10 Pearls Shine Internship Project Report

Project: Karachi AQI Predictor

Author: Muhammad Hani

Company: 10 Pearls

Mentor: Muhammad Mobeen

Program: Shine Internship Program (Data Science Track)

Table of Content

1. Project Objectives:

1. Collect and preprocess air quality data for Karachi from reliable sources.
2. Perform Exploratory Data Analysis (EDA) to identify trends, correlations, and patterns in pollutant levels.
3. Engineer features to improve the predictive capability of AQI models.
4. Develop and train machine learning models to forecast hourly and daily AQI levels.
5. Interpret model predictions using SHAP analysis to identify the most influential features.
6. Build a Streamlit dashboard for real-time visualization of AQI predictions and insights.
7. Automate data pipelines for continuous updates and model retraining.
8. Provide actionable insights to aid in monitoring and managing air quality in Karachi.

2. Tools and Technologies

- **Language:** python
- **Feature Store:** Hopsworks
- **Model Registry:** Hopsworks
- **CICD:** Github Actions
- **Frontend:** Streamlit

3. Methodology

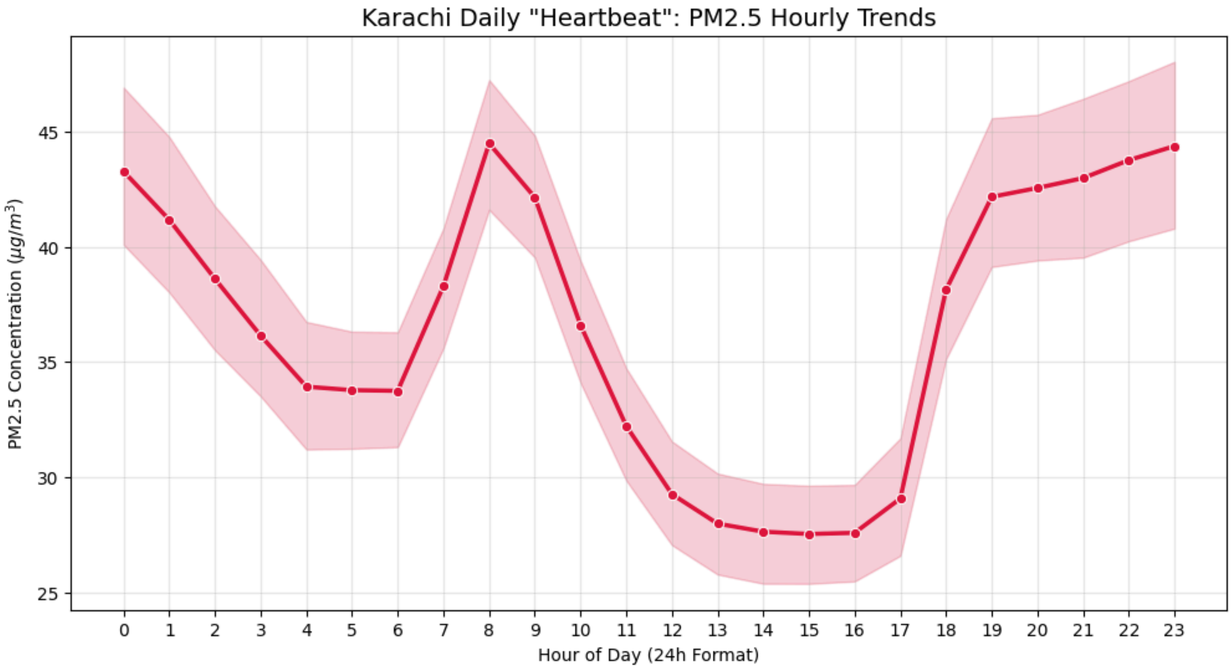
3. 1 Data Collection:

Historic weather and air pollutant data for Karachi were collected from the Open Meteo API, including key variables such as temperature, humidity, wind speed, PM2.5, PM10, and other pollutants. Collected data was cleaned to handle missing values and ensure consistency, providing a reliable dataset for subsequent analysis and predictions.

