Aerosol Science and Engineering
https://doi.org/10.1007/s41810-025-00292-6

ORIGINAL PAPER

Optimizing Air Pollution Prediction With Random Forest Algorithm

Sukhendra Singh[1] · **Manoj Kumar**[1] · **Birendra Kumar Verma**[1] · **Sushil Kumar**[2]

Received: 22 July 2024 / Revised: 14 January 2025 / Accepted: 27 January 2025

Abstract
Air pollution is a significant environmental concern, especially in urban areas with dense populations. Long-term exposure to air pollution can cause serious health problems, such as respiratory diseases, cardiac problems, and even premature mortality. Reducing air pollution is essential for protecting human health and the environment. This paper aims to predict air quality in National Capital Territory (NCT) of Delhi, India using machine learning algorithms, specifically focusing on the Random Forest model. Air pollution is a significant environmental concern, particularly in densely populated urban areas like NCT Delhi India. It involves the contamination of the atmosphere with hazardous substances such as Particulate Matter ($PM_{2.5}$, PM_{10}), Nitrogen Dioxide (NO_2), and Sulfur Dioxide (SO_2). In our research, we proposed a machine-learning framework using random forest algorithms to predict air quality. We used hourly measurements of key pollutants from January 2017 to December 2018 in the NCT Delhi India region, with predictions evaluated through k-fold cross-validation and grid search techniques. Our findings revealed that the Random Forest regression model outperforms simpler models, achieving an R^2 value of 98.89, highlighting its efficacy in capturing complex relationships among air quality parameters. The random forest model can be utilized by policymakers to make informed decisions regarding the management and regulation of air quality.

Keywords Air pollution · Cross-validation · Decision tree · Grid search · $PM_{2.5}$ · PM_{10} · Random forest · Regression · SO_2 · Support vector regression

1 Introduction

Air quality has long been a significant problem in every part of the globe. The impact of air quality on human health has become a focus of study in recent years. Dust, pollen, soot, smoke, and liquid droplets can pollute the air and various other particles. Particulate Matter (PM) is a complex mixture of microscopic particles and liquid droplets that enter the air, contribute to air pollution, and degrade air quality. $PM_{2.5}$, or particulate Matter with a diameter of 2.5 micrometers or smaller, provides the highest health risk and is frequently employed as a measure in regulatory air quality requirements. $PM_{2.5}$ is deeply absorbed into the circulation upon inhalation and has been related to stroke, heart disease, lung disease, and cancer. From a medical standpoint, particulate particles in the atmosphere can enter the respiratory system. Once breathed, these particles can have detrimental effects on the heart and lungs. Long-term exposure to high amounts of atmospheric PM can diminish lung function and hasten mortality.

According to the WHO (Khomenko et al. 2020), 99% of the world's population breathes polluted air, and 7 million premature deaths happen annually due to air pollution. Experts warn that governments must take immediate steps to increase air quality legislation, particularly the capacity to measure $PM_{2.5}$ and other pollutants, to manage the current air pollution issue. The Air Quality Index (AQI) is used to quantify air quality. AQI (Borbet et al. 2018) will demonstrate the changes in atmospheric air pollution. Air quality is crucial for maintaining both human health and the environment. Oxygen and Nitrogen, both of which are essential for life on Earth, constitute most of our atmosphere.

✉ Sushil Kumar
drsushil.cs@gmail.com

[1] Department of Information Technology, JSS Academy of Technical Education Noida, Noida, Uttar Pradesh 201301, India

[2] Data Science & Deep Learning (DSDL) Lab, Department of Computer Science and Engineering, KIET Group of Institutions, Delhi-NCR, Ghaziabad, Uttar Pradesh 201206, India

Published online: 03 March 2025

AQI monitors eight major air pollutants in the atmosphere, which are $PM_{2.5}$, PM_{10}, NO_2, SO_2, CO, O_3, NH_3, and Pb. To compute the AQI (Chakrabarti et al. 2015), a minimum of three pollutants must be measured, one of which must be either PM_{10} or $PM_{2.5}$. The AQI ranges from 0 to 500, with varying values for each pollutant and corresponding health impacts. The sensors on air quality monitors (Jo et al. 2020) are intended to identify particular contaminants. Others rely on satellite imagery to detect energy reflected or radiated by the Earth, while some employ lasers to scan particulate matter density in a cubic meter of air.

Numerous Indian cities have installed air quality monitoring stations for real-time monitoring of the levels of air pollutants, such as $PM_{2:5}$, PM_{10}, etc. Table 1 defines the AQI range with respect to the air category. In addition, real-time monitoring cannot entirely tackle the problem of air pollution. It is also vital to estimate future air quality to improve air pollution. Therefore, creating a scientific and precise prediction model is crucial to forecast future air quality. Based on the forecast findings, the relevant departments can take the necessary precautions in advance to mitigate the harm caused by air pollution.

1.1 Motivation

Individual, community, national, and global air pollution forecasting is a worthy investment. Accurate forecasts help individuals prepare ahead, reducing health risks and expenditures. If individuals are aware of fluctuations in air quality, the impact of pollutants on health, and concentrations likely to have detrimental consequences, they may take steps to reduce pollution. People wanting air quality information are more likely to affect their behavior and public policy. Such understanding can generate a healthier environment and population. Early forecasting also helps governments decrease local pollution.

In this experimentation, we are proposing computational intelligence-based machine learning prediction models, which were trained on NCT Delhi India AQI data, and their performance was compared and analyzed. The following are our main contributions:

1. Experimental results show that the random forest regression model outperforms other models, with an R^2 value of 98.89. This indicates that the random forest model can capture the complex relationships between air quality parameters and provide accurate predictions. The feature importance analysis revealed that $PM_{2.5}$ and NO_2 were the most important variables in predicting air quality in NCT Delhi India.
2. This study evaluates several machine learning algorithms for air pollution modeling and prediction and provides a comparative analysis of their performance. It shows that random forest regression is a particularly effective algorithm for air pollution modeling in NCT Delhi India, and provides insights into the most important variables for predicting air quality.
3. Experiment uses rigorous cross-validation techniques to ensure the accuracy and generalizability of the models and uses grid search to tune the hyperparameters of the algorithms. This leads to reducing the biases of the proposed model.
4. Experiment provides new insights into the patterns and trends of air pollution in NCT Delhi India and identifies the most significant contributors to poor air quality. This can inform targeted interventions to improve air quality in the city and other urban areas.

Table 1 Air quality category

AQI range	Air category
0–50	Good
51–100	Satisfactory
101–200	Moderate
201–300	Poor
301–400	Very Poor
401–500	Severe

The remaining paper is structured as follows: Sect. 2 frames the background details to build the foundation required for proposing the model. It also presents the recent findings of related research studies. Section 3 describes the methodology used, dataset characteristics, and performance evaluation parameters. Section 4 describes the experimentation and discussion of results. Section 5 presents the GridSearchCV outcomes and comparison of various regressions. Finaly, Sect. 6 concludes the proposed work.

2 Related Work

In this section, we present the key idea to build the foundation for the proposed work. Air quality modeling (Georgiou et al. 2020) estimates ambient air concentrations near our destination or other areas of interest. Air quality modeling is a mathematical simulation of how air pollution responds and disperses. Transportation involves a metamorphosis since movement changes appearance. Basic dispersion models assume gases and pollutants are inert or nonreactive. The shape will only convey whatever is released. We make this simplistic assumption often to simplify. Air quality modeling (Ye et al. 2021) can quantify the relationship between pollution sources and their consequences on receptors, the environment, and human health. A plume emerges and disperses from the stack. It dilutes x, y, and z-direction

contaminants. Emission data shows us how much pollution a source produces and what its characteristics are.

Mobile or fixed sources, at ground level or a set height, followed by the receptor's location. We are interested in how much diffusion from an industrial district to a hamlet influences air quality. Consider a soon-to-be-built industrial area. We want to know if this industrial region's emissions will impair the quality of life in a city, town, or hamlet. The modeling is vital if we want to identify an industrial region and evaluate if it would damage downstream receptors, such as this town, or environmentally sensitive locations, like a river or forest, whose air quality we do not want to affect. When emissions exceed permitted limits, modeling helps determine what policies and measures may be taken to reduce or address ecological damage. Air quality modeling (Sahoo et al. 2021) requires information regarding emission sources, including how much is emitted and what pollutants are present, as well as meteorological parameters like temperature, wind speed, etc. Receptors are connected as if in a sensitive eco-zone or near a school, children, or hospital. Topography (Deak et al. 2020) also affects the dispersion phenomena, whether it's level, undulating, or rocky. Also, it should include specifics. All are dispersion model inputs. The receptor's air pollution concentration is given.

