

Supplementary material

Multi-Machine Learning Approaches to Modeling Small-Scale Source Attribution of Ozone Formation

5 Zheng Xiao^{1,2}, Yifeng Lu³, Guangli Xiu^{1,2}

¹State Environmental Protection Key Lab of Environmental Risk Assessment and Control on Chemical Processes, School of Resources & Environmental Engineering, East China University of Science and Technology, Shanghai 200237, PR China

²Shanghai Environmental Protection Key Laboratory for environmental standard and risk management of chemical pollutants, School of Resources & Environmental Engineering, East China University of Science and Technology, Shanghai 200237, PR
10 China

³Shanghai Chemical Industry Park Administration Committee, Shanghai 201507, P.R. China

Correspondence to: Guangli Xiu (xiugl@ecust.edu.cn)

This supporting information contains the following content (Page 1-Page 8):

- (1) Text S1. Introduction to the models and methods, mainly including the machine learning (ML) models, SHAP algorithm;
15 (2) Fig. S1. The modeling results of different ML models on the test and train dataset; (3) Fig. S2. Feature importance analysis of different ML models; (4) Table. S1. Abbreviations and categories of the monitoring stations and information of the instruments.

S1. Text:

S1.1. ML Models

20 **S1.1.1. Decision Tree Regression (DTR)**

Decision Tree Regression (DTR) is a decision tree-based regression method that makes predictions by building a decision tree (Loh, 2011). In the decision tree, each node represents a feature and each leaf node represents a predicted value. As a predictive modeling technique, DTR uses decision tree structure to simulate the relationship between input features and target variables.

The general form of a decision tree can be represented by the following set of recursive segmentation Eq. (1):

$$25 \quad f(x) = \sum_{m=1}^M c_m \cdot I(x \in R_m) \quad (1)$$

where $f(x)$ is the predicted value for the input x , M is the number of terminal node (leaves) in the decision tree, c_m is the predicted output for all instances that belong to region R_r , and $I(\cdot)$ is the indicator function, which equal to 1 if the condition is true (i.e., if x belongs to region R_m) and 0 otherwise.

S1.1.2. Random Forest Regression (RF)

30 RF is an ensemble learning technique that constructs a multitude of decision trees during training and outputs the mean prediction of the individual trees for regression tasks (Breiman, 2001). In this method, each tree is grown by using a randomly selected subset of independent variables. The overall prediction \hat{y} of the RF model can be expressed mathematically as follows:

$$\hat{y} = \frac{1}{N} \sum_{i=1}^N T_i(x) \quad (2)$$

where \hat{y} is the predicted value for the input feature vector x , N is the total number of decision trees in the forest, and $T_i(x)$ is
 35 the prediction made by the i -th decision tree for the input x .

S1.1.3. Support Vector Regression (SVR)

SVR is a regression technique derived from Support Vector Machines (SVM) that aims to find a function that deviates from the actual target values y_i by a value no greater than a specified margin ϵ . The optimization problem for SVR can be expressed as Eq. (3):

$$40 \min_{w,b,\xi} \left(\frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i \right) \quad (3)$$

subject to the constraints:

$$y_i - (w^T \phi(x_i) + b) \leq \epsilon + \xi_i \quad (4)$$

$$(w^T \phi(x_i) + b) - y_i \leq \epsilon + \xi_i \quad (5)$$

In this framework, w represents the coefficients assigned to the input features, which determine the orientation of the
 45 hyperplane used for prediction. The bias term b adjusts the position of the hyperplane along the response axis. The slack variables ξ_i account for deviations from the ϵ tube around the predicted values, allowing flexibility in fitting the model to the training data. The parameter C controls the importance of these slack variables, with a larger C emphasizing the reduction of errors over achieving a wider margin. The mapping function $\phi(x_i)$ enables SVR to capture complex relationships by projecting

data into a higher-dimensional space, facilitating linear separation when the original data space is non-linear. Finally, ϵ defines
50 the threshold for tolerance, determining how much error is acceptable for the model.

S1.1.4. XGBoost Model

The XGBoost model represents an advanced version of the gradient boosting decision tree algorithm, capable of rapidly
constructing boosted trees and effectively addressing both classification and regression tasks. In this research, we utilized the
regression functionality of XGBoost, which focuses on optimizing the objective function and implementing machine learning
55 algorithms within the gradient boosting framework. Notably, the algorithm incorporates L1 and L2 regularization terms, which
serve to mitigate overfitting and enhance generalization performance. The typical structure of the XGBoost objective function
encompasses two components: the training loss term and the regularization term. This can be expressed mathematically as
follows:

$$Tar(\theta) = L(\theta) + \varphi(\theta) \tag{6}$$

60 Here, $L(\theta)$ denotes the training loss component, while $\varphi(\theta)$ represents the regularization component. The training loss
quantifies the model's performance on the training dataset, whereas the regularization term aims to manage model complexity
to prevent overfitting. Furthermore, the complexity of each tree can be quantified using the equation:

$$\varphi(f) = \gamma T + \frac{1}{2} \tau \sum_{i=1}^T \omega_i^2 \tag{7}$$

where T indicates the total number of leaves and ω is the vector of scores assigned to these leaves. Ultimately, the structural
65 score of XGBoost is defined as the objective function outlined in equation (5), where ω_j remains independent of other
parameters.

