AI7101 - Machine Learning with Python

African Air Quality Prediction

Khoi Minh Ho **Tung Thanh Do** **Truong Quang Vu**

Contents

Problem Description: A Regression Problem

The Problem

$PM_{2.5}$ pollution threatens public health across Africa

Dense sensor networks too costly for most cities

The Solution

Machine learning + satellite data = city-wide monitoring

Sentinel-5P fills spatial gaps in ground sensors

The Impact

Real-time air quality data for underserved communities

Supports health interventions & policy decisions

Target Cities

- Lagos & Accra (West Africa)
- Nairobi & Kampala (East Africa)
- Yaoundé & Bujumbura (Central)
- Kisumu & Gulu (Regional hubs)

Key Features

- Aerosol optical depth (AOD)
- NO_2 & ozone from Sentinel-5P
- Meteorological variables
- Ground sensor validation data

Performance Evaluation

RMSE on held-out locations & timepoints

Exploratory Data Analysis

Features:

Metadata

- Site identification & coordinates
- City location data
- Measurement timestamps

Multiple sites per city for comprehensive coverage

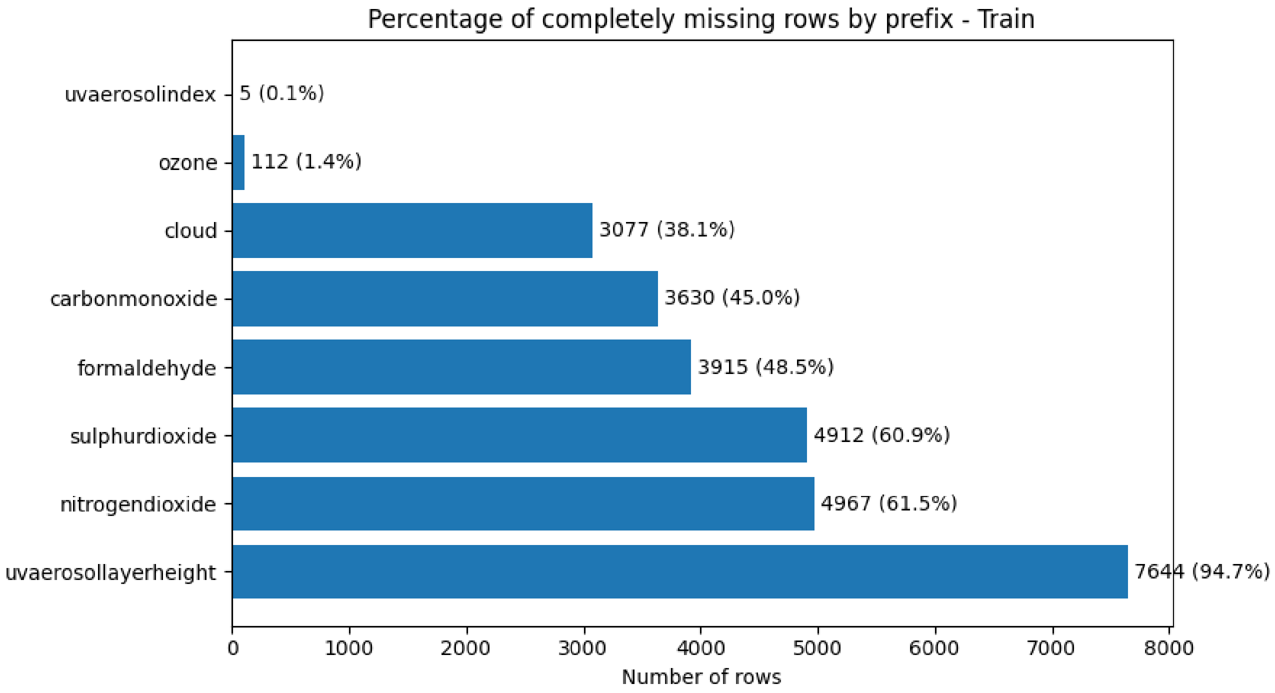
Measures (SO2, CO, NO2, HCHO, O3, UV, Cloud)

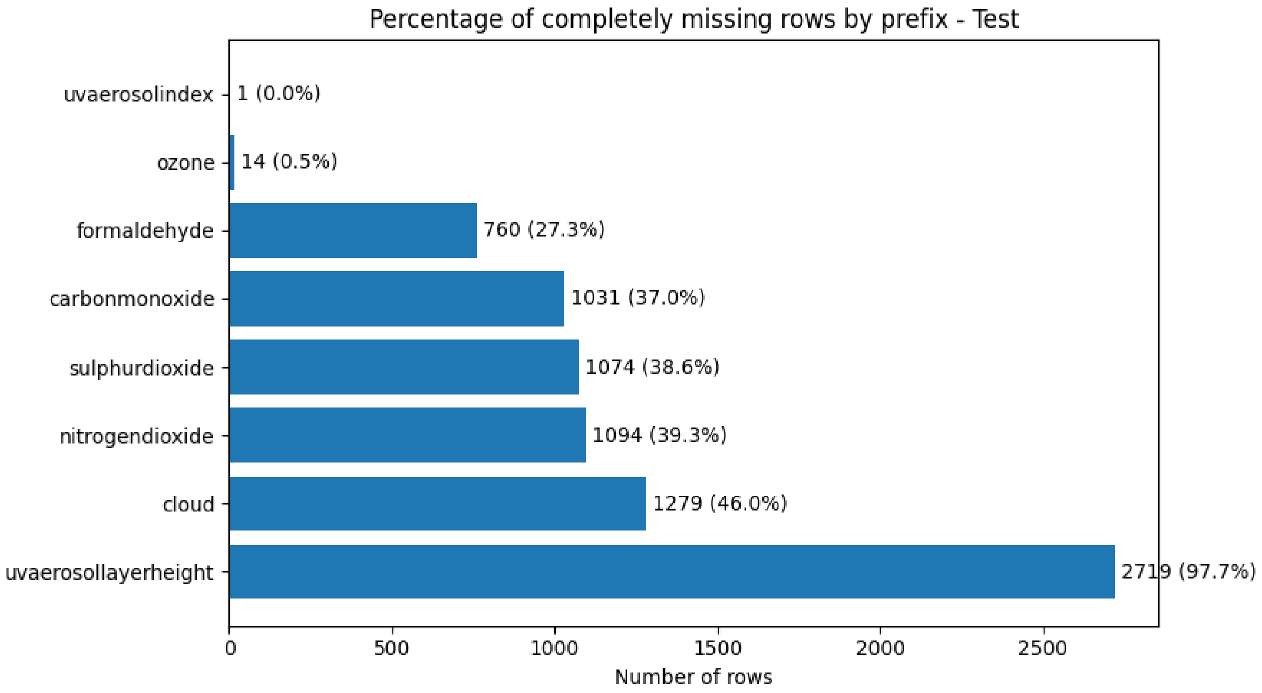
- Density
- Cloud fraction
- Spherical angle

