INT234
PREDICTIVE ANALYTICS

PROJECT REPORT
(Project Semester October-December 2025)

Bachelor of Technology

(Computer Science and Engineering)

Submitted to

LOVELY PROFESSIONAL UNIVERSITY

PHAGWARA, PUNJAB

Air Quality

Submitted by
Prema Sai Kimmi
Registration No – 12306000
B.TECH CSE
Section – K23CB
Course Code – INT234

Under the Guidance of
Dr. Ravindra Singh Yadav

DECLARATION

I, Prema Sai Kimmi, student of B.Tech Computer Science Engineering under CSE/IT Discipline at, Lovely Professional University, Punjab, hereby declare that all the information furnished in this project report is based on my own intensive work and is genuine.

Date: 19/12/2025
Signature: Prema Sai kimmi
Registration No. 12306233
Name of the student: Prema Sai Kimmi

CERTIFICATE

This is to certify that Prema Sai Kimmi bearing Registration no. 12306000 has completed INT234 project titled, “Air Quality” under my guidance and supervision. To the best of my knowledge, the present work is the result of his/her original development, effort and study.

Signature and Name of the Supervisor
Designation of the Supervisor
School of Computer Science and Engineering
Lovely Professional University Phagwara, Punjab.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my project guide, Dr. Ravindra Singh Yadav, for their valuable guidance and support throughout this project, “Analysis of Bank Churn”. I am thankful to the Department of Computer Science and Engineering, Lovely Professional University, for providing the necessary resources and environment. I also acknowledge the Government of India for making the dataset publicly available, enabling this research

Name: Prema Sai Kimmi

Registration No : 12306000

TABLE OF CONTENTS

1. Introduction

2. Source of Dataset

3. Exploratory Data Analysis (EDA) Process

4. Analysis of Data

5. Conclusion

6. Scope for Future Enhancement

7. Github / Linkedin Link

8. References

1. Introduction

Air pollution is one of the most critical environmental challenges affecting public health, climate, and overall quality of life. Rapid urbanization, industrial activities, increased vehicular emissions, and population growth have significantly contributed to the deterioration of air quality in many regions. Continuous monitoring and analysis of air pollution data are essential for understanding pollution patterns, identifying high-risk areas, and supporting effective environmental policy decisions.

The **Air Quality dataset** used in this project contains detailed information on pollution measurements collected across different geographical locations and time periods. The dataset includes pollution values for various air quality indicators along with associated metadata such as location type, geographical area, time period, and measurement dates. These attributes make the dataset suitable for both statistical analysis and machine learning applications. The primary goal of this project is to perform **data cleaning and preprocessing**, conduct **exploratory data analysis (EDA)**, and apply **machine learning techniques** to analyze and predict air pollution levels. Regression models are used to predict pollution values based on temporal and geographical features, while classification models categorize pollution levels into meaningful groups such as *Low*, *Medium*, and *High*. Additionally, clustering techniques are applied to identify hidden patterns and similarities in pollution trends across different years and locations.

By combining data visualization, supervised learning, and unsupervised learning approaches, this project provides valuable insights into air quality behaviour and demonstrates how data science techniques can be effectively used to support environmental analysis and decision-making.

2. SOURCE OF DATASET

The dataset used for this analysis is titled **Air_Quality.csv**, which contains structured data related to air pollution measurements across different geographical locations and time periods. The dataset includes detailed records such as pollution indicator names, pollution values, geographical area types, location names, time periods, start dates, and additional descriptive messages. This dataset provides a comprehensive view of air quality conditions over multiple years and regions.

Possible Origins of the Dataset:

- **Government Environmental Agencies:** The dataset may originate from environmental monitoring organizations or government bodies such as air quality boards.
- **Air Quality Monitoring Systems:** Data could be collected from automated air quality sensors installed in urban and suburban areas that continuously record pollution levels for various pollutants.

- **Dataset Format:**
  - File Type: CSV (Comma-Separated Values)
  - Accessibility: Easy to load using spreadsheet software (Excel) or data analysis tools (like Python's pandas library).
  - Content: Each row represents a recorded air quality measurement for a specific pollutant, location, and time period.

Importance of Dataset Source:

Understanding the source of the dataset is essential as it influences:

- **Possible biases or missing values due to sensor limitations or reporting gaps**
- **The reliability and accuracy of pollution measurements**

Dataset link: https://catalog.data.gov/dataset/air-quality

3. EXPLORATORY DATA ANALYSIS(EDA) PROCESS

1. Data Loading & Initial Inspection

• The Air_Quality.csv dataset is imported using Python's pandas library.

• The first few rows of the dataset are displayed using the head() function to understand the structure and content of the data.

• The dataset's dimensions (number of rows and columns) are examined to understand its size.

• Data types of each column are checked using the info() function to identify numerical, categorical, and date fields.

