

**Project Report**

**On**

**Air Quality Analysis and Forecasting**



Submitted in partial fulfillment for the award of

Post Graduate Diploma in Big Data Analytics (PGDBDA)

From Know IT(Pune)

**Guided by:**

Trupti Joshi Ma’am & Prasad Deshmukh Sir

**Submitted By:**

Himanshu Samal (240343025018)

Pritam Nimbalkar (240343025038)

Ritul Tayade (240343025042)

Shalu Sharma (240343025045)



**CERTIFICATE**

TO WHOMSOEVER IT MAY CONCERN

This is to certify that

Himanshu Samal (240343025018)

Pritam Nimbalkar (240343025038)

Ritul Tayade (240343025042)

Shalu Sharma (240343025045)

Have successfully completed their project on

Analysis of Air Quality and Forecasting

Under the guidance of Mrs. Trupti Joshi Ma’am and Prasad Deshmukh sir



**ACKNOWLEDGEMENT**

This project Analysis of Air Quality and Forecasting using time series Analysis was a great learning experience for us and we are submitting this work to CDAC Know IT (Pune).

We all are very glad to mention the name Mrs. Trupti Joshi and Mr. Prasad Deshmukh for their valuable guidance to work on this project. His guidance and support helped us to overcome various obstacles and intricacies during the course of project work.

We are highly grateful to Mr. Vaibhav Inamdar Manager (Know IT), CDAC, for his guidance and support whenever necessary while doing this course Post Graduate Diploma in Big Data Analytics (PGDBDA) through CDAC ACTS, Pune.

Our most heartfelt thanks go to Mr. Prasad Deshmukh (Course Coordinator, PGDBDA) who gave all the required support and kind coordination to provide all the necessities like required hardware, internet facility and extra Lab hours to complete the project and throughout the course up to the last day here in CDAC Know IT, Pune.



**TABLE OF CONTENTS**

1. ABSTRACT
2. INTRODUCTION
3. DATA COLLECTION AND FEATURES
4. SYSTEM REQUIREMENTS
   1. Software Requirements
   2. Hardware Requirements
5. FUNCTIONAL REQUIREMENTS
6. ARCHITECTURE
7. MACHINE LEARNING ALGORITHMS
8. DATA VISUALIZATION AND REPRESENTATION
9. CONCLUSION AND FUTURE SCOPE

10. REFERENCES



**Abstract**

This project focuses on developing a machine learning model to predict important factors effecting air pollution levels across Indian states, leveraging a comprehensive dataset spanning from 2010 to 2023 and covering 453 cities. We employ PySpark for efficient Extract, Transform, and Load (ETL) processes, enabling scalable data handling and processing. Advanced machine learning techniques are then applied to uncover trends and patterns in air pollution, assess the effectiveness of pollution control measures, and generate actionable insights. These insights, visualized using Tableau, aim to support policymakers and researchers in formulating strategies to enhance air quality and public health in India.

Our project addresses a critical need for precise and reliable air pollution forecasting, providing valuable tools to tackle environmental challenges in the country. In the era where technology plays a pivotal role in understanding complex systems, leveraging vast datasets through advanced analytics becomes crucial. We aim to harness the power of Big Data and employ cutting-edge machine learning algorithms to process extensive meteorological data, enabling us to unveil precise and reliable air quality patterns.



**Introduction**

As the largest growing industrial nation, India is producing record amount of pollutants specifically Co2, pm2.5, etc and other harmful aerial contaminants. Air quality of a particular state or a country is a measure on the effect of pollutants on the respected regions, as per the Indian air quality standard pollutants are indexed in terms of their scale, these air quality indexes indicate the levels of major pollutants on the atmosphere. There are various atmospheric gases which causes pollution on our environment.

Each pollution has individual index and scales at different levels. The major pollutants Such as (PM10, PM2.5, CO, O3, NO2, SO2, NH3, Pb.) indexes AQI is acquired, with this individual AQI, the data can be categorized based on the limits. We collected the data from the Indian government database, which contains pollutant concentration occurring at various places across India.

We have designed a model to predict the air quality of every available data points in the dataset, our model is capable of forecasting the air quality of India in any given area. By predicting the air quality, we can backtrack the major pollution causing pollutant and the location affected seriously by the pollutant across India. With this forecasting model, various knowledge about the data are extracted using various techniques to obtain heavily affected regions on a particular region(cluster). This give more information and knowledge about the cause and seniority of the pollutants.

By uniting machine learning models with cutting-edge big data technologies, our overarching goal is to provide accurate analysis and forecasting of air quality for selected states within our extensive dataset. This project not only contributes to the advancement of air quality forecasting methodologies but also underscores the significance of integrating data science approaches to enhance prediction accuracy in a field crucial to numerous industries.



**Dataset Collection and Features**

The dataset contains air quality data for Indian cities from the year 2010 to 2023.

It provides information about the air quality conditions in 453 cities across India. The data has been collected from the Central Control Room for Air Quality Management, which indicates that it is a reliable and authoritative source.

The data used in this project has been sourced from the Central Pollution Control Board (CPCB), which is the official portal of the Government of India. The CPCB has made the data publicly available and can be accessed at their website: [https://cpcb.nic.in](https://cpcb.nic.in/) The data was compiled from the CPCB website, which serves as the official body of the Government of India for monitoring and controlling pollution. Selenium is used for data collection. It is a powerful tool for web automation, which was instrumental in extracting and processing the data from the CPCB website. The utilization of Selenium greatly facilitated the completion of this task.

This dataset is valuable for understanding and analyzing the air quality trends and patterns in Indian cities over a span of 13 years. It can help researchers, policymakers, and the general public gain insights into the air pollution levels, identify areas with high pollution, assess the impact of air quality control measures, and develop strategies for improving air quality and public health.

