Indian Institute of Technology, Dharwad

CS209 : Artificial Intelligence
And
CS214 : Artificial Intelligence Laboratory

Air Quality Prediction using ML Models

Course Instructor:
Dr. Dileep A.D.
Mentor Name:
Neha

Submitted by:

1. Preksha
2. Srinivas
3. Deepthi
4. Sricharan

Contents

List of Figures

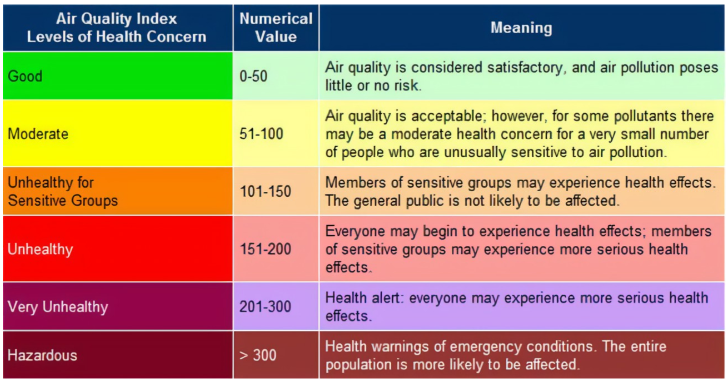
List of Tables

1 Abstract

This project aims to harness the power of Python and machine learning on a dataset comprising 5000 samples to predict values based on relevant environmental parameters.Using data-driven techniques and machine learning algorithms (Linear Regression, Random Forest Regressor, Decision Trees, Support Vector Machines etc.), we endeavor to create a robust model that takes environmental parameters as input and classify the air Quality into four levels–Good,Moderate,Poor, and Hazardous and to identify the factors which are the most influential to the air quality. The Random Forest model outperformed others with high accuracy and interpretability.

2 Introduction

Air is one of the most signifcant elements of the environment. The increasing global air pollution crisis poses an unavoidable threat to human health, environmental sustainability, ecosystems, and the earth's climate. Air pollution has been referred to as a silent killer due to its insidious nature. Its indirect impact on human health further underscores its dangerous efects. Early detection of air quality can potentially save millions of lives globally. A unique and transformative approach can harness the power of machine learning to combat air pollution. This research presents a manual and web-based automatic prediction system that provides real-time alerts on air quality status and can help prevent premature deaths, chronic diseases, and other health problems. Air pollutants, including carbon monoxide (CO), nitrogen dioxide (NO2), and particulate matter (PM 2.5), are used in this study for feature analysis and extraction.



Air Quality Index Levels of Health Concern	Numerical Value	Meaning
Good	0-50	Air quality is considered satisfactory, and air pollution poses little or no risk.
Moderate	51-100	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	101-150	Members of sensitive groups may experience health effects. The general public is not likely to be affected.
Unhealthy	151-200	Everyone may begin to experience health effects; members of sensitive groups may experience more serious health effects.
Very Unhealthy	201-300	Health alert: everyone may experience more serious health effects.
Hazardous	> 300	Health warnings of emergency conditions. The entire population is more likely to be affected.

Figure 2.1: Air Quality Infographics

3 Dataset Description

The dataset used in this project consists of 5000 records and includes both environmental and demographic factors that can influence air quality. The target variable categorizes air quality into four discrete levels: *Good*, *Moderate*, *Poor*, and *Hazardous*.

3.1 Environmental Factors

The environmental features capture various air pollutants and weather-related parameters that directly affect air quality:

Feature	Description
Temperature (°C)	Indicates the average temperature in the region, which can affect the dispersion of pollutants.
Humidity (%)	High humidity can influence the chemical transformation of pollutants.
PM2.5 Concentration (µg/m^3)	Fine particulate matter that penetrates deep into lungs and affects respiratory health.
PM10 Concentration (µg/m^3)	Coarse particulate matter that can cause breathing issues and eye irritation.
NO_2 Concentration (ppb)	Emitted from vehicles and industrial activity; harmful to the respiratory system.
SO_2 Concentration (ppb)	Produced from fossil fuel combustion; can irritate eyes and lungs.
CO Concentration (ppm)	A colorless, odorless gas that interferes with oxygen transport in the blood.

Table 3.1: Environmental Features and Descriptions

3.2 Demographic and Geographic Factors

These features describe characteristics of the region that may indirectly influence air quality:

Feature	Description
Proximity to Industrial Areas (km)	Distance from the region to the nearest industrial zone. Areas closer to industrial activity tend to have higher pollution levels.
Population Density (people/km^2)	Number of people living per square kilometer. Higher density often correlates with more vehicle emissions and energy usage.

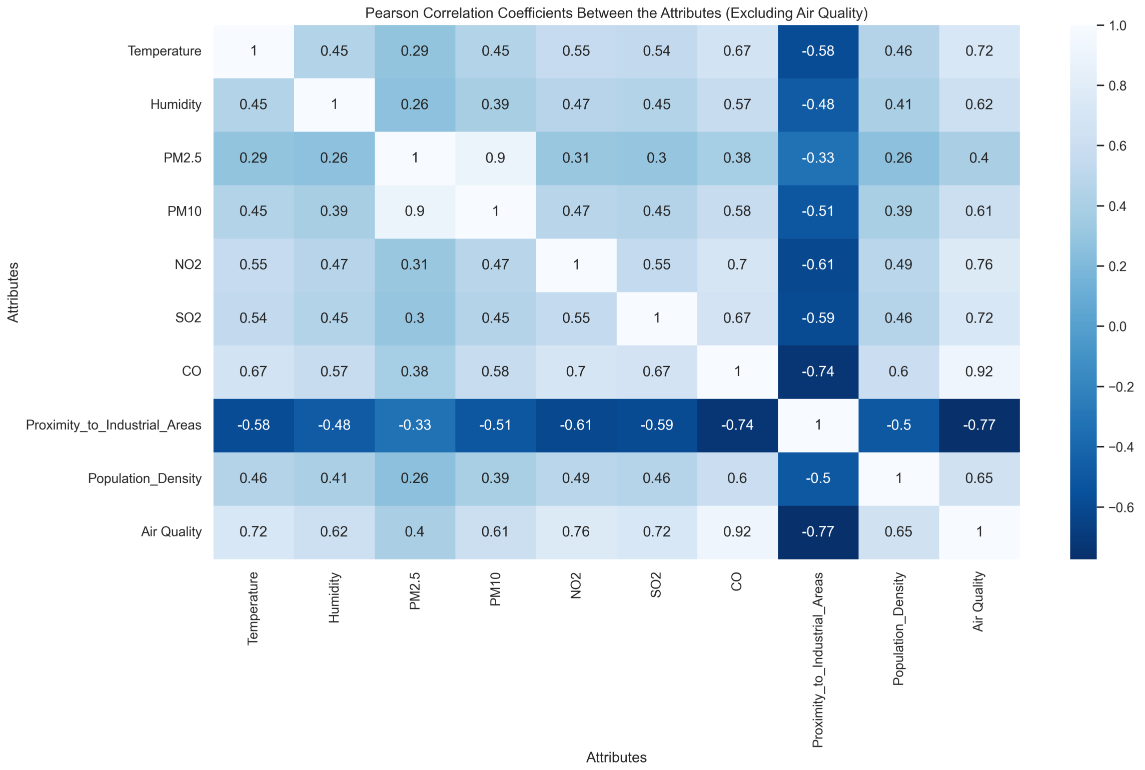
Table 3.2: Demographic and Geographic Features

3.3 Target Variable: Air Quality Levels

The dataset classifies air quality into the following categories:

Air Quality Level	Description
Good	Clean air with low levels of pollutants.
Moderate	Acceptable air quality with some pollutants present, minor risk to sensitive groups.
Poor	Pollution levels are high enough to affect sensitive individuals.
Hazardous	Very high pollution levels posing serious health risks to everyone.

