MDPI

Article

Effectiveness of Super Learners in Machine Learning Calibration of Low-Cost Particulate Matter Sensors

Gokul Balagopal [1,†,‡], Seth Lee [2,‡], Vardhan Agnihotri[3,‡] and David J. Lary [2,*]

[1] Affiliation 1; e-mail@e-mail.com
[2] Affiliation 2; e-mail@e-mail.com
* Correspondence: e-mail@e-mail.com; Tel.: (optional; include country code; if there are multiple corresponding authors, add author initials) +xx-xxxx-xxx-xxxx (F.L.)
† Current address: Affiliation 3.
‡ These authors contributed equally to this work.

Abstract: A single paragraph of about 200 words maximum. For research articles, abstracts should give a pertinent overview of the work. We strongly encourage authors to use the following style of structured abstracts, but without headings: (1) Background: place the question addressed in a broad context and highlight the purpose of the study; (2) Methods: describe briefly the main methods or treatments applied; (3) Results: summarize the article's main findings; (4) Conclusions: indicate the main conclusions or interpretations. The abstract should be an objective representation of the article, it must not contain results which are not presented and substantiated in the main text and should not exaggerate the main conclusions.

Keywords: airborne particulates, particulate sensors, machine learning calibration

Citation: Lastname, F.; Lastname, F.; Lastname, F. Title. *Journal Not Specified* **2023**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

1. Introduction

Particulate matter (PM), also referred to as aerosol particles, refers to a mixture of solid particles and liquid droplets commonly found in the air. Particulate matter is widespread due to the wide variety of sources that it comes from, such as emissions from internal combustion engines found in cars [1]. Particulate matter is of pressing concern because they are a significant yet overlooked contributor to illness and disease, specifically of the lungs and heart. PM can penetrate deep into the lungs and may even enter the bloodstream, resulting in adverse health effects and millions of deaths every year [2,3]. Particulate matter not only directly harms human health through irritation but also acts as a vector for other contaminants and chemicals to enter the human body [4]. In the context of human health, the detrimental effects of PM are dictated by both PM concentration and PM size. For the sake of this study, and in general, PM is measured in $\mu g/cm^3$, with higher concentrations of PM resulting in worse effects. PM size is also an important factor to consider, with PM encompassing a wide range of sizes ranging from around 10µm to below 0.1µm. The smaller the PM, the more harmful it is to human health due to its ability to penetrate deeper into the respiratory system compared to larger PM [5]. Due to the wide variety of PM sizes and complicated composition of PM, accurate PM sensors can incur a substantial cost, presenting a significant barrier to the wide-scale implementation of PM monitoring.

1.1. Motivation for this Study

Currently, accurate and reliable air pollution sensors can easily cost US$5000, with various high-grade sensors costing many times more [6]. As such, sensors that are both cost-effective and capable of precise measurements are of significant value due to their ability to be widely distributed. During the past two decades, there has been increasing research and development in the field of PM sensor calibration. As part of the Multi-Scale Integrated Intelligent Interactive Sensing Lab (MINTS), we have implemented the use of

machine learning to calibrate sensors of all kinds. Initially, machine learning calibration was done on sensors housed on orbital satellites [7]. Now, we are utilizing this approach to calibrate low-cost sensors deployed on a wide-scale across urban environments, specifically around the Dallas-Fort Worth metroplex [8]. These sensors provide useful information to the public and policymakers as they can be affordably deployed to accurately measure PM concentration by particle size across many areas.

Although there has been steady development in the calibration of low-cost sensors in recent years, there is significant room for growth since machine learning calibration still presents inaccuracies, especially when dealing with long-term usage. Because the weather patterns in a given area change frequently and can drastically alter over the course of multiple weeks to months, the machine learning model will likely predict PM concentration less accurately as time passes [9]. This means that although low-cost sensors may accurately predict PM concentration for a given time period from which the machine learning model was trained on, the calibration's effectiveness diminishes as weather patterns change and potential human developments take place. Therefore, it is of significant importance to improve machine learning calibration's accuracy in predicting PM concentration. More accurate machine learning algorithms will enable training models with higher precision over longer periods of time, allowing for the wider deployment of affordable PM sensors. If sensors have to be re-calibrated frequently due to inaccuracies in training the machine learning model, the effectiveness of gathering consistent and reliable data from low-cost PM sensors greatly diminishes.

