Forecasting Air Pollution with Machine Learning

COMP9417 Group: Cookies

Finn Devine (z5591178), Melinda Yang (z5574918), Chun Kin Chau (z5440431),
Hui Ting Wai (Jack) (z5515119) & Junchi Zhu (z5498685)

Introduction

This project utilises the Air Quality dataset from the UCI ML Repository from March 2004 to February 2005 to investigate machine learning approaches for forecasting air pollution. The dataset provides 9358 hourly observations containing sensor responses, meteorological variables and pollutant concentrations (Vito 2008). Our goal is to build data-driven models that capture temporal dependencies in pollutant behaviour.

Data Analysis - Exploratory Data Analysis (EDA)

While the meteorological features (T, RH, AH) are generally well-recorded in the dataset, some chemical sensors exhibit higher dropout rates. From initial inspection, several sensor variables, for instance NMHC(GT), contained large blocks of missing or corrupted values encoded as -200, which was replaced with NaN for proper handling (**Figure 1.1**).

Time-leading plots reveal that on a daily basis, pollutants such as CO(GT) and NOx(GT) typically increase during afternoon and evening traffic hours (**Figure 1.2, 1.3**). Seasonal effects, where pollutant levels all increased during the colder months, were also observed (**Figure 1.4**).

Additionally, strong associations were observed between pollutant concentrations and their corresponding metal-oxide sensor readings, reinforcing their relevance as input features (**Figure 1.5**). Although meteorological variables and pollutant concentrations were widely dispersed with no clear monotonic trend visible. This suggests meteorological factors alone do not directly explain variations in pollutant levels at the hourly scale (**Figure 1.6**).

Overall, these observations emphasise the incorporation of time-dependent features, relationships between pollutant concentrations and chemicals, and the relationship between meteorological variables and pollutant concentrations.

Methodology - Preprocessing and Feature Engineering

As the Air Quality dataset contained corrupted values and strong time-dependent patterns, data cleaning and feature engineering were necessary to create accurate and well structured models with visible trends.

Due to the pollutant concentrations being highly time-dependent as short-term fluctuations were huge, lag variables were added to capture past pollutant behaviours to improve accuracy by providing information about earlier observations, as this is useful for time-series data with strong autocorrelation. Rolling averages were also added to present short-term temporal smoothing by calculating the average value of a fixed period (**Table 1.1**).

Anomaly and Event Detection

A residual-driven anomaly detection method, using CO(GT) as the target pollutant, was used to identify unusual pollution spikes or potential sensor faults. A Random Forest Regressor was trained on 2004 data to learn normal pollutant behavior and its relationship with meteorological or calendar features, then tested on the 2005 data. This regressor was chosen as it captures non-linear patterns, performs well to minimal tuning and is robust to noise.

Residuals were calculated as the difference between the actual and predicted values and standardised using z-scores, where points with $|z| > 3$ were labelled as anomalies. The model achieved a low root mean squared error (RMSE) of 0.682 on the 2005 test set, indicating stable and predictable performance, and flagged 2.1% of the data points as anomalies. Residuals remained close to zero for the majority of the timestamps (**Figure 1.7**).

Most predictions adhered to the actual CO(GT) values. Comparing the anomalies with previous EDA findings suggests these short-term spikes reflect rapid changes in local conditions rather than a permanent shift in data (**Figure 1.8**). These clusters may be linked to temporal weather events. In contrast, isolated anomalies are likely to be caused by sensor noise.

The residual histogram (**Figure 1.9**) follows a bell-shaped distribution centered around zero, confirming generally accurate predictions with a few extreme errors. Balancing precision and recall ensures that notable anomalies are detected while reducing false positives and improving model robustness and data reliability.

