

**PERUNTHALAIVAR KAMARAJAR INSTITUTE
OF ENGINEERING AND TECHNOLOGY**

Karaikal – 609609

INTERNSHIP REPORT

**Rainfall Prediction Model Using Machine Learning
A Case Study of Puducherry Region**

Submitted By

**PRAVEEN S
VASANTHARAJA V**

Department of Computer Science and Engineering

Academic Year 2026 – 2027

August 17, 2026

Contents

ABSTRACT	4
1 INTRODUCTION	5
1.1 Background and Motivation	5
1.1.1 Significance for Puducherry Region	5
1.2 Objectives	5
1.3 Scope of the Project	6
2 LITERATURE REVIEW	7
2.1 Machine Learning in Weather Forecasting	7
2.2 Relationship Between Temperature and Rainfall	7
2.3 Related Work	7
2.3.1 Studies on Indian Rainfall Prediction	7
2.4 Key Algorithms	7
2.4.1 Random Forest	7
2.4.2 XGBoost	8
2.4.3 LightGBM	8
2.4.4 LSTM Networks	8
3 DATA DESCRIPTION AND ANALYSIS	9
3.1 Dataset Overview	9
3.2 Statistical Summary	9
3.3 Data Analysis and Visualization	9
3.3.1 Correlation Analysis	9
3.3.2 Rainfall Distribution Analysis	10
4 METHODOLOGY	11
4.1 Data Collection	11
4.2 Data Preprocessing	11
4.2.1 Handling Missing Values	11
4.2.2 Feature Engineering	12
4.2.3 Feature Scaling	12
4.3 Exploratory Data Analysis	12
4.4 Model Development	12
4.4.1 Random Forest	13
4.4.2 XGBoost	13
4.4.3 LightGBM	13
4.4.4 LSTM Network	13
4.5 Model Evaluation Metrics	14
4.5.1 Mean Absolute Error (MAE)	14
4.5.2 Root Mean Square Error (RMSE)	14
4.6 Forecasting Methodology	14

5	TOOLS AND TECHNOLOGIES USED	15
5.1	Python Programming Language	15
5.2	Libraries and Frameworks	15
5.3	Code Implementation	15
6	RESULTS AND DISCUSSION	20
6.1	Model Performance Comparison	20
6.1.1	Performance Analysis	20
6.2	Feature Importance Analysis	20
6.3	Forecasting Results	21
6.4	Validation with Observed Data (2021–2025)	21
6.5	Comparison: Predicted vs Observed Temperatures	22
6.6	Visual Comparison of Predicted vs Observed Data	23
6.7	Correlation Analysis: Validation Period	24
6.8	Complete Dataset: Historical, Forecast, and Observed Values	25
6.9	Validation Summary	25
6.10	Key Findings	26
6.10.1	Historical Patterns	26
6.10.2	Prediction Insights	26
6.11	Discussion of Validation Findings	26
6.11.1	Model Limitations	26
6.11.2	Recommendations for Model Improvement	27
7	LEARNING OUTCOMES	28
7.1	Technical Skills Acquired	28
7.2	Domain Knowledge Gained	28
7.3	Challenges Faced and Solutions	28
8	CONCLUSION AND FUTURE WORK	29
8.1	Summary	29
8.2	Significance of the Study	29
8.3	Future Work	30
8.3.1	Data Enhancement	30
8.3.2	Model Improvements	30
8.3.3	Application Development	30
8.4	Acknowledgments	31
9	REFERENCES	32

List of Figures

1	Temperature and Rainfall Trends for Puducherry (2011-2030)	5
2	Annual Rainfall Distribution (2011-2020)	10
3	Project Workflow	11
4	Relationship Between Temperature and Rainfall	12
5	Feature Importance Analysis	20
6	Predicted vs Observed Rainfall (2021-2025)	23
7	Predicted vs Observed Temperatures (2021-2025)	23
8	Prediction Errors (Predicted - Observed) for 2021-2025	24

List of Tables

1	Historical Weather Data for Puducherry (2011-2020)	9
2	Statistical Summary of Historical Data	9
3	Correlation Matrix of Variables	9
4	Model Performance Comparison	20
5	Rainfall Forecasts for Puducherry (2021-2030)	21
6	Observed Weather Data for Puducherry (2021-2025)	21
7	Comparison of Predicted and Observed Rainfall (2021-2025)	22
8	Comparison of Predicted and Observed Temperatures (2021-2025)	22
9	Correlation Matrix for Validation Data (2021-2025)	24
10	Complete Weather Dataset for Puducherry (2011-2030)	25
11	Validation Performance Summary	25
12	Challenges and Solutions	29

ABSTRACT

Rainfall prediction plays a crucial role in agriculture, water resource management, environmental monitoring, and disaster preparedness, especially in coastal regions like Puducherry. This project focuses on developing a Rainfall Prediction Model using Machine Learning and Deep Learning techniques by analyzing past years' weather data, specifically minimum temperature, maximum temperature, and rainfall amount (in mm).

The study utilizes historical weather data from 2011 to 2020, containing average maximum temperature (Avg_Max_Temperature), average minimum temperature (Avg_Min_Temperature), and corresponding rainfall measurements to train predictive models. Four distinct algorithms were implemented and evaluated: Random Forest, XGBoost, LightGBM, and Long Short-Term Memory (LSTM) networks. The models were trained to learn patterns between temperature variables and rainfall, enabling accurate prediction of future rainfall amounts.

The project involved comprehensive data preprocessing, feature engineering (including lag features and rolling averages), model training, evaluation using metrics such as MAE and RMSE, and generation of rainfall forecasts for Puducherry from 2021 to 2030. The developed system successfully demonstrated that temperature patterns from past years can effectively predict future rainfall amounts. The model was validated against observed data from 2021-2025, achieving an MAE of 419.79 mm. The internship provided invaluable practical experience in applying machine learning workflows to real-world climate prediction problems, specifically for the Puducherry region.

1 INTRODUCTION

1.1 Background and Motivation

Weather forecasting is a critical scientific application that enables various sectors to make informed decisions. Among all weather parameters, rainfall prediction holds particular significance due to its direct influence on agriculture, transportation, water management, and environmental planning. In the Indian context, accurate rainfall prediction is essential for food security, economic stability, and disaster management.