1.2. Bagging, Boosting, and Stacking

Bagging, boosting, and stacking are all instances of ensemble learning methods that can be used to improve machine learning performance. Ensemble learning methods work by combining multiple models, also known as weak-learners, into an ensemble learner, referred to as a strong learner. Although there are a wide variety of ensemble learning techniques, bagging, boosting, and stacking are the most commonly used ensemble methods. Because machine learning models often experience high variance (over-fitting) or high bias (under-fitting), ensemble methods are useful since they can combine different weak learners to reduce variance and bias. The first commonly used method is bagging.

1.3. Super Learners in Machine Learning

This study primarily focuses on the implementation of super learners for low-cost PM sensor calibration. Super learners are a type of machine learning algorithm that combines the predictions of multiple machine learning models, known as base models, to more accurately predict values [10]. Super learners work by splitting the data into multiple folds through cross-validation and training the base models on many of the folds. After the model is trained, each model makes out-of-fold predictions, which refers to making predictions with the untrained and unused folds. The super learner algorithm then uses a meta-learner, which is a model trained on the out-of-fold predictions, to assign weights (coefficients) to each of the base models. These weights allow the super learner to implement the optimal combination of the base models through a process called stacking, resulting in more precise predictions. Within contemporary research, super learners have already been used in a variety of applications, such as predicting soil properties based on environmental variables. Such research suggests that super learners exhibited a significant improvement in precision and a decrease in root mean square error over other regression algorithms [11]. Super learners are especially effective in situations where different machine learning models perform well under different conditions. This is particularly relevant to PM sensor calibration since predictions from machine learning models are highly dependent upon weather patterns and other variables that influence PM concentration. Depending on the circumstance, one machine learning model may more accurately estimate the PM concentration given the current environment. However, in another environment, a different machine learning model may be more accurate in

predicting the PM concentration. Therefore, this study aims to compare the effectiveness of calibration using super learners compared to base models, such as Gaussian process regression, random forest regression, decision trees, etc.

2. Materials and Methods

2.1. Sensors

In this study, a comprehensive approach using an array of sensors was necessary to collect sufficient data for machine learning calibration. For the collection of the input data (machine learning features), we used the LoRa Node, an ensemble of sensors that collect environmental data ranging from humidity to nitrogen dioxide (NO2). The LoRa Node uses LoRa (Long Range), a wireless communication method, to transmit the collected sensor data to the Central Node, a system capable of connecting directly to the internet. For the primary PM sensor, the LoRa Node utilizes the PPD42NS sensor, an affordable optical sensor which outputs PM concentration for two size ranges. The first PM size range includes data for particulates over 1µm in diameter, while the second PM size range measures particulates over 2.5µm in diameter. To calculate the specific concentrations for each PM size range, the PPD42NS sensor measures the low pulse occupancy (LPO) time per 15 seconds. The LPO time represents the amount of time, in microseconds, that the sensor reads a low pulse, which is proportional to the particle count concentration. Along with the LPO, the sensor also records the ratio between the LPO time and the entire sample time, which is 15 seconds. The LPO and ratio values are useful as they can both be used to calculate a rough estimation of the PM concentration.

The primary downside to these measurement ranges, however, is that the collected data doesn't include PM sizes below 1µm in diameter. Additionally, the PM sensor is only capable of roughly measuring the particulate count per volume rather than the standard PM concentration measurement of mass per volume. Although the PM sensor in the LoRa Node falls short of precisely measuring PM concentration, the LoRa Node also incorporates other varieties of sensors, extending beyond PM sensors. Specifically, the LoRa Node is equipped with the 101020088 Seeed Studio Multichannel Gas Sensor capable of measuring NH_3, NO_2, C_3H_8, C_4H_{10}, CH_4, H_2, and C_2H_5OH as referenced along with their respective gas sensor measurement ranges in Table 1.