It can be used for conducting statistical analyses, creating visualizations, developing predictive models, and generating actionable insights to address the air pollution challenges in India.



**Stations\_Info.CSV**

This file serves as a reference for obtaining relevant information about various stations, enabling further analysis and exploration of the data collected.

* The "stations\_info" file is a CSV file that contains information about different stations.
* The file has the following headers:  
  "file\_name," "state," "city," "agency," "station\_location," "start\_month," "start\_month\_num," and "start\_year.".

# State and Cities in Dataset

| State | Cities |
| --- | --- |
| Andhra Pradesh | Amaravati, Anantapur, Chittoor, Kadapa, Rajamahendravaram, Tirupati, Vijayawada, Visakhapatnam |
| Arunachal Pradesh | Naharlagun |
| Assam | Byrnihat, Guwahati, Nagaon, Nalbari, Silchar, Sivasagar |
|  |  |
| Chandigarh | Chandigarh |
| Chhattisgarh | Bhilai, Bilaspur, Chhal, Korba, Kunjemura, Milupara, Raipur, Tumidih |
| Delhi | Delhi |
| Gujarat | Ahmedabad, Ankleshwar, Gandhinagar, Nandesari, Surat, Vapi, Vatva |
| Haryana | Ambala, Bahadurgarh, Ballabgarh, Bhiwani, Charkhi Dadri, Dharuhera, Faridabad, Fatehabad, Gurugram, Hisar, Jind, Kaithal, Karnal, Kurukshetra, Mandikhera, Manesar, Narnaul, Palwal, Panchkula, Panipat, Rohtak, Sirsa, Sonipat, Yamuna Nagar |
|  |  |
| Jammu and Kashmir | Srinagar |
| Jharkhand | Dhanbad, Jorapokhar |
| Karnataka | Bagalkot, Belgaum, Bengaluru, Bidar, Chamarajanagar, Chikkaballapur, Chikkamagaluru, Davanagere, Dharwad, Gadag, Hassan, Haveri, Hubballi, Kalaburagi, Kolar, Koppal, Madikeri, Mangalore, Mysuru, Raichur, Ramanagara, Shivamogga, Tumakuru, Udupi, Vijayapura, Yadgir |
| Kerala | Eloor, Ernakulam, Kannur, Kochi, Kollam, Kozhikode, Thiruvananthapuram, Thrissur |
| Madhya Pradesh | Bhopal, Damoh, Dewas, Gwalior, Indore, Jabalpur, Katni, Maihar, Mandideep, Pithampur, Ratlam, Sagar, Satna, Singrauli, Ujjain |
| Maharashtra | Aurangabad, Chandrapur, Kalyan, Mumbai, Nagpur, Nashik, Navi Mumbai, Pune, Solapur, Thane |
| Manipur | Imphal |
| Meghalaya | Shillong |
|  |  |
| Nagaland | Kohima |
| Odisha | Baripada, Bileipada, Brajrajnagar, Keonjhar, Nayagarh, Rairangpur, Rourkela, Suakati, Talcher, Tensa |
| Puducherry | Puducherry |
| Punjab | Amritsar, Bathinda, Jalandhar, Khanna, Ludhiana, Mandi Gobindgarh, Patiala, Rupnagar |
| Rajasthan | Ajmer, Alwar, Banswara, Barmer, Bharatpur, Bhiwadi, Bikaner, Chittorgarh, Churu, Dausa, Dholpur, Hanumangarh, Jaipur, Jaisalmer, Jhalawar, Jhunjhunu, Jodhpur, Karauli, Kota, Pali, Pratapgarh, Rajsamand, Sawai Madhopur, Sikar, Sirohi, Sri Ganganagar, Udaipur |
| Sikkim | Gangtok |
| Tamil Nadu | Ariyalur, Chengalpattu, Chennai, Coimbatore, Cuddalore, Dindigul, Gummidipoondi, Hosur, Kanchipuram, Ooty, Palkalaiperur, Ramanathapuram, Salem, Thoothukudi, Tirupur, Vellore |
| Telangana | Hyderabad |
|  |  |
| Uttar Pradesh | Agra, Baghpat, Bareilly, Bulandshahr, Firozabad, Ghaziabad, Gorakhpur, Greater Noida, Hapur, Jhansi, Kanpur, Khurja, Lucknow, Meerut, Moradabad, Muzaffarnagar, Noida, Prayagraj, Varanasi, Vrindavan |
| Uttarakhand | Dehradun, Kashipur, Rishikesh |
| West Bengal | Asansol, Durgapur, Haldia, Howrah, Kolkata |