Exploratory Data Analysis

Null values:

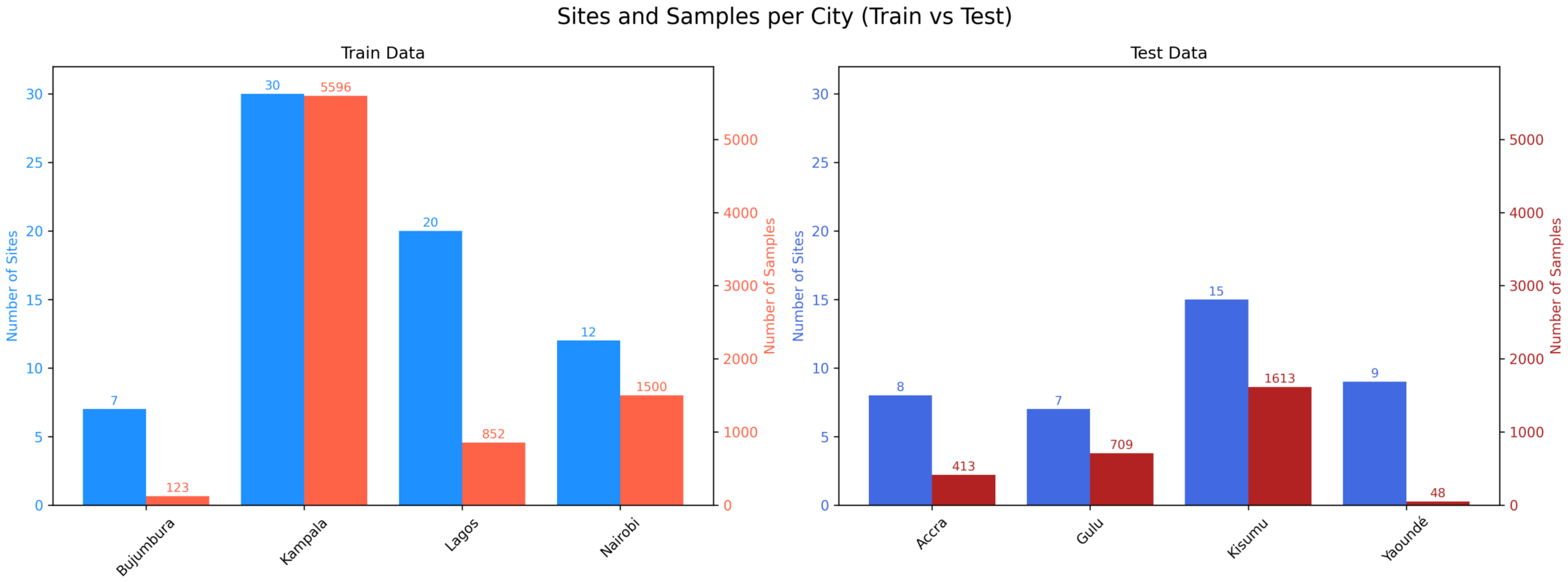
- “Metadata”: 0 nulls
- “Measures”: For any row, *all* attributes of a measure either has a value or is null





