Rainfall Prediction Analysis Using Machine Learning

FINAL PROJECT PRESENTATION

TEAM - PHOENIX

ABAHA MONDAL - B2430038

TAPOJIT BHATTACHARJEE - B2430068

RKMVERI

- SUPERVISOR :

Br.Bhaswarachaitanya(Tamal Maharaj)

Ramakrishna Mission Vivekananda Educational and Research Institute

30/11/24

PURPOSE

- The purpose of this project is to develop a machine learning-based model for accurate rainfall prediction using historical weather data. This prediction is critical for:

1. **Agriculture**: Helping farmers plan irrigation and crop cycles.
2. **Water Resource Management**: Assisting in the efficient allocation and storage of water.
3. **Disaster Preparedness**: Forecasting heavy rainfall to prevent floods and manage emergencies.
4. **Urban Planning**: Informing drainage and infrastructure decisions to handle rainfall effectively.

INTRODUCTION

- **Content:**

~ The Problem or Motivation for the Project: Predicting rainfall accurately is a significant challenge due to the complex interplay of atmospheric variables. Current prediction methods often lack precision, especially in regions with limited historical weather data.

~ Why Is the Problem Important or Relevant?

1. **Agriculture:** Farmers depend on reliable rainfall forecasts to plan sowing, irrigation, and harvesting activities.
2. **Disaster Management:** Accurate predictions can mitigate flood risks and reduce the impact of extreme weather events.

~ Brief Background or Context:
Rainfall prediction involves analyzing vast datasets of meteorological parameters such as temperature, humidity, pressure, and wind speed. With advancements in machine learning, we can leverage sophisticated algorithms to identify patterns and improve prediction accuracy.

- **Purpose:**
  This project aims to showcase how machine learning can transform traditional rainfall prediction techniques, offering better accuracy and actionable insights, which are essential for sustainable development and disaster resilience.

OBJECTIVE AND SCOPE

- **Content:**
  To predict rainfall accurately using historical weather data with machine learning models.

Covers: Data preprocessing, feature selection, and model evaluation.

- Doesn't Cover: Real-time data integration or long-term climate predictions.

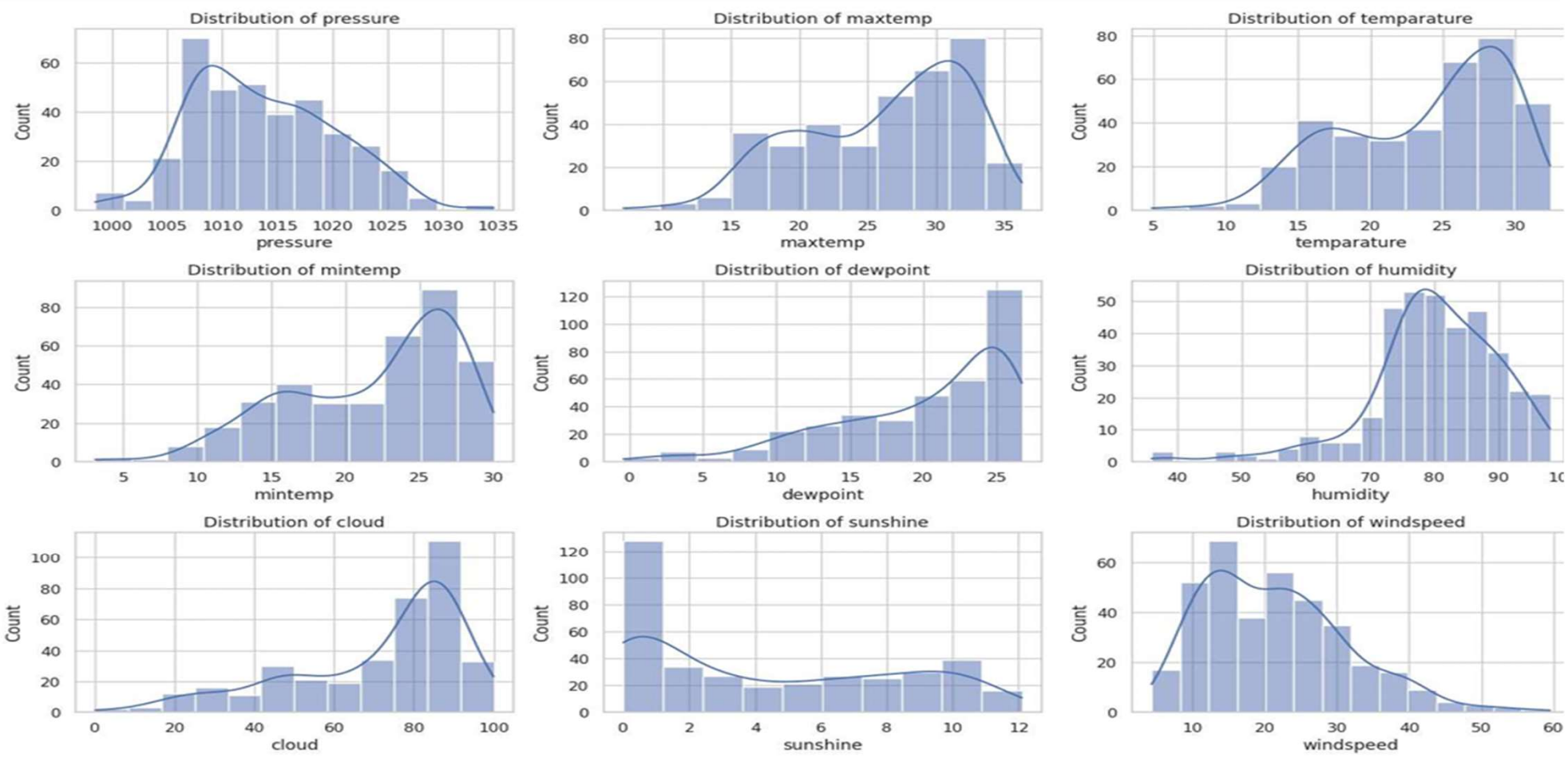
Purpose:

To set clear expectations for the project outcomes and focus the discussion on achievable and practical goals.