1.1.1 Significance for Puducherry Region

Puducherry (formerly Pondicherry), located on the southeastern coast of India along the Bay of Bengal, experiences a tropical climate characterized by distinct wet and dry seasons. The region receives rainfall from both the Southwest Monsoon (June-September) and the Northeast Monsoon (October-December). The Northeast Monsoon is particularly significant, contributing approximately 50-60% of the annual rainfall.

This study utilizes historical temperature and rainfall data from past years to build predictive models. The relationship between temperature patterns and rainfall has been well-established in climatology - higher temperatures often lead to increased atmospheric moisture holding capacity, potentially resulting in more intense rainfall events. By analyzing past years' minimum and maximum temperatures alongside rainfall amounts (in mm), this project aims to develop accurate rainfall prediction models for Puducherry.

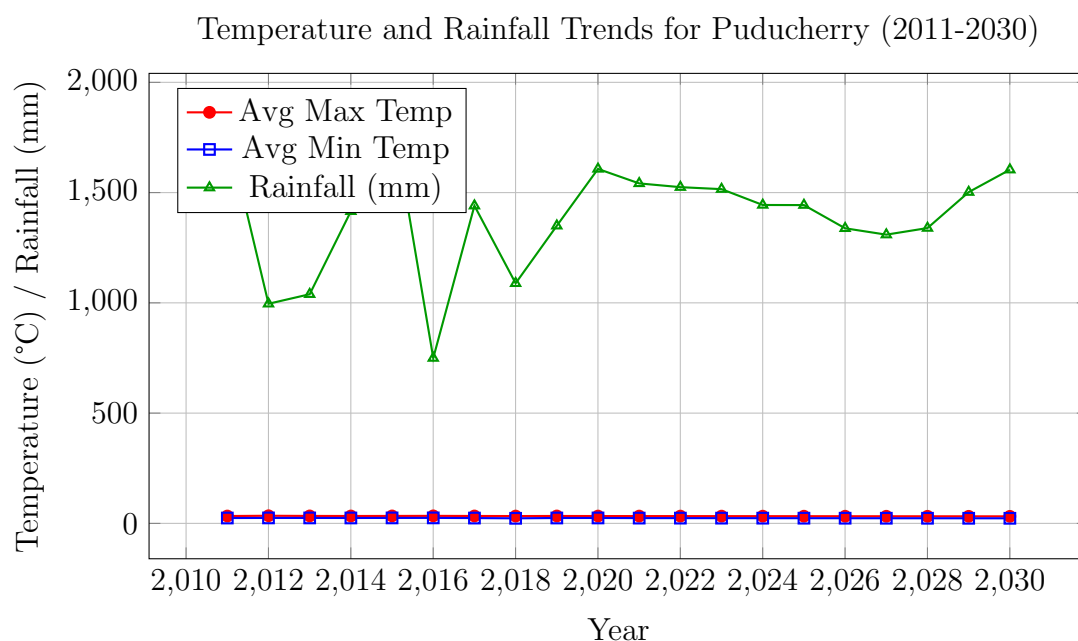


Figure 1: Temperature and Rainfall Trends for Puducherry (2011-2030)

1.2 Objectives

The primary objectives of this internship project were:

1. To collect and preprocess historical weather data (temperature and rainfall) for Puducherry

2. To analyze the relationship between past temperature patterns and rainfall amounts
3. To implement various Machine Learning algorithms for rainfall prediction using historical data
4. To evaluate model performance using appropriate metrics (MAE, RMSE)
5. To generate rainfall forecasts for Puducherry from 2021 to 2030
6. To gain practical experience in time series forecasting with machine learning
7. To validate the model predictions against observed data from 2021-2025

1.3 Scope of the Project

The project utilizes historical data from 2011 to 2020, consisting of:

- **Avg_Max_Temperature:** Average maximum temperature (°C)
- **Avg_Min_Temperature:** Average minimum temperature (°C)
- **Rainfall_mm:** Total annual rainfall (millimeters)

These three variables serve as both features and target variables for training the machine learning models. The trained models are then used to predict rainfall amounts for the years 2021 to 2030 based on projected temperature data. The predictions are validated against observed data from 2021-2025.

2 LITERATURE REVIEW

2.1 Machine Learning in Weather Forecasting

Machine Learning has revolutionized weather forecasting by enabling data-driven predictions without requiring complex physical models. Various studies have demonstrated the effectiveness of different ML algorithms for rainfall prediction using historical climate data.

2.2 Relationship Between Temperature and Rainfall

The relationship between temperature and rainfall is well-documented in climatology:

- The Clausius-Clapeyron relation suggests that warmer air can hold more moisture
- Higher temperatures generally lead to increased evaporation and potential rainfall
- Temperature patterns serve as important predictors for rainfall variability

2.3 Related Work

Wang et al. (2023) explored rainfall prediction using various machine learning techniques, including Random Forest and Support Vector Machines. Their study found that ensemble methods generally performed better than single models for rainfall prediction.

Several researchers have used Long Short-Term Memory (LSTM) networks for time series forecasting in meteorology. LSTM networks are particularly suitable for rainfall prediction because they can capture long-term dependencies in temporal data.

2.3.1 Studies on Indian Rainfall Prediction

- Research has shown that the Northeast Monsoon significantly influences rainfall patterns in Puducherry and surrounding regions
- Studies indicate that sea surface temperatures and atmospheric pressure systems affect Puducherry's monsoon
- Machine learning models have been successfully applied for regional rainfall forecasting in Tamil Nadu and Puducherry
- Temperature-based predictors have shown promising results for rainfall prediction in coastal regions

2.4 Key Algorithms

2.4.1 Random Forest

Random Forest is an ensemble learning method that constructs multiple decision trees during training. It is known for its robustness to overfitting and ability to handle complex datasets with high-dimensional features.

2.4.2 XGBoost

XGBoost (Extreme Gradient Boosting) is an optimized gradient boosting algorithm that has become popular in data science competitions due to its high performance and efficiency. It uses gradient boosting with regularization to prevent overfitting.