Air quality modeling (Yadav et al. 2019; Yadav 2016) lets us predict the atmospheric concentrations of a pollutant or many pollutants at a place. Mathematical models of how air contaminants disperse in the atmosphere, including physical/chemical processes, are used. Comparing concentrations to norms and guidelines helps evaluate the effectiveness and health risks. Or to establish if a new industrial development may harm a downwind neighborhood. These air quality models can be used to investigate. The Air Quality Index is a relationship between the weighted values of individual air pollution concentrations, and different air pollutants like SO_2, PM_{10}, and $PM_{2.5}$, merging them into a single number. So, for the calculation of AQI, first, we monitor various contaminants and their concentrations in ambient air and calculate their weighted average or weighted value depending upon their health impacts, then combine them. QI is just an indication of ambient air quality ranging between 0 and 500, according to Central Pollution Control Board (CPCB) (CPCB 2019). AQI is generally used to communicate the stringency of air pollution to people at any location, any place, and its potency of impact. The lower the AQI value cleaner the air; the higher the AQI, the higher the risk.

Computers are these days equipped to perform many complex tasks using various intelligent techniques. Computational Intelligence (CI) is the name given to superset of all such intelligence methodologies. The domain of CI (Pedrycz et al. 2016) has evolved in techniques like fuzzy logic, neural networks (Kollmannsberger et al. 2021), and genetic algorithm (Katoch et al. 2021; Gupta et al. 2019). Neural networks are one of the essential machine learning models inspired by computation inside the human brain. Feed-forward, feed-backward, and recurrent are the main types of neural networks. Machine learning is a branch of artificial intelligence. It was defined formally by machine learning pioneer Tom Mitchell as "Machine learning is the study of computer algorithms that allow computer programs to improve through experience automatically". Machine learning techniques are used in classification, clustering, prediction, object detection, summarization, etc. Just like a human who learns from past experiences and performs and changes his actions based on past experiences, a software system can also learn by improving its performance upon its completed tasks based on past data. This learning by design can be supervised or unsupervised. Supervised learning (Jiang et al. 2020; Maurya et al. 2024) is employed in the environment where the desired target variable is given, and the system performs by minimizing errors. Classification is a supervised learning task in which the system categorizes or predicts an instance of data to predefined labels or discrete categories. Regression is also another supervised learning task in which a system indicates a constant value on an example of the data. Classifying the air quality on detailed data into predefined categories, like sound, moderate, poor, etc., is a classification task. It is a regression task when a system predicts a substantial continuous AQI value on a day. The current experiment also deals with regression tasks. Backpropagation algorithm (Andrew 2001) is a classical supervised learning algorithm that tells how a system has to change its action based on weights in a given situation. Deep learning (Ibrahim et al. 2021; Singh 2022a; Kumar 2023; Singh 2022b; Singh et al. 2024), which is a trending machine learning. sub-discipline emerged from backpropagation. Recently, many researchers applied LSTM to model air quality data.

The literature review provides a comprehensive overview of several advanced machine learning and deep learning approaches (Singh et al. 2021; Bagwari et al. 2023; Kumar 2023; Kumar 2017; Kumar 2018) for predicting air quality and water quality. Each row represents a distinct study, summarizing the approach used, the characteristics of the dataset employed, and the significant findings of the research. The study (Li et al. 2019) utilizes a greedy-layer-by-layer trained stacked autoencoder (SAE) model to extract intrinsic key characteristics of air quality. The dataset comprises hourly $PM_{2.5}$ concentration data for Beijing City, collected by the Ministry of Environmental Protection of China from January 1, 2014, to May 28, 2016. The study's objective was to evaluate the performance of the proposed SAE model against other models such as STANN ARMA and SVR. The findings demonstrated that the SAE model

significantly outperformed the other models, showcasing its efficiency in accurately capturing and predicting air quality characteristics. This improvement is attributed to the SAE model's ability to effectively learn the underlying patterns in the air quality data (Sahoo et al. 2016) making it a robust tool for environmental monitoring. In the approach (Yang et al. 2021), a Convolutional Neural Network (CNN) combined with a Long Short-Term Memory (LSTM) network and an Attention mechanism is used to forecast water quality. The convolutional layers extract local features from the data, while the LSTM layers capture long-term correlations. The Attention mechanism dynamically determines the relevance of different time periods. The dataset includes weekly water quality data from the Beilun Estuary, comprising 239 sets of pH, DO, COD, and NH_3-N data over four years. The experimental results indicated that this hybrid model surpassed all baseline models in predicting pH and NH_3-N levels. The model's resilience and generalization capabilities were also highlighted, suggesting its potential for broader application in water quality prediction. The author (Zhao et al. 2020) presents a hybrid ensemble model named CERL, which leverages the strengths of both forward neural networks and recurrent neural networks to handle time-series data for hourly air quality estimation. The dataset is derived from China's online air quality monitoring and analysis system. The CERL model outperformed the baseline models, demonstrating superior predictive accuracy. The hybrid approach combines the temporal pattern recognition ability of recurrent networks with the feature extraction capability of forward neural networks, making it a powerful tool for time-series prediction in environmental data. The STE model (Wang 2018) comprises three components: weather-based partitioning, spatial correlation analysis using granger causalities, and deep LSTM for learning long-term and short-term air quality correlations. The dataset includes air quality data from 35 sites in Beijing, spanning from May 1, 2013, to April 30, 2017. The study found that the STE model effectively utilized air quality attributes to deliver superior predictive performance compared to baseline models. The model's comprehensive approach to incorporating spatial and temporal dependencies makes it a valuable tool for understanding and predicting air quality variations.

The author (Gilik et al. 2022) presented a hybrid machine-learning algorithm that combines CNN and LSTM to observe correlations between various pollutant concentrations. The model uses both spatial and temporal characteristics as inputs to predict pollutant concentrations at multiple locations. The dataset includes air quality data from Barcelona, Kocaeli, and Istanbul. The hybrid model showed an improvement in prediction performance by 11–53% for particulate matter, 20–31% for ozone, 9–47% for nitrogen oxides, and 18–47% for sulfur dioxide, compared to a one-hidden-layer LSTM network. This significant enhancement in predictive accuracy underscores the model's efficacy in dealing with complex environmental data. A Support Vector Machine (SVM) model integrated with Particle Swarm Optimization (PSO) was developed for classifying air quality grades. The model utilizes Partial Least Squares (PLS) to select pertinent data features. The dataset consists of Shenzhen air pollution data from January 1 to December 31, 2018. The experimental findings showed that this approach achieved an accuracy of 93.7%, the highest among the five methods tested, with a maximum error of 1. The high accuracy and low error rate of this model highlight its potential for precise air quality classification, making it a reliable tool for environmental monitoring and decision-making. The CT-LSTM approach combines the Chi-square Test (CT) with a LSTM network (Wang et al. 2021) to construct a prediction model. The dataset includes hourly air quality and meteorological data from January 1, 2017, to December 31, 2018. The results indicated that this novel approach achieved an accuracy of 93.7%, the highest among the five methods, with a maximum error of 1. The CT-LSTM model also demonstrated excellent Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) characteristics, making it highly accurate for predicting AQI. This model's ability to integrate statistical tests with deep learning techniques makes it a robust tool for environmental prediction. The VMDBiLSTM model integrates Variational Mode Decomposition (VMD) with Bidirectional LSTM (BiLSTM) (Zhang et al. 2021) to forecast $PM_{2.5}$ levels. VMD decomposes complex $PM_{2.5}$ time series data based on frequency, while BiLSTM predicts each sub-signal individually to enhance forecasting accuracy. The dataset comprises air quality data from the US Embassy in Beijing, covering 2013 to 2017, with 35,064 hourly samples over 1,461 days. The VMD-BiLSTM model was found to be the most stable and accurate among VMD-integrated forecasting models, combining forward and backward data characteristics in the LSTM neural network. This model's superior predictive performance underscores its potential for accurate and reliable air quality forecasting.

The previous literature encapsulates diverse and advanced methodologies for predicting air and water quality, utilizing a variety of deep learning and machine learning techniques. Each approach leverages different aspects of the datasets to improve predictive accuracy, demonstrating the evolving capabilities of these models in environmental monitoring. The consistent theme across these studies is the integration of spatial and temporal data to capture complex patterns and dependencies, leading to enhanced prediction performance. These advancements in predictive modeling hold significant promise for better understanding and

managing environmental quality, thereby contributing to more informed decision-making and policy development.

