Volume 11, Issue 2, April 2024 ISSN: 2348-8565 (Online)
International Journal of Modern Engineering and Research Technology
Website: http://www.ijmert.org Email: editor.ijmert@gmail.com

Analyzing Indian Air Quality: Exploratory Data Analysis and Regression Approach

Rama Dwivedi
Research Scholar, M.Tech.
Computer Science and Engineering
Takshshila Institute of Engineering and Technology
Jabalpur (M.P), India
Email: ramagautamdwivedi@gmail.com

Swati Soni
Assistant Professor
Department of Computer Science and Engineering
Takshshila Institute of Engineering and Technology
Jabalpur (M.P), India
Email: swatisoni@takshshila.org

ABSTRACT

The Air Quality Index (AQI) is a vital resource for evaluating air quality and the health hazards it poses, giving the general public and decision-makers access to crucial data. This research employs machine learning methods to forecast AQI levels in India. The data set, which is derived from the India Air Quality Index Data Source and covers the years 1990 to 2015, includes measurements of the AQI as well as other pollutants, including SO2, NO2, RSPM, and SPM.

The goal of the research is to capture the complex relationship between pollutant concentrations and AQI by using a regression framework to construct robust AQI prediction models. This work determines the most efficient machine learning techniques for AQI prediction by a thorough examination that includes exploratory data analysis (EDA), data preprocessing, and model validation. The procedure is described in a proposed framework, which includes processes for preprocessing the data, feature selection, creating the model, and evaluating it using metrics like R-squared (R^2), Root Mean Squared Error (RMSE), and accuracy of cross-validation. Four regression models are presented in the results: the XGB Regressor, the Random Forest Regressor, the Decision Tree Regressor, and the Linear Regressor. RF tested with RMSE values of 1.08276 and 1.82496 for training and testing, respectively, and R-squared values of 0.999879 and 0.99965for training and testing, also Random Forest Regressor is the model with the highest mean accuracy (k-fold cross validation CV=5) and precision of 99.8864 among them with RMSE: 3.681621343341035 and R-squared: 0.9986115689626822. This demonstrates how well it predicts AQI levels. The study emphasizes how important precise AQI prediction models are for guiding the development of air quality control strategy decision-making procedures. This research helps to lessen the negative consequences of air pollution on public health and the environment in India by giving policymakers and environmental authorities trustworthy tools.

Keywords:— *Air Quality Index (AQI), Machine Learning, Regression Analysis, Cross-Validation, RMSE, R-Squared*

I. INTRODUCTION

AQI Introduction

AQI stands for Air Quality Index. It is a standardized index used to communicate and assess the quality of the air in a specific location. The AQI is typically used to provide information about how polluted

or clean the air is and what associated health effects might be of concern to the public. The AQI is calculated based on the concentrations of various air pollutants, including particulate matter (PM2.5 and PM10), ground-level ozone (O3), sulfur dioxide (SO2), nitrogen dioxide (NO2), and carbon monoxide (CO).

AQI Launched in April 2015. With 14 Cities (Now more than 71 cities involved in it). AQI is a tool for the effective communication of air quality status to people in terms, which is easy to understand.

AQI Transforms complex air quality data of various pollutants into a single number called Index Value, Nomenclature and Color. The AQI is usually divided into several color-coded categories, such as "Good," "Moderate," "Unhealthy for Sensitive Groups," "Unhealthy," "Very Unhealthy," and "Hazardous," each representing a different level of health concern. People can use the AQI to make informed decisions about outdoor activities and to take precautions when air quality is poor, especially for vulnerable groups like children, the elderly, and individuals with respiratory or heart conditions.

Different countries may have their own systems for calculating and reporting AQI, but the basic concept is similar worldwide, providing a way to communicate air quality information to the public in a simple and understandable manner.

Air Quality Index (AQI) serves as a numerical scale designed to communicate the current or forecasted level of air pollution. This index is derived from the concentrations of various air pollutants, with each pollutant having its own subindex. Common pollutants considered in the AQI calculation include ground-level ozone, particulate matter (PM10 and PM2.5), sulfur dioxide (SO_2), nitrogen dioxide (NO_2), and carbon monoxide (CO). These subindices are then collectively used to determine the overall AQI value, providing a simplified and standardized way to convey the overall air quality to the public. The AQI is widely employed by environmental agencies to inform the public about potential health risks associated with different levels of air pollution. The highest subindex among the individual pollutants dictates the categorization of the overall AQI, allowing for clearer communication of the potential health implications of the observed air quality at a specific location and time.

List of 8 Pollutants of AQI are:

1. PM10: Particulate Matter with a diameter of 10 micrometers or less.
2. PM2.5: Particulate Matter with a diameter of 2.5 micrometers or less.
3. NO_2: Nitrogen Dioxide.
4. SO_2: Sulfur Dioxide.
5. CO: Carbon Monoxide.
6. O_3: Ozone.
7. NH3: Ammonia.
8. Lead (Pb): In the context of air quality, it refers to lead pollution.

Air Quality Index (AQI) Values	Levels of Health Concern	Colors
When the AQI is in this range:	*...air quality conditions are:*	*...as symbolized by this color:*
0 to 50	Good	Green
51 to 100	Moderate	Yellow
101 to 150	Unhealthy for Sensitive Groups	Orange
151 to 200	Unhealthy	Red
201 to 300	Very Unhealthy	Purple
301 to 500	Hazardous	Maroon

Figure 1: AQI Categories & Colors

Air Quality Index

Levels of Health Concern	Numerical Value	Meaning
Good	0 to 50	Air quality is considered satisfactory, and air pollution poses little or no risk.
Moderate	51 to 100	Air quality is acceptable; however, for some pollutants there may be a moderate health concern for a very small number of people who are unusually sensitive to air pollution.
Unhealthy for Sensitive Groups	100 to 151	Members of sensitive groups may experience health effects. The general public is not likely to be affected.
Unhealthy	151 to 200	Everyone may begin to experience health effects; members of sensitive groups may experience more serious health effects.
Very Unhealthy	201 to 300	Health warnings of emergency conditions. The entire population is more likely to be affected.
Hazardous	301 to 500	Health alert: everyone may experience more serious health effects.

Figure 2 : AQI Associated Health Impacts

Particulate Matter

1. ***Suspended Particulate Matter (SPM):*** SPM refers to a broad category of airborne particles that are suspended in the atmosphere. These particles can vary widely in size and composition. They include both coarse and fine particles, ranging from large dust particles to smaller particles that can be inhaled into the respiratory system.

2. ***Respirable Suspended Particulate Matter (RSPM):*** RSPM is a subset of SPM that specifically refers to particles that are small enough to penetrate deep into the lungs when inhaled. These particles are often considered more harmful to human health because of their ability to reach the lower respiratory tract. RSPM includes particles with a diameter less than 10 micrometers (PM10) and even finer particles, such as PM2.5.

3. ***PM2.5:*** This term refers to particulate matter with a diameter of 2.5 micrometers or smaller. These fine particles can penetrate deep into the respiratory system and even enter the bloodstream, posing health risks. Sources of PM2.5 include vehicle emissions, industrial processes, and combustion of fossil fuels.

4. ***PM10:*** PM10 encompasses particles with a diameter of 10 micrometers or smaller. This includes both coarse and fine particles. While larger particles may settle in the upper respiratory tract, smaller particles in this range, like PM2.5, can reach the lower respiratory tract and potentially cause health issues. Sources of PM10 include dust from construction sites, road dust, and industrial activities.

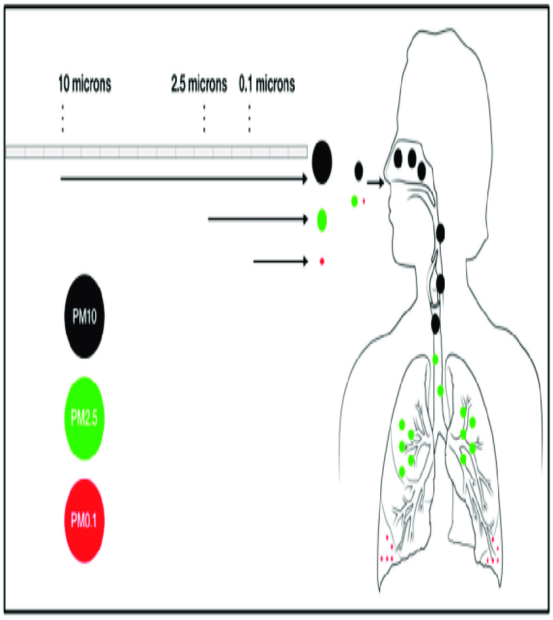

Figure 3 : Classification-of-particulate-matter-according-to-aerodynamic-diameter

In summary, SPM is a general term for all suspended particles in the air, while RSPM specifically focuses on particles that are respirable and can deeply penetrate the lungs. PM2.5 and PM10 are subsets of both SPM and RSPM, representing finer and coarser particles, respectively. Monitoring and controlling these particulate matters are crucial for understanding and addressing air quality and public health concerns.