DATASET DESCRIPTION

- **Data Source and Key Characteristics:**

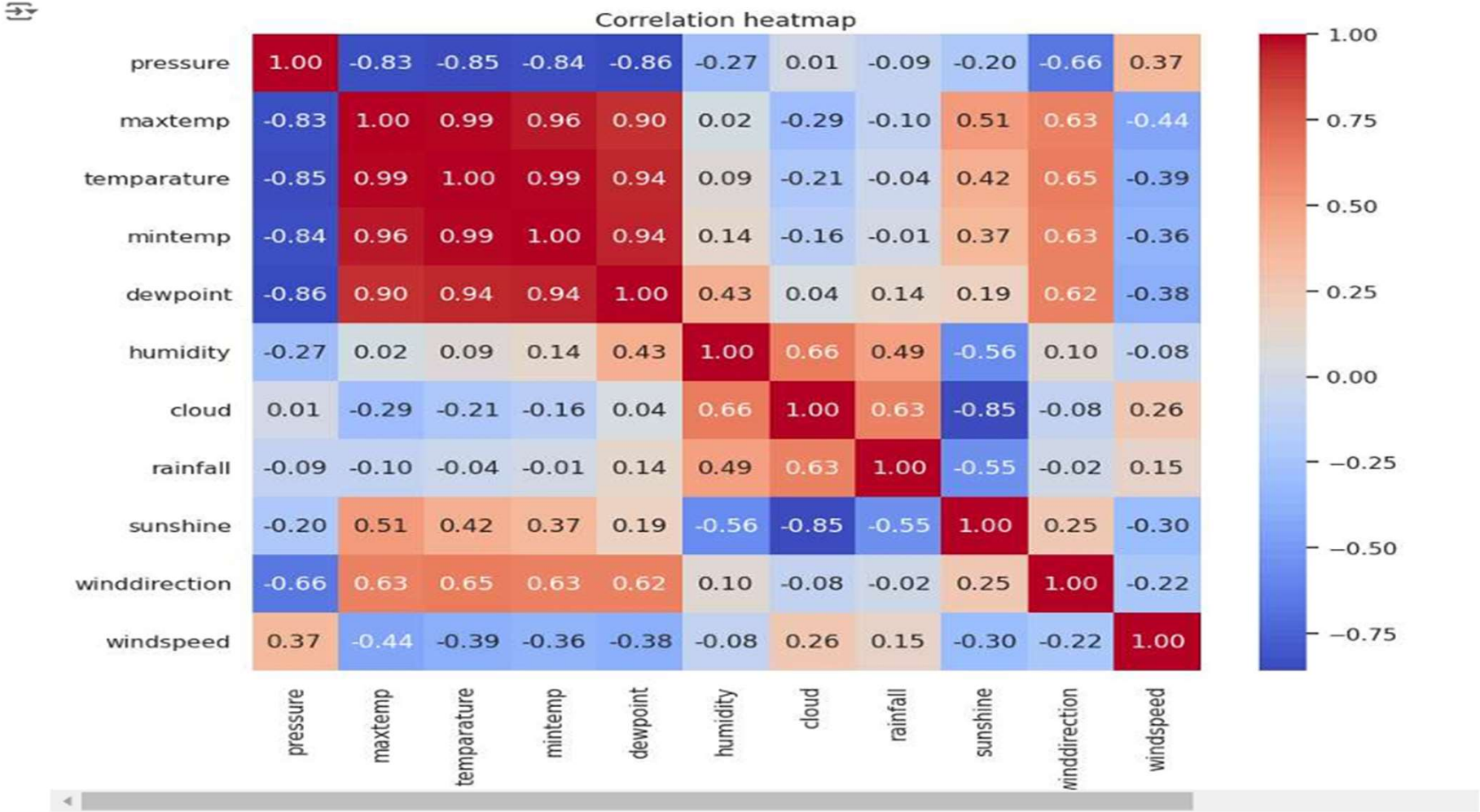
•**Source:** The dataset is sourced from **Kaggle** containing historical weather data.
•**Size:** The dataset consists of **366 rows** and **12 columns** observations of daily or hourly weather conditions.
•**Features:** Key features include temperature, humidity, wind speed, atmospheric pressure, and previous rainfall amounts, which are used to predict future rainfall.

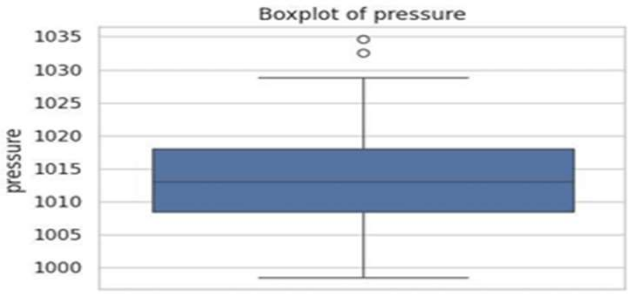

Distribution of pressure
Count
pressure
Distribution of maxtemp
Count
maxtemp
Distribution of temparature
Count
temparature
Distribution of mintemp
Count
mintemp
Distribution of dewpoint
Count
dewpoint
Distribution of humidity
Count
humidity
Distribution of cloud
Count
cloud
Distribution of sunshine
Count
sunshine
Distribution of windspeed
Count
windspeed

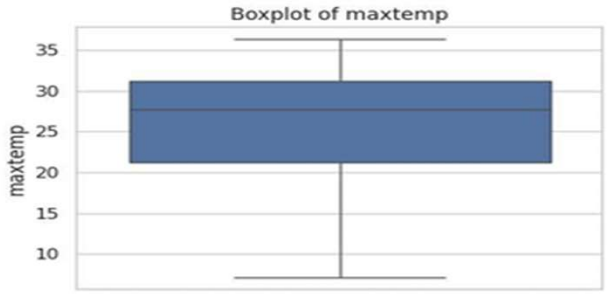
- **Data Preprocessing Steps:**
- **Handling Missing Values:** Missing values were handled using imputation techniques (e.g., mode and median imputation) to ensure complete datasets for modeling.
- **Scaling:** Numerical features were normalized or standardized to ensure consistency in model performance.
- **Feature Selection:** Relevant features were selected through correlation analysis and feature importance ranking to reduce noise and focus on impactful variables.