3 Proposed Method

We employed machine learning algorithms to model and predict air quality in NCT Delhi India using air pollution data. The data was preprocessed by removing missing values, dealing with outliers, and normalizing it. Feature selection was done using domain knowledge and feature selection techniques such as correlation analysis or mutual information. Random forest algorithm is used to design the proposed model and evaluated using k-fold cross-validation and grid search. Model evaluation metrics used included R^2, mean absolute error, and root mean square error. The paper provides insights into the factors contributing to air pollution in NCT Delhi India. The proposed architecture is shown in Fig. 1. In the experiment, we dealt with continuous and statistical data. We are performing feature extraction by model using random forest algorithms on fixed datasets, as detailed in the part on the system's design as shown in Fig. 1. Our preprocessing steps for our dataset include the type of data and the number of rows and columns. The first layout had 2125 rows and 11 columns that were all float64 types. In addition, we looked at the data to see if any numbers were missing at different points. In addition, we used the min-max normalization technique (Henderi et al. 2021) to bring our sample into the range of 0–1 for use in our regression analyses. The preprocessed dataset was further divided into train, validate, and test sets. The steps involved in the process are described in detail.

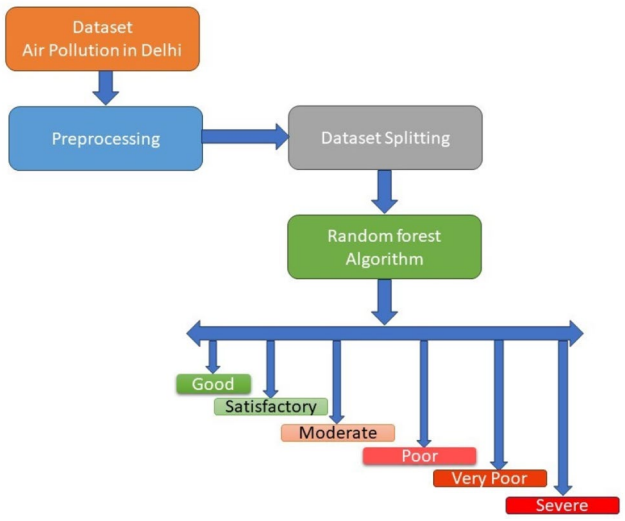

Fig. 1 Architecture of the proposed model

Data Collection Air pollution data for NCT Delhi India was obtained from CPCB, which is the regulatory body responsible for monitoring air quality in India. The data included measurements of pollutants such as $PM_{2.5}$, PM_{10}, NO_2, SO_2, CO, and O_3 from 27 monitoring stations in NCT Delhi India. The data covered the period from January 1, 2014, to December 31, 2018, with hourly measurements.

Data Preprocessing The data was preprocessed using the following steps.

- Missing values were removed by interpolating the missing values using a linear interpolation method.
- Outliers were removed by applying the Tukey's fences method (Adil 2020) with a threshold of 3.
- Data was normalized using min-max normalization.

Random Forest It is a robust machine-learning algorithm used for classification and regression applications. It belongs to the family of ensemble methods, which integrates the predictions of multiple individual models to increase overall accuracy and resiliency. Ensemble learning involves assembling multiple machine learning models also known as "weak learners" and combining their predictions to generate a final prediction that is frequently more accurate and stable than the predictions of individual models. Random Forest employs the "bagging" technique to generate an ensemble of decision trees. Random forest's building components are decision trees. It is a structure similar to a flowchart in which each internal node represents a test on an attribute, each branch represents the result of the test, and each leaf node represents a class label for classification and a numerical value in regression. Bagging is the process of generating multiple subsets of the training data by sampling at random with replacement. Each subset has the same size as the original dataset, but certain data points may appear multiple times and others may be absent. These subsets are used to train decision trees individually. In addition to creating bootstrap samples, random forest further incorporates randomization during training by selecting a random subset of features at each node of the decision tree. This method is referred to as "feature bagging" or "feature randomization". Typically, the quantity of characteristics in each subset is a user-defined hyperparameter. A decision tree is trained using the randomly selected subset of features for each bootstrap sample. The trees are grown until a termination criterion is met, such as reaching a maximum depth or possessing a minimum number of samples at each leaf. Once all individual decision trees have been trained, they predict new data points. In classification, the final prediction is determined by majority voting, in which each tree "votes" for a class and the class with the most votes is selected. In

regression, the final prediction is the mean of the predictions of the individual trees. The algorithm for regression with Random Forest is quite similar to the classification version. The primary difference is in how predictions are made, which involves averaging the output values from individual trees.

We chose the random forest for air quality predictions due to its ability to handle complex, non-linear relationships in data without extensive feature engineering. It is robust to noise and overfitting by averaging predictions from multiple decision trees, ensuring reliable results even with noisy datasets. Additionally, Random Forest offers feature importance analysis, aiding in identifying key pollutants, and supports extensive hyperparameter tuning for optimal accuracy.

Algorithm
Input: X_{train}, Y_{train}, N_{trees}
Output: Randon Forest Model: Forest
Step 1: Create an Empty list to store the ensemble of decision trees: Forest=[ ]
Step 2: For i in range N_{trees}:
Step 3: Randomly sample n samples data points from the training dataset with replacement
Step 4: Randomly selectmax feature from the total features
Step 5: Build a decision tree using the sampled data and features with a depth limit of Max_{depth}
Step 6: Add the trained tree to the forest: Forest.append (tree)

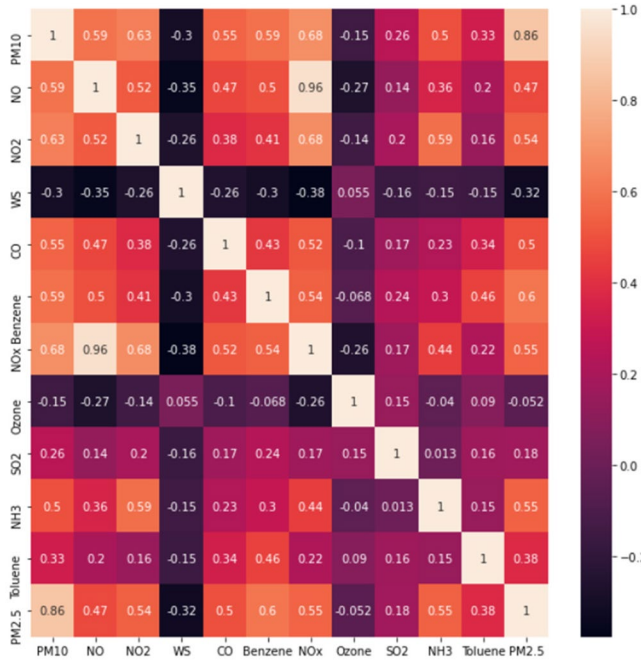

Fig. 2 Represents multicollinearity between different features present in our dataset

Step 7: Repeat step 3-6 for N_{trees}
Step 8: To make a prediction from new input point X_{test}
Step 9: For each tree in the forest:
Step 10: Traverse the tree to reach a leaf node based on the input features.
Step 11: Record the predicted output value associated with the leaf node.
Step 12: Calculate the average of the predicted output values from all trees as the final prediction
End: Returen the Random forest Model : Forest

4 Experiment and Result Discussion

To demonstrate the superiority of the proposed model, we compare our proposed network to several existing approaches that have been designated state-of-the-art in education and describe the datasets and experimental setup we used. The outcomes of the experimental ablation performed on various network components are then described.

Model evaluation: The performance of each model was evaluated using k-fold cross-validation with k = 10. The evaluation metrics: R^2, mean absolute error, and root mean square error were used. The random forest model was found to outperform the other models with an R^2 value of 98.89.

Dataset description: In this experiment, we examine NCT Delhi India's air pollution data (https://github.com/sydney-machine-learning/airpollution_deeplearning) from January 2, 2019, to December 31, 2020. It focused on PM_{10}, NO, NO_2, WS, CO, Benzene, NO_x, Ozone, SO_2, NH_3, Toluene, and $PM_{2.5}$. There are 2115 rows in the data, of which 70% were utilized for training and 30% for testing. The heatmap is shown in Fig. 2.

It is important to understand how different factors contribute to overall $PM_{2.5}$ levels of air pollution. In the histogram, the contrast between the dark and bright areas is important. A powerful positive association is represented by the color dark red, while a weak negative correlation is shown by the colors light red and brown. Since NO is equivalent to NO, NO_2 is equivalent to NO_2, etc., the darkest parts should be 1:1.