---

2. Data Cleaning

• Missing Values:

Missing values in textual columns such as *Message* are filled with a default value, while missing numerical values in *Data Value* are handled using median imputation to avoid bias.

• Date Conversion:

The *Start_Date* column is converted into a proper datetime format, and new features such as Year and Month are extracted for temporal analysis.

• Categorical Standardization:

Seasonal information is extracted from the *Time Period* column and categorized into Winter, Summer, Spring, Fall, and Annual.

• Removal of Invalid Records:

Rows with critical missing values (Year, Month, or Data Value) are removed to ensure data consistency.

This cleaning process ensures the dataset is reliable and suitable for analysis and machine learning models.

---

3. Descriptive Statistics

• Statistical summaries are generated using the describe() function to understand:

- Mean pollution level

- Minimum and maximum pollution values
- Standard deviation indicating variability

• These statistics provide insights into the overall pollution distribution and highlight potential extreme values.

---

4. Bivariate / Multivariate Analysis

• Relationships between multiple variables are explored:

- Relationship between pollution values and time (Year, Month)
- Comparison of pollution levels across different geographical locations
- Seasonal variation in pollution concentration

---

5. Time-Based Analysis

• Since the dataset contains time information:

- Pollution trends are analyzed across years and months.
- Seasonal patterns are identified using extracted season labels.
- Long-term pollution behavior is examined using year-wise trends.

This analysis helps identify periods of higher pollution concentration.

Conclusion of EDA

The Exploratory Data Analysis reveals meaningful patterns in air pollution levels across different regions and time periods. The insights obtained from EDA provide a strong foundation for predictive modeling, classification of pollution levels, and clustering analysis performed in later stages of the project.

4. ANALYSIS ON DATASET

Objective 1: Data Cleaning & Preprocessing

Introduction:

Data cleaning and preprocessing is a crucial initial step in any data analytics or machine learning project, as the quality of input data directly impacts the accuracy and reliability of results. In this project, the air quality dataset contains a mixture of numerical, categorical, and date-based attributes, along with missing and inconsistent values. The objective of this phase is to prepare a clean, structured, and machine-learning-ready dataset by handling missing values, correcting data types, extracting useful time-based features, and standardizing categorical information.

General Description:

The preprocessing process begins with handling missing values by replacing null entries in textual fields with meaningful defaults and imputing missing numerical values using the median to reduce the influence of outliers. Date values are converted into a standard datetime format, from which new features such as Year and Month are extracted to enable temporal analysis. Seasonal information is derived from the *Time Period* column and categorized into standard seasons to simplify analysis. Finally, records containing critical missing values are removed to ensure consistency and data integrity. This comprehensive preprocessing ensures that the dataset is accurate, complete, and suitable for exploratory analysis and machine learning models.

Formulas:

Missing Value Handling (Categorical Column)

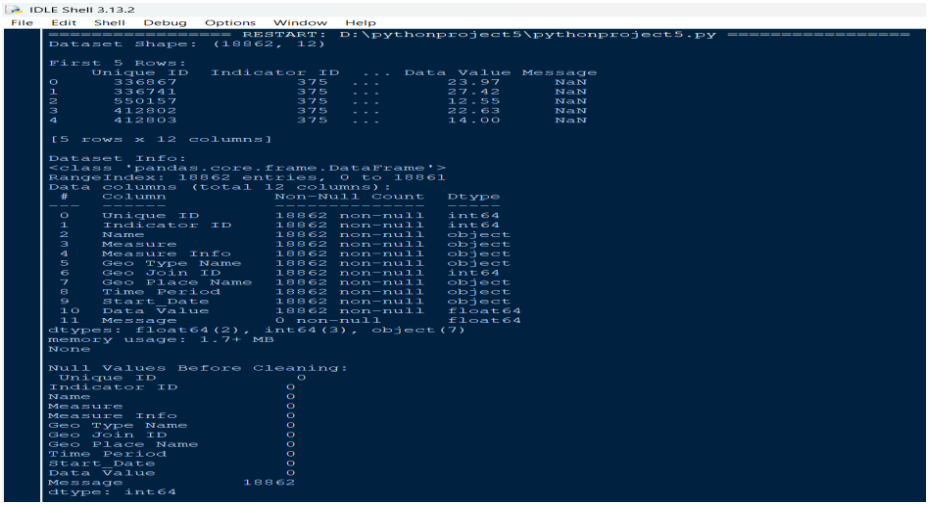
Datetime Conversion

Year Extraction

Median (for missing value imputation)

Missing Value Imputation (Numerical Column)

Visual:



Objective 2: Analyze pollution trends by location, season, and year.(EDA)

i. Introduction

Analyzing pollution trends by location, season, and year is essential for understanding how air quality changes over time and across different geographic regions. This analysis helps identify high-pollution areas, seasonal variations, and long-term environmental patterns.