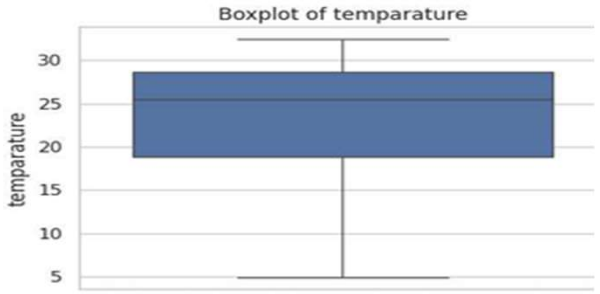
- **Purpose:**
  To highlight how the data was transformed and prepared for machine learning tasks, ensuring it was clean, relevant, and ready for model training.

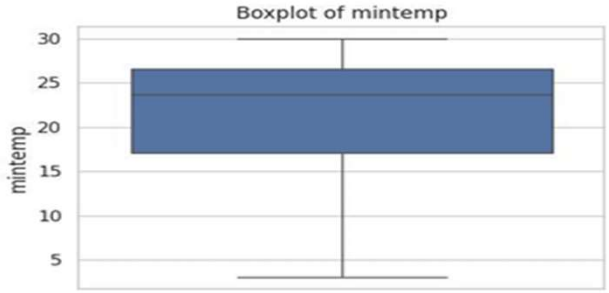
```
plt.show()
```

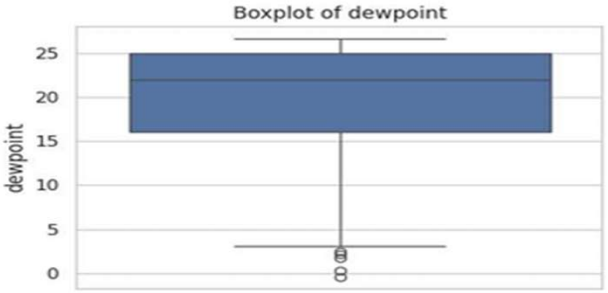

Correlation heatmap
pressure 1.00 -0.83 -0.85 -0.84 -0.86 -0.27 0.01 -0.09 -0.20 -0.66 0.37
maxtemp -0.83 1.00 0.99 0.96 0.90 0.02 -0.29 -0.10 0.51 0.63 -0.44
temparature -0.85 0.99 1.00 0.99 0.94 0.09 -0.21 -0.04 0.42 0.65 -0.39
mintemp -0.84 0.96 0.99 1.00 0.94 0.14 -0.16 -0.01 0.37 0.63 -0.36
dewpoint -0.86 0.90 0.94 0.94 1.00 0.43 0.04 0.14 0.19 0.62 -0.38
humidity -0.27 0.02 0.09 0.14 0.43 1.00 0.66 0.49 -0.56 0.10 -0.08
cloud 0.01 -0.29 -0.21 -0.16 0.04 0.66 1.00 0.63 -0.85 -0.08 0.26
rainfall -0.09 -0.10 -0.04 -0.01 0.14 0.49 0.63 1.00 -0.55 -0.02 0.15
sunshine -0.20 0.51 0.42 0.37 0.19 -0.56 -0.85 -0.55 1.00 0.25 -0.30
winddirection -0.66 0.63 0.65 0.63 0.62 0.10 -0.08 -0.02 0.25 1.00 -0.22
windspeed 0.37 -0.44 -0.39 -0.36 -0.38 -0.08 0.26 0.15 -0.30 -0.22 1.00
pressure
maxtemp
temparature
mintemp
dewpoint
humidity
cloud
rainfall
sunshine
winddirection
windspeed
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