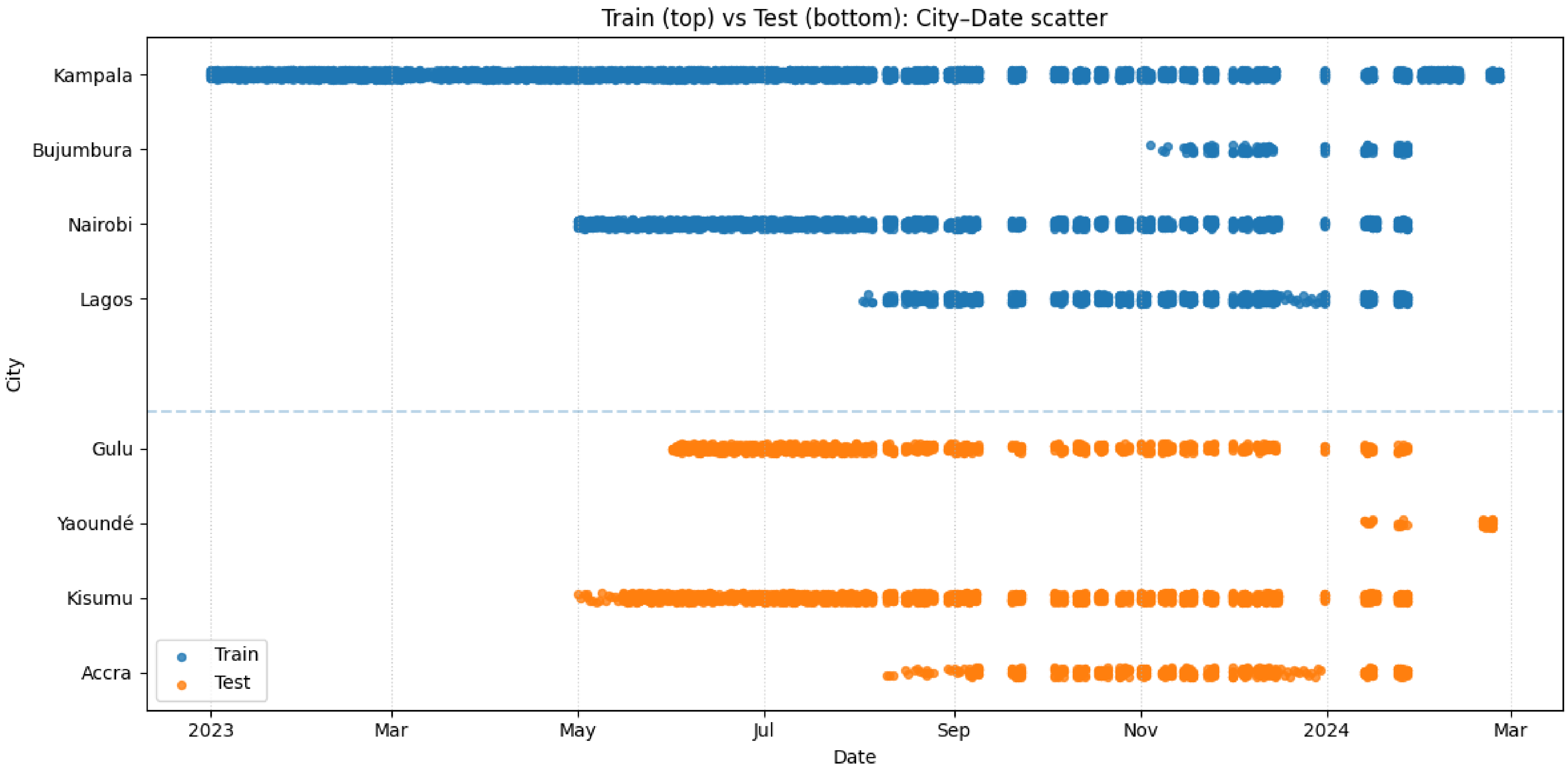
Exploratory Data Analysis

Finding 1: Unequal number of samples/measurement sites



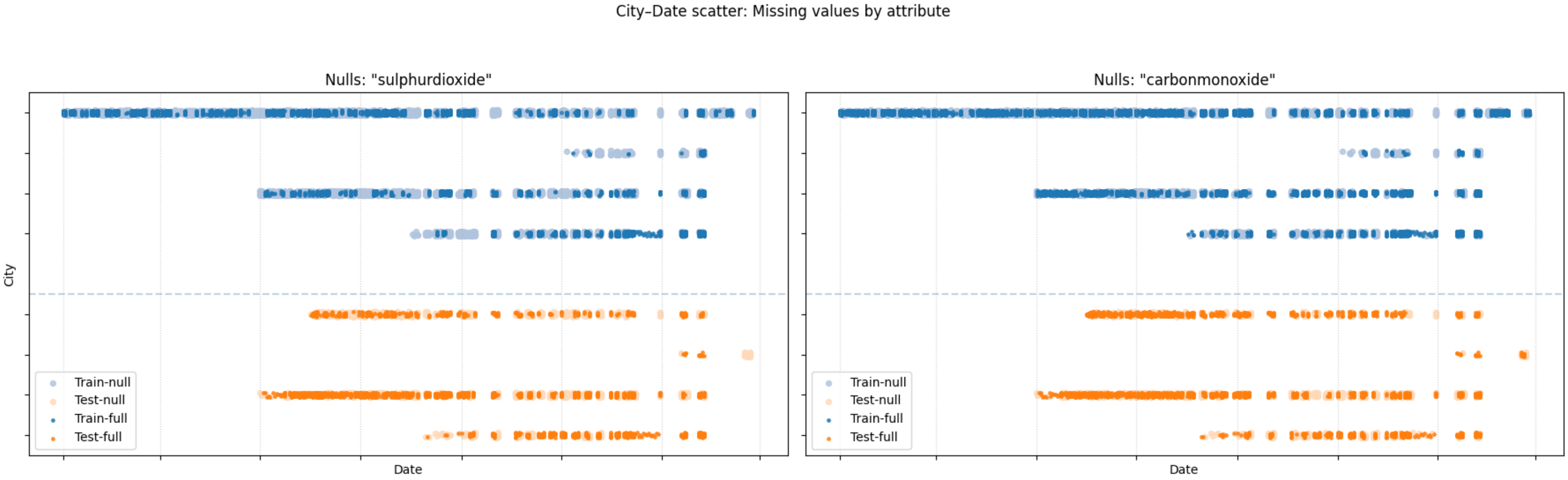
Exploratory Data Analysis

Finding 2: Timestamps are not evenly or equally distributed.



Exploratory Data Analysis

