Journal of Computer Science

Original Research Paper

PM2.5 Prediction Using Homogenous and Heterogenous Ensemble Learning: A Comprehensive Evaluation

[1]Shrabani Medhi and [2]Minakshi Gogoi

[1]Research Scholar, Department of Computer Science and Engineering, Girijananda Chowdhury Institute of Management and Technology, Assam Science and Technology University, Assam, India
[2]Department of Computer Science and Engineering, Girijananda Chowdhury University, Assam, India

Article history
Received: 11-01-2024
Revised: 23-02-2024
Accepted: 08-03-2024

Corresponding Author:
Shrabani Medhi
Research Scholar, Department of Computer Science and Engineering, Girijananda Chowdhury Institute of Management and Technology, Assam Science and Technology University, Assam, India
Email: shrabani26@gmail.com

Abstract: Air pollution is a global issue. PM2.5 is considered to be the most dangerous pollutant. Prediction of PM2.5 concentration is important so that effective measures can be taken beforehand. A multitude of machine learning methodologies have been employed in forecasting PM2.5 levels, utilizing diverse combinations of ensemble classifiers and regressors. However, there are three important issues that need to be addressed in order to construct ensemble classifiers and regressors. The first concern pertains to the selection of the base regressor or classifier technique. The second issue revolves around the choice of the amalgamation technique utilized to assemble multiple regressors or classifiers. Lastly, the third issue relates to determining the optimal number of regressors or classifiers to be ensembled. There is a limited number of related studies addressing these issues. We conducted a comprehensive comparative analysis of ensemble methods, including bagging and boosting for homogeneous ensemble methods and blending and super-learning (stacking) for heterogeneous ensemble methods, to predict PM2.5 concentration levels. Ensemble regressors and classifiers' performance based on these techniques has not been wholly scrutinized in the literature. The issues that we have addressed have not previously undergone scrutiny in the context of PM2.5 concentration prediction. We have used artificial neural networks, support vector machines and decision trees to construct 24 different ensemble regressors and classifiers. In constructing the decision tree, we employed the information gain approach to determine the most suitable property for each node within the generated tree. For SVM we have used the Radial Basis Function (RBF) kernel to create our models. For the ANN model we have used we have used Adam (adaptive moment estimation) optimizer. In each layer, the softmax activation function is used. We have done a model comparison using execution time, accuracy and error metrics on three air pollution datasets of Guwahati City, Delhi City and Kolkata City obtained from the central pollution control board, India. The results reveal that on average heterogenous ensemble techniques, namely, stacking (90-100%) and blending (80-100%) offer better prediction accuracy than homogenous ensemble techniques, namely, bagging (50-98%) and boosting (50-97%) over all the datasets. The root means square error reveals that heterogenous ensemble classifiers and regressors fit better as compared to homogenous classifiers and regressors. In conclusion, our findings indicate that an innovative approach to PM2.5 concentration prediction could incorporate both homogeneous and heterogeneous ensemble techniques into their algorithms. Our ethical data collection approach relies on the open dissemination of information by the central pollution control board, fostering a spirit of shared responsibility in advancing air quality research and public health initiatives.

Keywords: Ensembled Learning, PM2.5 Concentration Prediction, Bagging, Boosting, Stacking, Blending

Introduction

In recent decades, due to the increase in progress, urbanization and improved lifestyle in cities, air pollution has increased at a tremendous rate. We have selected Guwahati as the area of study because it has been recorded as one of the cities with the highest black carbon levels in the world (Barman and Gokhale, 2019). The particulate matter concentration in Guwahati is much higher than the permissible value. The high concentration of PM2.5 is extremely dangerous for both adults and children (Amnuaylojaroen and Parasin, 2023; Oliveira *et al*., 2016). Associations of PM2.5 with health issues such as cardiovascular diseases, respiratory diseases, asthma, cancer, metabolic diseases and obesity can be observed in papers (Evans *et al*., 2013; Laden *et al*., 2006; Rojas-Rueda *et al*., 2013). It is of utmost importance to predict the PM2.5 concentration in advance so that effective measures can be taken beforehand to reduce its extremely harmful effects (Pope III *et al*., 2015; Yang *et al*., 2022). The headquarters of the Pollution Control Board of Assam (PCBA) is Situated in Bamunimaidan, Guwahati. It monitors the city's ambient air quality and has notified that the level of PM2.5 concentration has been well above the prescribed values since 2008 (Kioumourtzoglou *et al*., 2016). Recent studies have revealed that Guwahati falls under one of the cities with the highest concentrations of black carbon (Medhi *et al*., 2023). The major reasons for the poor air quality are rapid urbanization and poor environmental control. Serious steps must be taken to deal with this problem. That is why it is of utmost importance to predict the PM2.5 concentration beforehand so that effective measures can be taken ahead of time.

PM2.5 and Air Quality Standard

PM2.5 refers to particulate matter with a diameter of less than 2.5 microns or less. These particles exist in solid and liquid forms suspended in the air, including examples such as ash, soot and dust (Khyat *et al*., 2023). Because of their exceptionally small size, PM2.5 particles have the ability to deeply penetrate the respiratory tract and easily reach the lungs. Exposure to PM2.5 is linked to a spectrum of health problems, encompassing both short-term effects and long-term consequences. (Kloog *et al*., 2013; Zhang *et al*., 2019). Short-term health effects attributed to PM2.5 exposure encompass irritation of the nose, throat and lungs, accompanied by symptoms such as coughing, sneezing, shortness of breath, runny nose and eye irritation (Banerjee *et al*., 2019). Long-term exposure may lead to serious health problems related to lung function, asthma and heart diseases (Gonzalez *et al*., 2015; Janssen *et al*., 2013; Kim *et al*., 2015). According to the World Health Organization (WHO), approximately three percent of cardiopulmonary deaths and five percent of lung cancer deaths globally are attributed to exposure to PM2.5 (Hadei *et al*., 2017; Han *et al*., 2022; Feng *et al*., 2020). PM2.5 comprises metals, nitrates, sulfates, acids and particles with diverse chemical compositions. Due to its tiny size, PM2.5 has the ability to traverse over longer distances and easily penetrate indoor areas (Martins and Da Graca, 2018). Monitoring stations are responsible for monitoring PM2.5 levels.

All the issues mentioned highlight the significant danger posed by PM2.5. Monitoring stations track PM2.5 levels, which are used to calculate the Air Quality Index (AQI) value. Governments utilize AQI numbers to inform the public about air quality, as shown in Table 1 (Safar-India) (Pope III and Dockery, 2006). An increase in AQI indicates higher air pollution levels and vice versa. As per the Central Pollution Control Board (CPCB), air quality is categorized into 6 stages: Good, satisfactory, moderate, poor, very poor and severe. An AQI value between 0 and 50 is considered good. PM2.5 concentration breakpoints determine air quality depending on the PM2.5 concentrations. For instance, air quality is considered good if PM2.5 concentrations range from 0.0-30.0 μg/m³. Table 1 illustrates the values for each category. This study considers four major pollutants for calculating AQI: Particulate Matter 2.5 (PM2.5), Nitrogen Dioxide (NO_2), Sulfur Dioxide (SO_2) and Particulate Matter 10 (PM10). AQI is computed for all the pollutants, with the maximum value determining the final AQI value for a specific location at a given time. According to the Indian central pollution control board, AQI is computed using the formula provided in Eq. 1 (Beig *et al*., 2010).

$$AQI = \frac{[Ihigh - Ilow]}{[BPhigh - BPlow]} * (CP - BPlow) + Ilow \qquad (1)$$

where:

AQI = Air quality index
CP = Pollutant concentration
$BPhigh$ = Concentration breakpoint that is $\geq CP$
$BPlow$ = Concentration breakpoint that is $<CP$
$Ihigh$ = AQI value corresponding to BP
$Ilow$ = AQI value corresponding to BPlow

Table 1: AQI values of PM2.5

	Air quality index	
Category of AQI	Index value	PM2.5 breakpoints (μg/m³, 24 h average)
Good	000-050	000.0-030.0
Satisfactory	051-100	031.0-060.0
Moderate	101-200	061.0-090.0
Poor	201-300	091.0-120.0
Very poor	301-400	121.0-250.0
Severe	401-500	250.0+000.0

Several computational models based on the machine learning paradigm and soft computing have been utilized for air pollution prediction and analysis. Supervised ML methodologies such as Support Vector Machine (SVM) (Leong *et al*., 2020) and Neural Networks (Cabaneros *et al*., 2019; Maleki *et al*., 2019) have been found to outperform traditional arithmetic methods like ridge regression and logistic regression in terms of accuracy and error metrics. However, ensemble learning methods, which combine various machine learning techniques, have shown promise. Ensemble methods, such as stacking and blending, create committees to enhance predictions, where bagging aims to reduce variance and boosting aims to decrease bias. The use of ensemble learning consistently outperforms single classifiers and regressors (Sun and Li, 2020; Xu *et al*., 2019).

Application of ensemble learning can be found in various sectors like health (Sarmadi *et al*., 2021), agriculture (Zhang *et al*., 2019), finance (Zhu *et al*., 2019) and energy (Wang *et al*., 2019), demonstrating improved performance over single classifiers or regressors. As a result, the advancement of superior ensemble models for both classification and regression tasks has emerged as a vibrant research domain within supervised learning.

Although numerous studies support the superior performance of ensemble techniques over single classifiers or regressors, most papers focus on ensemble-specific classifiers or regressors for air pollution prediction, namely, SVM (Gonzalez *et al*., 2015), NN (Bai *et al*., 2019; Ganesh *et al*., 2018) and DD (Danesh Yazdi *et al*., 2020; Gao *et al*., 2021). There has been a scarcity of research focused on the comparison of ensemble classifiers and regressors utilizing various amalgamation methods, employing either identical or diverse types of base learners, specifically for predicting air quality. Therefore, after reviewing existing literature, we have identified a gap in comprehensive comparative studies evaluating the accuracy of heterogeneous and homogeneous ensemble classifiers and regressors for PM2.5 concentration prediction. As a result, this study undertakes an extensive and comprehensive comparative study of homogeneous and heterogeneous ensemble learning models for both classification and regression in PM2.5 prediction. The following are the objectives of our study:

1. Perform a literature review of recently published research papers on ensembled learning for supervised machine learning tasks, namely, classification and regression in PM2.5 prediction
2. Set up homogenous and heterogenous ensembled classifiers and regressors with NN, DT and SVM using combination techniques, namely, bagging, boosting, blending and stacking
3. Examination and comparison of the models using accuracy, execution time and error metrics for three data sets

The objective of this study is to offer clearer insights into the best suitable ensemble methods for machine learning tasks in predicting the concentration of PM2.5. Furthermore, it provides guidance and support to newcomers in the machine learning domain, aiding them in making well-informed decisions regarding ensemble methods that swiftly deliver optimal and accurate outcomes in PM2.5 concentration prediction scenarios. Lastly, to the best of our knowledge, this research contributes to the existing literature by presenting the 1st comprehensive comparative analysis of ensemble techniques in PM2.5 prediction.

The remaining section of the paper is organized as follows: The "literature review" section presents a review of the related studies done. In the "system evaluation" section, representation of the type of system that is used for performing the processing is represented. Details of the basic supervised machine learning methods and basic and advanced ensembled methods used in the study are discussed. Evaluation criteria used to evaluate the methods are also discussed. In the "proposed architecture” section, the overall model proposed in our study is discussed in detail. The last section is “Results and Discussion" where the outcomes of our experiment are discussed.

In the past decade, many supervised ML techniques and ensemble learning techniques were utilized to solve various problems related to air pollution prediction. In this section, the thematic literature review is conducted to showcase a few of the prominent works in this field. In the paper by Kaewkiriya and Wisaeng (2023), a customer predictive model for investment is developed using the voting ensemble technique, where the neural network is found to be the most effective. A review is also conducted to present the dangerous effects of PM2.5. In the study by Zhu *et al*. (2019), the prediction of air pollutants like sulfur dioxide, PM2.5 and O_3 is performed using regularization and optimization techniques. Datasets from two stations are utilized, with Root Mean Square Error (RMSE) utilized for evaluation of accuracy. The main drawback is that linear regression fails to properly predict unforeseen events and generalization is limited as only two stations are considered. In the research by Betancourt *et al*. (2022), the authors predict ozone using the random forest regression technique. The limitations related to the computational model are studied, including the consideration of only one pollutant and a small dataset size. These are the drawbacks of the system.

DOI: 10.3844/jcssp.2024.931.954

Mitchell *et al*. (2012), the optimal classifier from Bayes is utilized, employing a classification technique based on the assumption that data is conditionally independent from the labeled classes to make calculations more feasible. Usmani *et al*. (2018) employed a comprised model using four supervised machine learning techniques, namely, single-layer perceptron, radial basis function, multiple-layer perceptron and SVM, achieving up to ninety-five percent in predicting stock exchange accuracy. Breiman (1996) conducted a study to obtain a better-performing model using ensemble learning, utilizing bagging predictors with each model assigned the same weight and drawing random subsets for training while boosting was employed to ensemble the models for improved accuracy. Zheng and Zhong (2011) investigated the performance of ensemble learning in time series prediction using ARIMA and ANN, showing that the ensemble model was able to decrease prediction error. Zamani Joharestani *et al*. (2019) studied the feature importance of particulate matter 2.5 to implement XGBoost, Random Forest and deep learning approaches, where XGBoost outperformed other methods.