Table 3.3: Target Variable: Air Quality Levels



	Temperature	Humidity	PM2.5	PM10	NO2	SO2	CO	Proximity_to_Industrial_Areas	Population_Density	Air Quality
Temperature	1	0.45	0.29	0.45	0.55	0.54	0.67	-0.58	0.46	0.72
Humidity	0.45	1	0.26	0.39	0.47	0.45	0.57	-0.48	0.41	0.62
PM2.5	0.29	0.26	1	0.9	0.31	0.3	0.38	-0.33	0.26	0.4
PM10	0.45	0.39	0.9	1	0.47	0.45	0.58	-0.51	0.39	0.61
NO2	0.55	0.47	0.31	0.47	1	0.55	0.7	-0.61	0.49	0.76
SO2	0.54	0.45	0.3	0.45	0.55	1	0.67	-0.59	0.46	0.72
CO	0.67	0.57	0.38	0.58	0.7	0.67	1	-0.74	0.6	0.92
Proximity_to_Industrial_Areas	-0.58	-0.48	-0.33	-0.51	-0.61	-0.59	-0.74	1	-0.5	-0.77
Population_Density	0.46	0.41	0.26	0.39	0.49	0.46	0.6	-0.5	1	0.65
Air Quality	0.72	0.62	0.4	0.61	0.76	0.72	0.92	-0.77	0.65	1

Figure 3.1: heat map of all the factors affecting the air quality

4 Libraries Used

This project utilized a wide range of Python libraries for data processing, visualization, machine learning modeling, and evaluation:

- **pandas** – For data manipulation and analysis.
- **numpy** – For numerical computations and array operations.
- **matplotlib** and **seaborn** – For generating visualizations including histograms, bar plots, and heatmaps.
- **torch (PyTorch)** – For implementing and training custom neural network models.
- **scikit-learn (sklearn)** – For preprocessing, dimensionality reduction (PCA), model building (Random Forest, Decision Tree, Logistic Regression, KNN, SVM, MLP-Classifier), evaluation metrics, and hyperparameter tuning using GridSearchCV.
- **imblearn (SMOTE)** – For handling class imbalance by oversampling the minority class using Synthetic Minority Over-sampling Technique (SMOTE).
- **scipy.stats** – For statistical computations and multivariate distributions.

5 Tasks and Methodology

5.1 Data Exploration, Preprocessing and Visualisation

Initial inspection confirmed that the dataset contained no null values. However, certain anomalies were present in the data:

- **Class Distribution**

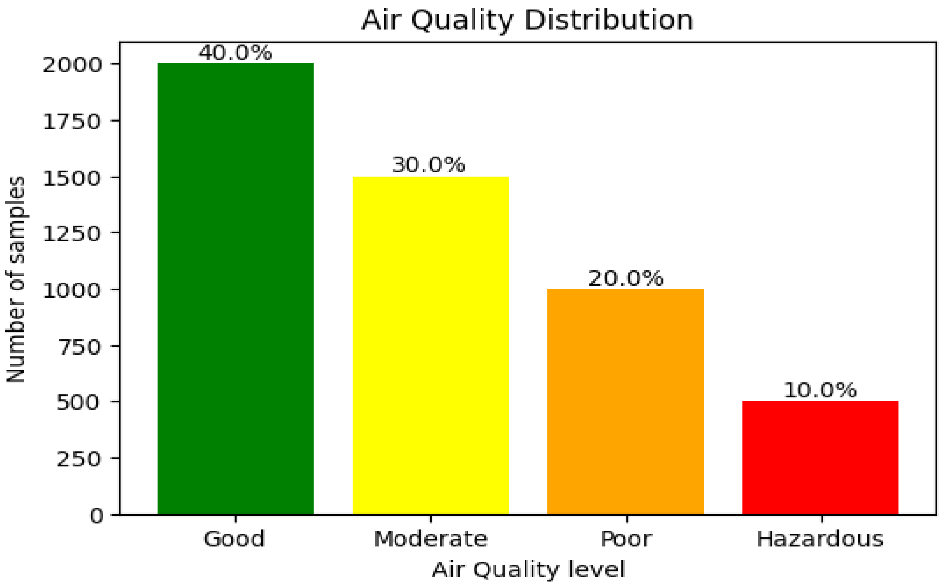


Figure 5.1: Class Distribution of Air Quality Levels

This visualization provides insights into the balance or imbalance of classes in the dataset, which is crucial for selecting appropriate modeling techniques and evaluating performance.

- **Negative Values:** Some entries had negative values, which are physically invalid. These were corrected by clipping using lower bound clipping.

- **Outlier Treatment:** Several pollutant concentration features, including PM2.5, PM10, NO_2, and CO, exhibited extreme outliers.(fig 4.3) To handle these:

  - A **logarithmic transformation** was applied to reduce skewness and minimize the impact of large values. Since most features were right-skewed with wide ranges and outliers, the transformation helped stabilize variance and made the data more suitable for modeling.(fig 4.2)

$$x_{\text{transformed}} = \log(x + 1)$$

  - Remaining outliers after transformation were identified using interquartile range (IQR) and replaced with the mean of the respective feature.

- **Feature Scaling and Encoding:** All continuous variables were scaled using z-score normalization. The categorical target variable, representing air quality levels, was label-encoded i.e 0 for 'Good', 1 for 'Moderate', 2 for 'Poor' and 3 for 'Hazardous' for use in machine learning models.