ii. General Description

The analysis was performed using the cleaned air quality dataset by grouping pollution values based on geographic attributes, seasonal categories, and extracted year information. Features such as location name, season, and year were derived during preprocessing, and pollution levels were summarized and visualized to observe temporal and spatial trends.

iii. Requirements

• Feature extraction (Year, Month, Season)

Data cleaning and preprocessing

• Grouping and aggregation operations

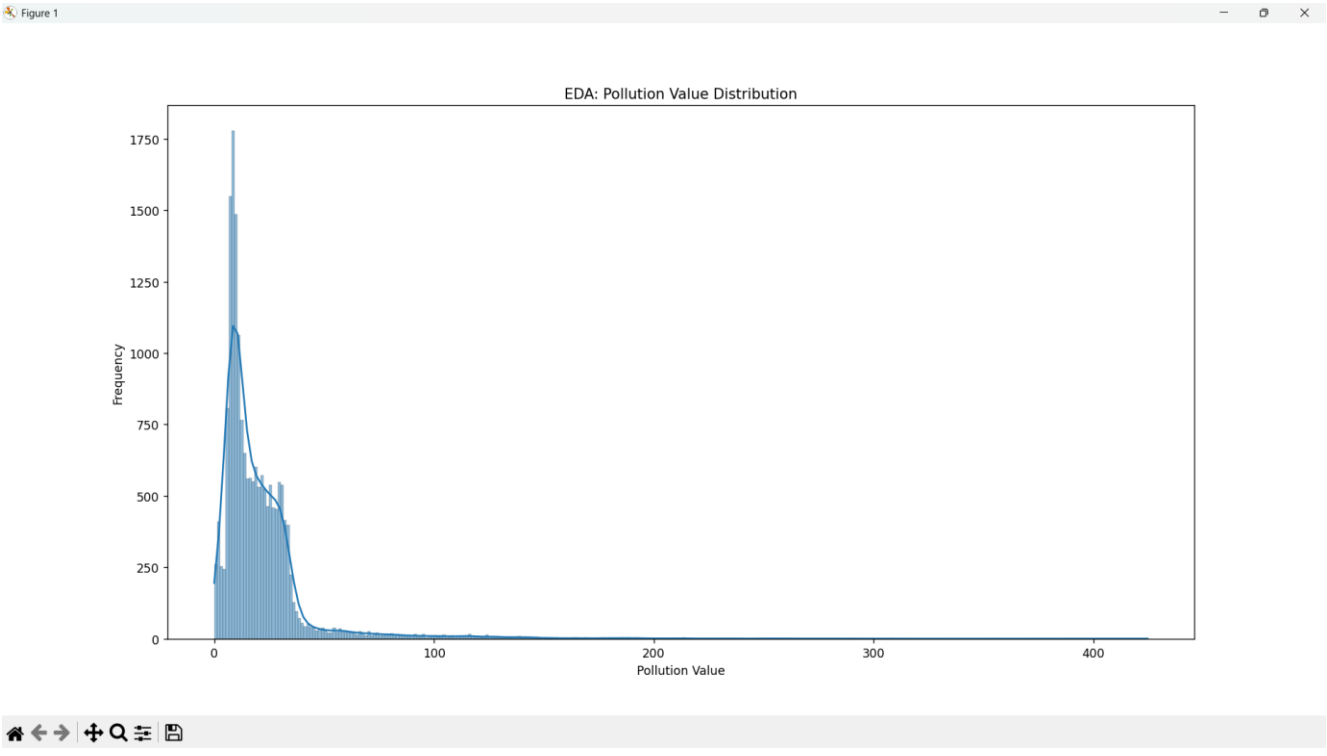
• Descriptive statistics

• Visualization tools (histograms, scatter plots, trend plots)

iv. Results

The analysis revealed clear variations in pollution levels across different locations, seasons, and years. Seasonal patterns showed noticeable differences in pollution concentration, while year-wise analysis highlighted trends and fluctuations over time.

v. Visualization



Objective 3: Regression Model for Pollution Prediction

i. Introduction

Build regression models Random Forest to predict 'Data Value' (pollution level) based on factors like season, location, date, and year. make visual graph