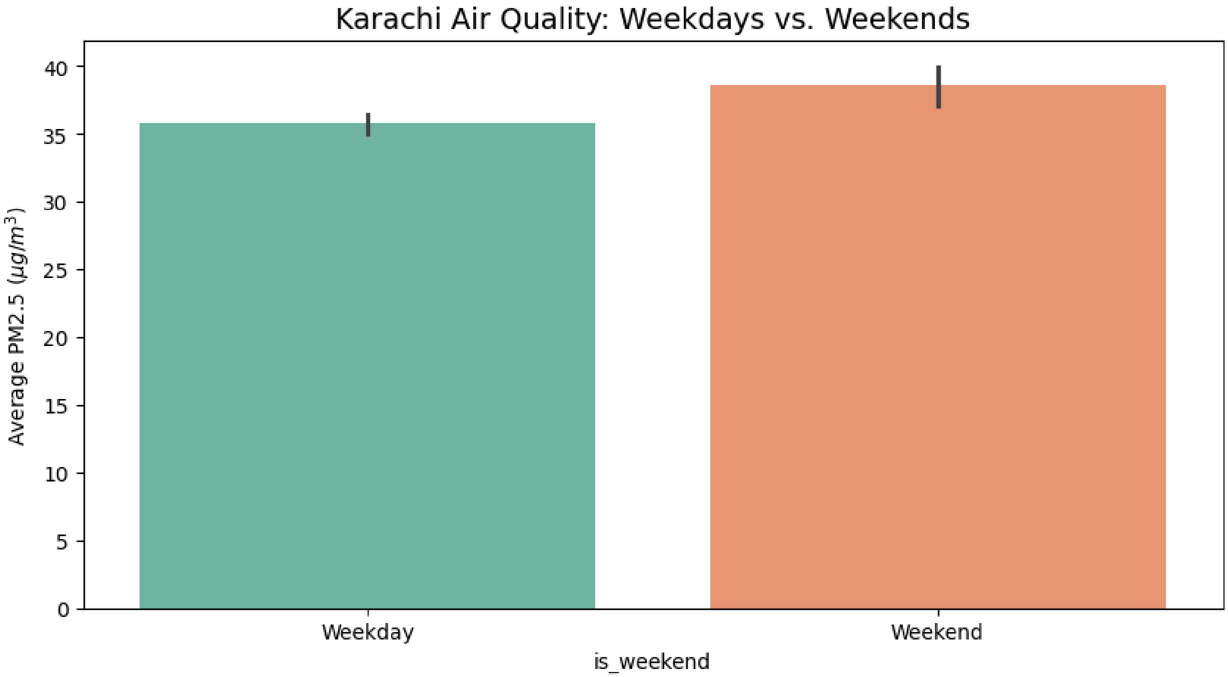
3.2 Exploratory Data Analysis:

Exploratory Data Analysis was performed on the previous three months of data to understand patterns and relationships between different weather variables and air pollutants. Correlation analysis and visualizations, such as scatter plots and heatmaps, were used to examine how features like temperature, humidity, PM2.5, and PM10 interact with each other and affect air quality.