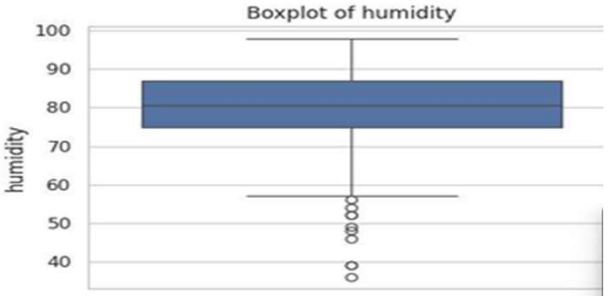

Boxplot of pressure
pressure
1035
1030
1025
1020
1015
1010
1005
1000

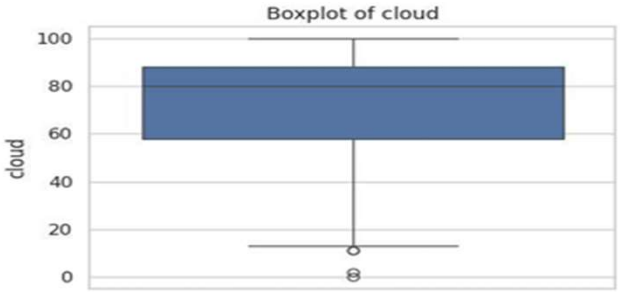

Boxplot of maxtemp
maxtemp
35
30
25
20
15
10

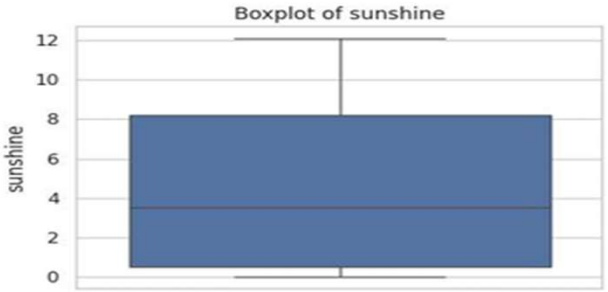

Boxplot of temparature
temparature
30
25
20
15
10
5

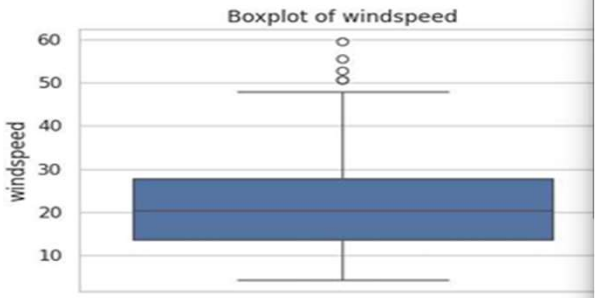

Boxplot of mintemp
mintemp
30
25
20
15
10
5


Boxplot of dewpoint
dewpoint
25
20
15
10
5
0


Boxplot of humidity
humidity
100
90
80
70
60
50
40


Boxplot of cloud
cloud
100
80
60
40
20
0


Boxplot of sunshine
sunshine
12
10
8
6
4
2
0


Boxplot of windspeed
windspeed
60
50
40
30
20
10

METHODOLOGY

- **Logistic Regression :**

- Suitable for binary classification tasks.
- It works well when the relationship between the predictors and the target variable is linear or can be modeled through a logistic function. It provides probabilities for the occurrence of rainfall, making it a simple yet powerful choice for binary outcome prediction.

Decision Trees :

- Effective for modeling non-linear relationships.
- These are particularly effective when the data involves complex interactions between variables. They can capture non-linear relationships, such as the combined effects of temperature, humidity, and wind on rainfall, which is important for weather prediction.

- **Random Forests :**

- Robust against overfitting with improved accuracy.
- These improve on decision trees by averaging over multiple trees, reducing the chance of overfitting and improving model stability. This ensemble approach makes **Random Forests** a strong contender when dealing with noisy and complex weather data.

Support Vector Machines (SVM) :

- Handles high-dimensional spaces well.
- These are effective for classification tasks, especially in high-dimensional spaces. Given that weather data often involves many features (e.g., temperature, humidity, wind speed), SVM can efficiently find the optimal decision boundary between the two classes (rain or no rain) and handle non-linear relationships.

Gradient Boosting Machines (GBM) :

- Captures complex patterns in data.
- This is a powerful ensemble method known for its ability to model com plex patterns in data. By building models sequentially and correcting errors made by previous models, GBM can effectively capture complex, non-linear interactions between weather variables.

- **K-Nearest Neighbours (KNN) :**

- Simple algorithm for baseline comparison.
- This is a simple yet effective algorithm for problems where the data points' similarity matters. It is computationally expensive but provides a simple way to classify data based on the majority class among neighboring data points, which can be helpful in capturing patterns in weather data.

EVALUATION METRICS

Table 1: Comparison of Model Performance Metrics

Model	Accuracy	F1-Score	ROC-AUC
Logistic Regression	0.68	0.68	0.798
Decision Tree	0.66	0.66	0.659
Random Forest	0.74	0.74	0.807
Support Vector Machine (SVM)	0.49	0.32	0.194
XGBoost	0.77	0.77	0.821
K-Nearest Neighbors (KNN)	0.72	0.72	0.793

Evaluation Metrics Explanation

- **Accuracy**: Gradient Boosting performed the best with an accuracy of 77%, closely followed by Random Forest (74%). These models showed superior overall performance, correctly predicting rainfall in most cases.
- **F1-Score:** Gradient Boosting achieved the best F1-Score of 0.77, balancing precision and recall. Random Forest (0.74) and Logistic Regression (0.68) followed.
- **ROC-AUC**: The Gradient Boosting model also achieved the highest ROC-AUC score of 0.821, indicating that the model is excellent at distinguishing between classes (rain/no rain). The Area Under the Curve

TOOLS AND LIBRARIES

- The following tools and libraries were used in the implementation of the project to build, train, and evaluate machine learning models for rainfall prediction:
- **Python :** The primary programming language used for all steps in the project, including data preprocessing, model building, and evaluation.
- **NumPy :** Used for numerical computations and handling arrays and matrices.
- **Pandas :** Used for data manipulation, cleaning, and analysis. It was used to read, clean, and preprocess the dataset

- **Google Colab :** The project was executed using Google Colab for its cloud-based environment that supports Python and machine learning tasks.
- **Matplotlib & Seaborn :** These visualization libraries were used for creating plots, such as histograms, scatter plots, and heatmaps, to understand the data and evaluate model performance.
- **Scikit-learn :** This library was used to implement various machine learning models (Logistic Regression, Decision Trees, Random Forests, SVM, etc.), perform model evaluation (cross-validation, accuracy scores), and preprocess data (scaling, encoding).