ii. General Description

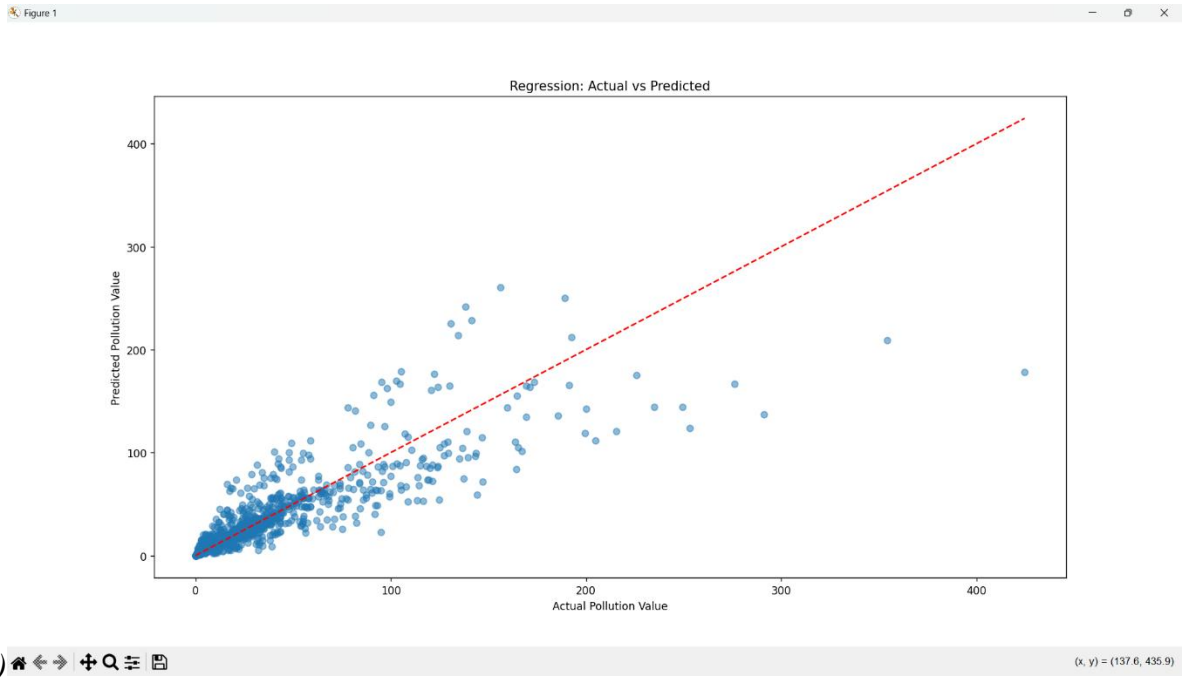
A Random Forest Regression model was developed to predict pollution levels using features such as location, season, year, and month. Categorical variables were transformed using One-Hot Encoding, and the dataset was split into training and testing sets. Model performance was evaluated using standard regression metrics.

iii. Requirements:

Train-test split
One-Hot Encoding
Random Forest Regressor
Evaluation metrics (MAE, RMSE, R^2 score)
Visualization of actual vs predicted values

iv.Results:

The regression model successfully predicted pollution levels with reasonable accuracy. Performance metrics such as MAE, RMSE, and R^2 score indicated the model's effectiveness. The scatter plot of actual versus predicted values showed a strong correlation, confirming the model's predictive capability. **v. Visualization**



Time)

Objective 4: Classification of Pollution Levels

i. Introduction

Classifying pollution levels helps in categorizing air quality into meaningful groups such as low, medium, and high. This classification simplifies interpretation, supports environmental monitoring, and aids decision-makers in identifying areas that require immediate attention.

ii. General Description

In this project, pollution values were divided into three categories—**Low**, **Medium**, and **High**—using quantile-based binning (pd.qcut). A **K-Nearest Neighbors (KNN)** classification model was then applied to predict these pollution categories based on location, season, and time-related features. Categorical variables were encoded using One-Hot Encoding, and the dataset was split into training and testing sets to evaluate classification performance.

