Bayesian Optimization based Hyperparameter Tuning of Ensemble Regression Models in Smart City Air Quality Monitoring Data Analytics

Saptarshi Das
Centre for Environmental Mathematics, Faculty of Environment, Science and Economy, University of Exeter, Penryn Campus, Cornwall TR10 9FE, United Kingdom. Email: saptarshi.das@ieee.org, s.das3@exeter.ac.uk

Ahmed Alzimami
Centre for Environmental Mathematics, Faculty of Environment, Science and Economy, University of Exeter, Penryn Campus, Cornwall TR10 9FE, United Kingdom. Email: aa1106@exeter.ac.uk

***Abstract*—This paper uses the Bayesian optimization for fitting Ensemble regression models for tuning the machine learning model hyperparameters with reduced computation. We use the Pune Smart City air quality monitoring dataset with temporal variation of hazardous chemical pollutants in the air. The aim here is to reliably predict the suspended particulates as the air quality metrics using other environmental variables, considering linear models and nonlinear ensemble of tree models. To achieve good predictive accuracy a computationally expensive optimization method is required which has been achieved using the Gaussian Process surrogate assisted Bayesian optimization. We also show the diagnostics plots of the residuals from the nonlinear models to explain model quality.**

Keywords—air quality, smart city, pollution monitoring, ensemble regression, Bayesian optimization, hyper-parameter

I. Introduction

With the advent of cheap sensors and internet of things (IoTs), the smart city concept is gradually being built up even in large metropolitan areas of the developing nations. Smart city sensors monitor a wide urban area with different objectives amongst which waste or pollutants released to the environment is of a major concern. Degradation of the air quality in densely populated urban areas and the presence of large number of industrial and commercial cohabitation has put the public health at high risk. In order to develop modern smart cities, data driven artificial intelligence (AI) and machine learning (ML) based health exposure models, decision support systems, along with wide variety of sensing and computing technologies are being used e.g. geographic information system (GIS), satellite images, cloud computing to social media, crowd sourcing, mobile phones, wearable devices and low cost sensors (LCS) [1].

Air quality monitoring in smart cities is an important area of recent research using internet of things (IoTs) with big data analytics tools. For example, air quality index was developed in [2] using wide variety of sensors like carbon monoxide (CO), methane (CH_4), and volatile organic compounds (VOCs) utilizing modern technologies like wireless sensor networks (WSNs), IoTs, global positioning system (GPS), androids, and cloud services. Correlation analysis among different air pollutants have been previously tackled using association rules, semantics and taxonomy in [3]. A wide range of machine learning and statistical models were reviewed in [4] e.g. neural networks and deep learning models, spatio-temporal prediction of particulates using regression, ensemble and hybrid models using many standard performance metrics. Air quality prediction in smart cities using long short term memory (LSTM) and support vector regression (SVR) have been compared in [5] using predefined hyper-parameters which may be computationally expensive to fine tune for the optimal performance of the machine learning pipeline. A cloud based machine learning model comparison for Chinese city's air quality dataset has been reported in [6] with the comparison of decision tree, random forest, gradient boosting and multilayer perceptron neural networks, although the metrics used like root mean square error (RMSE), normalized RMSE, mean absolute error (MAE) may vary depending on the choice of target variable. Therefore, in this paper, we report the comparisons using a generalized metric adjusted R^2 (coefficient of determination) which is easy to interpret in a scale of 0-1. A spatial regression model with hypothesis testing for pollutant variable selection was reported in [7] for several smart cities in China.

II. Smart City Dataset Description and Exploratory Data Analysis

A. Pune Smart City Dataset Description

In a densely populated country like in India, the urban areas in metropolitan cities are crowded with railways, vehicles, industrial and residential buildings, giving rise to the higher health risk for the public. Previous studies around Pune city have found that there are many causes of air pollution like use of cookstoves [8], Diwali festival firecrackers [9] and some researchers also used optical and acoustic radars for data collection [10]. The dataset was collected during April to September 2019 in the Pune Smart City Development Corporation Limited (PSCDCL) at 10 different sample sights. The following variables are used to predict the air quality index (AQI) for railway stations, bus stop, information technology (IT) hubs etc. From the environmental chemical sensors, in the original dataset [11] the following variables were recorded with a sampling time of 15 mins *viz.* X_1 = humidity, X_2 = light, X_3 = Nitrogen Dioxide (NO_2) max/min, X_4 = Ozone (O_3) max/min, X_5 = particulates <10 microns (PM_{10}) max/min, X_6 = particulates<2.5 microns (PM_2) max/min, X_7 = Sulphur Dioxide (SO_2) max/min, X_8 = Carbon Monoxide (CO) max/min, X_9 = Carbon Dioxide (CO_2) max/min, X_{10} = sound (noise in db), X_{11} = temperature max/min, X_{12} = ultraviolet max/min, X_{13} = air pressure in bar, X_{14} = latitude and X_{15} = longitude. Here, the maximum and minimum both values for the same variables were recorded within a 15 mins window, only the maximum were considered in our data analytics pipeline optimization. Most of these pollutants were monitored using both fixed and mobile sensors, CCTV camera [12]. Some studies have focussed on comparing these variables in indoor vs. outdoor setting [13]. Other studies have investigated the relationship between noise and air pollution for environmental intelligence (EI) using crowdsensing and crowdsourcing methods [14]. Amongst these only the first 13 variables were chosen which are related to serious health hazards in a crowded city. Due to the spatial sparsity, the latitude and longitude variables were not considered in our optimized machine learning pipeline.