Results and Discussion The results showed that the random forest model performed best in predicting air quality in NCT Delhi India, with the weather variables, traffic volume, and time of day being the most important predictors of air pollution. The study provides insights into the factors

contributing to air pollution in NCT Delhi India and can aid in developing effective air pollution control strategies.

Performance Evaluation The performance of the proposed model is evaluated on the following evaluation metrics.

Cross Validation Cross-validation (Arlot 2010) tests trained machine-learning models and independently evaluates their performance. The underlying data collection is separated into training data and test data for this purpose. However, the model's accuracy is then assessed only on the test data set to see how well the model responds to previously unseen data. Cross-validation refers to the capability of measuring the model's accuracy or quality with new, unobserved data during training.

Explained Variance Score It is the share of the variance of the difference between the actual samples of the dataset and the model's predictions (Devi et al. 2019). More significant proportions of explained variance indicate a stronger association. It also implies that you make more accurate predictions.

Mean Squared Logarithmic Error (MSLE) The MSLE starts with pointing out the natural logarithm of each predicted value (Istaiteh et al. 2020). It can also be seen as a way to compare the actual value to the expected value. This value is calculated by Eq. (1) as.

$$MSLE(y,\widehat{y}) = \frac{1}{n}(log\,(y_i+1) - log\,(\widehat{y}_i+1))^2 \tag{1}$$

Where $\widehat{y}_i$ is predicted value.

R-squared (R2) R^2 measures the degree of connection and reliance between two variables. It determines how well one variable explains another. With the R^2 formula, we can determine by Eq. (2) how well the independent variables explain the dependent variable as.

$$R^2 = 1 - \frac{\sum_i (y_i - \widehat{y}_i)^2}{\sum_i (y_i - \bar{y}_i\ )^2} \tag{2}$$

Mean Absolute Error (MAE) The difference between the predicted value and the actual value, expressed in absolute terms, is the total error. MAE gives an idea of how often a forecast is wrong. This is calculated by Eq. (3) as.

$$MAE = \frac{1}{n}\sum\nolimits_{i=1}^{n} |y_i - \widehat{y}_i| \tag{3}$$

where y_i is the actual value and $\widehat{y}_i$ is predicted.

Mean Squared Error (MSE) MSE is a statistic used to quantify the extent of model mistake. To do this, it determines the average squared error between the observed and forecasted values (Minitab 2013). If the model is flawless, then the MSE will be 0. Depending on how off the underlying model is, its value goes up. Mean squared error and mean squared deviation (MSD) both refer to the same metric. This is determined by Eq. (4) as.

$$MSE = \frac{\sum\nolimits_{i=1}^{n} (\boldsymbol{y_i} - \widehat{\boldsymbol{y}}_{\boldsymbol{i}})^2}{\boldsymbol{n}} \tag{4}$$

Root Mean Squared Error (RMSE) RMSE is the most widely used metric for assessing the efficacy of regression models (Minitab 2013). The goal is to quantify the degree to which the model's predictions deviate from the truth compared to independent data. Therefore, a small RMSE is preferred, and a large RMSE is to be avoided. This value is calculated using Eq. (5) as.

$$RMSE = \sqrt{\frac{\sum\nolimits_{i=1}^{n} (y - \widehat{y}_i)^2}{n}} \tag{5}$$

Table 2 demonstrates the use of explained variance, MSLE, R^2, MAE, MSE, and RMSE measures with the testing set at 30% and the training set at 70%. RMSE, MAE, MSLE, MSE, and variance explained are utilized to evaluate the outcomes of each regression model. Based on the R^2 values of linear, DT, RF, KNN, AdaBoost, and SVR models, Figs. 3 and 4 compare the training and testing outcomes. Figure 5 shows the results of a 10-fold cross-validation for these models. The linear, DT, RF, KNN, AdaBoost, and SVR models are compared here where the further findings are reported and evaluated.

Table 2 displays the explained variance, MSLE, R^2, MAE, MSE, and RMSE for the linear, DT, RF, KNN, AdaBoost, and SVR models presented before. The outcomes were favorable, particularly for the Random Forest regression, which performed exceptionally well on five of the six measures employed to evaluate performance.

This model's variance was 0.9943, its R^2 was 0.9941, its MSLE was 0.0001, its MAE was 0.0076, its MSE was 0.0055, and its RMSE was 0.0744. Comparing the regressions based on variance explained and R^2 values reveals KNN to be the weakest. This model achieved an R^2 value of 0.9418 and an explained variance of 0.9459. Compared to other regressions, the performance of the LR, DT, and AdaBoost regressions is around average. Regarding the

Table 2 Training and testing results of different regressions

	Measures	Variance	MSLE	R^2	MAE	MSE	RMSE
Training results	Linear Regression	0.9684	0.0038	0.9684	0.3646	0.2058	0.4536
	Decision Tree Regression	0.9884	0.0001	0.9884	0.0021	0.0019	0.0595
	Random Forest Regression	0.9972	0.0001	0.9972	0.0025	0.0009	0.0295
	K-Nearest Neighbour Regression	0.9452	0.0001	0.9446	0.0084	0.0053	0.0731
	AdaBoost Regression	0.9691	0.0045	0.9690	0.4441	0.3176	0.5635
	Support Vector Regression	0.9767	0.0479	0.9702	0.9814	0.8129	0.9998
Testing results	Linear Regression	0.9666	0.0038	0.9666	0.3661	0.2067	0.4547
	Decision Tree Regression	0.9649	0.0002	0.9649	0.0074	0.0097	0.0985
	Random Forest Regression	0.9943	0.0001	0.9941	0.0076	0.0055	0.0744
	K-Nearest Neighbour Regression	0.9459	0.0002	0.9418	0.0103	0.0077	0.0876
	AdaBoost Regression	0.9667	0.0046	0.9667	0.4459	0.3198	0.5655
	Support Vector Regression	0.9702	0.0379	0.9700	0.9819	0.7136	0.8968

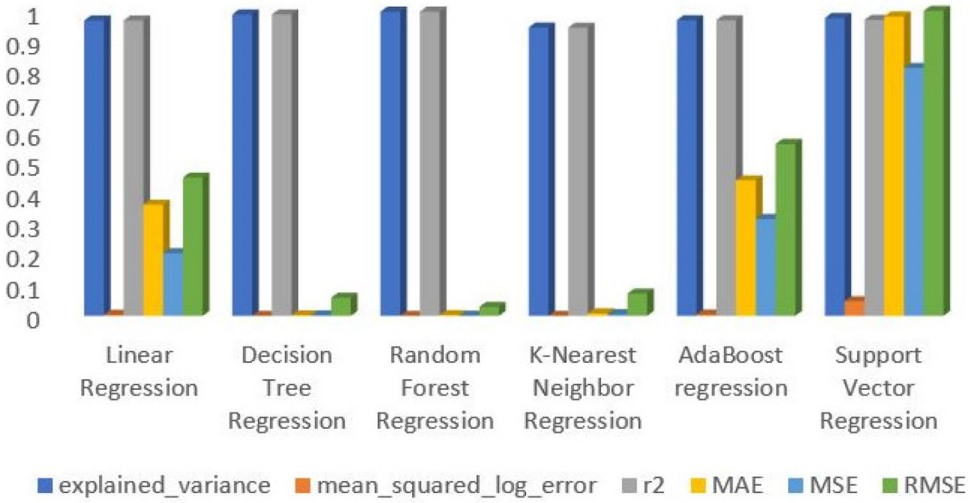

Fig. 3 Training results of different regression

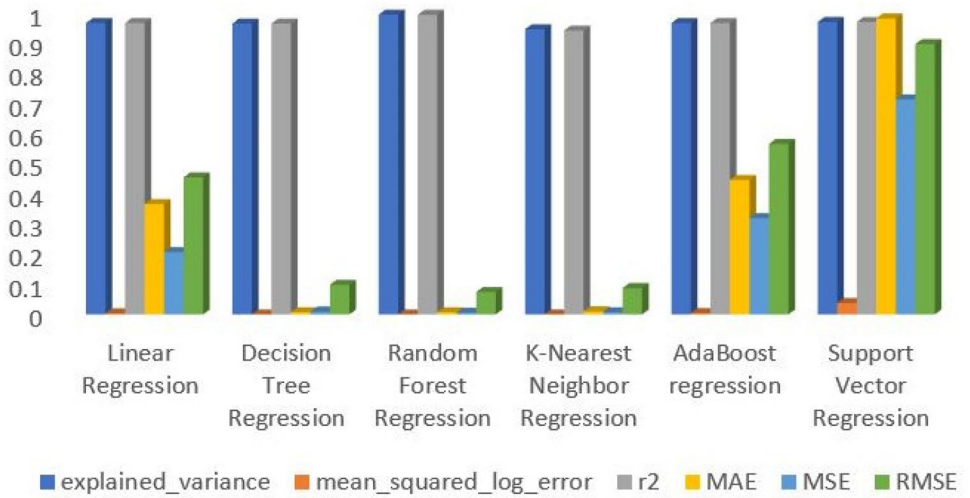

Fig. 4 Testing results of different regression

strictest R^2 metric, the outputs for the RF, DT, KNN, SVR, Linear, and AdaBoost models are identical, with corresponding values of 0.9943, 0.9649, 0.9459, 0.9702, 0.9666, and 0.9667. Here is a breakdown of the different regression results so that a more comprehensive evaluation of the proposed method can be performed. When we compare the R^2 score to the explained variance score, we are effectively checking the mean error; if both scores are identical, then the mean error is likewise equal to zero. In other words, the explained variance uses the biased variance to