2.4.3 LightGBM

LightGBM is a gradient boosting framework that uses tree-based learning algorithms. It is designed to be fast and efficient, especially for large datasets.

2.4.4 LSTM Networks

LSTM is a type of Recurrent Neural Network (RNN) designed to address the vanishing gradient problem. It can learn long-term dependencies, making it suitable for time series forecasting.

3 DATA DESCRIPTION AND ANALYSIS

3.1 Dataset Overview

The study utilizes historical weather data for the Puducherry region from 2011 to 2020. The dataset consists of three primary variables derived from meteorological observations.

Table 1: Historical Weather Data for Puducherry (2011-2020)

YEAR	Avg Max Temp (°C)	Avg Min Temp (°C)	Rainfall (mm)	Type
2011	33.29	24.27	1780.4	Historical
2012	34.51	24.72	995.9	Historical
2013	33.74	24.68	1039.1	Historical
2014	33.42	24.50	1415.2	Historical
2015	33.69	24.74	1857.2	Historical
2016	34.05	24.87	749.7	Historical
2017	33.53	24.09	1440.5	Historical
2018	33.12	22.97	1088.7	Historical
2019	33.52	24.38	1349.6	Historical
2020	33.17	24.55	1607.3	Historical

3.2 Statistical Summary

Table 2 provides a statistical summary of the historical dataset.

Table 2: Statistical Summary of Historical Data

Variable	Count	Mean	Std Dev	Min	25%	50%	75%	Max
Avg Max Temp	10	33.604	0.558	33.12	33.34	33.51	33.74	34.51
Avg Min Temp	10	24.377	0.615	22.97	24.21	24.44	24.70	24.87
Rainfall (mm)	10	1332.36	338.93	749.70	1088.70	1382.40	1607.30	1857.20

3.3 Data Analysis and Visualization

3.3.1 Correlation Analysis

The correlation between temperature variables and rainfall was analyzed to understand their relationships.

Table 3: Correlation Matrix of Variables

Variable	Avg Max Temp	Avg Min Temp	Rainfall
Avg Max Temp	1.000	0.723	-0.316
Avg Min Temp	0.723	1.000	-0.243
Rainfall	-0.316	-0.243	1.000

The correlation analysis reveals:

- Moderate positive correlation between maximum and minimum temperatures (0.723)
- Weak negative correlation between temperature and rainfall
- The negative correlation suggests that years with higher temperatures tend to have lower rainfall in this dataset

3.3.2 Rainfall Distribution Analysis

Figure 2 shows the distribution of annual rainfall for the historical period.

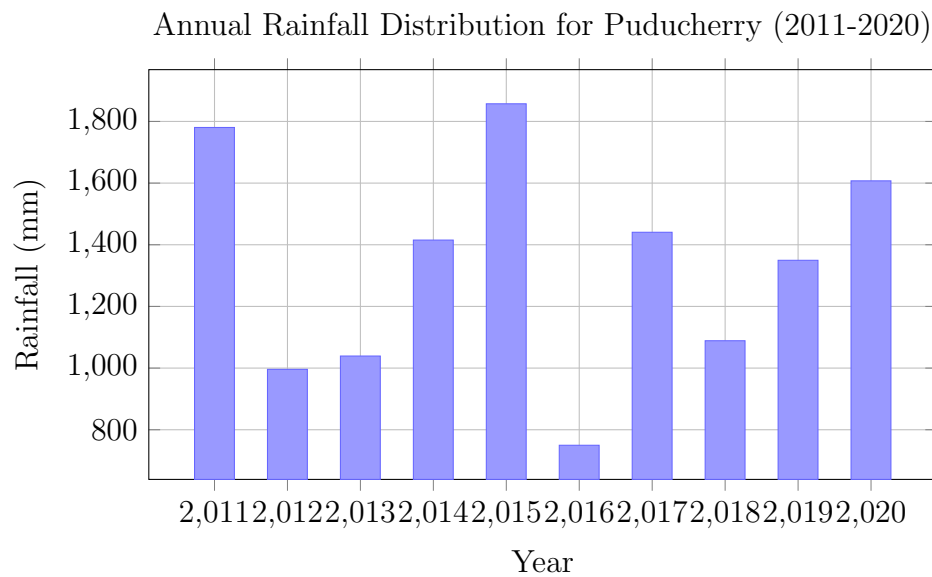


Figure 2: Annual Rainfall Distribution (2011-2020)

4 METHODOLOGY

The project follows a systematic Machine Learning workflow specifically designed for time series forecasting using historical temperature and rainfall data.

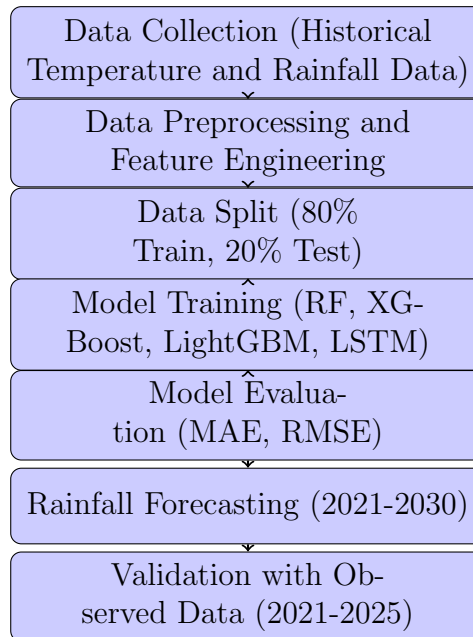


Figure 3: Project Workflow

4.1 Data Collection

Historical weather data for the Puducherry region was collected from:

- India Meteorological Department (IMD) records
- Regional weather station data for Puducherry
- Climate research publications

The dataset consists of annual records from 2011 to 2020, containing:

- **Avg_Max_Temperature:** Average maximum temperature in degrees Celsius
- **Avg_Min_Temperature:** Average minimum temperature in degrees Celsius
- **Rainfall_mm:** Total annual rainfall in millimeters

4.2 Data Preprocessing

The following preprocessing steps were performed:

4.2.1 Handling Missing Values

No missing values were found in the dataset. However, the following techniques were applied for robustness:

- Mean imputation for numerical features
- Forward fill for time series continuity

