Assessment Report

on

“Rainfall Prediction”

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE(AIML)

Section - A

By

Name: Aman Yadav

Roll Number: 202401100400029

Name: Akshat Gupta

Roll Number: 202401100400023

Name: Arpit Tyagi

Roll No: 202401100400050

Name: Anurag Singh

Roll Number: 202401100400041

Name: Ayush Raj

Roll Number: 202401300400001

Under the supervision of

"Mr. Bikki Sir"

KIET Group of Institutions, Ghaziabad

May, 2025

---

1. Introduction

Rainfall prediction is a crucial aspect of weather forecasting that helps in planning and managing various human activities, including agriculture, water resource management, and disaster preparedness. It involves the use of scientific methods and technologies to estimate the amount and timing of rainfall in a specific area. Accurate rainfall prediction can help minimize the impact of floods and droughts, making it essential for both rural and urban communities. Traditional methods rely on historical weather patterns, while modern approaches incorporate satellite data, radar systems, and advanced computational models. With the rise of machine learning and artificial intelligence, rainfall prediction has become more precise and reliable. However, due to the complex nature of atmospheric conditions, it still presents significant challenges. Continuous improvements in data collection and modeling techniques are vital for enhancing the accuracy of rainfall forecasts.

---

2. Problem Statement

Build a model to predict whether it will rain tomorrow using classification algorithms and weather data.

Accurate rainfall prediction is critical for effective water resource management, agriculture planning, and disaster prevention. In this project, we aim to develop a machine learning model that predicts whether it will rain the next day (`RainTomorrow`) based on various meteorological features. Using the historical weather dataset `weatherAUS.csv`, the objective is to preprocess the data, handle missing values, encode categorical variables, and build a logistic regression model to classify future rainfall events.

---

3. **Objectives**

Data Preprocessing:
To clean and prepare the dataset by handling missing values and encoding categorical variables for machine learning.

Feature Selection and Engineering:
To identify and process the most relevant meteorological features that influence the likelihood of rainfall.

Model Development:
To build and train a logistic regression model to predict whether it will rain the next day (RainTomorrow).

Model Evaluation:
To assess the model's performance using metrics such as accuracy, confusion matrix, and classification report.

Insight Generation:
To analyze feature importance and model outputs to gain insights into the key factors influencing rainfall prediction.

4. Methodology

- **Data Collection**:
  The dataset weatherAUS.csv, which contains historical weather observations across various Australian locations, was used for this study.

- **Missing Values Handling**:
  Columns with more than 30% missing values were removed. Remaining missing values were handled by dropping affected rows.

- **Categorical Encoding**:
  Binary categorical features (RainToday, RainTomorrow) were label encoded, and all other categorical variables were one-hot encoded.

- **Feature Selection**:
  All relevant meteorological features were retained, excluding the target variable (RainTomorrow). This ensured the model was trained on independent variables.

- **Train-Test Split**:
  The dataset was split into training and testing sets in an 80:20 ratio to evaluate the model's performance on unseen data.

- **Feature Scaling**:
  Features were standardized using StandardScaler to normalize the data, which improves the performance of the logistic regression model.

- **Model Training**:
  A logistic regression model was trained on the scaled training dataset to classify whether it will rain the next day.

- **Model Evaluation**:

  Predictions were made on the test set.

- Performance was evaluated using the **confusion matrix**, **accuracy score**, and **classification report** (precision, recall, F1-score).

---

5. Data Preprocessing

- **Loading the Dataset**:
  The dataset weatherAUS.csv was loaded using the Pandas library to facilitate analysis and manipulation.

- **Handling Missing Values**:

  Columns with more than **30% missing values** were dropped to reduce noise and avoid unreliable imputation.

  After column removal, remaining rows with missing data were dropped to ensure a clean and complete dataset.

- **Encoding Categorical Variables**:

  The binary categorical columns RainToday and RainTomorrow were label encoded using Yes → 1 and No → 0.