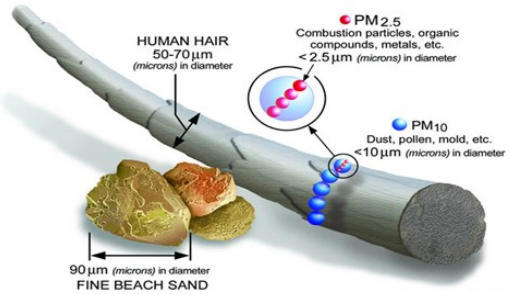

Figure 4 : Particulate Matter (PM) Sizes

Machine Learning

Machine learning is a subset of artificial intelligence (AI) that focuses on the development of algorithms and models that enable computers to learn from data and make predictions or decisions without being explicitly programmed for the given task. The fundamental idea behind machine learning is to allow systems to automatically improve their performance over time as they are exposed to more data.

Supervised Learning: In supervised learning, the algorithm is trained on a labeled dataset, where the input data is paired with corresponding output labels.The goal is for the model to learn a mapping from inputs to outputs, enabling it to make accurate predictions on new, unseen data.

Unsupervised Learning: In unsupervised learning the algorithm is trained on an unlabeled dataset, and it must find patterns, structure, or relationships within the data without explicit guidance.

Common tasks include clustering similar data points together or reducing the dimensionality of the data.

Reinforcement Learning: The algorithm learns by interacting with an environment and receiving feedback in the form of rewards or penalties based on its actions. The goal is to learn a policy that guides the agent (algorithm) to take actions in the environment to maximize cumulative rewards over time.

Machine learning is a subset of artificial intelligence (AI) that focuses on the development of algorithms and models that enable computers to learn from data and make predictions or decisions without being explicitly programmed for the given task. The fundamental idea behind machine learning is to allow systems to automatically improve their performance over time as they are exposed to more data.

There are two main types of tasks in machine learning: classification and regression.

Classification*:* the goal is to assign input data points to predefined categories or classes. The output variable is discrete and represents a label or category. The algorithm learns from a labeled training dataset, where each input is associated with the correct output class. The trained model is then used to predict the class labels of new, unseen data.

Regression: the goal is to predict continuous numerical values based on input data. The output variable is a real number, and the algorithm learns to establish a relationship between the input features and the continuous output variable. Regression is used when the target variable is quantitative and not categorical.

Linear Regression: A common statistical technique is linear regression, which is used to describe the relationship between one or more independent variables (also called predictors or features) and a dependent variable (sometimes called the target or outcome variable). It is predicated on the idea that the independent and dependent variables have a linear relationship.

Finding the best-fitting line-or hyperplane, in the case of numerous independent variables-that minimizes the difference between the dependent variable's actual and anticipated values is the aim of linear regression. An intercept and a slope, sometimes referred to as a weight or coefficient, define the line.

- A basic linear regression model is represented by the equation **Yi = b1xi + b0 (y = mx + c)**, where:
- $\mathbf{Y_i}$is the dependent variable (the one being predicted).
- $\mathbf{x_i}$is the independent variable (the predictor).
- b_1 is the Regression Slope or coefficient, representing the change in Y for a one-unit change in x.
- $\mathbf{b_0}$ is the intercept, representing the value of Y when x is 0.

This equation is to fit a straight line to a set of data points in a way that minimizes the differences between the actual Y values and the values predicted by the equation for corresponding x values.

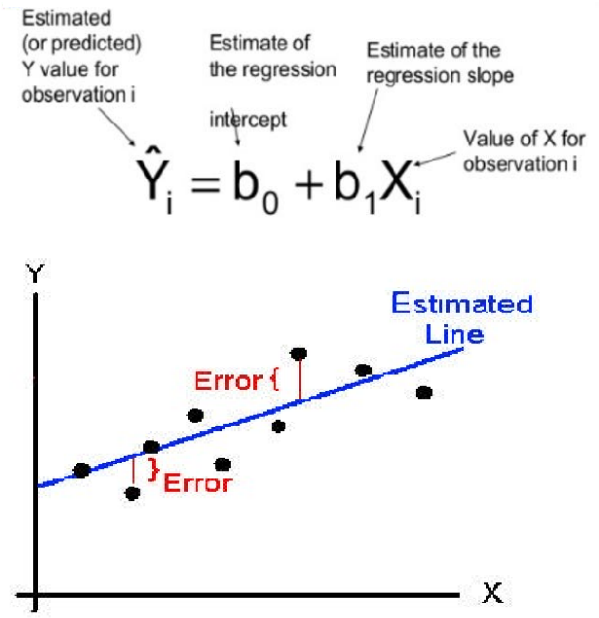

Figure 5 : Linear Regression

Decision Tree Regressor: It is a machine learning algorithm used for regression tasks. It's a type of decision tree model that is designed to predict continuous numeric values as opposed to categorical labels. Decision trees, in general, are a type of supervised learning algorithm that makes decisions based on asking a series of questions about the input features and eventually arriving at a prediction.

Here's how a Decision Tree Regressor works:

Tree Structure: The algorithm constructs a tree-like structure where each node represents a decision based on a specific feature and a threshold value. The tree structure consists of nodes and branches. The top node is called the root node, and the nodes that follow are internal nodes or leaf nodes.

Splitting Criteria: At each internal node, the algorithm selects the feature and threshold that best splits the data into subsets. The objective is to minimize the variance of the target values within each subset.

Leaf Nodes: As the tree grows, the algorithm continues to split the data based on the selected features and thresholds. Eventually, it stops when a predefined stopping criterion is met, such as reaching a maximum depth, having a minimum number of samples at a node, or not achieving a significant reduction in variance through splitting.

Predictions: When a new instance is fed into the trained Decision Tree Regressor, it traverses the tree from the root node to a leaf node by following the decision rules at each internal node. The predicted value for the new instance is the average of the target values in the training data that belong to the leaf node.

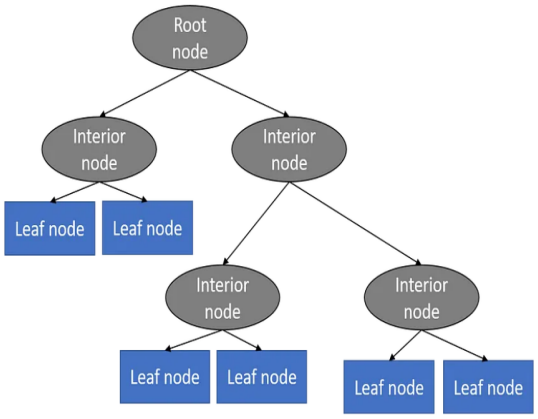

Figure 6 : Decision Tree Regressor

Random Forest Regressor: *It*is an ensemble machine learning algorithm that combines multiple Decision Tree Regressors to improve predictive performance and reduce overfitting. It's a popular method for regression tasks where the goal is to predict continuous numeric values. The algorithm creates a "forest" of decision trees and aggregates their predictions to provide a more accurate and robust result. Here's how the Random Forest Regressor works:

Bootstrapped Sampling: The algorithm starts by randomly selecting subsets of the original training data (with replacement). These subsets are called "bootstrap samples." Each subset is used to train a separate Decision Tree Regressor.

Random Feature Selection: For each decision tree, at each node's split, the algorithm selects a random subset of features from the available features. This randomness helps in creating diverse trees and reducing the correlation among them.

Building Trees: Multiple decision trees are grown based on the bootstrapped samples and random feature selections. Each tree is built until a certain stopping criterion is met, such as reaching a maximum depth or not having enough samples to split further.

Aggregation: When making predictions, each tree in the forest independently predicts the target value for a given input. The final prediction is then obtained by aggregating the individual predictions. For regression tasks, the most common aggregation method is taking the average of the predictions from all the trees.

Random Forest Regressors offer several benefits:

Reduced Overfitting: By combining multiple decision trees and aggregating their predictions, Random Forests tend to reduce overfitting compared to a single Decision Tree Regressor.

Better Generalization: The randomness introduced during both sampling and feature selection helps to create a diverse set of trees, leading to better generalization on unseen data.

Feature Importance: Random Forests can provide insights into feature importance. By evaluating how much each feature contributes to the overall prediction across all trees, you can identify which features have the most significant impact on the target variable.

Robustness: Random Forests are less sensitive to outliers and noisy data compared to individual decision trees.

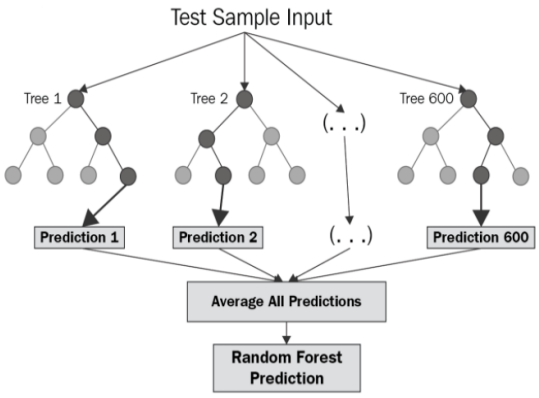

Figure 7 : Random Forest Regressor

XGBRegressor: It is a part of the XGBoost library, and it specifically refers to the XGBoost algorithm designed for regression tasks. XGBoost stands for eXtreme Gradient Boosting, and it's an ensemble learning method that has gained popularity for its performance and efficiency in predictive modeling.

Here are some key points about the `XGBRegressor`:

1. ***Regression Task:*** The primary purpose of `XGBRegressor` is to solve regression problems, where the goal is to predict a continuous numeric outcome. For example, predicting house prices or stock prices.
2. ***Ensemble Learning:*** XGBoost is an ensemble learning algorithm, meaning it builds a model by combining the predictions of multiple weaker models. In this case, the weaker models are decision trees.
3. ***Gradient Boosting:*** It uses a technique called gradient boosting, where each new decision tree corrects the errors of the previous ones. This iterative process results in a strong predictive model.
4. ***Regularization:*** XGBoost incorporates L1 (LASSO) and L2 (ridge) regularization terms in its objective function. This helps prevent overfitting and enhances the model's generalization to new, unseen data.
5. ***Tree Pruning:*** During the construction of decision trees, XGBoost uses a process called tree pruning to remove parts of the trees that do not contribute significantly to improving predictions. This enhances model efficiency.
6. ***Handling Missing Data:*** XGBoost has built-in mechanisms to handle missing data, allowing it to effectively deal with datasets that have incomplete information.
7. ***Parallel and Distributed Computing:*** XGBoost is designed to be computationally efficient and supports parallel and distributed computing. This makes it scalable and capable of handling large datasets.
8. ***Customizable:*** Users can customize various hyperparameters of the `XGBRegressor` to adapt it to different types of regression problems and data characteristics.
9. ***Feature Importance:*** XGBoost provides a feature importance score, helping users understand which features are most influential in making predictions.
10. ***Early Stopping:*** The algorithm supports early stopping, allowing the training process to stop when the model's performance on a validation set ceases to improve.

The `XGBRegressor` is a robust and versatile algorithm for regression tasks, known for its ability to handle complex relationships in data, prevent overfitting, and provide accurate predictions.

Mean *Absolute Error (MAE):* MAE is calculated as the average of the absolute differences between predicted and actual values. Unlike MSE, it treats all errors with equal weight, regardless of their magnitude.

MAE is more robust to outliers because it does not square the errors. It gives a more balanced representation of the overall error, which can be beneficial when dealing with datasets that contain significant outliers.

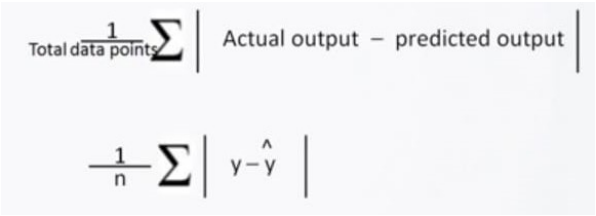

Figure 8: Mean Absolute Error

Mean Squared Error (MSE): Mean Squared Error (MSE) and Mean Absolute Error (MAE) are two commonly used metrics to evaluate the performance of regression models. Both metrics measure the difference between the predicted values and the actual (ground truth) values of the dependent variable. The main difference between them lies in how they treat the errors and their sensitivity to outliers.

MSE is calculated as the average of the squared differences between predicted and actual values. It gives higher weight to larger errors due to the squaring operation.

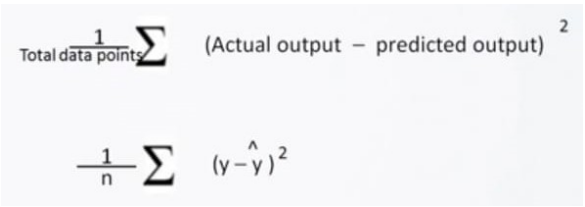

Figure 9: Mean Squared Error

Root Mean Square Error (RMSE): RMSE stands for Root Mean Squared Error. It is a commonly used metric for evaluating the performance of regression models, especially in cases where the dependent variable (target variable) has different units or scales.RMSE is calculated as the square root of the Mean Squared Error (MSE).

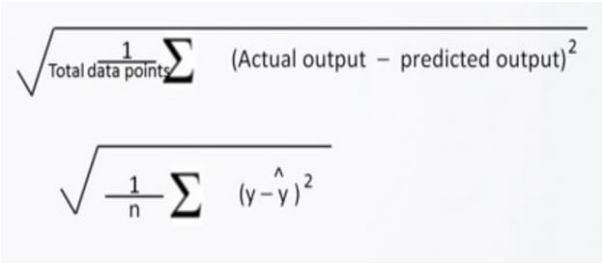

Figure 10: Root Mean Square Error

R-Squared: Mean Squared Error (MSE) and R-squared (R^2) are two commonly used metrics to evaluate the performance of regression models. They both assess how well the predicted values of the model match the actual target values, but they represent different aspects of model performance.

R-squared is a statistical measure that represents the proportion of the variance in the dependent variable (target) that is predictable from the independent variables (features) in the model. It measures the goodness of fit of the model to the data.

$$R^2 = 1 - (SSR / SST)$$

- SSR (Sum of Squared Residuals) is the sum of the squared differences between the predicted values and the mean of the actual target values.
- SST (Sum of SquaresTotal) is the sum of the squared differences between the actual target values and their mean.

R^2 values range from 0 to 1, where 0 indicates that the model explains none of the variance in the data (it performs no better than predicting the mean), and 1 indicates a perfect fit where the model explains all the variance in the data.

A higher R^2 value indicates a better-performing model, as it suggests that a larger proportion of the variance in the target variable is explained by the model's predictions.

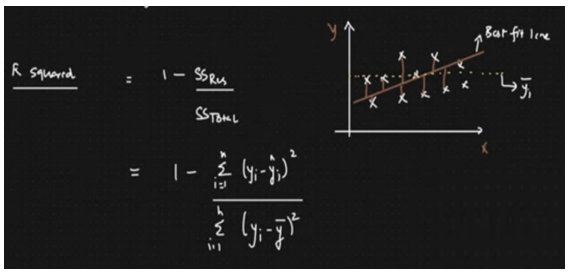

Figure 11 : R-Squared Error

1. ***RMSE (Root Mean Square Error):*** You want RMSE to be low, indicating that the model's predictions are close to the actual values. A lower RMSE indicates better accuracy.

2. ***R-squared (Coefficient of Determination):*** You want R-squared to be high, closer to 1.0, indicating that the model explains a large portion of the variability in the data. A higher R-squared indicates better accuracy in explaining the variation in the dependent variable.

RMSE and R-squared are not used to directly calculate model accuracy percentage. Instead, they are used to assess the model's performance and understand how well it fits the data. They provide information about the model's predictive power and goodness of fit but do not give a percentage accuracy.

Both MSE and R^2 are essential metrics for evaluating regression models, and they complement each other. While MSE directly measures the accuracy of the predictions, R^2 provides an overall assessment of how well the model fits the data. It is often recommended to use both metrics together to get a comprehensive understanding of the model's performance.

Train Test Splitting: Train-test splitting is straightforward and easy to implement. It's a good starting point for model evaluation. It's computationally less intensive than cross-validation because you're training the model once on the training set.

1. ***Variability:*** The performance of the model can vary significantly depending on how the data is split into training and testing sets. You might get different results with different random splits.

2. ***Overfitting Risk:*** If you perform multiple experiments with different random splits, you may inadvertently overfit the model to a particular split that happens to work well for that specific randomization.

3. ***Limited Data:*** When you split the data into two parts, you have less data available for training, which can be a limitation, especially if your dataset is small.

Cross-Validation: Robustness: Cross-validation provides a more robust estimate of a model's performance by repeatedly training and testing on different subsets of the data. It helps reduce the impact of data variability.

Better Generalization: By using multiple splits, cross-validation helps assess how well the model generalizes to new, unseen data. It gives you a more realistic view of how the model might perform in practice. Effective Use of Data: Cross-validation makes more efficient use of the available data.

It ensures that all data points are used for both training and testing, increasing the amount of information the model can learn from. Hyperparameter Tuning: Cross-validation is often used for hyperparameter tuning. It allows you to test different hyperparameter settings and choose the ones that lead to the best model performance.

Bias-Variance Tradeoff: Cross-validation helps you understand the tradeoff between bias and variance in your model. It can reveal whether your model is underfitting or overfitting.

Confidence Intervals: Cross-validation results can be used to calculate confidence

intervals for performance metrics, giving you a sense of the model's reliability.

In summary, train-test splitting is a simple method for initial model evaluation, but it has limitations related to data variability and overfitting. Cross-validation, on the other hand, is a more comprehensive and robust technique for assessing a model's performance, making efficient use of the data, and tuning hyperparameters. It provides a better understanding of how well the model will generalize to new data and helps mitigate the issues associated with train-test splitting.

The **default form of cross-validation**, often used as a standard method in machine learning, is k-fold cross-**validation**. In k-fold cross-validation, the dataset is divided into k equally sized (or nearly equally sized) "folds" or subsets. The model is trained and tested k times, with each fold serving as the testing set once, and the remaining k-1 folds used as the training set in each iteration.

Here's how k-fold cross-validation works:

1. The dataset is randomly divided into k roughly equal parts (folds).
2. The model is trained on k-1 of these folds (the training set) in each iteration.
3. The model's performance is evaluated on the fold that was not used for training (the testing set) in that particular iteration.
4. This process is repeated k times, with each of the k folds serving as the testing set exactly once.
5. The performance metrics (e.g., accuracy, mean squared error) are typically averaged over the k iterations to obtain an overall estimate of the model's performance.

Common choices for the value of k include 5-fold and 10-fold cross-validation, but the choice may vary depending on the dataset size and specific requirements. Smaller values of k may be chosen if you have a large dataset, while larger values of k are preferred for smaller datasets.

K-fold cross-validation is a widely used technique for model evaluation because it provides a more reliable estimate of a model's performance compared to a single train-test split. It helps in assessing how well the model generalizes to different subsets of the data, reduces the risk of overfitting, and allows for a more robust evaluation of the model's performance.

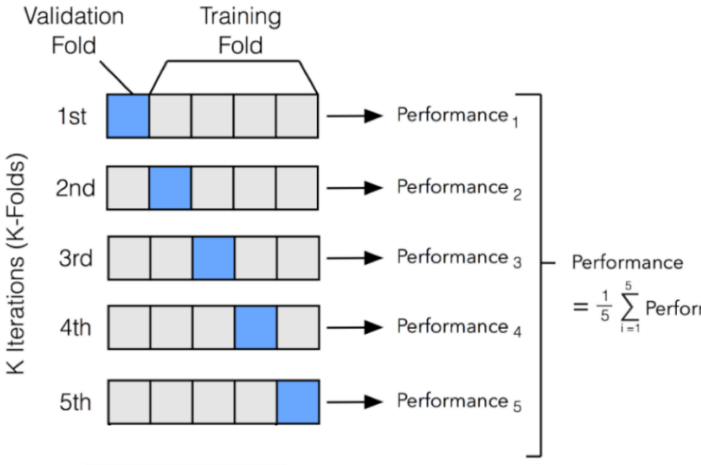

Figure 12 : k-fold Cross Validation

Task involves predicting a continuous numerical value (AQI) based on input features (SO_2, NO_2, RSPM, SPM). This is a regression problem, and the objective is to model the underlying relationship between the input features and the output variable (AQI) using machine learning algorithms.

The mathematical representation of a regression model could be something like:

$AQI = f(SO_2, NO_2, RSPM, SPM) + \epsilon$

where:

f is the **regression function** representing the relationship between input features and AQI.

ϵ is the **error term**, accounting for any noise or unexplained variability in the data.

The goal during the training phase is to learn the parameters of the regression function f) from the provided dataset so that the model can make accurate predictions for new input values.

"The escalating threat of air pollution demands accurate and efficient Air Quality Index (AQI) prediction models. Leveraging machine learning algorithms, this research aims to identify the most effective methodologies across diverse datasets, pollutants, and geographical locations. Robust AQI prediction models are crucial for informing policymakers and environmental agencies in mitigating the global impact of air pollution on public health and the environment."

Through a thorough comparative analysis of models, employing metrics such as Root Mean Squared Error (RMSE) and R-squared (R^2), the study seeks to identify the algorithm that attains the utmost precision and applicability in predicting the Air Quality Index (AQI). This endeavor will furnish individuals with a dependable tool for accurately estimating AQI, assisting them in making well-informed decisions to enhance air quality management strategies.

Table 1. Training Testing Results of Different Models

Model Used	Training Data:	Testing Data:
Decision Tree Regression (DTR):	R2 = 0.9473 MAE = 3.8971 MSE = 27.4344 RMSE = 5.2378	R2 = 0.9404 MAE = 3.9298 MSE = 28.9519 RMSE = 5.3807
Random Forest Regression (RFR):	R2 = 0.9998 MAE = 0.0112 MSE = 0.0602 RMSE = 0.2453	R2 = 0.9981 MAE = 0.0386 MSE = 0.9201 RMSE = 0.9592
Linear Regression (LR):	R2 = 0.8684 MAE = 6.0460 MSE = 68.5194 RMSE = 8.2776	R2 = 0.8562 MAE = 6.0805 MSE = 69.8326 RMSE = 8.3566
Support Vector Regression with Radial Basis Function kernel (SVR-RBF):	R2 = 0.9948 MAE = 0.7012 MSE = 2.6567 RMSE = 1.6299	R2 = 0.9940 MAE = 0.6958 MSE = 2.9043 RMSE = 1.7042
Support Vector Regression with Polynomial Kernel (SVR-P):	R2 = 0.6723 MAE = 6.7512 MSE = 170.5909 RMSE = 13.0610	R2 = 0.7160 MAE = 6.5073 MSE = 131.0610 RMSE = 11.7455

II. Background and Related Work

Parameshachari B Det al. [1], The main focus of the research is on addressing the critical issue of pollution, particularly air pollution, which poses a significant threat to human health worldwide. The U.S. Pollution dataset used in this research is from Kaggle. The dataset specifically deals with pollution in the U.S. and is stated to be managed and verified by the U.S. EPA. The dataset covers pollution data for four major pollutants: NO_2, SO_2, CO, and O3.The data spans the years 2000-2016, providing a temporal perspective on pollution levels. Data preprocessing is performed by dropping unwanted fields and observations. The dataset is adapted for further analysis, ensuring that machine learning algorithms can work effectively with the data. The research involves the application of various machine learning algorithms for predicting the Air Quality Index (AQI). The algorithms used include Decision Tree Regression (DTR), Linear Regression (LR), Random Forest Regression (RFR), Support Vector Regression with Radial Basis Function kernel (SVR-RBF), and Support Vector Regression with Polynomial Kernel (SVR-P).

The Random Forest Regression (RFR) model appears to have the best performance on both the training and testing datasets, as it achieves the highest R2 and the lowest MAE, MSE, and RMSE values among the models compared.

K.M.O.V.K. Kekulanadaraet al. [2],This study highlights the pressing issue of air pollution exacerbated by urbanization and industrialization. Utilizing various machine learning algorithms—Decision Tree, Random Forest, Support Vector Machine (SVM), and Artificial Neural Network (ANN)—the research conducts a comparative analysis to predict the Air Quality Index (AQI) based on major pollutants. The results affirm the effectiveness of machine-learning algorithms in predicting

AQI, with Random Forest and Neural Network demonstrating notable accuracy levels of 74.039% and 73.82%, respectively. This research provides valuable insights for global efforts in assessing and managing air quality, offering practical tools for policymakers and environmental agencies. Continued refinement and application of machine learning techniques hold the potential to significantly contribute to mitigating the adverse impacts of air pollution on human health and the environment.

The specific objective of this research is to predict the AQI by considering major pollutants in the air. Decision Tree, Random Forest, SVM, and ANN algorithms were used to create prediction models. The experiments were carried out using a Windows 10 installed computer with an Intel Core i7-7500U, 2.70 CPU, and 8GB RAM. Weka software, Anaconda Navigator GUI, Jupyter Notebook IDE, and python were used for implementations. The proposed method consists of a five-step process. They are 1) data collection, 2) data preprocessing, 3) feature selection, 4) training models 5) evaluation.

The dataset used in this research, sourced from Kaggle, covers hourly and daily Air Quality Index (AQI) and air quality levels across diverse Indian cities from January 15, 2015, to July 1, 2020. With 16 attributes and 100,000 instances, it includes station ID, date, time, and concentrations of pollutants like PM2.5, PM10, NO, NO2, NOx, NH3, SO2, O3, CO, Benzene, Toluene, Xylene, and AQI. Emphasizing particulate matter (PM) significance, particularly PM2.5 and PM10, the dataset is vital for predicting air quality.

Measurements are in µg/m3, except for NOx (ppb) and CO (mg/m3).

Table 2. Comparison of accuracy level among concerned algorithms

	Decision Tree	SVM	Random Forest	Neural Network
Accuracy	64.943	61.345	74.039	73.82
RMSE	0.314	0.328	0.248	0.57

Karlapudi Saikiran et **al. [3],** This research endeavors to predict the Air Quality Index (AQI) using various machine learning algorithms to address the critical issue of pollution and its impact on public health. The study focuses on pollutants such as particulate matters, nitrous oxide (NO2), sulfur dioxide (SO2), and carbon monoxide (CO) and employs machine learning algorithms, including Multiple Linear Regression (MLR), Support Vector Regression (SVR), and Random Forest Regression (RFR), for accurate predictions.The specific Data source of the air pollution data is not explicitly mentioned. However, the data is collected from various sensors, and the study emphasizes the utilization of the Internet of Things (IoT) and machine learning algorithms for more precise monitoring and prediction.

The machine learning algorithms employed in the study are Multiple Linear Regression (MLR), Support Vector Regression (SVR), and Random Forest Regression (RFR). Methodology of the research is:

1. Data Pre-processing: The air pollution dataset undergoes pre-processing, including normalization, attributes selection, and discretization, to enhance its quality for machine learning model training.

2. Dataset Splitting: The dataset is divided into training and testing sets, with 70% allocated for training and 30% for testing.

3. Supervised Machine Learning Algorithms:

- ***Multiple Linear Regression (MLR):*** Models the relationship between a dependent continuous variable and multiple independent variables.
- ***Support Vector Regression (SVR):*** Utilizes support vector machine algorithms for regression problems, focusing on determining the maximum number of data points within boundary lines.
- ***Random Forest Regression (RFR):*** An ensemble learning technique combining multiple decision trees to predict pollution parameters.

4. ***Model Evaluation:*** The accuracy of the models is measured using the Root Mean Square Error (RMSE) technique.RMSE values for different algorithms are reported as follows:

- MLR: 1.926
- SVR: 1.231
- RFR: 0.812

The results indicate that the Random Forest Regression algorithm outperforms the other techniques, exhibiting higher accuracy and reduced overfitting. The study concludes that Random Forest Regression is the most effective model for predicting future Air Quality Index values, making it a valuable tool for managing and mitigating air pollution.

III. Proposed Work

India Air Quality Index DataSource(1990 - 2015)

Data Source- https://www.kaggle.com/datasets/shrutibhargava94/india-air-quality-dataFiles Data Set Size : 62.54 MB (1 File data.csv / 13 Columns)

Content

1. stn_code - Station Code
2. sampling_date - Date of Sample Collection
3. state - Indian state Name
4. location - Location of Sample Collection
5. agency - Agency
6. type - Type of Area
7. so2 - Sulphur Dioxide Concentration
8. no2 - Nitrogen Dioxide Concentration
9. rspm - Respirable Suspended Particualte Matter Concentration
10. spm - Suspended Particulate Matter
11. location_monitoring_station
12. pm2_5 - Particulate matter 2.5
13. date

where input values for SO2, NO2, RSPM, and SPM to predict the Air Quality Index (AQI), falls under the category of regression. In regression tasks, the goal is to predict a continuous numerical value based on input features.

	state	location	type	so2	no2	rspm	spm	pm2_5	SO2_SubIndex	NO2_SubIndex	rspm_SubIndex	spm_SubIndex	AQI	AQI_Range
0	Andhra Pradesh	Hyderabad	Residential, Rural and other Areas	4.8	17.4	0.0	0.0	0.0	6.000	21.750	0.0	0.0	21.750	Good
1	Andhra Pradesh	Hyderabad	Industrial Area	3.1	7.0	0.0	0.0	0.0	3.875	8.750	0.0	0.0	8.750	Good
2	Andhra Pradesh	Hyderabad	Residential, Rural and other Areas	6.2	28.5	0.0	0.0	0.0	7.750	35.625	0.0	0.0	35.625	Good
3	Andhra Pradesh	Hyderabad	Residential, Rural and other Areas	6.3	14.7	0.0	0.0	0.0	7.875	18.375	0.0	0.0	18.375	Good
4	Andhra Pradesh	Hyderabad	Industrial Area	4.7	7.5	0.0	0.0	0.0	5.875	9.375	0.0	0.0	9.375	Good

Figure 13 : AQI: Target variable (Continuous Values)

Regression Problem = As All Values of AQI are Numerical and Continuous. Thus, it is Not a Classification Problem.

Proposed-Framework

- Implementation IDE - Google Collaborator
- Python–Python 3
- Applied Regression Problemon Data
- Importing Liabraries

Reading Dataset:

Reading CSV file (data.csv) using pandas into data frame df and EDA.

Data Preprocessing

- Dropping Unnecessary Columns
- Check forNull Values and Repacing them with MODE
- Calculating AQI SubIndex for Pollutants (SO_2, NO_2, RSPM, SPM) and Setting AQI Range
- Feature Target Splitting
- Train Test Splitting (80% - 20%)

Model Creation:

Four Regression Models are created and evaluated using Root Mean Squared Error (MSE) and R-squared as evaluation metrics.

Applied Cross Validation & Calculating Model's Mean Accuracy (CV=5)

Model Evaluation is Done by Giving Feature Values and Predicting Target Values

Linear Regression Model

Input: x_train: Training features, y_train: Training target variable, x_test: Testing features

y_test: Testing target variable

1. **Instantiate a Linear Regression:**
   - $\mathrm{LR} = \mathrm{LinearRegression}()$
2. **Train the Linear Regression:**
   - $\mathrm{LR.fit}(x_{\mathrm{train}}, y_{\mathrm{train}})$
3. **Predict on Training Data & Testing Data:**
   - Training Data:
     - $y_{\mathrm{train\_pred}} = \mathrm{LR.predict}(x_{\mathrm{train}})$
   - Testing Data:
     - $y_{\mathrm{test\_pred}} = \mathrm{LR.predict}(x_{\mathrm{test}})$
4. **Calculate Root Mean Squared Error (RMSE) for Training Data & Testing Data:**
   - Training Data:
     - $\mathrm{RMSE}_{\mathrm{train}}^{\mathrm{LR}} = \sqrt{\mathrm{mean\_squared\_error}(y_{\mathrm{train}}, y_{\mathrm{train\_pred}})}$
   - Testing Data:
     - $\mathrm{RMSE}_{\mathrm{test}}^{\mathrm{LR}} = \sqrt{\mathrm{mean\_squared\_error}(y_{\mathrm{test}}, y_{\mathrm{test\_pred}})}$
5. **Calculate R-squared (RSquared) for Training Data & Testing Data:**
   - Training Data:
     - $\mathrm{score}_{\mathrm{train}}^{\mathrm{LR}} = \mathrm{LR.score}(x_{\mathrm{train}}, y_{\mathrm{train}})$
   - Testing Data:
     - $\mathrm{score}_{\mathrm{test}}^{\mathrm{LR}} = \mathrm{LR.score}(x_{\mathrm{test}}, y_{\mathrm{test}})$

Figure 14 : Algorithm: Linear Regression for Air Quality Prediction

Decision Tree Regressor Model

1. **Instantiate a Decision Tree Regressor:**
   - $\mathrm{DT} = \mathrm{DecisionTreeRegressor}()$
2. **Train the Decision Tree Regressor:**
   - $\mathrm{DT.fit}(x_{\mathrm{train}}, y_{\mathrm{train}})$
3. **Predict on Training Data & Testing Data:**
   - Training Data:
     - $y_{\mathrm{train\_pred}} = \mathrm{DT.predict}(x_{\mathrm{train}})$
   - Testing Data:
     - $y_{\mathrm{test\_pred}} = \mathrm{DT.predict}(x_{\mathrm{test}})$
4. **Calculate Root Mean Squared Error (RMSE) for Training Data & Testing Data:**
   - Training Data:
     - $\mathrm{RMSE}_{\mathrm{train}}^{\mathrm{DT}} = \sqrt{\mathrm{mean\_squared\_error}(y_{\mathrm{train}}, y_{\mathrm{train\_pred}})}$
   - Testing Data:
     - $\mathrm{RMSE}_{\mathrm{test}}^{\mathrm{DT}} = \sqrt{\mathrm{mean\_squared\_error}(y_{\mathrm{test}}, y_{\mathrm{test\_pred}})}$
5. **Calculate R-squared (RSquared) for Training Data & Testing Data:**
   - Training Data:
     - $\mathrm{score}_{\mathrm{train}}^{\mathrm{DT}} = \mathrm{DT.score}(x_{\mathrm{train}}, y_{\mathrm{train}})$
   - Testing Data:
     - $\mathrm{score}_{\mathrm{test}}^{\mathrm{DT}} = \mathrm{DT.score}(x_{\mathrm{test}}, y_{\mathrm{test}})$

Figure 15 : Algorithm: Decision Tree Regressor for Air Quality Prediction

RandomForestRegressor Model

1. **Instantiate RandomForestRegressor:**
   - $\mathrm{RF} = \mathrm{RandomForestRegressor}()$
2. **Train the RandomForestRegressor:**
   - $\mathrm{RF.fit}(x_{\mathrm{train}}, y_{\mathrm{train}})$
3. **Predict on Training and Testing Data:**
   - Training Data:
     - $y_{\mathrm{train\_pred}} = \mathrm{RF.predict}(x_{\mathrm{train}})$
   - Testing Data:
     - $y_{\mathrm{test\_pred}} = \mathrm{RF.predict}(x_{\mathrm{test}})$
4. **Calculate Root Mean Squared Error (RMSE):**
   - Training Data:
     - $\mathrm{RMSE}_{\mathrm{train}}^{\mathrm{RF}} = \sqrt{\mathrm{mean\_squared\_error}(y_{\mathrm{train}}, y_{\mathrm{train\_pred}})}$
   - Testing Data:
     - $\mathrm{RMSE}_{\mathrm{test}}^{\mathrm{RF}} = \sqrt{\mathrm{mean\_squared\_error}(y_{\mathrm{test}}, y_{\mathrm{test\_pred}})}$
5. **Calculate R-squared (RSquared):**
   - Training Data:
     - $\mathrm{score}_{\mathrm{train}}^{\mathrm{RF}} = \mathrm{RF.score}(x_{\mathrm{train}}, y_{\mathrm{train}})$
   - Testing Data:
     - $\mathrm{score}_{\mathrm{test}}^{\mathrm{RF}} = \mathrm{RF.score}(x_{\mathrm{test}}, y_{\mathrm{test}})$

Figure 16 : Algorithm: RandomForestRegressor for Air Quality Prediction

XGBoost Regressor Model

Instantiate a XGB Regressor for Air Quality Prediction

```
XGBR = xgb.XGBRegressor(objective='reg:squarederror', random_state=42)
```

2. **Train the XGB Regressor:**
   - $\mathrm{XGBR.fit}(x_{\mathrm{train}}, y_{\mathrm{train}})$
3. **Predict on Training and Testing Data:**
   - Training Data:
     - $y_{\mathrm{train\_pred}} = \mathrm{XGBR.predict}(x_{\mathrm{train}})$
   - Testing Data:
     - $y_{\mathrm{test\_pred}} = \mathrm{XGBR.predict}(x_{\mathrm{test}})$
4. **Calculate Root Mean Squared Error (RMSE):**
   - Training Data:
     - $\mathrm{RMSE}_{\mathrm{train}}^{\mathrm{XGBR}} = \sqrt{\mathrm{mean\_squared\_error}(y_{\mathrm{train}}, y_{\mathrm{train\_pred}})}$
   - Testing Data:
     - $\mathrm{RMSE}_{\mathrm{test}}^{\mathrm{XGBR}} = \sqrt{\mathrm{mean\_squared\_error}(y_{\mathrm{test}}, y_{\mathrm{test\_pred}})}$
5. **Calculate R-squared (RSquared):**
   - Training Data:
     - $\mathrm{score}_{\mathrm{train}}^{\mathrm{XGBR}} = \mathrm{XGBR.score}(x_{\mathrm{train}}, y_{\mathrm{train}})$
   - Testing Data:
     - $\mathrm{score}_{\mathrm{test}}^{\mathrm{XGBR}} = \mathrm{XGBR.score}(x_{\mathrm{test}}, y_{\mathrm{test}})$

Figure 17 : Algorithm: eXtreme Gradient Boosting Regressor for Air Quality Prediction

These algorithms describe the steps involved in training the models for predicting air quality, evaluating its performance using RMSE and R-squared, and printing the results.

- D be the dataset.
- $\text{Metric}(D)$ represent the evaluation metric applied to dataset D.
- $\text{Split}(D, k)$ represent the function that splits dataset D into k equal-sized subsets.
- $\text{Train}(D_i)$ represent the function that trains the model on dataset D_i.
- $\text{Validate}(D_j)$ represent the function that evaluates the model on dataset D_j.

The k-fold cross-validation can be described as follows:

1. **Split the Data:**
   - $D_1, D_2, ..., D_k = \text{Split}(D, k)$
2. **Iterate through Folds:**
   - For $i = 1$ to k:
     - $\text{Model}_i = \text{Train}(D_1 \cup D_2 \cup ... \cup D_{i-1} \cup D_{i+1} \cup ... \cup D_k)$
     - $\text{Score}_i = \text{Metric}(\text{Validate}(D_i))$
3. **Calculate the Mean Score:**
   - $\text{MeanScore} = \frac{1}{k} \sum_{i=1}^{k} \text{Score}_i$

Figure 18 : Algorithm for k-Fold Cross Validation

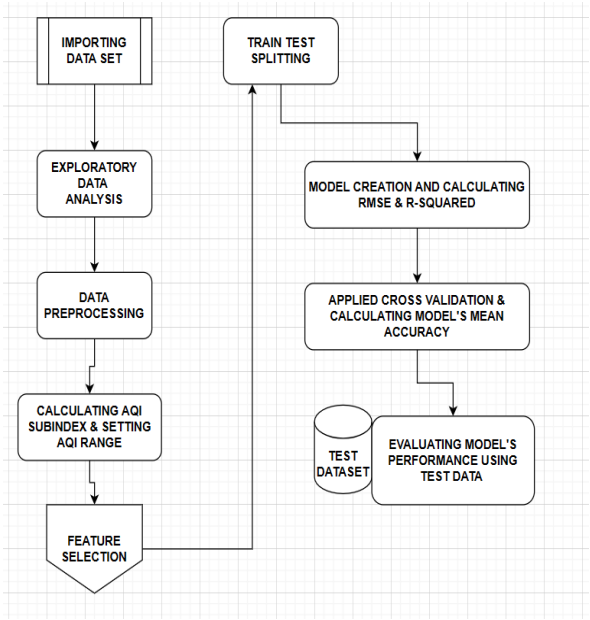

Figure 19 : Proposed Work Flow

IV. Results and Comparision Analysis

Root Mean Squared Error and R-Squared Results:

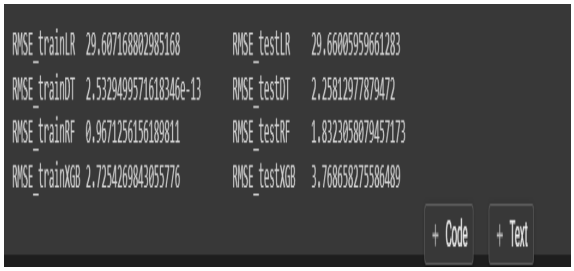

Figure 20 : RMSE Summary of Different Models

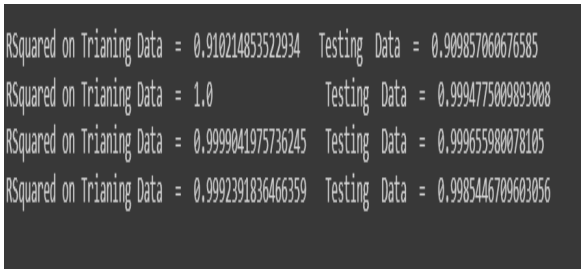

Figure 21: R-Squared Summary of Different Models

Cross Validation Results:

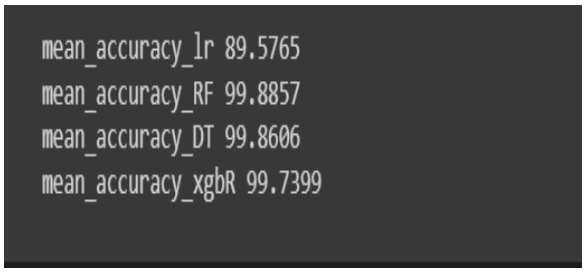

Figure 22 : Mean Accuracy Summary of Different Models

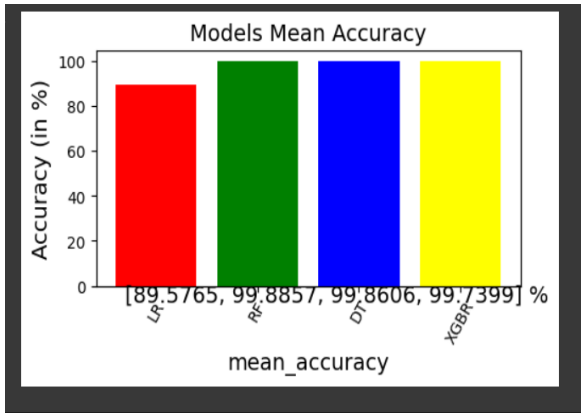

Figure 23 : Mean Accuracy Visualization of Different Models

Models Evaluation:

```
Enter the values for -  so2     no2     rspm    spm
so2 7.750
no2 35.625
rspm 0.0
spm 0.0
LR Predicted AQI: 24.90442028973592
AQI Status Good
RF Predicted AQI: 35.624375
AQI Status Good
DT Predicted AQI: 35.625
AQI Status Good
XGB Predicted AQI: 33.863834
AQI Status Good
```

Figure 24 : AQI Values and Range as per Given Pollutants of different Models

```
Cross-validation scores for Linear Regression: [0.89688004 0.87795633 0.89313751 0.89333724 0.91751261]
Enter the values for - so2 no2 rspm spm
so2 7.750
no2 35.625
rspm 0.0
spm 0.0
LR Predicted AQI: 24.886025486061204
AQI Status: Good
```

Figure 25 : CV scores for Linear Regression

```
Cross-validation scores for Decision Tree Regressor: [0.99980184 0.99928583 0.9979996  0.99980496 0.99626468]
Enter the values for - so2 no2 rspm spm
so2 7.750
no2 35.625
rspm 0.0
spm 0.0
Decision Tree Predicted AQI: 35.625
AQI Status: Good
```

Figure 26 : CV scores for Decision Tree Regressor

```
Cross-validation scores for Random Forest Regressor: [0.99986366 0.99981872 0.99831777 0.99981139 0.99653575]
Enter the values for - so2 no2 rspm spm
so2 7.750
no2 35.625
rspm 0.0
spm 0.0
Random Forest Predicted AQI: 35.625
AQI Status: Good
```

Figure 27 : CV scores for Random Forest Regressor

```
Cross-validation scores for XGBoost Regression: [0.99795345 0.99887457 0.99682827 0.99790659 0.99543225]
Enter the values for - so2 no2 rspm spm
so2 7.750
no2 35.625
rspm 0.0
spm 0.0
XGBoost Predicted AQI: 35.242325
AQI Status: Good
```

Figure 28 : CV scores for XGBoost Regression

```
Cross-validation scores for Random Forest Regressor: [0.99987494 0.99982264 0.99829808 0.99980998 0.99651009]
RMSE: 3.681621343341035
R-squared: 0.9986115689626822
Enter the values for - so2 no2 rspm spm
so2 7.750
no2 35.625
rspm 0.0
spm 0.0
Random Forest Predicted AQI: 35.625625
AQI Status: Good
```

Figure 29 : Model AQI Values and Range Calculated

Existing Work- Results:

Existing Work (Base Paper)	Parameshachari B D, Siddesh G M, V. Sridhar, Latha M, Khalid Nazim Abdul Sattar, Manjula. G, "Prediction and Analysis of Air Quality Index using Machine Learning Algorithms", 2022 IEEE International Conference on Data Science and Information System (ICDSIS) | 978-1-6654-9801-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICDSIS55133.2022.9915802.
Data Source	The dataset specifically deals with pollution in the U.S. and is stated to be managed and verified by the U.S. EPA. The dataset covers pollution data for four major pollutants: NO2, SO2, CO, and O3. The data spans the years 2000-2016, providing a temporal perspective on pollution levels.

Train Test Splitting – Training Dataset 80%, Testing Dataset 20%

Table 3. Existing Work (Base Paper) Summary

Machine Learning Model Used	Training Data:	Testing Data:
Decision Tree Regression (DTR):	R2 = 0.9473 MAE = 3.8971 MSE = 27.4344 RMSE = 5.2378	R2 = 0.9404 MAE = 3.9298 MSE = 28.9519 RMSE = 5.3807
Random Forest Regression (RFR):	R2 = 0.9998 MAE = 0.0112 MSE = 0.0602 RMSE = 0.2453	R2 = 0.9981 MAE = 0.0386 MSE = 0.9201 RMSE = 0.9592
Linear Regression (LR):	R2 = 0.8684 MAE = 6.0460 MSE = 68.5194 RMSE = 8.2776	R2 = 0.8562 MAE = 6.0805 MSE = 69.8326 RMSE = 8.3566
Support Vector Regression with Radial Basis Function kernel (SVR-RBF):	R2 = 0.9948 MAE = 0.7012 MSE = 2.6567 RMSE = 1.6299	R2 = 0.9940 MAE = 0.6958 MSE = 2.9043 RMSE = 1.7042
Support Vector Regression with Polynomial Kernel (SVR-P):	R2 = 0.6723 MAE = 6.7512 MSE = 170.5909 RMSE = 13.0610	R2 = 0.7160 MAE = 6.5073 MSE = 131.0610 RMSE = 11.7455

Table 4. Existing Work (Supporting Paper) Summary

Existing Work (Supporting Paper)	K.M.O.V.K. Kekulanadara, B.T.G.S Kumara, Banujan Kuhaneswaran, "Machine Learning Approach for Predicting Air Quality Index", 2021 International Conference on Decision Aid Sciences and Application (DASA) | 978-1-6654-1634-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/DASA53625.2021.9682221
Data Source	The dataset that is used in this research was taken from Kaggle online dataset library. Data is collected from 15 January 2015 to 1 July 2020. It is contained hourly and daily levels of AQI and air quality at different stations across various cities in India.

Table 5. Summary Details of The Dataset

	Before	After
# of Instance	100,000	61,285
Dataset Features	Station_ID, DateTime, $PM_{2.5}$, PM_{10}, NO, NO_2, NO_x, NH_3, SO_2, O_3, CO, Benzene, Toluene, Xylene, AQI, AQI_Bucket	$PM_{2.5}$, PM_{10}, NO, NO_2, NO_x, NH_3, SO_2, O_3, CO, Benzene, Toluene, Xylene, AQI_Bucket
Number of Features	16	13

Train Test Splitting – Training Dataset 80%, Testing Dataset 20%

Table 6. Comparison of Accuracy Level Among Concerned Algorithms

	Decision Tree	SVM	Random Forest	Neural Network
Accuracy	64.943	61.345	74.039	73.82
RMSE	0.314	0.328	0.248	0.57

Proposed Work Results:

DATASET USED in Proposed Work: India Air Quality Data (1990 - 2015)

Data Set Size : 62.54 MB (1 File data.csv / 13 Columns)

https://www.kaggle.com/datasets/shrutibhargava94/india-air-quality-data

Proposed Work – "Analysing Indian Air Quality: Exploratory Data Analysis and Regression Approach"

Instances – 435742

Number of Features – Before 13, After Feature Selection 5

Table 7 : Results of Proposed Work

Machine Learning Models		RMSE		R^2		Mean Accuracy After Cross Validation
		Training	Testing	Training	Testing	CV = 5
1.	Linear Regressor	29.607168802985168	29.66005959661283	0.9102148535229 34	0.9098570606765 85	89.5765
2.	Decision Tree Regression	2.6168853993921783e-13	2.2653428032431377	1.0	0.9994741576762547	99.8668
3.	Random Forest Regressor	1.0827612195796907	1.8249620079475746	0.9998799184953494	0.9996587321848925	99.8864 RMSE: 3.681621343341035 R-squared: 0.9986115689626822
4.	XGB Regressor	2.7254269843055776	3.768658275586489	0.9992391836466359	0.9985446709603056	99.7399

Table 8 : Model Evaluation

Input Pollutants	Expected AQI	Regression Models	After Cross Validation -Predicted AQIValues / Range
SO2 7.750 NO2 35.625 RSPM 0.0 SPM 0.0	35.625 Good	Linear Regressor DecisionTree Regression RandomForest Regressor XGB Regressor	24.886025486061204Good 35.625Good 35.625 Good 35.242325 Good

V. Conclusion and Future Work

In conclusion, this research endeavors to tackle the critical task of predicting the Air Quality Index (AQI) in India, given the escalating threat of air pollution. The regression problem focuses on modeling the intricate relationship between key pollutants (SO_2, NO_2, RSPM, SPM) and AQI using machine learning algorithms. The primary objective is to develop robust AQI prediction models that can effectively inform policymakers and environmental agencies, aiding in the formulation of strategies to combat the global impact of air pollution on public health and the environment.

The proposed framework involves a meticulous process, including importing libraries, reading and exploring the dataset, data preprocessing, AQI subindex calculation, feature-target splitting, and the creation of four regression models (Linear Regressor, Decision Tree Regressor, Random Forest Regressor, XGB Regressor). The models are evaluated using Root Mean Squared Error (RMSE), R-squared (R^2), and mean accuracy after cross-validation.

Evaluation:

RMSE: The lower the RMSE, the better. In this regard, Decision Tree Regressor and

Random Forest Regressor outperform Linear Regressor and XGB Regressor.

R^2: The closer to 1, the better. All models exhibit high R^2 values, with Decision Tree Regressor and Random Forest Regressor leading in capturing the variance in the data.

Mean Accuracy (CV): The higher, the better. Random Forest Regressor achieves the highest mean accuracy, closely followed by Decision Tree Regressor.

The Decision Tree Regressor and Random Forest Regressor appear to be the most effective models for predicting the Air Quality Index (AQI) in this context. These models demonstrate superior performance in terms of both accuracy and precision compared to the Linear Regressor and XGB Regressor.

Future Work involves: Feature Engineering which Explores additional features or transformations to enhance predictive performance. Hyperparameter Tuning - Optimize model hyperparameters for improved accuracy. Ensemble Methods- Investigate combining predictions from multiple models for enhanced robustness. Geospatial and Temporal Analysis- Capture spatial and temporal variations in air quality.

External Factors: Incorporate external factors influencing air quality. Real-time Prediction- Develop real-time models for timely information. User Feedback- Gather user feedback for model refinement.

Comparison with Base Paper: Validate models against existing work for effectiveness. By addressing these aspects, the research aims to contribute accurate AQI prediction models for informed decision-making in India's air quality management.

REFERENCES:

[1] Parameshachari B D, Siddesh G M, V. Sridhar, Latha M, Khalid Nazim Abdul Sattar, Manjula.G, "Prediction and Analysis of Air Quality Index using Machine Learning Algorithms", 2022 IEEE International Conference on Data Science and Information System (ICDSIS) | 978-1-6654-9801-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICDSIS55133.2022.9915802.

[2] K.M.O.V.K. Kekulanadara, B.T.G.S Kumara, Banujan Kuhaneswaran, "Machine Learning Approach for Predicting Air Quality Index",2021 International Conference on Decision Aid Sciences and Application (DASA) | 978-1-6654-1634-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/DASA53625.2021.9682221.

[3] Karlapudi Saikiran, IEEE, Gottapu Lithesh, IEEE, Birru Srinivas, IEEE, Student Member, "Prediction of Air Quality Index Using Supervised Machine Learning Algorithms", 021 2nd International Conference on Advances in Computing, Communication, Embedded and Secure Systems (ACCESS) | 978-1-7281-7136-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/ACCESS51619.2021.9563323.

[4] Rishanti Murugan, Naveen Palanichamy, "Smart City Air Quality Prediction using Machine Learning", 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS) | 978-1-6654-1272-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICICCS51141.2021.9432074.

[5] Limei Ma, Yijun Gao, Chen Zhao,

"Research on Machine Learning Prediction of Air Quality Index Based on SPSS", 978-1-7281-7083-1/20/ $31.00 ©2020 IEEE DOI 10.1109/ ICCNEA50255.2020.00011.

[6] Usha Mahalingam1, Kirthiga Elangovan, Himanshu Dobhal, Chocko Valliappa, Sindhu Shrestha, and Giriprasad Kedam6, "A Machine Learning Model for Air Quality Prediction for Smart Cities", 978-1-5386-9279-0/19/$31.00 c 2019 IEEE.

[7] Pooja Bhalgat, Sejal Pitale, Sachin Bhoite, "Air Quality Prediction using Machine Learning Algorithms", International Journal of Computer Applications Technology and Research Volume 8–Issue 09, 367-370, 2019, ISSN:-2319–8656.

[8] Zhongjie Fu1, Haiping Lin1, Bingqiang Huang and Jiana Yao, "Research on air quality prediction method in Hangzhou based on machine learning", CISAT 2021 Journal of Physics: Conference Series 2010 (2021) 012011 IOP Publishing doi:10.1088/1742-6596/2010/1/012011.

[9] Nilesh N. Maltare, Safvan Vahora, "Air Quality Index prediction using machine learning for Ahmedabad city", Digital Chemical Engineering, https://doi.org/10.1016/ j.dche.2023.100093.

[10] Raunaq Singh Suri1, Ajay Kumar Jain, Nishant Raj Kapoor, Aman Kumar, Harish Chandra Arora, Krishna Kumar, Hashem Jahangir, "Air Quality Prediction - A Study Using Neural Network Based", Journal of Soft Computing in Civil Engineering, https://doi.org/10.22115/ scce.2022.352017.1488.

[11] N. Srinivasa Gupta, Yashvi Mohta, Khyati Heda, Raahil Armaan, B. Valarmathi, and G. Arulkumaran, "Prediction of Air Quality Index Using Machine Learning Techniques: A Comparative Analysis", Hindawi Journal of Environmental and Public Health Volume 2023, Article ID 4916267, 26 pages https:// doi.org/10.1155/2023/4916267.

[12] Tanisha Madan, Shrddha Sagar, Deepali Virmani, "Air Quality Prediction using Machine Learning Algorithms –A Review", 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN) | 978-1-7281-8337-4/20/$31.00 ©2020 IEEE | DOI:10.1109/ICACCCN51052. 2020.9362912.

[13] V. Devasekhar, Dr. P. Natarajan, "Prediction of Air Quality and Pollution using Statistical Methods and Machine Learning Techniques", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 14, No. 4, 2023.

[14] Miss. Sampada Bharat Deshmukh Miss. Komal Prakash Shirsat, Miss. Sakshi Prashant Dhotre Miss. Pooja RavsahebJejurkar, "A Survey on Machine Learning-Based Prediction of Air Quality Index", Vol-7 Issue-6 2021 IJARIIE-ISSN(O)-2395-4396.

[15] Shailesh Munge1, Sagar Kharche1, Piyush Joshi1 Kirti Bathe1, "Air Quality Prediction based on Machine Learning", www.ijcrt.org © 2021 IJCRT | Volume 9, Issue 7 July 2021 |

ISSN: 2320-2882.

[16] Mrs. A. Gnana Soundari, Mrs. J. Gnana Jeslin M.E, Akshaya A.C, "Indian Air Quality Prediction and Analysis Using Machine Learning", International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 11, 2019 (Special Issue) © Research India Publications. http://www.ripublication.com.

[17] Gaganjot Kaur Kang, Jerry Zeyu Gao, Sen Chiao, Shengqiang Lu, and Gang Xie, "Air Quality Prediction: Big Data and Machine Learning Approaches". International Journal of Environmental Science and Development, Vol. 9, No. 1, January 2018.

[18] K. Kumar1, B. P. Pande, "Air pollution prediction with machine learning: a case study of Indian cities", International Journal of Environmental Science and Technology (2023) 20:5333–5348 https://doi.org/10.1007/s13762-022-04241-5.

* * * * *