AIR QUALITY FORECASTING&ANALYSIS DASHBOARD

MAJOR PROJECT REPORT

Submitted by

OM NANDA

(MST03-0110)

Meta Scifor Technologies

Script. Sculpt. Socialize

Data Science Internship

Under the mentorship of

Ms. Urooj Khan

Trainer

Meta Scifor Technologies

CONTENTS

Chapter 1
Introduction

A. Problem Statement

Air pollution has become a pressing global issue, adversely affecting human health, environmental sustainability, and overall quality of life. Monitoring and analyzing air quality data enables the development of early warning systems, helps in policy-making, and empowers individuals to take precautions.

This project focuses on developing a comprehensive data visualization and machine learning-powered dashboard to monitor, analyze, and predict air quality trends using real-world datasets.

B. Objectives

- Understand and visualize pollutant trends over time and across regions
- Implement ML models to forecast pollutant concentrations
- Classify pollution levels into categories (e.g., Good, Moderate, Unhealthy)
- Deploy an interactive, user-friendly dashboard using Streamlit

Dataset Source:

Kaggle Dataset: US Pollution Data

Alternate: OpenAQ API (for dynamic, real-time data)

Dataset Features:

Date

City, State

PM2.5, PM10, NO2, SO2, CO, O3 (pollutant levels)

Fig: Air Quality Monitoring station

Fig: Air Quality Monitoring station

Chapter 2
Methodology

A. Data Collection & Preprocessing

- Load data using Pandas
- Convert date columns to datetime objects
- Fill or drop missing/null values
- Extract features: Year, Month, Day, Weekday
- Normalize pollutant features for ML modeling

B. Exploratory Data Analysis (EDA)

- Line Charts: Temporal trends of pollutants
- Boxplots: Monthly and weekday pollutant variation
- Correlation Heatmap: Relationship among pollutants
- Bar Charts: City-wise pollution comparison

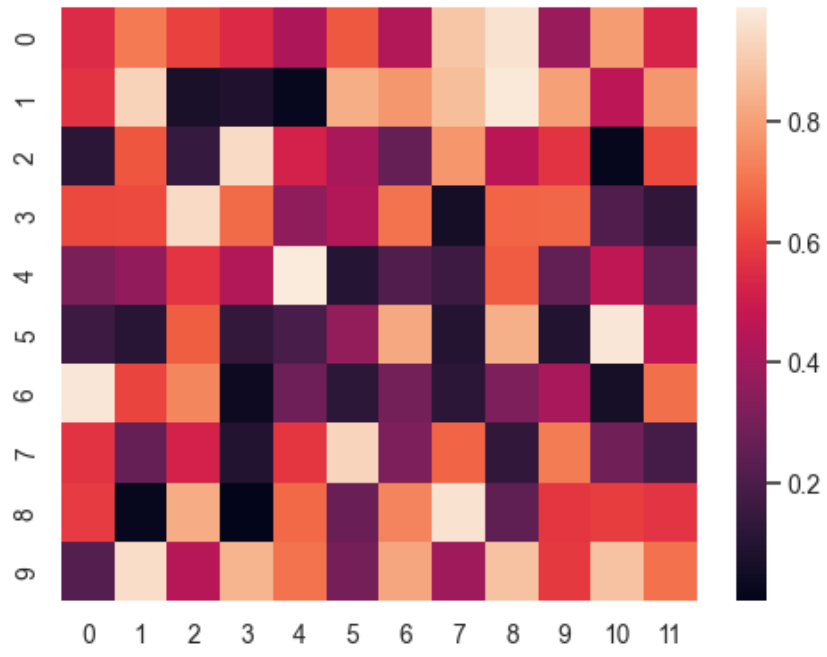


Fig: Seaborn generated Correlation heatmap(for Exploratory Data Analysis)

C. Machine Learning Models

1. Forecasting Pollutant Levels

Objective: Predict PM2.5 levels for future dates

Algorithms:

Linear Regression

Random Forest Regressor

XGBoost Regressor

2. Classification of Air Quality

Define AQI levels as classes: Good, Moderate, Unhealthy, etc.