Table 1. 101020088 Seeed Studio Multichannel Gas Sensor measurement specifications

Gas Type	Measurement Range (ppm)	Measurement Error
Ammonia (NH_3)	1–500	±15-25%
Nitrogen Dioxide (NO_2)	0.05–10	±15-25%
Propane (C_3H_8)	>1000	±15-25%
Iso-Butane (C_4H_{10})	>1000	±15-25%
Methane (CH_4)	>1000	±15-25%
Hydrogen (H_2)	1–1000	±15-25%
Ethanol (C_2H_5OH)	10–500	±15-25%

ppm = Parts per million

Additionally, the LoRa Node contains an Adafruit BME280 sensor to collect temperature, pressure, and humidity data. Because temperature, pressure, and humidity are key indicators of the given environment, they are useful data points to have for calibrating the PM sensors. The BME280 sensor is capable of measuring temperatures ranging from -40°C to 85°C (±1.0°C), pressure ranging from 300 to 1100 hPa (±1.0 hPa), and humidity ranging from 0% to 100% relative humidity as listed in Table 2.

Table 2. BME280 sensor measurement specifications

Weather Variable	Measurement Range
Temperature	-40–85°C (±1.0°C)
Pressure	300–1100 hPa (±1.0 hPa)
Humidity	0–100% Relative Humidity (±3%)

hPa = Hectopascal

These additional sensors make the LoRa Node optimal for particulate sensor calibration, using other measured variables to better draw a conclusion on . To make the LoRa Node data useful in accurately predicting particulate matter concentration, the sensors from the LoRa Nodes were calibrated with the Palas Fidas® Frog sensor, which is comparatively more precise than the PM sensor housed in the LoRa Node. The Fidas® Frog sensor is an optical aerosol spectrometer designed to measure particulate matter concentration of particulates with a diameter of 0.18–40 µm by detecting the scattering of light caused by the presence of airborne particles. Specifically, the Fidas® Frog sensor reports PM_1, $PM_{2.5}$, PM_4, PM_{10}, and total PM concentration, as well as total particle count density (dCn). These precise measurements allow us to calibrate the LoRa Node, predict precise PM concentration values by specific sizes, such as $PM_{2.5}$.

2.2. Calibration Process

Both the LoRa Node and the Fidas® Frog were placed on-site in the Waterview Science and Technology Center (WSTC) on the University of Texas at Dallas (UTD) campus to ensure that the sensors collected data under the same environmental conditions with a 30 second sampling period. To prepare the data for calibration, the data collected from the LoRa Nodes and the Fidas® Frog was converted into a CSV (Comma-Separated Values) file. Before training the machine learning models, we loaded the data from the CSV file into a data frame and partitioned the data for both training the model and testing prediction accuracy. There were a total of 16 input variables from the LoRa data and 1 target variable from the Fidas® Frog data that were used to train each machine learning model using a supervised learning regression algorithm. Each model would then be able to predict the given target variable given only LoRa Node input data, commonly referred to as features. For this study, we analyzed a total of 6 target variables from the Fidas® Frog: PM_1, $PM_{2.5}$, PM_4, PM_{10}, total PM, and dCn data. To achieve a baseline of what base models were effective, multiple machine learning algorithms were tried, including neural networks, support vector regression, decision trees, linear regression, random forest regression, and gaussian process regression.

In order to determine the effectiveness of these various machine learning algorithms, we calculated the coefficient of determination, also known as the r^2 value. To calculate the value, we used the equation

$$R^2 = 1 - \frac{\sum_{i=1}^{N}[y_i - f(x_i)]^2}{\sum_{i=1}^{N}[y_i - \overline{y}]^2}$$

where $y_i - f(x_i)$ represents the actual value of a given point minus the predicted value of a given point, $y_i - \overline{y}$ represents the actual value of a given point minus the mean of all points, and N represents the number of data points. The coefficient of determination is a useful indicator of how well a statistical model predicts an outcome. Specifically, the R^2 value would indicate the proportion of variance for the target variable that is explained by the data set. For example, a 0.9 R^2 value would indicate that 90% of the variance for the predicted PM value is explained by the LoRa Node sensor data.