B. *Correlation Analysis and Spatio-Temporal Patterns*

Now, the correlation matrix between these 13 variables have been shown as heatmap in Figure 1 where the Pearson correlation coefficient of each pair has been calculated as:

$$\rho(x,y) = \frac{1}{N-1}\sum_{i=1}^{N}\left(\frac{x_i-\mu_x}{\sigma_x}\right)\left(\frac{y_i-\mu_y}{\sigma_x}\right) = \frac{\mathrm{cov}(x,y)}{\sigma_x\sigma_y}, \quad (1)$$

where, $\{\mu_x, \mu_y, \sigma_x, \sigma_y\}$ represents the mean and covariance of the any two variables x and y respectively and $\mathrm{cov}(x,y)$ represents their covariance matrix.

Figure 1. Correlation heatmap of 13 hazardous variables in smart city data.

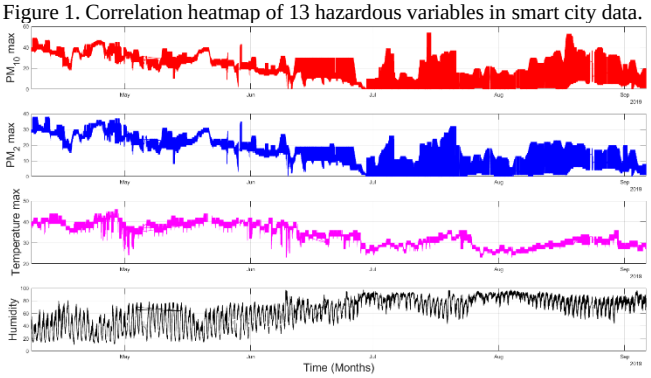

Figure 2: Timeseries plots of the three positively correlated variables – PM_{10} max, PM_2 max, temperature max and one negatively correlated varaible – humidity in the months of April to September 2019 in Pune smart city.

It is observed that the PM_{10} max and PM_2 max two variables are highly positively correlated ($\rho_{5,6}$ = 0.95). Also, these two levels of particulates are highly correlated to the ambient temperature (X_{11}) with correlation coefficient $\rho_{5,11}$ = 0.71 and $\rho_{6,11}$ = 0.79. This indicates that the presence of a greater number of particulate matters resists the atmosphere from natural cooling during the night-time giving rise to higher air temperature. Only significantly large negative correlations have been found between humidity with PM_2 and temperature with $\rho_{1,6}$ = -0.73 and $\rho_{1,11}$ = -0.77. This indicates that higher humidity helps the smaller suspended particulates to coagulate and get removed from the air and help reduce the ambient temperature. The three positively correlated variables in Figure 1 i.e. X_5 = PM_{10} max, X_6 = PM_2 max, X_{11} = temperature max and the negatively correlated variable i.e. X_1 = humidity have been shown in Figure 2. The time series plots in Figure 2 indicate that some of these variables go dangerously upwards together in the months of late-May, mid-July, mid-August. Especially during hot and humid days of the Indian subcontinent, both the PM concentrations rapidly fluctuates although the maximum ambient temperature gradually decreases. Next, we explore the three negatively correlated variables in the bottom most subplot of Figure 2 which shows that the humid days help in reducing the suspended particulates of both size from the air, during July to September months. A closer look at the correlation structure of these four variables in Figure 3 reveal that they are sparse in some areas. Figure 3 also shows that the PM_{10} max, PM_2 max, temperature max variables are either left or right skewed and not symmetric in their univariate distributions whereas the temperature shows a bimodal distribution with two peaks.

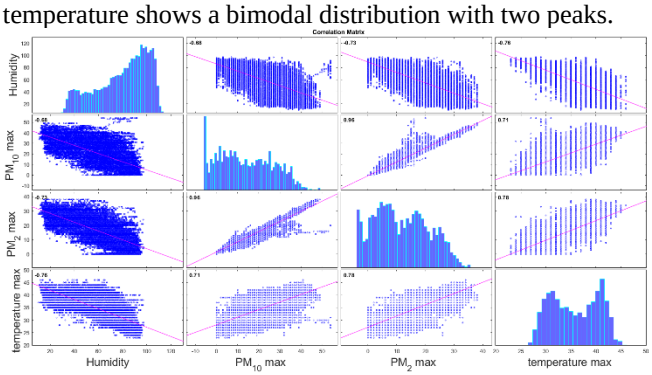

Figure 3: Univariate/bivariate structures of the top four correlated variables.

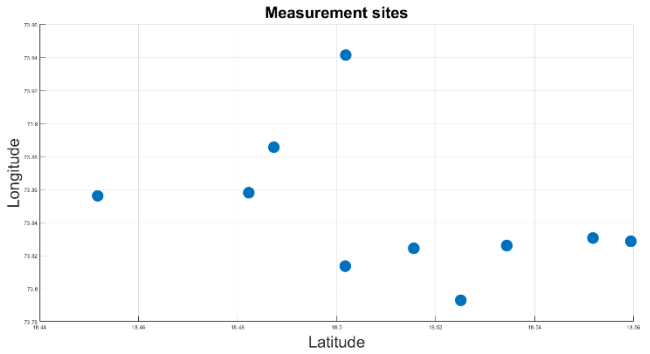

Figure 4: Pune air-quality measurement sites location latitude/longitudes.