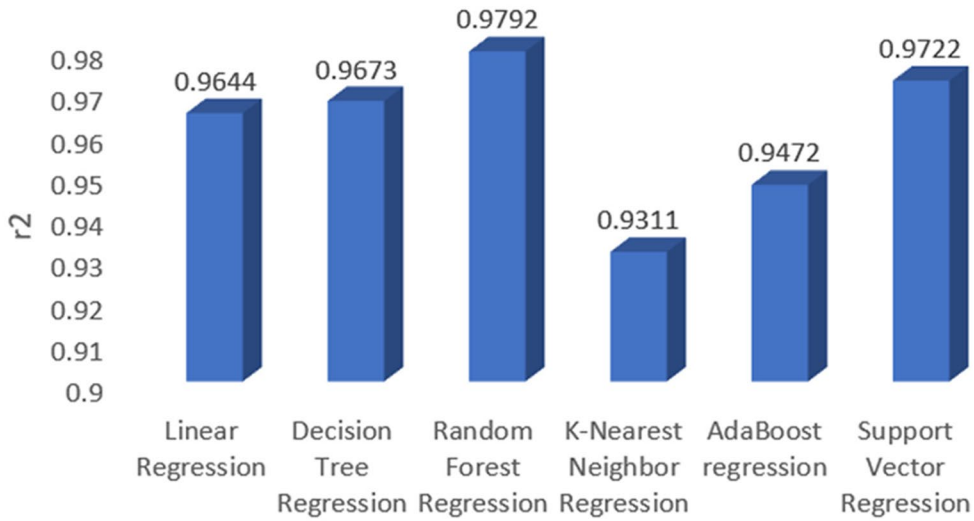

Fig. 5 10-fold cross validation results of different regression

produce an explanation percentage. If the predictor's error was genuinely random, both scores should be identical. On the other hand, RF and KNN models are biased because their explained variance and R^2 scores are 0.9943, 0.9941, and 0.9459, and 0.9418, respectively. According to Table 2; Figs. 3 and 4, the explained variance and R^2 values for the linear, DT, AdaBoost, and SVR models are 0.9666, 0.9649, 0.9667, and 0.9702, respectively. The findings of the studies reveal that the linear, DT, AdaBoost, and SVR models are unbiased, as their explained variance and R^2 values are 0.9666, 0.9649, 0.9667, and 0.9702, respectively. The significance (presented in bold text) of in Table 2 is only to emphasize the results because the table contains six metrics, six regressors, and their values. As we can see, out of the six, five values in column three are bold because, during the training of different regressors, only the Random Forest got the best five metrics. Similarly, when testing, the random forest got the best results for five metrics, while the decision tree got only one.

On the MSLE metric, the findings are comparable to those of the RF, DT, KNN, Linear, Ada Boost, and SVR models, with values of 0.0001, 0.0002, 0.0002, 0.0038, 0.0046, and 0.0379, respectively, supporting a rigorous R^2 measure. The evaluation of performance was based on a total of six distinct indicators. The random forest regression performed well in five of the metrics but not so well in the MAE metric, where the decision tree performed better. The MAE for the linear, DT, RF, KNN, Ada Boost, and SVR models are 0.3661, 0.0074, 0.0076, 0.0103, 0.4459, and 0.9819, respectively. According to the results, most MSEs and RMSEs for all regressions fall between the range (0.0055, 0.7135) and (0.0744, 0.8968), respectively. Figure 5 displays R^2-based 10-fold cross-validation results for linear, DT, RF, KNN, AdaBoost, and SVR. All of these regressions performed well in 10-fold cross-validation, but the Random Forest regression performed exceptionally well on both the testing and validation sets. The validation results for the linear, DT, RF, KNN, Ada boost and SVR regressions are as follows: 0.964481347, 0.967303552, 0.979033379, 0.931154338, and 0.947219125, respectively.

As shown in Fig. 6, Random Forest beats Linear, DT, KNN, AdaBoost, and SVR in terms of R^2 values for training, testing, 10-fold, and GridSearchCV findings. The GridSearchCV results for linear, DT, RF, KNN, Ada boost, and SVR are 0.965047, 0.967927, 0.988991, 0.938107, and 0.958116, respectively. Random Forest's GridSearchCV result is 0.988991, while its train score is 0.9972, the test score is 0.9941, and the 10-fold score is 0.975016297. From Table 2; Figs. 5 and 6, and 7, we can infer that the random forest regression performed exceptionally well on five of the six criteria used to assess performance and was validated by 10-fold cross-validation and GridSearchCV results.

5 GridSearchCV Outcomes and Comparison of Various Regressions

Linear Regression: As stated previously, cross-validation using GridSearchCV used to evaluate the accuracy of linear regression by utilizing various variables that can serve as predictors. To provide the range of to-be-tuned hyperparameters, use the syntax hyperparameters = n features followed by a list range (1, 10). The GridSearchCV method was invoked with the following parameters: n estimators = 10, random state 0, estimator RFE = hyperparameters, scoring R^2, and verbose equals 1. With nine candidates, ninety fits were achieved by fitting the model with ten folds for each candidate. The function GridSearchCV produced the following information: RFE (estimator = linear regression ()) "n_ features _to_ select": 7, best_ parameters (1), mean_ score