Results & Discussion

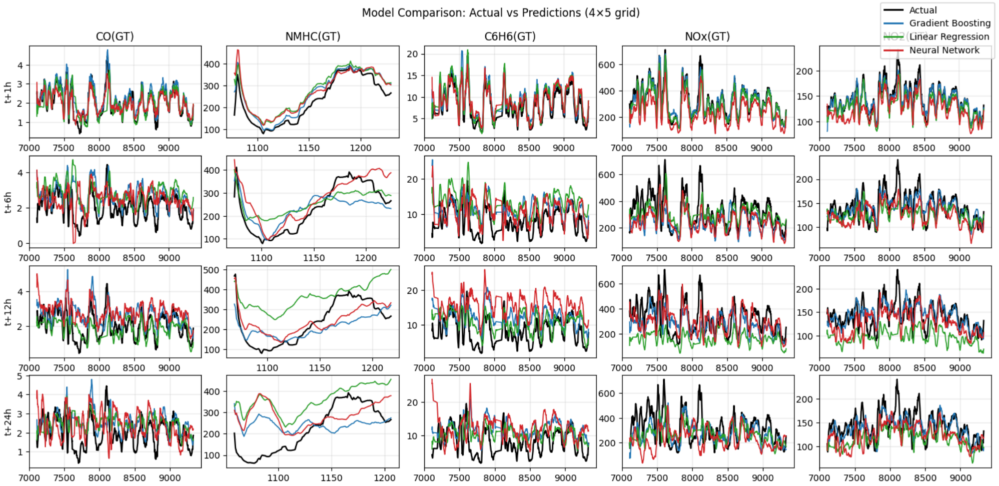
Regression Model Development

To predict the the five pollutants's concentrations at horizons of 1, 6, 12 and 24 hours ahead, Linear Regression (LR), Gradient Boosting (GB) and Neural Networks (NN) were trained using a time-ordered train-test split (2004[train] - 2005[test] except for NMHC(GT) which was split 50-50) to ensure past observations were used to predict future values.

The LR models show mixed prediction performance across the five pollutants. For CO(GT), C_6H_6(GT) and NO_2(GT), the RMSE values remained relatively low at the 1 hour horizon, but increased at later horizons (**Table 1.2**). The model for C_6H_6(GT) highlights LR's tendency to underfit spikes and oversmooth the time series, reflecting modelling limitations for longer term non-linear behaviour. NO_x(GT)'s high RMSE values across all horizons indicates LR cannot effectively model pollutants with strong variability and non-linear dynamics.

The GB model was developed with 500 estimators, maximum depth of 4 and a learning rate of 0.05. These parameters were chosen to ensure high accuracy, while preventing overfitting. For the GB model, it generally shows a lower RMSE across all pollutant concentrations (**Table 1.2**). This was expected as pollutant concentrations' trends are non-linear and include huge fluctuations. NO_x(GT) had the greatest fluctuations among all five pollutants and its model highlights GB's tendency to capture non-linear behaviour.

The NN model was developed using MLPRegressor, with 2000 maximum iterations, RELU activation, ADAM solver and initial learning rate of 0.01. These parameters were chosen to create a high accuracy model that learns quickly while still being able to converge in time. The NN model had the worst result, producing the worst RMSE results across all pollutant concentrations (**Table 1.2**). Although NN is known for capturing complex non-linear relationships, it generally works better with clean data. The air quality dataset is noisy and contains missing values, resulting in poor results. It also shows an overfitting trend in some pollutants, such as C_6H_6(GT) and CO(GT).



Classification Model Development

Gradient Boosting, Decision Trees, Random Forest and Logistic Regression were selected for classification model development. The test accuracy varied noticeably with the prediction horizon, from approximately 74% to 57% to 57% to 54% (**Figure 1.10**). For 6-h and 12-h forecasts, all models exceeded the naïve baseline, indicating that short-term temporal patterns in the data are highly learnable. At the 24-h horizon, all models fall below the baseline, suggesting that predictive information decays substantially over longer timespans. At the 1-h horizon, only GB and RF outperform the baseline, showing ensemble models capture short-range nonlinear dynamics more effectively than simpler classifiers (**Figure 1.11**).

Interpretation of results

Regression Models

The GB model performs best on average compared to other models. It is designed to capture previous errors by building many small trees. This allows the model to perform better with non-linear patterns, short-term fluctuations and seasonal structures. However, the model performed poorly when handling long-term trends and tended to overfit easily, as exhibited by CO(GT).

Alternatively, the LR model is designed to predict future pollutant concentrations by learning linear relationships between current sensor readings and meteorological variables. The model captured low-frequency trends but struggled to capture non-linear behaviour, strong noise or sudden spikes. This was expected as pollutant spikes are non-linear and highly dependent on multiple factors, such as weather events and chemical variables. This limitation highlights LR's role as a simple baseline model and indicates the need for more flexible non-linear algorithms as a reliable prediction approach for highly variable pollutant series.

The NN model is designed to learn complex, non-linear relationships between past pollutant values and future concentrations. This allows the NN model to better capture non-linear and unstable relationships. However, as the dataset is relatively small and noisy, the NN struggled to generalise and often overfitted fluctuations in the data. As a result, it performed worse than GB on most pollutants, even though in theory it can model non-linear behaviour. Similarly, MLP is not a standard time-series model, and is likely to underperform in such circumstances. With a larger dataset and cleaner data provided with stronger trends, the NN model would definitely increase in performance.

Classification Models

The GB model shows the closest trend to the actual class distribution across all horizons, as the iterative learning focuses on correcting large residual errors. However, this model tended to capture extreme points more than other models, making its nature being all-time low in the mid (1.5-2.5) section (**Figure 1.11**).