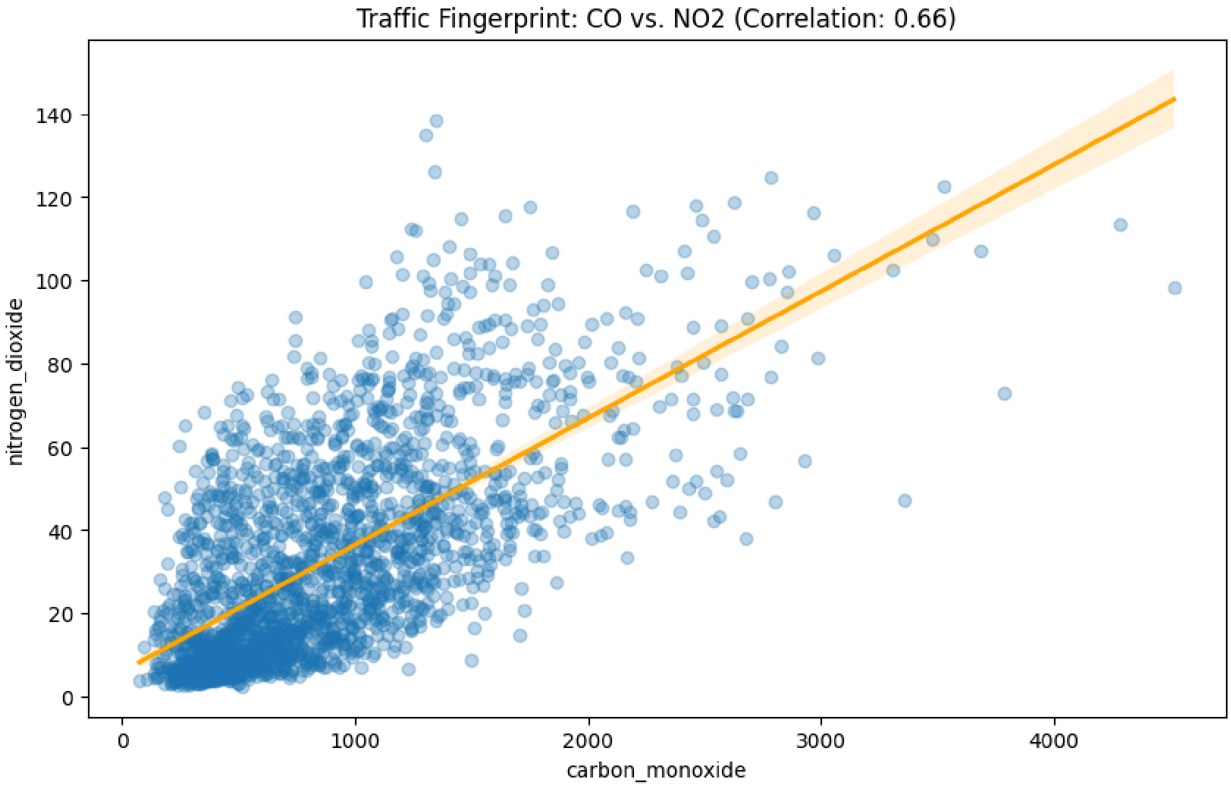
The following Heartbeat plot shows that PM2.5 Concentration is higher at morning hours between 8 to 9 am, due to schools going and office going hours.



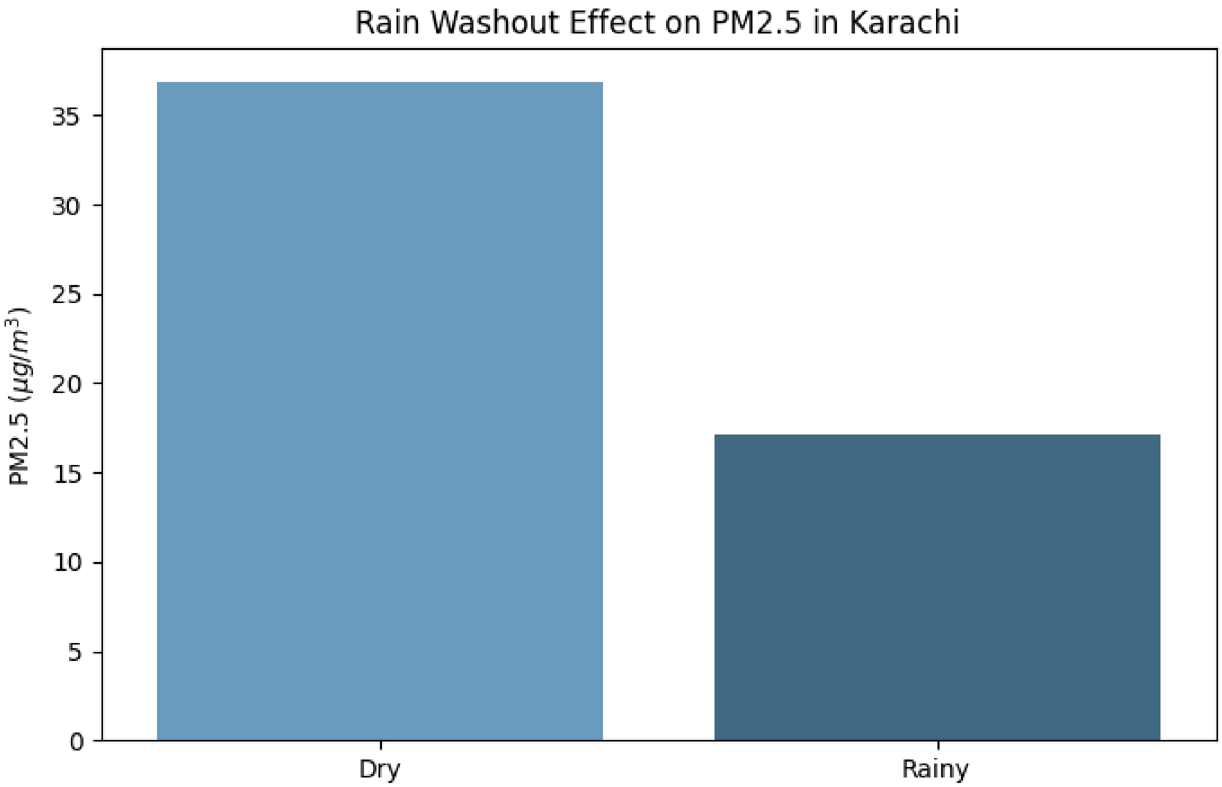
The following plot indicates that Karachi Air Quality is worst at weekends with higher concentration of PM2.5.