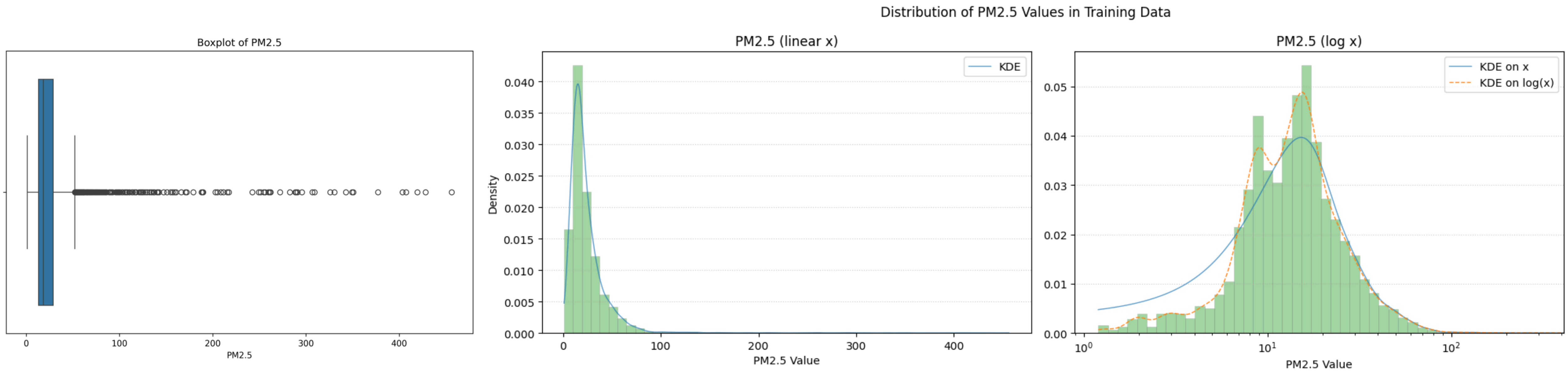
Finding 3: Null values are assigned randomly, unrelated to city/date



Exploratory Data Analysis

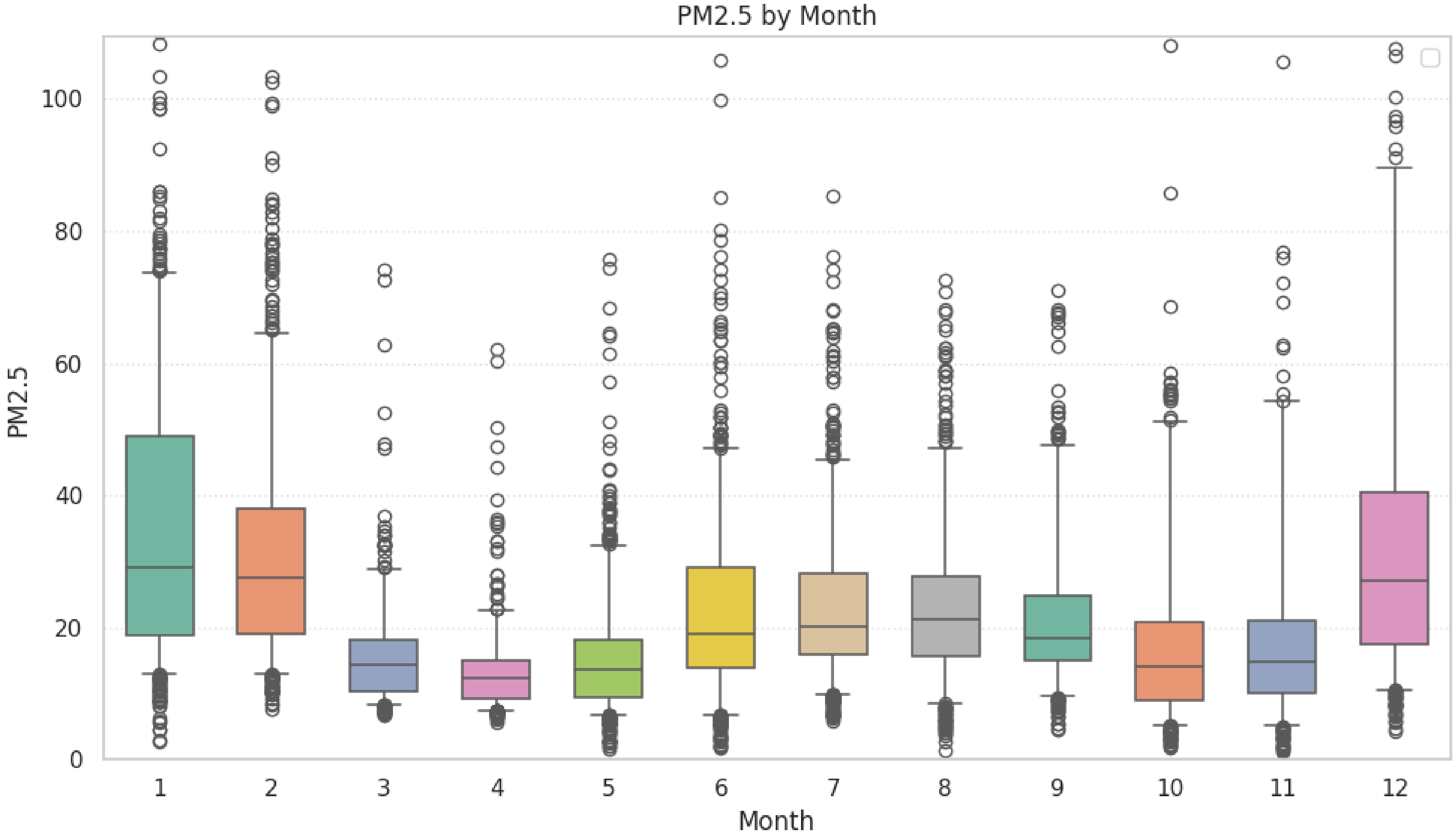
Finding 4: The target variable is heavily skewed and has a noticeable amount of outliers