The Decision Tree model shows a trend overfitting in most of the pollutant classes. This is caused by it being a non-linear rule-based model and its high variance results in the model being very sensitive to extreme points. This shows that the Decision Tree model is not suitable for predicting concentrations in the Air Quality Dataset.

Alternatively, Random Forest is an ensemble of many decision trees, which allows an increase of stability by comparing multiple trees, resulting in more robust noise and complex relationships. This is reflected in 1-h, 6-h and 12-h horizons (**Figure 1.10**).

Logistic Regression as a simple baseline model assumes log-odds as a linear function of features. Its simplicity allows the model to conservatively predict, establishing decent accuracy across most horizons. However, the model struggles to capture fluctuations with the lack of non-linearity and flexibility.

Limitations & Future Improvements

As the training dataset was only one year, it was difficult to capture longer term trends such as seasonality, where such trends may exist. Future improvements include trying to gather data for a longer time period in order to improve model accuracy. Significant amounts of missing data points made it difficult to accurately model the true data. Future models would be more efficient if missing information was improved upon. Each pollutant and horizon is modelled separately, ignoring potential joint structure between pollutants that a multivariate model could exploit. Future models may try to use a multivariate model, to improve model accuracy.

Conclusion

This project demonstrated the importance of incorporating temporal preprocessing and feature engineering in modelling patterns for air-quality data. Anomaly and event detection helped identify pollutant spikes and sensor issues, strengthening data reliability. Air-quality prediction was investigated by using different regression and classification models. Among all the models, Gradient Boosting delivered the strongest average performance for both regression and classification. Its tree-based structure was well suited in capturing complex patterns and handling the non-linear, highly fluctuating behaviour of pollutant levels. Future attempts may try improving dataset modelling by improving which inputs are utilised to capture trends found in EDA, or comparing different models such as LSTM. Therefore, these findings highlight the value of flexible, non-linear methods for environmental data prediction.

References

Vito S, 2008. Air Quality [Dataset]. *UCI Machine Learning Repository,* accessed 10 November 2025, <https://archive.ics.uci.edu/dataset/360/air+quality>.