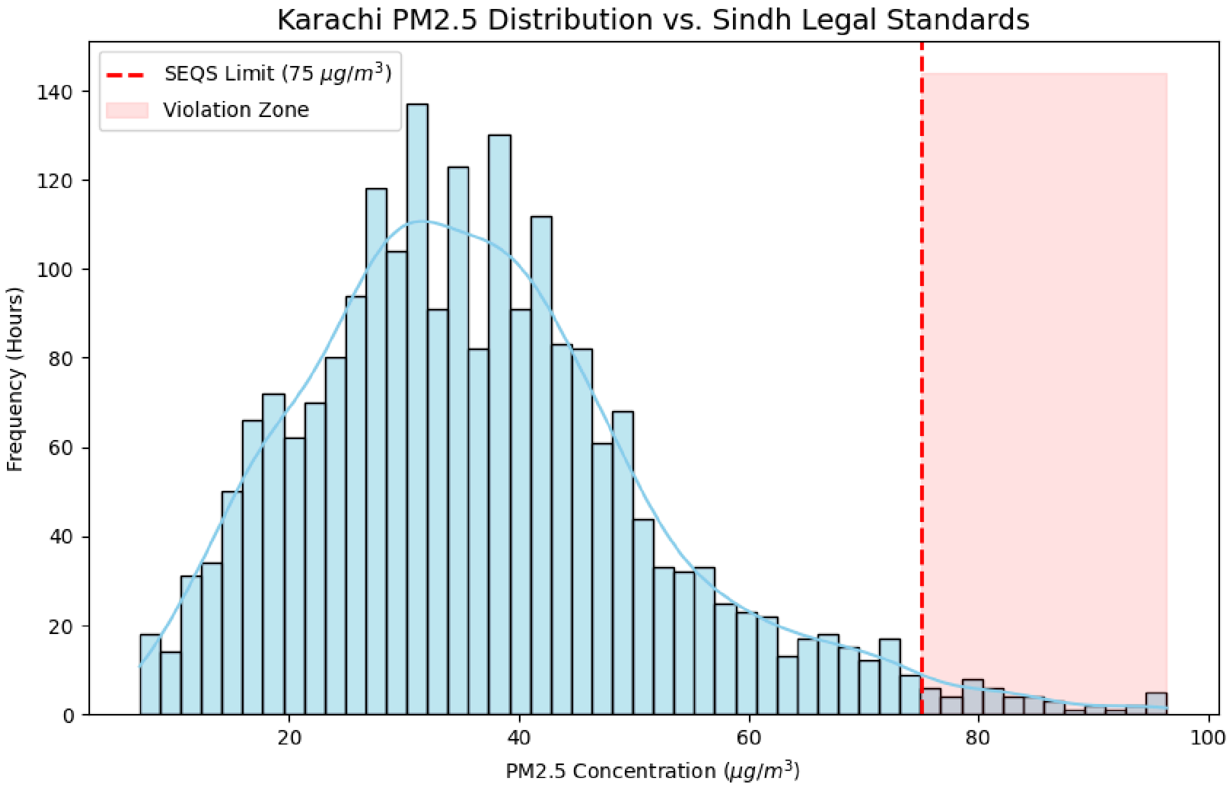
The Traffic Fingerprint Plot shows a correlation between CO vs NO2. With correlation being 0.66, it indicates that Karachi has high traffic pollution.



The below plot indicates that rainy days cause the PM2.5 concentration to decrease as compared to dry days.



The plot indicates that Karachi has violated the legal limit for PM2.5 Concentration.



3.3 Feature Engineering:

Features were engineered to help the model better understand temporal patterns, pollutant behavior, and AQI trends.