2.3. Bagging, Boosting, and Stacking

3. Results

3.1. Base Models

Using the regression models mentioned above, we initially trained the models on Fidas® Frog and Lora Node data that was collected over the span of roughly two days. We then predicted PM concentrations by inputting the LoRa Node test data, which was not used in the calibration process, into the trained model. Figure 1 shows a comparison between the actual PM data recorded from the Fidas® Frog and the predicted PM value using a decision tree model:

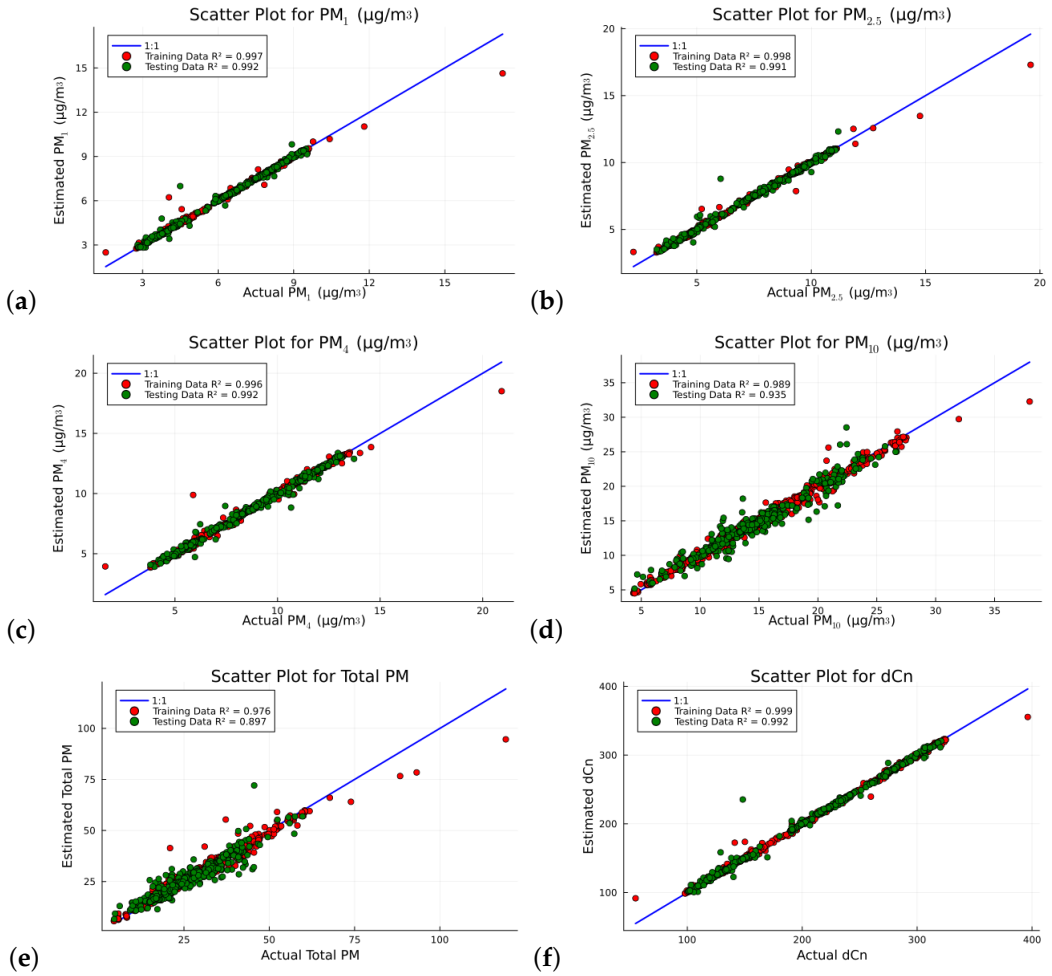

Figure 1. This figure shows multiple scatter plots comparing the actual PM data against the estimated PM value. 6 different target variables were estimated using decision tree regression as seen in the scatter plots. (**a**) displays PM_1 data. (**b**) displays $PM_{2.5}$ data. (**c**) displays PM_4 data. (**d**) displays PM_{10} data. (**e**) displays total PM data. (**f**) displays dCn data. On each of the graphs, the green circles represent the testing data, the red circles represent the training data, and the blue line represents a 1:1 ideal correlation between the estimated and actual PM values. The legend displays the coefficient of determination for both the testing and training data.