Solution: Clip the PM2.5 to a maximum value before training for a better stability *or* transform it to log scale.



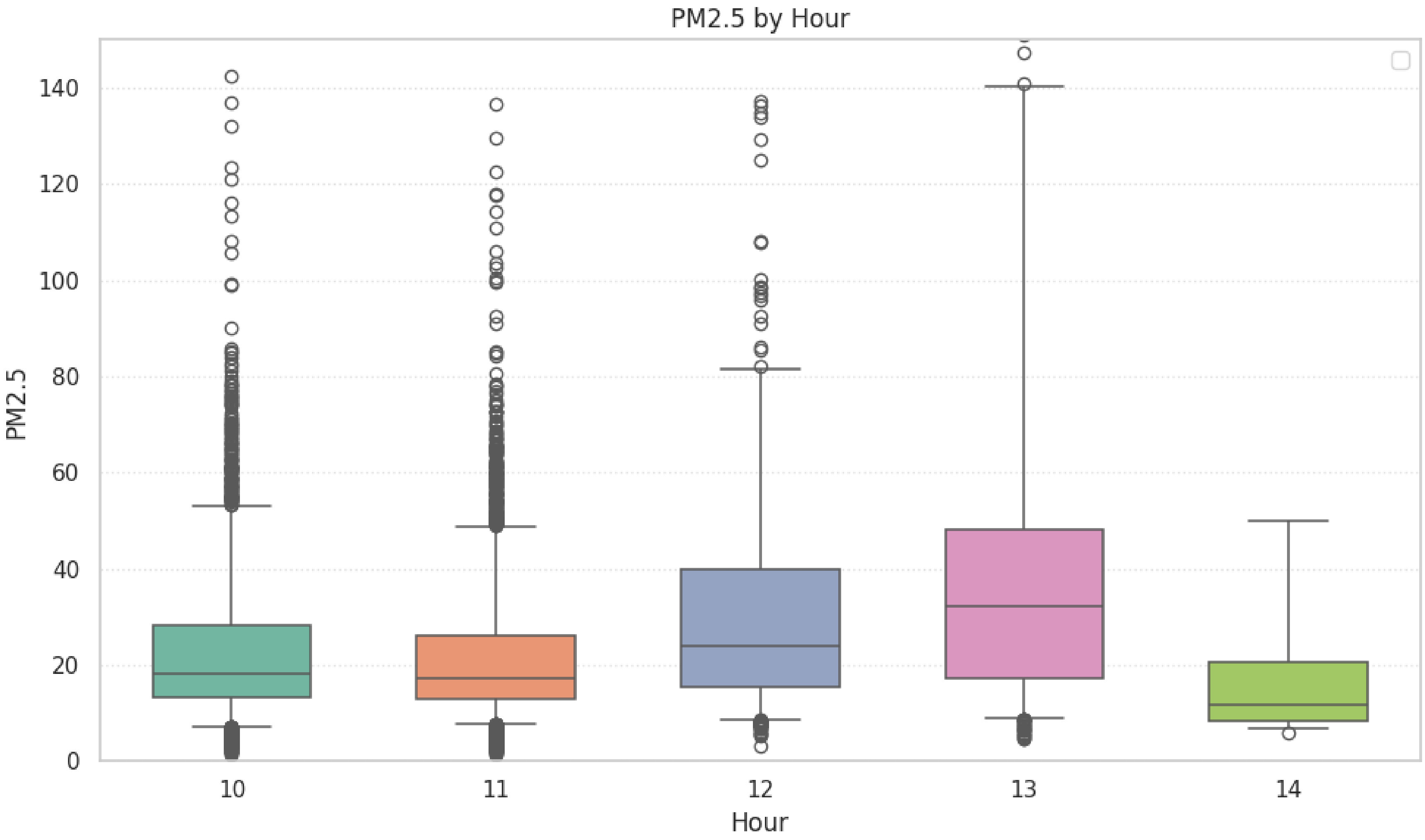
Exploratory Data Analysis

Finding 5: Time vs PM2.5: Higher PM2.5 during winter



Exploratory Data Analysis

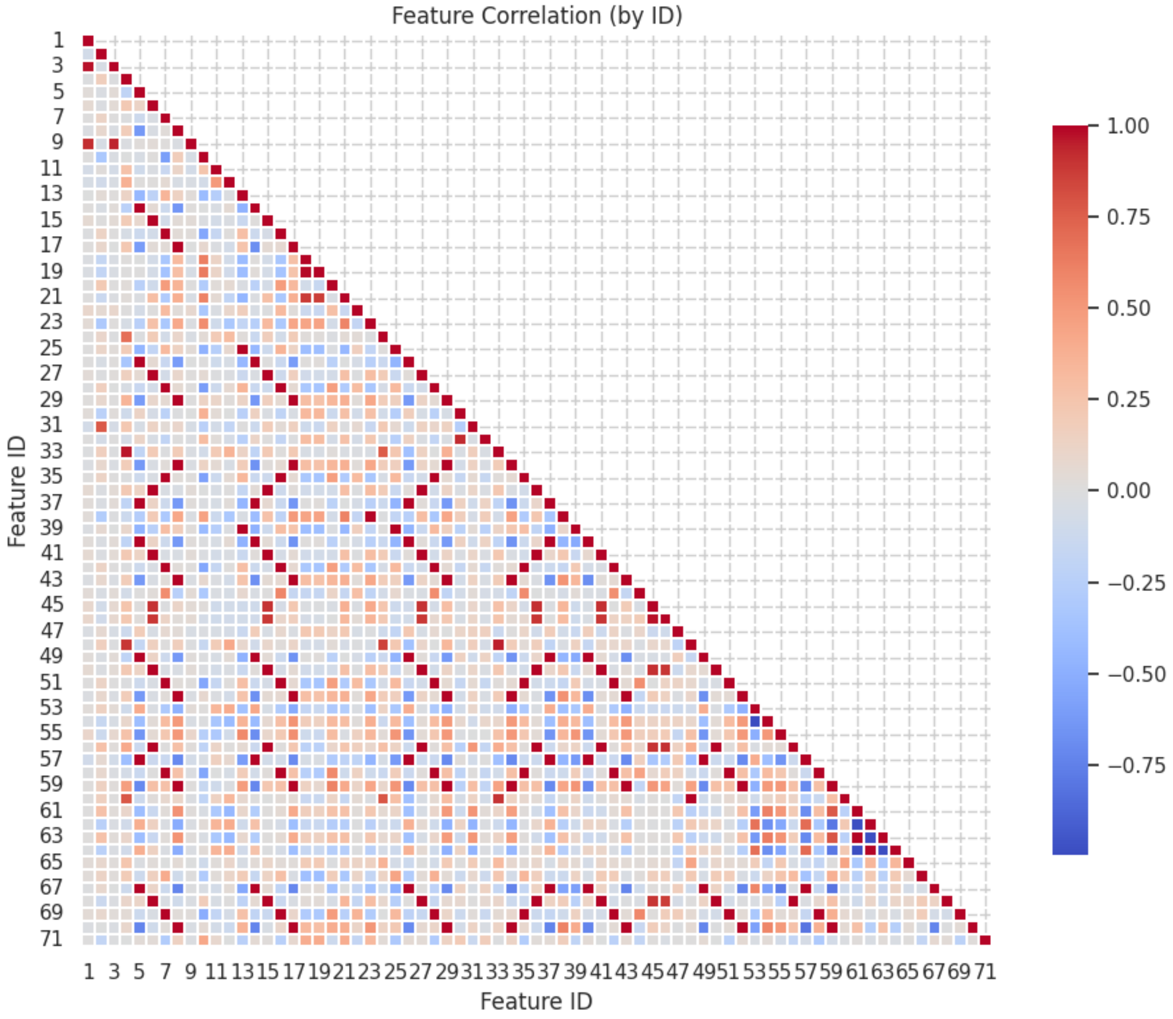
Finding 5: Time vs PM2.5: Higher & varied PM2.5 measured at noon (12-13)



Exploratory Data Analysis

Finding 6: Feature correlation

- We start with a rather unclear correlation heatmap
- Notice that there are some repeated patterns within the heatmap
- When printed out correlation values, many returns 1