Appendix

See README.md for figure/script references

Figure 1.1

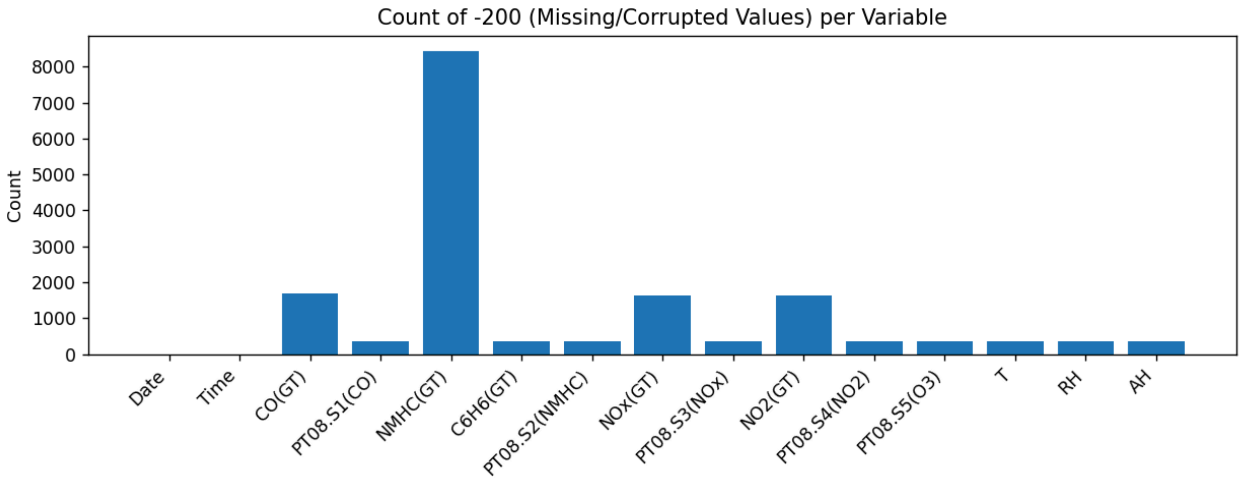


Figure 1.2

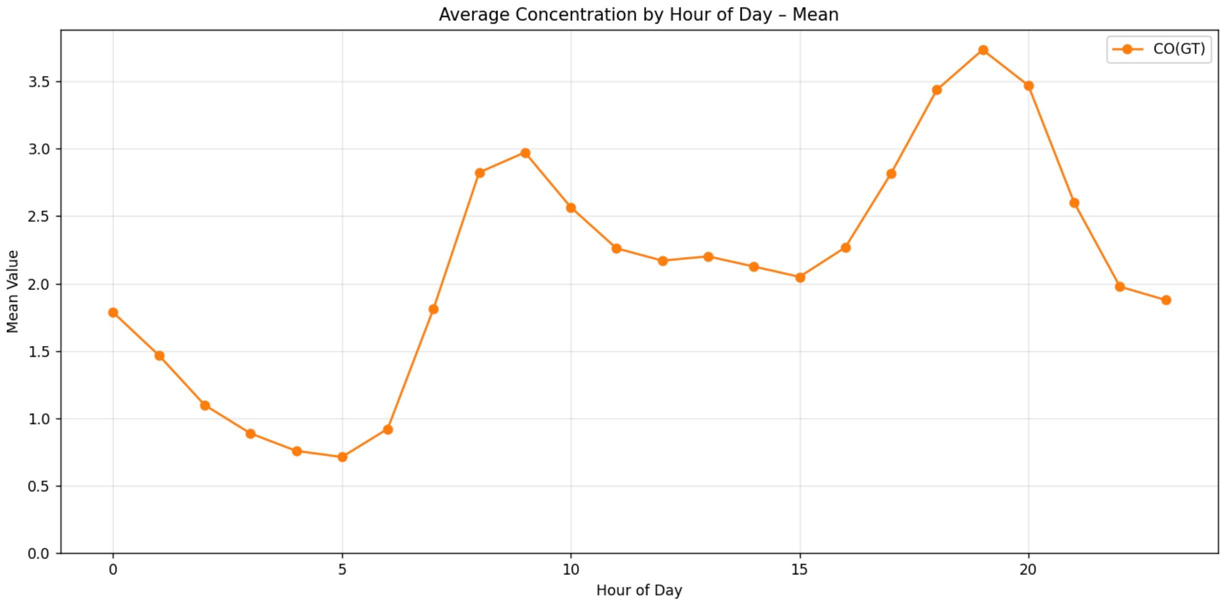


Figure 1.3

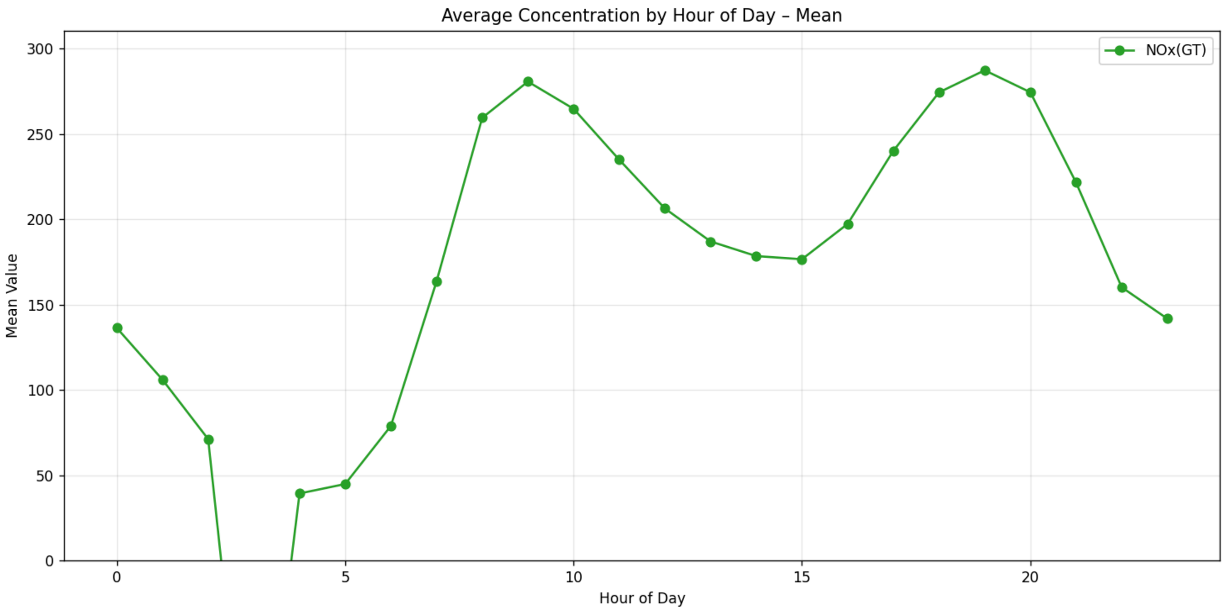


Figure 1.4

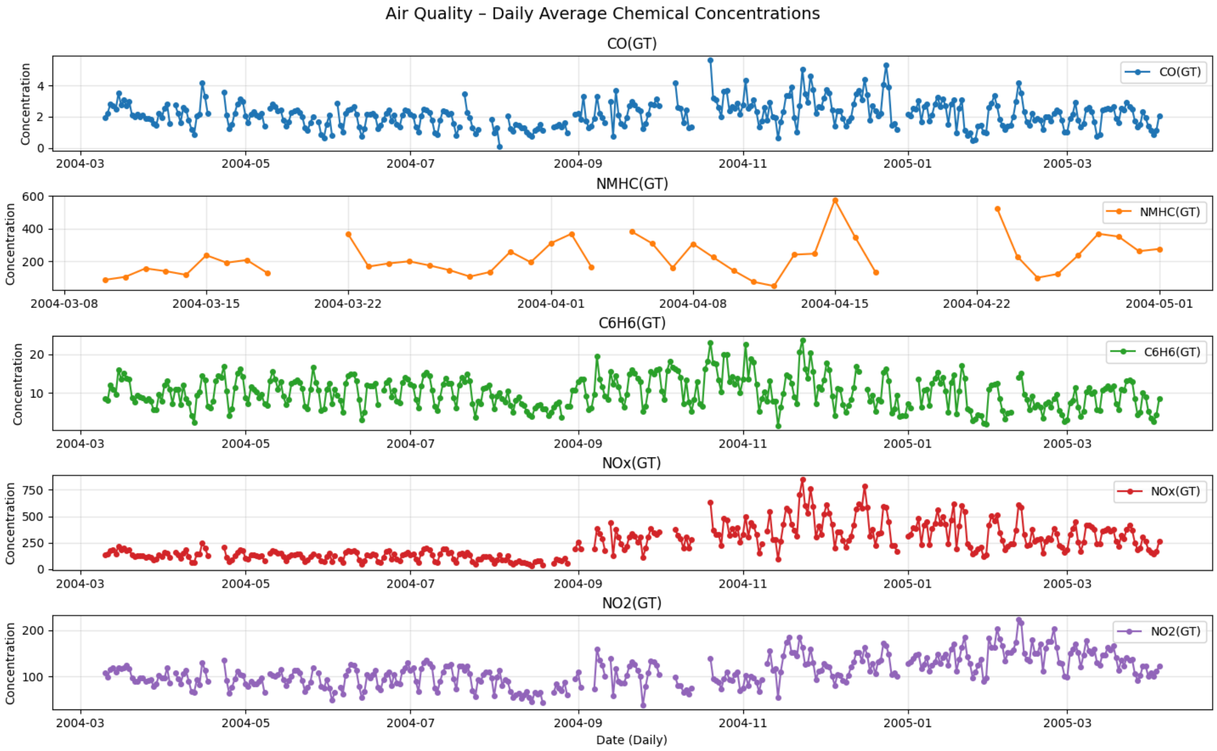


Figure 1.5

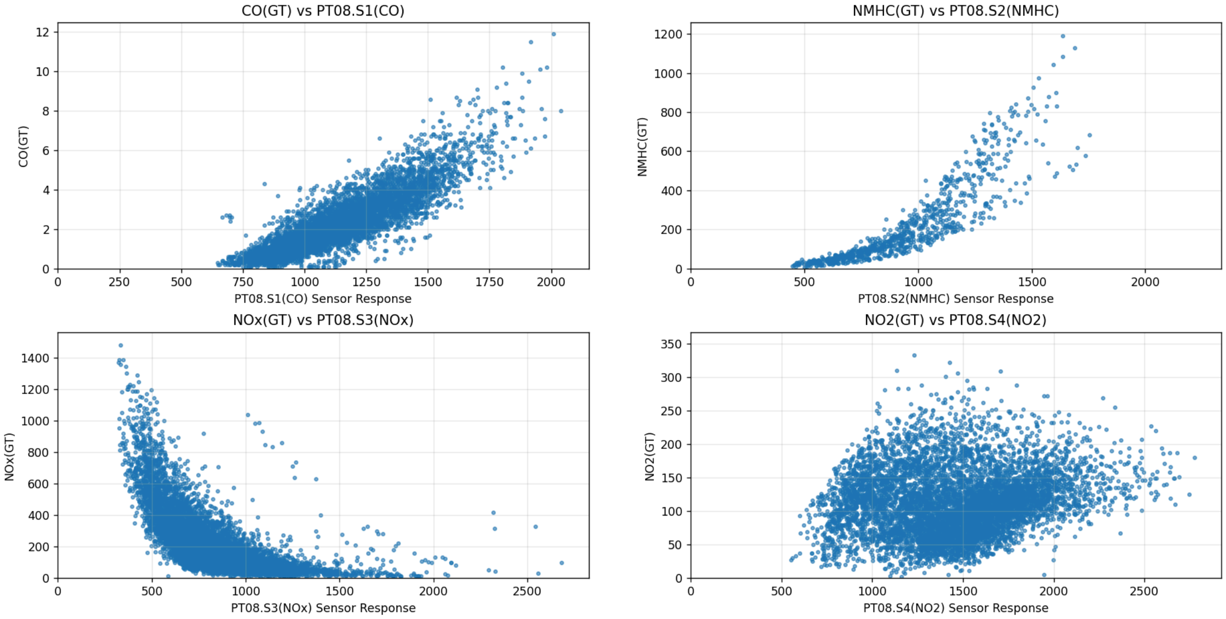


Figure 1.6

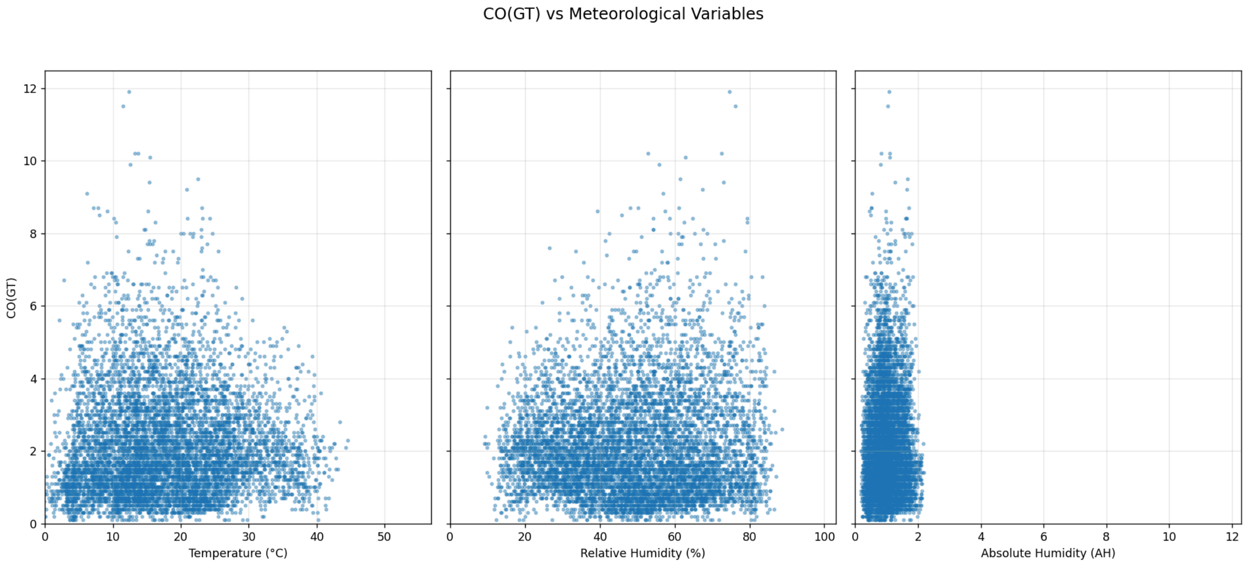


Figure 1.7

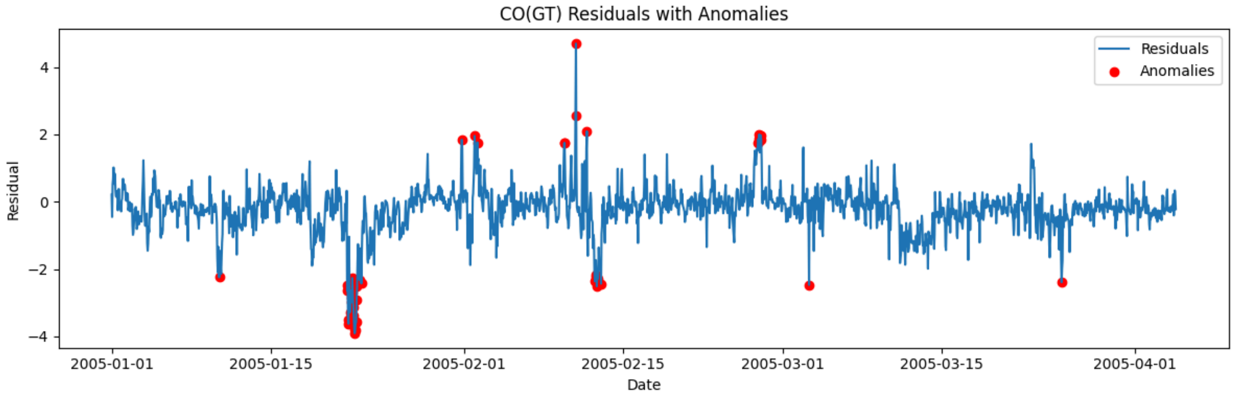


Figure 1.8

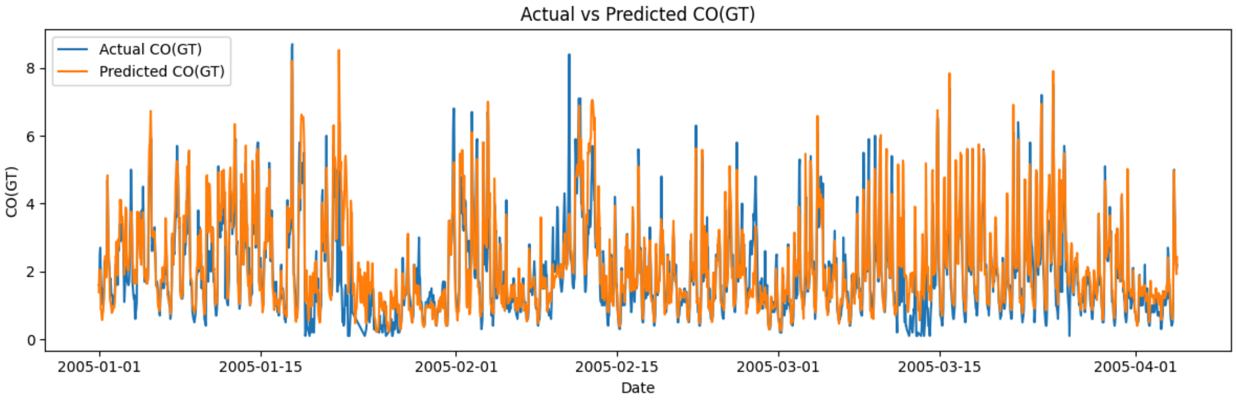


Figure 1.9