PM2.5 is considered the most dangerous pollutant, as detailed in Janssen *et al*. (2013); Pope III and Dockery (2006), which studied various diseases caused by PM2.5 exposure. The Global Burden of Disease Project conducted by the World Health Organization (WHO) revealed that a high percentage of health issues result from PM2.5 exposure (Yazdani, 2021). WHO also presented a study indicating that 5% of lung cancer deaths and 3% of cardiopulmonary deaths occur due to PM2.5 exposure. Due to its minute size, PM2.5 particles can penetrate deep into the lungs and remain suspended for prolonged durations. Medhi and Gogoi (2021) emphasized that high PM2.5 levels pose more adverse effects on children, infants and aged adults with heart and lung diseases and asthmatics. Various reasons why PM2.5 is considered very dangerous are studied by Wilson and Suh (1997). Jebamalar and Kamalakannan (2021) predict particulate matter 2.5 using an enhanced technique employing a stacking ensemble machine learning model, outperforming other ensemble techniques. Liu *et al*. (2019) proposed a prediction model to find PM2.5 concentration. It uses a Bagging-Gradient Boosting Decision Tree (GBDT), based on a bagging ensemble learning framework. Our proposed model outdoes SVM and Random Forest (RF) models, better-reducing prediction bias and variance.

In the paper by Jarah *et al*. (2023), a new algorithm for earthquake prediction using machine learning is proposed and observations are made that the neural network model performs better than the other machine learning models. Devi *et al*. (2022), three different modeling techniques, namely MLP, ANN and Bagged Artificial Neural Network (BANN), were used to predict SO_2 pollution trends. The models were evaluated using evaluation criteria, namely, Coefficient Correlation (CC), MAE, RMSE, Nash-Sutcliffe Efficiency (NSE), Willmott Index (WI) and normalized RMSE. It was found that BANN outperformed ANN and MLP. In the review by Liu *et al*. (2019), an introduction is provided on simple prediction models along with their background, advantages, limitations and applications. To enhance prediction ability, a review is conducted on data processing and two auxiliary methods, namely, ensemble learning and metaheuristic optimization. The review also considers spatiotemporal aspects and provides direction on research areas that can be explored for air pollution prediction. A thorough comparison of various ensemble methods is essential to gain a comprehensive understanding of their effectiveness.

A summarized study on prediction done with ensemble techniques is presented in Table 2. The criteria considered for the study include (i) Selection and quantity of base learner (ii) ML task (regression or classification) (iii) Amalgamation technique used and (iv) Evaluation metrics used. As evident from Table 2, the modeling of ensemble classifiers and regressors for predictive purposes has garnered considerable attention in recent studies. Most of the studies (Booth *et al*., 2014; Breiman, 1996; Devi *et al*., 2022; Jacobsen *et al*., 2020; Liu *et al*., 2019; Mabu *et al*., 2015; Weng, 2017) are based on bagging and boosting amalgamation methods. Very few studies (Macchiarulo, 2018; Morales *et al*., 2012; Pasupulety *et al*., 2019; Wang *et al*., 2019) concentrated on the use of stacking and blending combination techniques. Another observation is that most studies compared ensemble classifiers or ensemble regressors but not both. A literature survey reveals that most ML techniques have the ability to be applied for both classification as well as regression, with some of the techniques better suitable for one task over the other. Therefore, a comprehensive comparison should be conducted for both classification and regression using the same base learners. The quantity of base learners used in the studies is also diverse, with some using a fixed number while others using different numbers. To the best of our knowledge, prior studies have not conducted comparisons of ensemble classifiers and regressors utilizing the same amalgamation technique. Based on the points discussed in Table 2, there exists a research gap that warrants a comprehensive comparative study of ensemble classifiers and regressors. This study would consider utilizing the same or different numbers of weak learners, employing various amalgamation techniques for predicting PM2.5 concentration.

Table 2: Comparison of related studies

Articles	Base learners	Number of base learner	Amalga mation method	Machine learning task Classification	Regression	Evaluation criteria
Mabu *et al*. (2015)	MLP	-	BAG	√		-
Devi *et al*. (2022)	MLP, ANN, BANN	-	BAG	√		CC, MAE, RMSE, NSE, WI
Booth *et al*. (2014)	RF	200	BAG		√	CV, MAPE, RMSE
Jacobsen *et al*. (2020)	BMA WALS and LASSO	-	BAG-BOT	√		R^2
Macchiarulo (2018)	SVM and NN	-	STK	√		Cross-validation
Pasupulety *et al*. (2019)	Extra tree and SVM	1-250	STK		√	RMSE
Mehta *et al*. (2019)	Multiple regression, SVM, LSTM	-	-	√		Accuracy
Liu *et al*. (2019)	GBDT, SVM, RF	-	BAG	√		RMSE, MAE, R^2
Morales *et al*. (2012)	LSTM and trees	50-150	BOT, STK	√		F-score, AUC, accuracy
Zamani Joharestani *et al*. (2019)	XGBoost, RF, DL	-	-	√		Accuracy
Wang *et al*. (2019)	RNN	-	STK	√		AUC, accuracy
Weng. (2017)	DT, ANN, RF	-	BAG-BOT		√	MAPE, R^2, RMSE
Gonzalez *et al*. (2015)	SVM	10	MV	√		10-fold CV
De Mello Assis *et al*. (2018)	NN	30	-	√		Recall and Precision
Breiman (1996)						
ML	-	BAG-BOT	√	√		RMSE, MAE, R^2

Background Study

Predictive Models

Based on the studies conducted, we have chosen three machine learning algorithms as the base learners, namely, DT, ANN and SVM, considering their effectiveness in ensemble learning for air pollution prediction.

Decision Tree (DT)

A flowchart-like tree structure is utilized in Decision Trees (DT). This technique employs branching to determine the most probable outcome of a decision. DT is commonly employed for classification tasks due to its advantageous features, including simplicity, interpretability, low computational cost and its graphical representation (Miller *et al*., 2019). The selection of the optimal property for each and every node within the generated tree relies on Information Gain (IG). This entails choosing the attribute with the highest IG as the test attribute for each current node.

The operations on data are as follows. Entropy *En(S)* of a dataset is given in Eq. 2:

$$En(S) = \sum_{i=1}^{k} -q_i log_2 q_i \quad (2)$$

where, *En(S)* represents the entropy of a dataset, *k* represents the number of classes in the dataset, q_i represents the number of instances that belong to class *i*.

Information gain of an attribute *D* is calculated for a collection *S* and is represented by Eq. 3:

$$IG(S,D) = En(S) - \sum \mu \notin values(D) - \frac{S_\mu}{S} En(S\mu) \quad (3)$$

where, *En(S)* represents the entropy of the entire dataset and S_u represents a set of instances that has value *u* for attribute D.

Support Vector Machine (SVM)

SVM is a supervised ML technique that is utilized for the task of both classification and regression (Agarwal *et al*., 2017). Between two data nodes, SVM serves as the linear separator. SVM is employed to discern between two distinct classes within multidimensional environments. SVM is implemented using the following steps.

Let *S* be the training dataset. $S = \{(p_j, q_j, \ldots (p_n, q_n))\}$ where j=1, 2, . . . , *n*. *SVM* denotes the dataset as points in the n-dimensional space. A hyperplane is constructed to divide the space into distinct class labels available in the dataset with the right error margin. The algorithm for SVM optimization is given in Eqs. 4-5:

$$min(d,bw)\frac{1}{2}W^TW+C\sum_{i=1}^{n}wi \quad (4)$$

$$Subject\ to\ yi\left(W^T\theta(xi+b)\geq 1\text{-}wi\right), wi>0 \quad (5)$$

Each vector xi in the dataset is mapped to a function θ within the higher-dimension space. In the higher space, SVM tries to find a linearly separating hyperplane that has the optimal margin. The formula of the kernel function is given by $KF(xi, yi) = \theta(xi)^T\theta(xj)$.

Neural Network (NN)

NN is a network that consists of components that are interrelated. It performs the task of accepting inputs, actuating and forwarding them to the next layer. In our study, we have adopted MLP for NN. MLP is a supervised machine learning technique. It works on a function $f(nn) = N^I \rightarrow N^J$. The training is conducted on dataset S, where I represent the input data dimension and J represents the output dimension. MLP can perform both classification and regression by using a non-linear function approximator. Many optimizers are available. For our study, we have used the Adam (adaptive moment estimation) optimizer. In each layer, the softmax activation function (Eq. 6) is used in each and every layer. The function for mapping each and every layer is represented by Eq. 7:

$$\sigma(z) = \frac{e^{zi}}{\sum_{j=1}^{k} e^{zj}} \quad (6)$$

where, σ is the softmax, z is the input vector, $e^{zi}$is the standard exponential function for the input vector, k is the number of classes in the classifier and $e^{zj}$is the standard exponential function for the output vector:

$$MF = W^{[l]T} \times a^{l-1} + b^l \quad (7)$$

where, $W^{[l]}$ is the weight matrix and $b^l$is the bias.

Ensemble Methods (EM)

EM techniques are responsible for merging multiple single regressors and classifiers to form a committee. It is done to achieve better decisions and accurate results compared to a single regressor or classifier (Guzman *et al*., 2015; Srisuradetchai and Panichkitkosolkul, 2022). In the ensemble process, many diverse single classifiers or regressors are trained independently using the same or different dataset. The same parameters are not used. The final prediction is determined by averaging the outputs of all individual base classifiers or regressors. There are three important issues that need to be put into consideration while creating an ensemble classifier and regressor model. (1) The type of regression and classification method that should be used out of the many methods that are available. (2) The number of base learners that should be used to obtain better accuracy. (3) The combination technique that should be applied to amalgamate the results of single base learners so as to obtain the final output. Some of the basic and advanced combination techniques are discussed below.

Basic Ensemble Techniques

We are going to discuss three basic ensemble techniques: (i) Weighted Averaging (WAv) (ii) Max Voting (MV) (iii) Averaging

Weighted Averaging (WAv)

WAv represents the extension of the averaging technique. Various types of weights, for example (0.5, 0.2, 0.7, etc.,) are assigned to each of the models (M1, M2, M3, etc.,) based on the importance of each model for prediction. The final prediction (Fwa) is given in Eq. 8:

$$Fwa = ((0.5 \times z1) + (0.2 \times z2) + (0.7 \times z3) + \cdots \quad (8)$$

where, $z1$, $z2$, $z3$, … are the forecasting output of the models $M1$, $M2$, $M3$, …

Max Voting (MVo)

MVo is primarily used for classification tasks. Several single classifiers are used to train the dataset. The output of each of the individual classifiers is used as a 'vote. The final output is determined by taking the majority vote. The training of the individual models is done using the same training dataset and applying the same testing set. The ultimate prediction is determined by aggregating a majority vote among the individual predicted outputs.

Averaging

The technique of averaging can be applied to both classification and regression tasks. It bears similarity to the Majority Voting (MVo) technique. In this approach, the final output is determined by calculating the average of all forecasting outputs from the individual classifiers or regressors. Here, each individual model is trained and tested separately using the same dataset. The ultimate Forecasting (Fa) using averaging is given in Eq. 9:

$$Fa = \sum_{i=1}^{n} \frac{(z1+z2+\cdots)}{n} \quad (9)$$

where, $z1$, $z2$, … are the forecasting outputs of the distinct models $M1$, $M2$, … n is the number of individual classifiers used.

Advanced Ensemble Learning Techniques

Here we will discuss four unconventional amalgamation techniques in brief.

DOI: 10.3844/jcssp.2024.931.954

Bagging (BAGG)

BAGG is also known as bootstrap aggregating. The forecast of several models (e.g., n number of decision trees) is combined to achieve the final result. The bootstrapping technique involves creating multiple subsets, referred to as bags, of the original training dataset with replacement. Bags aid in acquiring an unbiased representation of the entire dataset (Tsai *et al*., 2014). The size of the bags is less than the original dataset. The variance of the models can be decreased by using bagging.

Boosting (BOTT)

Another name for BOTT is meta-algorithm. It is a progressive process wherein each succeeding model attempts to correct the errors, weaknesses, etc., of the preceding model. The performance of the successive model is dependent on the preceding model (Mayr *et al*., 2014). The aim of BOTT is to reduce the bias of the models. A strong learner is formed by lumping together several weak learners. Single models may perform better for some parts of the dataset. They may not obtain necessarily better accuracy for the entire dataset. Each model progressively enhances the performance of the entire ensemble. Examples of BOTT algorithms include CatBoost, AdaBoost, gradient boosting machine, extreme gradient boosting and others.

Stacking (STK)

In Stacking (STK), the individual predictions from multiple models are utilized to construct the final model (Chaurasia and Pal, 2021). The final model is used to perform prediction on the testing dataset. The aim of stacking is to improve the forecasting power of a regressor or classifier (Khairalla *et al*., 2018). To implement stacking in our study mlens library was used. The process of stacking is represented in Eq. 10:

$$Fstk = \sum_{i=1}^{n} wifi(x) \tag{10}$$

Here, *Fstk* represents the final output, *n* represents the total number of models, *wi* is the weight vector learned by the models *M*1, *M*2, …, *Mn* and output of the individual models *M*1, *M*2, …, *Mn* is represented by *fi*(*x*).

Blending (BLD)

BLD is analogous to the stacking technique. In stacking test dataset is used for prediction. While in blending a validation dataset from the training dataset is used for prediction. The final model for prediction on the testing dataset is obtained by using the outcome of the predicted dataset and the validation dataset.