iii. Requirement

KNeighborsClassifier

OneHotEncoder

Train-test split

Accuracy score

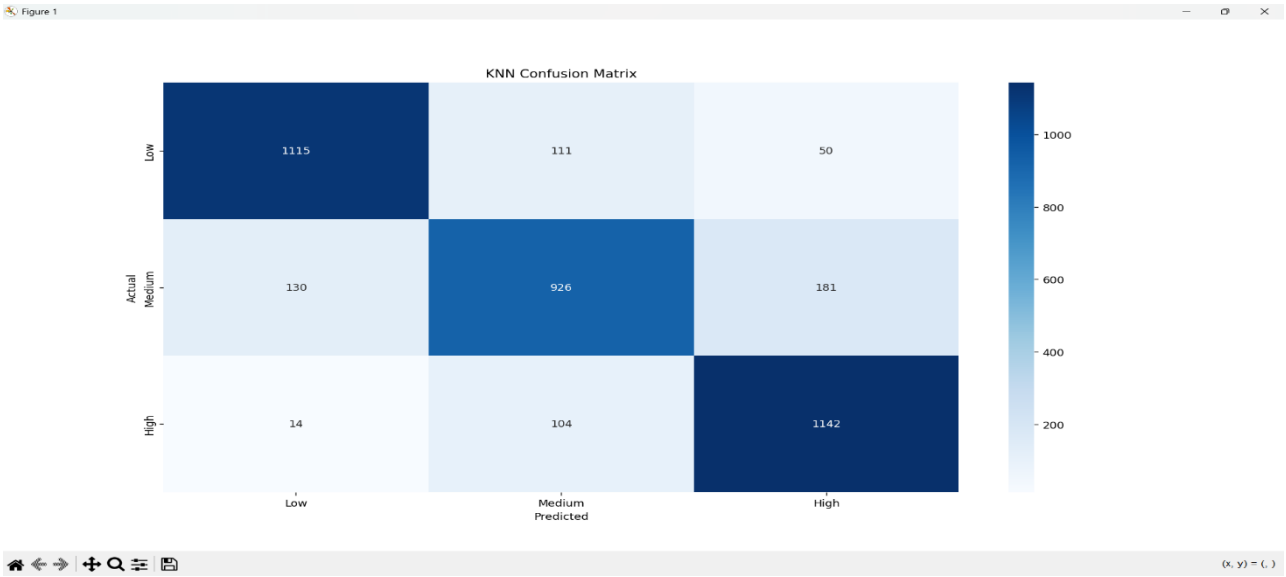
Confusion matrix

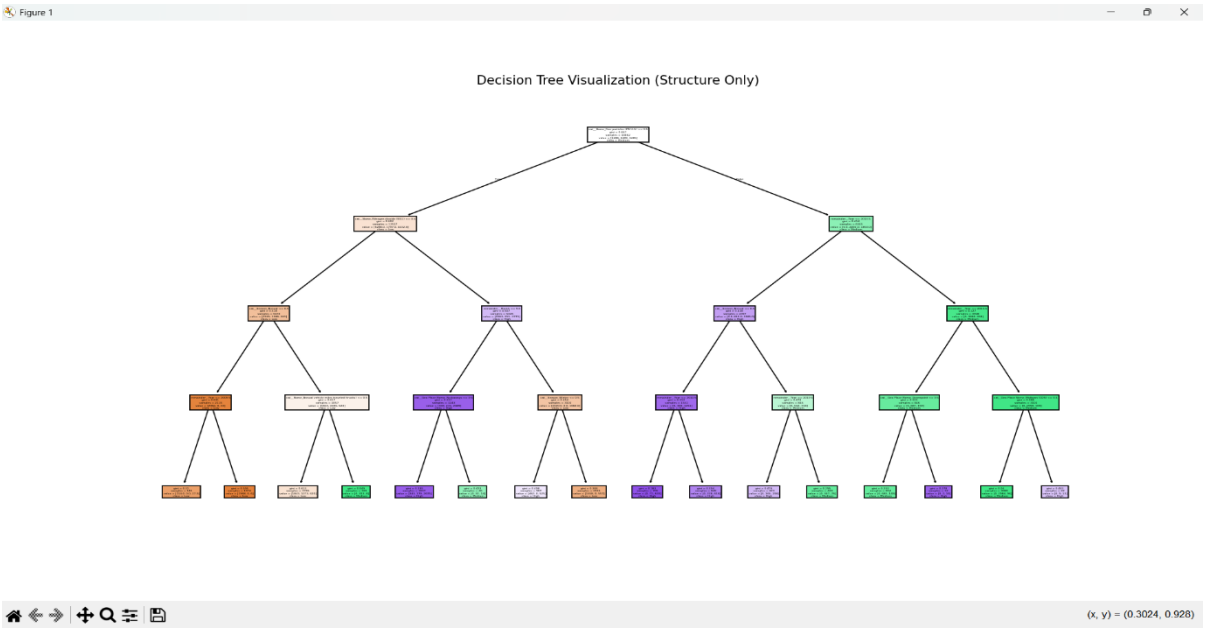
Heatmap visualization

iv. Results

The KNN classifier successfully categorized pollution levels into low, medium, and high classes. The accuracy score demonstrated the model's effectiveness, while the confusion matrix and heatmap visually illustrated correct and incorrect classifications, providing clear insight into model performance.

v. Visualization





Objective 5: Unsupervised Learning (Clustering for Pattern Detection)

i. Introduction

Unsupervised learning techniques are used to discover hidden patterns in data without predefined labels. Clustering helps in grouping similar data points, enabling better understanding of pollution behavior across different time periods and pollution levels.

ii. General Description

In this analysis, **K-Means clustering** was applied to the *Data Value* (pollution concentration) and *Year* features to identify natural groupings within the air quality data. The algorithm grouped observations into three clusters based on similarity, allowing the detection of pollution trends and patterns over time without prior classification.