The measurement sites are shown in Figure 4 as a function of latitude and longitude and due their sparsity, spatial variations have not been considered in our data analysis. From the previous literature it has been shown that amongst the 13 variables, the particulates PM_{10} and PM_2 create greater health risk. Hence, other environmental variables are considered as the explanatory variables to help predict these two target variables within a cross-validation based regression modelling framework using both linear and nonlinear family of models.

III. Hyperparameter Tuning of Regression Models using Bayesian Optimization

A. *Bayesian Optimization of Linear Regression Models*

For the current regression modelling problem, initially we employ the simpler linear regression methods and then moved to more complex nonlinear ensemble regression models for better accuracy and performance. The linear regression model employs two different family of learners – least squares (LS) and support vector machine (SVM) as categorical variables and the regularization term strength (λ) as real variable [15]. The same hyperparameter optimization technique has been applied for both the particulate levels and the comparative performances are reported in Table 1 which shows slightly better results for predicting PM_2. However, we observe that using linear models the 5-fold cross-validated accuracy is not very great which motivates us to compare more complex and computationally expensive nonlinear regression models. Figure 5 shows that the optimization landscape based on the two search variables where Gaussian Process (GP) surrogate models have been used to search for new datapoints in the hyperparameter space to reduce the computation as opposed

to an extensive grid-search method [16]. In the Bayesian optimization, from a set of initial seed points, a GP surrogate model is fitted to search for new evaluations of the function. Iterations were stopped after 30 steps of the Bayesian optimization which is sufficient for the steady convergence of the machine learning model hyperparameter search. For numerical comparison, the best-found points from the hyperparameter optimization of linear regression models are reported in Table 1. From the corresponding linear fit between the ground truth and predicted values, the adjusted R^2 values of the two target variables or particulate levels, it is evident that none of them can be modelled adequately ($R^2 \sim 0.01$) with the linear family of ML models (SVM) even after employing Bayesian optimization for hyper-parameter tuning.

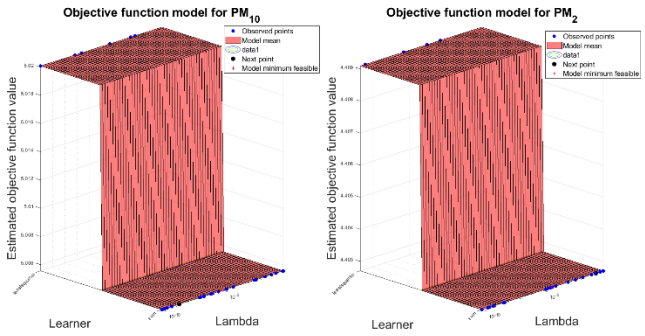

Figure 5: Sampled points and fitted surface by the Bayesian optimizer for hyperparameter tuning of linear regression models for both particulates.

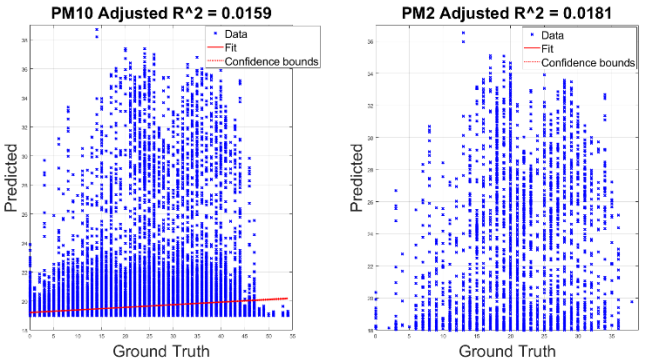

Figure 6: Hyper-parameter optimized prediction performance of linear regression models for the prediction of $X_5 = PM_{10}$ max and $X_6 = PM_2$ max.

TABLE 1: OPTIMIZED HYPERPARAMETERS OF THE LINEAR REGRESSION

Target Variables	Lambda (λ)	Learner	Optimization Time (sec)	Objective Function Minima
$X_5 = PM_{10}$	4.37×10^{-6}	SVM	29.2063	5.0056
$X_6 = PM_2$	2.39×10^{-5}	SVM	25.4746	4.4027

B. *Bayesian Optimization of Ensemble Regression Models*

Due to inadequacy of the linear models, we now employ more complex nonlinear ensemble regression models. The ensemble model trains 100 regression trees based on arbitrarily set of hyperparameters and such models are computationally more expensive as compared to the linear regression models [17]. This justifies the use of Bayesian optimization method which fits a GP surrogate model to decide the expected minima of the optimization landscape while optimizing over variables in both the categorical and real space. It is evident from Table 2 that unlike the linear regression model optimization, the ensemble models take ~20-75 mins even in a parallel computing mode to complete the hyperparameter optimization task. The hyperparameter optimization tuning for ensemble regression models were run for 30 iterations which are sufficient to reach convergence where no better solutions can be found over last few sets of function evaluations as shown in Figure 7. However, the obtained objective function minima are much smaller than the linear models. Also, improved performance of the Bayesian optimization based optimal hyperparameter tuning for the ensemble model can be confirmed from the prediction results shown in Figure 8 with R^2>0.96 for both PM_{10} and PM_2.