- Time-based features, day_of_week, hour, month, is_weekend, hour_sin, hour_cos were created to capture daily and seasonal patterns in air quality.
- Pollutant variables pm2_5, pm10, no2, o3, co, so2 were included because they directly influence AQI levels.
- Lag features were added to capture historical dependencies, as AQI is strongly influenced by previous hours.
- Rolling and change-based features were engineered to represent short- and long-term trends, variability, and sudden shifts in air quality.
- Future targets (aqi_next_24h, 48h, 72h) were created to enable multi-step forecasting.

3.4 Model Selection

Three models were selected based on their type and strengths in handling structured environmental data.

- Random Forest: A basic ensemble learning model selected as a strong baseline due to its robustness, ability to reduce overfitting, and effectiveness on tabular data.
- XGBoost: An advanced gradient boosting model chosen for its high predictive accuracy and ability to capture complex non-linear relationships.
- LightGBM: A fast and efficient gradient boosting model selected for its speed, scalability, and strong performance on large datasets.

4 Challenges

4.1 Historic Data Api:

One major challenge was identifying a reliable API that provided historical air quality data, specifically including the US AQI values. This was crucial because many engineered features, such as lag variables and rolling statistics, were directly dependent on accurate past AQI data. Selecting the right data source ensured consistency in feature engineering and improved overall model reliability.

4.2 Target Data Leakage:

Another significant challenge was unintended data leakage during target creation, which initially resulted in unrealistically high R² scores. The issue occurred because future information was indirectly influencing the training data. To resolve this, the target variables were correctly shifted using a shift(-24), shift(-48), and shift(-72) approach, ensuring that only past data was used to predict future AQI values. This adjustment made the evaluation metrics more realistic and reliable.

4.3 Model inference issue in updated models:

After updating and versioning the trained models, an issue occurred during inference where predictions failed due to input format mismatches. The model expected data in a specific format, and passing a DataFrame caused errors in certain versions. This was resolved by converting the inference input data to a NumPy array before prediction, ensuring compatibility with the trained model and stable deployment in the application.

4.4 incremental updating data in feature store:

Another challenge was that NaNs in previous target rows were not being correctly computed as new data arrived. This happened because there was no buffer system to maintain historical rows for computing lag and rolling features. Without this buffer, newly arriving data could not reference earlier values, causing incomplete feature calculations. Implementing a proper buffer ensured that all lagged and rolling target features were consistently and accurately computed.

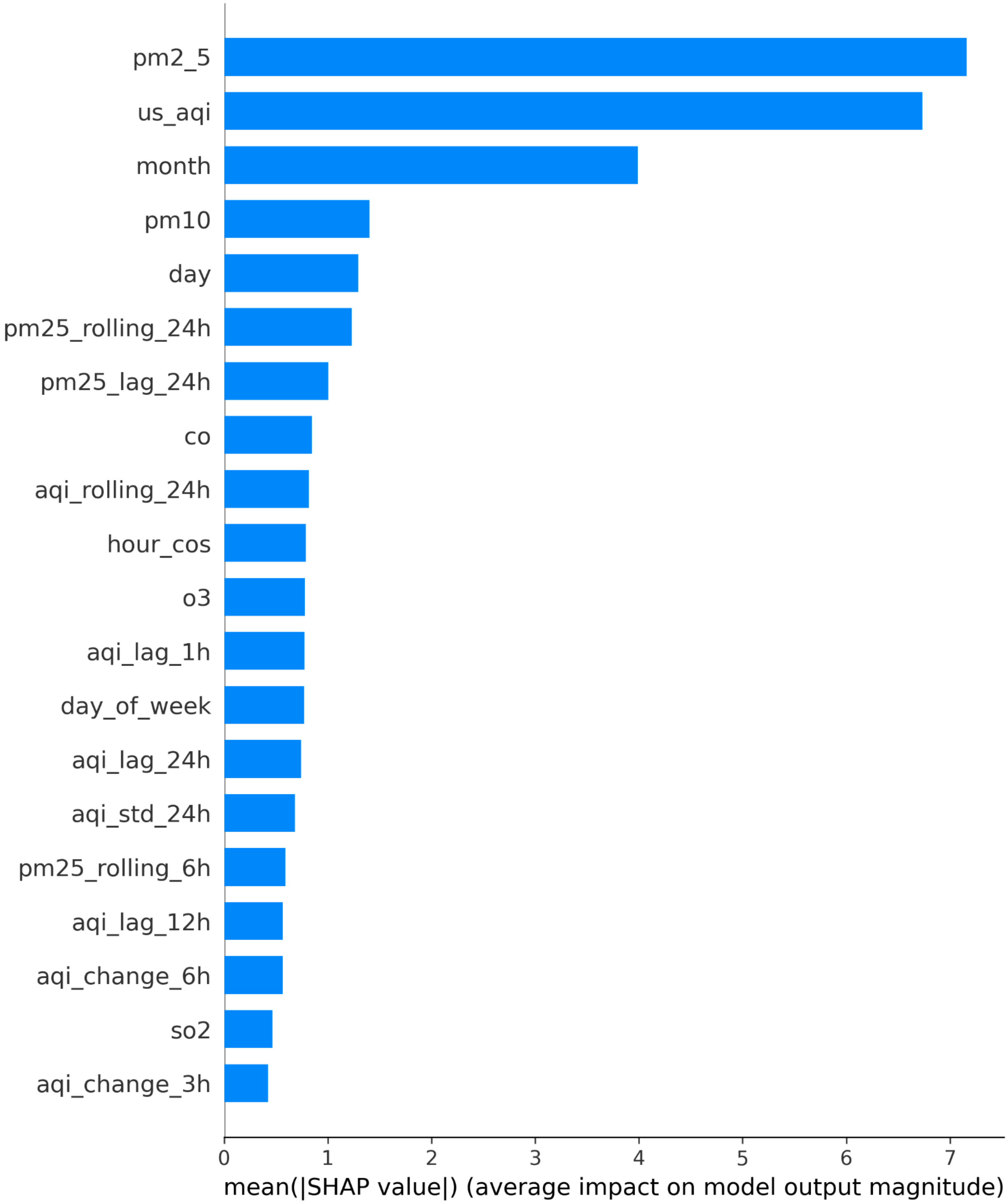
4.5 API giving future data:

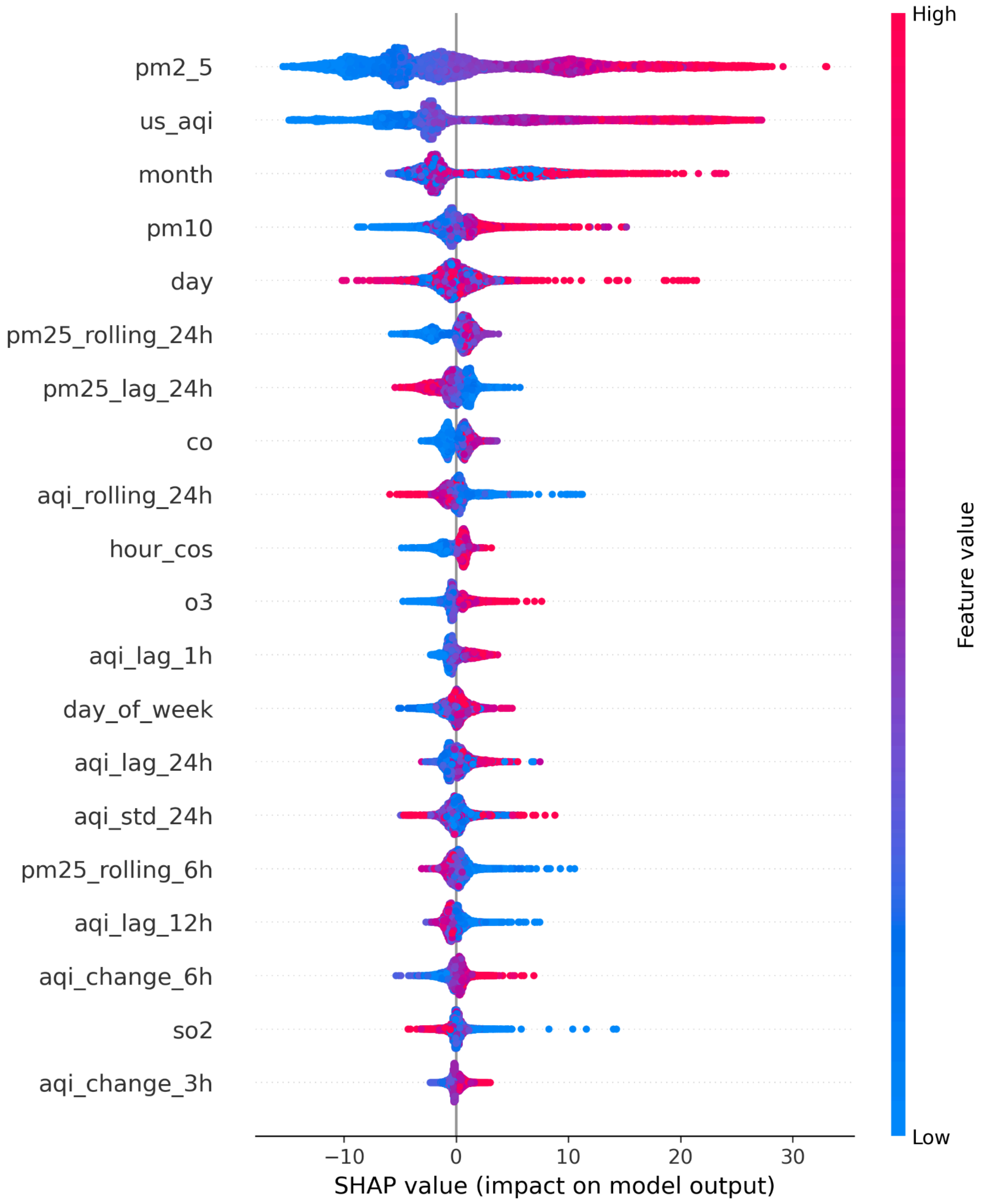
The AQI data source provided full-day data, which included forecasted values for future hours. Using this data directly would have caused the model to train on information from the future, leading to unrealistic predictions. To solve this, a condition was implemented to fetch only data for hours less than or equal to the current hour, ensuring that the raw data stored in Hopsworks contained only actual observed values and no forecasted data.