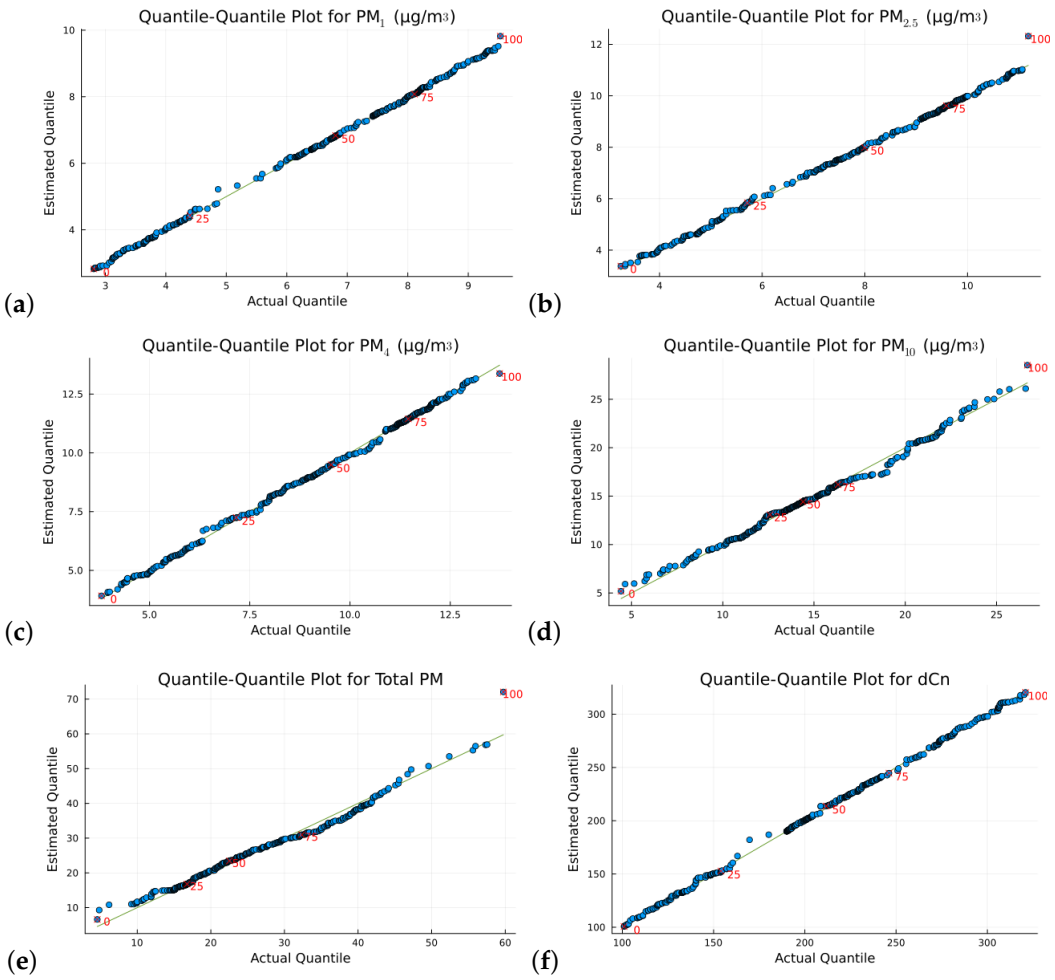

Figure 2. This figure shows multiple quantile-quantile plots comparing the quantile distribution of the PM concentration data measured by the Fidas® Frog sensor against the quantile distribution of the estimated PM concentration provided by the calibration of the LoRa Node. (**a**) displays PM_1 data. (**b**) displays $PM_{2.5}$ data. (**c**) displays PM_4 data. (**d**) displays PM_{10} data. (**e**) displays total PM data.(**f**) displays dCn data.