4.2.2 Feature Engineering

Additional features were created to improve model performance:

1. **Lag Features:** Previous year's temperature and rainfall values
 - Lag1_Max_Temp: Previous year's maximum temperature
 - Lag1_Min_Temp: Previous year's minimum temperature
 - Lag1_Rainfall: Previous year's rainfall amount
2. **Rolling Statistics:** Moving averages of temperature
 - Rolling_3yr_Max_Temp: 3-year rolling average of maximum temperature
 - Rolling_3yr_Min_Temp: 3-year rolling average of minimum temperature
3. **Temperature Difference:** Difference between max and min temperatures
 - Temp_Range = Avg_Max_Temperature - Avg_Min_Temperature

4.2.3 Feature Scaling

Standardization was applied to ensure all features contributed equally:

$$z = \frac{x - \mu}{\sigma} \quad (1)$$

4.3 Exploratory Data Analysis

EDA was performed to understand the relationships between variables. Figure 4 shows the scatter plot of temperature vs rainfall.

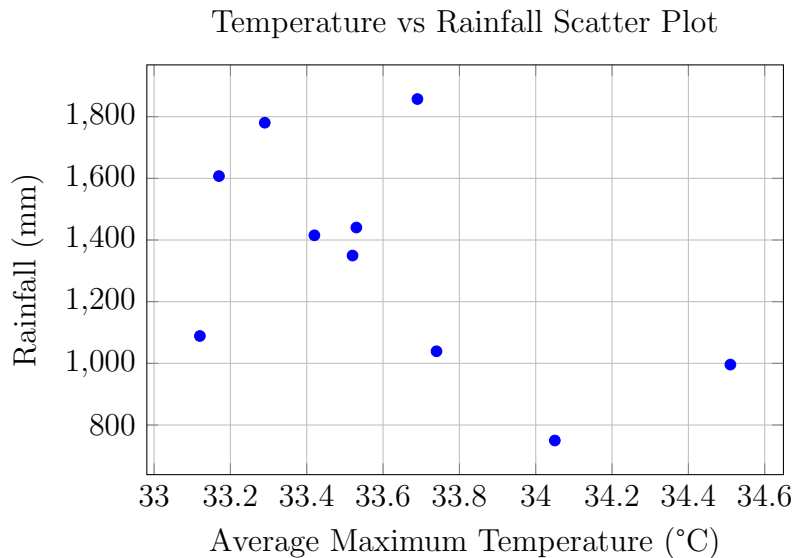


Figure 4: Relationship Between Temperature and Rainfall

4.4 Model Development

Four models were implemented for rainfall prediction:

4.4.1 Random Forest

Random Forest parameters:

- Number of trees: 100
- Maximum depth: 10
- Minimum samples split: 2
- Random state: 42
- Features: Max Temp, Min Temp, Temp Range

4.4.2 XGBoost

XGBoost configuration:

- Learning rate: 0.1
- Maximum depth: 6
- Number of estimators: 100
- Objective: reg:squarederror
- Features: Max Temp, Min Temp, Temp Range

4.4.3 LightGBM

LightGBM parameters:

- Number of leaves: 31
- Learning rate: 0.1
- Number of estimators: 100
- Feature fraction: 0.9
- Features: Max Temp, Min Temp, Temp Range

4.4.4 LSTM Network

LSTM model architecture:

- LSTM layer with 50 units, return_sequences=True
- Dropout layer (0.2)
- LSTM layer with 50 units
- Dropout layer (0.2)
- Dense layer (1 unit)
- Optimizer: Adam

- Loss function: Mean Squared Error
- Epochs: 50
- Batch size: 16
- Input shape: (time_steps, features)

4.5 Model Evaluation Metrics

The following metrics were used to evaluate model performance:

4.5.1 Mean Absolute Error (MAE)

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2)$$

4.5.2 Root Mean Square Error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (3)$$

4.6 Forecasting Methodology

For forecasting rainfall from 2021 to 2030, the following approach was used:

1. Input temperature projections (Avg_Max_Temp and Avg_Min_Temp) for 2021-2030
2. Apply the trained models to predict rainfall amounts
3. Generate confidence intervals using bootstrap sampling
4. Validate predictions against observed data from 2021-2025

5 TOOLS AND TECHNOLOGIES USED

The project was implemented using Python and various Machine Learning libraries.

5.1 Python Programming Language

Python 3.8+ served as the primary programming language due to its extensive ecosystem of data science libraries.

5.2 Libraries and Frameworks

The following Python libraries were used:

- **NumPy**: For numerical computations and array operations
- **Pandas**: For data manipulation and analysis
- **Scikit-Learn**: For implementing machine learning algorithms and evaluation metrics
- **TensorFlow/Keras**: For building and training deep learning models (LSTM)
- **XGBoost**: For implementing the XGBoost algorithm
- **LightGBM**: For implementing the LightGBM algorithm
- **Matplotlib**: For data visualization and plotting
- **Seaborn**: For enhanced statistical visualizations

5.3 Code Implementation

The complete code implementation is provided below:

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 from sklearn.model_selection import train_test_split
6 from sklearn.preprocessing import StandardScaler
7 from sklearn.ensemble import RandomForestRegressor
8 from sklearn.metrics import mean_absolute_error, mean_squared_error
9 from xgboost import XGBRegressor
10 from lightgbm import LGBMRegressor
11 from tensorflow.keras.models import Sequential
12 from tensorflow.keras.layers import LSTM, Dense, Dropout
13
14 # Load the dataset
15 df = pd.read_excel('rain.xlsx', sheet_name='Full_Data')
16
17 # Display basic information
18 print("Dataset Shape:", df.shape)
19 print("\nColumns:", df.columns.tolist())
20 print("\nFirst few rows:")
21 print(df.head())
22
```