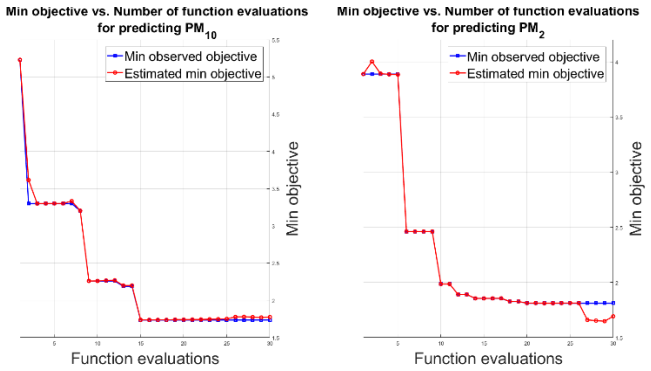

Figure 7: Hyper-parameter optimization characteristics of ensemble regression model for predicting $X_5 = PM_{10}$ max and $X_6 = PM_2$ max.

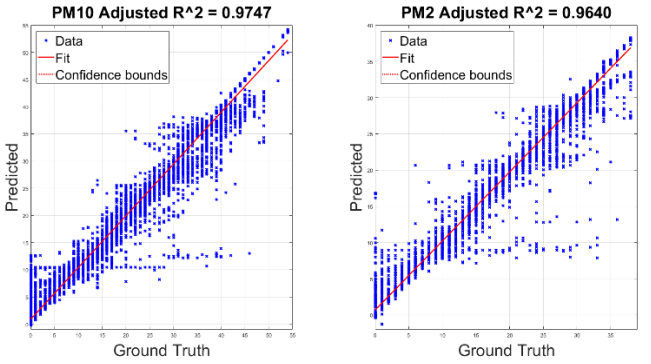

Figure 8: Hyper-parameter optimized prediction performance of ensemble regression model for $X_5 = PM_{10}$ max and $X_6 = PM_2$ max.

TABLE 2: 3 HYPERPARAMETER BAYESIAN OPTIMIZATION COMPUTING TIME

Target variable	Computing time	Objective Function evaluation time	Estimated Objective function value	Estimated Function evaluation time
$X_5 = PM_{10}$	74.46 min	5.75 hours	1.7751	7.8 mins
$X_6 = PM_2$	19.46 min	1.61 hours	1.458	1.88 mins

TABLE 3: OPTIMIZED HYPERPARAMETERS OF THE ENSEMBLE REGRESSION

Target variables	NumLearningCycles	LearnRate	MaxNumSplits
$X_5 = PM_{10}$	51	0.14162	44252
$X_6 = PM_2$	13	0.57958	13134

For optimizing the ensemble regression models, we employ the GP regression based Bayesian optimization method to search for the best combination of method - Bagging or least square boost (LSBoost) [18], number of learning cycles, learning rate and maximum number of splits as the three hyperparameters by minimizing the objective function log(1+loss), where loss represent the loss after cross-validation [19]. This uses a template tree surrogate model to search for the optimal combination of the model

hyperparameters for predicting both the particulate matter levels PM_{10} and PM_2. The acquisition function has been chosen as the 'expected improvement plus' and a parallel computation of the objective function on an 8-core CPU has been used to carry out the Bayesian optimization. We observed that the adjusted R^2 value has been improved after the hyperparameter tuning for both the target variables as compared to their default choice. From the hyperparameter optimized regression model, we then calculate the R^2 values between the ground truth vs. the predicted target variables, similar to the linear family of models, shown in Figure 6. For the ensemble regression models, the optimized hyperparameters have been reported in Table 3 which yield satisfactory predictive performance shown in Figure 8. For the hyper-parameter optimized ensemble regression models, we also show the re-substitution loss with respect to the number of learning cycles in Figure 9 for both the pollutant levels. This shows that as the number of trained ensemble learners increases the cumulative substitution loss decreases. However, PM_{10} takes a greater number of cycles for converges as opposed to PM_2 for both individual and cumulative loss.

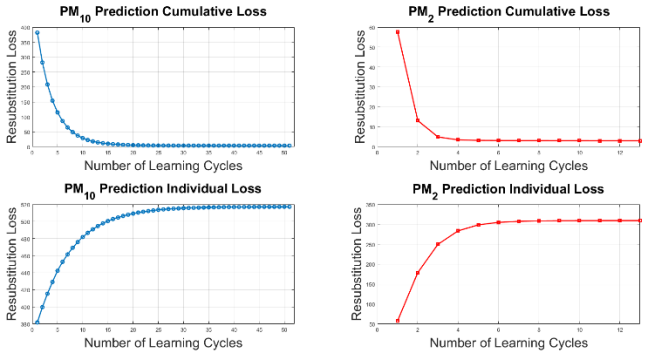

Figure 9: Cumulative and individual resubstition loss of the all three hyperparameter optimized ensemble regression models for both pollutants.