Exploratory Data Analysis

Finding 6: Feature correlation

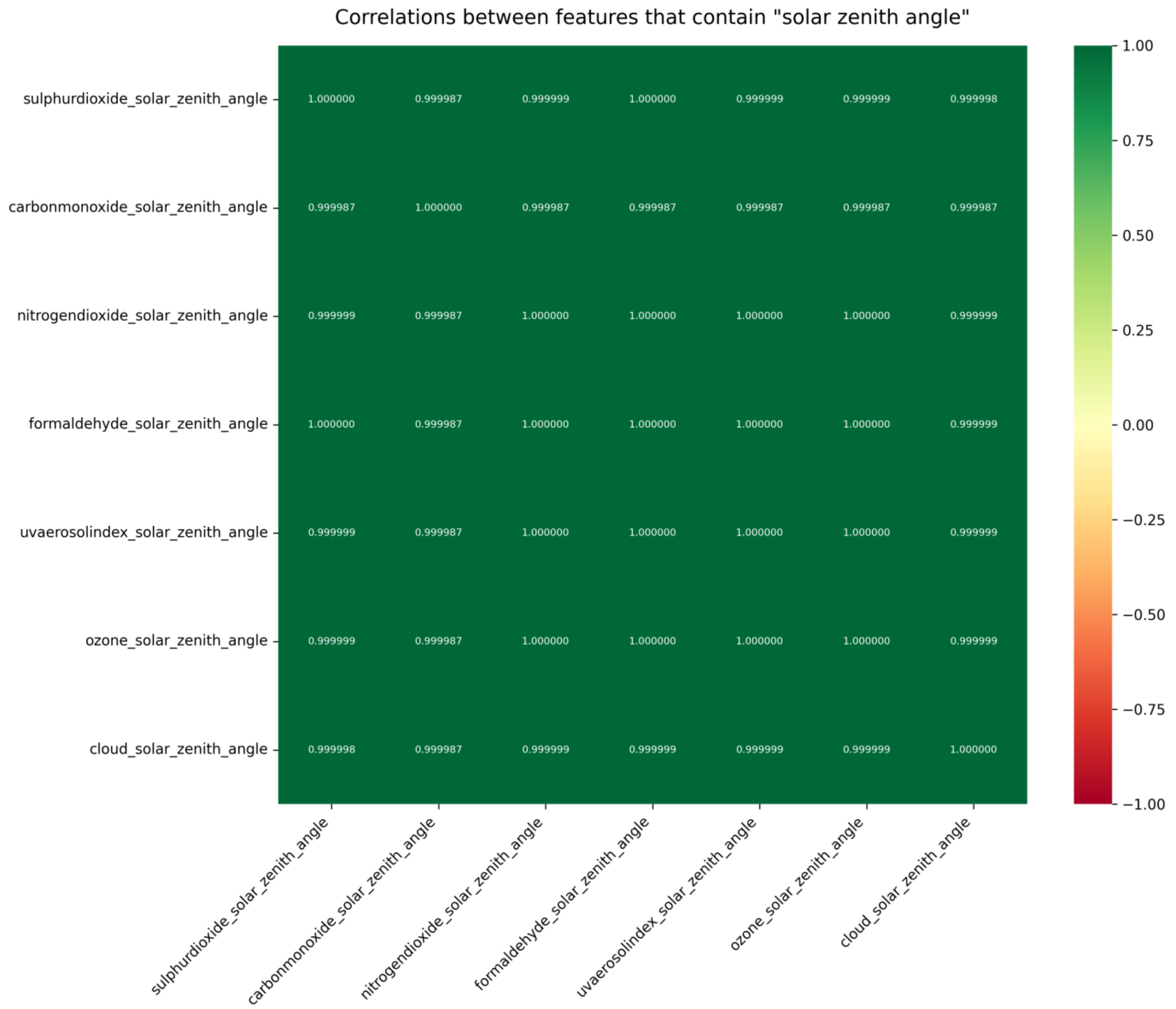
- **Perfect correlations found**

Many correlation values of exactly 1.0 detected

Investigation reveals approximately equal feature values

Key Insight

Feature reduction and null filling opportunity identified



Exploratory Data Analysis

Finding 6: Feature correlation

- **Perfect correlations found**

Many correlation values of exactly 1.0 detected

Investigation reveals approximately equal feature values

Key Insight

Feature reduction and null filling opportunity identified

carbonmonoxide_solar_zenith_angle	nitrogendioxide_solar_zenith_angle	formaldehyde_solar_zenith_angle	uvaerosolindex_solar_zenith_angle	ozone_solar_zenith_angle
NaN	NaN	NaN	33.745914	33.745914
26.566997	NaN	26.525513	26.525513	26.525513
NaN	NaN	NaN	41.898113	41.898113
NaN	NaN	NaN	43.923038	43.923038
40.144183	40.167336	40.167336	40.167336	40.167336
…	…	…	…	…

Exploratory Data Analysis

Finding 6: Feature correlation

- **Weak feature relationships**

Final feature set shows minimal correlation to target variable

Top 5 features: Pollutant measurements contributing to PM2.5 levels

abs_corr	feature
0.422418	**carbonmonoxide_co_column_number_density**
0.403459	**nitrogendioxide_tropospheric_no2_column_number_density**
0.398677	**nitrogendioxide_no2_column_number_density**
0.395134	**nitrogendioxide_no2_slant_column_number_density**
0.327197	**nitrogendioxide_absorbing_aerosol_index**
0.227742	solar_azimuth_angle
0.199219	formaldehyde_tropospheric_hcho_column_number_density
0.199143	altitude
0.185608	sulphurdioxide_so2_column_number_density_amf
0.182241	ozone_o3_column_number_density
0.179090	formaldehyde_hcho_slant_column_number_density
0.174712	cloud_surface_albedo
0.171257	nitrogendioxide_tropopause_pressure
0.166570	uvaerosolindex_absorbing_aerosol_index

Exploratory Data Analysis

Key takeaways:

- **Data Quality Issues**

  Purposeful random nullification detected across dataset

- **Imbalanced Distribution**

  Uneven data collection between stations and cities

- **Target Variable Skew**

  Log-scaled output transformation or clipping recommended

- **Categorical Limitations**

  City/station info unusable with different test distribution

- **Time Series Constraints**

  Missing timeframes prevent temporal analysis & not suitable for test data

- **Feature Correlations**

  Perfect feature correlations and weak feature-target correlation found and can be utilized to perform correlation-based feature selection

Feature Engineering

01

Location Features

Create composite location identifiers from coordinates

02

Temporal Features

Extract meaningful date and time components

03

Missing Value Treatment

Apply forward/backward fill with fallback strategies

04

Categorical Encoding

Transform categorical variables to numeric format

Step 1: Location Features

Create Location Composite

Generate unique location identifiers by combining latitude and longitude coordinates as strings.

Formula: `site_latitude + '_' + site_longitude`

We treat each pair as a **category**.

Step 2: Temporal Features

Date Components

- Month: `dt.month`
- Week: `dt.isocalendar().week`
- Day: `dt.day`
- Day of week: `dt.dayofweek`

Weekend Indicator

Binary feature identifying weekends using `dayofweek.isin([5,6])`

Saturday = 5, Sunday = 6 in pandas datetime

Step 3: Missing Value Treatment

Three-tier approach for handling missing numerical data:

1 **Forward Fill**

Use previous valid observation within city-location groups

2 **Backward Fill**

Use next valid observation if forward fill fails

3 **Global Median**

Fallback to **overall** median for remaining missing values

Date	Location	Raw	After ffill+bfill
2023-01-01	L1	NaN	12
2023-01-02	L1	12	12
2023-01-03	L1	NaN	12
2023-01-04	L1	18	18
2023-01-05	L1	17	17
2023-01-06	L1	NaN	17
2023-01-07	L1	25	25

Grouped by city and location to preserve local patterns in the data.

Step 4: Categorical Encoding

Feature Types

Categorical Columns

Use `select_dtypes(include='object')` to identify string-based features

Exclude metadata columns: date, id, city, country

Numerical Columns

Use `select_dtypes(exclude='object')` for numeric features

Remove target, folds, and coordinate columns from processing list

Label Encoder Usage

Apply `LabelEncoder` from scikit-learn to convert **categorical features** to integers.

Includes date column after datetime conversion for temporal ordering.

```
for col in categorical_cols + ['date']:
    data[col] = le.fit_transform(data[col])
```

Cross-Validation Procedure

Group K-Fold CV

Uses `GroupKFold` from `scikit-learn`

City-Based Groups

Groups defined by city column

No Data Leakage

Same city never in both train/validation in each fold

Fold	Training				Inference
	Kampala	Nairobi	Lagos	Bujumbura	
1	×	✓	✓	✓	Kampala
2	✓	×	✓	✓	Nairobi
3	✓	✓	×	✓	Lagos
4	✓	✓	✓	×	Bujumbura
Test	✓	✓	✓	✓	Test set

1

Split

Split the original training set into training and validation set of each fold

2

Feature Selection

Perform feature selection on the training set of each fold

3

Train & Evaluate

Train the model and predict on validation set of each fold to obtain fold-level RMSE

Report Mean RMSE

Average across all fold-level RMSE values

Feature selection

Two-stage pipeline

Stage 1

Top-K Feature Selection

Utilizes **permutation feature importance** with CatBoost as the base model to select the **top K** most predictive features.

Stage 2

Correlation Filtering

Removes redundant features by analyzing the Pearson correlation matrix, minimizing multicollinearity for better model stability.