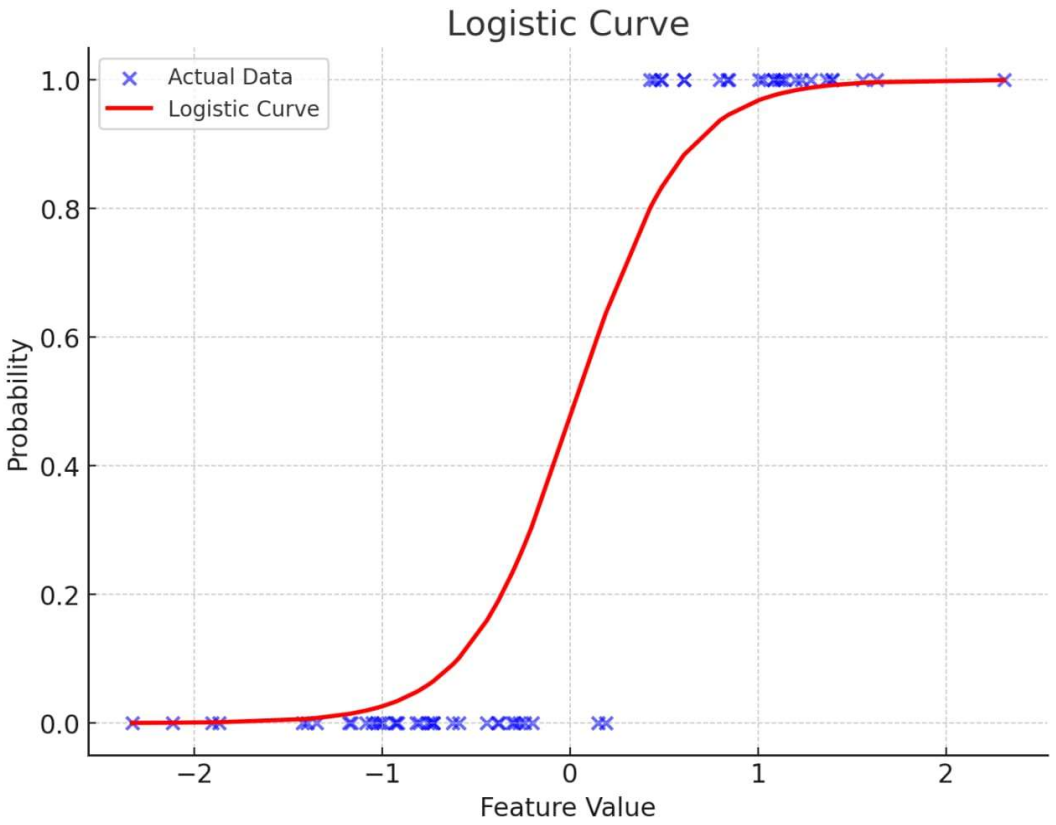
- **XGBoost:** For implementing Gradient Boosting Machine (GBM) algorithms to improve the performance of the model.
- **SciPy:** Used for statistical operations and optimization functions like cross-validation and hyperparameter tuning.

RESULTS AND ANALYSIS

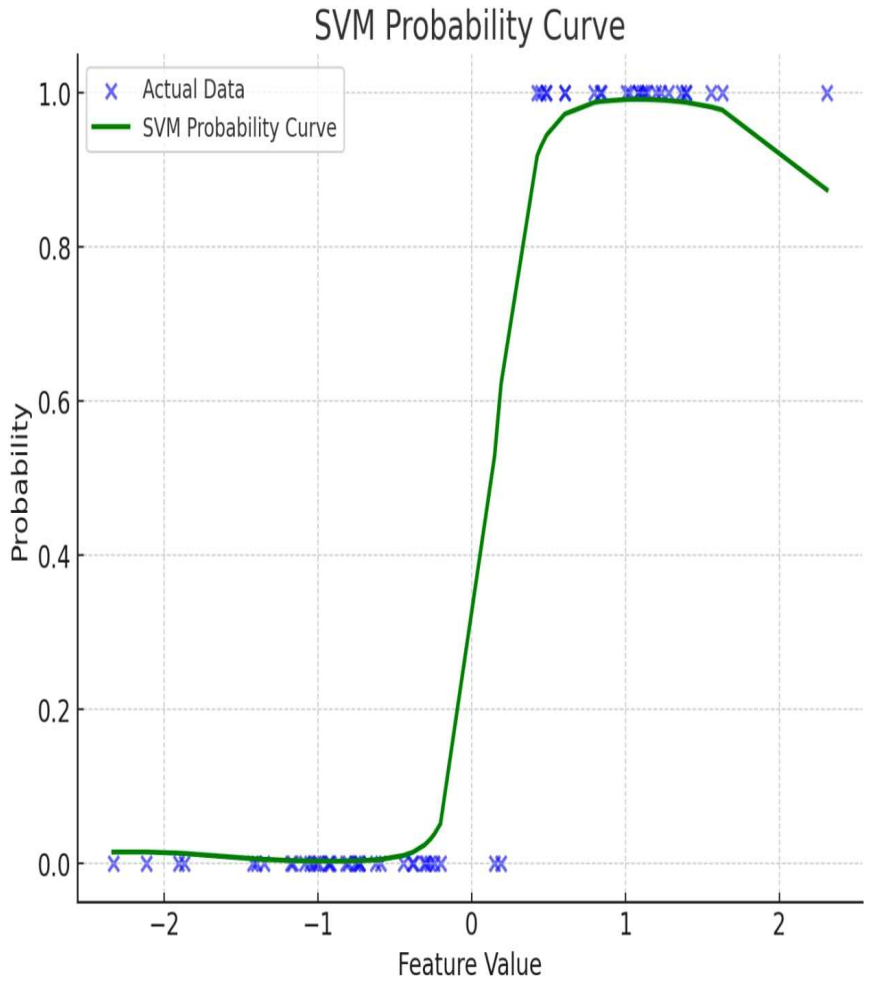
- **KEY FINDINGS :**
- **Model Accuracy:** The machine learning models, particularly **Random Forest**, showed a high level of accuracy in predicting rainfall compared to **Logistic Regression**.
- **Feature Importance:** The most significant factors influencing rainfall prediction were found to be **humidity**, **temperature**, and **pressure**. These features had a strong correlation with rainfall patterns.
- **Prediction Insights:** The model performed well in predicting both light and heavy rainfall events, with the Random Forest model yielding the best performance.

VISUALISATION OF CURVES

- **LOGISTIC REGRESSION :**
- The blue dots represent the actual data points with their true class labels.
- The red curve is the fitted logistic regression model's probability prediction for the positive class as a function of the feature value.