# 



**Parameters**



| **Parameter** | **Description** | **Unit** |
| --- | --- | --- |
| **From Date** | **Starting date of data collection** | **-** |
| **To Date** | **Ending date of data collection** | **-** |
| **PM10** | **Particulate Matter 10** | **ug/m3** |
| **PM2.5** | **Particulate Matter 2.5** | **ug/m3** |
| **CO** | **Carbon Monoxide** | **mg/Nm3, mg/m3, ng/m3, ug/m3** |
| **CO2** | **Carbon Dioxide** | **mg/m3** |
| **NO** | **Nitric Oxide** | **mg/m3, ppb, ppm, ug/m3** |
| **NO2** | **Nitrogen Dioxide** | **ug/m3** |
| **NOx** | **Nitrogen Oxides** | **ppb, ppm, ug/m3** |
|  |  |  |
|  |  |  |
| **Temp** | **Temperature** | **degrees Celsius or ug/m3** |
| **AT** | **Air Temperature** | **degrees Celsius or ug/m3** |
| **BP** | **Barometric Pressure** | **W/mt2, mg/m3, mmHg** |
| **Benzene** | **Concentration of Benzene in the air** | **mg/m3 or ug/m3** |
| **CH4** | **Methane** | **ug/m3** |
| **Eth-Benzene** | **Concentration of Ethylbenzene in the air** | **ug/m3** |
| **Gust** | **Wind Gust** | **kl/h, km/hr, m/s** |
|  |  |  |
|  |  |  |
| **MH** | **Mixing Height** | **meters (m)** |
| **MP-Xylene** | **Concentration of Meta-Para Xylene in the air** | **ug/m3** |
| **NMHC** | **Non-Methane Hydrocarbons** | **ug/m3** |
| **O Xylene** | **Concentration of Ortho-Xylene in the air** | **ug/m3** |
| **Ozone** | **Ozone Concentration** | **ppb, ug/m3** |
| **Power** | **Power Consumption** | **Watts (W)** |
| **RF** | **Rainfall** | **m/s, mm** |
| **RH** | **Relative Humidity** | **%, W/mt2, degree** |
| **SPM** | **Suspended Particulate Matter** | **ug/m3** |
| **SR** | **Solar Radiation** | **W/mt2, ug/m3** |
| **THC** | **Total Hydrocarbons** | **ug/m3** |
| **Toluene** | **Concentration of Toluene in the air** | **ug/m3** |
| **VWS** | **Wind Speed** | **degree, m/s** |
| **Variance** | **Variance** | **n** |
| **WD** | **Wind Direction** | **deg, degree C, degree** |
| **WS** | **Wind Speed** | **m/s, ug/m3** |
| **Xylene** | **Concentration of Xylene in the air** | **ug/m3** |



**Features for Air Quality Analysis**

* Air Quality Metrics:

Concentration of Pollutants: Levels of specific pollutants measured in µg/m³ or ppb (parts per billion). Common pollutants include:

* + **PM2.5:** Particulate matter with a diameter of 2.5 micrometers or less.
  + **PM10:** Particulate matter with a diameter of 10 micrometers or less.
  + **NO2:** Nitrogen dioxide.
  + **SO2:** Sulfur dioxide.
  + **O3:** Ozone.
  + **CO:** Carbon monoxide.
  + **VOC:** Volatile organic compounds.
  + **NH3:** Ammonia.



* Meteorological Data
  + **Temperature**
  + **Air temperature**, typically measured in °C or °F.
  + **Humidity:** Relative humidity percentage.
  + **Wind Speed and Direction:** Wind speed (km/h or mph) and direction (degrees or cardinal directions).
  + **Precipitation:** Amount of rainfall or snowfall (mm or inches).
  + **Atmospheric Pressure:** Pressure measurements (hPa or mmHg).
* Temporal Features
* **Date and Time**: Timestamp for each data point to analyze temporal patterns and trends.
* **Day of the Week**: To account for variations in air quality related to daily activities.
* **Season:** Seasonal patterns in air quality related to weather and pollution sources.

**Features for Forecasting:**

Historical Air Quality Data:

* **Lagged Values**: Previous time steps of air quality metrics used for time series forecasting.
* **Moving Averages**: Average values over specific time windows to smooth out fluctuations.

Weather Forecast Data:

* **Forecasted Meteorological Variables**: Predicted values for temperature, humidity, wind speed, etc.

Emission Data:

* **Predicted Emissions:** Forecasts of emissions from sources such as vehicles, industries, or agriculture.

Additional Contextual Data:



* **Traffic Volume**: Forecasted traffic data to estimate future vehicular emissions.
* **Industrial Activity:** Predictions of industrial activities that could affect air quality.
* **Special Events**: Information on upcoming events (e.g., festivals, industrial operations) that might impact air quality.



**Data Preprocessing and Feature Engineering**

we implemented several key data preprocessing steps using PySpark to ensure the dataset's consistency and quality across all state-level CSV files.

* **Data Loading and Column Normalization**:

We started by loading the dataset for each state, ensuring column names were standardized by removing special characters and converting them to a consistent format. This step prevented discrepancies due to variations in column naming conventions across different files.

* **Missing Value Handling:**

We systematically handled missing values by first filtering out rows where over 15% of the data was missing and dropping columns with more than 50% missing values. This ensured that only high-quality data was retained for analysis.

* **Outlier Detection and Treatment:**

Outliers in numeric columns were identified using the Interquartile Range (IQR) method. We replaced these outliers with the quartile value of the respective columns to avoid skewed distributions, ensuring the data was more representative of typical observations.

* **Date Conversion and Column Standardization:**

For columns containing dates, we converted them into a standardized format. This allowed for easier time-based analysis and consistency across different data points.

* **Duplicate Removal and Final Cleansing:**

After dealing with missing values and outliers, duplicates were removed from the dataset. We also applied column renaming functions to ensure all columns adhered to a uniform naming convention across all datasets.

* **Feature Engineering for State-Level Aggregation:**

To facilitate state-wise analysis, we engineered a new feature to identify the state for each data entry. This feature was derived from the filenames and included as a new column in the dataset.