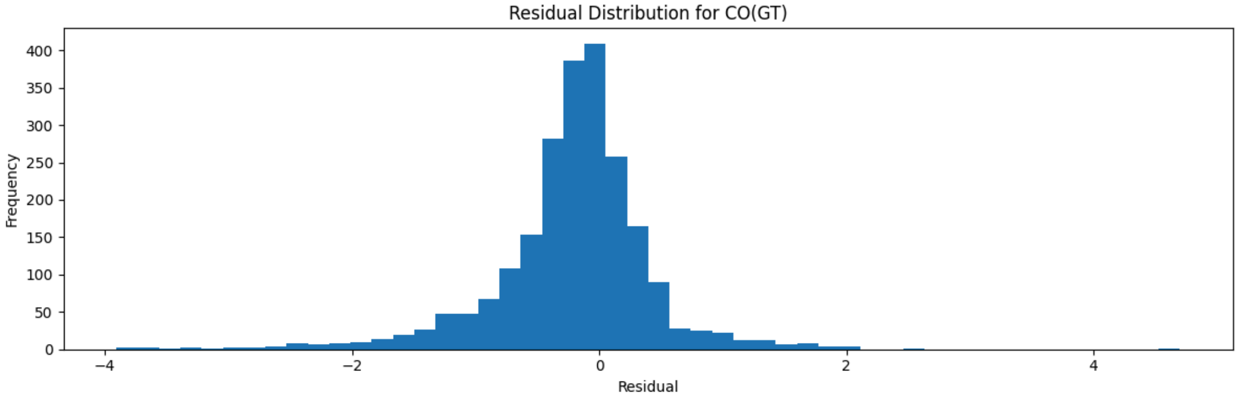


Figure 1.10

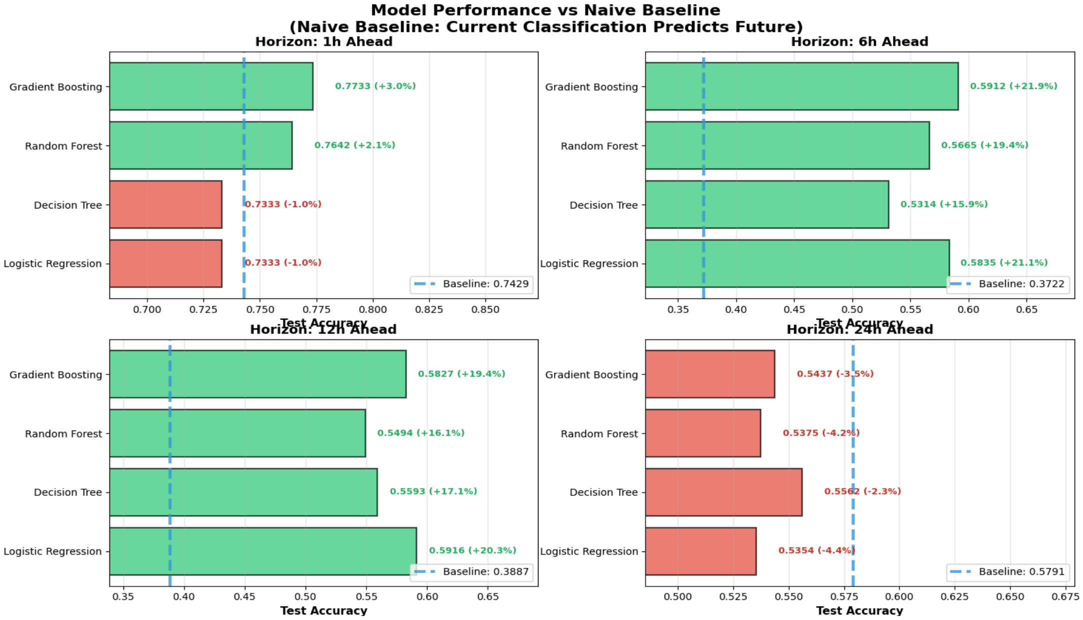


Figure 1.11

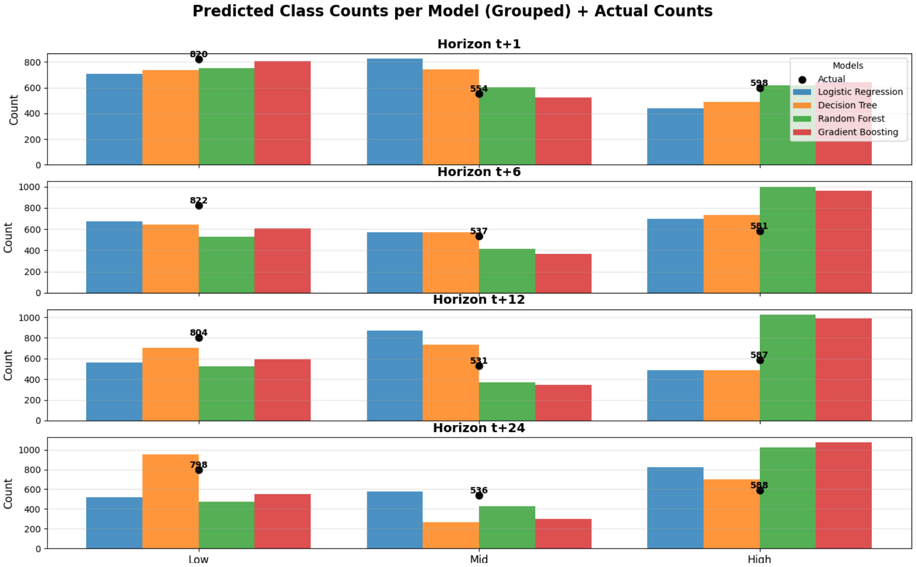


Table 1.1 - Summary of Feature Engineering

Feature Type	Description	Columns Created	Purpose in Time-Series Forecasting
Lag Features	Previous pollutant values shifted backward in time	For each pollutant we added lag up to 1, 3, 6, 12, 24 hours	Helps reduce noise by showing the dataset in previous hours.
Rolling Averages	Smooths short-term noise using moving windows	Rolling mean columns