Materials and Methods

All our experiments are conducted on an 11th gen intel (R) core (TM) i7-1165G7 @ 2.80GHz, 2803 Mhz, 4 core (s), 8 logical processor(s) running on Microsoft windows 11, home single language. Data preprocessing, time series evaluation and modeling of ensemble classifiers and regressors using base learners ranging from 1-200 are implemented using Python programming and its various libraries. Performance evaluation of the models is done using sklearn metrics. Figure 1 depicts our proposed framework. The combination methods that we have used are BAG, BOT, STK and BLD. The base learner models that we have adopted are DT, SVM and NN. These concepts are already discussed in the previous section. We have developed both homogeneous and heterogeneous ensemble regressors and classifiers for predicting PM2.5 concentration. Subsequently, we compare their accuracy and error metrics. There are three phases of our proposed framework: (i) Data pre-processing (ii) Construction of homogeneous and heterogenous classifiers and regressors and (iii) Comparison of error metrics and accuracy of the models.

Data Source

For Guwahati city, Delhi city and Kolkata city, the Continuous Ambient Air Quality Monitoring Station (CAAQMS) data is obtained from the Central Pollution Control Board, India Central Pollution Control Board (CPCB). A total of 3 years of data from 2019-2022 is collected for each city. The features utilized in the study are provided in Tables 3-4 present descriptive statistics for the Guwahati city dataset, including meteorological conditions, criteria gases and particulate measures such as count, mean, standard deviation, minimum, 25th percentile, 50th percentile, 75th percentile, maximum, skewness, kurtosis and variance. Notably, there is no high value of skewness in the data, suggesting no sharp increases in the dataset. However, the high value of kurtosis in PM2.5 indicates the presence of data discontinuities. Similar analyses are conducted for the datasets of Delhi city and Kolkata city. The objective is to predict the 1-h ahead PM2.5 concentration for both classification and regression and descriptive statistics have been performed for all datasets.

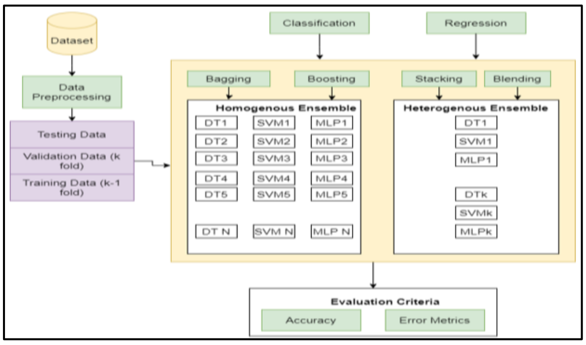

Fig. 1: Proposed framework

Stationarity Check

It is crucial to verify whether the time series data is stationary or not. We have used three methods to check the stationarity of the time series data. From the methods described below it is found that our data is stationary.

Time Plots

For time series analysis, time plots are very important as they are used as a descriptive tool that may show both seasonality and trend, outliers and discontinuities. This allows us to make better decisions in choosing the appropriate technique to perform the prediction. The time plots of PM2.5 for the Guwahati city dataset are shown in Fig. 2. The time plots of all the features used are created to check for stationarity.

From the plots, it can be observed that the distribution for each of the data is non-linear. A time series is stationary if the variance remains the same over time. The plot in Fig. 2 indicates the stationarity of the data.

Gaussian Distribution

Data is stationary if it follows a Gaussian distribution. The histogram of the Guwahati data is plotted in Fig. 3 and it shows a Gaussian distribution indicating the stationarity of data.

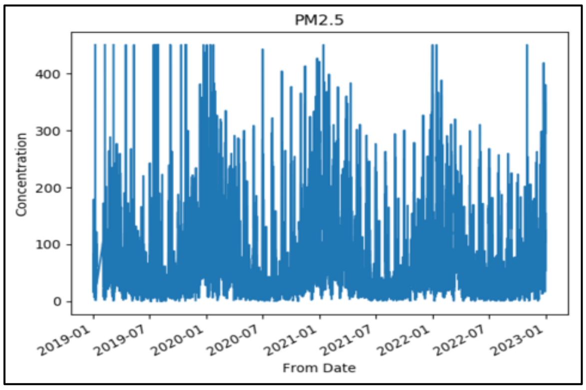

Fig. 2: Time plots of the Guwahati city data

Table 3: Summary of measurement site and observed variables

Measurement site	Type	Variables
Guwahati city	Meteorological	Relative humidity, wind
Data count:	conditions	speed, wind direction,
33067		temperature, Rainfall,
Delhi city		pressure
Data count:	Criteria gases	NO_2, SO_2, NO, NOx,
59030		NH_3, CO, Ozone,
Kolkata city		Benzene, Eth-Benzene,
Data count:	Particulates	MP-Xylene
48953		PM2.5, PM10

Table 4: Dataset descriptive statistics

Parameters	Base count	mean	Std	min	25%	50%	75%	max	skew	kurt	Var
PM2.5	33067	59.3	61.9	0.0	18.0	36.0	80.0	450.0	2.2	6.7	3841.9
PM10	33067	114.0	123.0	0.5	34.0	71.0	145.0	1000.0	2.6	10.7	15200.0
NO	33067	14.9	25.7	0.0	2.6	5.4	14.9	398.6	2.9	9.6	665.0
NO^2	33067	9.8	11.1	0.0	2.6	5.7	12.1	107.0	3.0	12.4	123.2
Nox	33067	23.6	40.0	0.0	5.2	6.2	22.0	347.4	3.1	11.3	1607.9
NH3	33067	7.9	6.4	0.0	3.4	6.1	10.8	161.0	2.0	15.0	41.1
SO^2	33067	17.1	7.0	1.5	11.7	15.3	22.3	172.6	1.7	14.7	49.8
CO	33067	0.7	0.6	0.0	0.3	0.5	0.9	6.6	2.3	7.2	0.4
Ozone	33067	25.8	18.0	0.0	17.2	20.0	26.9	171.2	2.4	7.0	327.0
Benzene	33067	1.8	12.8	0.0	0.1	0.8	1.8	491.5	21.5	562.2	164.7
Eth-Benzene	33067	3.2	20.7	0.0	0.2	3.2	3.2	492.7	17.4	329.7	432.4
MP-Xylene	33067	2.5	20.9	0.0	0.1	0.5	2.5	492.7	17.3	327.0	438.6
WS	33067	1.1	00.7	0.0	0.5	0.9	1.4	23.7	2.4	31.1	0.5
WD	33067	153.8	59.0	13.9	103.6	141.6	197.0	328.1	0.6	-0.7	3487.5
SR	33067	241.9	187.9	0.0	90.3	241.9	256.5	923.2	0.9	0.7	35327.0
BP	33067	971.8	34.2	703.1	971.8	971.8	995.7	1011.8	-2.3	7.3	1175.3
AT	33067	24.7	4.9	6.1	21.6	25.1	28.1	38.2	-0.4	-0.2	24.4
RF	33067	0.0	0.2	0.0	0.0	0.0	0.00	3.7	6.2	60.3	0.0

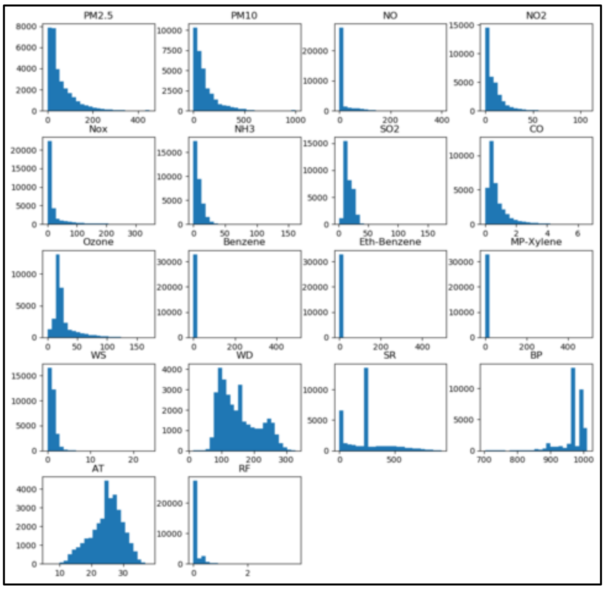

Fig. 3: Histogram of Guwahati dataset

Table 5: Mean and variance of partitioned data of Guwahati dataset

	Mean-variance	
Partition 1	49.36	6506.42
Partition 2	49.15	5971.78

Summary Statistics

For all datasets, the data is divided into two parts and a thorough examination is conducted to identify any notable and noteworthy variances in summary statistics. The mean and variance of the two partitions for the Guwahati dataset are shown in Table 5. It can be seen that the mean and variance of the two partitioned data are almost the same indicating the stationarity of the data.

Data Preprocessing

The performance of an ML model is often influenced by the data preprocessing step (Wilson and Suh, 1997). The data preprocessing part is divided into two parts (i) Data cleaning and (ii) Data transformation. An imputer function is used to perform the process of interpolation.

Techniques like filling in missing values and rectifying inconsistencies are pivotal for data cleaning, ensuring robustness in subsequent modeling endeavors. Furthermore, outlier identification through outlier classification aids in pinpointing maximum and minimum outliers, offering insights into data distribution characteristics. For instance, upon analysis, it was noted that "rainfall” exhibited the highest number of missing values, while “relative humidity, wind speed and pressure” displayed relatively fewer instances of missing data. Imputation techniques, employing strategies such as mean value interpolation, are utilized to address missing data instances effectively. Moreover, outliers are detected using the Inter Quantile Range (IQR) method, with quantile-based flooring and capping approaches employed for outlier management.

Given the diverse units of multiple input variables, normalization emerges as a critical step to standardize data attributes onto a consistent scale. Normalization is done so that an attribute having lesser significance with a large scale doesn’t suppress another attribute of greater significance. Min-max scaler is used for normalization.

It involves subtracting the minimum value from the attribute and then dividing it by the range. The difference is defined by the difference between the maximum and the minimum value. The mathematical formula used to normalize the dataset is given in Eq. 11 (Patro *et al*., 2015).

$$\text{Xscaled} = \frac{\text{X}-\text{Xmin}}{Xmax-Xmin} * (D - C) + C \quad (11)$$

Construction of Homogeneous and Heterogeneous Classifiers and Regressors

Table 6, for the homogenous ensemble experiment, the number of base learners used ranges from 1-100. We have constructed 12 homogenous ensemble classifiers and regressor models with the help of bagging and boosting techniques. As seen in Table 7, we have constructed 12 heterogeneous ensemble classifier and regressor models based on stacking and blending.

Table 6: Homogenous ensemble classifier and regressor models

Base learner	Classification BAGG	Classification BOTT	Regression BAGG	Regression BOTT	Number of estimator
DT	√	√	√	√	1-100
SVM	√	√	√	√	1-100
MLP	√	√	√	√	1-100

(BAGG: Bagging; BOTT: Boosting)

Table 7: Heterogenous ensemble classifier and regressor models

Base learner	Meta estimator or	Classification STK	Classification BLD	Regression STK	Regression BLD
DT-SVM	MLP	√	√	√	√
SVM-MLP	DT	√	√	√	√
MLP-DT	SVM	√	√	√	√

(STK: Stacking; BLD: Blending)

In our paper, we implemented a 10-fold Cross-Validation (10-CVa) approach to achieve a refined evaluation of accuracy during training. Utilizing the (10-CVa) technique involved partitioning the training dataset into 10 distinct subsets, with nine of these sub-sets employed in training each model. Concurrently, the remaining subset (1) was designated as the test data. This iterative process was replicated ten times, aligning with the number of folds in the (10-CV) approach. Notably, 80% of each dataset was dedicated to the training phase, while the residual 20% was exclusively reserved for testing purposes.

Model Evaluation

The performance of classifiers and regressors can be measured by using various evaluation metrics (Nti *et al*., 2020). We have selected twelve accuracy and closeness evaluation metrics for our study which is given in Table 8. These metrics were used based on the effectiveness of these metrics for classification and regression purposes.

Results and Discussion

Here, we present the results and discuss the findings of our experiment.

Table 8: Evaluation metrics used

Acronym	Full name	Formula
MAE	Mean absolute error	$MAE = \frac{1}{2}\sum_{i=1}^{n}(yi - yp)$
RMSE	Root mean square error	$RMAE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(yi - yp)}$
R^2	F1-score	$R^2 = 1 \frac{RSS}{TSS}$
STD	Standard deviation	
AUC	Area under ROC curve	$AUC = \int_0^1 \frac{TP}{(TP+FN)} d \frac{FP}{(FP+TN)}$
EVS	Explained variance score	
MedAE	Median absolute error	$MedAE(y,y) = median(\lvert y1 - y1), \ldots, (\lvert yn - yn \rvert)$
RMSLE	Root mean squared logarithmic error	$RMSLE = MSE\begin{pmatrix}\log(yn+1) \\ ,\log(yn+1)\end{pmatrix}$

Analysis of Homogenous Ensemble Classifier by Bagging and Boosting

The forecasting accuracy of the homogenous ensemble classifier by bagging and boosting over the Guwahati, Delhi and Kolkata datasets is shown in Figs. 4-6, respectively. The number of base learners is represented by the X-axis and the Y-axis represents the prediction accuracy. We observe that the Decision Tree Boosting Classifier (DTBoC) and Decision Tree Bagging Classifier (DTBC) with an accuracy above 99% with (20-100) estimators outperformed all other bagging and boosting classifiers over the Guwahati, Delhi and Kolkata dataset. The performance of the Multi-Layer Perceptron Bagging Classifier (MLPC) and Multi-Layer Perceptron Boosting Classifier (MLPBoC) is lower than bagging and boosting obtained using Decision Tree (DT) but higher than bagging and boosting obtained using Support Vector Machine (SVM) for all the datasets. MLP ensemble using a bagging classifier obtained an accuracy of (94-98)% over the Guwahati dataset, (92-100)% over the Delhi dataset and 100% over the Kolkata dataset. SVM bagging classifier recorded (88-89)% over Guwahati, (92-93)% over Delhi and (96-97)% over Kolkata. The performance of the SVM Boosting Classifier (SVMBoC) is lowest for all the datasets with an accuracy of an average of 52% for all the range of estimators. It is found that for the DT ensemble classifier, the accuracy increases as the estimator quantity increases. Thus, we can conclude that to achieve higher and improved accuracy, it is advisable to increase the number of estimators for the decision tree ensemble classifier. However, in the case of the SVM ensemble classifier, the accuracy remains unaffected by the number of estimators utilized. Therefore, we can conclude that the accuracy of the SVM ensemble classifier is not impacted by the number of estimators employed. The same trend is seen for all the datasets. The variation in accurately predicting the PM2.5 concentration using homogenous classifiers over different datasets suggests that homogenous ensemble methods depend on the data that is being analyzed which supports the literature (Feng *et al*., 2016; Triana and Osowski, 2020).