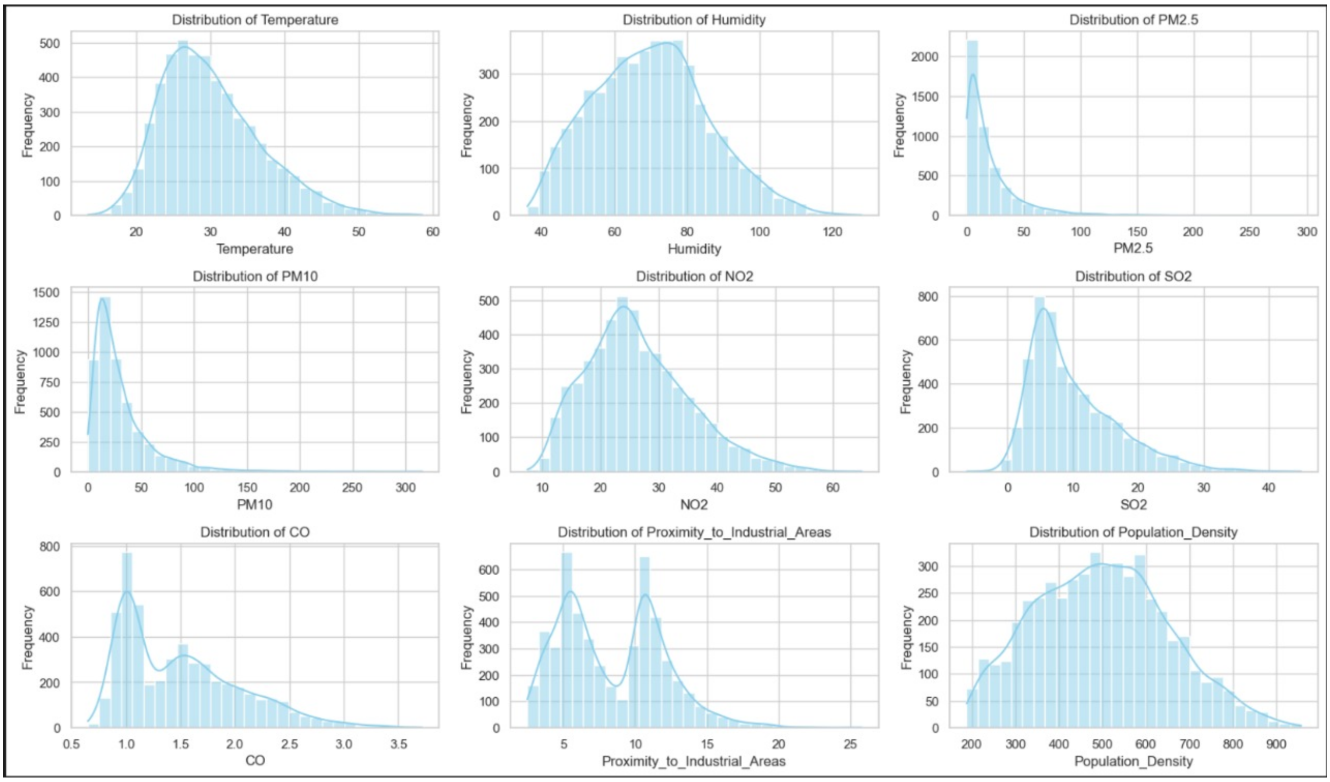


Figure 5.2: distribution plots after Log Transformation

- **Train-Validation-Test Split:** The dataset, including the target variable (air quality), was divided into training, validation, and test sets in the proportions of 60%, 20%, and 20%, respectively.

  While splitting the dataset, class balance was maintained across the training, validation, and test sets to ensure fair representation of all classes and reliable model evaluation.(fig 4.4)

These preprocessing and data preparation steps ensured that the data was clean, consistent, suitable and ready for model training.

5.2 Machine Learning Models

The performance of various ML models is assessed by eight classifiers:Random Forest, Decision Trees, Bayes Classifer, Navie Bayes Classifier, Support Vector Machines, Logistic Regression, Neural Networks, KNN Classifer. Subsequently, among all these classifiers, the model that performs the best is assessed with the greatest accuracy.

Target variable: Multiclass classification(Good, Moderate, Poor, Hazardous)

5.2.1 Support Vector Machine(SVM)

Support Vector Machine (SVM) is a supervised classification algorithm that aims to find the optimal hyperplane which best separates data points of different classes by maximizing

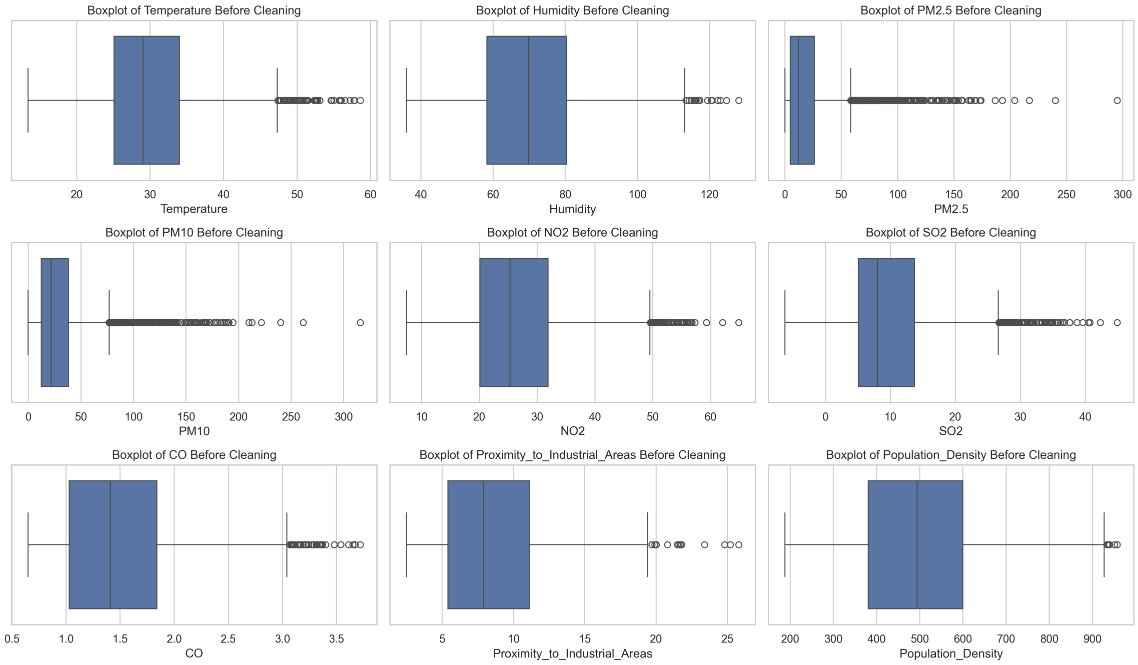


Figure 5.3: box plots of different features before handling the outliers

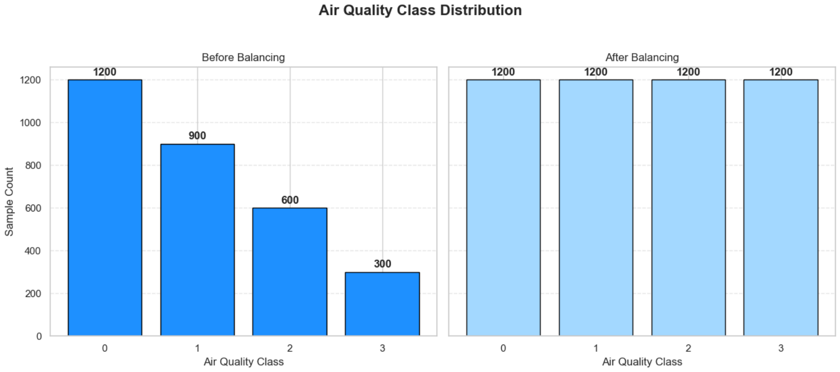


Figure 5.4: Air Quality Distribution

the margin between them. It is particularly effective for both linear and non-linear classification tasks, depending on the kernel function applied.