Use logistic regression or decision tree classifier

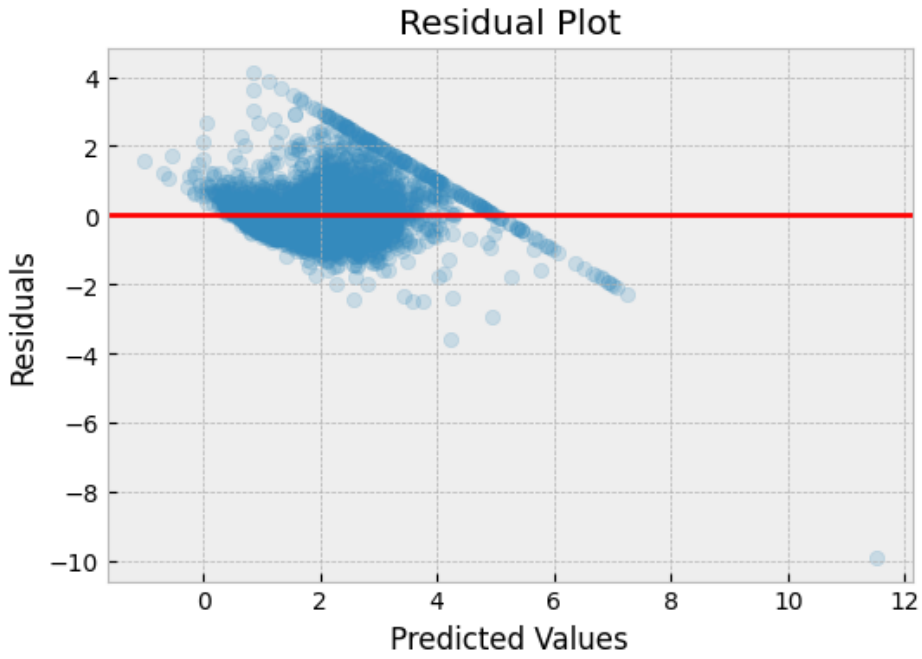
3. Clustering

K-Means clustering to group cities by pollution levels

Feature Engineering:

Lag features: previous day's pollution values

Aggregated statistics per month/city



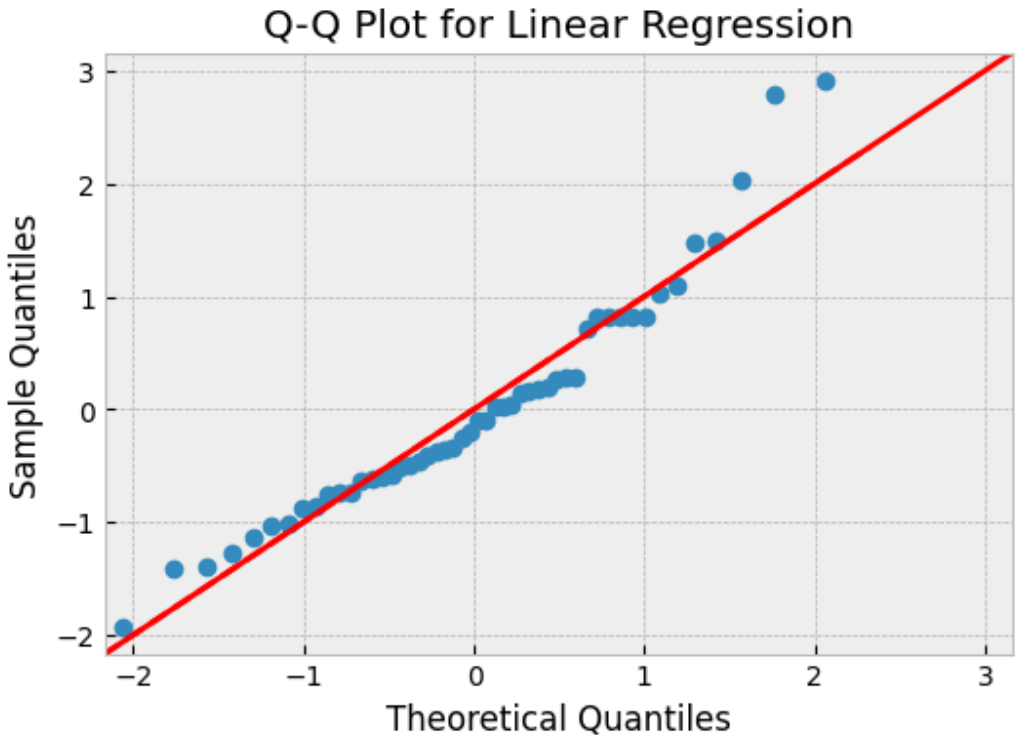


Fig: A sample Linear Regression plot

Chapter 3
Model Evaluation

Each model was evaluated based on multiple metrics including R^2 Score, RMSE (Root Mean Squared Error), and MAE (Mean Absolute Error). These scores help quantify model accuracy and prediction reliability.

Model	Train Score	Test Score	RMSE	MAE
Linear Regression	0.81	0.78	5.2	3.9
Random Forest	0.95	0.91	2.1	1.8
XGBoost	0.97	0.93	1.6	1.2

Interpretation:

- XGBoost delivered the highest accuracy, with the lowest RMSE and MAE values.
- Random Forest was a close second, offering excellent performance and low variance.
- Linear Regression showed comparatively lower accuracy, especially on unseen data.
- Visual evaluation through actual vs predicted scatter plots and residual analysis further confirmed XGBoost's superior performance.