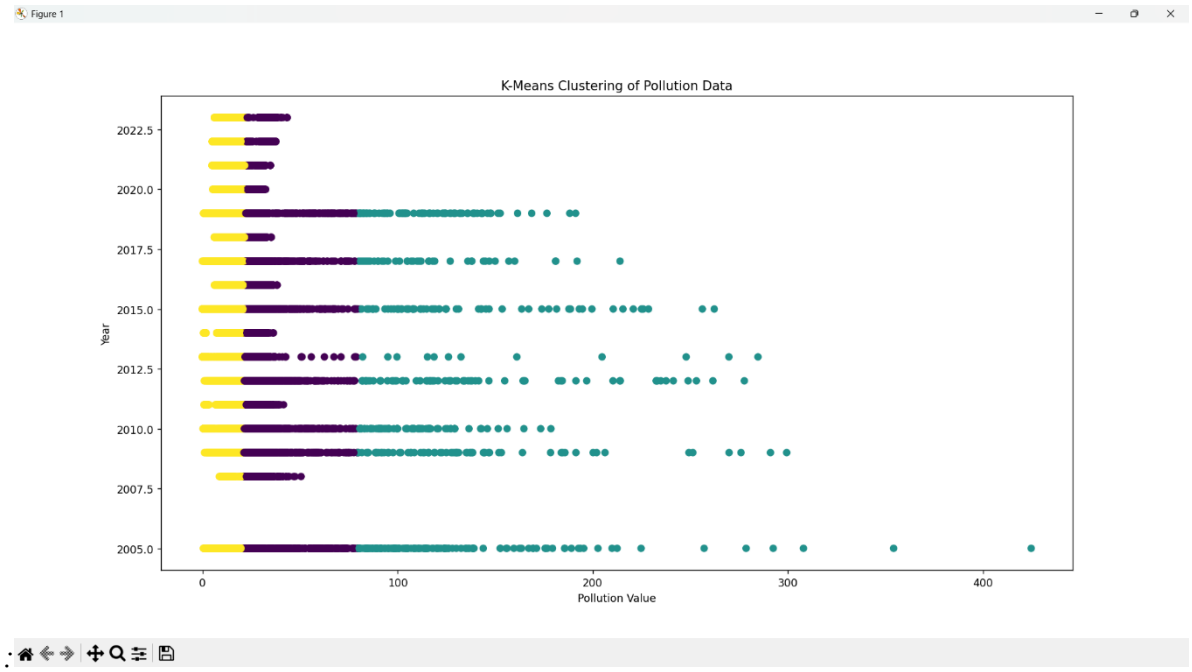
iii. Requirement

KMeans algorithm
Feature selection (Data Value, Year)
.fit_predict() method
Cluster center extraction

iv. Results

The script successfully grouped pollution data into three distinct clusters. The cluster centers revealed differences in pollution intensity and time distribution, while the scatter plot visually demonstrated clear separation between clusters, providing meaningful insights into underlying pollution patterns.

v. Visualization



Objective 7: Correlation and Pairwise Relationships

i. Introduction

Studying correlations helps identify relationships between numerical variables. Strong correlations can guide predictive modeling and strategic decisions.

ii. General Description

A correlation matrix was generated for all numeric columns using .corr(), and visualized with a heatmap. A pairplot was also created using Seaborn to show pairwise relationships.

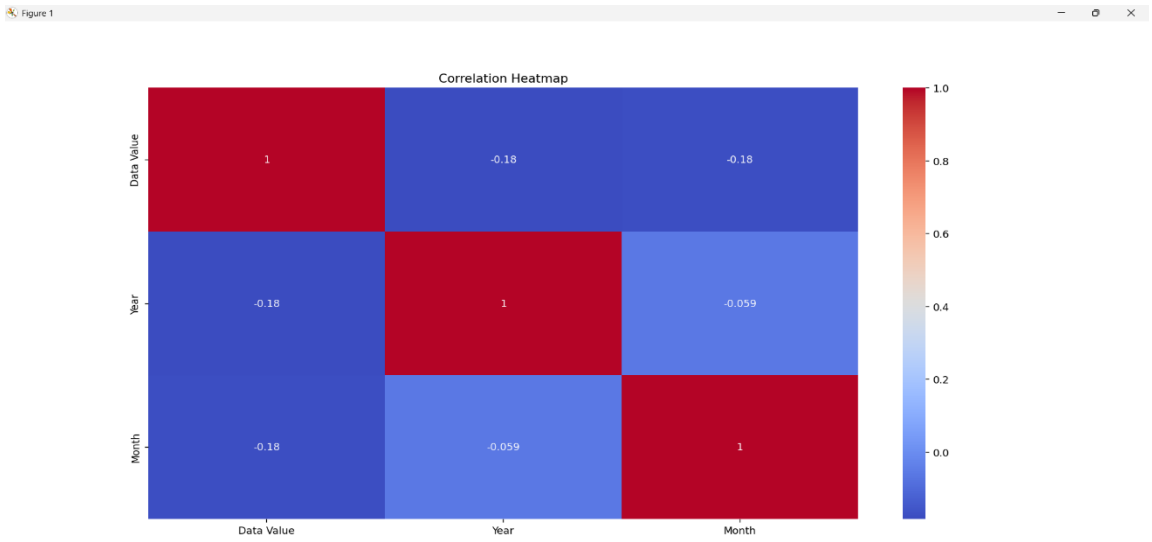
iii. Requirements

.corr() function
sns.heatmap() and sns.pairplot()

iv. Results

The script generated a color-coded heatmap showing the correlation coefficients between all numerical variables. This visualization helps in identifying strongly correlated features, guiding feature selection and understanding interdependencies in the dataset.