In this project, we experimented with three types of kernels:

- **Linear Kernel**: Implemented with C=1.0, demonstrating robust performance for linearly separable components of the data
- **Polynomial Kernel**: Captures non-linear relationships through polynomial interactions between features. It offered more flexibility but was computationally more expensive.
  Utilized with degree=5 and C=1000.0, capturing complex poly nomial relationships
- **RBF (Radial Basis Function) Kernel**: Projects data into higher-dimensional space using a Gaussian function. It provided the best performance by effectively handling non-linear class boundaries.
  Configured with C=1.0 and γ=0.1, effectively mapping features to a higher-dimensional space

Model	Accuracy	Precision	Recall	F1-score
SVM (Linear Kernel)	0.938	0.938110	0.938	0.938023
SVM (Polynomial)	0.907	0.905042	0.907	0.905478
SVM (RBF Kernel)	0.9410	0.9411	0.9410	0.9410

Table 5.1: Performance Comparison of Different kernels

Observation:

- **Parameter Sensitivity:** Higher values of γ led to overfitting, while lower values resulted in reduced model capacity. The selected parameters balanced model complexity and generalization.
- **Consistency Across Metrics:** For all three models, the precision, recall, and F1-score values are very close to the overall accuracy. This implies balanced class performance, meaning no single class is dominating the evaluation.
- **Linear Kernel Performs Similarly to RBF:** Suggests that the dataset may be close to linearly separable, which allows the linear kernel to perform strongly.
- SVM with the RBF kernel among the other kernels showed the best classification performance in this project, especially after applying feature normalization.

5.2.2 Logistic Regression

Logistic regression is a key player in supervised machine learning, particularly suited for classifcation tasks. It transforms outcomes into binary decisions, efectively drawing a decision boundary that separates data points into distinct categories. It achieves this using a logistic function that strikes a balance between minimizing false positives and false negatives, making it a valuable tool for classifying data.

Model	Accuracy	Precision	Recall	F1-score
Logistic Regression	0.936	0.936520	0.936	0.936166

Table 5.2: Performance Metrics of Logistic Regression

Observation:

- **Performance Consistency:** Logistic Regression demonstrated nearly 94% accuracy with optimal maximum number of iterations as 100.
- **Competitive with SVM:** Its performance is very close to SVM with a linear kernel (accuracy 0.938), reinforcing that the data may be linearly separable or nearly so.
- **Efficient and Interpretable:** Logistic Regression not only performs well but also offers simplicity, speed, and interpretability, making it a strong baseline model.

5.2.3 Neural Networks

A basic feedforward neural network was implemented using the Multi-Layer Perceptron (MLP) classifier. It consists of an input layer, one or more hidden layers, and an output layer. Each neuron performs a weighted sum of inputs followed by a non-linear activation function to capture complex feature interactions.

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
Neural Networks	0.940	0.939567	0.940	0.939711

Table 5.3: Performance Metrics of Neural Networks

- Hyperparameters such as learning rate, number of hidden layers, and batch size were tuned to optimize performance. To monitor training progress, **Loss vs. Epochs** graphs were plotted, showing how the loss decreased over time as the model learned.

Observation:
The neural network performed comparably to SVM and Decision Trees, but slightly below Random Forest in terms of accuracy and F1-score.

- **Effective Hidden Layer Configuration:** The deep architecture with layers [512, 256, 128] contributed to high accuracy without overfitting on validation data.
  - This structure likely provided sufficient capacity to learn intricate patterns while maintaining generalization.
- **Outperforms Simpler Models:** Slightly edges out SVM and Logistic Regression, highlighting the advantage of deep learning when ample data and features are available.

5.2.4 K-Nearest Neighbors(KNN)

K-Nearest Neighbors (KNN) stands as a straight forward yet effective instance-based learning method in predictive modeling. It operates by fnding the k-nearest data points in the training set, determined through distance metrics like Euclidean or Mahalabonis distance. This local neighborhood approach allows k-NN to make predictions for classification or regression tasks, drawing upon the class labels or values of nearby points.

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
KNN	0.920	0.919905	0.920	0.919819

Table 5.4: Performance Metrics of KNN

- During hyperparameter tuning, the performance of KNN was evaluated for various values of k: 3, 5, 7, 11, 13, 17 .The primary evaluation criterion was the Accuracy of the model

Observation:

- **Optimal K :** At k = 3, the model achieved the highest accuracy, suggesting this is the optimal value for k

- **Simple Yet Effective:** Despite being computationally intensive (especially for large datasets), KNN provides good accuracy and is easy to interpret.
- **Possibility of Noise Sensitivity:** KNN is sensitive to **outliers** and noisy data because it looks at the k nearest neighbors for classification. Even a small amount of noise can impact predictions.

The hyperparameter tuning process underscores the importance of carefully selecting k to balance model accuracy and generalization.

5.2.5 Bayes Classifier

Bayes Classification is a probabilistic classifier based on Baye's Theorem. It predicts the class of a given data point by calculating the posterior probability for each class and picking the most probable one

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
Bayes	0.937	0.937519	0.937	0.937177

Table 5.5: Performance Metrics of Bayes Classifier

- we have observed that the accuracy is nearly 93%, demonstrating that Bayes-based classification can perform well even with relatively simple probabilistic assumptions. This strong performance suggests its potential as an efficient and interpretable model for air quality prediction.

Observation:

- **Balanced Class Handling:** The closeness of precision and recall suggests the classifier is treating all classes fairly, with no significant bias or skew in predictions.
- **Flexible and Interpretable:** Bayesian classifiers can incorporate prior knowledge and update beliefs based on evidence.

5.2.6 Navie Bayes Classifier

The Gaussian Naive Bayes classifier assumes each feature dimension is normally distributed and conditionally independent given the class. It estimates likelihood parameters from the training data and applies Bayes' theorem to compute posterior probabilities. Despite its simplifying assumptions, the classifier efficiently handles the high-dimensional audio embeddings by modeling each feature separately

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
Navie Bayes	0.932	0.932464	0.932	0.932192

Table 5.6: Performance Metrics of Navie Bayes

- Works well with high-dimensional data,fast and efficient

Observation:

- **Effectiveness Despite Simplicity:** Even with its naive assumption, the model performs remarkably well, indicating that the dataset might have features that are reasonably independent or well-separated by class.
- **Slightly Lower Than Full Bayes / SVM / NN:** Compared to models like Bayes (0.937), SVM RBF (0.939), and Neural Networks (0.940), Naive Bayes shows slightly lower performance, which is expected due to its **strong independence assumption.**

5.2.7 Decision Trees

Decision Tree is a non-parametric supervised learning algorithm used for classification and regression tasks. It works by recursively splitting the dataset into subsets based on feature values that best separate the target classes. Each internal node represents a decision on a feature, and each leaf node represents a class label.

Decision Trees are widely used because they:

- Are easy to interpret and visualize,
- Handle both numerical and categorical data,
- Require little preprocessing (e.g., no feature scaling needed).

In this project, Decision Trees were used to classify air quality levels based on pollutant and demographic features. They provided decent performance and highlighted which features (e.g., PM2.5, PM10) were most important in splitting the data. Decision Trees are especially useful when interpretability is important, or when feature interactions are non-linear.