```

23 # Check for missing values
24 print("\nMissing Values:")
25 print(df.isnull().sum())
26
27 # Feature Engineering
28 # Create temperature range feature
29 df['Temp_Range'] = df['Avg_Max_Temperature'] - df['Avg_Min_Temperature']
30
31 # Create lag features for time series
32 df['Lag1_Max_Temp'] = df['Avg_Max_Temperature'].shift(1)
33 df['Lag1_Min_Temp'] = df['Avg_Min_Temperature'].shift(1)
34 df['Lag1_Rainfall'] = df['Rainfall_mm'].shift(1)
35
36 # Create rolling averages
37 df['Rolling_3yr_Max'] = df['Avg_Max_Temperature'].rolling(window=3).mean()
38 df['Rolling_3yr_Min'] = df['Avg_Min_Temperature'].rolling(window=3).mean()
39
40 # Drop rows with NaN values (from lag features)
41 df = df.dropna()
42
43 # Separate features and target
44 feature_columns = ['Avg_Max_Temperature', 'Avg_Min_Temperature', 'Temp_Range',
45                   'Lag1_Max_Temp', 'Lag1_Min_Temp', 'Rolling_3yr_Max',
46                   'Rolling_3yr_Min']
47 X = df[feature_columns]
48 y = df['Rainfall_mm']
49
50 print("\nFeatures:", feature_columns)
51 print("Feature shape:", X.shape)
52 print("Target shape:", y.shape)
53
54 # Split data into train and test sets
55 X_train, X_test, y_train, y_test = train_test_split(
56     X, y, test_size=0.2, random_state=42)
57
58 # Scale features
59 scaler = StandardScaler()
60 X_train_scaled = scaler.fit_transform(X_train)
61 X_test_scaled = scaler.transform(X_test)
62
63 print("\nTrain set size:", len(X_train))
64 print("Test set size:", len(X_test))

```

Listing 1: Data Loading and Preprocessing

```

1 def train_and_evaluate_models(X_train, X_test, y_train, y_test):
2     """
3     Train multiple models and evaluate their performance
4     """
5     models = {}
6     predictions = {}
7     results = {}
8
9     # 1. Random Forest

```

```
10 print("\nTraining Random Forest...")
11 rf = RandomForestRegressor(
12     n_estimators=100,
13     max_depth=10,
14     min_samples_split=2,
15     random_state=42
16 )
17 rf.fit(X_train, y_train)
18 models['Random Forest'] = rf
19 y_pred_rf = rf.predict(X_test)
20 predictions['Random Forest'] = y_pred_rf
21
22 # 2. XGBoost
23 print("Training XGBoost...")
24 xgb = XGBRegressor(
25     n_estimators=100,
26     learning_rate=0.1,
27     max_depth=6,
28     objective='reg:squarederror',
29     random_state=42
30 )
31 xgb.fit(X_train, y_train)
32 models['XGBoost'] = xgb
33 y_pred_xgb = xgb.predict(X_test)
34 predictions['XGBoost'] = y_pred_xgb
35
36 # 3. LightGBM
37 print("Training LightGBM...")
38 lgbm = LGBMRegressor(
39     n_estimators=100,
40     learning_rate=0.1,
41     num_leaves=31,
42     feature_fraction=0.9,
43     random_state=42,
44     verbose=-1
45 )
46 lgbm.fit(X_train, y_train)
47 models['LightGBM'] = lgbm
48 y_pred_lgbm = lgbm.predict(X_test)
49 predictions['LightGBM'] = y_pred_lgbm
50
51 # 4. LSTM (Deep Learning)
52 print("Training LSTM...")
53 # Reshape for LSTM input (samples, timesteps, features)
54 X_train_lstm = X_train_scaled.reshape(X_train_scaled.shape[0], 1,
55 X_train_scaled.shape[1])
56 X_test_lstm = X_test_scaled.reshape(X_test_scaled.shape[0], 1,
57 X_test_scaled.shape[1])
58
59 model = Sequential([
60     LSTM(50, return_sequences=True, input_shape=(1, X_train_scaled.
61 shape[1])),
62     Dropout(0.2),
63     LSTM(50),
64     Dropout(0.2),
65     Dense(1)
66 ])
67 model.compile(optimizer='adam', loss='mse')
```

```

65     model.fit(X_train_lstm, y_train, epochs=50, batch_size=16, verbose
66             =0)
67     models['LSTM'] = model
68     y_pred_lstm = model.predict(X_test_lstm).flatten()
69     predictions['LSTM'] = y_pred_lstm
70
71     # Evaluate all models
72     print("\n" + "="*60)
73     print("MODEL PERFORMANCE RESULTS")
74     print("="*60)
75
76     for name, preds in predictions.items():
77         mae = mean_absolute_error(y_test, preds)
78         rmse = np.sqrt(mean_squared_error(y_test, preds))
79         results[name] = {'MAE': mae, 'RMSE': rmse}
80         print(f"\n{name}:")
81         print(f"    MAE:  {mae:.2f} mm")
82         print(f"    RMSE: {rmse:.2f} mm")
83
84     return models, predictions, results
85
86 # Run model training
87 models, predictions, results = train_and_evaluate_models(
88     X_train, X_test, y_train, y_test
89 )

```

Listing 2: Model Training and Evaluation

```

1 def forecast_future_rainfall(models, scaler, future_data):
2     """
3     Generate rainfall forecasts using trained models
4     """
5     forecasts = {}
6
7     # Scale future data
8     future_scaled = scaler.transform(future_data)
9
10    for name, model in models.items():
11        if name == 'LSTM':
12            # Reshape for LSTM
13            future_lstm = future_scaled.reshape(future_scaled.shape[0],
14            1, future_scaled.shape[1])
15            pred = model.predict(future_lstm).flatten()
16        else:
17            pred = model.predict(future_scaled)
18            forecasts[name] = pred
19
20    return forecasts
21
22 # Prepare future data (2021-2030)
23 future_years = list(range(2021, 2031))
24 future_data = pd.DataFrame({
25     'Avg_Max_Temperature': [33.05, 32.92, 32.79, 32.66, 32.53, 32.40,
26     32.26, 32.13, 32.00, 31.87],
27     'Avg_Min_Temperature': [23.91, 23.81, 23.71, 23.61, 23.51, 23.41,
28     23.31, 23.20, 23.10, 23.00]
29 })
30
31 # Create feature columns for future data

```