C. *Pairwise Hyperparameter Optimization Results*

In the previous section 3 hyperparameters were optimized simultaneously for the ensemble learning models which is computationally expensive as shown in Table 2-Table 3. Next, in order to see the effect of most important pair of hyperparameters among these three and the performance degradation due to fewer number of hyperparameter optimization, we consider pairwise optimization of the three hyperparameters i.e. number of learning cycles, learning rate and maximum number of splits on a 8-core CPU using similar parallel computing method in the previous section. The pairwise hyper-parameter optimization also allows us to intuitively look at the surface of the estimated objective function value as a 2D function of the chosen hyperparameters as well as the sampled points used to build the GP based surrogate model on the hyperparameter space.

Next we show the convergence characteristics of the three pairs of hyperparameter optimization in Figure 10, Figure 12 and Figure 14 whereas the corresponding optimization surface developed by fitting a GP surrogate model during the Bayesian optimization process are shown in Figure 11, Figure 13 and Figure 15 respectively. All three pairs of hyperparameter optimization of ensemble regression models converge within 30 iterations although the smoothness of the surrogate surface may vary. Moreover, we find from the respective prediction performances shown in Figure 16-Figure 18 we find that that only optimizing pair 1 yields relatively inferior results in terms of the R^2 metric of both pollutants.

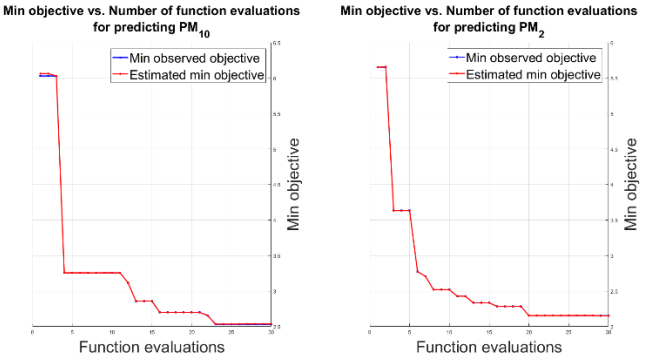

Figure 10: Pair 1-NumLearningCycles/LearnRate optimization convergence.

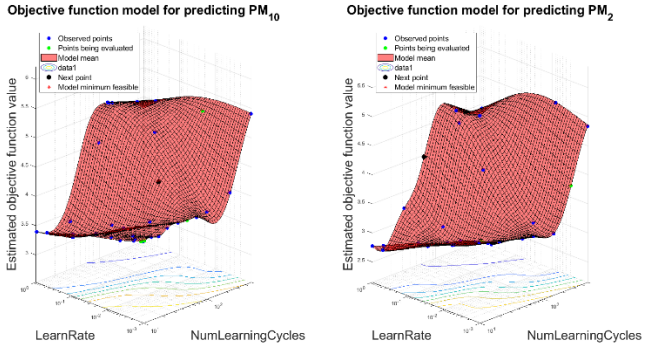

Figure 11: Pair 1 - NumLearningCycles/LearnRate optimization surface.

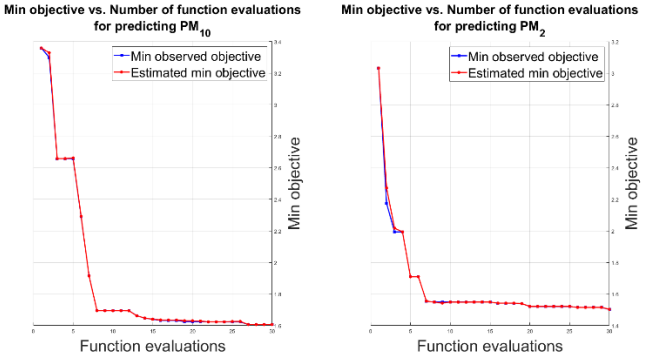

Figure 12: Pair 2 - NumLearningCycles/MaxNumSplits optimization convergence characteristics.

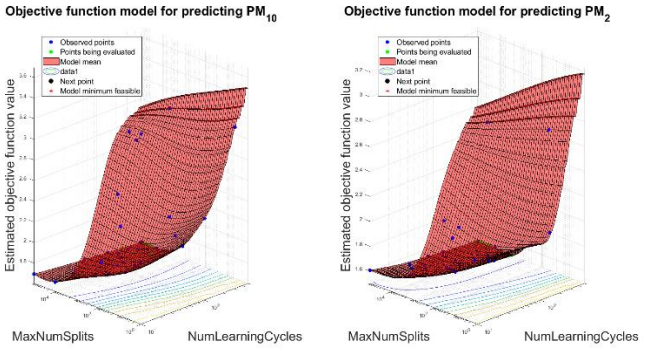

Figure 13: Pair 2-NumLearningCycles/ MaxNumSplits optimization surface.

The pairwise hyperparameter optimization results show that within ensemble regression models, the MaxNumSplits has higher importance as compared to the other two hyperparameters – NumLearningCycles and LearnRate. We also observe that dropping the less important hyperparameter even yields better R^2 as compared to when all the three were simultaneously optimized which may be an overly complex exercise. It is evident from the comparison of the reduction in computing time in Table 4 for the pairwise hyperparameter optimization as opposed to tuning all three hyperparameters.