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
Decision Trees	0.925	0.924890	0.925	0.924936

Table 5.7: Performance Metrics of Decision Tree

Observation:

- **Consistent Metrics:** All values are closely aligned (92.5%), indicating balanced classification performance.
- **Parameter Sensitivity:** Decision Trees are intuitive models that split data based on feature thresholds. A smaller tree depth leads to simpler models that may underfit the data, while deeper trees capture more complexity but risk overfitting. Optimal depth balances bias and variance. In our case, limiting the depth helped reduce overfitting and improved generalization with depth = 13.

5.2.8 Random Forest Classifier

Random Forest is an ensemble learning algorithm that builds multiple decision trees on random subsets of the data and features, and aggregates their outputs to improve overall prediction accuracy. This bagging approach helps reduce overfitting and enhances model generalization.

Model	Accuracy	Precision	Recall	F1-score
Random Forest	0.949	0.950459	0.949	0.949365

Table 5.8: Performance Metrics of Random Forest

- Random Forests have several hyperparameters that significantly influence performance. Tuning them helps reduce overfitting, improve generalization, and enhance accuracy.

Observation:
We observed several key points after evaluating the Random Forest classifier models

- **Highest Accuracy So Far:** With 0.949 accuracy, Random Forest outperforms all other models in your list, indicating superior predictive power on the validation set
- **Effective Ensemble Strategy:** The result reflects the strength of ensemble learning: combining multiple decision trees leads to higher accuracy and robustness compared to single models like Decision Trees (0.925).
- **Excellent Precision and F1-Score:** A precision of 0.950 and F1-score of 0.949 show that the model makes accurate and reliable predictions, minimizing both false positives and false negatives.

5.3 Evaluation Metrics

We have evaluated the performance of the models based on how well they have predicted the correct air quality levels by Calculating Accuracy, Precision, Recall and F1-score.

Confusion matrix: The matrix provides a comprehensive breakdown of four key metrics: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). These metrics offer a clear picture of the model's ability to correctly identify and classify instances of positive and negative outcomes.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$\text{Sensitivity (Recall)} = \frac{TP}{TP + FN} \tag{9}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{10}$$

$$\text{F1-score} = 2 \cdot \frac{\text{Sensitivity} \cdot \text{Precision}}{\text{Sensitivity} + \text{Precision}} \tag{11}$$

As we can observe from the above table that Random Forest classifer has comparatively higher Evaluation metric levels.

6 Results and Analysis

6.1 Various Models insights

Logistic Regression: Feature Weights

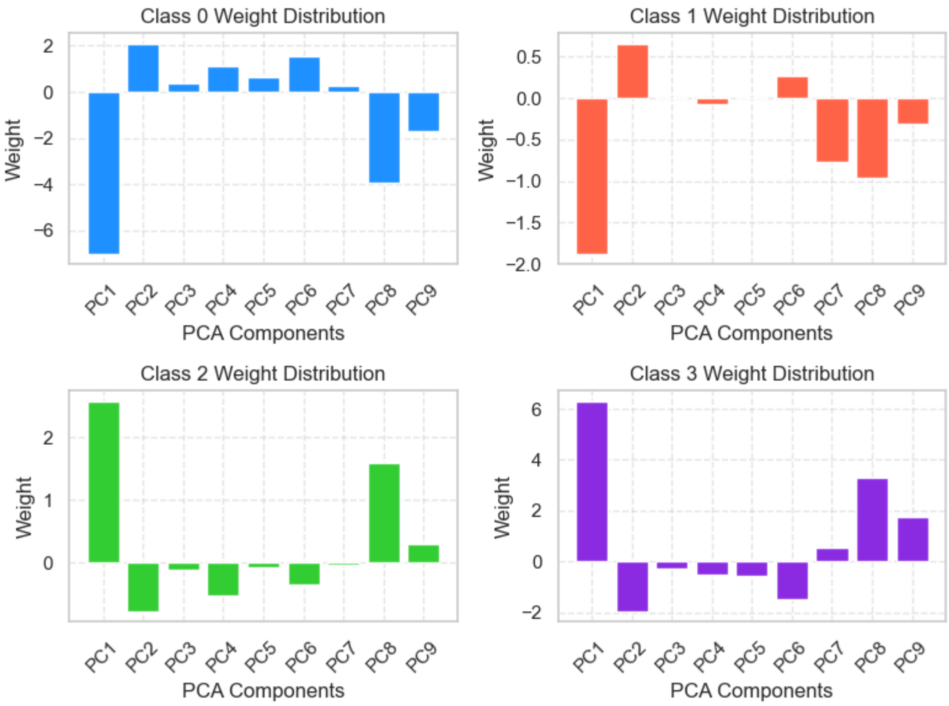


Figure 6.1: Accuracies of various models

Neural Networks : Loss vs Epochs

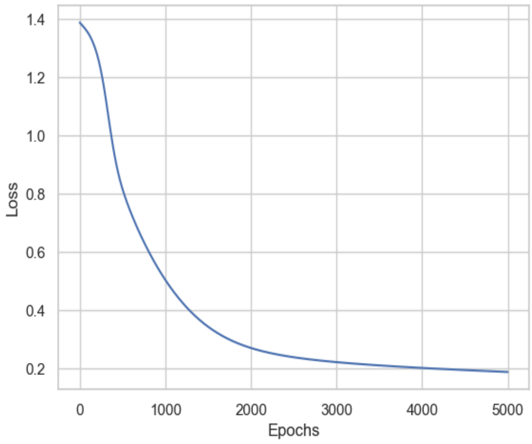


Figure 6.2: For hidden layers [64, 32, 16]

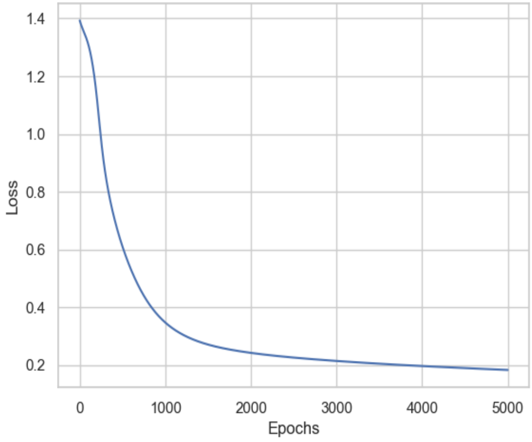


Figure 6.3: For hidden layers [128, 64, 32]

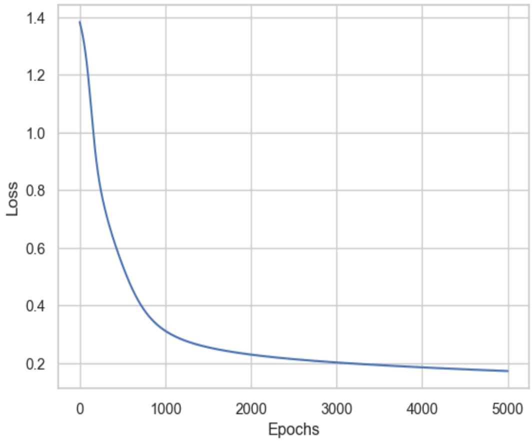


Figure 6.4: For hidden layers [256, 128, 64]