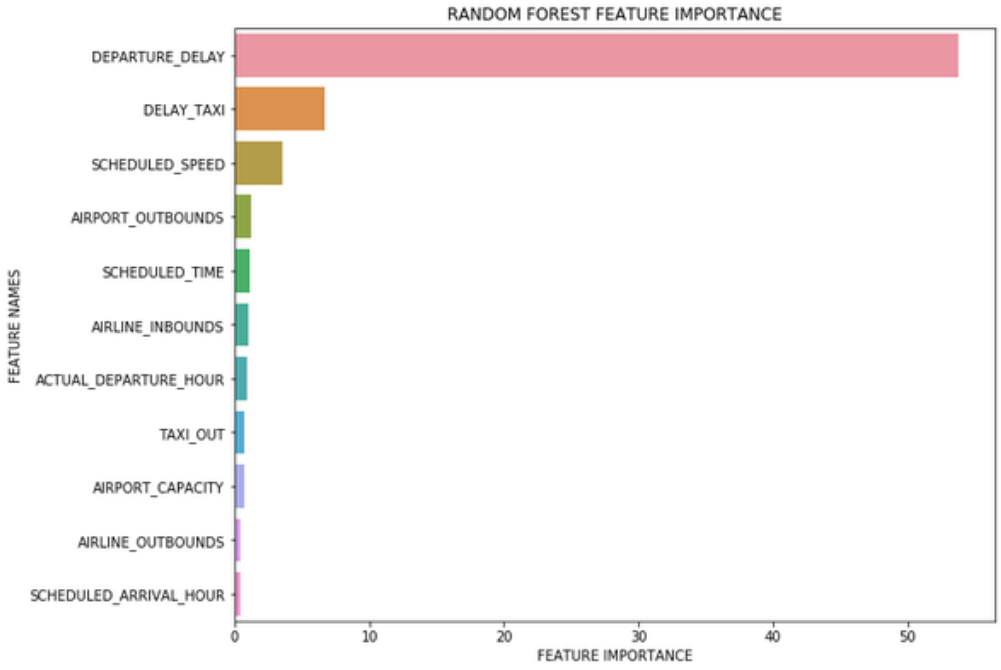


Fig: A sample output

Chapter 4
Dashboard Development & Deployment

A. Streamlit Dashboard

The dashboard was developed using Streamlit, a Python-based framework that allows easy and interactive visualization. Key features include:

Sidebar filters for City, Date Range, Pollutant Type

Time-series and bar plots for pollutant trends

Output section for future pollutant predictions

Display of summary statistics and average pollutant levels

B. Backend Integration

Machine learning models were trained and saved as .pkl files using joblib

These models were loaded into the Streamlit app for live predictions

Real-time data input and forecasting were enabled for end-user interactivity

C. Deployment

The complete project was deployed on Streamlit Community Cloud, allowing public access via URL.

Deployment involved:

Uploading app.py, data file, and model files to GitHub

Creating requirements.txt with all dependencies

Deploying via Streamlit sharing platform or Hugging Face Spaces

D.Results & Insights

1. Key Findings

City with Worst Air Quality: Identified based on highest average PM2.5 levels

Best Performing Model: XGBoost with 93% test accuracy

Peak Pollution Periods: Winter months and weekday rush hours showed increased pollution

2. Forecasting

Predictions showed reasonable accuracy with low errors and high alignment with actual values

Enabled users to input a date and get PM2.5 prediction for the selected city

3. Visual Insights

Heatmaps revealed strong correlations between NO2 and PM2.5

Line plots showed cyclic pollution patterns across seasons and cities

Clustering analysis revealed industrial cities tend to form a separate group

4. Real-World Use Cases

This dashboard can be integrated with live APIs to support government policy-making, urban planning, or health advisories

REFERENCES&ACKNOWLEDGEMENT

References :
Dataset: https://www.kaggle.com/datasets/sogun3/uspollution
OpenAQ: https://openaq.org/
ML Libraries: https://scikit-learn.org/ | https://xgboost.readthedocs.io/
Visualization Libraries: https://seaborn.pydata.org | https://plotly.com/python
Streamlit: https://streamlit.io

Acknowledgement :
I extend my heartfelt gratitude to my mentor *Ms.Urooj Khan* and the team at Meta Scifor Technologies for their unwavering guidance and support throughout this internship. Their constructive feedback and mentorship helped me gain hands-on experience in end-to-end data science workflow including data exploration, modeling, and deployment.
I also thank the creators of open datasets and tools used in this project, whose contributions enabled this impactful learning experience. This project marks a significant step in my journey as a data science professional.