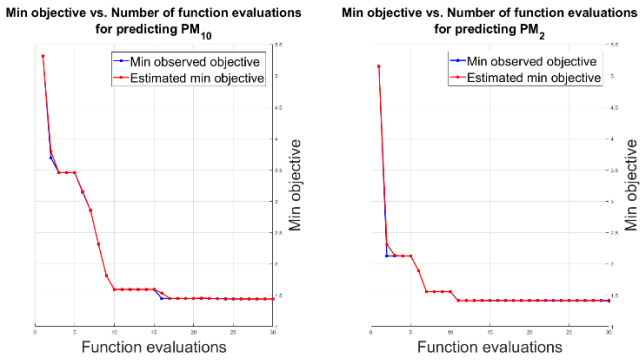

Figure 14: Pair 3-LearnRate/ MaxNumSplits optimization convergence.

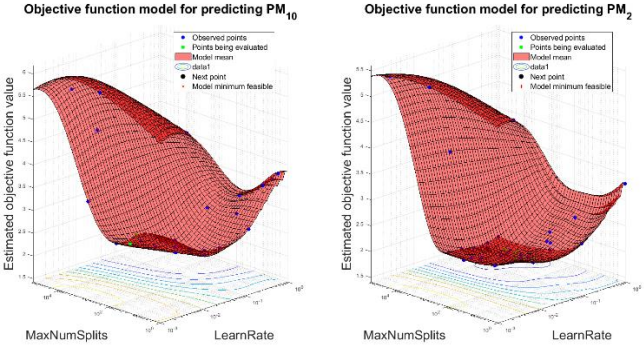

Figure 15: Pair 3-LearnRate/MaxNumSplits optimization surface.

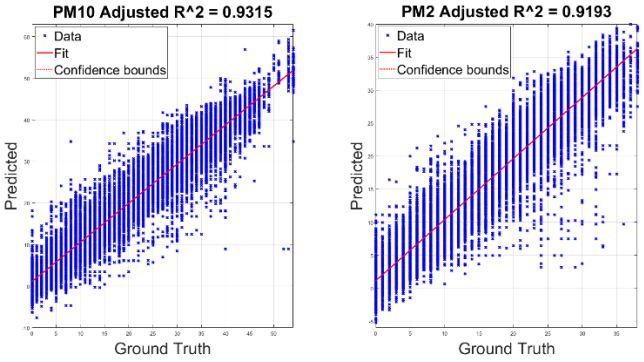

Figure 16: 1st pair of hyper-parameter optimized prediction performance of ensemble regression models for PM_{10} max and PM_2 max.

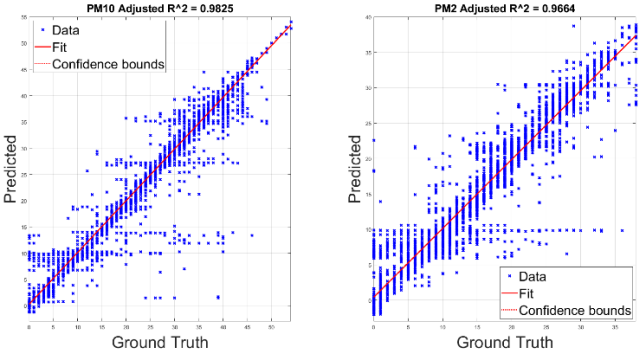

Figure 17: 2nd pair of hyper-parameter optimized prediction performance of ensemble regression models for PM_{10} max and PM_2 max.

Next, we show the re-substitution losses for three pairs of hyperparameter optimization in Figure 19-Figure 21. It is found that the pair 3 takes a smaller number of cycles/learners for the convergence of the re-substitution loss. Moreover, the individual losses converges faster than the cumulative loss for all the three cases. Also, for pair 2 optimization, non-smooth convergence characteristics are observed in Figure 20.

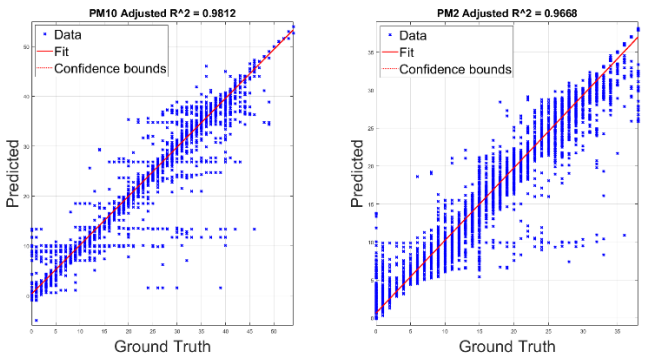

Figure 18: 3rd pair of hyper-parameter optimized prediction performance of ensemble regression models for PM_{10} max and PM_2 max.

TABLE 4: 3 HYPERPARAMETER BAYESIAN OPTIMIZATION COMPUTING TIME

Variable	Hyperparameters	Computing time (mins)	Objective Function evaluation time (hours)	Estimated Objective function value	Estimated Function evaluation time (mins)
X_5 = PM_{10}	Pair 1	23.13	2.45	2.54	11.22
	Pair 2	93.48	9.29	1.61	65.30
	Pair 3	52.50	5.70	1.44	16.58
X_6 = PM_2	Pair 1	25.07	2.32	2.15	11.16
	Pair 2	86.32	7.75	1.50	55.08
	Pair 3	56.63	6.54	1.42	18.59

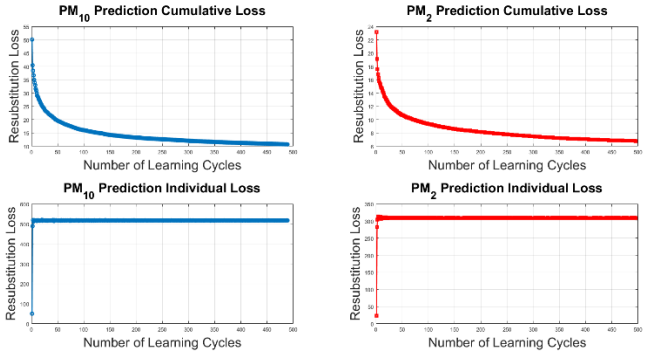

Figure 19: Resubstitution loss for pair 1 hyperparameter optimization.