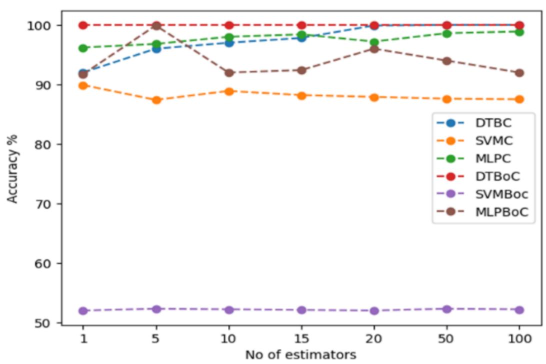

Fig. 4: Bagging and boosting classifier accuracy of the Guwahati dataset

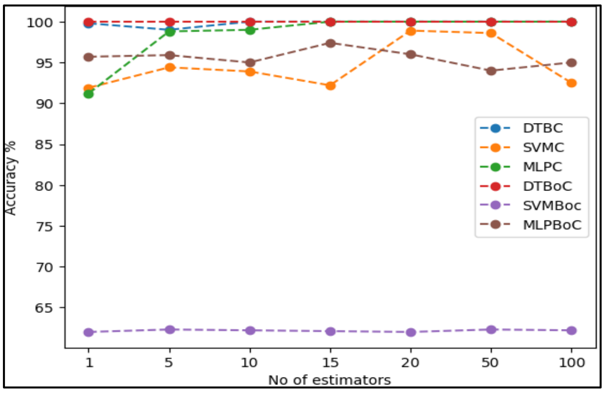

Fig. 5: Bagging and boosting classifier accuracy of the Delhi dataset

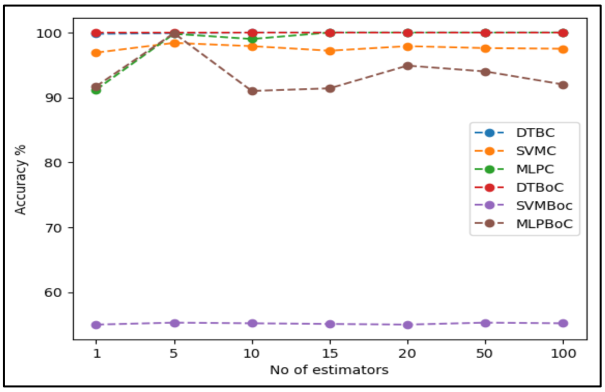

Fig. 6: Bagging and boosting classifier accuracy of Kolkata dataset

Merely relying on accuracy scores is insufficient to evaluate the performance of classifier and regressor models. That is why we calculated some more error metrics. Tables 9-14 The performance of MLP, DT and SVM ensemble classifiers built on bagging and boosting over Guwahati, Delhi and Kolkata datasets is shown using the error metrics.

For DT ensemble classifiers with (1-100) estimators the Area Under Curve (AUC) falls between (0.91-1) for all the datasets. This explains the best accuracy of DT ensemble classifiers that are obtained among all the homogenous ensemble classifiers. The R2 score of DT ensemble classifiers demonstrates a balance between the recall and precision values of the models. DT ensemble classifier with (10-100) estimators obtained RMSE and MAE values of approximately 0.000. This observation proves that the accuracy of DT ensemble classifiers is significantly influenced by the quantity of estimators. The AUC values of (0.94-0.99), (1.00-1.00) and (1.00-1.00) over Guwahati, Delhi and Kolkata datasets, respectively are obtained with (1-100) estimators. This implies that the MLP bagging classifier outperformed the MLP boosting classifier in PM2.5 concentration prediction for all the datasets.

Overall, it is concluded that the SVM ensemble classifier performed low compared to DT and MLP ensemble classifiers for all the datasets. It is affirmed by the literature (Medhi *et al*., 2023; Obodoeze *et al*., 2021; Srisuradetchai and Panichkitkosolkul, 2022).

Table 9: Error metrics of bagging ensemble classifier over the Guwahati dataset

Models	No. of estimators	MAE	RMSE	R^2	MN	STD	PS	RC	AUC	TrT	TeT
DTBC	01	0.074	0.342	0.702	0.841	0.046	0.927	0.926	0.916	0.096	0.010
SVMBC		0.274	0.462	0.548	0.849	0.041	0.964	0.886	0.857	0.251	0.005
MLPBC		0.062	0.278	0.761	0.941	0.047	0.949	0.942	0.937	3.034	0.267
DTBC	05	0.037	0.258	0.875	0.871	0.036	0.954	0.963	0.957	0.356	0.036
SVMBC		0.264	0.471	0.478	0.848	0.041	0.851	0.858	0.832	0.524	0.026
MLPBC		0.051	0.362	0.815	0.964	0.026	0.946	0.943	0.936	10.573	0.726
DTBC	10	0.028	0.248	0.915	0.873	0.027	0.964	0.975	0.963	0.579	0.057
SVMBC		0.217	0.471	0.548	0.854	0.047	0.867	0.884	0.856	0.972	0.051
MLPBC		0.026	0.278	0.913	0.967	0.036	0.963	0.971	0.962	19.745	2.746
DTBC	15	0.024	0.227	0.938	0.864	0.035	0.974	0.975	0.978	0.652	0.067
SVMBC		0.238	0.461	0.515	0.852	0.046	0.856	0.886	0.843	2.165	0.032
MLPBC		0.024	0.258	0.934	0.976	0.025	0.971	0.975	0.975	32.786	3.647
DTBC	20	0.008	0.028	0.964	0.873	0.035	0.986	0.981	0.983	0.843	0.095
SVMBC		0.256	0.472	0.503	0.859	0.041	0.857	0.876	0.846	2.634	0.061
MLPBC		0.034	0.274	0.906	0.973	0.025	0.963	0.964	0.964	49.637	4.764
DTBC	50	0.000	0.000	1.000	0.885	0.026	1.000	1.000	1.000	2.874	0.267
SVMBC		0.216	0.475	0.506	0.858	0.043	0.857	0.876	0.836	4.674	0.245
MLPBC		0.026	0.274	0.936	0.961	0.036	0.973	0.974	0.975	139.856	8.462
DTBC	100	0.000	0.000	1.000	0.885	0.025	1.000	1.000	1.000	4.846	0.317
SVMBC		0.251	0.417	0.514	0.851	0.045	0.856	0.885	0.846	11.486	0.456
MLPBC		.006	0.092	0.968	0.963	0.027	0.983	0.983	0.974	270.547	14.573

(MN: Mean; STD: Standard Deviation; PS: Precision; RC: Recall; TrT: Training Time; TeT: Testing Time)

Table 10: Error metrics of boosting ensemble classifier over Guwahati dataset

Model	No of estim ators	MAE	RMSE	R^2	MN	STD	PS	RC	AUC	TrT	TeT
DTBoC	01	0.053	0.256	0.615	0.852	0.032	0.945	0.926	0.928	0.062	0.004
SVMBoC		0.367	0.683	-0.593	0.634	0.036	0.648	0.647	0.510	0.365	0.007
MLPBoC		0.128	0.318	0.418	0.834	0.071	0.867	0.871	0.852	0.014	0.002
DTBoC	05	0.032	0.167	0.837	0.824	0.027	0.961	0.975	0.965	0.276	0.015
SVMBoC		0.387	0.635	-0.518	0.665	0.031	0.647	0.648	0.500	1.256	0.018
MLPBoC		0.165	0.383	0.359	0.813	0.078	0.885	0.862	0.842	0.163	0.025
DTBoC	10	0.000	0.000	1.000	0.846	0.027	1.000	1.000	1.000	0.695	0.047
SVMBoC		0.365	0.623	-0.528	0.632	0.031	0.648	0.648	0.506	1.432	0.013
MLPBoC		0.185	0.318	0.524	0.851	0.046	0.891	0.893	0.875	0.274	0.025
DTBoC	15	0.000	0.000	1.000	0.876	0.026	1.000	1.000	1.000	0.639	0.048
SVMBoC		0.362	0.654	-0.563	0.614	0.038	0.643	0.647	0.529	1.365	0.024
MLPBoC		0.128	0.398	0.529	0.895	0.047	0.895	0.893	0.863	0.254	0.035
DTBoC	20	0.000	0.000	1.000	0.893	0.027	1.000	1.000	1.000	0.828	0.074
SVMBoC		0.369	0.632	-0.539	0.617	0.032	0.647	0.648	0.532	1.386	0.015
MLPBoC		0.113	0.359	0.521	0.852	0.058	0.882	0.886	0.864	0.263	0.009
DTBoC	50	0.000	0.000	1.000	0.893	0.016	1.000	1.000	1.000	1.945	0.164
SVMBoC		0.360	0.632	-0.518	0.616	0.036	0.645	0.642	0.527	1.328	0.036
MLPBoC		0.115	0.381	0.475	0.858	0.014	0.861	0.875	0.847	0.241	0.017
DTBoC	100	0.000	0.000	1.000	0.893	0.027	1.000	1.000	1.000	5.764	0.375
SVMBoC		0.378	0.643	-0.528	0.615	0.032	0.648	0.648	0.513	1.742	0.027
MLPBoC		0.127	0.391	0.427	0.862	0.028	0.913	0.862	0.863	0.421	0.013

(MN: Mean; STD: Standard Deviation; PS: Precision; RC: Recall; TrT: Training Time; TeT: Testing Time)

Table 11: Error metrics of bagging ensemble classifier over Delhi dataset

Models	No. of estimate tors	MAE	RMSE	R^2	MN	STD	PS	RC	AUC	TrT	TeT
DTBC	1	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.094	0.011
SVMBC		0.082	0.271	0.625	0.915	0.047	0.964	0.917	0.895	0.097	0.012
MLPBC		0.081	0.261	0.638	0.973	0.028	0.848	0.914	0.931	0.926	0.042
DTBC	5	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.279	0.026
SVMBC		0.062	0.231	0.715	0.915	0.031	0.983	0.937	0.927	0.432	0.052
MLPBC		0.015	0.112	0.947	0.987	0.011	0.987	0.996	0.995	4.235	0.267
DTBC	10	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.319	0.061
SVMBC		0.072	0.261	0.657	0.916	0.025	0.984	0.926	0.904	0.748	0.138
MLPBC		0.001	0.052	0.982	0.995	0.010	0.994	0.994	1.000	9.451	0.724
DTBC	15	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.562	0.037
SVMBC		0.074	0.263	0.663	0.916	0.026	0.982	0.926	0.904	2.452	0.216
MLPBC		0.000	0.000	1.000	0.995	0.011	1.000	1.000	1.000	13.231	0.846
DTBC	20	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	2.519	0.067
SVMBC		0.021	0.143	0.900	0.963	0.023	0.996	0.973	0.973	5.273	0.275
MLPBC		0.000	0.000	1.000	0.991	0.005	1.000	1.000	1.000	110.861	1.156
DTBC	50	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	1.263	0.165
SVMBC		0.021	0.143	0.900	0.915	0.024	0.993	0.973	0.974	4.174	0.538
MLPBC		0.000	0.000	1.000	0.993	0.010	1.000	1.000	1.000	56.372	3.178
DTBC	100	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	2.673	0.265
SVMBC		0.072	0.263	0.678	0.915	0.031	0.985	0.927	0.904	9.375	1.219
MLPBC		0.000	0.000	1.000	0.991	0.010	1.000	1.000	1.000	112.362	8.271

(MN: Mean; STD: Standard Deviation; PS: Precision; RC: Recall; TrT: Training Time; TeT: Testing Time)

Table 12: Error metrics of boosting ensemble classifier over the Delhi dataset

Model	No. of estimators	MAE	RMSE	R^2	MN	STD	PS	RC	AUC	TrT	TeT
DTBoC	01	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.056	0.002
SVMBoC		0.384	0.618	-0.617	0.594	0.065	0.000	0.615	0.502	0.334	0.016
MLPBoC		0.026	0.275	0.738	0.915	0.058	0.947	0.946	0.935	0.042	0.003
DTBoC	05	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.074	0.010
SVMBoC		0.321	0.623	-0.528	0.595	0.062	0.000	0.617	0.509	1.167	0.105
MLPBoC		0.064	0.157	0.838	0.936	0.047	0.962	0.963	0.935	0.264	0.053
DTBoC	10	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.075	0.010
SVMBoC		0.364	0.636	-0.624	0.593	0.062	0.000	0.614	0.500	1.135	0.103
MLPBoC		0.012	0.163	0.836	0.931	0.048	0.968	0.967	0.956	2.354	0.064
DTBoC	15	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.074	0.004
SVMBoC		0.376	0.639	-0.684	0.596	0.063	0.000	0.614	0.502	1.763	0.052
MLPBoC		0.052	0.264	0.836	0.947	0.048	0.936	0.957	0.967	4.265	0.562
DTBoC	20	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.078	0.003
SVMBoC		0.386	0.624	-0.685	0.594	0.067	0.000	0.615	0.503	1.756	0.056
MLPBoC		0.016	0.163	0.925	0.936	0.036	0.935	0.985	0.985	7.354	0.724
DTBoC	50	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.067	0.004
SVMBoC		0.386	0.643	-0.673	0.598	0.064	0.000	0.613	0.500	2.845	0.036
MLPBoC		0.031	0.182	0.857	0.935	0.038	0.964	0.967	0.962	14.764	0.823
DTBoC	100	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.067	0.004
SVMBoC		0.317	0.617	-0.634	0.597	0.064	0.000	0.614	0.504	5.756	0.052
MLPBoC		0.023	0.183	0.873	0.952	0.025	0.952	0.964	0.967	53.276	4.278