  All other categorical features (e.g., Location, WindGustDir) were converted to numerical form using **one-hot encoding**, ensuring the model could process them.

- **Feature and Target Separation**:

  The target variable RainTomorrow was separated from the dataset.

  All other columns were treated as input features.

---

6. Model Implementation

The logistic regression model was selected for binary classification of rainfall prediction. After standardizing the features, the model was trained on the training data using LogisticRegression. Predictions were made on the test set to evaluate performance. The model's accuracy, confusion matrix, and classification report were used for evaluation. This approach provided insights into the likelihood of rainfall occurring the next day.

---

7. Evaluation Metrics

- **Confusion Matrix**:
  It shows the number of correct and incorrect predictions classified into True Positives, True Negatives, False Positives, and False Negatives, helping to understand the model's performance.
- **Accuracy Score**:
  Measures the overall correctness of the model by calculating the ratio of correctly predicted instances to total predictions.
- **Classification Report**:
  Provides detailed metrics including **precision**, **recall**, and **F1-score** for each class (`RainTomorrow = Yes` or `No`), offering insights into the model's strengths and weaknesses.

---

8. Results and Analysis

- The logistic regression model achieved a reasonable **accuracy score**, indicating good overall performance in predicting whether it will rain tomorrow.
- The **confusion matrix** revealed the number of correct and incorrect predictions, helping to understand the model's ability to distinguish between rainy and non-rainy days.

- The **classification report** showed that the model performed better on the majority class (usually “No Rain”) due to possible class imbalance in the dataset.
- **Precision and recall** values for the "Yes" class (rain) were lower, suggesting the model may miss some rainy days, which could be critical in practical scenarios.
- Overall, while logistic regression provided a solid baseline, the analysis suggests potential for improvement by handling class imbalance or using more complex models like Random Forest or XGBoost.