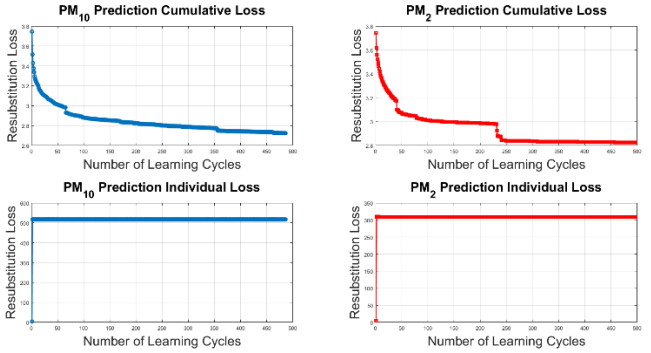

Figure 20: Resubstitution loss for pair 2 hyperparameter optimization.

D. Residual Diagnostic Analysis of the Fitted Models

The regression modelling results after the hyperparameter optimization may not perfectly match the target variable. As such the discrepancy or error between the actual and predicted values of the should be normally distributed. In order to check that for both the target variables PM_{10} and PM_2, we show the diagnostic analysis results for the 3 and pairwise hyperparameter optimized regression models in Figure 22 and

Figure 23 respectively. A fitted quadratic model through the residual vs fitted value plot show almost horizontal pattern showing validity of the learnt models in different extents.

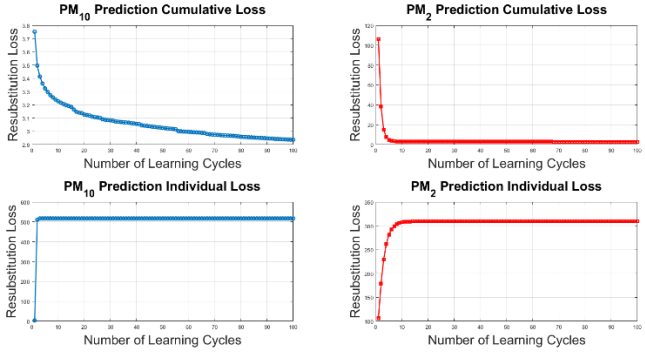

Figure 21: Resubstitution loss for pair 3 hyperparameter optimization.

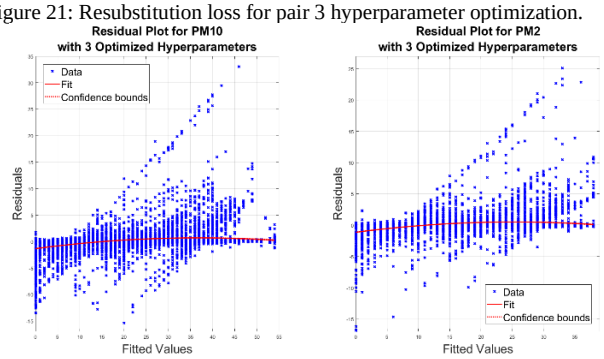

Figure 22: Residual plots for the 3 hyperparameter optimized ensemble regresion models for both PM_{10} and PM_2 predictions.

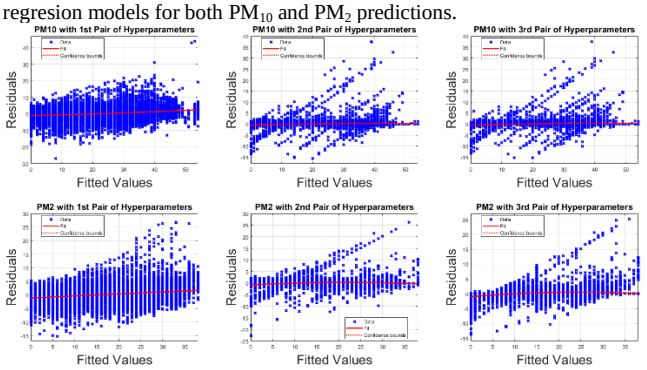

Figure 23: Residual plots for the 3 different pairs of the 2 hyperparameter optimized ensemble regresion models for both PM_{10} and PM_2 predictions.

IV. Conclusions

In order to analyze the Pune Smart city air quality dataset, we compare the hyperparameter optimized machine learning models from the linear regression and ensemble regression families. We show a cross-validated regression performance comparison between these families as well as computational time comparison for the prediction of particulates PM_{10} and PM_2 as a function of other environmental variables. We compare three vs. pairwise hyperparameter optimization results. In the future, we shall explore more families of machine learning models for modelling similar smart city environmental monitoring datasets in other cities and countries. Also, the temporal patterns of this dataset can be explored using time-series models in the future.