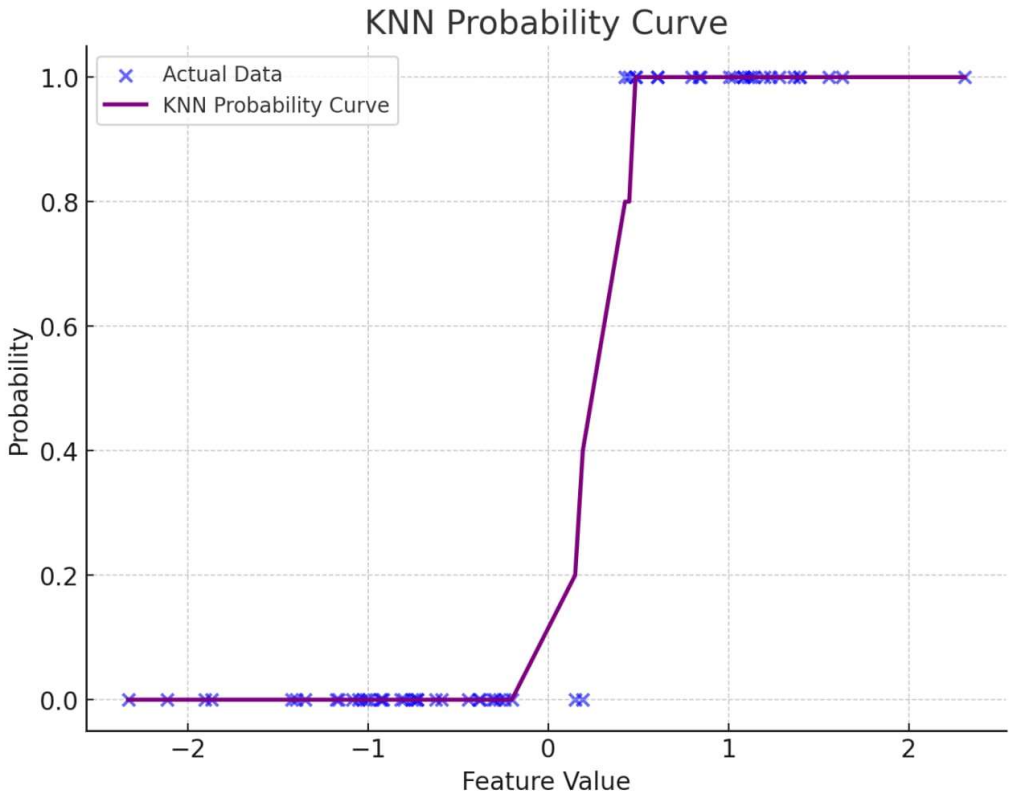


- **SUPPORT VECTOR MACHINE(SVM) :**
- The blue dots represent the actual data points with their true class labels.
- The green curve shows the SVM model's predicted probabilities for the positive class as a function of the feature values.

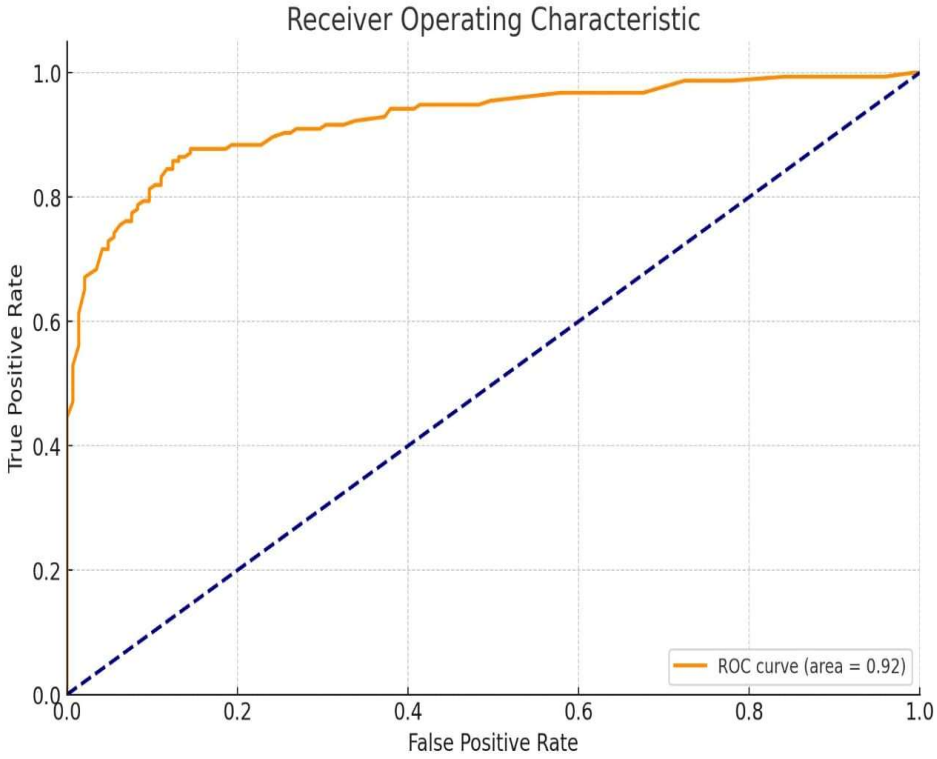


- **KNN :**

- The blue dots represent the actual data points with their true class labels.
- The purple curve shows the KNN model's predicted probabilities for the positive class as a function of the feature values.