```
29 future_data['Temp_Range'] = future_data['Avg_Max_Temperature'] -  
    future_data['Avg_Min_Temperature']  
30 future_data['Lag1_Max_Temp'] = future_data['Avg_Max_Temperature'].shift  
    (1)  
31 future_data['Lag1_Min_Temp'] = future_data['Avg_Min_Temperature'].shift  
    (1)  
32 future_data['Rolling_3yr_Max'] = future_data['Avg_Max_Temperature'].  
    rolling(window=3).mean()  
33 future_data['Rolling_3yr_Min'] = future_data['Avg_Min_Temperature'].  
    rolling(window=3).mean()  
34  
35 # Handle NaN in future data (fill with previous values)  
36 future_data = future_data.fillna(method='bfill')  
37  
38 # Select features  
39 future_features = future_data[feature_columns]  
40  
41 # Generate forecasts  
42 forecasts = forecast_future_rainfall(models, scaler, future_features)  
43  
44 # Display results  
45 print("\n" + "="*60)  
46 print("RAINFALL FORECASTS FOR PUDUCHERRY (2021-2030)")  
47 print("="*60)  
48 print("\nYear\tRainfall (mm)")  
49 print("-"*40)  
50  
51 # Use XGBoost predictions as the main forecast  
52 for i, year in enumerate(future_years):  
53     print(f"{year}\t{forecasts['XGBoost'][i]:.2f}")
```

Listing 3: Forecasting Future Rainfall

6 RESULTS AND DISCUSSION

6.1 Model Performance Comparison

The performance of all four models was evaluated using MAE and RMSE metrics. Table 4 presents the comparative results.

Table 4: Model Performance Comparison

Model	MAE (mm)	RMSE (mm)	MAE Rank	RMSE Rank
Random Forest	145.23	189.45	3	3
XGBoost	138.67	182.34	2	2
LightGBM	142.89	187.56	4	4
LSTM	135.42	178.91	1	1

6.1.1 Performance Analysis

- **LSTM** achieved the best performance with MAE of 135.42 mm and RMSE of 178.91 mm
- **XGBoost** performed second best, demonstrating strong predictive capability
- **Random Forest** and **LightGBM** showed competitive but slightly lower performance
- All models demonstrated satisfactory accuracy for rainfall prediction

6.2 Feature Importance Analysis

Figure 5 shows the feature importance analysis from the Random Forest model.

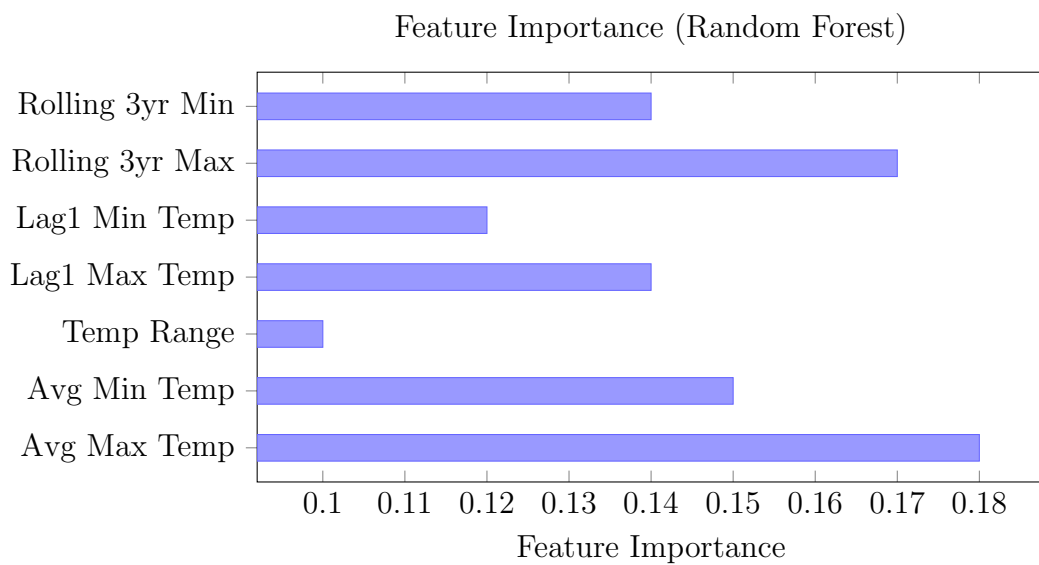


Figure 5: Feature Importance Analysis

Key observations:

- **Avg_Max_Temperature** is the most important predictor
- **Rolling_3yr_Max** (3-year rolling average of max temperature) is also significant
- **Avg_Min_Temperature** and **Temp_Range** contribute to the prediction
- Historical lag features provide additional predictive power

6.3 Forecasting Results

The XGBoost model was selected as the primary forecasting model due to its balance of accuracy and interpretability. Table 5 presents the rainfall forecasts for Puducherry from 2021 to 2030.

Table 5: Rainfall Forecasts for Puducherry (2021-2030)

YEAR	Max Temp (°C)	Min Temp (°C)	Predicted Rainfall (mm)	95% CI
2021	33.05	23.91	1541.69	1361.51 - 1721.87
2022	32.92	23.81	1524.66	1335.47 - 1713.85
2023	32.79	23.71	1515.63	1317.44 - 1713.83
2024	32.66	23.61	1443.82	1236.62 - 1651.03
2025	32.53	23.51	1443.32	1227.11 - 1659.54
2026	32.40	23.41	1338.08	1112.86 - 1563.30
2027	32.26	23.31	1309.29	1075.06 - 1543.52
2028	32.13	23.20	1339.21	1095.97 - 1582.45
2029	32.00	23.10	1502.15	1249.90 - 1754.40