(MN: Mean; STD: Standard Deviation; PS: Precision; RC: Recall; TrT: Training Time; TeT: Testing Time)

Table 13: Error metrics of bagging ensemble classifier over the Kolkata dataset

Models	No. of estimators	MAE	RMSE	R^2	MN	STD	PS	RC	AUC	TrT	TeT
DTBC	1	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	000.096	00.007
SVMBC		0.036	0.175	0.874	0.986	0.016	0.984	0.967	0.973	000.165	00.008
MLPBC		0.000	0.000	1.000	0.992	0.005	1.000	1.000	1.000	000.563	00.060
DTBC	5	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	000.253	00.025
SVMBC		0.018	0.116	0.942	0.986	0.016	0.988	0.984	0.997	000.452	00.019
MLPBC		0.000	0.000	1.000	0.991	0.002	1.000	1.000	1.000	002.762	00.245
DTBC	10	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	000.264	00.036
SVMBC		0.025	0.158	0.905	0.986	0.010	0.974	0.975	0.985	000.976	00.038
MLPBC		0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	006.263	00.527
DTBC	15	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	000.462	00.078
SVMBC		0.025	0.164	0.895	0.985	0.007	0.963	0.975	0.972	001.473	00.063
MLPBC		0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	012.473	00.623
DTBC	20	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	000.567	00.105
SVMBC		0.028	0.156	0.905	0.986	0.010	0.978	0.977	0.986	000.728	00.115
MLPBC		0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	011.473	01.538
DTBC	50	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	002.836	01.067
SVMBC		0.023	0.163	0.892	0.984	0.009	0.963	0.972	0.972	010.362	00.754
MLPBC		0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	086.272	08.383
DTBC	100	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	009.478	00.835
SVMBC		0.026	0.162	0.896	0.984	0.005	0.967	0.975	0.975	032.563	01.173
MLPBC		0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	215.473	18.362

(MN: Mean; STD: Standard Deviation; PS: Precision; RC: Recall; TrT: Training Time; TeT: Testing Time)

DOI: 10.3844/jcssp.2024.931.954

Table 14: Error metrics of boosting ensemble classifier over Kolkata dataset

Model	No. of estimators	MAE	RMSE	R^2	MN	STD	PS	RC	AUC	TrT	TeT
DTBoC	1	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.053	0.002
SVMBoC		0.437	0.634	-0.896	0.537	0.034	0.527	0.527	0.502	0.964	0.027
MLPBoC		0.063	0.267	0.723	0.953	0.040	0.952	0.936	0.932	1.543	0.052
DTBoC	5	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.056	0.003
SVMBoC		0.427	0.674	-0.896	0.537	0.031	0.527	0.527	0.500	2.856	0.153
MLPBoC		0.004	0.063	0.984	0.979	0.010	1.000	0.995	0.993	3.176	0.268
DTBoC	10	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.053	0.004
SVMBoC		0.417	0.634	-0.895	0.534	0.035	0.528	0.523	0.500	5.763	0.026
MLPBoC		0.047	0.256	0.805	0.975	0.031	0.986	0.956	0.950	2.567	0.186
DTBoC	15	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.052	0.005
SVMBoC		0.467	0.632	-0.892	0.532	0.035	0.527	0.523	0.500	4.782	0.016
MLPBoC		0.041	0.267	0.803	0.974	0.031	0.988	0.953	0.954	4.267	0.264
DTBoC	20	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.052	0.003
SVMBoC		0.428	0.623	-0.893	0.532	0.036	0.524	0.528	0.502	4.267	0.186
MLPBoC		0.026	0.176	0.899	0.965	0.061	0.984	0.975	0.974	5.273	0.276
DTBoC	50	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.063	0.004
SVMBoC		0.427	0.625	-0.895	0.534	0.036	0.527	0.527	0.50	3.726	0.115
MLPBoC		0.046	0.156	0.845	0.974	0.030	0.975	0.964	0.963	17.487	1.186
DTBoC	100	0.000	0.000	1.000	1.000	0.000	1.000	1.000	1.000	0.062	0.148
SVMBoC		0.476	0.627	-0.895	0.538	0.032	0.525	0.524	0.502	4.267	0.116
MLPBoC		0.063	0.236	0.745	0.976	0.035	0.952	0.936	0.937	21.483	0.500

(MN: Mean; STD: Standard Deviation; PS: Precision; RC: Recall; TrT: Training Time; TeT: Testing Time)

Analysis of Homogenous Ensemble Regressor by Bagging and Boosting

To test the efficiency of some ML models as ensemble classifiers and regressors, the ML models DT, SVM and MLP were used as homogenous ensemble regressors using combination techniques of bagging and boosting techniques.

The performance of DT, SVM and MLP ensemble regressors based on bagging and boosting over the Guwahati, Delhi and Kolkata datasets is shown in Table 15-20 using the error metrics.

It is evident that homogeneous MLP ensemble regressors consistently exhibit superior accuracy compared to DT and SVM ensemble regressors across all datasets. No significant difference is seen between the Multi-Layer Perceptron Bagging Regressor (MLPBR) and Multi-Layer Perceptron Boosting Regressor (MLPBoR).

Out of all the regressor models, SVM ensemble regressors performed worst. Still, if we compare Tables 9-14 it can be observed that the SVM Bagging Regressor (SVMBR)and SVM Boosting Regressor (SVMBoR) perform better than the SVM Bagging Classifier (SVMBC) and SVM Boosting Classifier (SVMBoC). From this, we can conclude that the SVM ensemble is more suitable for regression tasks as compared to classification task supporting literature (Choubin *et al*., 2019; Ren *et al*., 2016). Homogenous ensemble regressors take more training time and testing time as compared to homogenous ensemble classifiers which support literature (Adhikari and Agrawal, 2013; Bian and Wang, 2007). On average, out of all the models, the MLP ensemble regressor and classifier take the highest amount of testing and training time as compared to other models over all the datasets.

Analysis of Heterogenous Ensembled Classifier and Regressor by Stacking and Blending

In this section, we have discussed the empirical results of heterogeneous ensembled models developed using DT, SVM and MLP. Stacking and blending are used as the combination technique. Three stacked ensembled classifier models are created, namely, S-DSMC (base learner: DT, SVM and Meta-Learner: MLP), S-SMDC (base learner: SVM, MLP and meta-learner: DT) and S-DMSC (base learner: DT, MLP and meta-learner: SVM). Similarly, three blended ensembled classifier models are created, namely, B-DSMC (base learners: DT, SVM and meta-learner: MLP), B-SMDC (base learners: SVM, MLP are base-learners and DT meta-learner) and B-DMSC (base learners: DT, MLP and meta learner: SVM).

Figure 7 it is observed that all the stacking ensemble classifiers achieved an average accuracy of 98, 100% and 100%, respectively over the Guwahati, Delhi and Kolkata datasets. However, blending ensemble classifiers obtained an average accuracy of 93, 100 and 100%, respectively, for the Guwahati, Delhi and Kolkata datasets. We can conclude that the performance of stacking ensemble is better than bagging and blending ensemble classifiers over all the datasets which is supported by literature (Dou *et al*., 2020; Kumar *et al*., 2022)

Table 15: Error metrics of bagging ensemble regressor over the Guwahati dataset

Model	No. of estim ators	MAE	RMSE	R^2	ExVS	MdAE	RMSLE	TrT	TeT
DTBR	01	0.001	0.002	0.968	0.963	0.001	0.002	0.007	0.003
SVMBR		0.008	0.010	-0.261	0.000	0.009	0.008	0.006	0.002
MLPBR		0.009	0.011	0.713	-0.669	0.009	0.011	0.362	0.015
DTBR	05	0.002	0.002	0.988	0.983	0.001	0.001	0.020	0.004
SVMBR		0.009	0.011	-0.221	0.000	0.009	0.009	0.025	0.004
MLPBR		0.004	0.005	0.773	0.769	0.003	0.004	0.527	0.005
DTBR	10	0.002	0.002	0.988	0.983	0.001	0.001	0.047	0.006
SVMBR		0.009	0.010	-0.221	0.000	0.009	0.009	0.049	0.006
MLPBR		0.004	0.005	0.873	0.784	0.002	0.003	1.597	0.009
DTBR	15	0.001	0.001	0.988	0.983	0.983	0.001	0.047	0.004
SVMBR		0.008	0.010	-0.221	0.000	0.220	0.009	0.039	0.005
MLPBR		0.001	0.004	0.973	0.984	0.984	0.003	2.817	0.008
DTBR	20	0.001	0.001	0.988	0.973	0.001	0.001	0.316	0.076
SVMBR		0.008	0.010	-0.217	0.000	0.009	0.009	0.386	0.265
MLPBR		0.001	0.001	0.977	0.994	0.001	0.001	8.156	0.045
DTBR	50	0.001	0.001	0.978	0.973	0.001	0.001	0.181	0.026
SVMBR		0.008	0.010	-0.212	0.000	0.009	0.009	0.258	0.035
MLPBR		0.001	0.001	0.987	0.944	0.001	0.001	5.753	0.055
DTBR	100	0.001	0.001	0.978	0.973	0.001	0.001	0.267	0.029
SVMBR		0.008	0.010	-0.211	0.000	0.009	0.009	0.193	0.072
MLPBR		0.001	0.001	0.995	0.992	0.001	0.001	21.673	0.381

(ExVS: Explained Variance Score; MdAE: Median Absolute Error; TrT: Training Time; TeT: Testing Time)

Table 16: Error metrics of boosting ensemble regressor over the Guwahati dataset

Model	No. of estim ators	MAE	RMSE	R^2	ExVS	MdAE	RMSLE	TrT	TeT
DTBoR	1	0.002	0.003	0.867	0.882	0.002	0.003	0.004	0.002
SVMBoR		0.008	0.010	-0.215	0.000	0.009	0.009	0.005	0.002
MLPBoR		0.008	0.010	-0.341	-0.328	0.006	0.010	0.015	0.002
DTBoR	5	0.002	0.002	0.936	0.936	0.001	0.002	0.016	0.002
SVMBoR		0.008	0.010	-0.217	0.000	0.009	0.009	0.156	0.012
MLPBoR		0.004	0.007	0.406	0.408	0.003	0.006	1.945	0.003
DTBoR	10	0.002	0.001	0.936	0.984	0.001	0.001	0.075	0.022
SVMBoR		0.008	0.010	-0.217	0.000	0.009	0.009	0.178	0.016
MLPBoR		0.003	0.005	0.825	0.825	0.002	0.004	2.278	0.006
DTBoR	15	0.001	0.001	0.983	0.982	0.984	0.001	0.049	0.004
SVMBoR		0.008	0.010	-0.211	0.000	0.216	0.009	0.043	0.005
MLPBoR		0.002	0.003	0.863	0.873	0.863	0.003	2.296	0.050
DTBoR	20	0.001	0.001	0.981	0.982	0.001	0.001	0.104	0.009
SVMBoR		0.008	0.010	-0.214	0.000	0.009	0.009	0.095	0.015
MLPBoR		0.001	0.002	0.959	0.973	0.001	0.002	7.365	0.023
DTBoR	50	0.001	0.001	0.991	0.994	0.001	0.001	0.645	0.026
SVMBoR		0.008	0.010	-0.212	0.000	0.009	0.009	0.217	0.094
MLPBoR		0.001	0.001	0.984	0.987	0.001	0.001	15.547	0.042
DTBoR	100	0.001	0.001	0.993	0.996	0.000	0.001	0.385	0.024
SVMBoR		0.008	0.010	-0.216	0.000	0.009	0.009	0.246	0.035
MLPBoR		0.001	0.001	0.988	0.989	0.001	0.001	20.216	0.041

(ExVS: Explained Variance Score; MdAE: Median Absolute Error; TrT: Training Time; TeT: Testing Time)

Table 17: Error metrics of bagging ensemble regressor over the Delhi dataset

Model	No. of estimators	MAE	RMSE	R^2	ExVS	MdAE	RMSLE	TrT	TeT
DTBR	1	0.215	0.264	0.447	0.457	0.216	0.124	0.014	0.005
SVMBR		0.067	0.118	0.903	0.901	0.058	0.057	0.009	0.002
MLPBR		0.038	0.065	0.973	0.974	0.023	0.023	0.167	0.003
DTBR	5	0.069	0.091	0.937	0.937	0.057	0.041	0.015	0.002
SVMBR		0.057	0.084	0.946	0.948	0.042	0.039	0.014	0.003
MLPBR		0.016	0.028	0.991	0.995	0.008	0.014	0.678	0.003
DTBR	10	0.069	0.084	0.945	0.945	0.063	0.039	0.038	0.004
SVMBR		0.057	0.077	0.957	0.957	0.048	0.037	0.034	0.011
MLPBR		0.015	0.022	0.994	0.994	0.010	0.010	4.256	0.008
DTBR	15	0.075	0.089	0.938	0.935	0.073	0.046	0.062	0.006
SVMBR		0.057	0.075	0.951	0.952	0.041	0.032	0.044	0.016
MLPBR		0.015	0.022	0.997	0.991	0.017	0.010	5.352	0.023
DTBR	20	0.064	0.078	0.953	0.951	0.053	0.038	0.116	0.015
SVMBR		0.055	0.073	0.962	0.967	0.041	0.037	0.225	0.053
MLPBR		0.010	0.016	0.998	0.993	0.007	0.008	14.564	0.024
DTBR	50	0.061	0.075	0.951	0.953	0.057	0.036	0.327	0.042
SVMBR		0.055	0.073	0.958	0.952	0.043	0.036	0.558	0.162
MLPBR		0.010	0.014	0.998	0.998	0.006	0.007	35.238	0.053
DTBR	100	0.056	0.075	0.954	0.951	0.052	0.037	0.339	0.048
SVMBR		0.052	0.073	0.968	0.968	0.046	0.033	0.453	0.116
MLPBR		0.015	0.014	0.992	0.998	0.008	0.008	31.531	0.085