Modelling

Model Ensemble

Tree-Based Gradient Boosting

- LightGBM
- XGBoost
- CatBoost

Linear Model

- Lasso Regression

Kernel-Based Model

- Support Vector Regressor

Ensemble Benefits

- Captures diverse data relationships
- Reduces model bias
- Improves generalization

Hyperparameter Search

1. Optimization Strategy

Objective Function

Minimize mean RMSE across GroupKFold splits

Search Strategy

Bayesian optimization in predefined ranges of hyperparameters in **N** trials

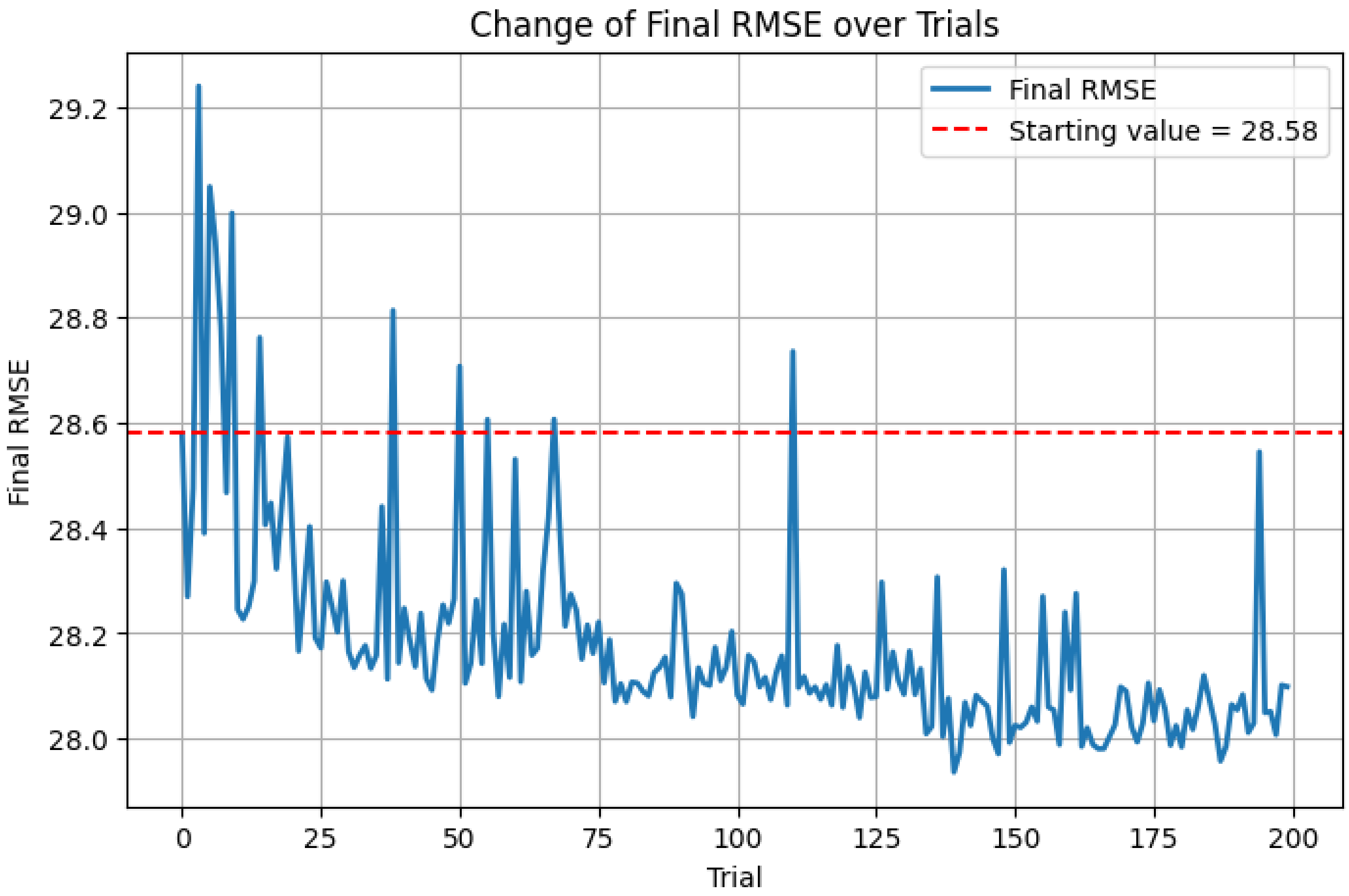
2. Model Parameters Tuned

- **CatBoost:** learning rate, depth
- **LightGBM:** learning rate, max depth
- **XGBoost:** learning rate, max depth
- **Lasso:** alpha
- **SVR:** C, epsilon

Best models retrained on full training data after optimization

Results

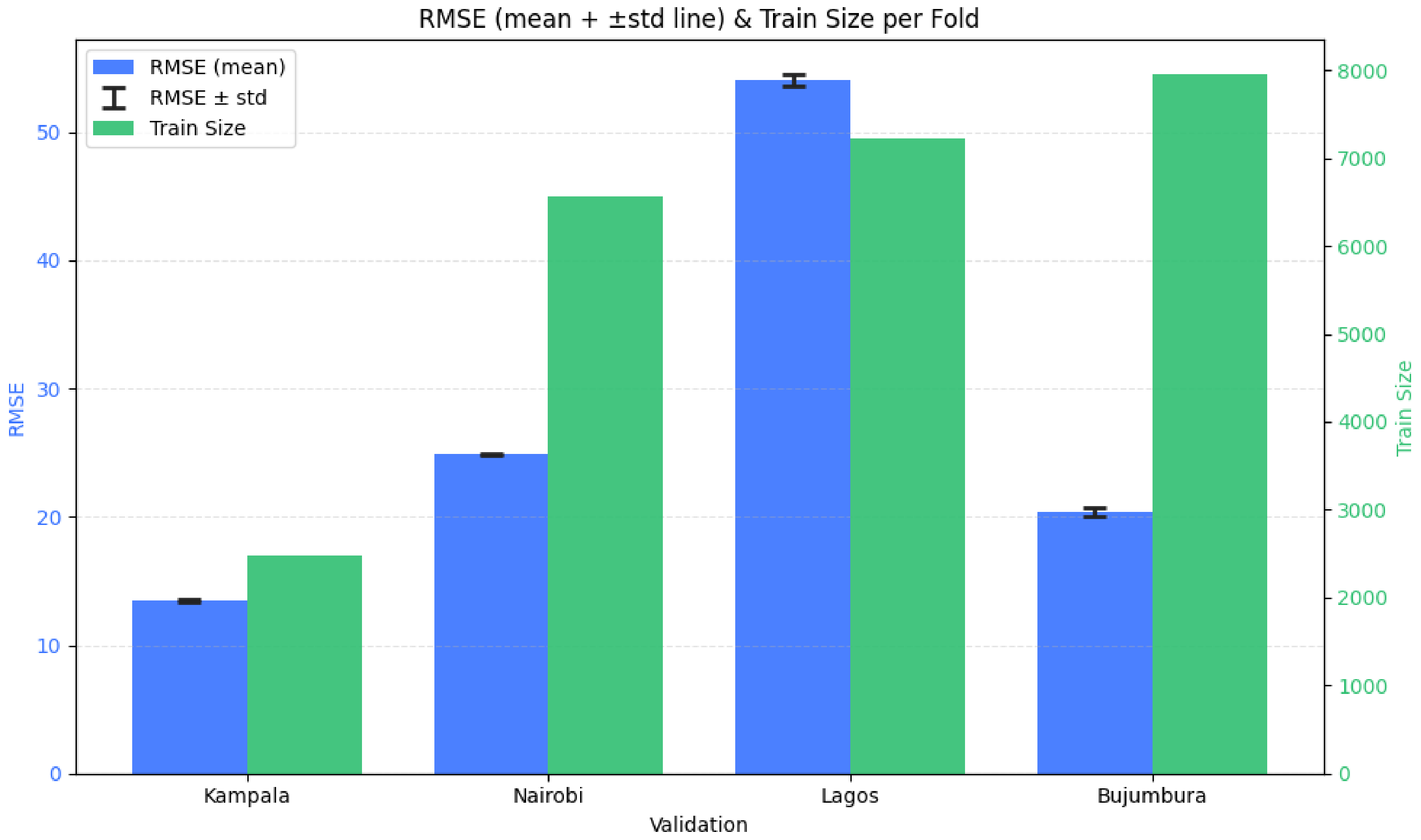
Effectiveness of hyperparameter search



Results

Cross-validation results

- **Extreme outliers** in Lagos is a pain when validated on
- Yet, having trained on it make more robust (significantly lower RMSE) => *Data quality > quantity*