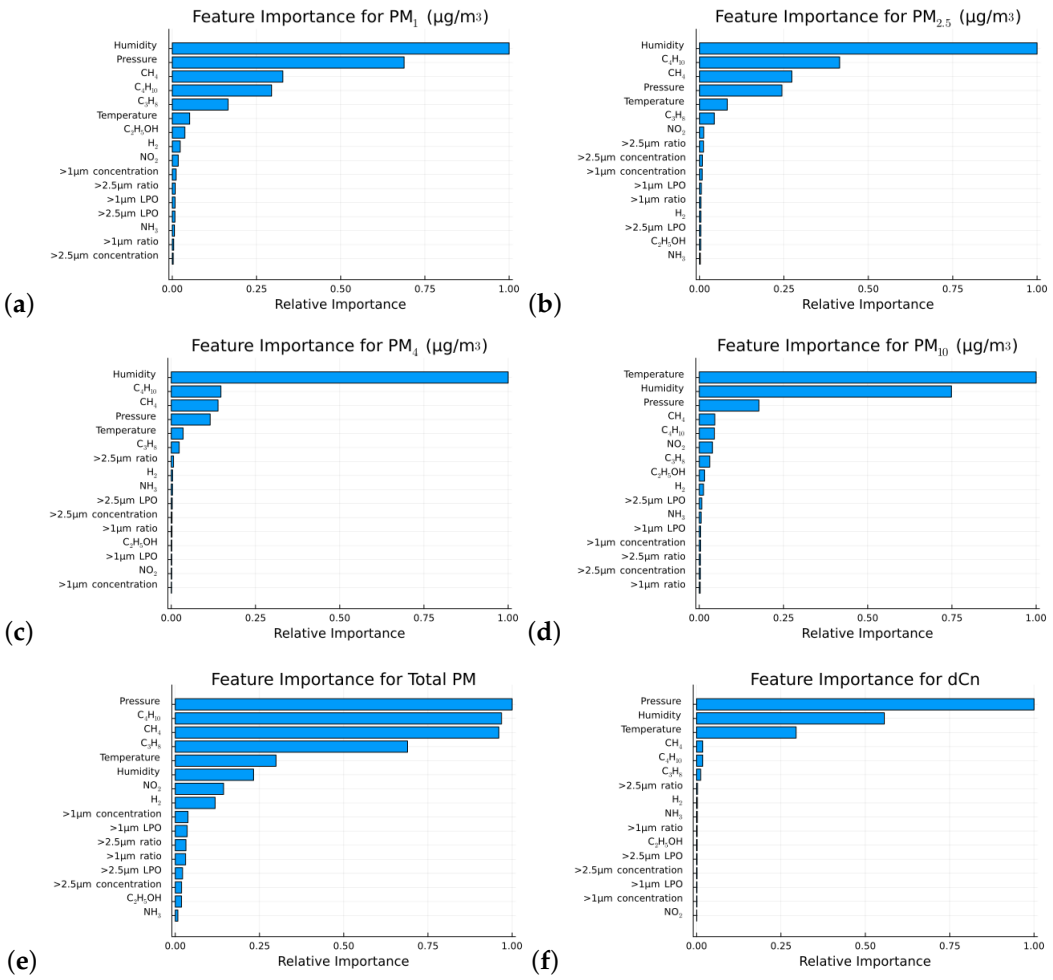

Figure 3. This figure shows multiple bar graphs representing the feature importance in predicting each of the 6 different target variables. (**a**) displays feature importance for PM_1. (**b**) displays feature importance for $PM_{2.5}$. (**c**) displays feature importance for PM_4. (**d**) displays feature importance for PM_10. (**e**) displays feature importance for the total PM. (**f**) displays feature importance for dCn. On each of the graphs, the y axis represents the relative importance of each feature in building the predictive model, and the x axis labels the 16 distinct features from the LoRa Node used in training and testing the model.