(ExVS: Explained Variance Score; MdAE: Median Absolute Error; TrT: Training Time; TeT: Testing Time)

Table 18: Error metrics of boosting ensemble regressor over the Delhi dataset

Model	No. of Estimators	MAE	RMSE	R^2	ExVS	MdAE	RMSLE	TrT	TeT
DTBoR	01	0.073	0.106	0.915	0.912	0.056	0.042	00.004	0.000
SVMBoR		0.056	0.084	0.947	0.948	0.042	0.045	00.005	0.001
MLPBoR		0.021	0.035	0.996	0.992	0.025	0.019	00.145	0.001
DTBoR	05	0.062	0.086	0.956	0.953	0.048	0.038	00.016	0.001
SVMBoR		0.047	0.064	0.972	0.978	0.042	0.036	00.025	0.004
MLPBoR		0.018	0.025	0.998	0.997	0.017	0.013	00.795	0.003
DTBoR	10	0.056	0.063	0.964	0.968	0.042	0.031	00.095	0.013
SVMBoR		0.042	0.052	0.978	0.974	0.045	0.026	00.225	0.031
MLPBoR		0.018	0.015	0.993	0.992	0.008	0.007	04.634	0.025
DTBoR	15	0.053	0.075	0.954	0.954	0.045	0.036	00.053	0.006
SVMBoR		0.048	0.059	0.976	0.978	0.041	0.022	00.084	0.026
MLPBoR		0.019	0.012	0.992	0.993	0.008	0.006	04.729	0.009
DTBoR	20	0.037	0.047	0.984	0.984	0.036	0.028	00.211	0.013
SVMBoR		0.042	0.052	0.979	0.976	0.048	0.024	00.175	0.031
MLPBoR		0.009	0.018	0.994	0.998	0.006	0.006	22.638	0.015
DTBoR	50	0.031	0.034	0.984	0.982	0.026	0.026	00.313	0.024
SVMBoR		0.044	0.051	0.979	0.977	0.048	0.028	00.785	0.341
MLPBoR		0.008	0.015	0.994	0.996	0.006	0.005	28.452	0.036
DTBoR	100	0.033	0.031	0.988	0.991	0.022	0.016	00.341	0.025
SVMBoR		0.042	0.054	0.972	0.972	0.047	0.028	00.153	0.014
MLPBoR		0.008	0.015	0.994	0.995	0.006	0.005	24.634	0.184

(ExVS: Explained Variance Score; MdAE: Median Absolute Error; TrT: Training Time; TeT: Testing Time)

Table 19: Error metrics of bagging ensemble regressor over Kolkata dataset

Models	No. of Estimators	MAE	RMSE	R^2	ExVS	MdAE	RMSLE	TrT	TeT
DTBR	01	0.034	0.041	0.906	0.904	0.028	0.024	0.004	0.002
SVMBR		0.031	0.048	0.896	0.898	0.025	0.021	0.004	0.002
MLPBR		0.018	0.015	0.992	0.993	0.015	0.006	0.137	0.003
DTBR	05	0.025	0.024	0.952	0.956	0.028	0.017	0.013	0.002
SVMBR		0.039	0.041	0.893	0.896	0.024	0.023	0.016	0.001
MLPBR		0.006	0.008	0.996	0.992	0.004	0.004	2.164	0.002
DTBR	10	0.022	0.023	0.954	0.954	0.023	0.013	0.025	0.007
SVMBR		0.036	0.048	0.887	0.893	0.028	0.028	0.028	0.009
MLPBR		0.006	0.008	0.992	0.992	0.005	0.004	3.572	0.025
DTBR	15	0.027	0.023	0.964	0.962	0.027	0.015	0.038	0.006
SVMBR		0.034	0.048	0.894	0.894	0.023	0.021	0.031	0.007
MLPBR		0.007	0.009	0.997	0.992	0.005	0.004	3.682	0.019
DTBR	20	0.017	0.026	0.978	0.973	0.027	0.012	0.088	0.009
SVMBR		0.034	0.047	0.892	0.898	0.023	0.026	0.072	0.024
MLPBR		0.007	0.009	0.992	0.995	0.006	0.004	8.238	0.026
DTBR	50	0.015	0.023	0.973	0.978	0.013	0.017	0.157	0.012
SVMBR		0.038	0.049	0.895	0.892	0.028	0.023	0.157	0.026
MLPBR		0.007	0.009	0.991	0.992	0.006	0.004	19.643	0.056
DTBR	100	0.016	0.025	0.974	0.973	0.014	0.013	0.275	0.052
SVMBR		0.032	0.048	0.896	0.895	0.022	0.028	0.168	0.036
MLPBR		0.007	0.009	0.993	0.991	0.006	0.004	24.642	0.071

(ExVS: Explained Variance Score; MdAE: Median Absolute Error; TrT: Training Time; TeT: Testing Time)

Table 20: Error metrics of boosting ensemble regressor over the Kolkata dataset

Model	No. of Estimators	MAE	RMSE	R^2	ExVS	MdAE	RMSLE	TrT	TeT
DTBoR	1	0.037	0.048	0.897	0.895	0.037	0.025	0.004	0.001
SVMBoR		0.031	0.042	0.893	0.898	0.021	0.023	0.004	0.001
MLPBoR		0.006	0.008	0.994	0.992	0.004	0.004	0.157	0.001
DTBoR	5	0.017	0.018	0.986	0.984	0.017	0.009	0.015	0.002
SVMBoR		0.031	0.045	0.892	0.898	0.021	0.027	0.013	0.002
MLPBoR		0.004	0.007	0.995	0.992	0.003	0.003	0.494	0.002
DTBoR	10	0.018	0.017	0.983	0.985	0.018	0.009	0.094	0.009
SVMBoR		0.035	0.041	0.898	0.892	0.022	0.022	0.147	0.022
MLPBoR		0.005	0.007	0.991	0.997	0.004	0.003	2.551	0.007
DTBoR	15	0.013	0.012	0.982	0.986	0.017	0.008	0.045	0.005
SVMBoR		0.039	0.047	0.894	0.892	0.021	0.027	0.043	0.005
MLPBoR		0.005	0.007	0.994	0.995	0.004	0.003	3.492	0.009
DTBoR	20	0.018	0.018	0.985	0.982	0.018	0.008	0.115	0.016
SVMBoR		0.032	0.043	0.892	0.894	0.022	0.022	0.148	0.013
MLPBoR		0.005	0.007	0.997	0.994	0.004	0.003	6.852	0.014
DTBoR	50	0.015	0.012	0.992	0.995	0.015	0.006	0.427	0.022
SVMBoR		0.039	0.043	0.894	0.892	0.025	0.025	0.297	0.068
MLPBoR		0.005	0.007	0.994	0.997	0.004	0.003	6.781	0.014
DTBoR	100	0.009	0.012	0.995	0.992	0.008	0.005	0.506	0.047
SVMBoR		0.032	0.049	0.892	0.894	0.025	0.029	1.015	0.173
MLPBoR		0.005	0.007	0.997	0.994	0.004	0.003	6.624	0.013

(ExVS: Explained Variance Score; MdAE: Median Absolute Error; TrT: Training Time; TeT: Testing Time)

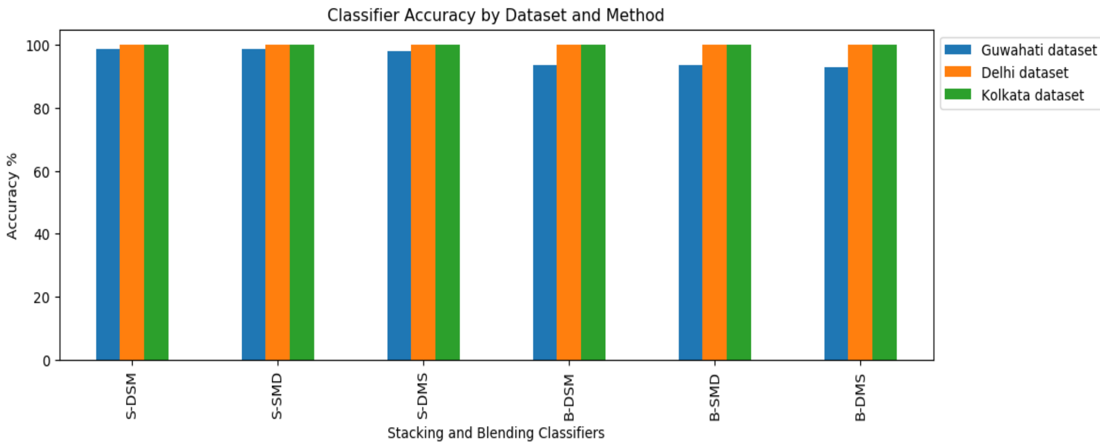

Fig. 7: Stacking and blending classifier accuracy

Table 21: Stacking and blending ensemble classifier error metrics over the Guwahati dataset

Models	MAE	RMSE	R^2	MN	STD	PS	RC	AUC	TrT	TeT
S-DSMC	0.068	0.257	0.725	0.895	0.027	0.935	0.936	0.935	3.635	0.146
S-SMDC	0.000	0.000	1.000	0.961	0.025	1.000	1.000	1.000	9.362	0.934
S-DMSC	0.000	0.000	1.000	0.967	0.016	1.000	1.000	1.000	7.957	0.193
B-DSMC	0.137	0.361	0.452	0.846	0.047	0.857	0.852	0.856	24.275	3.624
B-SMDC	0.073	0.276	0.697	0.941	0.025	1.000	0.936	0.939	22.583	0.836
B-DMSC	0.000	0.000	1.000	0.954	0.011	1.000	1.000	1.000	18.582	2.845

(MN: Mean; STD: Standard Deviation; PS: Precision; RC: Recall; TrT: Training Time; TeT: Testing Time)

Table 22: Stacking and blending ensemble classifier error metrics over the Delhi dataset

Models	MAE	RMSE	R^2	MN	STD	PS	RC	AUC	TrT	TeT
S-DSMC	0	0	1	1.000	0.000	1	1	1	1.873	0.026
S-SMDC	0	0	1	0.985	0.016	1	1	1	1.368	0.163
S-DMSC	0	0	1	0.997	0.009	1	1	1	1.974	0.075
B-DSMC	0	0	1	1.000	0.000	1	1	1	6.473	0.473
B-SMDC	0	0	1	0.984	0.016	1	1	1	8.382	0.482
B-DMSC	0	0	1	1.000	0.000	1	1	1	6.463	2.573

(MN: Mean; STD: Standard Deviation; PS: Precision; RC: Recall; TrT: Training Time; TeT: Testing Time)

Table 23: Stacking and blending ensemble classifier error metrics over the Kolkata dataset

Models	MAE	RMSE	R^2	MN	STD	PS	RC	AUC	TrT	TeT
S-DSMC	0	0	1	1	0	1	1	1	2.863	0.0372
S-SMDC	0	0	1	1	0	1	1	1	3.273	0.076
S-DMSC	0	0	1	1	0	1	1	1	2.452	0.047
B-DSMC	0	0	1	1	0	1	1	1	7.283	0.372
B-SMDC	0	0	1	1	0	1	1	1	8.137	0.462
B-DMSC	0	0	1	1	0	1	1	1	9.483	0.492

(MN: Mean; STD: Standard Deviation; PS: Precision; RC: Recall; TrT: Training Time; TeT: Testing Time)

Table 24: Stacking and blending ensemble regressors error metrics over the Guwahati dataset

Models	MAE	RMSE	R^2	ExVS	MdAE	RMSLE	TrT	TeT
S-DSMR	0.065	0.072	0.992	0.994	0.058	0.017	0.248	0.004
S-SMDR	0.152	0.284	0.945	0.947	0.175	0.046	7.296	0.001
S-DMSR	0.043	0.059	0.995	0.998	0.057	0.014	4.836	0.001
B-DSMR	0.582	0.572	0.582	0.914	0.413	0.158	0.386	0.472
B-SMDR	0.172	0.238	0.932	0.904	0.137	0.062	3.638	0.573
B-DMSR	0.053	0.067	0.994	0.996	0.041	0.015	2.836	0.479

(ExVS: Explained Variance Score; MdAE: Median Absolute Error; TrT: Training Time; TeT: Testing Time)

Table 25: Stacking and blending ensemble regressors error metrics over the Delhi dataset

Models	MAE	RMSE	R^2	ExVS	MdAE	RMSLE	TrT	TeT
S-DSMR	0.045	0.063	0.982	0.984	0.036	0.036	0.372	0.002
S-SMDR	0.094	0.261	0.915	0.915	0.067	0.062	3.826	0.001
S-DMSR	0.067	0.072	0.984	0.984	0.067	0.047	0.986	0.001
B-DSMR	0.195	0.381	0.672	0.936	0.372	0.184	0.482	0.372
B-SMDR	0.104	0.234	0.875	0.874	0.098	0.083	0.782	0.682
B-DMSR	0.049	0.064	0.980	0.982	0.046	0.037	2.583	0.371