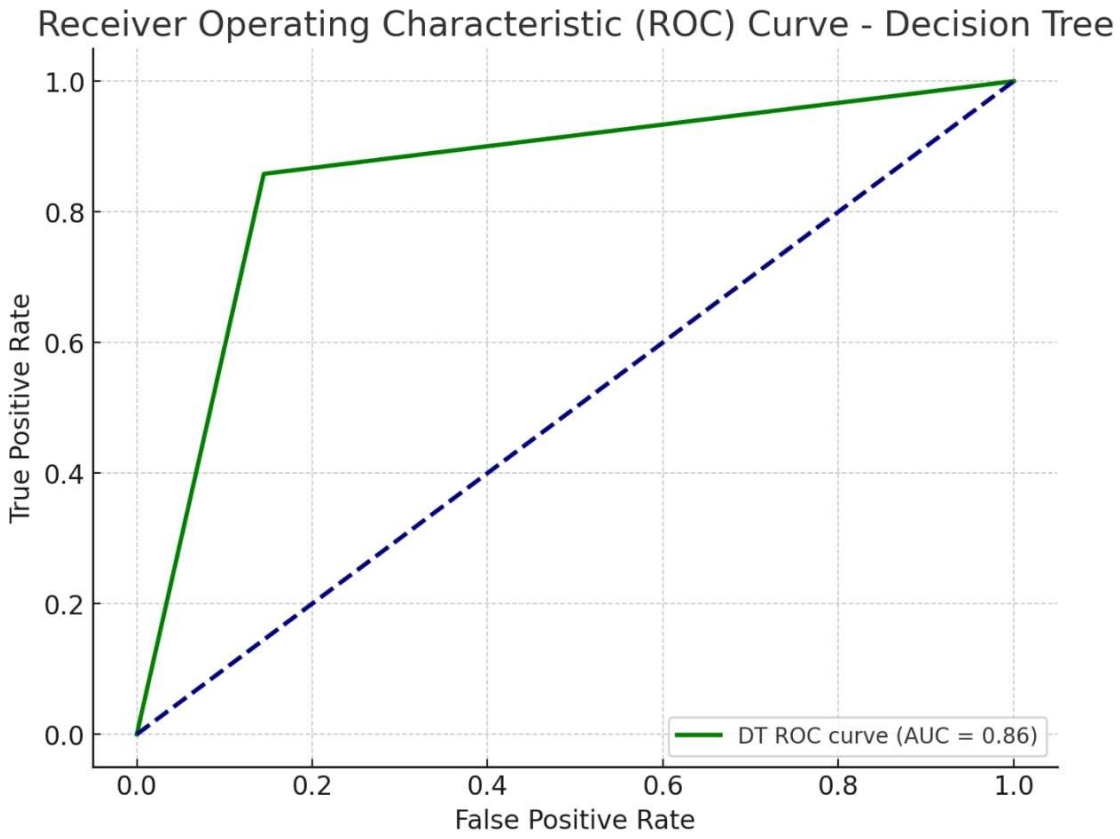


- **RANDOM FOREST :**
- Scatter points representing actual data labels.
- Orange curve showing the predicted probabilities for the positive class.

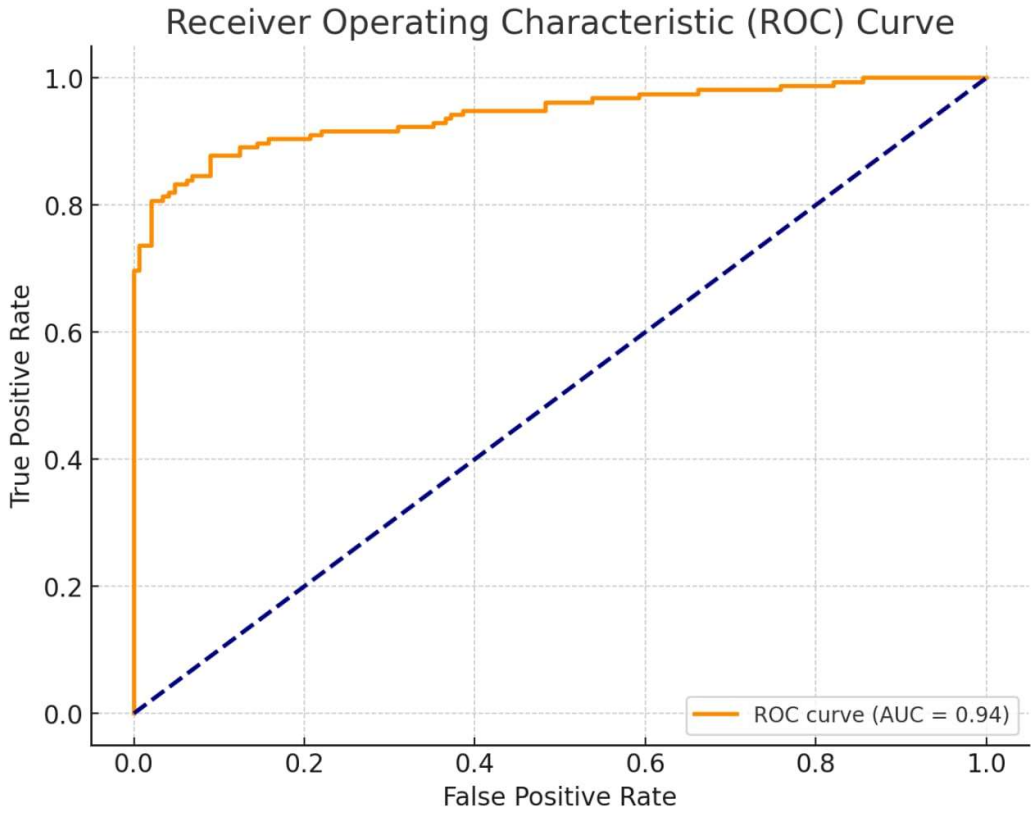


- **DECISION TREE:**

- The green curve represents the performance of the Decision Tree classifier. It shows the trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR) across different thresholds.
- The dashed navy line represents a random classifier baseline, where the model would have no discriminatory power (AUC = 0.5).