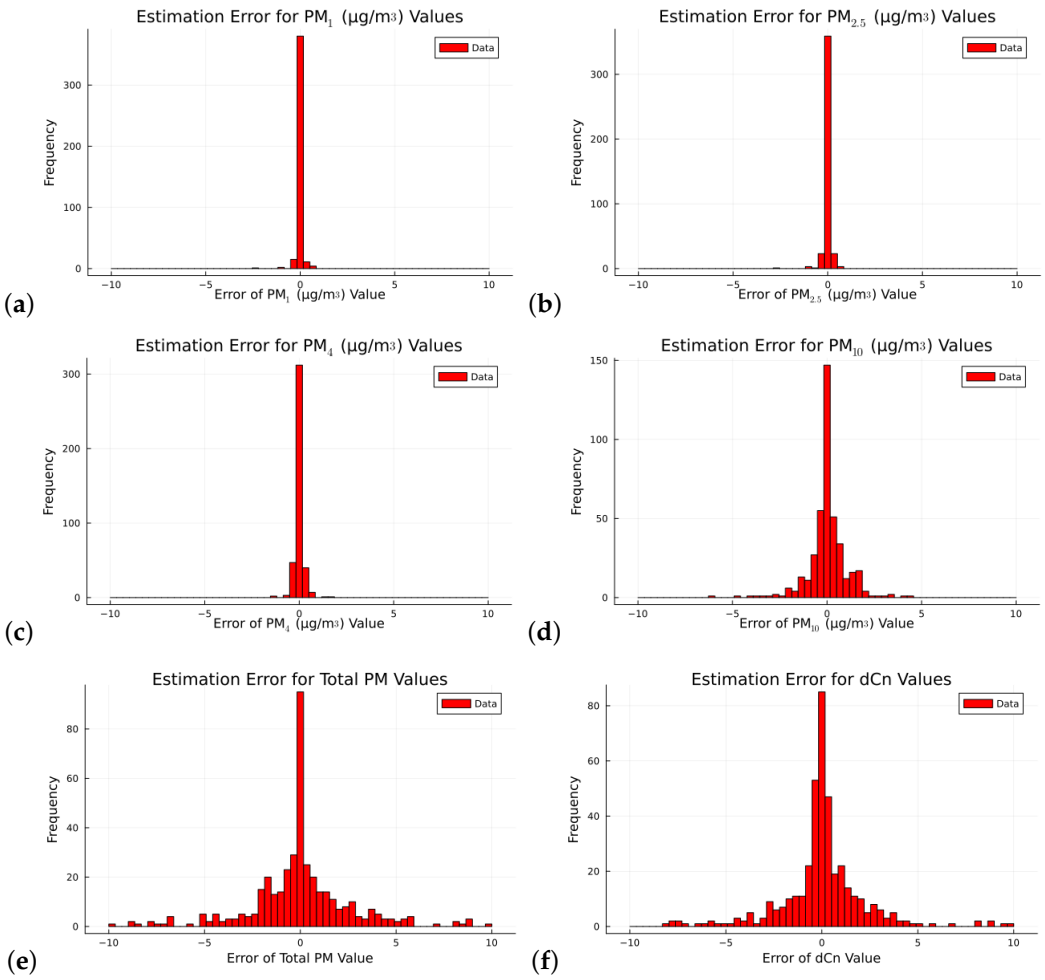

Figure 4. This figure shows multiple histograms representing the estimation error for each of the 6 different target variables, with 60 bins from an error of -10 to 10. To calculate estimation error, the actual value was subtracted by the predicted value. (**a**) displays error for PM_1. (**b**) displays error for $PM_{2.5}$. (**c**) displays error for PM_4. (**d**) displays error for PM_10. (**e**) displays error for the total PM. (**f**) displays error for dCn. On each of the graphs, the y axis represents the frequency for each bin, and the x axis represents the error, which is calculated by taking the actual value minus the predicted value.

Table 3. Comparison of the coefficient of determination between different base models using LoRa Node testing data.

Model	PM_1 Coefficient of Determination	$PM_{2.5}$ Coefficient of Determination	PM_4 Coefficient of Determination	PM_{10} Coefficient of Determination	Total PM Coefficient of Determination	dCn Coefficient of Determination
Linear Regression	0.626	0.662	0.692	0.469	0.367	
Decision Tree	0.987	0.986	0.980	0.879	0.840	
Random Forest Regression	0.992	0.991	0.992	0.936	0.904	
Support Vector Regression	0.056	0.088	0.109	-0.017	-0.060	
Neural Network Regression	-11.478	-12.700	-12.716	-13.693	-6.163	
Gaussian Process Regression	-10.983	-12.168	-12.182	-13.094	-5.866	

4. Discussion

Authors should discuss the results and how they can be interpreted from the perspective of previous studies and of the working hypotheses. The findings and their implications should be discussed in the broadest context possible. Future research directions may also be highlighted.

5. Conclusions

This section is not mandatory, but can be added to the manuscript if the discussion is unusually long or complex.

Author Contributions: For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used "Conceptualization, X.X. and Y.Y.; methodology, X.X.; software, X.X.; validation, X.X., Y.Y. and Z.Z.; formal analysis, X.X.; investigation, X.X.; resources, X.X.; data curation, X.X.; writing—original draft preparation, X.X.; writing—review and editing, X.X.; visualization, X.X.; supervision, X.X.; project administration, X.X.; funding acquisition, Y.Y. All authors have read and agreed to the published version of the manuscript.", please turn to the CRediT taxonomy for the term explanation. Authorship must be limited to those who have contributed substantially to the work reported.