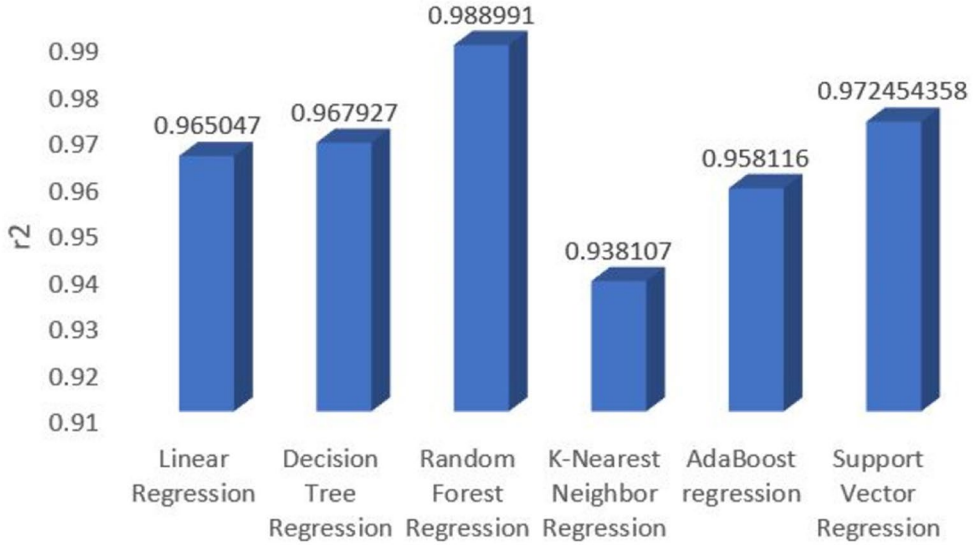

Fig. 6 GridSearchCV results of different regression models

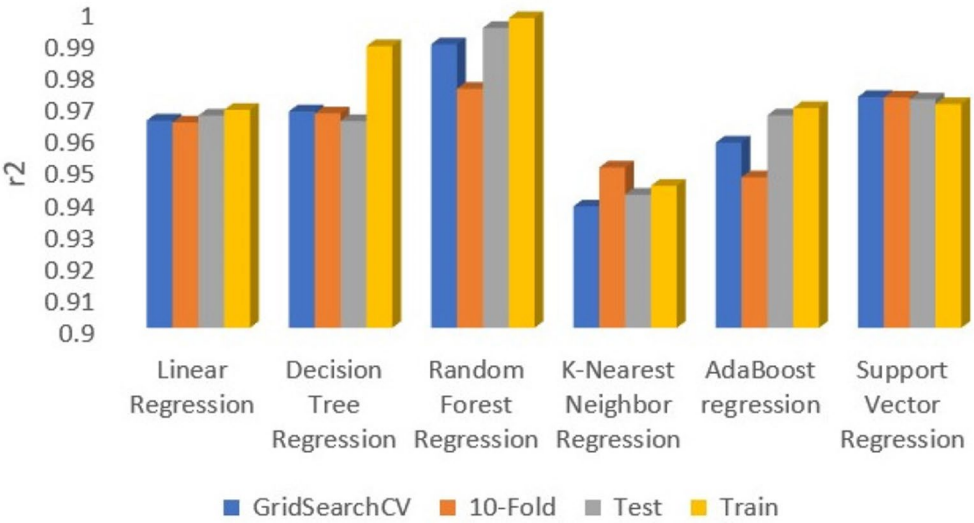

Fig. 7 Train, test, 10-fold, and GridSearchCV results of different regression based on R^2 values

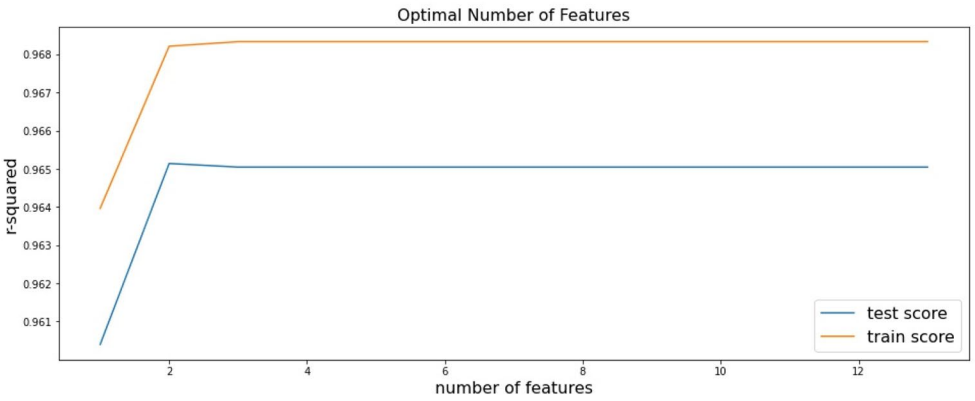

Fig. 8 Mean train and mean test score of R^2 for linear regression

over time (0.000506), and best_ score obtained (0.965047), an optimal number of features selected is 9. Even though GridSearchCV returns 22 predictors, we validated the findings by plotting the test score versus the training score, as shown in Fig. 8.

Decision Tree: GridSearchCV-based cross-validation issued to evaluate the accuracy of the Random Forest Regression with folds = K Fold (n splits = 5, shuffle = True, random state = 10), as described in the section preceding this one. GridSearchCV with Hyperparameters (CV = 10;

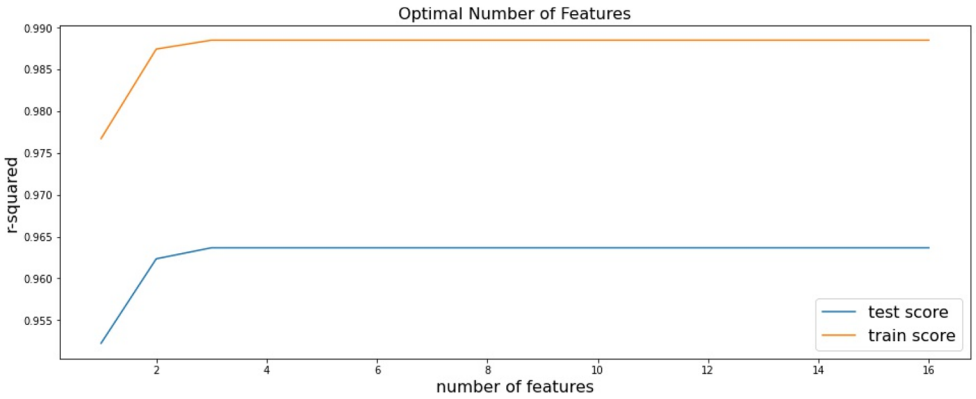

Fig. 9 Mean train and mean test score of R^2 for DT

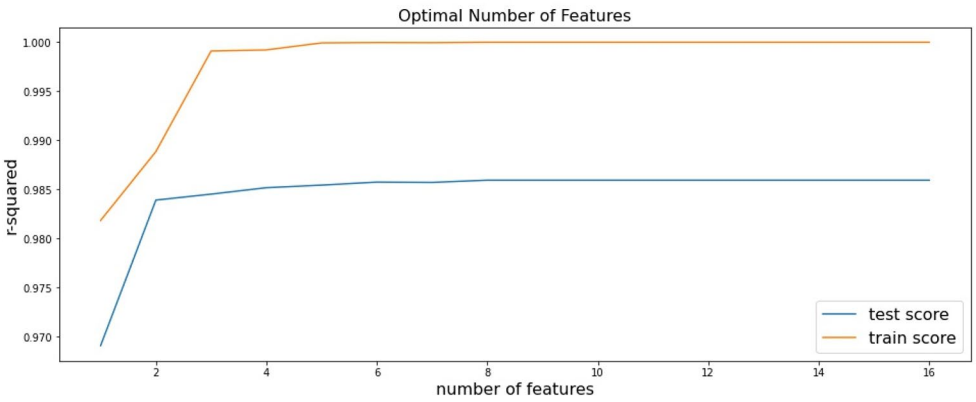

Fig. 10 Mean train and mean test score of R^2 for RF

estimator = Decision Tree Regression (max depth = 10; random state = 10); n jobs = 1; param grid = max depth range (1, 11); max features range (4, 10) with ten folds for each of the six candidates, a total of 360 suitable candidates were obtained. RFE GridSearchCV provided the values "max_ depth": 5, "max_ features": 9, "mean_ score _time" (0.003122) s, and "best_ score achieved" (0.967927). Even though GridSearchCV 21 predictors, we validated the results by plotting the mean test score with the mean train score, as indicated in Fig. 9.

Random Forest: Cross-validation via GridSearchCV is used to assess the accuracy of Random Forest Regression. The GridSearchCV function's parameters are estimator = Random Forest Regression (n_ estimators = 10, random state = 10), n-jobs = 1, param grid = max depth range (2, 10), max feature range (4, 11), return_ train_ score = True, verbose = True, cv = 5, scoring set to R^2. The model was fitted using ten folds for each of the eight possibilities, for a total of 560 fits. The function GridSearchCV returned the following data: RFE max depth: 6, max features: 6, mean score time (0.001058) s, and best score (0.988991). Despite the fact that GridSearchCV returns 21 predictors, we validated the results by creating a graph between the mean test score and mean train score, as represented in Fig. 10.

K-Nearest Neighbor: Cross-validation using GridSearchCV is used to test the KNN model's accuracy. A cross-validation plan was created using the formula cv = K shuffle = True, estimator = RF Estimator = K Neighbors Regression (n_ neighbors = 10), param-grid = n- features to pick [1–11]; return train score = True; scoring = R^2, verbose = 1; When the model was fitted using ten folds for each candidate and 10 candidates, 500 fits were achieved. GridSearchCV returned the following results: best_ params (knn_ n_ neighbors: 9), mean_ score _time (0.002053) s, and best_ score obtained (0.938107). Despite the fact that GridSearchCV returns 21 predictors, we validated the results by plotting the test score versus the training score, as represented in Fig. 11.