Acknowledgment

SD was partially supported by the ERDF Deep Digital Cornwall project number: 05R18P02820.

References

[1] A. Kaginalkar, S. Kumar, P. Gargava, and D. Niyogi, ‘Review of urban computing in air quality management as smart city service: An integrated IoT, AI, and cloud technology perspective’, *Urban Clim.*, vol. 39, p. 100972, Sep. 2021.

[2] S. Dhingra, R. B. Madda, A. H. Gandomi, R. Patan, and M. Daneshmand, ‘Internet of Things Mobile–Air Pollution Monitoring System (IoT-Mobair)’, *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5577–5584, Jun. 2019.

[3] L. Cagliero, T. Cerquitelli, S. Chiusano, P. Garza, G. Ricupero, and X. Xiao, ‘Modeling Correlations among Air Pollution-Related Data through Generalized Association Rules’, in *2016 IEEE International Conference on Smart Computing (SMARTCOMP)*, St Louis, MO, USA, May 2016, pp. 1–6.

[4] D. Iskandaryan, F. Ramos, and S. Trilles, ‘Air Quality Prediction in Smart Cities Using Machine Learning Technologies based on Sensor Data: A Review’, *Appl. Sci.*, vol. 10, no. 7, p. 2401, Apr. 2020.

[5] I. Kok, M. U. Simsek, and S. Ozdemir, ‘A deep learning model for air quality prediction in smart cities’, in *2017 IEEE International Conference on Big Data (Big Data)*, Boston, MA, USA, Dec. 2017, pp. 1983–1990.

[6] S. Ameer *et al.*, ‘Comparative Analysis of Machine Learning Techniques for Predicting Air Quality in Smart Cities’, *IEEE Access*, vol. 7, pp. 128325–128338, 2019.

[7] L. Li, Y. Zheng, S. Zheng, and H. Ke, ‘The new smart city programme: Evaluating the effect of the internet of energy on air quality in China’, *Sci. Total Environ.*, vol. 714, p. 136380, Apr. 2020.

[8] K. Dutta, K. N. Shields, R. Edwards, and K. R. Smith, ‘Impact of improved biomass cookstoves on indoor air quality near Pune, India’, *Energy Sustain. Dev.*, vol. 11, no. 2, pp. 19–32, Jun. 2007.

[9] P. C. S. Devara, K. Vijayakumar, P. D. Safai, P. R. Made, and P. S. P. Rao, ‘Celebration-induced air quality over a tropical urban station, Pune, India’, *Atmospheric Pollut. Res.*, vol. 6, no. 3, pp. 511–520, May 2015.

[10] P. C. S. Devara, R. S. Maheskumar, B. S. Murthy, and K. G. Vernekar, ‘Air-quality monitoring by optical and acoustic radars at Pune, India’, *Meas. Sci. Technol.*, vol. 8, no. 10, pp. 1160–1165, Oct. 1997.

[11] A. Mahajan, ‘Pune Smart City Dataset’. Kaggle. [Online]. Available: https://www.kaggle.com/datasets/akshman/pune-smartcity-test-dataset

[12] Y. Mehta, M. M. Manohara Pai, S. Mallissery, and S. Singh, ‘Cloud enabled air quality detection, analysis and prediction - A smart city application for smart health’, in *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*, Muscat, Mar. 2016, pp. 1–7.

[13] G. Marques, J. Saini, M. Dutta, P. K. Singh, and W.-C. Hong, ‘Indoor Air Quality Monitoring Systems for Enhanced Living Environments: A Review toward Sustainable Smart Cities’, *Sustainability*, vol. 12, no. 10, p. 4024, May 2020.

[14] A. Bardoutsos, G. Filios, I. Katsidimas, T. Krousarlis, S. Nikoletseas, and P. Tzamalis, ‘A multidimensional human-centric framework for environmental intelligence: Air pollution and noise in smart cities’, in *2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, Marina del Rey, CA, USA, May 2020, pp. 155–164.

[15] C.-J. Hsieh, K.-W. Chang, C.-J. Lin, S. S. Keerthi, and S. Sundararajan, ‘A dual coordinate descent method for large-scale linear SVM’, in *Proceedings of the 25th international conference on Machine learning - ICML '08*, Helsinki, Finland, 2008, pp. 408–415.

[16] V. Nguyen, ‘Bayesian Optimization for Accelerating Hyper-Parameter Tuning’, in *2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, Sardinia, Italy, Jun. 2019, pp. 302–305.

[17] Y. Ren, L. Zhang, and P. N. Suganthan, ‘Ensemble Classification and Regression-Recent Developments, Applications and Future Directions [Review Article]’, *IEEE Comput. Intell. Mag.*, vol. 11, no. 1, pp. 41–53, Feb. 2016.

[18] J. Isabona, A. L. Imoize, and Y. Kim, ‘Machine Learning-Based Boosted Regression Ensemble Combined with Hyperparameter Tuning for Optimal Adaptive Learning’, *Sensors*, vol. 22, no. 10, p. 3776, May 2022.

[19] P. Bühlmann, ‘Bagging, Boosting and Ensemble Methods’, in *Handbook of Computational Statistics*, J. E. Gentle, W. K. Härdle, and Y. Mori, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 985–1022.