Funding: Please add: "This research received no external funding" or "This research was funded by NAME OF FUNDER grant number XXX." and and "The APC was funded by XXX". Check carefully that the details given are accurate and use the standard spelling of funding agency names at https://search.crossref.org/funding, any errors may affect your future funding.

Data Availability Statement: We encourage all authors of articles published in MDPI journals to share their research data. In this section, please provide details regarding where data supporting reported results can be found, including links to publicly archived datasets analyzed or generated during the study. Where no new data were created, or where data is unavailable due to privacy or

ethical re-strictions, a statement is still required. Suggested Data Availability Statements are available in section "MDPI Research Data Policies" at https://www.mdpi.com/ethics.

Acknowledgments: In this section you can acknowledge any support given which is not covered by the author contribution or funding sections. This may include administrative and technical support, or donations in kind (e.g., materials used for experiments).

Conflicts of Interest: Declare conflicts of interest or state "The authors declare no conflict of interest." Authors must identify and declare any personal circumstances or interest that may be perceived as inappropriately influencing the representation or interpretation of reported research results. Any role of the funders in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript; or in the decision to publish the results must be declared in this section. If there is no role, please state "The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results".

Sample Availability: Samples of the compounds ... are available from the authors.

Abbreviations

The following abbreviations are used in this manuscript:

PM	Particulate Matter
LoRa	Long Range

Appendix A

Appendix A.1

The appendix is an optional section that can contain details and data supplemental to the main text—for example, explanations of experimental details that would disrupt the flow of the main text but nonetheless remain crucial to understanding and reproducing the research shown; figures of replicates for experiments of which representative data are shown in the main text can be added here if brief, or as Supplementary Data. Mathematical proofs of results not central to the paper can be added as an appendix.

Table A1. This is a table caption.

Title 1	Title 2	Title 3
Entry 1	Data	Data
Entry 2	Data	Data

Appendix B

All appendix sections must be cited in the main text. In the appendices, Figures, Tables, etc. should be labeled, starting with "A"—e.g., Figure A1, Figure A2, etc.

References

1. Mukherjee, A.; Agrawal, M. World air particulate matter: sources, distribution and health effects. *Environmental chemistry letters* **2017**, *15*, 283–309.
2. Hamanaka, R.B.; Mutlu, G.M. Particulate matter air pollution: effects on the cardiovascular system. *Frontiers in endocrinology* **2018**, *9*, 680.
3. Ambient (outdoor) air pollution. Available online: https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health, 2022. [Accessed on 27 June 2023].
4. Kelly, F.J.; Fussell, J.C. Size, source and chemical composition as determinants of toxicity attributable to ambient particulate matter. *Atmospheric environment* **2012**, *60*, 504–526.
5. Araujo, J.A.; Nel, A.E. Particulate matter and atherosclerosis: role of particle size, composition and oxidative stress. *Particle and fibre toxicology* **2009**, *6*, 1–19.
6. Ali, S.; Glass, T.; Parr, B.; Potgieter, J.; Alam, F. Low cost sensor with IoT LoRaWAN connectivity and machine learning-based calibration for air pollution monitoring. *IEEE Transactions on Instrumentation and Measurement* **2020**, *70*, 1–11.

7. Lary, D.J.; Remer, L.; MacNeill, D.; Roscoe, B.; Paradise, S. Machine learning and bias correction of MODIS aerosol optical depth. *IEEE Geoscience and Remote Sensing Letters* **2009**, *6*, 694–698.
8. Wijeratne, L.O.; Kiv, D.R.; Aker, A.R.; Talebi, S.; Lary, D.J. Using machine learning for the calibration of airborne particulate sensors. *Sensors* **2019**, *20*, 99.
9. Masic, A.; Bibic, D.; Pikula, B.; Blazevic, A.; Huremovic, J.; Zero, S. Evaluation of optical particulate matter sensors under realistic conditions of strong and mild urban pollution. *Atmospheric Measurement Techniques* **2020**, *13*, 6427–6443.
10. Polley, E.C.; Van der Laan, M.J. Super learner in prediction **2010**.
11. Taghizadeh-Mehrjardi, R.; Hamzehpour, N.; Hassanzadeh, M.; Heung, B.; Goydaragh, M.G.; Schmidt, K.; Scholten, T. Enhancing the accuracy of machine learning models using the super learner technique in digital soil mapping. *Geoderma* **2021**, *399*, 115108.