AdaBoost Regression: Cross-validation with GridSearchCV is used to assess the AdaBoost regression model's accuracy. This cross-validation strategy was computed using the following parameters: estimator = RFE (Ada Boost Regression; (random state = 0, n-estimators = 10), parameter grid = hyperparameters return-train-score = True, scoring = R^2, verbose = 1, cv = folds (n_ splits = 10, shuffle = True, random_ state = 10), cv = folds (n_ splits = 10, shuffle = True, random_ state = 10), cv = folds (n_ splits = 10, shuffle = True, random_ state = 10). GridSearchCV produced the following outcomes: best-params learning rate 0.1, n-estimators: 10, mean score time (0.002026s), and best score (0.958116). Despite the fact that GridSearchCV returns 21 predictors,

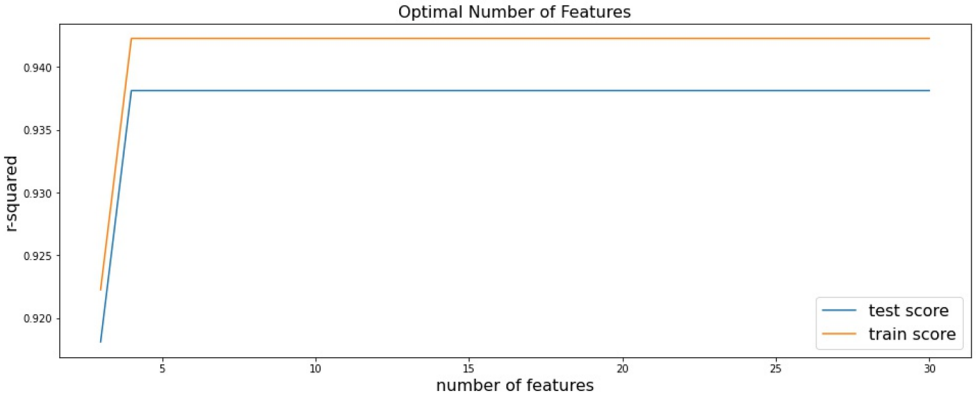

Fig. 11 Mean train and mean test score of R^2 for KNN

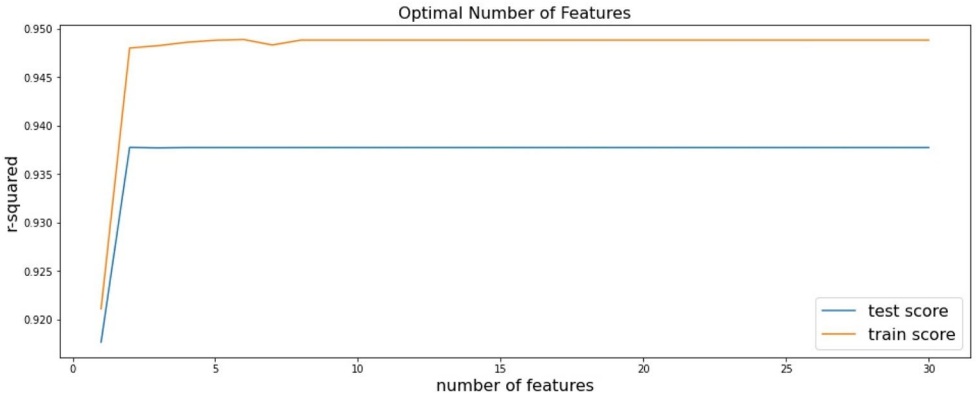

Fig. 12 Mean train and mean test score of R^2 for AdaBoost

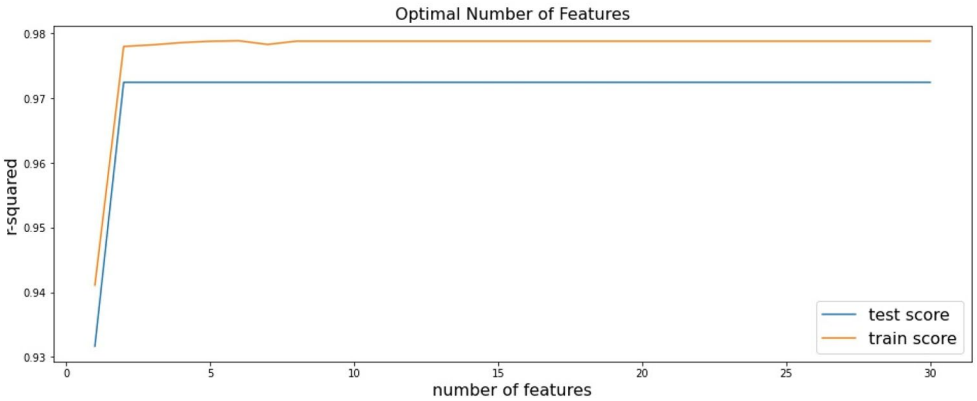

Fig. 13 Mean train and mean test score of R^2 for SVR

we validated the results by plotting the test score versus the training score, as represented in Fig. 12.

Support Vector Regression: Cross-validation with GridSearchCV is utilized to evaluate the effectiveness of the support vector model. Hyperparameters can aid in parameter selection. As indicated, GridSearchCV's hyperparameters have been predefined. We did this by constructing a dictionary with the potential values for each hyperparameter. Here's one instance: C {0.01, 0.01, 0.1, 1,1.5,2}, gamma {0.01,0.001, 0.001, 0.1, 1}, kernel {RBF, C, gamma}, and kernels are hyperparameters, the others have default values in an SVM model. GridSearchCV employs cross-validation to examine each potential dictionary value combination. This function produces the GridSearchCV with the following parameters: best_ params 'C': 1, 'gamma': 0.01, mean_ score _time (0.000912)s, and best score obtained (0.972454358). Even though GridSearchCV returns 21 predictors, we validated the results by plotting the test score versus the training score, as indicated in Fig. 13.

In the proposed CI-based models, we used linear, DT, RF, KNN, AdaBoost, and SVR machine learning prediction models. RMSE, MAE, MSLE, MSE, and explained variance are used to judge the results of each regression model. The results were positive overall, especially for the Random Forest regression, which excelled across five of the six metrics used to evaluate performance. This random forest had a

variance of 0.9943, an R^2 of 0.9941, an MSLE of 0.0001, an MAE of 0.0076, an MSE of 0.0055, and an RMSE of 0.0744. The performance of the LR, DT, and AdaBoost regressions is roughly average when compared to other regressions. The results from the RF, DT, KNN, SVR, Linear, and AdaBoost models are in terms of the most stringent R^2 metric, with values of 0.9943, 0.9649, 0.9459, 0.9702, 0.9666, and 0.9667, respectively. All of these regressions did well in 10-fold cross-validation, but the Random Forest regression did especially well on both the testing and validation sets. For the linear, DT, RF, KNN, Ada boost, and SVR regressions, the validation results are 0.964481347, 0.967303552, 0.979033379, 0.931154338, and 0.947219125. For linear, DT, RF, KNN, Ada boost, and SVR, the GridSearchCV results are 0.965047, 0.967927, 0.988991, 0.938107, and 0.958116, respectively. The GridSearchCV result for Random Forest is 0.988991. Its train score is 0.9972, its test score is 0.9941, and its 10-fold score is 0.975016297. From the above results, we can see that the random forest regression did very well on five of the six criteria used to measure performance, and this was confirmed by the 10-fold cross-validation and GridSearchCV results.

6 Conclusion

In this paper, we have demonstrated the effectiveness of random forest algorithms for air pollution modeling and prediction in National Capital Territory (NCT), Delhi, India. We evaluated multiple algorithms, including linear regression, decision tree, K-nearest neighbors, and support vector regression, and found that the random forest regression algorithm was the most effective for predicting air quality with an R^2 value of 98.89. Our results provide valuable insights into the most significant contributors to poor air quality in NCT Delhi India. We analyzed on the basis of the correlation matrix that PM2.5 and NO2 are the most important factors for pollution. We also provide a comparative analysis of multiple algorithms, which can guide future research in this field. Moreover, our paper highlights the importance of rigorous cross-validation and hyper-parameter tuning to ensure the accuracy and generalizability of the models. Cross-validation is used to reduce the biases of the proposed model. Further research is needed to explore the application of these algorithms in other geographical locations and to investigate the potential of ensemble learning approaches to further improve the accuracy of air pollution models. Limitations of the study, such as the limited scope of the data or potential biases in the sampling methods were also discussed, and future research directions, such as exploring other machine learning algorithms or incorporating additional data sources such as meteorological data or satellite data were identified.