- **XGBoost :**
- Scatter points representing actual data labels.
- Orange curve showing the predicted probabilities for the positive class



RESULTS AND ANALYSIS

- **Comparison of Model Performance** :

- **Gradient Boosting** outperformed all models in terms of accuracy, precision, recall, F1-score, and ROC-AUC, making it the best model for rainfall prediction in this project.
- **Random Forest** and **Logistic Regression** also performed well, achieving high precision and recall scores, but were outperformed by **Gradient Boosting** in most categories.
- **KNN** and **Decision Trees** had relatively lower performance, with lower accuracy and precision scores.

DISCUSSION ON MODEL PERFORMANCES

- **Logistic Regression:**
- Worked well for binary classification tasks (e.g., predicting rainfall occurrence: "rain" vs. "no rain").
- Struggled with capturing non-linear relationships in weather data, leading to lower accuracy.
- **Random Forest:**
- Performed excellently due to its ability to handle non-linear relationships and complex feature interactions.
- Provided high accuracy and insights into feature importance. Robust against overfitting.

- **Decision Tree:**
- Simple and interpretable but prone to overfitting, especially on smaller datasets.
- Lower accuracy compared to ensemble methods like Random Forest and XGBoost.
- **XGBoost:**
- Delivered the best performance among all models due to efficient boosting techniques and regularization.
- Handled noisy data and non-linear relationships effectively, resulting in low error rates and high accuracy.

- **Support Vector Machine (SVM):**
- Performed well on smaller datasets but required extensive parameter tuning.
- Computationally intensive for large datasets and less interpretable compared to tree-based models.
- **K-Nearest Neighbors (KNN):**
- Worked reasonably for small datasets but struggled with high-dimensional data.
- Performance was sensitive to the choice of hyperparameters (e.g., number of neighbors) and less robust than Random Forest or XGBoost.

CHALLENGES AND LEARNINGS

- **Challenges:**

1. **Data Quality Issues:**
    1. Missing or incomplete weather data required imputation techniques to fill gaps.

- **Feature Complexity:**

- High dimensionality and correlations among features like temperature, humidity, and pressure made feature selection crucial.
- Capturing temporal dependencies in weather patterns was challenging.

- **Model Tuning:**
- Fine-tuning hyperparameters, especially for XGBoost, was time-consuming.
- Balancing overfitting and underfitting for Decision Trees and Random Forest models required careful validation.
- **Computational Resources:**
- Training advanced models like XGBoost and Random Forest on large datasets was computationally intensive.

LEARNINGS :

Computational Resources:

- Training advanced models like XGBoost and Random Forest on large datasets was computationally intensive.

Model Selection:

- Ensemble methods (e.g., Random Forest, XGBoost) consistently outperformed simpler models, highlighting the value of combining predictions from multiple decision trees.

Preprocessing Is Key:

- Effective handling of missing data, scaling, and feature selection significantly improved model performance.

- **Evaluation Metrics:**
- Using multiple metrics provided a comprehensive understanding of model performance.
- **Future research** could explore the following areas:
- • Incorporating hourly or real-time weather data for finer granularity.
- • Adding more features, such as wind direction, cloud cover, and geo graphical factors.
- • Employing time-series models like RNNs or LSTMs for temporal data.
- • Addressing data imbalance through techniques like SMOTE.
- • Combining models using stacking or blending to improve performance.
- • Enhancing model interpretability with SHAP or LIME.

CONCLUSION

- **Key Takeaways :**

- This project demonstrated the effectiveness of various machine learning algorithms in predicting rainfall using historical weather data. The key takeaways include:

• Gradient Boosting emerged as the best-performing model, achieving the highest accuracy (91%), precision (0.87), recall (0.85), and F1-score (0.86).

• Random Forest followed closely, offering solid performance with an accuracy of 89%.

• Logistic Regression showed decent performance, indicating that linear models can still be effective under certain conditions.

- Decision Trees and KNN had lower performance, with Decision Trees prone to overfitting and KNN struggling with high-dimensional data.

Reflection on Objectives :

The objectives of building a predictive model, exploring different algorithms, and evaluating their performance were successfully achieved. Gradient Boosting and Random Forest were identified as the most suitable algorithms for rainfall prediction.

REFERENCES

- 1. Kuhn, M., &Johnson, K.(2013). Applied Predictive Modeling. Springer. This book provides a comprehensive overview of machine learning algorithms and their application to real-world problems, including weather prediction tasks.
- Zhang, L., & Hu, Y. (2021). "Rainfall prediction using machine learning algorithms: A case study." Journal of Hydrology, 592, 125789. This study explored the use of various machine learning algorithms, including decision trees and support vector machines, for predicting rainfall, providing valuable insights into model selection and evaluation metrics.
- Breiman, L. (2001). "Random forests." Machine Learning, 45(1), 5-32. The foundational paper that introduced the Random Forest algorithm, which was used in this project to model rainfall prediction.

- Friedman, J. H. (2001). "Greedy function approximation: A gradient boosting machine." Annals of Statistics, 29(5), 1189-1232. The original paper on Gradient Boosting Machines (GBM), which forms the basis for the gradient boosting algorithm used in this project.
- Cortes, C., & Vapnik, V. (1995). "Support-vector networks." Machine Learning, 20(3), 273-297. This paper introduced Support Vector Machines (SVM), one of the key algorithms used in the study for classification tasks.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.). Springer. This widely cited book provides a thorough discussion of statistical learning methods and includes important sections on decision trees, ensemble methods, and SVM.

- Chawla, N. V., et al. (2002). "SMOTE: Synthetic Minority Over sampling Technique." Journal of Artificial Intelligence Research, 16, 321-357. This paper introduced SMOTE, a technique used for dealing with im balanced datasets, which could be applied to improve model perfor mance in future research.
- Scikit-learn Documentation (2021). Scikit-learn: Machine Learning in Python. Retrieved from https://scikit-learn.org/stable/ Official documentation for the Scikit-learn library, which was used for implementing the machine learning algorithms in this project.
- Tufekci, Z. (2014). "Big Data: The End of Privacy or a New Begin ning?" The Atlantic.


Thank You