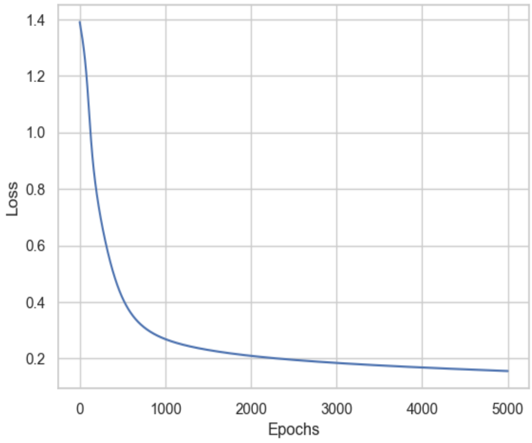
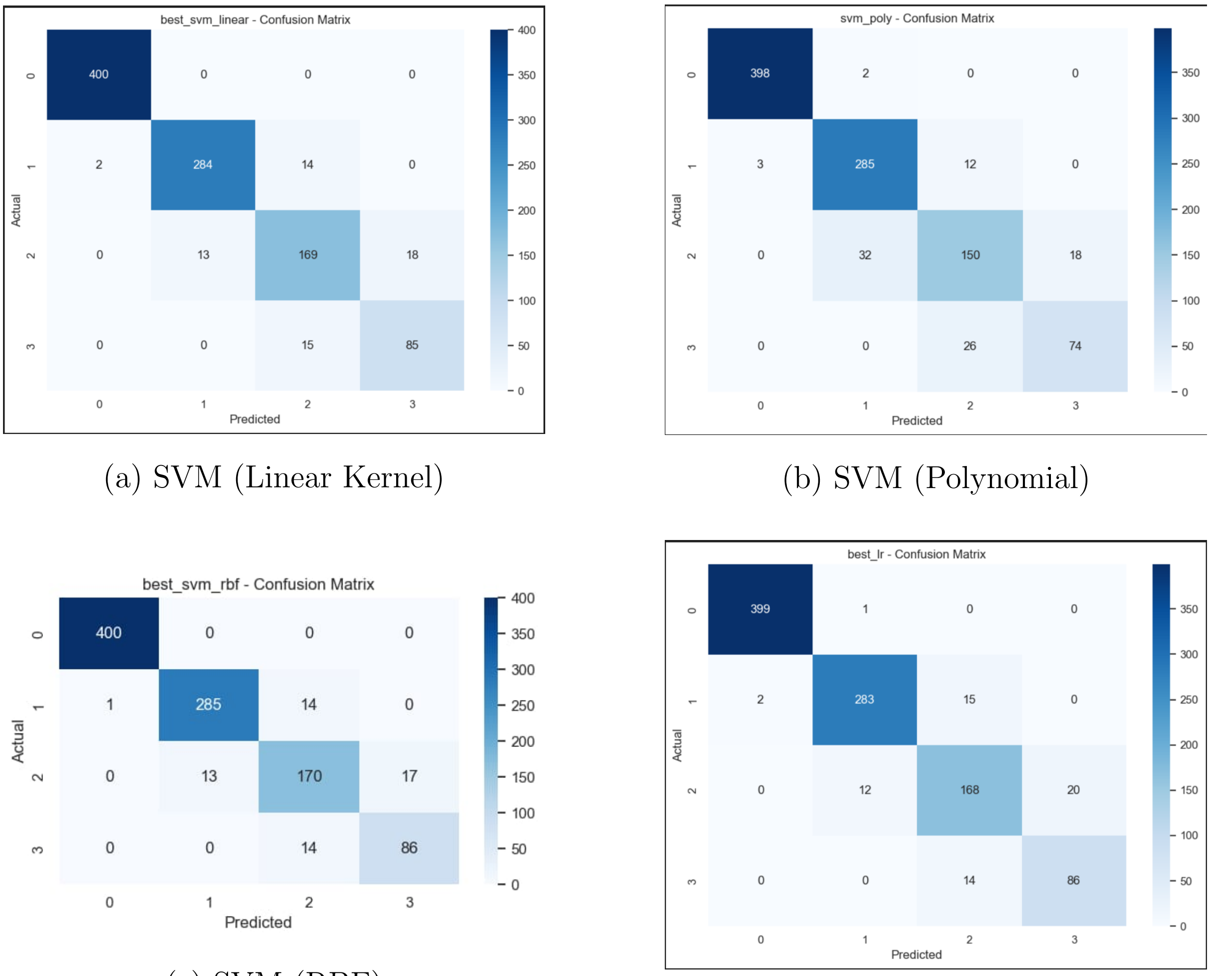


Figure 6.5: For hidden layers [512, 256, 128]

Confusion Matrices for various models:



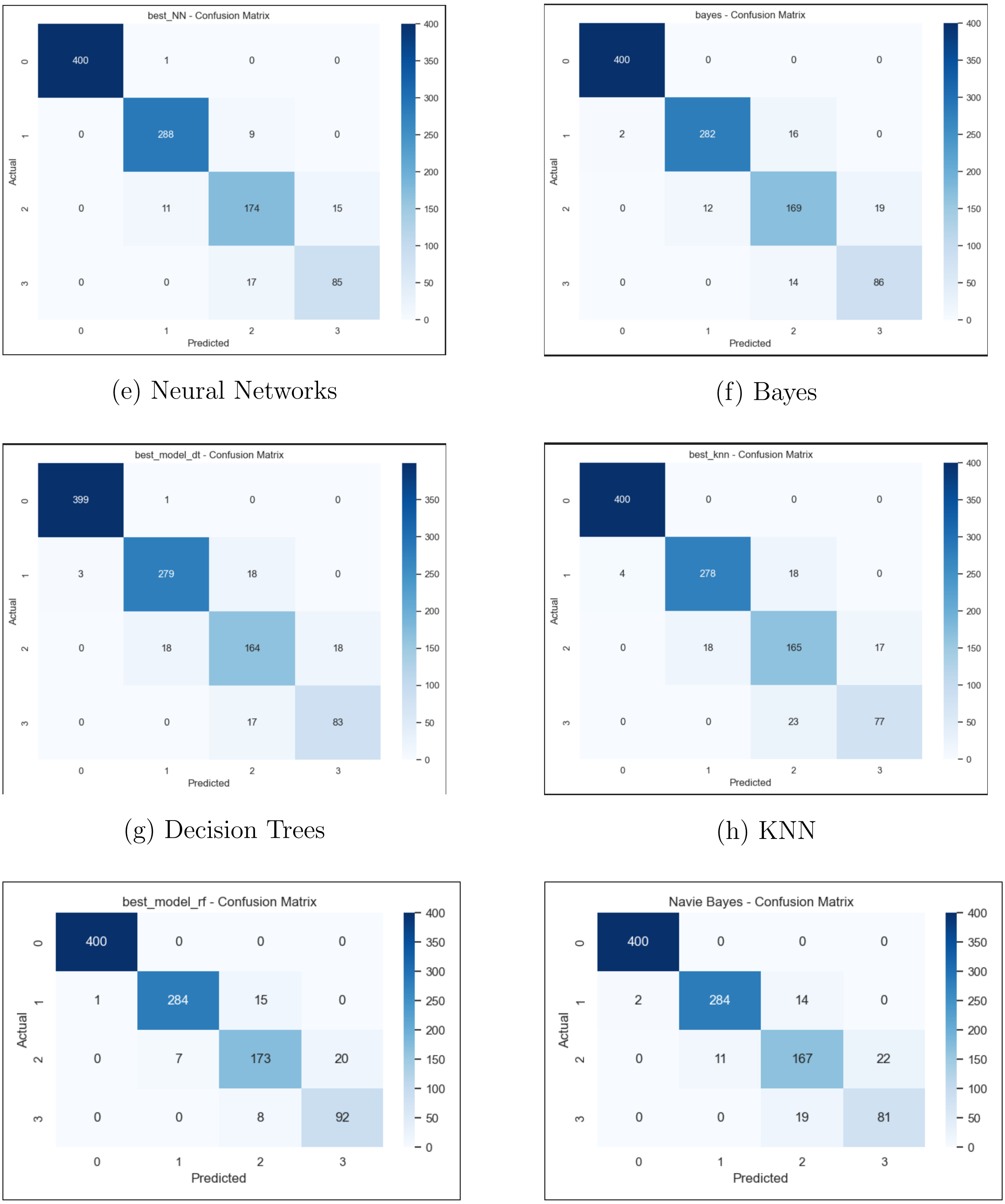
(a) SVM (Linear Kernel)