Declarations

Conflict of interest The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

Adil IH, Zaman A (2020) Outliers detection in skewed distributions: split sample skewness based boxplot. Econ Comput Econ Cybern Stud Res 54(3). https://doi.org/10.24818/18423264/54.3.20.08

Andrew AM (2001) Backpropagation. Kybernetes. https://doi.org/10.1108/K-01-2001-0017

Arlot S, Celisse A (2010) A survey of cross-validation procedures for model selection. Stat Surv 4:40–79. https://doi.org/10.1214/09-SS054

Bagwari N, Kumar S, Verma VS (2023) A comprehensive review on segmentation techniques for satellite images. Arch Comput Methods Eng 30:4325–4358. https://doi.org/10.1007/s11831-022-09918-w

Borbet TC, Gladson LA, Cromar KR (2018) Assessing air quality index awareness and use in Mexico City. BMC Public Health 18:5418. https://doi.org/10.1186/s12889-018-5418-y

Chakrabarti S, Singh AP, Kumar S (2015) Assessment of air quality in Haora river basin. Pollut Res 34(2):425–432

CPCB (2019) Central pollution control board, ministry of environment, forest, and climate change, government of India. http://www.cpcb.nic.in/. Accessed 20 Jan 2019

Deak G et al (2020) Meteorological parameters and air pollution in urban environments in the context of sustainable development. IOP Conf Ser: Earth Environ Sci 616(1):012003. https://doi.org/10.1088/1755-1315/616/1/012003

Devi MS et al (2019) Linear attribute projection and performance assessment for signifying the absenteeism at work using machine learning. Int J Recent Technol Eng 8(3):1262–1267

Georgiou GK, Kushta J, Christoudias T, Proestos Y, Lelieveld J (2020) Air quality modelling over the Eastern mediterranean: seasonal sensitivity to anthropogenic emissions. Atmos Environ 222:117119. https://doi.org/10.1016/j.atmosenv.2019.117119

Gilik A, Ogrenci AS, Ozmen A (2022) Air quality prediction using CNN+LSTM-based hybrid deep learning architecture. Environ Sci Pollut Res 29(8):11920–11938. https://doi.org/10.1007/s11356-021-16500-6

Gupta IK, Yadav V, Kumar S (2019) Medical data clustering based on particle swarm optimisation and genetic algorithm. Int J Adv Intell Paradigms 14(3–4):345–358. https://doi.org/10.1504/IJAIP.2019.097222

Henderi H, Wahyuningsih T, Rahwanto E (2021) Comparison of Min-Max normalization and Z-Score normalization in the K-nearest neighbor (kNN) algorithm to test the accuracy of types of breast Cancer. Int J Inf Inform Syst 4(1):13–20

Ibrahim AU et al (2021) Pneumonia classification using deep learning from chest X-ray images during COVID-19. Cogn Comput. https://doi.org/10.1007/s12559-021-09900-1

Istaiteh O et al (2020) Machine learning approaches for COVID-19 forecasting. In: 2020 International conference on intelligent data science technologies and applications (IDSTA 2020), pp 50–57. https://doi.org/10.1109/IDSTA49946.2020.9337144

Jiang T, Gradus JL, Rosellini AJ (2020) Supervised machine learning: A brief primer. Behav Ther 51(5):675–687. https://doi.org/10.1016/j.beth.2019.10.001

Jo J, Jo B, Kim J, Kim S, Han W (2020) Development of an IoT-Based indoor air quality monitoring platform. J Sens 2020:8749764. https://doi.org/10.1155/2020/8749764

Katoch S, Chauhan SS, Kumar V (2021) A review on genetic algorithm: past, present, and future. Multimed Tools Appl 80(5):8091–8126. https://doi.org/10.1007/s11042-020-10139-6

Khomenko S, Breatnach S, Delhomme L, Kossakowski J, Smirnova A (2020) Predictive models in air quality monitoring: Data-driven approaches and developments. Environ Int 135:105491. https://doi.org/10.1016/j.envint.2019.105491

Kollmannsberger S, D'Angella D, Jokeit M, Herrmann L (2021) Neural networks. Studies in computational intelligence. Springer, Cham, pp 19–45. https://doi.org/10.1007/978-3-030-83560-0_2

Kousi T et al (2022) Comparative study of deep learning algorithms for PM2.5 prediction. Environ Sci Pollut Res 29(10):14644–14657. https://doi.org/10.1007/s11356-021-14195-7

Kumar S, Rastogi U (2023) A comprehensive review on the advancement of high-dimensional neural networks in quaternionic domain with relevant applications. Arch Comput Methods Eng 30(6):3941–3968. https://doi.org/10.1007/s11831-022-09822-3

Kumar S, Tripathi BK (2017) Machine learning with resilient propagation in quaternionic domain. Int J Intell Eng Syst 10(4)

Kumar S, Tripathi BK (2018) On the root-power mean aggregation based neuron in quaternionic domain. Int J Intell Syst Appl 10(7):11

Lee H, Lee H, Lee H (2021) A deep learning-based prediction model for air quality using LSTM and attention mechanism. Sci Total Environ 747:141054. https://doi.org/10.1016/j.scitotenv.2020.141054

Li Z, Xie J, Wang Z, Zhang B, Yang L (2019) Ensemble deep learning for PM2.5 prediction. J Clean Prod 239:118071. https://doi.org/10.1016/j.jclepro.2019.118071

Maurya SP, Sisodia PS, Mishra R, Singh DP (2024) Performance of machine learning algorithms for lung cancer prediction: a comparative approach. Sci Rep 14(1):58345. https://doi.org/10.1038/s41598-024-58345-8

Minitab (2013) Regression analysis: how do I interpret R-squared and assess the goodness-of-fit? Regres Anal, 1–6

Pedrycz W, Sillitti A, Succi G (2016) Computational intelligence: an introduction. Studies in computational intelligence. Springer, Cham, pp 13–31. https://doi.org/10.1007/978-3-319-37283-7_1

Sahoo SK, Kumar AV, Yadav AK, Tripathi RM (2016) Metal characterization of airborne particulate matters in a coastal region. Toxicol Environ Chem 98(7):768–777. https://doi.org/10.1080/02772248.2016.1210331

Sahoo PK et al (2021) COVID-19 pandemic: an outlook on its impact on air quality and its association with environmental variables in major cities of Punjab and Chandigarh, India. Environ Forensics 22(1–2):143–154. https://doi.org/10.1080/15275922.2021.1903285

Singh S, Tripathi BK (2022a) Deep quaternion residual learning for breast Cancer classification. Int J Comput Inf Syst Ind Manag Appl 14:262–269

Singh S, Tripathi BK (2022b) Pneumonia classification using quaternion deep learning. Multimed Tools Appl 81(2):1743–1764. https://doi.org/10.1007/s11042-021-11473-y

Singh S et al (2021) Evaluation of transfer learning based deep learning architectures for waste classification. In: 2021 4th Int Symp Adv Electr Commun Technol ISAECT 2021. https://doi.org/10.1109/ISAECT53624.2021.9670967

Singh S, Kumar S, Tripathi BK (2024) A comprehensive analysis of quaternion deep neural networks: architectures, applications, challenges, and future scope. In: Archives of computational methods in engineering, pp 1–28. https://doi.org/10.1007/s11831-024-10216-1

Wang J, Song G (2018) A deep Spatial-Temporal ensemble model for air quality prediction. Neurocomputing 314:198–206. https://doi.org/10.1016/j.neucom.2018.05.089

Wang J et al (2021) Air quality prediction using CT-LSTM. Neural Comput Appl 33(10):4779–4792. https://doi.org/10.1007/s00521-020-05432-y

Yadav AK, Jamal A (2016) A review on the present scenario of air quality associated with Indian mining operations. Environ Qual Manag 25(3):99–105. https://doi.org/10.1002/tqem.21471

Yadav AK et al (2019) Assessment of particulate matter, metals of toxicological concentration, and health risk around a mining area, Odisha, India. Air Qual Atmos Health 12(7):775–783. https://doi.org/10.1007/s11869-019-00722-5

Yang Y et al (2021) A study on water quality prediction by a hybrid CNN-LSTM model with attention mechanism. Environ Sci Pollut Res 28(39):55129–55139. https://doi.org/10.1007/s11356-021-14418-1

Ye Q et al (2021) High-resolution modeling of the distribution of surface air pollutants and their intercontinental transport by a global tropospheric atmospheric chemistry source-receptor model (GNAQPMS-SM). Geosci Model Dev 14(12):7573–7604. https://doi.org/10.5194/gmd-14-7573-2021

Zhang H et al (2020) Long Short-Term memory (LSTM)-based air quality forecasting models: A review. Atmosphere 11(3):342. https://doi.org/10.3390/atmos11030342

Zhang Z et al (2021) A hybrid deep learning technology for PM2.5 air quality forecasting. Environ Sci Pollut Res 28(4):39409–39422. https://doi.org/10.1007/s11356-020-12086-w

Zhao Z et al (2020) Combining forward with recurrent neural networks for hourly air quality prediction in Northwest of China. Environ Sci Pollut Res 27(23):28931–28948. https://doi.org/10.1007/s11356-020-09168-y