S1.1.5. LightGBM Model

The LightGBM model, developed by Microsoft, is a scalable version of a tree-based gradient boosting algorithm that
optimizes parallel learning through network connection techniques (Ke et al., 2017). This model employs multiple T-regression
70 trees to derive the final predictions, operating on a per-leaf basis rather than a per-level approach. The algorithm can be
succinctly represented as follows:

$$\hat{Y} = \sum_{j=1}^T g_t(X) \quad (8)$$

LightGBM is frequently applied in domains such as big data analytics. Research indicates that it can achieve linear acceleration by leveraging multiple machines for targeted training. Consequently, the advantages of this algorithm include reduced training
75 times, lower memory requirements, and improved model accuracy.

S1.1.6. CatBoost Model

CatBoost represents an innovative iteration of the gradient boosting decision tree algorithm, distinguished by its robust learning capabilities for handling highly nonlinear data (Prokhorenkova et al., 2018). The algorithm employs random permutations of samples and utilizes computed datasets, resulting in fewer hyperparameters and reduced training durations. The prediction
80 function can be mathematically represented as:

$$h^t = \operatorname{argmin} \frac{1}{N} \sum (-f^t(X_k, Y_k) - h(X_k))^2 \quad (9)$$

In this equation, $h(X)$ denotes the output of the decision tree, while $f^t(X_k, Y_k)$ signifies the conditional distribution of gradients at the k -th sample. By introducing a distinct approach to modifying the gradient estimation system, CatBoost effectively mitigates prediction variations due to gradient bias, thereby enhancing the model's generalization capabilities.

85 S1.2. SHAP algorithm

The SHAP (SHapley Additive exPlanations) algorithm, introduced by Lundberg et al. (2020) (Lundberg, 2017), is an interpretative framework grounded in Shapley values (Shapley, 1953), aimed at elucidating individual predictions by assessing the contribution of each feature. For any predicted instance, the model produces a predicted value, and the SHAP value represents the assigned contribution of each feature within that instance (Shapley, 1953). In the context of an XGBoost model
90 where a group N with n features is utilized for prediction, SHAP assigns the contribution of a specific feature j to the model output $f(N)$ based on its marginal effect. The SHAP value for feature X_j can be expressed mathematically as:

$$SHAP(X_j) = \sum_{S \in N} \frac{k!(p-k-1)!}{p!} [f(S \cup \{j\}) - f(S)] \quad (10)$$

where p is the total number of features, N is a set of all possible combinations of features excluding X_j , S is a feature set in N , $f(S)$ is the model prediction with features in S , and $f(S \cup \{j\})$ is the model prediction with features in S plus feature X_j .

95 The interpretation of Eq. (10)

In summary, the SHAP value is derived from the Shapley value of the conditional expectation function corresponding to the original model. We employed SHAP values to quantify the approximate contributions of predictors to model predictions, based on average training outputs. Compared to traditional evaluation metrics, SHAP values are firmly rooted in statistical theory and are predominantly influenced by the performance of machine learning models. Given their efficacy in addressing complex
100 nonlinear relationships, SHAP values can offer valuable insights into certain mechanistic problems.