(b) SVM (Polynomial)

(c) SVM (RBF)

(d) Logistic Regression

Figure 6.6: Model Output Visualizations for Different Classifiers (a)–(d)



(e) Neural Networks

(f) Bayes

(g) Decision Trees

(h) KNN

(i) Random Forests

(j) Naive Bayes

6.2 Identical Accuracy and Recall Values Across Models

- **Balanced Dataset:** dataset is likely well-balanced, meaning all classes have roughly equal representation. In such cases, Accuracy and Recall can often align closely, since the model doesn't benefit from class imbalance.

- **High True Positive Rates:** Models are consistently making correct predictions for each class, so the true positives (used in Recall) dominate the confusion matrix—mirroring overall accuracy.

- **No Major Class Bias:** When there's no significant class imbalance and no model is biased toward specific classes, performance measures like Accuracy, Recall, and F1 tend to converge.

7 Model Results

Here is some information before the images.

7.1 Model Performance Comparison

Model	Accuracy	Precision	Recall	F1-score
SVM (Linear Kernel)	0.938	0.938110	0.938	0.938023
SVM (Polynomial)	0.907	0.905042	0.907	0.905478
SVM (RBF Kernel)	0.9410	0.9411	0.9410	0.9410
Logistic Regression	0.936	0.936520	0.936	0.936166
Neural Network	0.944	0.943550	0.944	0.943707
K-Nearest Neighbors	0.920	0.919905	0.920	0.919819
Bayes	0.937	0.937519	0.937	0.937177
Navie Bayes	0.932	0.932464	0.932	0.932192
Decision Tree	0.925	0.924890	0.925	0.924936
Random Forest	**0.948**	**0.948958**	**0.948**	**0.948170**