(ExVS: Explained Variance Score; MdAE: Median Absolute Error; TrT: Training Time; TeT: Testing Time)

Table 26: Stacking and blending ensemble regressors error metrics over the Kolkata dataset

Models	MAE	RMSE	R^2	ExVS	MdAE	RMSLE	TrT	TeT
S-DSMR	0.026	0.026	0.983	0.985	0.008	0.008	0.264	0.001
S-SMDR	0.029	0.037	0.972	0.974	0.026	0.026	3.284	0.000
S-DMSR	0.047	0.055	0.892	0.893	0.037	0.036	3.184	0.003
B-DSMR	0.053	0.063	0.866	0.884	0.046	0.038	0.692	0.452
B-SMDR	0.068	0.084	0.748	0.747	0.067	0.046	2.539	0.492
B-DMSR	0.042	0.057	0.926	0.947	0.035	0.036	0.682	0.463

(ExVS: Explained Variance Score; MdAE: Median Absolute Error; TrT: Training Time; TeT: Testing Time)

The performance of SVM, DT and MLP ensemble regressors created on the basis of bagging and boosting is shown in Table 15-19 using the error metrics. Though we have used 1-100 base estimators in bagging and boosting, still stacking and blending with only two base learners and one meta-learner offered better accuracy. The overall training time and testing time for stacking and blending is less than bagging and boosting which supports the literature (Chaurasia and Pal, 2021; Wu *et al*., 2021). Stacking and blending error metrics over the Guwahati, Delhi and Kolkata dataset is represented in Tables 21- 23. From the tables it is observed that the overall testing and training time for blending is high as compared to stacking. This exposes that the accuracy of a heterogenous ensemble classifier is independent of the amount of time taken by a classifier to perform forecasting. In Tables 24- 26 we can observe that the S-DSNR model outperformed all other stacking and blending ensemble regressors. This reveals that the appropriate choice of base learners and meta-learners is very important in order to achieve better predictive models. We can sum up that stacking performed best among all other techniques used to ensemble classifiers and regressors. DT homogenous ensemble with bagging and boosting with (10-100) estimators offered good accuracy. Despite its higher accuracy, SVM and MLP ensemble models are more stable as compared to DT. SVM and MLP were less affected by the number of estimators.

In order to find the relations among the different models and to verify the results statistical significance test is performed so as to validate our data and results and then make some conclusions. We have calculated p-values using the F-test (one-way ANOVA) for each performance metric across the 24 models. By observing the values, we observed that SVM methods (SVMBC, SVMBoC, SVMBR, SVMBoR) generally have lower p-values compared to other methods, indicating they often perform differently from other methods. On the contrary, MLP methods (MLPBC, MLPBoC, MLPBR, MLPBoR) tend to have higher p-values suggesting they may not perform significantly differently from other methods in many cases. All the stacking models have generally higher p-values compared to blending models indicating less variability in performance.

Conclusion

In this study, we have tried to perform PM2.5 concentration prediction over three datasets using ensembled methods, namely, bagging and boosting for homogenous model construction and stacking and boosting for heterogeneous model construction. Till now an extensive comparative analysis of PM2.5 prediction using ensembled classifiers and regressors based on these techniques has not been thoroughly scrutinized in the literature. In this study, we have tried to address the following issues:

1. The combination technique (namely, bagging, boosting, blending, stacking) that is best suited for classification and regression tasks in PM2.5 concentration prediction
2. The appropriate number of base learners is required to build a homogenous ensemble classifier and regressor
3. Selection of base regressor and classifier in order to achieve maximum accuracy

In order to deal with the above issues, three popular ML models: SVM, MLP and DT are used. We have constructed twenty-four (24) different ensembled classifiers and regressors for PM2.5 concentration forecasting. The outputs found reveal the following:

1. Considering accuracy as the major factor, the stacking model for building a classifier and regressor outperforms other techniques like bagging, boosting and blending. The second best is performed by blending classifier and regressor followed by DT ensemble by bagging and boosting
2. Though stacking and blending offer better precision than DT bagging and boosting still they are computationally expensive because of the high time taken during training and testing. Because of this reason, if computation cost is a major factor, then DT ensemble with (10-100) estimators by boosting can be considered as a good option to perform prediction of PM2.5 concentration
3. The performance of the SVM ensemble by boosting and blending was stable but its performance was poor. The performance of SVM improved significantly when for base learners MLP and was used and for meta learners, SVM was used. Typical SVM makes an assumption that the parameters of the dataset provide an equivalent contribution to the target variable which is not true in the case of real-life problem situations

Through our study, we addressed key issues in PM2.5 concentration prediction, including identifying the most effective combination technique, determining the ideal quantity of base learners and selecting appropriate base classifiers and regressors for maximum accuracy. Leveraging popular machine learning techniques, namely SVM, MLP and DT, we built twenty-four different ensemble classifiers and regressors.

On the basis of these findings, we offer the following recommendations:

1. Practitioners seeking optimal accuracy should prioritize stacking for PM2.5 concentration prediction
2. For those constrained by computational resources, DT ensembles with boosting are a favorable choice
3. Consider leveraging a combination of diverse base learners to enhance the performance of SVM ensembles

By providing specific recommendations grounded in our empirical findings, our study aims to guide practitioners in selecting the most effective ensemble techniques for PM2.5 concentration prediction, thereby contributing to improved air quality monitoring and public health management.

Acknowledgment

We extend our gratitude to the almighty for guiding us through every experience and fostering patience, dedication and courage in our endeavors. We extend our sincere appreciation to the central pollution control Board, India, for generously supplying us with the essential dataset. Without their contribution, our research would not have been feasible. Finally, we express our heartfelt gratitude to our institution, "Girijananda Chowdhury institute of Management and Technology-Guwahati," affiliated with Assam science and Technology University, for their unwavering support and encouragement in pursuing research within our domain.

Funding Information

This research did not obtain financial support from either public or private sources.

Author’s Contributions

Shrabani Medhi: Participated in all the experiments, including data collected, coding and building all the models. Made significant contributions to the writing of the research paper.

Minakshi Gogoi: Provided essential guidance and oversight throughout the project. Gave the idea of the research, designed the workflow diagram and did analysis of data and results.

Ethics

This research does not involve human or animal subjects. So, no ethics approval is required for this research.

References

Adhikari, R., & Agrawal, R. K. (2013). A Homogeneous Ensemble of Artificial Neural Networks for Time Series Forecasting [dataset]. In *ArXiv*. https://doi.org/10.48550/arXiv.1302.6210

Agarwal, P., Bajpai, S., Pathak, A., & Angira, R. (2017). Stock Market Price Trend Forecasting Using Machine Learning. *International Journal for Research in Applied Science and Engineering Technology*, *5*(4), 1673-1676.

Amnuaylojaroen, T., & Parasin, N. (2023). Future health risk assessment of exposure to PM2. 5 in different age groups of children in Northern Thailand. *Toxics*, *11*(3), 291. https://doi.org/10.3390/toxics11030291

Bai, W., Zhou, Q., Li, T., & Li, H. (2019). Adaptive reinforcement learning neural network control for uncertain nonlinear system with input saturation. *IEEE Transactions on Cybernetics*, *50*(8), 3433–3443. https://doi.org/10.1109/TCYB.2019.2921057

Banerjee, S., Banerjee, A., Palit, D., & Roy, P. (2019). Assessment of vegetation under air pollution stress in urban industrial area for greenbelt development. *International Journal of Environmental Science and Technology*, *16*(10), 5857-5870. https://doi.org/10.1007/s13762-018-1963-9

Barman, N., & Gokhale, S. (2019). Urban black carbon-source apportionment, emissions and long-range transport over the Brahmaputra River Valley. *Science of the Total Environment*, *693*, 133577. https://doi.org/10.1016/j.scitotenv.2019.07.383

Beig, G., Ghude, S. D., & Deshpande, A. (2010). Scientific evaluation of air quality standards and defining air quality index for India (pp. 1-27). Indian Institute of Tropical Meteorology. ISSN: 0252-1075

Betancourt, C., Stomberg, T. T., Edrich, A.-K., Patnala, A., Schultz, M. G., Roscher, R., Kowalski, J., & Stadtler, S. (2022). Global, high-resolution mapping of tropospheric ozone–explainable machine learning and impact of uncertainties. *Geoscientific Model Development*, *15*(11), 4331–4354. https://doi.org/10.5194/gmd-15-4331-2022

Bian, S., & Wang, W. (2007). On Diversity and Accuracy of Homogeneous and Heterogeneous Ensembles. *International Journal of Hybrid Intelligent Systems*, *4*(2), 103-128. https://doi.org/10.3233/his-2007-4204

Booth, A., Gerding, E., & McGroarty, F. (2014). Automated Trading with Performance Weighted Random Forests and Seasonality. *Expert Systems with Applications*, *41*(8), 3651-3661. https://doi.org/10.1016/j.eswa.2013.12.009

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*, 123–140. https://doi.org/10.1007/BF00058655

Cabaneros, S. M., Calautit, J. K., & Hughes, B. R. (2019). A review of artificial neural network models for ambient air pollution prediction. *Environmental Modelling & Software*, *119*, 285–304. https://doi.org/10.1016/j.envsoft.2019.06.014

Chaurasia, V., & Pal, S. (2021). Stacking-Based Ensemble Framework and Feature Selection Technique for the Detection of Breast Cancer. *SN Computer Science*, *2*(2), 67. https://doi.org/10.1007/s42979-021-00465-3

Choubin, B., Moradi, E., Golshan, M., Adamowski, J., Sajedi-Hosseini, F., & Mosavi, A. (2019). An Ensemble Prediction of Flood Susceptibility Using Multivariate Discriminant Analysis, Classification and Regression Trees and Support Vector Machines. *Science of the Total Environment*, *651*, 2087-2096. https://doi.org/10.1016/j.scitotenv.2018.10.064

Danesh Yazdi, M., Kuang, Z., Dimakopoulou, K., Barratt, B., Suel, E., Amini, H., Lyapustin, A., Katsouyanni, K., & Schwartz, J. (2020). Predicting fine particulate matter (PM2. 5) in the greater london area: An ensemble approach using machine learning methods. *Remote Sensing*, *12*(6), 914. https://doi.org/10.3390/rs12060914

De Mello Assis, J., Pereira, A. C. M., & Rodrigo Couto, E. S. (2018). Designing Financial Strategies Based on Artificial Neural Networks Ensembles for Stock Markets. *2018 International Joint Conference on Neural Networks (IJCNN)*, 1-8. https://doi.org/10.1109/ijcnn.2018.8489688

Devi, S., Esmaeilbeiki, F., Karimi, S. M., & Sihag, P. (2022). Prediction of Sulphur Dioxide (So2) in Air by Using Bagging, Ann and M5p: A Case Study, Gaya and Tirupati, India. *Arabian Journal of Geosciences*, *15*(7), 613. https://doi.org/10.1007/s12517-022-09725-9

Dou, J., Yunus, A. P., Bui, D. T., Merghadi, A., Sahana, M., Zhu, Z., Chen, C.-W., Han, Z., & Pham, B. T. (2020). Improved landslide assessment using support vector machine with bagging, boosting and stacking ensemble machine learning framework in a mountainous watershed, Japan. *Landslides*, *17*, 641–658. https://doi.org/10.1007/s10346-019-01286-5

Evans, J., van Donkelaar, A., Martin, R. V., Burnett, R., Rainham, D. G., Birkett, N. J., & Krewski, D. (2013). Estimates of global mortality attributable to particulate air pollution using satellite imagery. *Environmental research*, *120*, 33-42. https://doi.org/10.1016/j.envres.2012.08.005

Feng, L., Li, Y., Wang, Y., & Du, Q. (2020). Estimating Hourly and Continuous Ground-Level Pm2.5 Concentrations Using an Ensemble Learning Algorithm: The St-Stacking Model. *Atmospheric Environment*, *223*, 117242. https://doi.org/10.1016/j.atmosenv.2019.117242

Feng, S., Gao, D., Liao, F., Zhou, F., & Wang, X. (2016). The Health Effects of Ambient PM2.5 and Potential Mechanisms. *Ecotoxicology and Environmental Safety*, *128*, 67-74. https://doi.org/10.1016/j.ecoenv.2016.01.030

Ganesh, S. S., Arulmozhivarman, P., & Tatavarti, V. R. (2018). Prediction of PM 2.5 using an ensemble of artificial neural networks and regression models. *Journal of Ambient Intelligence and Humanized Computing*, 1–11. https://doi.org/10.1007/s12652-018-0801-8

Gao, D., Zhu, Y., Zhang, J., He, Y., Yan, K., & Yan, B. (2021). A novel MP-LSTM method for ship trajectory prediction based on AIS data. *Ocean Engineering*, *228*, 108956. https://doi.org/10.1016/j.oceaneng.2021.108956

Gonzalez, R. T., Padilha, C. A., & Barone, D. A. C. (2015). Ensemble System Based on Genetic Algorithm for Stock Market Forecasting. *IEEE Xplore*, 3102-3108. https://doi.org/10.1109/cec.2015.7257276

Guzman, E., El-Haliby, M., & Bruegge, B. (2015). Ensemble Methods for App Review Classification: An Approach for Software Evolution (N). *IEEE Xplore*, 771-776. https://doi.org/10.1109/ase.2015.88

Jacobsen, B., Jiang, F., & Zhang, H. (2020). Equity premium prediction with bagged machine learning. *AsianFA 2019*, *AMES 2019*, *FMND 2019*, *3310289*, 56. https://doi.org/10.2139/ssrn.3310289