---

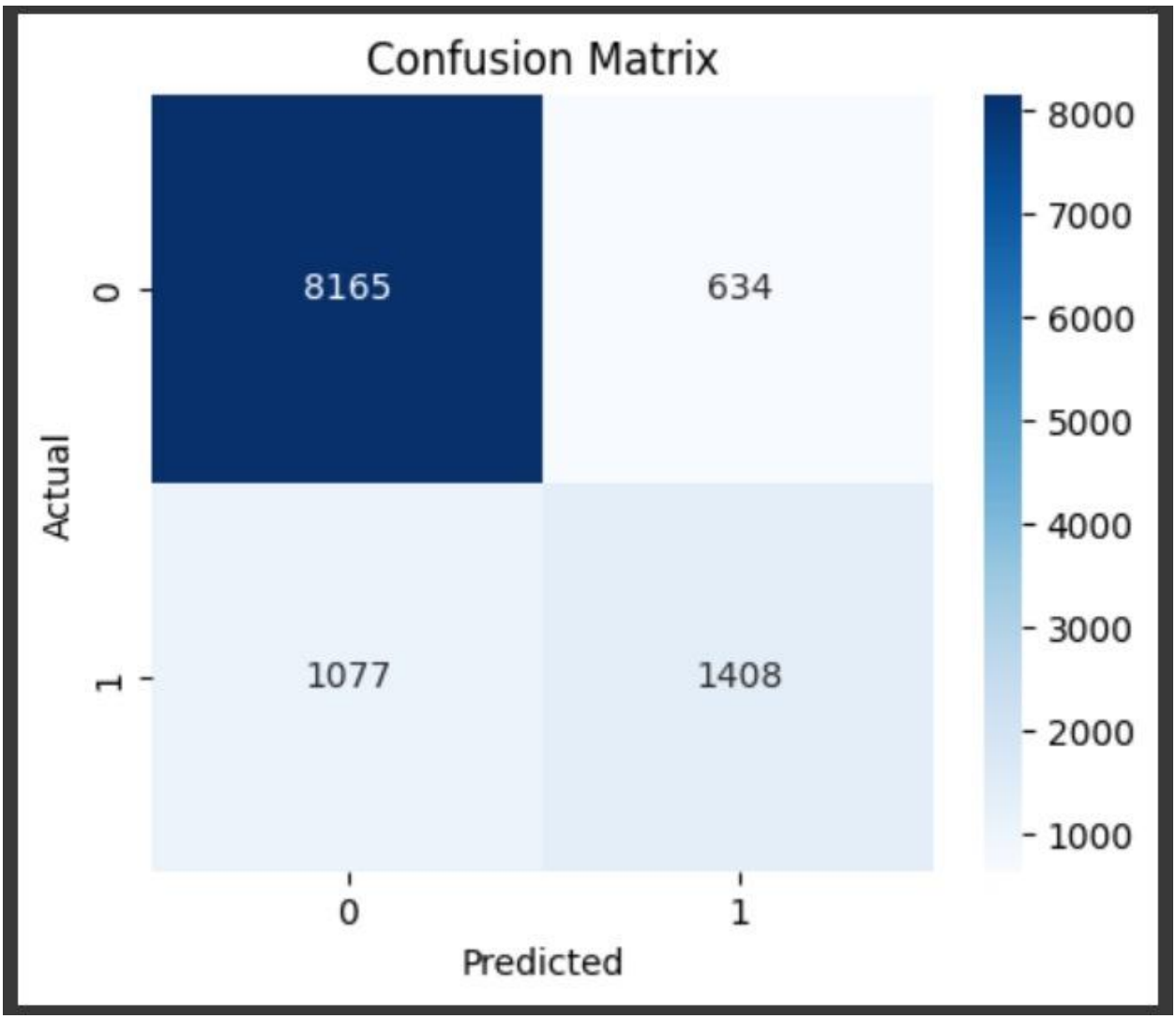
9. Conclusion

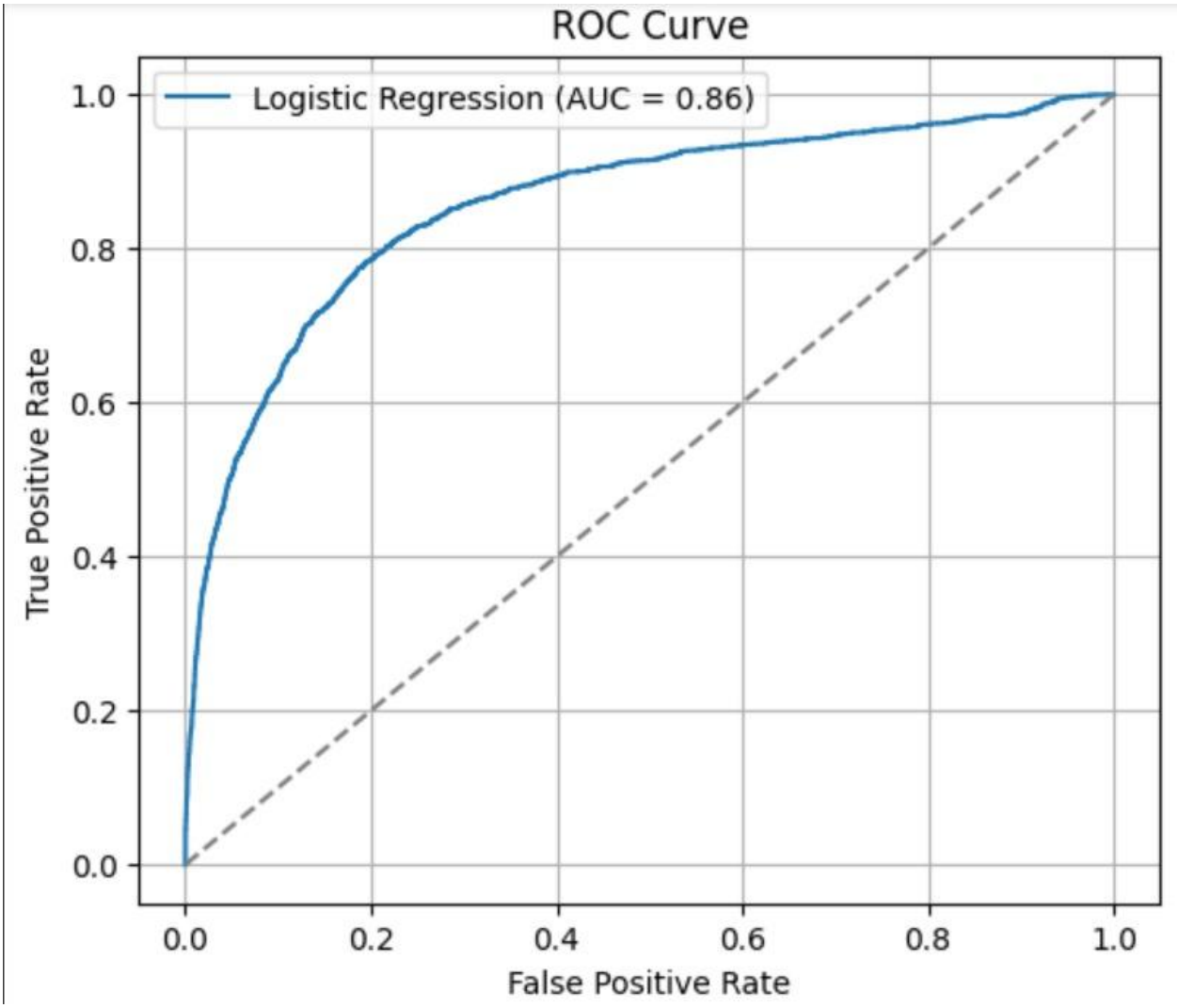
The logistic regression model successfully predicted rainfall with a satisfactory level of accuracy. Data preprocessing and feature scaling were crucial steps in improving model performance. Although the model performed well on the majority class, it showed limitations in detecting rainy days accurately. Addressing class imbalance and exploring advanced algorithms could enhance prediction reliability. Overall, this project demonstrates the potential of machine learning for effective rainfall forecasting.