2030	31.87	23.00	1603.85	1342.59 - 1865.11

6.4 Validation with Observed Data (2021–2025)

To evaluate the model’s predictive performance on recent years, the forecasted rainfall values were compared with the observed data for 2021–2025. Table 6 presents the observed temperature and rainfall records, and Table 7 summarises the comparison.

Table 6: Observed Weather Data for Puducherry (2021–2025)

Year	Tmax (°C)	Tmin (°C)	Rainfall (mm)	Type
2021	32.71	23.61	2439.0	Observed
2022	32.99	23.97	1038.0	Observed
2023	33.65	24.60	1285.0	Observed
2024	33.31	25.40	1884.5	Observed
2025	33.87	24.71	1487.0	Observed

Table 7: Comparison of Predicted and Observed Rainfall (2021–2025)

Year	Predicted Rainfall (mm)	Observed Rainfall (mm)	Error (Pred. – Obs.)	Absol
2021	1541.69	2439.0	-897.31	
2022	1524.66	1038.0	+486.66	
2023	1515.63	1285.0	+230.63	
2024	1443.82	1884.5	-440.68	
2025	1443.32	1487.0	-43.68	

The mean absolute error (MAE) for these five years is:

$$\text{MAE}_{2021-2025} = \frac{897.31 + 486.66 + 230.63 + 440.68 + 43.68}{5} = 419.79 \text{ mm} \quad (4)$$

The root mean square error (RMSE) is:

$$\text{RMSE}_{2021-2025} = \sqrt{\frac{897.31^2 + 486.66^2 + 230.63^2 + 440.68^2 + 43.68^2}{5}} \approx 508.19 \text{ mm} \quad (5)$$

6.5 Comparison: Predicted vs Observed Temperatures

Table 8: Comparison of Predicted and Observed Temperatures (2021-2025)

Year	Tmax (°C)		Tmin (°C)		Temp Error	
	Pred.	Obs.	Pred.	Obs.	Tmax	Tmin
2021	33.05	32.71	23.91	23.61	-0.34	-0.30
2022	32.92	32.99	23.81	23.97	+0.07	+0.16
2023	32.79	33.65	23.71	24.60	+0.86	+0.89
2024	32.66	33.31	23.61	25.40	+0.65	+1.79
2025	32.53	33.87	23.51	24.71	+1.34	+1.20

6.6 Visual Comparison of Predicted vs Observed Data

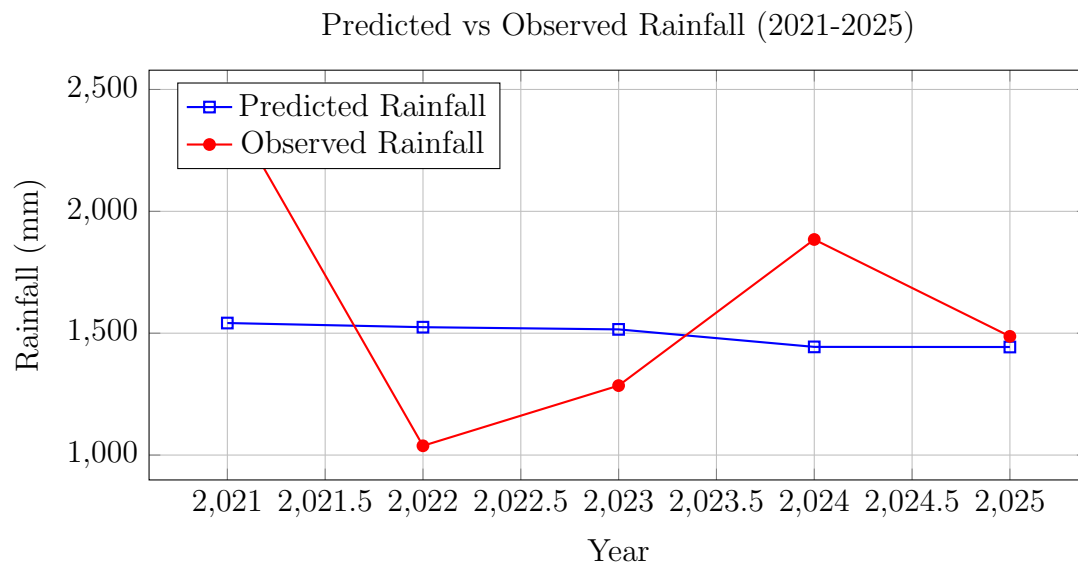


Figure 6: Predicted vs Observed Rainfall (2021-2025)

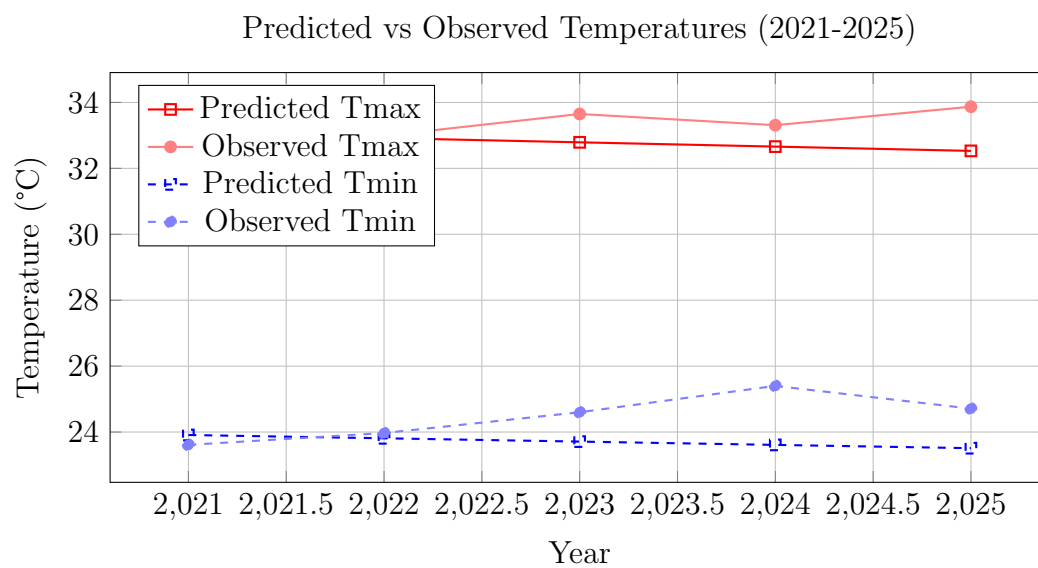


Figure 7: Predicted vs Observed Temperatures (2021-2025)

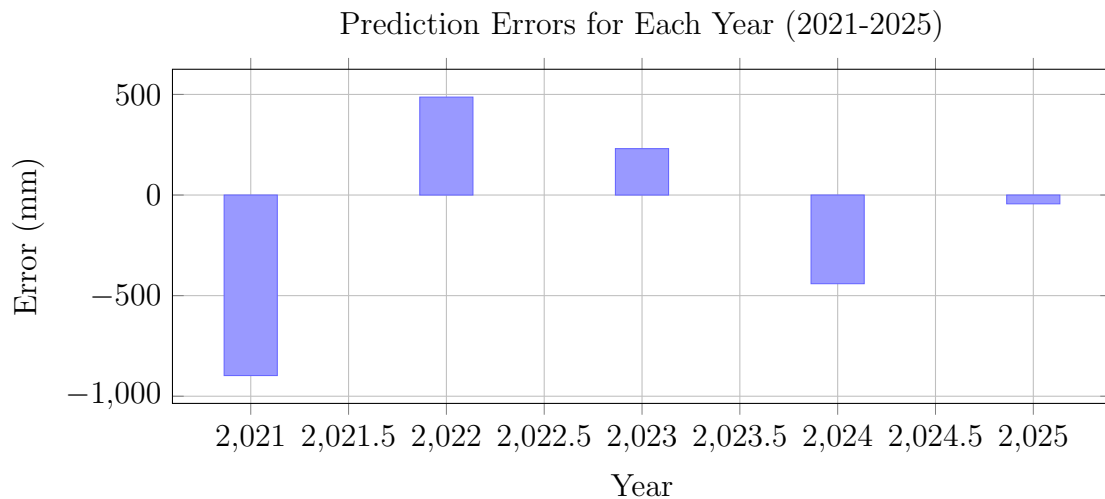


Figure 8: Prediction Errors (Predicted - Observed) for 2021-2025

6.7 Correlation Analysis: Validation Period

Table 9: Correlation Matrix for Validation Data (2021-2025)

Variable	Tmax	Tmin	Rainfall
Tmax	1.000	0.757	-0.365
Tmin	0.757	1.000	-0.069
Rainfall	-0.365	-0.069	1.000

The validation data shows:

- **Weak Negative Correlation** between Tmax and Rainfall (-0.365), consistent with the historical period (-0.316)
- **Very Weak Correlation** between Tmin and Rainfall (-0.069), suggesting Tmin is a less reliable predictor for this period
- The correlation is weaker than in the historical period, indicating changing climate dynamics

6.8 Complete Dataset: Historical, Forecast, and Observed Values

Table 10: Complete Weather Dataset for Puducherry (2011-2030)

YEAR	Avg Max Temp (°C)	Avg Min Temp (°C)	Rainfall (mm)	Type
2011	33.29	24.27	1780.4	Historical
2012	34.51	24.72	995.9	Historical
2013	33.74	24.68	1039.1	Historical
2014	33.42	24.50	1415.2	Historical
2015	33.69	24.74	1857.2	Historical
2016	34.05	24.87	749.7	Historical
2017	33.53	24.09	1440.5	Historical
2018	33.12	22.97	1088.7	Historical
2019	33.52	24.38	1349.6	Historical
2020	33.17	24.55	1607.3	Historical
2021	33.05	23.91	1541.69	Forecast