of 3, 12, 24	Provides smoothed trend information, helps model low-frequency variations, and stabilises noisy sensor behaviour.
Datetime Features	Extracted important date info for training.	Hour, Day of the week	Allows models to capture daily/ seasonal patterns
Sensor Features	Original dataset sensor	PT08.S1(CO), PT08.S2(NMHC), PT08.S3(NOx), PT08.S4(NO2), PT08.S5(O3)	Includes original sensor data.
Meteorological Features	Temperature and humidity variables	T, RH, AH	Capture environmental effects discussed above
Missing Value Handling	Replaces corrupted values coded as –200	All numeric columns	Ensures models are trained on meaningful data and prevents bias from invalid sensor values.
Median Imputation + Scaling	Applied after feature creation	All numeric predictors	Reduces the impact of extreme outliers and allows LR/NN to train on numerically stable inputs.

Table 1.2 - RSME Comparison Between Five Pollutants Across Horizons

		RMSE (4 d.p.)		
Pollutant	Horizon (hr)	LR	GB	NN
CO(GT)	1	0.7017	0.6605	0.7295
	6	1.5962	1.2675	1.4178
	12	1.1033	1.3251	1.4215
	24	1.2726	1.2407	1.4183
NMHC(GT)	1	120.3678	112.0068	124.7625
	6	167.1915	149.3373	155.9538
	12	152.1716	166.0420	146.8038
	24	190.0539	180.6025	197.5809
C6H6(GT)	1	3.7318	3.3943	3.5005
	6	7.7141	6.0547	7.8659
	12	5.5672	6.5988	9.4987
	24	6.0571	6.8764	7.7720
NOx(GT)	1	100.3824	93.1264	123.1276
	6	185.5127	182.6310	186.2007
	12	198.6119	184.4776	191.8430
	24	187.9352	183.8932	227.8393
NO2(GT)	1	25.7970	21.9215	30.4204
	6	45.2181	37.2085	43.3200
	12	45.2996	37.9415	46.6679
	24	43.7479	39.8063	45.8401