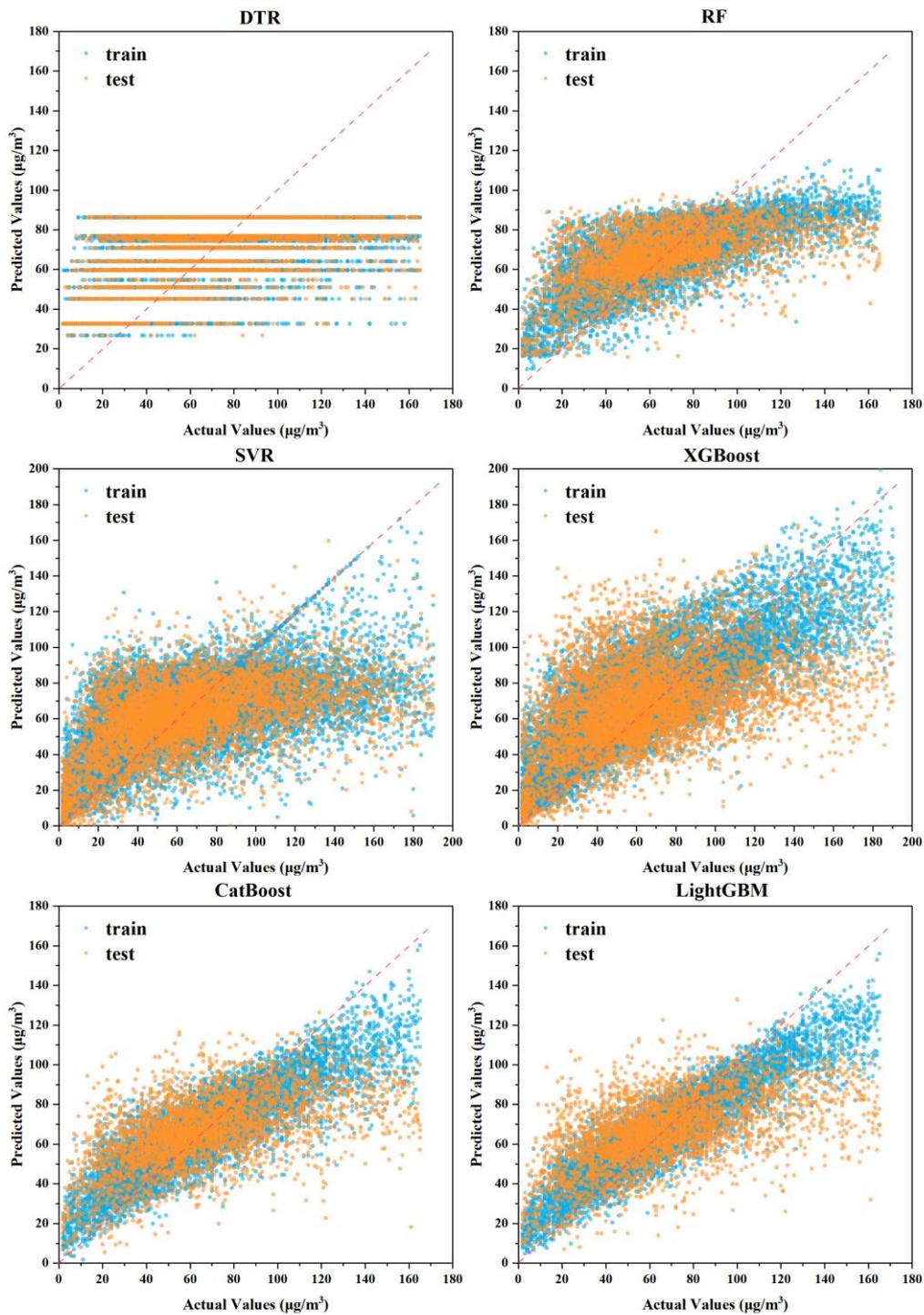


Fig. S1. The modeling results of different ML models on the test and train dataset.