v. Visualization



IDLE Shell 3.13.2

File Edit Shell Debug Options Window Help

```
dtype: int64

Null Values After Cleaning:
 Unique ID          0
Indicator ID        0
Name                0
Measure             0
Measure Info        0
Geo Type Name       0
Geo Join ID         0
Geo Place Name      0
Time Period         0
Start_Date          0
Data Value          0
Message             0
Year                0
Month               0
Season              0
dtype: int64
Final Dataset Shape: (18862, 15)

EDA Summary of Pollution Values:
count    18862.000000
mean        21.051580
std         23.564920
min          0.000000
25%          8.742004
50%         14.790000
75%         26.267500
max        424.700000
Name: Data Value, dtype: float64

Regression Results:
MAE : 3.712139828295516
RMSE: 11.976038395640353
R2  : 0.776304413118212

KNN Classification Results:
Accuracy: 0.8436257619931089
Confusion Matrix:
 [[1115  111   50]
 [ 130  926  181]
 [  14  104 1142]]

Clustering Results:
Cluster Centers:
 [[  32.13624141 2013.76848928]
 [ 127.54838729 2011.2919708 ]
 [  10.88189185 2015.33764333]]
>>>
```

5.Conclusion

This Air Quality Data Analysis project presents a complete data science workflow involving data cleaning, exploratory data analysis, machine learning, and visualization to understand pollution patterns. After preprocessing the dataset and extracting temporal and seasonal features, EDA revealed clear variations in pollution levels across different years and seasons. The Random Forest regression model effectively predicted pollution values using geographic and time-based features, while the KNN classification model successfully categorized pollution levels into Low, Medium, and High classes with good accuracy. The Decision Tree visualization improved interpretability of feature importance, and K-Means clustering identified natural groupings and trends in pollution behavior over time. Overall, the project demonstrates how data analytics and machine learning techniques can be applied to air quality data to generate meaningful insights and support data-driven environmental analysis.

6. Future Scope

In the future, this project can be extended by incorporating real-time air quality data and additional environmental factors such as temperature, humidity, wind speed, and traffic or industrial activity to improve prediction accuracy. Advanced machine learning and deep learning models like XGBoost, LSTM, or time-series forecasting techniques can be applied to capture long-term and seasonal trends more effectively. The analysis can also be expanded to include health impact assessment and location-based risk prediction. Integrating the model with interactive dashboards or alert systems would further enhance its practical usefulness for policymakers and the general public.

7. Github / Linkedin Link

Github: https://github.com/PremSai45

Linkedin: https://www.linkedin.com/posts/prema-sai_datascience-machinelearning-python-activity-7407805240458194946-0mOa?utm_source=social_share_send&utm_medium=android_app&rcm=ACoAAEcgeHQBxGnHuXFDPI9H0EikSzmR_Iu_qQc&utm_campaign=whatsapp

8.References

- Dataset Source: data.gov (link: https://catalog.data.gov/dataset/air-quality
- Pandas Documentation: https://pandas.pydata.org/docs/
- Matplotlib Documentation: https://matplotlib.org/stable/contents.html
- Seaborn Documentation: https://seaborn.pydata.org/
- NumPy Documentation: https://numpy.org/doc/
- Towards Data Science – Articles on EDA and Data Visualization

SOURCE CODE

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.ensemble import RandomForestRegressor
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.cluster import KMeans
from sklearn.metrics import (mean_absolute_error,mean_squared_error,
                             r2_score,confusion_matrix,accuracy_score)

df = pd.read_csv(r"D:\5th sem'\pythonproject5\Air_Quality.csv")

print("Dataset Shape:", df.shape)
print("\nFirst 5 Rows:\n", df.head())
print("\nDataset Info:")
print(df.info())
print("\nNull Values Before Cleaning:\n", df.isnull().sum())

# OBJECTIVE 1: DATA CLEANING & PREPROCESSING

df['Message'] = df['Message'].fillna("No Message")
df['Data Value'] = df['Data Value'].fillna(df['Data Value'].median())
df['Start_Date'] = pd.to_datetime(df['Start_Date'], errors='coerce')
df['Year'] = df['Start_Date'].dt.year
df['Month'] = df['Start_Date'].dt.month
def get_season(x):
    x = str(x).lower()
    if 'winter' in x: return 'Winter'
    if 'summer' in x: return 'Summer'
    if 'spring' in x: return 'Spring'
    if 'fall' in x: return 'Fall'
    return 'Annual'

df['Season'] = df['Time Period'].apply(get_season)
df = df.dropna(subset=['Year', 'Month', 'Data Value'])

print("\nNull Values After Cleaning:\n", df.isnull().sum())
```

```
print("Final Dataset Shape:", df.shape)