* **Final Merging and Parquet Conversion:**

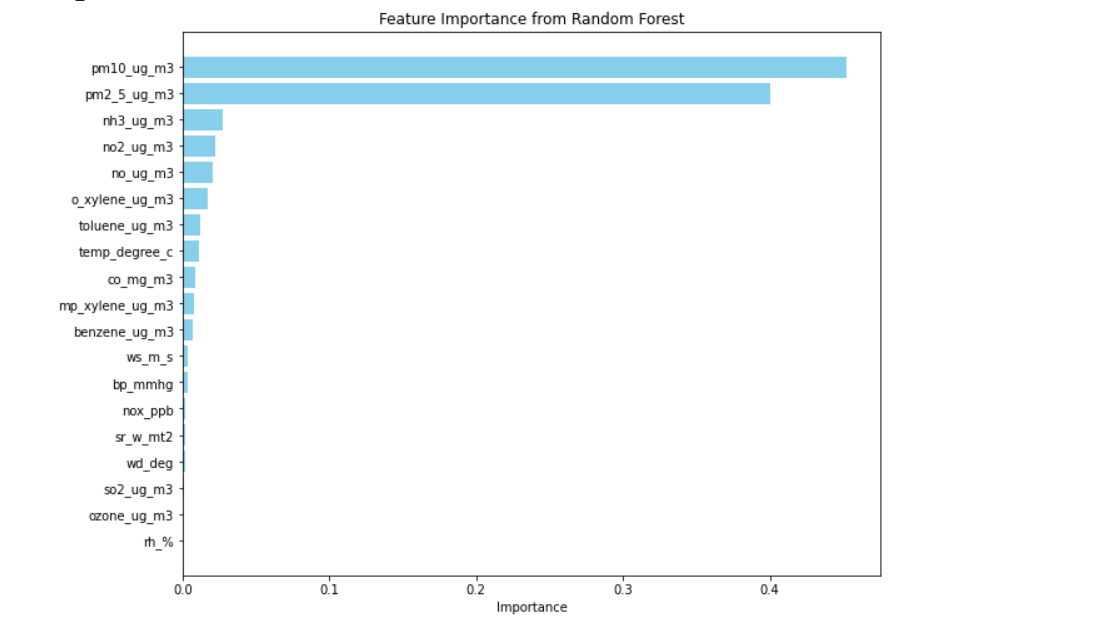
Once the individual state-level datasets were cleaned and standardized, they were merged into a single dataset. This final dataset was saved as a Parquet file for efficient storage and retrieval in subsequent steps.

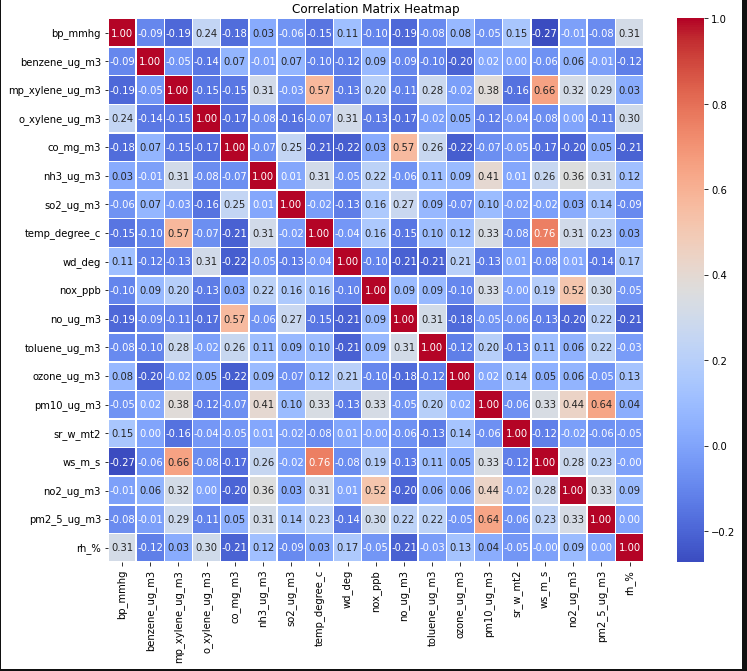
**Feature Engineering**

Feature engineering was performed on the merged parquet file, where numeric columns were evaluated for correlation to understand relationships between different variables. A correlation matrix was generated using PySpark to identify highly correlated features.

In parallel, Random Forest modeling was utilized to determine feature importance. The data was transformed using a VectorAssembler to combine multiple features into a single vector, and a StringIndexer was applied to the target column. This process helped to identify key variables influencing air quality, such as **PM10 ,PM2.5 ,NO2 ,CO ,CO2** as well as relatively less important features like **SO2**.

This information was crucial in refining the predictive models and ensuring that the most impactful variables were included in the forecasting processes.







From the correlation matrix heatmap, we can extract several insights about the relationships between different variables. Some key observations are:

* **bp\_mmhg** shows a strong negative correlation with **temp\_degree\_c** (-0.76), indicating that as the temperature increases, atmospheric pressure tends to decrease.
* **nh3\_ug\_m3** has moderate positive correlations with **benzene\_ug\_m3** (0.31) and **toluene\_ug\_m3** (0.41), suggesting a relationship between ammonia and volatile organic compounds.
* **so2\_ug\_m3** has low correlations with most variables, indicating it might not be strongly related to other pollutants or meteorological factors in this dataset.
* **ws\_m\_s** (wind speed) has a moderate negative correlation with **co\_mg\_m3** (-0.26) and **pm10\_ug\_m3** (-0.44), implying that higher wind speeds may help disperse particulate matter and carbon monoxide, reducing their concentrations.
* **o\_xylene\_ug\_m3** and **mp\_xylene\_ug\_m3** are highly correlated (0.66), which makes sense as they are both isomers of xylene and likely to be emitted from similar sources.
* **no\_ug\_m3** and **no2\_ug\_m3** have a relatively high positive correlation (0.52), reflecting their common origin from combustion processes, such as vehicular emissions.
* **pm10\_ug\_m3** and **pm2\_5\_ug\_m3** exhibit a moderate positive correlation (0.64), which indicates that particulate matter of different sizes may often coexist in similar concentrations



**System Requirement**

**Hardware Requirements:**

* Computer: Ensure your computer has sufficient processing power and memory to execute data processing and analysis tasks effectively. A modern multicore processor and at least 25 GB of RAM are recommended for handling the complexities of our project.
* Storage: Allocate ample storage space for storing the generated dataset and any additional datasets if required. Utilizing an SSD (Solid State Drive) is advisable to facilitate faster data access, particularly when managing large volumes of structured and unstructured data.
* Internet Connection: A stable internet connection is critical for downloading and installing necessary software packages, libraries, and accessing online resources essential throughout our project.

**Software Requirements:**

**Operating System:** The project environment is set up on VMware, which runs Linux as the operating system. This virtualized setup ensures scalability and flexibility during development and execution.

**Python:** Python (3.10.12) is the primary language used for data processing, analysis, and machine learning, serving as the foundation of the project.

**PySpark:** We use PySpark (3.4.0) for the preprocessing pipeline. PySpark handles the initial ingestion of raw data, which is typically in complex csv formats, and transforms it into a structured format through defined schemas. The structured data is subsequently stored in Hive for efficient querying.

**Hive:** Apache Hive is employed to store and manage large datasets after they are preprocessed. It supports our queries and aids in feeding data into the machine learning and visualization pipelines.

**Machine Learning Libraries**: For time-series modeling and forecasting, statsmodels (0.14.1) is utilized. Specifically, ARIMA and SARIMAX models are chosen for their ability to handle the seasonal and temporal components of the air quality data.

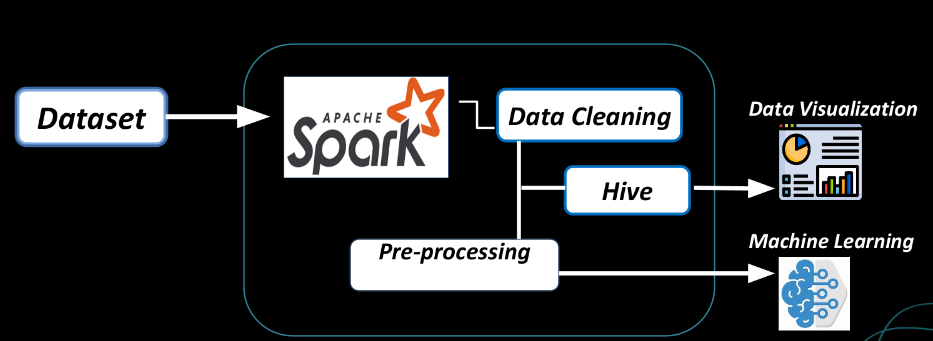


**Data Visualization:** Visualization is a critical component of the project, aiding in conveying insights and trends derived from the data. Tableau Public

2 024.1 is used for comprehensive visual representations, making it easier to communicate findings with stakeholders.

This comprehensive set of hardware and software requirements showcases the robust infrastructure and tools employed in our project, emphasizing efficiency and scalability in handling large datasets and complex data processing tasks. The adjustment to 25 GB of RAM reflects the optimization made during the project, maintaining effective performance while conserving resources.

**Architecture**



This architecture outlines the data flow, preprocessing, storage, and model development steps utilized in the Air Quality Prediction Project:

**1. Data Ingestion**

* **Source**: The dataset is collected from various air quality monitoring stations across different regions of India
* **Format**: Raw data is typically in complex csv formats containing air quality parameters like temperature, pollutant levels (PM2.5, PM10, etc.), and meteorological variables.

**2. Apache Spark for Preprocessing**

* **Data Cleaning**:
  + The raw data is first ingested into **Apache Spark** where data cleaning processes are performed. This involves handling missing values, removing duplicates, and ensuring consistency in data formats.
  + **PySpark** is utilized to define schemas, parse the complex csv data, and transform it into a structured tabular format.



* **Preprocessing**:
  + After data cleaning, feature engineering is performed, including:
    - **Correlation Analysis**: A correlation matrix is generated using PySpark to identify relationships between numeric variables (e.g., temperature, xylene, and pm10).
    - **Random Forest Modeling**: Used for determining feature importance, which highlights key variables like temperature, o-xylene, and mp-xylene as influential in determining air quality.
  + The structured data is further transformed using a **VectorAssembler** to combine multiple features into a single vector.
  + **StringIndexer** is applied to the target column to encode categorical values.

**3. Data Storage**

* **Hive**: After preprocessing, the structured data is stored in **Hive** for efficient querying and management. Hive acts as the primary storage layer where cleaned and processed data is saved.

**4. Machine Learning**

* **ARIMA/SARIMAX** models from the statsmodels (v0.14.1) library are used for forecasting air quality trends, particularly focusing on seasonal effects.

**5. Data Visualization**

* **Tableau**: Visualization of the insights derived from the processed data and machine learning models is done using Tableau. The visualization highlights trends, patterns, and the effectiveness of air quality measures, aiding policymakers and researchers.



**Algorithms**

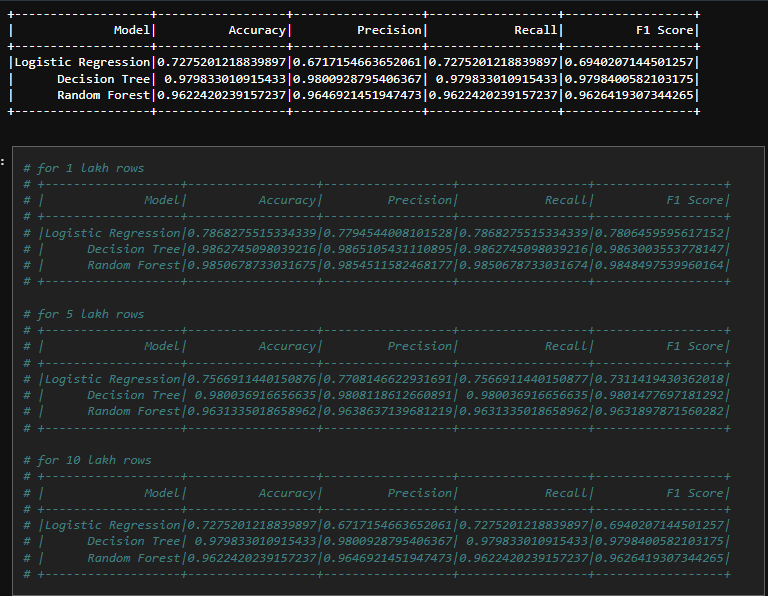


**Model Training and Hyperparameter Tuning:**

* Three models are defined: **Logistic Regression** (LR), **Decision Tree** (DT), and **Random Forest** (RF).
* Hyperparameter tuning is performed using cross-validation for each model. For example, for Logistic Regression, both regParam and elasticNetParam are tuned. For Decision Trees and Random Forests, the number of trees, depth, and other parameters are varied.
* The cross-validation results are used to select the best model based on accuracy.

**Model Evaluation**:

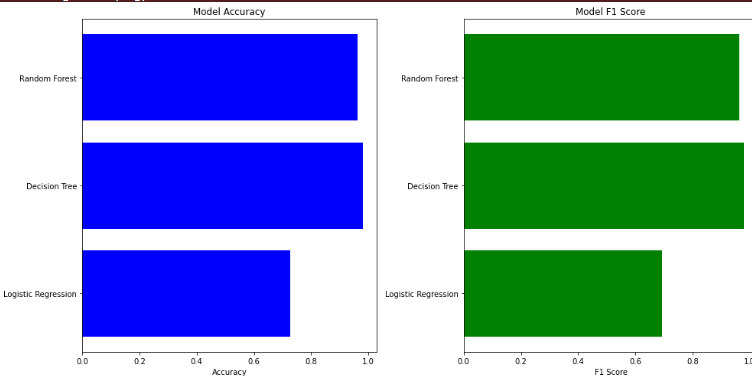
* The models are evaluated on the test dataset using multiple evaluation metrics: Accuracy, Precision, Recall, and F1 Score.
* Evaluation is performed using MulticlassClassificationEvaluator.





**Summary of Model Performance:**

* **Decision Tree & Random Forest Performance**: Both Decision Tree and Random Forest classifiers show high accuracy (~98%), likely due to their ability to handle non-linear relationships and complex interactions between features.
* **Logistic Regression Performance**: Logistic Regression underperformed with an accuracy of ~72%, indicating that the data might not be linearly separable or that further feature engineering is needed.



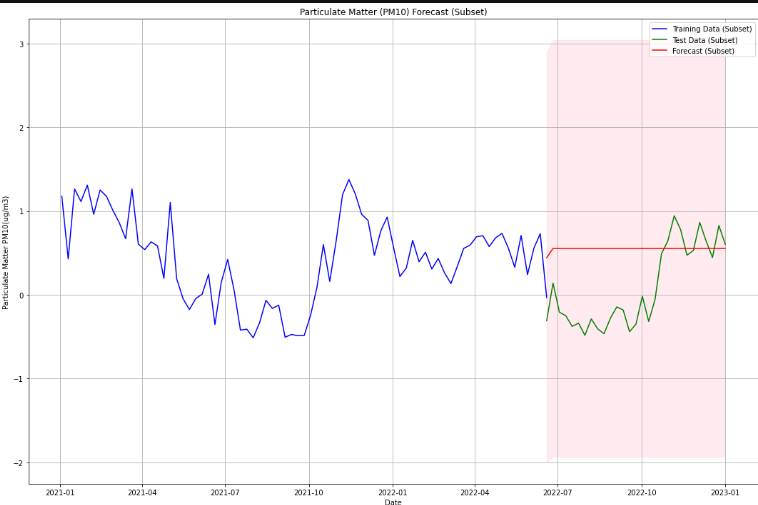
**AutoRegressive Integrated Moving Average (ARIMA):**

The AutoRegressive Integrated Moving Average (ARIMA) model is a widely used statistical tool for time-series forecasting, including weather temperature prediction. The model combines autoregression, differencing, and moving average components to analyze and predict future values based on historical data.

The ARIMA model was applied to the scaled air quality data to forecast particulate matter (PM10) levels. Here's a summary of the approach and results:

* **Data Preparation**:
* Scaled data was converted to a Pandas DataFrame .
  + The date column was set as the index to facilitate time series analysis.
* **Model Fitting**:
  + The ARIMA model was trained using the **pmdarima** library with automatic order selection to minimize the AIC (Akaike Information Criterion).
  + The best model identified was ARIMA(3,0,1) with an intercept, which was selected based on the lowest AIC value.

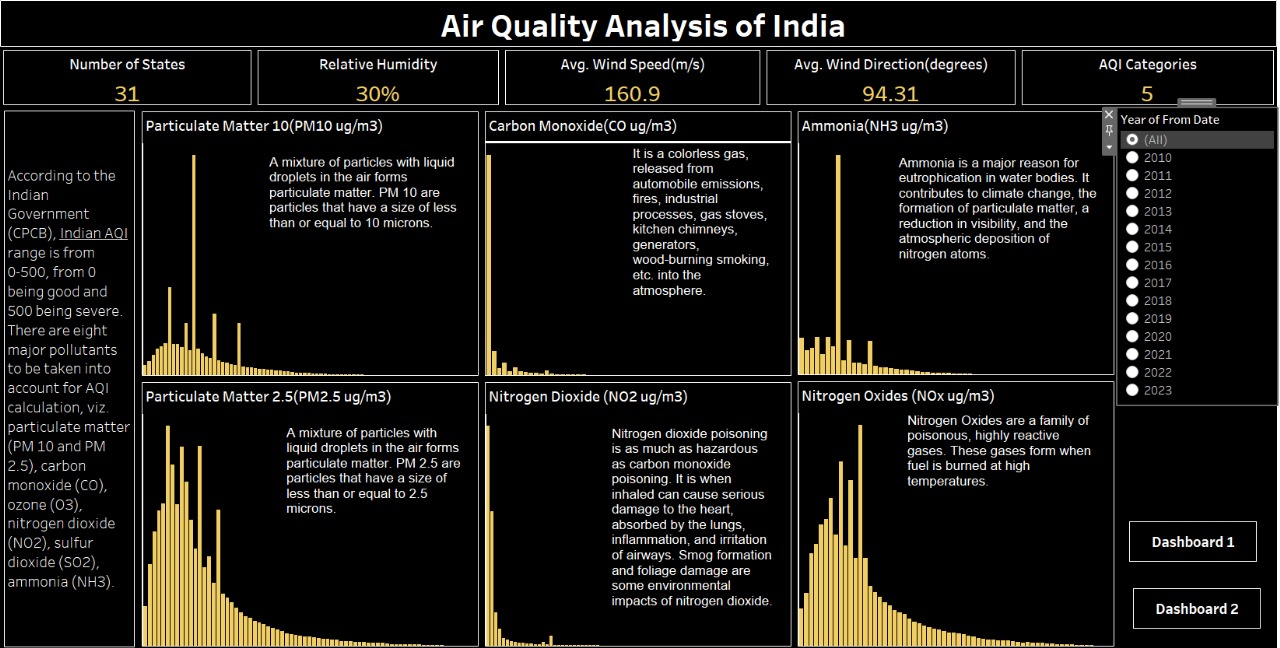


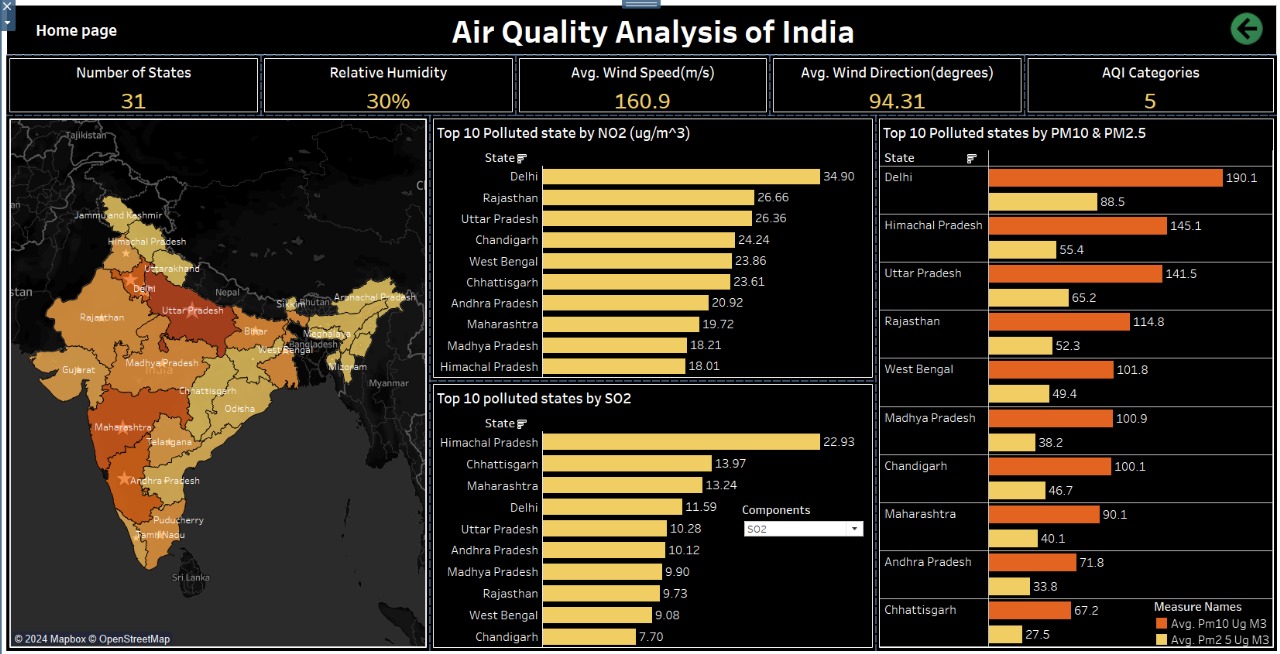
* **Forecasting**:
  + Forecasts were generated for the test period, and confidence intervals were computed.
  + Forecast results were then smoothed using a rolling mean to enhance clarity.
* **Visualization**:
  + Plots were created to compare the smoothed actual data (train and test) with the forecasted values.
  + Confidence intervals were visualized to show the range of uncertainty around the forecasts.
  + Data was resampled to weekly averages to reduce noise and better illustrate trends.
  + Additionally, a subset of the data was plotted to provide a focused view of the forecast results over a specific period.

The ARIMA model demonstrated its ability to forecast PM10 levels effectively, with visualizations highlighting the model's performance and predictive accuracy over time.