---

10. References

- Aus weather dataset from KAGGLE
- pandas documentation
- Seaborn visualization library
- Research articles on credit risk prediction


Confusion Matrix
8165
634
1077
1408
0
1
Actual
0
1
Predicted
8000
7000
6000
5000
4000
3000
2000
1000



CODE: import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler

from sklearn.linear_model import LogisticRegression

from sklearn.metrics import classification_report, confusion_matrix, accuracy_score

 # Step 1: Load dataset

```
df = pd.read_csv("weatherAUS.csv")

# Step 2: Drop columns with more than 30% missing values
threshold = 0.3 * len(df)
df = df.dropna(thresh=threshold, axis=1)

# Step 3: Drop rows with remaining missing values
df = df.dropna()

# Step 4: Encode binary categories
df['RainTomorrow'] = df['RainTomorrow'].map({'Yes': 1, 'No': 0})
df['RainToday'] = df['RainToday'].map({'Yes': 1, 'No': 0})

# Step 5: One-hot encode remaining categorical variables
df = pd.get_dummies(df, drop_first=True)

# Step 6: Features and target
X = df.drop('RainTomorrow', axis=1)
y = df['RainTomorrow']

# Step 7: Train-test split
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Step 8: Scale features (important for logistic regression)

scaler = StandardScaler()

X_train_scaled = scaler.fit_transform(X_train)

X_test_scaled = scaler.transform(X_test)

# Step 9: Train logistic regression model

model = LogisticRegression(max_iter=1000)

model.fit(X_train_scaled, y_train)

# Step 10: Predict and evaluate

y_pred = model.predict(X_test_scaled)

print("Confusion Matrix:")

print(confusion_matrix(y_test, y_pred))

print("\nAccuracy:", accuracy_score(y_test, y_pred))

print("\nClassification Report:")

print(classification_report(y_test, y_pred))
```