5 Results and Insights

5.1 SHAP Analysis:

SHAP analysis was performed to understand how the model was making predictions and which features had the most influence on AQI forecasts. This interpretability method highlighted the relative importance of each feature, including lagged AQI values, pollutant levels, and time-based features.


pm2_5
us_aqi
month
pm10
day
pm25_rolling_24h
pm25_lag_24h
co
aqi_rolling_24h
hour_cos
o3
aqi_lag_1h
day_of_week
aqi_lag_24h
aqi_std_24h
pm25_rolling_6h
aqi_lag_12h
aqi_change_6h
so2
aqi_change_3h
0
1
2
3
4
5
6
7
mean(|SHAP value|) (average impact on model output magnitude)


pm2_5
us_aqi
month
pm10
day
pm25_rolling_24h
pm25_lag_24h
co
aqi_rolling_24h
hour_cos
o3
aqi_lag_1h
day_of_week
aqi_lag_24h
aqi_std_24h
pm25_rolling_6h
aqi_lag_12h
aqi_change_6h
so2
aqi_change_3h
−10
0
10
20
30
SHAP value (impact on model output)
High
Low
Feature value

SHAP analysis for the 24-hour AQI prediction horizon revealed which features the model relied on most. The results showed that PM2.5 was the most crucial pollutant affecting predictions, followed by US AQI, month, PM10, day, and various rolling features capturing short- and long-term trends. This analysis provided clear insight into how different factors influence air quality forecasts in Karachi.

6 Achievements

6.1 Realistic R^2, RMSE score:

The XGBoost model achieved a reliable R^2 score of 0.30, RMSE 19, and MAE 15 which is realistic given the highly volatile and spiking nature of Karachi's pollutant data. A higher R^2 would have indicated potential data leakage or overfitting, similar to the earlier case with rolling mean targets, where R^2 exceeded 0.9 but the model showed no real learning and was effectively predicting today's AQI ≈ yesterday's AQI. This validated that the current model captures meaningful patterns without relying on leaked or artificial information.

6.2 Model Updating:

The model is trained daily, and the best-performing version is stored in the model registry. The front-end dashboard loads the model with the lowest RMSE from the registry for inference, ensuring that predictions are always generated using the most accurate and up-to-date model.

6.3 Learning Outcomes:

- Gained hands-on experience in data collection, preprocessing, and feature engineering for time-series air quality data.
- Learned to handle challenges like data leakage, missing values, and pipeline automation.
- Developed skills in machine learning model selection, training, and evaluation (Random Forest, XGBoost, LightGBM).
- Applied SHAP analysis to interpret model predictions and understand feature importance.
- Built a Streamlit dashboard to visualize predictions and insights effectively.