(df,vars=['Stn Code','Sampling Date','State','Type of Location','SO2','NO2','RSPM/PM10','SPM'])  plt.show() |

**OUTPUT:**



**Train and testing the Data:**

* The modelling data is divided into training and testing data.
* The simplest way to split the modelling dataset into training and testing sets is to assign 2/3 data points to the former and the remaining one-third to the latter.
* The dataframe gets divided into X\_train,X\_test , y\_train and y\_test.
* X\_train and y\_train sets are used for training and fitting the model.
* The X\_test and y\_test sets are used for testing the model if it's predicting the right outputs/labels.

CODING :

|  |
| --- |
| #separate dependent and independent variable  X=df.drop(columns='State',axis=1)  y=df['State']  print(y)  #splitting dataset into training and test  X\_train,X\_test,y\_train,y\_test=train\_test\_split(X,y,test\_size=0.20,random\_state=0)  #shape  df.shape  #shape of train and test data  print(X\_train.shape)  print(X\_test.shape) |

OUTPUT :

|  |
| --- |
| 0 Tamil Nadu  1 Tamil Nadu  2 Tamil Nadu  3 Tamil Nadu  4 Tamil Nadu  ...  12347 Tamil Nadu  12348 Tamil Nadu  12349 Tamil Nadu  12350 Tamil Nadu  Name: State, Length: 12351, dtype: object  (12351, 11)  (9880, 10)  (2471, 10) |

**Accuracy and Metrics :**

* Accuracy can also be defined as the ratio of the number of correctly classified cases to the total of cases under evaluation.
* The best value of accuracy is 1 and the worst value is 0.
* Gaussian Naive Bayes (GNB) is a classification technique used in Machine Learning (ML) based on the probabilistic approach and Gaussian distribution.
* Gaussian Naïve Bayes is the extension of naïve Bayes.
* While other functions are used to estimate data distribution, Gaussian or normal distribution is the simplest to implement as you

will need to calculate the mean and standard deviation for the training data.

CODING:

|  |
| --- |
| import pandas as pd  from sklearn.model\_selection import train\_test\_split  from sklearn.naive\_bayes import GaussianNB  from sklearn import metrics  df=pd.read\_excel("C:\\Users\\DELL2\\Downloads\\Air\_Quality.xlsx")  X=df.drop(columns='State',axis=1)  y=df['State']  X\_train,X\_test,y\_train,y\_test=train\_test\_split(X,y,test\_size=0.20,random\_state=0)  print(X)  gnb=GaussianNB()  gnb.fit(X\_train,y\_train)  y\_pred=gnb.predict(X\_test)  print("accuracy is :",metrics.accuracy\_score(y\_test,y\_pred)\*100) |

OUTPUT:

