EVEREST ENGINEERING COLLEGE

(AFFILIATED TO POKHARA UNIVERSITY)

A Minor Project Final Report

On

Air Quality Analysis and Prediction Using Machine Learning

Submitted By:

Kanchan Thapa	[21075415]
Manita Joshi	[21075417]
Sangita Bhattarai	[21075433]
Sunita Bashyal	[21075439]

Submitted To:

Department of Computer and IT Engineering

Everest Engineering College

Sanepa-2, Lalitpur

Feb 25, 2025

ACKNOWLEDGEMENT

We would like to express our deepest gratitude towards the project committee members for their support and suggestions throughout this project. We would also like to thank all the teachers and faculty members who encouraged and guided us during this project.

ABSTRACT

In the developing country like ours, air pollution is becoming the major problem. Air pollution directly affects the human health and generates issues like global warming and acid rain. Among all the particulate matters that determine the quality of air, PM2.5 needs more attention which is found in outdoor air due to combustion of gasoline, oil, diesel and wood. When its level is high in the air, it causes serious issues on people's health. Therefore, the constant checking of the level of particulate matter is necessary. Some of the past researches were done on few parameters and air quality prediction was done. In this project, we studied the correlation between values (Dew point, humidity, precipitation, pressure, temperature) and target value PM2.5. Then we used machine learning model (Decision Tree, Random Forest and Gradient Boosting) for predicting the PM2.5 values in testing set. Comparison between actual PM2.5 and the predicted values of PM2.5 was done. Then the evaluation metrics (MAE, RMSE, R^2) were calculated and Random Forest determined as best model . The graph between actual and predicted values were plotted for every model. Residual test was done and this also shows Random Forest as the best model.

Keywords: *Machine Learning, Particulate matter, dataset, model, regressor.*

Table of Contents

LIST OF FIGURES

LIST OF TABLES

‘

ABBREVIATIONS

CFC	Chlorofloro Carbon
SPM	Suspended Particulate Matters
RMSE	Root Mean Square Error
MAE	Mean Absolute Error
PM	Particulate Matter
RF	Reinforcement Learning
SVR	Support Vector Regression

Chapter 1: INTRODUCTION

1.1 Background

Air quality refers to the purity of the air we breathe, typically assessed by the levels of various pollutants. Air Pollution involves the presence of harmful substances in the atmosphere that can negatively impact human health, the environment, and the climate. Air pollution directly affects human health and contributes to issues like global warming and acid rain, which significantly impact the respiratory systems of living beings. The rapid increase in industrial activity due to economic growth has worsened air pollution. Key pollutants include Temperature, Pressure, Wind dusts, CFCs, Humidity, Precipitation, Dew point etc. and others gases released through the combustion of natural gas, coal, and wood, as well as from factories and vehicles[1].

Particulate Matter (PM) can originate from both human activities and natural sources. It is released during the combustion of solid and liquid fuels used for power generation, heating, and vehicle operation. PM10 and PM2.5 have different chemical compositions and sources [2]. Much of the PM2.5 found in outdoor air comes from the combustion of gasoline, oil, diesel, and wood, while PM10 can include dust from construction sites, landfills, agriculture, wildfires, industrial activities, and pollen. Particularly, PM2.5 affects visibility by altering how light is absorbed and scattered in the atmosphere. It can also harm ecosystems by affecting soil, water quality, and plant health through its deposition and subsequent uptake by organisms.

Machine Learning is a class of algorithms which is data-driven i.e. unlike “normal” algorithms, it is the data that “tells” what the “good answer” is. It uses the data to detect patterns in a data set and adjust program action accordingly. It is a method of data analysis that automates analytical model. Machine learning models fall into three primary categories:

Supervised learning, also known as supervised machine learning, is defined by its use of labeled datasets to train algorithms to classify data or predict outcomes accurately. Some methods used in supervised learning include neural networks, naive bayes, linear regression, logistic regression, random forest, and support vector machine (SVM).

Unsupervised learning, also known as unsupervised machine learning, uses machine learning algorithms to analyze and cluster unlabeled datasets (subsets called clusters). Algorithms used in unsupervised learning include neural networks, k-means clustering, and probabilistic clustering methods.

Reinforcement machine learning, is a machine learning model that is similar to supervised learning, but the algorithm isn't trained using sample data. Algorithms used in reinforcement learning include Q-learning, policy gradient methods, and actor-critic methods.

1.2 Statement of Problem

The rapid urbanization and industrialization in many regions have led to an increase in air pollution levels, significantly affecting the health and well-being of the population. Accompanying the rapid urbanization, many cities in developing countries including Kathmandu have led to environmental problems such as air pollution, water pollution, noise pollution and many more. Air pollution has a direct impact on human's health. Global warming, acid rains, increase in the number of asthma patients are some of the long-term consequences of air pollution. The demand for predicting air quality is becoming increasingly more important to government's policy-making and people's decision making. Precised air quality forecasting can reduce the effect of maximal pollution on the humans and biosphere as well. Hence, finding the best model for prediction becomes crucial.

1.3 Objectives

- To analyze the factors affecting PM2.5 and finding the best predictive model for PM2.5.

1.4 Scopes

The project aims to develop a predictive model for PM2.5 levels using machine learning models. The Air Quality Analysis was done to evaluate historical air pollution data, focusing on PM2.5 levels in Kathmandu city. This helps government agencies, health organizations, and policymakers understand air quality variations and make informed decisions for pollution control measures. While currently limited to Kathmandu, the system can be extended to other cities with sufficient data.

1.5 Applications

- Best model selection for future air quality prediction systems
  Government agencies, environmental researchers, and developers can use this research to determine the best model for air quality prediction.
- Environment policy making guidance
  Kathmandu is our capital city and seasonal impact of PM2.5 need to be studied. So that makes the ease on making policies related this field.
- Public awareness and forecasting system improvement
  Air pollution forecasting apps and websites can use this research to improve their research to improve their prediction models. Meteorologists and Environmentalists can gain insights into the most critical variables affecting air quality.
- Further research and academic use
  This research can be served as reference for future researchers. It can be baseline for other learning models.

Chapter 2: LITERATURE REVIEW

In the recent years, many prediction activities have done in air quality field. As this field is a concern of many researchers, henceforth numerous research has been done.

The study [3] presented a project whose objective is to predict the AQI based on the concentration of PM2.5, PM10, SO_2, NO_2, CO as well as weather conditions like temperature, pressure and humidity. This prediction is brought about with the help of some supervised machine learning algorithms (Linear Regression, Super Vector Regression, Decision Tree Regression, Random Forest Regression).

A study [4] conducted on PM concentrations, were trained with machine learning algorithms. The PM10 concentrations of the years 2009-2017 of 6 stations in Ankara were given as input and the PM_{10} concentrations of the seventh station for the year 2018 were predicted. The spatial distribution of the estimated concentration results was provided through Geographic Information System and spatial strategies for improving air pollution over land use were established.

A work [5] carried out to compare the various machine learning methods such as SARIMA, SVM, LSTM for the prediction of air quality index for Ahmedabad city of Gujarat, India. In this research, different processing methods are used to manage the data before providing to the machine learning models. This study is carried out based on the data provided by the Central Pollution Control Board of India and it focuses on the support vector machine algorithm with RBF kernel model.

Paper published by CR and team[2], logistic regression is employed to detect whether a data sample is either polluted or not polluted. Autoregression is employed to predict future values of PM2.5 based on the previous PM2.5 reading. This system attempts to predict PM2.5 level and detect air quality based on a data set consisting of daily atmospheric conditions in a specific city .

Air Quality Prediction System [6] is the web-based platform that provides current, hourly and daily air quality prediction with which anyone can know about the air quality state and weather conditions of Kathmandu. The system compares different air pollutants like ozone and PM2.5 (particulate matter) and forecast the Air Quality Index. After comparing different machine learning algorithms like Multi-Linear Regression, Random Forest, Neural Network and LSTM (Long Short-Term memory) for efficiency, the system uses Neural Network which predicts hourly, daily and current air quality index.

After researching on these platforms, we aimed to study on the effect of meteorological factors on air pollutant PM2.5. PM2.5 refers to fine inhalable particles with a diameter of 2.5 micrometers or smaller. These tiny particles are so small that they can penetrate deep into the lungs and even enter the bloodstream. So, there must be a proper study conducted on this field. For a city like Kathmandu, pollution is increasing day by day. Our small research on these factors can help a lot to use the safety measures.

Chapter 3: METHODOLOGY

3.1 Work Flow Diagram

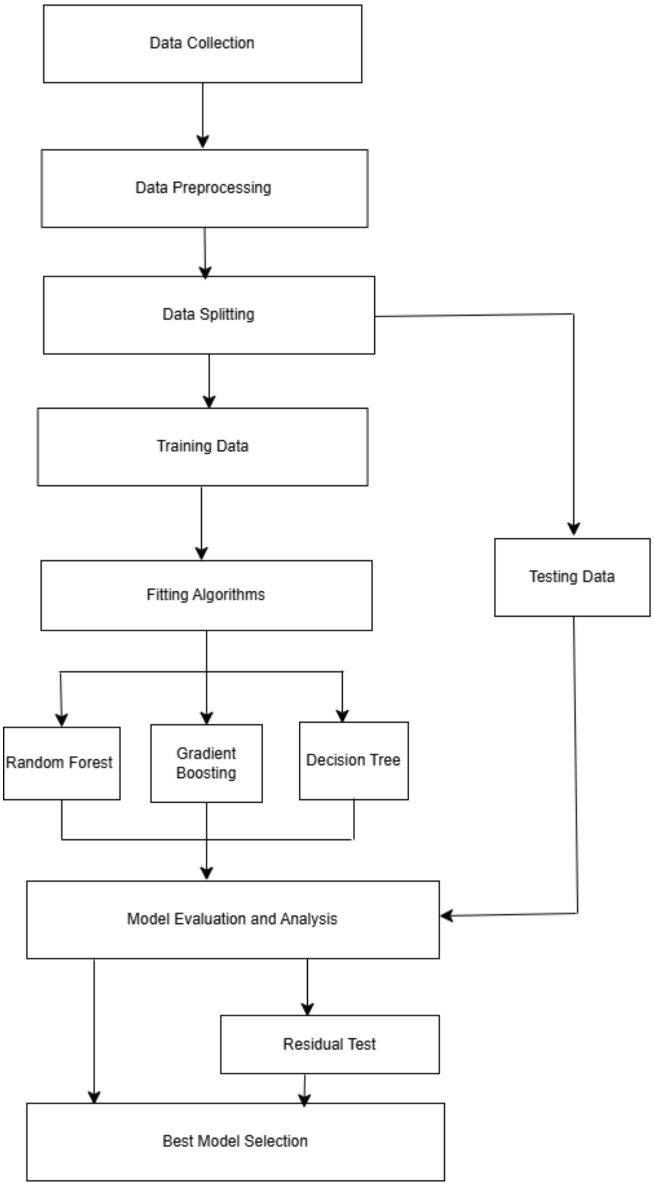


Figure 3. 1 Work Flow Diagram

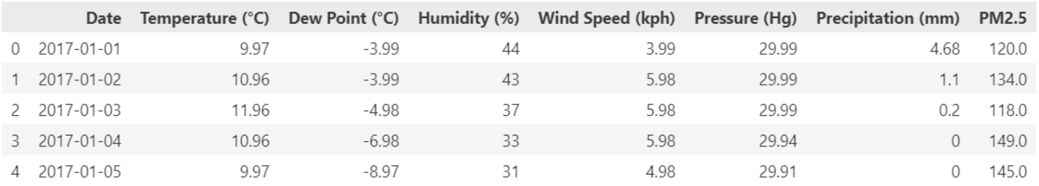
3.2 Work Flow

3.2.1 Data Collection

Data collection is the crucial part of our project. Dataset includes 1461 data of various meteorological factors and pollutant concentrations stored in .xlsx format. That format was changed to CSV directly. The data is sourced from weather and climate and open-source repositories US Embassy.

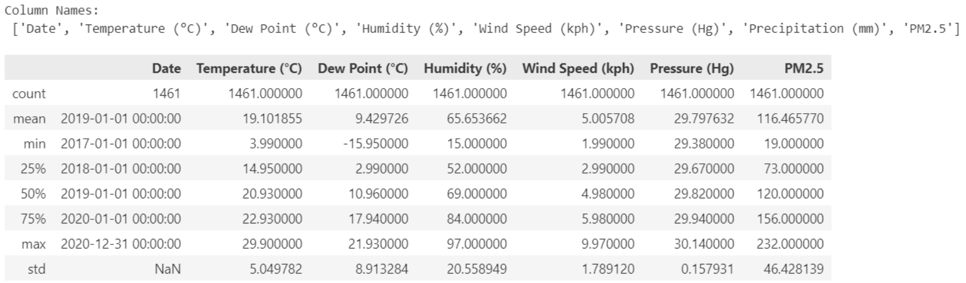
The dataset contains the following features:

- Date: The recorded date of the observation.
- Temperature (°C): The ambient temperature.
- Dew Point (°C): The temperature at which air becomes saturated with moisture.
- Humidity (%): The relative humidity level.
- Wind Speed (kph): The speed of the wind during the measurement.
- Pressure (Hg): The atmospheric pressure recorded in inches of mercury.
- Precipitation (mm): The recorded precipitation level.
- PM2.5: Particulate matter concentration in micrograms per cubic meter.



	Date	Temperature (°C)	Dew Point (°C)	Humidity (%)	Wind Speed (kph)	Pressure (Hg)	Precipitation (mm)	PM2.5
0	2017-01-01	9.97	-3.99	44	3.99	29.99	4.68	120.0
1	2017-01-02	10.96	-3.99	43	5.98	29.99	1.1	134.0
2	2017-01-03	11.96	-4.98	37	5.98	29.99	0.2	118.0
3	2017-01-04	10.96	-6.98	33	5.98	29.94	0	149.0
4	2017-01-05	9.97	-8.97	31	4.98	29.91	0	145.0

Figure 3.2. 1 Dataset



```
Column Names:
 ['Date', 'Temperature (°C)', 'Dew Point (°C)', 'Humidity (%)', 'Wind Speed (kph)', 'Pressure (Hg)', 'Precipitation (mm)', 'PM2.5']
```

	Date	Temperature (°C)	Dew Point (°C)	Humidity (%)	Wind Speed (kph)	Pressure (Hg)	PM2.5
count	1461	1461.000000	1461.000000	1461.000000	1461.000000	1461.000000	1461.000000
mean	2019-01-01 00:00:00	19.101855	9.429726	65.653662	5.005708	29.797632	116.465770
min	2017-01-01 00:00:00	3.990000	-15.950000	15.000000	1.990000	29.380000	19.000000
25%	2018-01-01 00:00:00	14.950000	2.990000	52.000000	2.990000	29.670000	73.000000
50%	2019-01-01 00:00:00	20.930000	10.960000	69.000000	4.980000	29.820000	120.000000
75%	2020-01-01 00:00:00	22.930000	17.940000	84.000000	5.980000	29.940000	156.000000
max	2020-12-31 00:00:00	29.900000	21.930000	97.000000	9.970000	30.140000	232.000000
std	NaN	5.049782	8.913284	20.558949	1.789120	0.157931	46.428139

Figure 3.2. 2 Dataset Information

Here the figure shows the total data count, mean, minimum, first quartile, second quartile, third quartile, maximum and standard deviation clearly.

3.2.2 Data Preprocessing

Data quality is crucial for effective visualization and building accurate machine learning (ML) models. At first, Data was in raw state so we merged and handled the missing values. We also visualized the correlation between the independent variables and dependent variables.

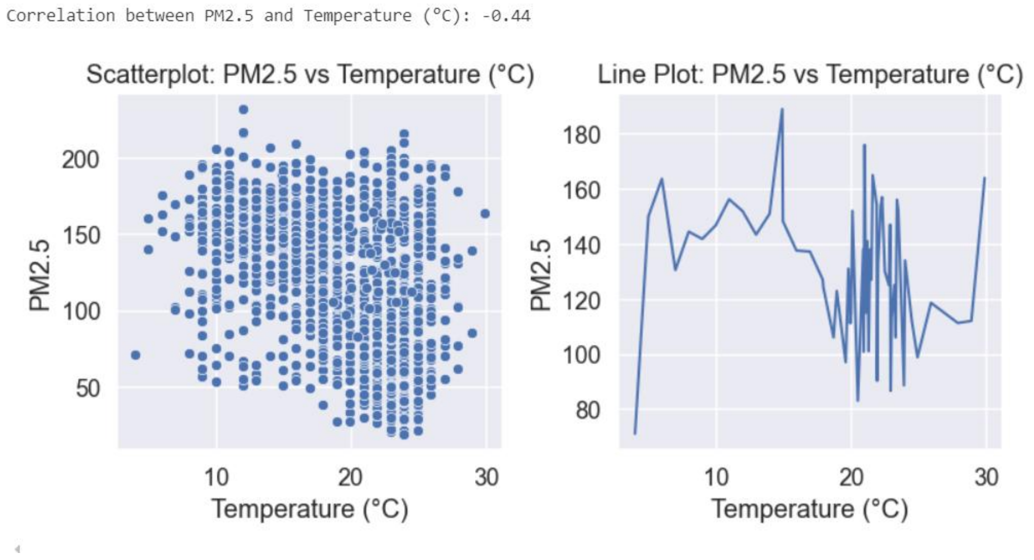
Merging datasets:

Datasets from both platforms (Meteorological dataset and PM2.5) are merged on the basis of date.

Handling Missing Values:

Missing values were addressed using linear interpolation technique.

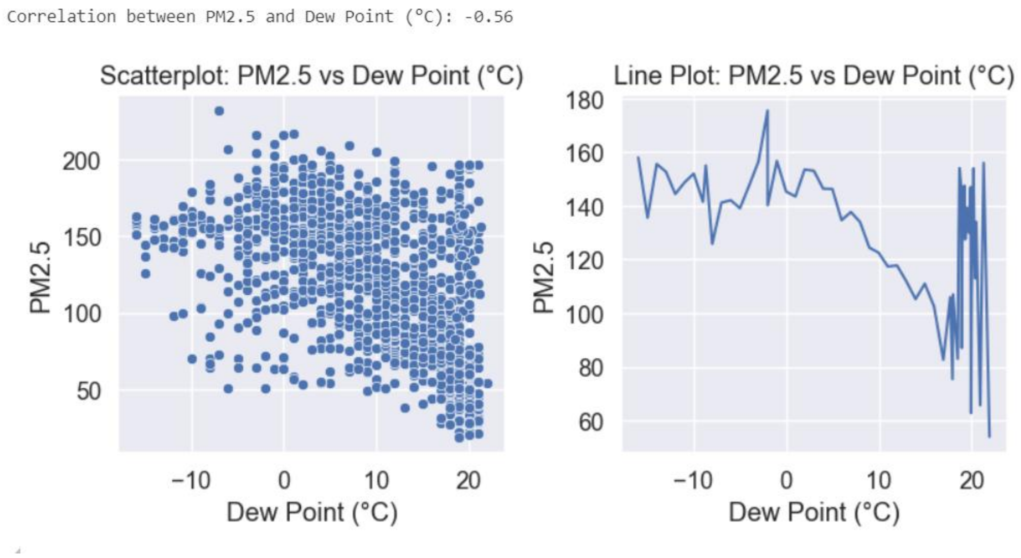
3.2.2.1 Scatter plot and line plot



```
Correlation between PM2.5 and Temperature (°C): -0.44
```

Figure 3.2. 3 Scatter Plot and Line Plot of PM2.5 vs Temperature

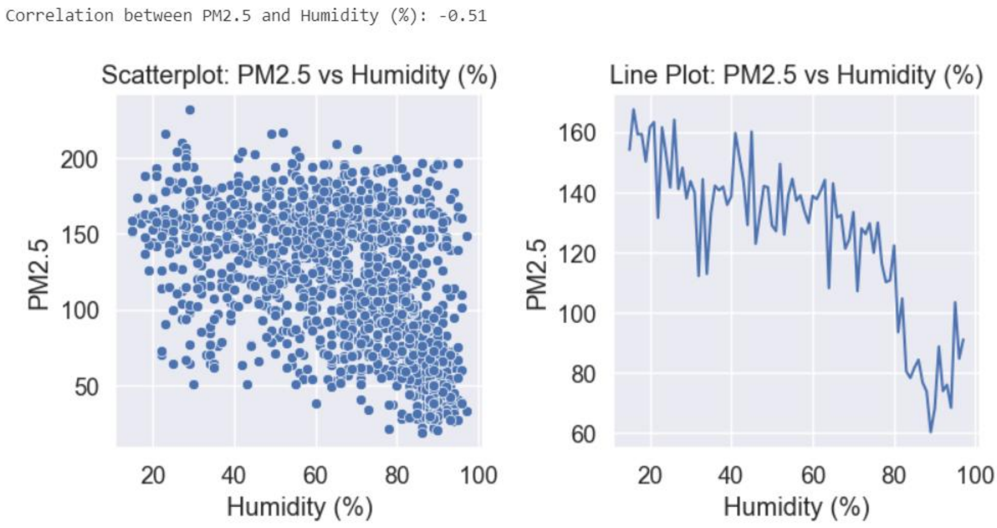
This scatter plot shows the relationship between PM2.5 and Temperature. Correlation is -0.44 i.e. moderate negative correlation which means when temperature increases PM2.5 decreases. The line plot shows the negative correlation graphically.



```
Correlation between PM2.5 and Dew Point (°C): -0.56
```

Figure 3.2. 4 Scatter Plot and Line Plot of PM2.5 vs Dew Point

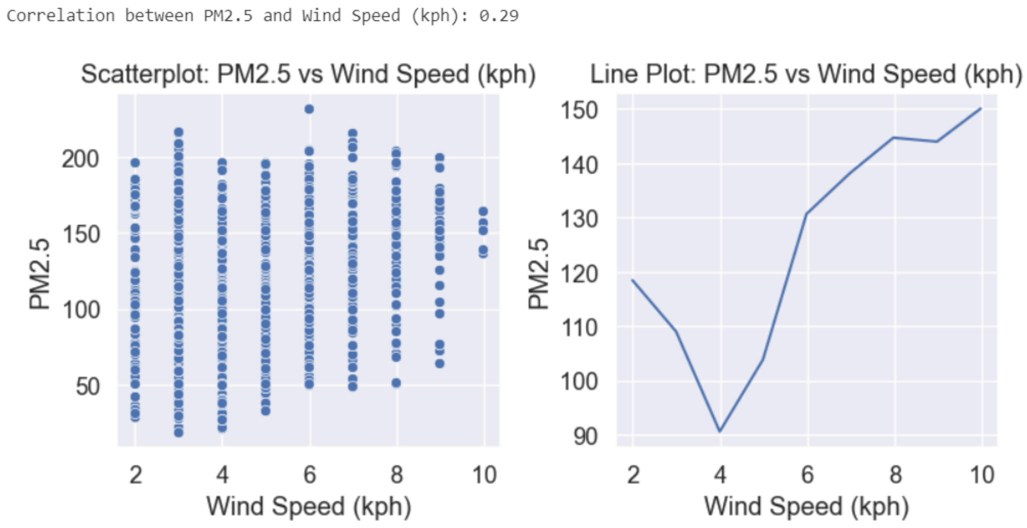
This scatter plot shows the relationship between PM2.5 and Dew Point.The correlation between them is -0.56. As dew point increases, PM2.5 decreases as shown in line plot.



```
Correlation between PM2.5 and Humidity (%): -0.51
```

Figure 3.2. 5 Scatter Plot and Line Plot of PM2.5 vs Humidity

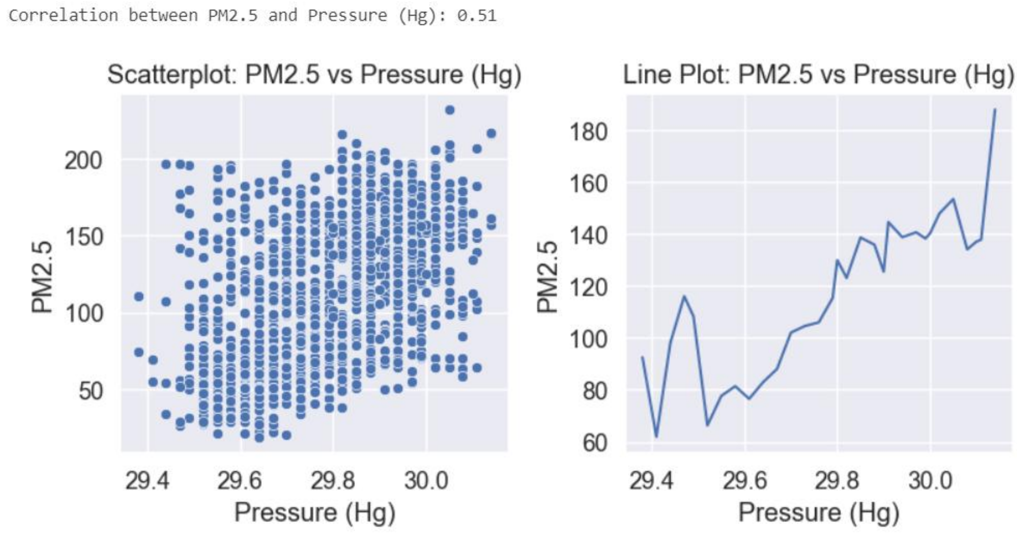
This scatter plot shows correlation between PM2.5 and Humidity. The correlation is -0.51 which is moderate negative correlation. As humidity increases PM2.5 decreases also shown in line plot.



```
Correlation between PM2.5 and Wind Speed (kph): 0.29
```

Figure 3.2. 6 Scatter Plot and Line Plot of PM2.5 vs WindSpeed

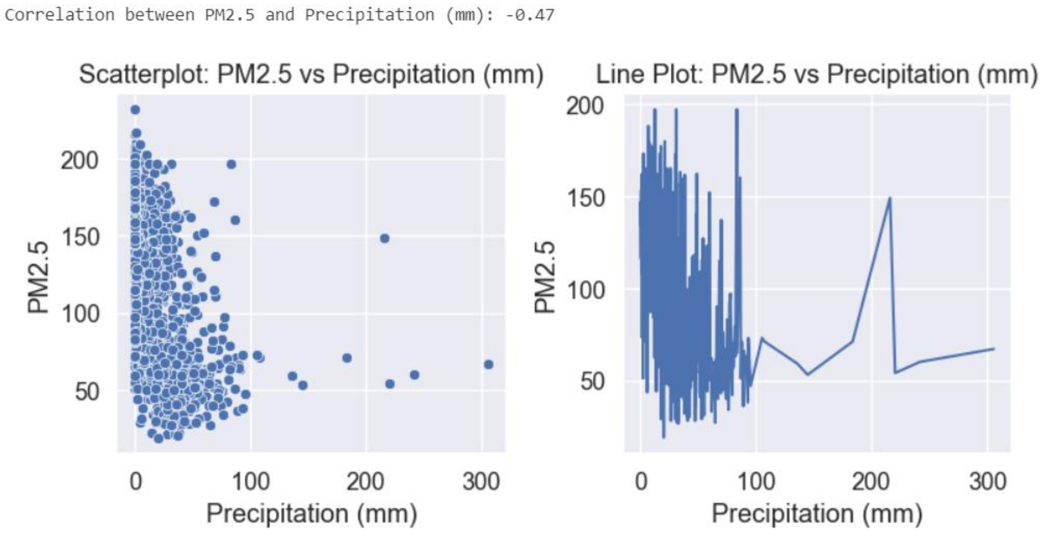
The correlation coefficient between PM2.5 and wind speed are 0.29. This value indicates a weak positive relationship between the two variables. As wind speed increases, PM2.5 levels tend to increase slightly, but the relationship is not strong.



```
Correlation between PM2.5 and Pressure (Hg): 0.51
```

Figure 3.2. 7 Scatter Plot and Line Plot of PM2.5 vs Pressure

The correlation coefficient between PM2.5 and Pressure is 0.51. This value indicates a moderate positive relationship. As Pressure increases, PM2.5 levels tend to increases moderately.



```
Correlation between PM2.5 and Precipitation (mm): -0.47
```

Figure 3.2. 8 Scatter Plot and Line Plot of PM2.5 vs Precipitation

The correlation coefficient between PM2.5 and Precipitation is -0.47. This value indicates a moderate negative relationship. As Precipitation increases, PM2.5 levels tend to decreases.

3.2.2.2 Correlation Heatmap

This correlation heatmap presents the relationships between various environmental factors and PM2.5 levels using Pearson correlation coefficients. Each value indicates the strength and direction of the linear relationship between two variables. A coefficient closer to +1 signifies a strong positive correlation, where an increase in one variable corresponds to an increase in the other. Conversely, a value near -1 indicates a strong negative correlation, where an increase in one variable leads to a decrease in the other. Values near 0 suggest little to no linear relationship.

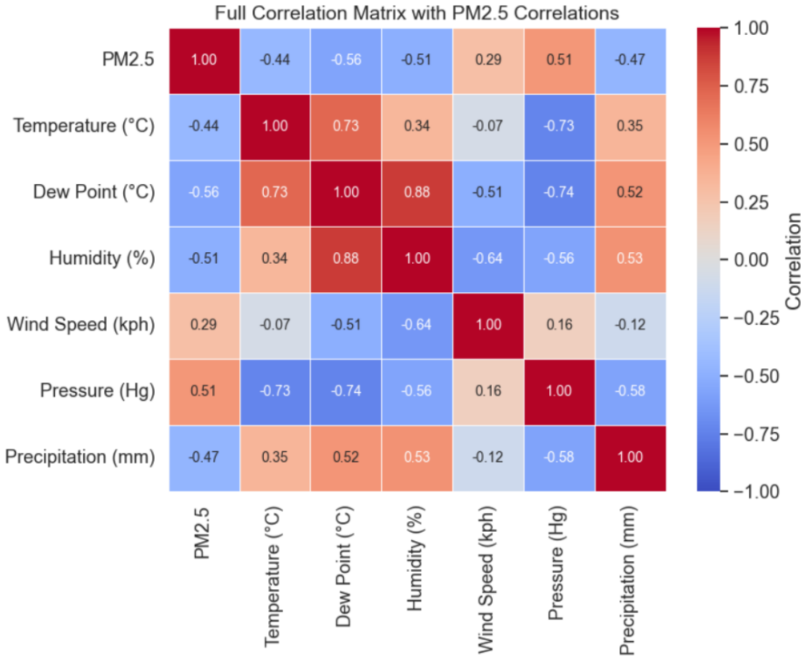


Figure 3.2. 9 Heatmap

It was observed that PM2.5 levels tend to increase when atmospheric pressure is high, showing a moderate positive correlation (0.51). In contrast, PM2.5 levels decrease as temperature rises, indicated by a negative correlation (-0.44). Similarly, dew point and humidity have a strong positive correlation (0.88), meaning that as the dew point increases, the air becomes more humid. On the other hand, pressure and temperature have a strong negative relationship (-0.73), indicating that high atmospheric pressure is often associated with lower temperatures. Wind speed, however, showed only a weak correlation (0.29) with PM2.5, we conclude that wind speed has a minimal impact on air pollution levels.

3.2.2.3 Time Series graphical representation of PM2.5

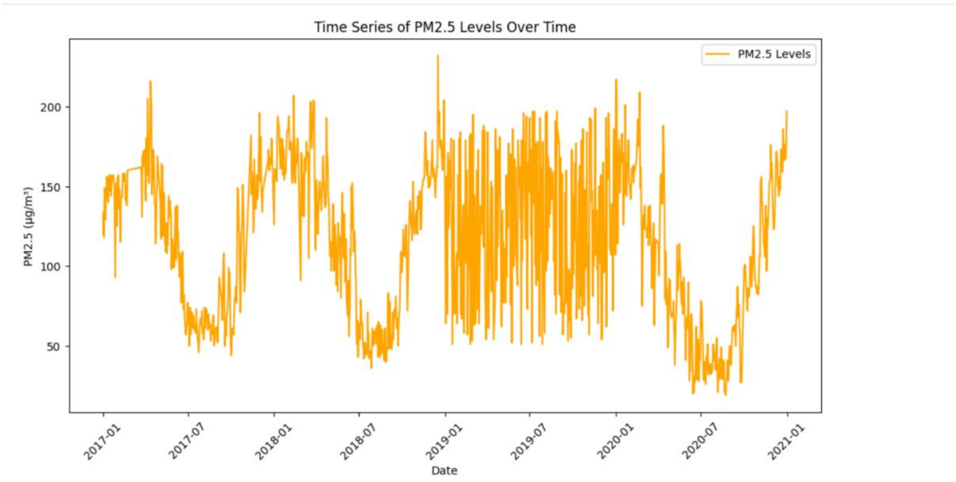
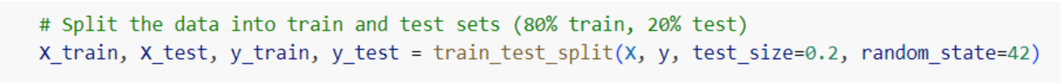


Figure 3.2. 10 Time series representation of PM2.5

The time series graph shows how PM2.5 levels changed from 2017 to early 2021. It shows that pollution is higher in winter and lower in summer. This happens because of factors like temperature, humidity, and pressure. In winter, pollution is higher because of things like temperature traps, more heating, and still air that holds pollutants. In summer, pollution goes down because of rain and better air movement. Overall, PM2.5 levels have been increasing, meaning air quality is getting worse. This shows the need for better pollution control, especially in seasons with more pollution, to protect health and the environment.

3.2.3 Data Splitting

Data splitting is the process of dividing the dataset into distinct subsets to train and test the performance of the models. Proper data splitting is crucial for avoiding overfitting or underfitting. Data should be split so that data sets can have a high amount training data. So, we used Train-Test Split in which training set is 80%, and test set is 20%. Then we saved the training set and testing set into different file. Which makes easier to load whenever the model is to be trained and evaluation.



```
# Split the data into train and test sets (80% train, 20% test)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

3.2.4 Machine Learning Algorithms

After splitting dataset into train-test, we used some machine learning models and predicted the value of PM2.5 and compared with the actual PM2.5.

3.2.4.1 DecisionTreeRegressor

A Decision Tree is a machine learning model used for both classification and regression tasks. It works like a flowchart, where each decision splits the data into smaller groups based on conditions. We are doing regression task so Decision Tree Regressor was used.

We tuned parameters like max_depth, min_samples_split, min_samples_leaf, and max_features to optimize performance. GridSearchCV was used to find the best combination of these parameters.

```
dt_model = DecisionTreeRegressor(random_state=42)

# Define a range of hyperparameters to search over
dt_param_grid = {
    'max_depth': [3, 5, 7, 10, 15],
    'min_samples_split': [2, 10, 20],
    'min_samples_leaf': [1, 5, 10],
    'max_features': [None, 'sqrt', 'log2']
}

# Apply Grid Search to find the best hyperparameters
dt_grid_search = GridSearchCV(dt_model, dt_param_grid, cv=5,
scoring='neg_mean_squared_error', n_jobs=-1, verbose=1)
dt_grid_search.fit(X_train, y_train)


# Train Decision Tree model with the best parameters
best_dt_model = DecisionTreeRegressor(**best_dt_params,
random_state=42)
best_dt_model.fit(X_train, y_train)
dt_y_test_pred = best_dt_model.predict(X_test)
predictions["Decision Tree"] = dt_y_test_pred
```

3.2.4.2 RandomForestRegressor

Random Forest is an ensemble learning method that combines multiple Decision Trees to improve accuracy and reduce overfitting. RandomForestRegresor was used to train the model. Parameters like n_estimators, max_depth, min_samples_split, min_samples_leaf, and max_features were fine-tuned using GridSearchCV. Random Forest performed best in our evaluation.

```
rf_model = RandomForestRegressor(n_estimators=100,
random_state=42)
rf_model.fit(X_train, y_train)
rf_y_test_pred = rf_model.predict(X_test)
predictions["Random Forest"] = rf_y_test_pred
best_rf_model = RandomForestRegressor(**best_rf_params,
random_state=42)
best_rf_model.fit(X_train, y_train)
rf_y_test_pred = best_rf_model.predict(X_test)
predictions["Random Forest"] = rf_y_test_pred
```

3.2.4.3 GradientBoostingRegressor

GradientBoostingRegressor is an advanced machine learning algorithm used for predictive modeling. It builds decision trees sequentially, where each tree corrects the errors of the previous one. This results in a strong predictive model that minimizes errors and improves accuracy. Hyperparameters like n_estimators, learning_rate, max_depth, subsample, and min_samples_split were optimized.

```
gbr_model = GradientBoostingRegressor(random_state=42)

# Define a range of hyperparameters to search over
gbr_param_grid = {
'n_estimators': [50,100, 200, 300,500],
'learning_rate': [0.01, 0.05, 0.1, 0.2],
'max_depth': [3, 5, 7, 10],
'subsample': [0.8, 0.9, 1.0],
'min_samples_split': [2, 10, 50]
}

# Apply Grid Search to find the best hyperparameters
```

```
gbr_grid_search = GridSearchCV(gbr_model, gbr_param_grid,
cv=5, scoring='neg_mean_squared_error', n_jobs=-1, verbose=1)
gbr_grid_search.fit(X_train, y_train)
```

```
best_gbr_model = GradientBoostingRegressor(**best_gbr_params,
random_state=42)
best_gbr_model.fit(X_train, y_train)
gbr_y_test_pred = best_gbr_model.predict(X_test)
predictions["Gradient Boosting"] = gbr_y_test_pred
```

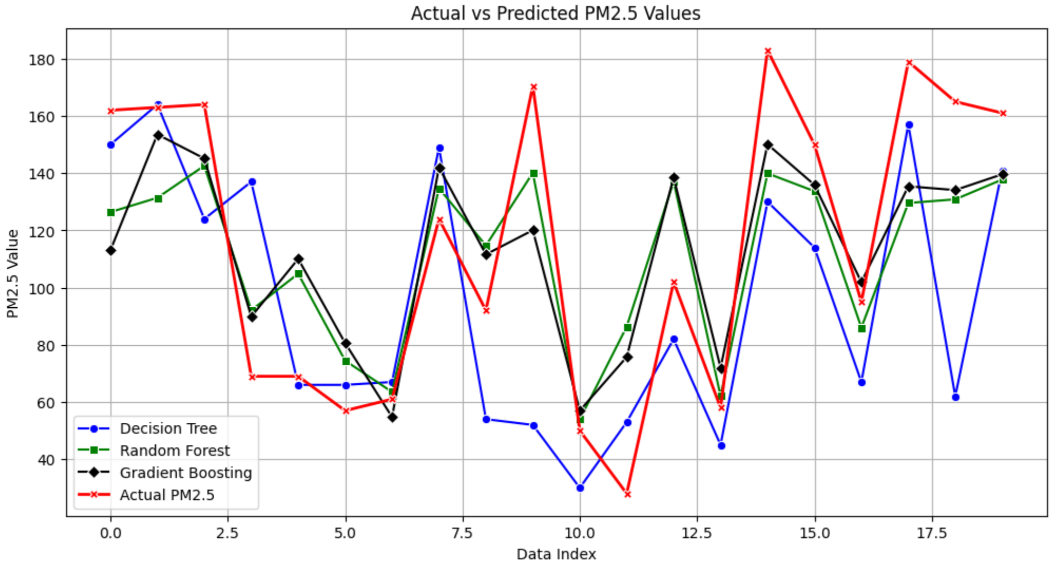


Figure 3.2. 11 Actual PM2.5 vs Predicted PM2.5

This line plot shows the actual vs predicted PM2.5 values using three models. Where purple colour represents the decision tree prediction, green colour represents the random forest prediction, black colour represents the gradient boosting prediction and red colour represents the actual PM2.5. In this plot, Gradient Boosting and Random Forest are following the actual PM2.5 more closely than the Decision Tree which is having more fluctuations indicating overfitting to some points.

3.2.5 Evaluation Metrices

Here are the key metrics we will use for evaluating:

- Root Mean Squared Error: Root Mean Squared Error is a common metric used to evaluate the performance of regression models. It measures the average squared difference between the actual and predicted values.

  RMSE= $\sqrt{\frac{1}{n}}\sum_{i=1}^{n}(yi - yi')$

  Where,

  yi= Actual values

  yi'=Predicted values

  n=Number of data points

```
rmse = np.sqrt(mean_squared_error(y_true, y_pred))
```

In terms of Root Mean Squared Error, Random Forest performs the best with the lowest value of 24.86. This indicates that Random Forest has the least squared error compared to Decision Tree (29.56) and Gradient Boosting (26.15). The smaller RMSE value suggests that Random Forest is the most accurate model when it comes to minimizing the magnitude of errors, making it the best choice for this region.

- Mean Absolute Error: It tells you, on average, how far your predictions are from the actual values. It gives a straight forward measure of how well your model is doing.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|yi - yi'|$$

  yi= Actual values

  yi'=Predicted values

  n=Number of data points

```
mae = mean_absolute_error(y_true, y_pred) Using this MAE of
different models was calculated.
```

  Among the three models, Random Forest performs the best in terms of Mean Absolute Error. With an MAE of 17.49, it has the smallest average absolute errors compared to Decision Tree (21.47) and Gradient Boosting (18.39). This indicates that Random Forest is the most accurate model when considering MAE, making it the best choice for minimizing average prediction errors in this region.

- R-Squared: R^2 shows how well your model explains the variation in the data. It ranges from 0 to 1:0 means the model explains none of the variation (bad). 1 means the model explains all the variation (good).

$$R^2 = 1 - \frac{RSS}{TSS}$$

Where, RSS= Residual Sum of Squares

TSS= Total Sum of Squares

$RSS = \sum_{i=1}^{n}(yi - yi')^2$

$$TSS = \sum_{i=1}^{n}(yi - yi^{-})^2$$

Where, yi= actual values

yi'= predicted values

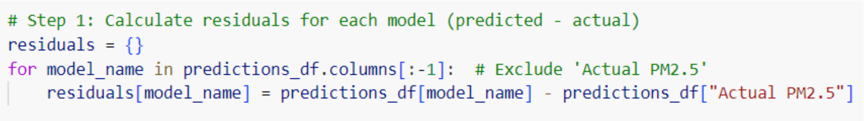
```
r2 = r2_score(y_true, y_pred)
```

When comparing the models based on R-squared (R^2), Random Forest stands out with the highest value of 0.72, meaning it explains 72% of the variance in the target variable. This indicates that Random Forest captures the relationship between the features and the target variable more effectively than the other models. In contrast, Decision Tree has an R^2 of 0.60, and Gradient Boosting has 0.69, both of which are lower than Random Forest. Therefore, Random Forest is the best model in terms of how well it fits the data and explains the variance in the target variable.

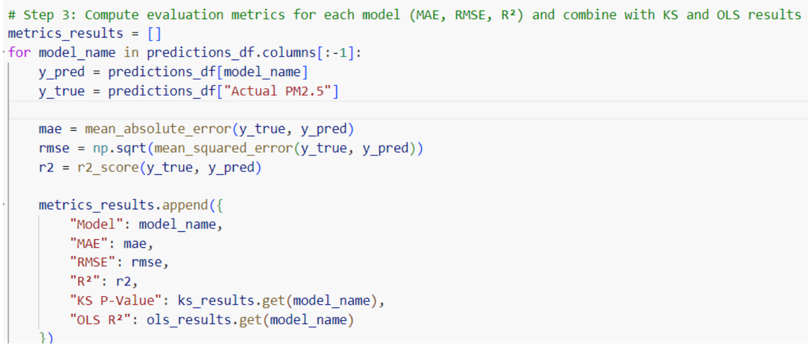
3.2.6 Residual Test

The residual test is done to evaluate the accuracy and validity of a regression model by analyzing the differences (residuals) between the actual and predicted values. This helps in identifying patterns that indicate whether the model assumptions hold true or if there are any issues like bias, heteroscedasticity, or autocorrelation.

Evaluation metrics like RMSE, MAE, R2 may have the respective values to be the best model but not all the time it is sure that they are giving the best model. So, to check the model is best or not residual test is done. In this project residual test was done like below:



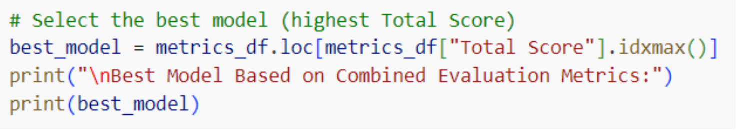
```
# Step 1: Calculate residuals for each model (predicted - actual)
residuals = {}
for model_name in predictions_df.columns[:-1]:  # Exclude 'Actual PM2.5'
    residuals[model_name] = predictions_df[model_name] - predictions_df["Actual PM2.5"]
```



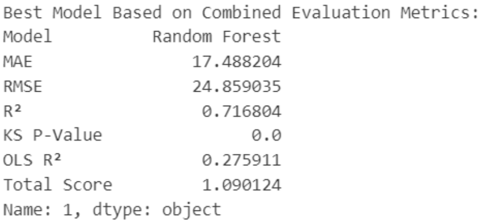
```
# Step 3: Compute evaluation metrics for each model (MAE, RMSE, R²) and combine with KS and OLS results
metrics_results = []
for model_name in predictions_df.columns[:-1]:
    y_pred = predictions_df[model_name]
    y_true = predictions_df["Actual PM2.5"]

    mae = mean_absolute_error(y_true, y_pred)
    rmse = np.sqrt(mean_squared_error(y_true, y_pred))
    r2 = r2_score(y_true, y_pred)

    metrics_results.append({
        "Model": model_name,
        "MAE": mae,
        "RMSE": rmse,
        "R²": r2,
        "KS P-Value": ks_results.get(model_name),
        "OLS R²": ols_results.get(model_name)
    })
```



```
# Select the best model (highest Total Score)
best_model = metrics_df.loc[metrics_df["Total Score"].idxmax()]
print("\nBest Model Based on Combined Evaluation Metrics:")
print(best_model)
```



```
Best Model Based on Combined Evaluation Metrics:
Model            Random Forest
MAE                  17.488204
RMSE                 24.859035
R²                    0.716804
KS P-Value                 0.0
OLS R²                0.275911
Total Score           1.090124
Name: 1, dtype: object
```

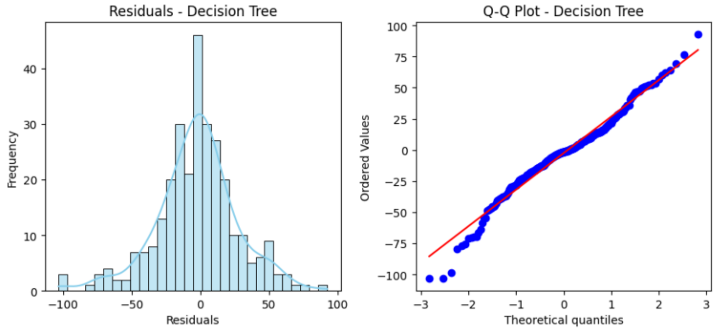


Figure 3.2. 12 Residual Plot of Decision Tree

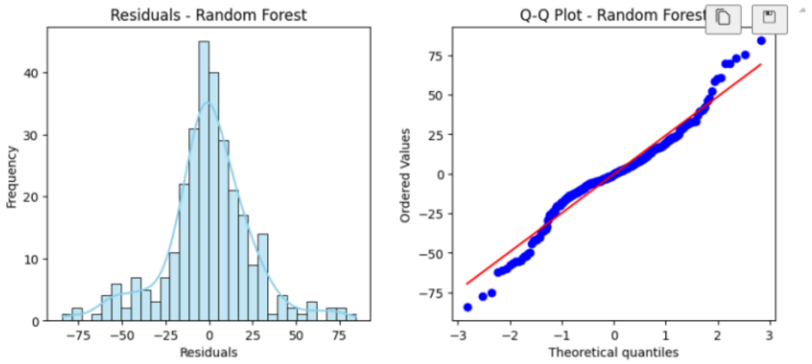


Figure 3.2. 13 Residual Plot of Random Forest

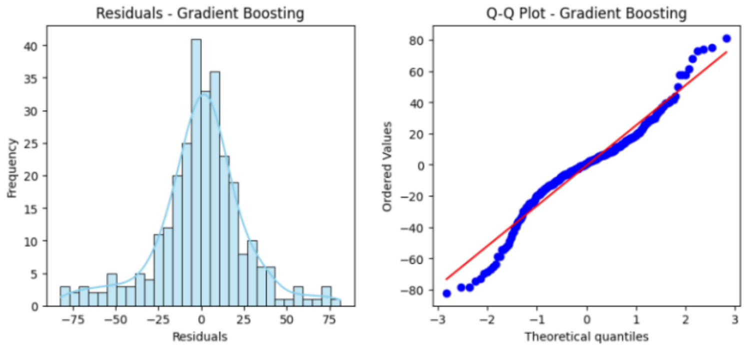


Figure 3.2. 14 Residual Plot of Gradient Boosting

In the graph of Decision Tree The residuals show a moderate spread with a peak at zero, but there are visible outliers at both ends. This suggests that the model overfits certain data points and does not generalize well. In the graph of Random Forest the residuals appear more symmetrically distributed compared to the Decision Tree. The distribution is closer to normal, with fewer extreme outliers, indicating improved stability. In the graph of Gradient Boosting the residuals also appear relatively normally distributed but exhibit a slightly higher peak and slightly more outliers than Random Forest.

The histogram of residuals for the Random Forest model is more symmetrically distributed and closely follows the normal distribution compared to the Decision Tree and Gradient Boosting models. This indicated less skewness and fewer extreme errors.

Q-Q Plot Analysis:

In the graph of Decision Tree, Q-Q plot shows deviations at the tails, indicating that the residuals are not perfectly normal. This suggests that extreme values (outliers) exist in the residuals. In the graph of Random Forest, the Q-Q plot aligns more closely with the diagonal, implying that the residuals follow a normal distribution more closely than the Decision Tree. In the graph of Gradient Boosting the Q-Q plot is similar to that of the Random Forest but slightly deviates at the upper tail, showing some degree of skewness.

The Q-Q plot for the Random Forest model shows residuals that align more closely with the theoretical normal distribution line. While some deviation is present at the tails, the overall fit is better than the Decision Tree and Gradient Boosting models, which show larger deviations.

Chapter 4: RESULTS AND ANALYSIS

In this project, we collected data from Weather and Climate and US Embassy, Kathmandu. Weather and Climate is the main source of meteorological data (humidity, wind speed, precipitation, pressure etc.) which are the independent variables. PM2.5 was taken from US Embassy, Kathmandu which is our target variable. Then both the datasets were merged according to the date (2017-2020).

We have used 3 regressor models (Decision Tree Regressor, Random Forest Regressor, Gradient Boosting Regressor) to train the model and predict the PM2.5 for test set. Then we compared both of them using evaluation metrices to check the error and variation of data. Evaluation metrices used are RMSE, MAE, R^2.

Here is the distribution of evaluation metrices of regressor models:

Table 4. 1 Evaluation metrices of models

Models	**MAE**	**RMSE**	**R^2**
Decision Tree	21.473964	29.562766	0.599495
Random Forest	17.488204	24.859035	0.716804
Gradient Boosting	18.390497	26.152742	0.686562

Based on the evaluation metrics, Random Forest emerged as the best model for air quality prediction. It has the lowest MAE (17.49), indicating the least average prediction error, and the lowest RMSE (24.86). Additionally, it achieved the highest R^2 (0.717), explaining 71.7% of the variance in the data, making it the most reliable model among the three.

Gradient Boosting is the second-best model, performing slightly worse than Random Forest but still better than the Decision Tree. Although its errors are slightly higher, it achieves an R² of 0.687, meaning it explains 68.7% of the variance in the data, making it a strong alternative.

On the other hand, Decision Tree performed the worst among the three models. It has the highest MAE (21.47) and RMSE (29.56), indicating the largest prediction errors. Additionally, its R² score of 0.599 shows that it explains only 59.9% of the variance, making it the least effective model for air quality prediction.

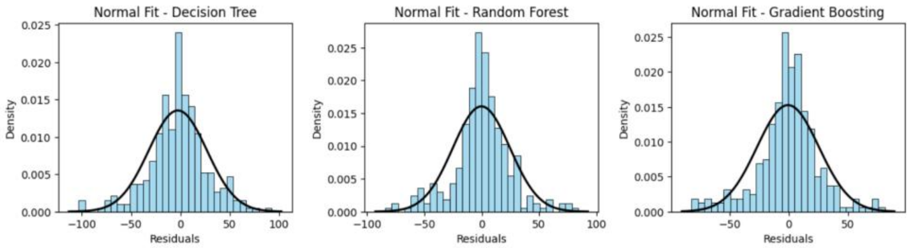


Figure 4. 1 Normal Fit comparison of different models

Residual test was done to check if the evaluation metrices gave the best model or not. The histogram of residuals for the Random Forest model appeared more symmetrically distributed and closely followed the normal distribution compared to the Decision Tree and Gradient Boosting models. This indicated less skewness and fewer extreme errors. The Q-Q plot for the Random Forest model showed residuals that aligned more closely with the theoretical normal distribution line. While some deviation is present at the tails, the overall fit is better than the Decision Tree and Gradient Boosting models, which show larger deviations.

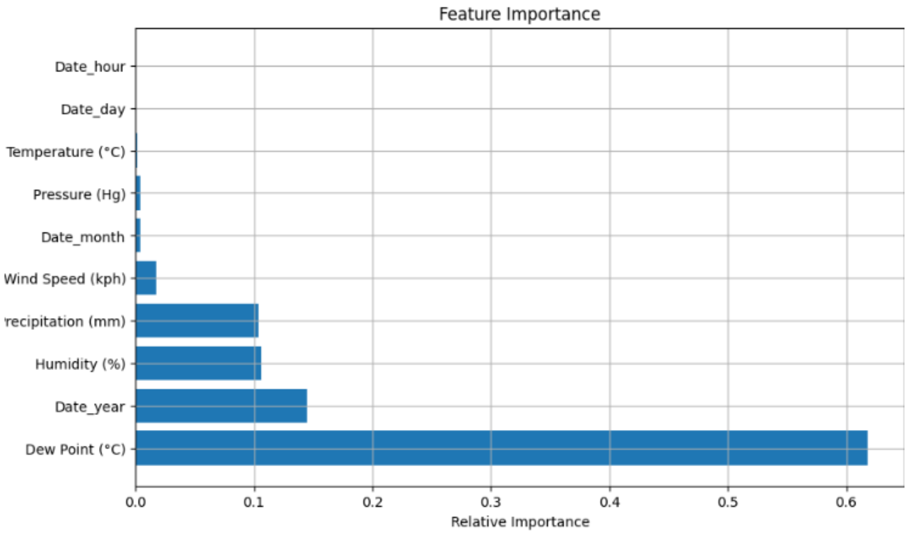


Figure 4. 2 Feature Importance of Random Forest

In Random Forest, Dew Point (°C) remains the most influential factor, but its dominance is slightly reduced compared to Gradient Boosting. Date_year, Humidity (%), and Precipitation (mm) contribute noticeably, while Wind Speed (kph) has the lowest impact. Random Forest refines feature importance better than a single Decision Tree by considering multiple trees, leading to a more generalized model.

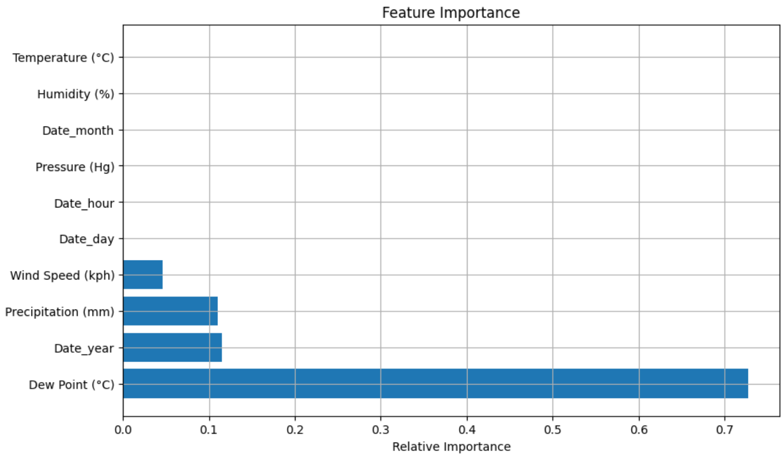


Figure 4. 3 Feature Importance of Decision Tree

In Decision Tree, dew Point (°C) holds the highest importance in the Decision Tree model. Date_year has some significance but is much lower than Dew Point. Humidity (%), Precipitation (mm), and Wind Speed (kph) contribute minimally.

The Decision Tree model relies heavily on a single feature, making it prone to overfitting.

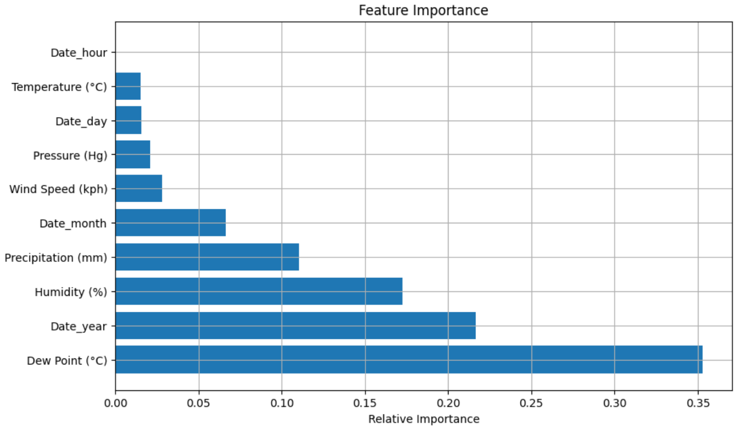


Figure 4. 4 Feature Importance of Gradient Boosting

In Gradient Boosting, dew Point (°C) remains the most important feature, but the distribution is more balanced compared to the Decision Tree model. Humidity (%), Date_year, Precipitation (mm), and Wind Speed (kph) contribute significantly. Unlike the Decision Tree model, none of the features have negligible importance, meaning all features play a role in predictions. Gradient Boosting distributes feature importance more evenly, making it a more robust model.

Dew Point (°C) is highly important in our dataset as it's closely related to air moisture content, which directly influences PM2.5 concentration. Higher moisture levels can either accumulate or remove pollutants, making Dew Point a strong predictor of air quality.

Chapter 5: CONCLUSION

We utilized DecisionTreeRegressor, RandomForestRegressor, and Gradient Boosting Regressor to train our model for predicting PM2.5 levels. After training, we tested the model's performance on the testing dataset and evaluated it using key metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R^2 score to compare different models and determine the most effective one.

Among the models, RandomForestRegressor turned out to be a best-performing model with an MAE of 17.48, RMSE of 24.85, and an R^2 score of 0.71. The Gradient Boosting Regressor and Decision Tree Regressor followed closely in performance.

To further validate our selection, we conducted residual testing, as evaluation metrics alone do not always guarantee the best model. The residual analysis reaffirmed that RandomForestRegressor was the most suitable model for PM2.5 prediction, reinforcing our initial findings based on evaluation metrics.

Chapter 6: FUTURE WORK

Currently, the model uses meteorological data, but we can include other factors like traffic data, industrial emissions, and even satellite-based air pollution data to improve accuracy. We can explore deep learning techniques like Long Short-Term Memory for time series forecasting or hybrid models that combine multiple approaches can lead to better predictions. We can deploy the model as a real-time air quality monitoring system using IoT sensors and integrating it into a mobile or web application for public use. The model can be extended to predict air quality in different cities or regions by using geospatial data and advanced mapping techniques.

REFERENCES

[1] K. Kumar and B. P. Pande, “Air pollution prediction with machine learning: a case study of Indian cities,” *Int. J. Environ. Sci. Technol.*, vol. 20, no. 5, pp. 5333–5348, May 2023, doi: 10.1007/s13762-022-04241-5.

[2] C. Aditya, C. R. Deshmukh, D. Nayana, and P. G. Vidyavastu, “Detection and prediction of air pollution using machine learning models,” *Int. J. Eng. Trends Technol. IJETT*, vol. 59, no. 4, pp. 204–207, 2018.

[3] S. Halsana, “Air Quality Prediction Model using Supervised Machine Learning Algorithms,” *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, pp. 190–201, Jul. 2020, doi: 10.32628/CSEIT206435.

[4] A. Bozdağ, Y. Dokuz, and Ö. B. Gökçek, “Spatial prediction of PM10 concentration using machine learning algorithms in Ankara, Turkey,” *Environ. Pollut.*, vol. 263, p. 114635, Aug. 2020, doi: 10.1016/j.envpol.2020.114635.

[5] N. N. Maltare and S. Vahora, “Air Quality Index prediction using machine learning for Ahmedabad city,” *Digit. Chem. Eng.*, vol. 7, p. 100093, Jun. 2023, doi: 10.1016/j.dche.2023.100093.

[6] “Aqi To Print | PDF | Regression Analysis | Air Pollution.” Accessed: Dec. 26, 2024. [Online]. Available: https://www.scribd.com/document/510300340/Aqi-to-Print