Table 7.1: Performance Comparison of Different Classification Models

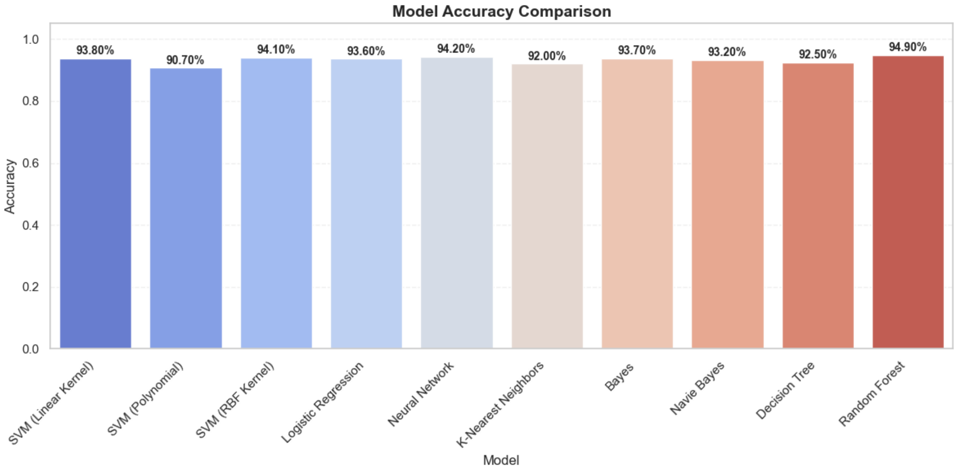


Figure 7.1: accuracies of various models

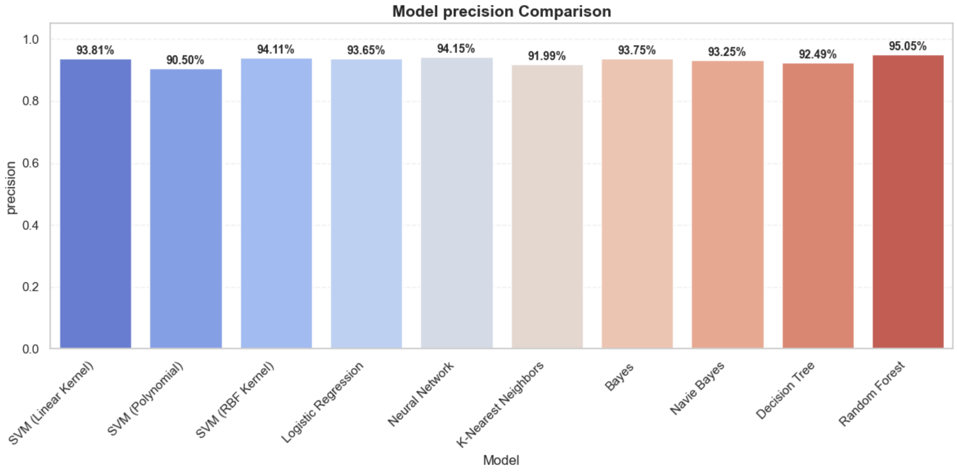


Figure 7.2: precision of various models

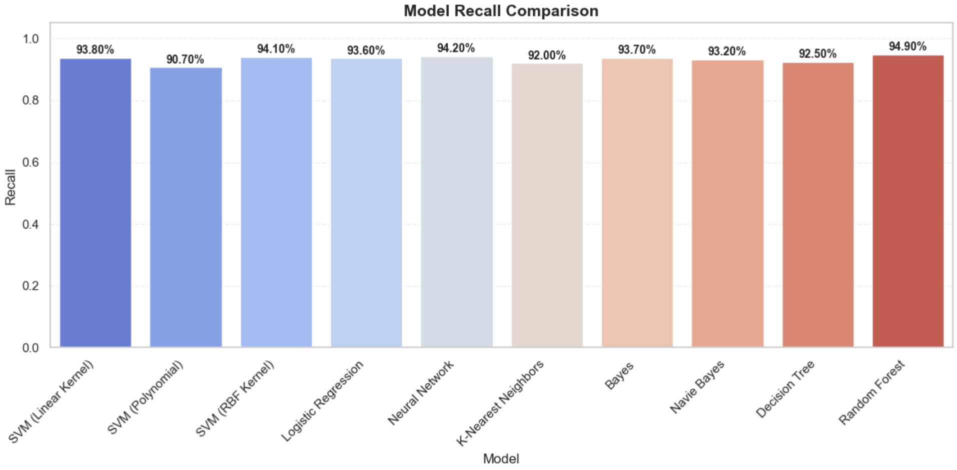


Figure 7.3: Recall of various models

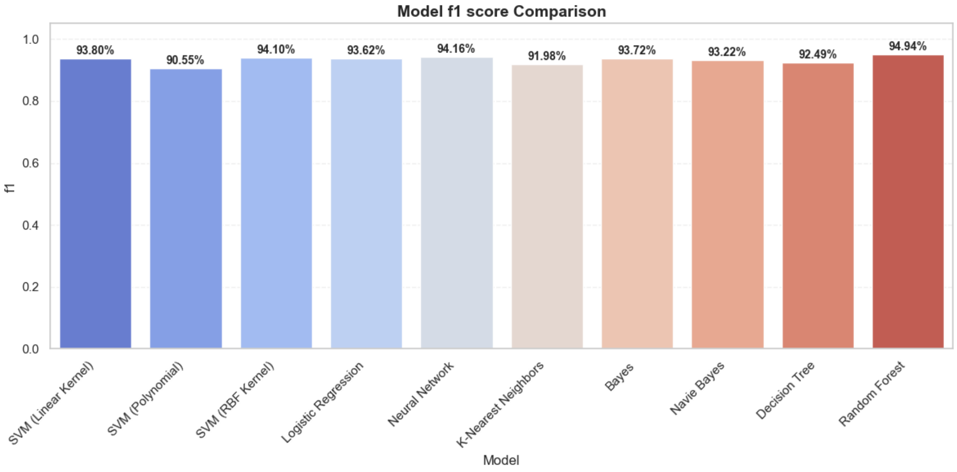


Figure 7.4: f1 score of various models

The performance evaluation of several machine learning models revealed varying levels of effectiveness based on precision, recall, F1-score, and accuracy.

The **Neural Network** (MLP) model followed closely, demonstrating robust performance with slightly lower but still high values in precision, recall, and F1-score. It effectively captured complex patterns in the data, especially after tuning the learning rate and architecture.

The **Support Vector Machine** with the **RBF kernel** also performed exceptionally well, highlighting its strength in handling non-linear boundaries. Among the three SVM variants, the RBF kernel outperformed both the linear and polynomial kernels, confirming the non-linear nature of the feature space.

Other models like **Logistic Regression**, **Naive Bayes**, and **K-Nearest Neighbors** delivered competitive results.The **Decision Tree** model provided decent performance and interpretability but lacked the generalization power of ensemble and deep learning methods.

The results suggest that ensemble methods like Random Forest and non-linear models such as Neural Networks and RBF-SVM are better suited for the air quality classification task due to their ability to handle complex interactions among features.

7.2 Best Model Performing Insight

Overall the **Random Forest** model achieved the highest scores across all metrics, indicating its superior classification performance.
Random Forest's feature importance analysis revealed that CO, Proximity to Industrial Areas, NO_2, and SO_2 were the most influential in determining air quality levels. Interestingly, PM2.5 was found to be the least important and was removed in a refined model. After removal, the model's accuracy increased, confirming the benefit of excluding irrelevant or noisy features.

Performance Metric	**Value**
Accuracy	0.9630
Precision	0.9636
Recall	0.9630
F1-score	0.9632

Table 7.2: Performance metrics of the best model on the test set

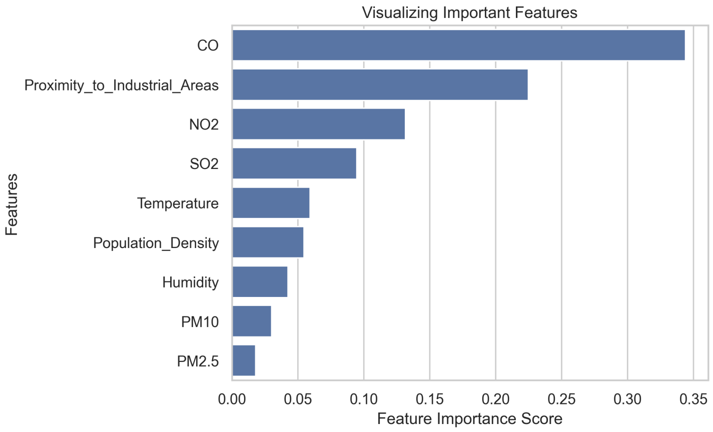
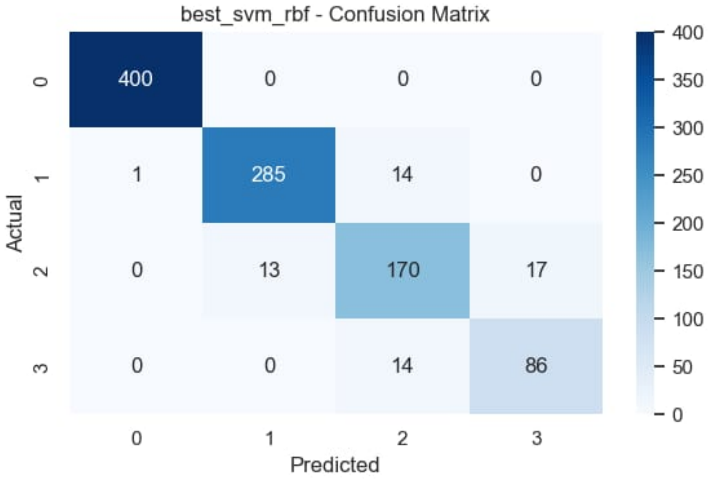


Figure 7.5: Feature Importances

Figure 7.6: Confusion Matrix of Random Forest Classifier



8 Conclusion

In this work, we focus on the early detection of air quality, crucial for safeguarding public health and the environment, with the potential to save millions of lives globally. we introduced an advanced air quality prediction system that integrates various machine learning techniques, including K-Nearest Neighbors, Bayes, Navie Bayes, Random Forest, Support Vector Machine, Logistic Regression, Decision Tree, and SVC achieving a remarkable 96% classification accuracy with Random Forest. The model's limitations include its reliance on historical data, challenges in real-time data integration, and the need for further validation across diverse conditions

In addition, our findings highlight the importance of feature engineering, data preprocessing, and model selection in achieving reliable and interpretable predictions. The high accuracy of the Random Forest classifier underscores its robustness and suitability for complex, nonlinear relationships among pollutant variables. This research sets the foundation for deploying intelligent monitoring systems in urban areas to enable timely alerts and data-driven policy-making.

Future directions may involve incorporating real-time sensor data, improving generalizability with larger datasets from multiple regions, and exploring deep learning models for temporal pattern recognition. By enhancing predictive accuracy and deployment feasibility, such systems can play a pivotal role in environmental management and public health strategies.

9 References

1. Arunkumar, K., et al. (2024). A comparative analysis of machine learning algorithms for air quality index prediction. *Environmental Systems Research.* Available at: SpringerOpen Article

2. DataFlair. Predicting Air Quality Index using Python. Available at: DataFlair Medium Article

3. Breiman, L. (2001). Random Forests. *Machine Learning.* Available at: Springer Article on Random Forest