Janssen, N. A. H., Fischer, P., Marra, M., Ameling, C., & Cassee, F. R. (2013). Short-term effects of PM2. 5, PM10 and PM2. 5-10 on daily mortality in the Netherlands. *Science of the Total Environment*, *463*, 20-26. https://doi.org/10.1016/j.scitotenv.2013.05.062

Jarah, N. B., AlAsadi, A. H., & Hashim, K. M. (2023). Processing and analyzing data to predict earthquakes in Iraq. *J. Basrah Res. (Sci.)*, *49*(2), 11-123. https://doi.org/10.56714/bjrs.49.2.10

Jebamalar, J. A., & Kamalakannan, T. (2021). Enhanced stacking ensemble model in predictive analytics of environmental sensor data. *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, 482–486. https://ieeexplore.ieee.org/abstract/document/9395755/

Khairalla, M. A., Ning, X., AL-Jallad, N. T., & El-Faroug, M. O. (2018). Short-Term Forecasting for Energy Consumption through Stacking Heterogeneous Ensemble Learning Model. *Energies*, *11*(6), 1605. https://doi.org/10.3390/en11061605

Kim, K.-H., Kabir, E., & Kabir, S. (2015). A review on the human health impact of airborne particulate matter. *Environment International*, *74*, 136–143. https://doi.org/10.1016/j.envint.2014.10.005

Kloog, I., Ridgway, B., Koutrakis, P., Coull, B. A., & Schwartz, J. D. (2013). Long-and short-term exposure to PM2. 5 and mortality: Using novel exposure models. *Epidemiology*, *24*(4), 555–561. https://doi.org/10.1097/EDE.0b013e318294beaa

Kumar, M., Bajaj, K., Sharma, B., & Narang, S. (2022). A Comparative Performance Assessment of Optimized Multilevel Ensemble Learning Model with Existing Classifier Models. *Big Data*, *10*(5), 371–387. https://doi.org/10.1089/big.2021.0257

Laden, F., Schwartz, J., Speizer, F. E., & Dockery, D. W. (2006). Reduction in fine particulate air pollution and mortality: extended follow-up of the Harvard Six Cities study. *American journal of respiratory and critical care medicine*, *173*(6), 667-672. https://doi.org/10.1164/rccm.200503-443OC

Leong, L.-Y., Hew, T.-S., Ooi, K.-B., & Wei, J. (2020). Predicting mobile wallet resistance: A two-staged structural equation modeling-artificial neural network approach. *International Journal of Information Management*, *51*, 102047. https://doi.org/10.1016/j.ijinfomgt.2019.102047

Liu, X., Tan, W., & Tang, S. (2019). A Bagging-GBDT Ensemble Learning Model for City Air Pollutant Concentration Prediction. *IOP Conference Series: Earth and Environmental Science*, *237*(2), 022027. https://doi.org/10.1088/1755-1315/237/2/022027

Mabu, S., Obayashi, M., & Kuremoto, T. (2015). Ensemble learning of rule-based evolutionary algorithm using multi-layer perceptron for supporting decisions in stock trading problems. *Applied Soft Computing*, *36*, 357-367. https://doi.org/10.1016/j.asoc.2015.07.020

Macchiarulo, A. (2018). Predicting and Beating the Stock Market with Machine Learning and Technical Analysis. *Journal of Internet Banking and Commerce*, *23*(1), 1-22.

Maleki, H., Sorooshian, A., Goudarzi, G., Baboli, Z., Tahmasebi Birgani, Y., & Rahmati, M. (2019). Air pollution prediction by using an artificial neural network model. *Clean Technologies and Environmental Policy*, *21*, 1341–1352. https://doi.org/10.1007/s10098-019-01709-w

Martins, N. R., & Da Graca, G. C. (2018). Impact of PM2. 5 in indoor urban environments: A review. *Sustainable Cities and Society*, *42*, 259-275. https://doi.org/10.1016/j.scs.2018.07.011

Mayr, A., Binder, H., Gefeller, O., & Schmid, M. (2014). The Evolution of Boosting Algorithms. *Methods of Information in Medicine*, *53*(6), 419-427. https://doi.org/10.3414/ME13-01-0122

Medhi, S., & Gogoi, M. (2021). Visualization and Analysis of COVID-19 Impact on PM2. 5 Concentration in Guwahati city. *2021 International Conference on Computational Performance Evaluation (ComPE)*, 012–016. https://doi.org/10.1109/ComPE53109.2021.9752244

Medhi, S., Kashyap, P., Das, A., & Sarma, J. (2023). PM2. 5 Prediction using Heterogeneous Ensemble Learning. *Journal of Artificial Intelligence and Capsule Networks*, *5*(4), 481-498.

Mehta, S., Rana, P., Singh, S., Sharma, A., & Agarwal, P. (2019). Ensemble Learning Approach for Enhanced Stock Prediction. *IEEE Xplore*, 1-5. https://doi.org/10.1109/ic3.2019.8844891

Miller, P. E., Pawar, S., Vaccaro, B., McCullough, M., Rao, P., Ghosh, R., ... & Ahmad, T. (2019). Predictive abilities of machine learning techniques may be limited by dataset characteristics: insights from the UNOS database. *Journal of cardiac failure*, *25*(6), 479-483. https://doi.org/10.1016/j.cardfail.2019.01.018

Mitchell, H. B. (2012). Bayesian Inference. In Data Fusion: Concepts and Ideas (1st Ed., pp. 171–192). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-27222-6_9

Morales, A. J., Losada, J. C., & Benito, R. M. (2012). Users Structure and Behavior on an Online Social Network During a Political Protest. *Physica A: Statistical Mechanics and Its Applications*, *391*(21), 5244-5253. https://doi.org/10.1016/j.physa.2012.05.015

Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2020). A Systematic Review of Fundamental and Technical Analysis of Stock Market Predictions. *Artificial Intelligence Review*, *53*(4), 3007-3057. https://doi.org/10.1007/s10462-019-09754-z

Obodoeze, F. C., Nwabueze, C. A., & Akaneme, S. A. (2021). Comparative Evaluation of Machine Learning Regression Algorithms for PM2.5 Monitoring. *American Journal of Engineering Research*, *10*(12), 19–33.

Oliveira, M., Slezakova, K., Delerue-Matos, C., Pereira, M. C., & Morais, S. (2016). Assessment of air quality in preschool environments (3–5 years old children) with emphasis on elemental composition of PM10 and PM2. 5. *Environmental Pollution*, *214*, 430-439. https://doi.org/10.1016/j.envpol.2016.04.046

Kaewkiriya, T. & Wisaeng, K. (2023). Development of Customer Predictive Model for Investment Using Ensemble Learning Technique. *Journal of Computer Science*, *19*(6), 775-785. https://doi.org/10.3844/jcssp.2023.775.785

Kioumourtzoglou, M.-A., Schwartz, J. D., Weisskopf, M. G., Melly, S. J., Wang, Y., Dominici, F., & Zanobetti, A. (2016). Long-term PM2. 5 exposure and neurological hospital admissions in the northeastern United States. *Environmental Health Perspectives*, *124*(1), 23–29. https://doi.org/10.1289/ehp.1408973

Patro, S. G. O. P. A. L., & Sahu, K. K. (2015). Normalization: A preprocessing stage. ArXiv preprint. https://doi.org/10.48550/arXiv.1503.06462

Pasupulety, U., Abdullah Anees, A., Anmol, S., & Mohan, B. R. (2019). Predicting Stock Prices using Ensemble Learning and Sentiment Analysis. *IEEE Xplore*, 215-222. https://doi.org/10.1109/aike.2019.00045

Pope III, C. A., & Dockery, D. W. (2006). Health effects of fine particulate air pollution: lines that connect. *Journal of the Air & Waste Management Association*, *56*(6), 709-742. https://doi.org/10.1080/10473289.2006.10464485

Pope III, C. A., Turner, M. C., Burnett, R. T., Jerrett, M., Gapstur, S. M., Diver, W. R., ... & Brook, R. D. (2015). Relationships between fine particulate air pollution, cardiometabolic disorders and cardiovascular mortality. *Circulation Research*, *116*(1), 108-115. https://doi.org/10.1161/CIRCRESAHA.116.305060

Ren, Y., Zhang, L., & Suganthan, P. N. (2016). Ensemble Classification and Regression-Recent Developments, Applications and Future Directions [Review Article]. *IEEE Computational Intelligence Magazine*, *11*(1), 41-53. https://doi.org/10.1109/mci.2015.2471235.

Ribeiro, M. H. D. M., & Dos Santos Coelho, L. (2020). Ensemble Approach Based on Bagging, Boosting and Stacking for Short-Term Prediction in Agribusiness Time Series. *Applied Soft Computing*, *86*, 105837. https://doi.org/10.1016/j.asoc.2019.105837

Rojas-Rueda, D., Alsufyani, W., Herbst, C., AlBalawi, S., Alsukait, R., & Alomran, M. (2021). Ambient particulate matter burden of disease in the Kingdom of Saudi Arabia. *Environmental Research*, *197*, 111036. https://doi.org/10.1016/j.envres.2021.111036

Srisuradetchai, P., & Panichkitkosolkul, W. (2022). Using Ensemble Machine Learning Methods to Forecast Particulate Matter (PM2.5) in Bangkok, Thailand. In *Springer Link* (Vol. *13651*, pp. 204-215). Springer International Publishing. https://doi.org/10.1007/978-3-031-20992-5_18

Sun, W., & Li, Z. (2020). Hourly PM2. 5 concentration forecasting based on mode decomposition-recombination technique and ensemble learning approach in severe haze episodes of China. *Journal of Cleaner Production*, *263*, 121442. https://doi.org/10.1007/978-3-031-20992-5_18

Triana, D., & Osowski, S. (2020). Bagging and Boosting Techniques in Prediction of Particulate Matters. *Bulletin of the Polish Academy of Sciences Technical Sciences*, *68*(5), 1207-1215. https://doi.org/10.24425/bpasts.2020.134659

Tsai, C.-F., Hsu, Y.-F., & Yen, D. C. (2014). A Comparative Study of Classifier Ensembles for Bankruptcy Prediction. *Applied Soft Computing*, *24*, 977-984. https://doi.org/10.1016/j.asoc.2014.08.047

Usmani, M., Ebrahim, M., Adil, S. H., & Raza, K. (2018). Predicting market performance with hybrid model. *2018 3rd International Conference on Emerging Trends in Engineering, Sciences and Technology (ICEEST)*, 1–4. https://doi.org/10.1109/ICEEST.2018.8643327

Wang, Q., Xu, W., Huang, X., & Yang, K. (2019). Enhancing Intraday Stock Price Manipulation Detection by Leveraging Recurrent Neural Networks with Ensemble Learning. *Neurocomputing*, *347*, 46-58. https://doi.org/10.1016/j.neucom.2019.03.006

Weng, B. (2017). *Application of machine learning techniques for stock market prediction* (Doctoral dissertation, Auburn University).

Wilson, W. E., & Suh, H. H. (1997). Fine Particles and Coarse Particles: Concentration Relationships Relevant to Epidemiologic Studies. *Journal of the Air and Waste Management Association*, *47*(12), 1238-1249. https://doi.org/10.1080/10473289.1997.10464074

Wu, T., Zhang, W., Jiao, X., Guo, W., & Alhaj Hamoud, Y. (2021). Evaluation of Stacking and Blending Ensemble Learning Methods for Estimating Daily Reference Evapotranspiration. *Computers and Electronics in Agriculture*, *184*, 106039. https://doi.org/10.1016/j.compag.2021.106039

Xu, L., Chen, N., Zhang, X., Chen, Z., Hu, C., & Wang, C. (2019). Improving the North American Multi-Model Ensemble (NMME) precipitation forecasts at local areas using wavelet and machine learning. *Climate Dynamics*, *53*, 601-615. https://doi.org/10.1007/s00382-018-04605-z

Yang, Z., Wang, Y., Xu, X.-H., Yang, J., & Ou, C.-Q. (2022). Quantifying and characterizing the impacts of PM2. 5 and humidity on atmospheric visibility in 182 Chinese cities: A nationwide time-series study. *Journal of Cleaner Production*, *368*, 133182. https://doi.org/10.1016/j.jclepro.2022.133182

Yazdani, A. (2021). Supplement of Characterization of primary and aged wood burning and coal combustion organic aerosols in an environmental chamber and its implications for atmospheric aerosols. *Supplement of Atmos. Chem. Phys*, *21*, 10273–10293. https://doi.org/10.5194/acp-21-10273-2021-supplement

Zamani Joharestani, M., Cao, C., Ni, X., Bashir, B., & Talebiesfandarani, S. (2019). PM2. 5 prediction based on random forest, XGBoost and deep learning using multisource remote sensing data. *Atmosphere*, *10*(7), 373. https://doi.org/10.3390/atmos10070373

Zhang, Z., Shao, C., Guan, Y., & Xue, C. (2019). Socioeconomic factors and regional differences of PM2. 5 health risks in China. *Journal of Environmental Management*, *251*, 109564. https://doi.org/10.1016/j.jenvman.2019.109564

Zheng, F., & Zhong, S. (2011). Time series forecasting using an ensemble model incorporating ARIMA and ANN based on combined objectives. *2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)*, 2671–2674. https://doi.org/10.1109/AIMSEC.2011.6011011

Zhu, Y., Zhou, L., Xie, C., Wang, G. J., & Nguyen, T. V. (2019). Forecasting SMEs' Credit Risk in Supply Chain Finance with an Enhanced Hybrid Ensemble Machine Learning Approach. *International Journal of Production Economics*, *211*, 22-33. https://doi.org/10.1016/j.ijpe.2019.01.032