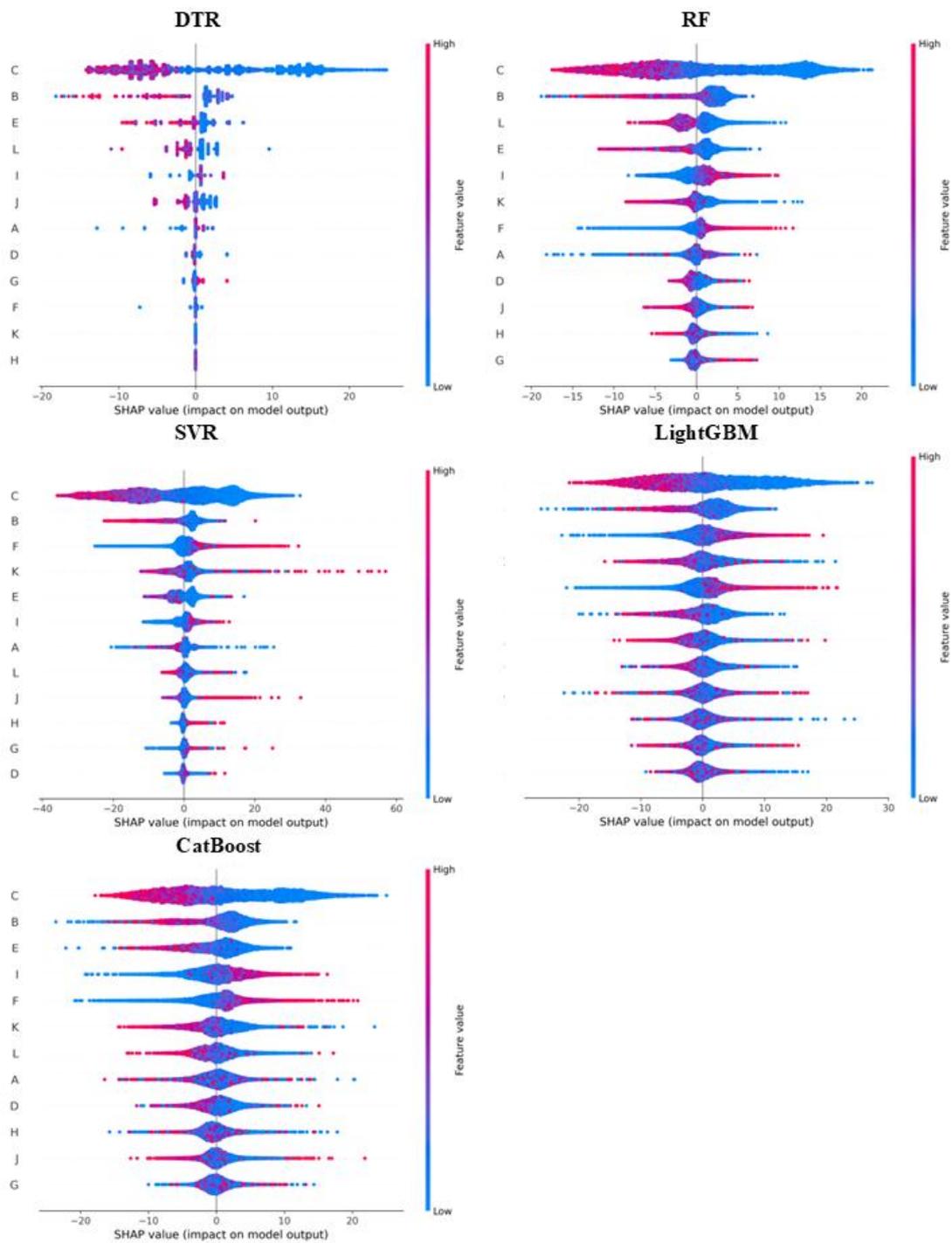


Fig. S2. Feature importance analysis of different ML models.

Site designator	Monitoring factors	Equipment brand model	Methodology
Site A	VOCs (C6-C12)	GC955-615	GC-FID/PID
	VOCs(C2-C6)	GC955-815	GC-FID/PID
Site B	VOCs (C6-C12)	GC955-615	GC-FID/PID
	VOCs(C2-C6)	GC955-815	GC-FID/PID
Site C	VOCs	PerkinElmer GC580	GC-FID
Site D	VOCs (C6-C12)	GC955-615	GC-FID/PID
	VOCs(C2-C6)	GC955-815	GC-FID/PID
Site E	VOCs (C6-C12)	GC955-615	GC-FID/PID
	VOCs(C2-C6)	GC955-815	GC-FID/PID
Site F	VOCs	Synspec GC955-615/815	GC-FID/PID
Site G	VOCs	Chromatotec Airmo VOC	FID
Site H	VOCs	GC300-FID/MS	GC-FID/MS
Site I	VOCs	Pu Yu GC3000-MS	CG-MS
Site J	VOCs	TH-300b	GC-FID/MS
Site K	VOCs	TH-300b	GC-FID/MS
Site L	VOCs (C6-C12)	Pu Yu GC3000-310	GC-FID
	VOCs(C2-C6)	Pu Yu GC3000-410	GC-FID

References

- Breiman, L.: Random forests, *Mach. Learn.*, 45, 5-32, <https://doi.org/10.1023/A:1010933404324>, 2001.
- Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y.: Lightgbm: A highly efficient gradient
110 boosting decision tree, *Adv. Neur. In.*, 30, <http://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree>., 2017.
- Loh, W. Y.: Classification and regression trees, *Wiley Interdiscip. Rev.: Data Min. Knowl. Discovery*, 1, 14-23, <https://doi.org/10.1002/widm.8>, 2011.
- Lundberg, S.: A unified approach to interpreting model predictions, *arXiv preprint arXiv:1705.07874*,
115 <https://dl.acm.org/doi/epdf/10.5555/3295222.3295230>, 2017.
- Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A.: CatBoost: unbiased boosting with categorical features, *Adv. Neur. In.*, 31, 2018.
- Shapley, L. S.: A value for n-person games, *Contribution to the Theory of Games*, 2, <https://doi.org/10.1515/9781400881970-018>, 1953.

Baklanov, A. and Korsholm, U.: On-line integrated meteorological and chemical transport modelling: advantages and perspectives, *Air Pollution Modeling and Its Application XIX*, Springer Netherlands, 3-17, https://doi.org/10.1007/978-1-4020-8453-9_1, 2008.

- 125 Burdett, I. D. and Eisinger, R. S.: Ethylene polymerization processes and manufacture of polyethylene, *Handbook of Industrial Polyethylene and Technology: Definitive Guide to Manufacturing, Properties, Processing, Applications and Markets*, 61-103, <https://doi.org/10.1002/9781119159797>, 2017.