Results

Ensemble Model

- **Model:** SVR, CatBoost, LightGBM, XGBoost, Lasso

Ensemble model					Private score	Public score
SVR	CatBoost	LightGBoost	XGBoost	Lasso		
✓					19.394	15.866
✓	✓				18.159	14.254
✓	✓	✓			18.066	14.233
✓	✓	✓	✓		**17.892**	**14.101**
✓	✓	✓	✓	✓	18.099	14.297

- We use the setting (SVR, CatBoost, LightGBoost, XGBoost) for any subsequent experiments.

Results

Feature Selection Strategy

- **CatBoost:** feature importance from the trained CatBoost model
- **ANOVA:** perform ANOVA F-test to select feature
- **Lasso:** drop zero coefficient in the weights
- **Permutation:** shuffle one feature at a time and measure drop in model performance.

Feature selection	Private score	Public score
Baseline	17.892	14.101
Catboost	17.987	14.074
ANOVA	17.931	14.196
LASSO	18.250	14.454
Permutation	**17.855**	**14.068**

- We use Permutation strategy for any subsequent experiments.

Results

Feature Selection Strategy

- For the number of top feature in Feature Selection, we choose 40 as its high result on private score.

Num top features	Private score	Public score
20	18.009	**14.026**
40	**17.855**	14.068
60	17.883	14.132

Results

Feature Selection Strategy

- The model performs the best when we do not remove any features based on correlation threshold.
- Permutation strategy has already removed all high correlation features.

Correlation threshold	Private score	Public score
0.75	17.957	14.081
0.90	17.921	14.094
1.00	**17.855**	**14.068**

Results

Feature Engineering Strategy

- **Location feature:** use location information (longitude and latitude) to make a prediction.
- **Temporal feature:** instead of taking raw date data, preprocess into useful information such as (month, day, day of week,...)
- **Missing value treatment:** perform filling null and feature unification by majority voting
- **Cloud pressure:** use the difference between cloud pressure at top and base altitude.

Feature Engineering	Private score	Public score
Baseline	17.855	14.068
+ drop location	18.332	14.594
+ augment date	17.828	14.035
+ unify	18.084	14.359
+ cloud pressure residual	17.944	14.179

Results

Target preprocessing

- **Log scale:** take the logarithm of target value
- **Clip:** clamp the maximum value of target in 97% quantile *(~PM2.5 = 65, in WHO's warning zone)*
- Based on the result, the model performs best with clipping technique.

Target preprocessing		Private score	Public score
Log Scale	Clip		
-	-	17.828	14.035
-	✓	17.770	13.813
✓	-	18.031	14.172
✓	✓	18.188	14.270

Results

Final model

- **Model:** SVR, CatBoost, LightGBoost, XGBoost
- **Target preprocessing:** clamp the maximum value of target in 97% quantile
- **Feature engineering:** augment location + augment date
- **Feature selection strategy:** top 40 features with Permutation strategy

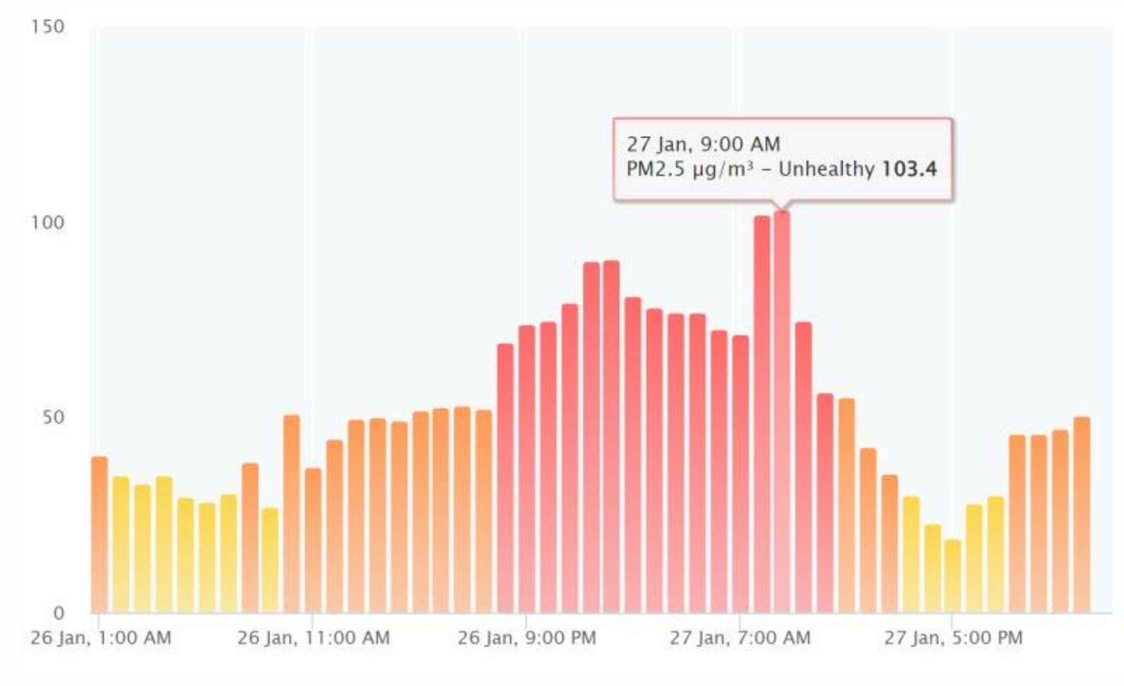
The final score: 13.813 (public score) - 17.770 (private score)

Conclusion

- **ML takeaways:** A simple ensemble with permutation-selected features, target clipping and null filling was most robust; extra correlation pruning added little value.

- **Applicability & impact:** Delivers city-wide estimates from satellite + sparse sensors for operational monitoring, public alerts, and policy

- **Future works:**
  - Data collection for high-PM regions (e.g. Abu Dhabi)
  - Additional modelling (time series)



Contributions

- **Khoi Minh Ho:** Problem formulation, EDA, cross-validation results

- **Truong Quang Vu:** Feature engineering, modelling, cross-validation

- **Tung Thanh Do:** Experiment, ablation study, conclusion