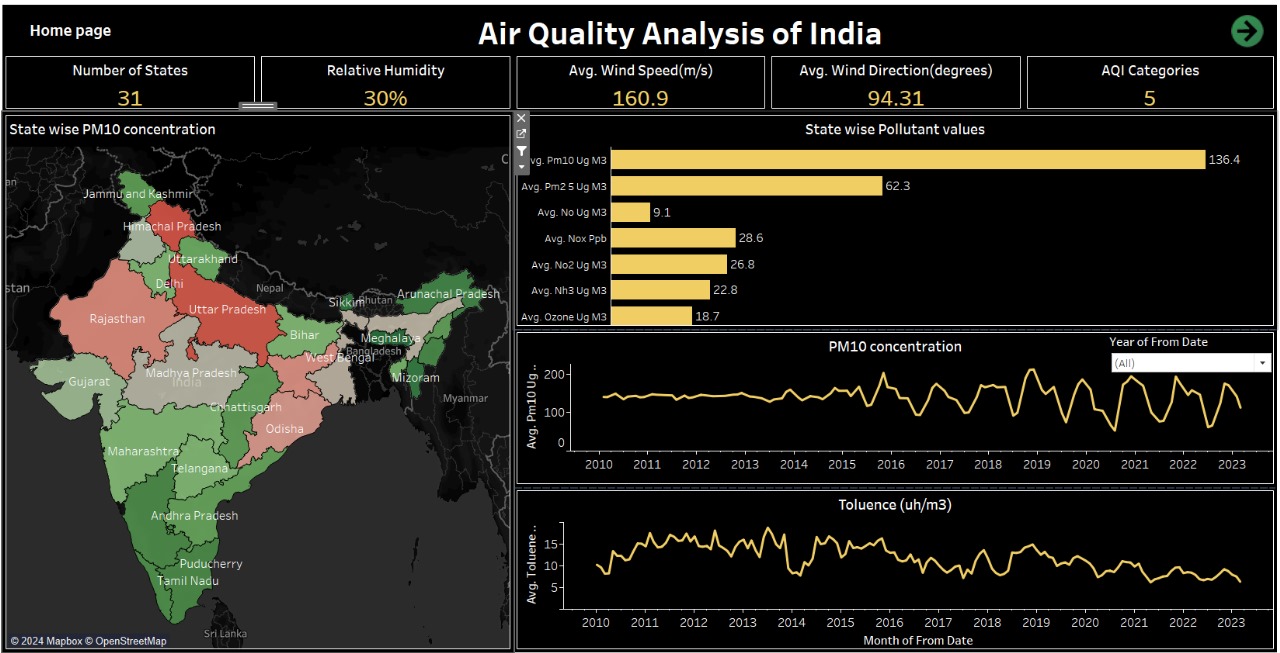
**Data Visualization:**













**Conclusion And Future Scope**

Our project represents a sophisticated and comprehensive system for predicting the current data with 98% accuracy. It will successfully predict the upcoming air quality of any particular data within a given region. With this model we can forecast the air quality and alert the respected region of the country also it a progressive learning model it is capable of tracing back to the particular location needed attention provided the time series data of every possible region needed attention. The air quality information utilized in air quality checking and investigation stage, and incorporates the normal every day fine particulate issue (PM2.5), inhalable particulate issue (PM10), ozone (O3), CO, SO2, NO2 fixation and air quality record(AQI).The essential perspectives that should be viewed as with regards to gauging of the poison focus are its different sources alongside the components that impact its fixation

In conclusion, our project represents a sophisticated and comprehensive system for predicting and analyzing data across 400+ cities in India. The integration of Python, PySpark and Tableau facilitated a robust and scalable infrastructure for data retrieval, preprocessing, machine learning modeling, and user interaction. The combination of PySpark and played a pivotal role in efficiently handling the complex and unstructured initial data, enabling us to derive valuable insights through machine learning models. The use of MongoDB ensured structured data storage and retrieval, while Tableau enhanced the visualization of our findings.



**Future Scope:**

The project lays a solid foundation for future enhancements and expansions. Here are some potential avenues for future development:

* **Enhanced Data Collection**

**Advanced Sensors:** Development of more accurate, cost-effective, and compact sensors for detecting various pollutants.

**IoT Integration:** Increased use of Internet of Things (IoT) devices for real-time air quality monitoring.

**Satellite Technology:** Improved satellite systems for broader and more detailed air quality data collection.

* **Policy and Decision Support**

**Regulatory Frameworks:** Development of more effective air quality regulations based on advanced forecasting models and data analysis.

**Urban Planning:** Integration of air quality forecasts into urban planning and development to create healthier living environments.

* Emerging Pollutants

Novel Pollutants: Identification and analysis of emerging pollutants (e.g., microplastics, new chemicals) and their impact on air quality.

Innovative Detection Methods: Creation of new methods for detecting and measuring these pollutants.

* International Collaboration

Global Networks: Enhanced international cooperation for sharing air quality data and forecasting models across borders.

Standardization: Development of global standards for air quality monitoring and reporting.

**References**



* Apache Spark. [https://spark.apache.org/]
* https://www.timlrx.com/blog/feature-selection-using-feature-importance-score-creating-a-pyspark-estimator
* Python. [[https://ww](http://www.python.org/)w.py[thon.](http://www.python.org/)org/]
* https://www.pranaair.com/in/blog/what-is-air-quality-index-aqi-and-its-calculation/?srsltid=AfmBOoo8u86s-wXd8BQgNtHugqdrMN1IBZzEYb9c3HHTUgipRtuGT8mg
* [<https://www.ripublication.com/ijaerspl2019/ijaerv14n11spl_34.pdf>]
* [https://cpcb.nic.in]
* <https://medium.com/analytics-vidhya/time-series-forecasting-using-spark-ml-part-2-31506514c643>