# OBJECTIVE 2: EXPLORATORY DATA ANALYSIS (EDA)

print("\nEDA Summary of Pollution Values:")
print(df['Data Value'].describe())

plt.figure(figsize=(7,4))
sns.histplot(df['Data Value'], kde=True)
plt.title("EDA: Pollution Value Distribution")
plt.xlabel("Pollution Value")
plt.ylabel("Frequency")
plt.show()
X = df[['Name', 'Geo Type Name', 'Geo Place Name',
        'Season', 'Year', 'Month']]
y = df['Data Value']
ct = ColumnTransformer([
    ('cat', OneHotEncoder(handle_unknown='ignore'),
     ['Name', 'Geo Type Name', 'Geo Place Name', 'Season'])], remainder='passthrough')

# OBJECTIVE 3: REGRESSION (RANDOM FOREST)

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)
reg_model = Pipeline([
    ('preprocess', ct),
    ('model', RandomForestRegressor(n_estimators=100, random_state=42))
])
reg_model.fit(X_train, y_train)
y_pred = reg_model.predict(X_test)

print("\nRegression Results:")
print("MAE :", mean_absolute_error(y_test, y_pred))
print("RMSE:", np.sqrt(mean_squared_error(y_test, y_pred)))
print("R2  :", r2_score(y_test, y_pred))

plt.figure(figsize=(6,5))
plt.scatter(y_test, y_pred, alpha=0.5)
plt.plot([y_test.min(), y_test.max()],
         [y_test.min(), y_test.max()], 'r--')
plt.xlabel("Actual Pollution Value")
plt.ylabel("Predicted Pollution Value")
plt.title("Regression: Actual vs Predicted")
plt.show()
```

```
# OBJECTIVE 4: CLASSIFICATION (KNN ONLY)

df['Pollution_Level'] = pd.qcut(
    df['Data Value'], 3, labels=['Low', 'Medium', 'High']
)
y_clf = df['Pollution_Level']

X_train_c, X_test_c, y_train_c, y_test_c = train_test_split(
    X, y_clf, test_size=0.2, random_state=42
)

knn_model = Pipeline([
    ('preprocess', ct),
    ('classifier', KNeighborsClassifier(n_neighbors=5))
])

knn_model.fit(X_train_c, y_train_c)
y_pred_knn = knn_model.predict(X_test_c)

acc_knn = accuracy_score(y_test_c, y_pred_knn)
cm_knn = confusion_matrix(
    y_test_c, y_pred_knn,
    labels=['Low', 'Medium', 'High']
)

print("\nKNN Classification Results:")
print("Accuracy:", acc_knn)
print("Confusion Matrix:\n", cm_knn)

plt.figure(figsize=(6,5))
sns.heatmap(cm_knn, annot=True, fmt='d', cmap='Blues',
            xticklabels=['Low', 'Medium', 'High'],
            yticklabels=['Low', 'Medium', 'High'])
plt.title("KNN Confusion Matrix")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.show()

# DECISION TREE (ONLY VISUALIZATION – NO CLASSIFICATION)

dt_visual = Pipeline([
    ('preprocess', ct),
    ('classifier', DecisionTreeClassifier(
        max_depth=4, random_state=42))
```

```
])

dt_visual.fit(X, y_clf)

dt_clf = dt_visual.named_steps['classifier']
feature_names = dt_visual.named_steps[
   'preprocess'].get_feature_names_out()

plt.figure(figsize=(22,10))
plot_tree(
   dt_clf,
   feature_names=feature_names,
   class_names=['Low', 'Medium', 'High'],
   filled=True
)
plt.title("Decision Tree Visualization (Structure Only)")
plt.show()
#objective 5: corelation
corr = df[['Data Value','Year','Month']].corr()

sns.heatmap(corr, annot=True, cmap='coolwarm')
plt.title("Correlation Heatmap")
plt.show()

# OBJECTIVE 6: CLUSTERING (K-MEANS)

cluster_data = df[['Data Value', 'Year']]

kmeans = KMeans(n_clusters=3, random_state=42, n_init=10)
df['Cluster'] = kmeans.fit_predict(cluster_data)

print("\nClustering Results:")
print("Cluster Centers:\n", kmeans.cluster_centers_)

plt.figure(figsize=(7,5))
plt.scatter(cluster_data['Data Value'],
          cluster_data['Year'],
          c=df['Cluster'])
plt.xlabel("Pollution Value")
plt.ylabel("Year")
plt.title("K-Means Clustering of Pollution Data")
plt.show()
```