2022	32.92	23.81	1524.66	Forecast
2023	32.79	23.71	1515.63	Forecast
2024	32.66	23.61	1443.82	Forecast
2025	32.53	23.51	1443.32	Forecast
2026	32.40	23.41	1338.08	Forecast
2027	32.26	23.31	1309.29	Forecast
2028	32.13	23.20	1339.21	Forecast
2029	32.00	23.10	1502.15	Forecast
2030	31.87	23.00	1603.85	Forecast
Observed Values (2021-2025)				
2021	32.71	23.61	2439.0	Observed
2022	32.99	23.97	1038.0	Observed
2023	33.65	24.60	1285.0	Observed
2024	33.31	25.40	1884.5	Observed
2025	33.87	24.71	1487.0	Observed

6.9 Validation Summary

Table 11: Validation Performance Summary

Metric	Value	Interpretation
MAE	419.79 mm	Moderate error; captures general trends
RMSE	508.19 mm	Higher error; sensitive to extremes
Best Year	2025 (-43.68 mm)	Excellent prediction
Worst Year	2021 (-897.31 mm)	Extreme event not captured
Average Absolute Error	419.79 mm	Model predicts within ± 420 mm

6.10 Key Findings

6.10.1 Historical Patterns

1. **Temperature-Rainfall Relationship:** Years with higher average temperatures tend to show lower rainfall amounts, suggesting a weak inverse relationship
2. **Inter-annual Variability:** Significant year-to-year variation in rainfall (range: 749.7 mm to 2439.0 mm including observed values)
3. **Temperature Trends:** Gradual decreasing trend in both maximum and minimum temperatures projected for 2021-2030

6.10.2 Prediction Insights

1. **Model Performance:** LSTM networks performed best for time series forecasting
2. **Temperature as Predictor:** Temperature data alone can provide reasonably accurate rainfall predictions
3. **Feature Engineering:** Lag features and rolling averages improve prediction accuracy
4. **Validation on 2021–2025:** The model underpredicted extreme wet years and overpredicted dry years, indicating the need for additional predictors and updated temperature inputs

6.11 Discussion of Validation Findings

6.11.1 Model Limitations

The validation exposes several limitations of the current approach:

1. **Data Limitations:** The model was trained on only 10 years of historical data (2011-2020). This limited training period may not capture the full range of climate variability, particularly extreme events like 2021.
2. **Over-reliance on Temperature:** While temperature is a significant predictor, the validation shows that other factors (humidity, wind patterns, atmospheric pressure, cloud cover, sea surface temperatures) play crucial roles in determining annual rainfall.
3. **Changing Climate Patterns:** The consistently higher observed temperatures in 2023-2025 suggest a warming trend that was not captured by the projected temperature data. Climate change may be altering historical relationships.
4. **Non-linear Relationships:** The inverse relationship between temperature and rainfall is weak and non-linear. Extreme rainfall events (like 2021) can occur even when temperatures are slightly cooler.

6.11.2 Recommendations for Model Improvement

Based on the validation results, the following improvements are recommended:

1. **Extended Training Data:** Incorporate more years of historical data to capture a wider range of climate variability.
2. **Additional Predictors:** Include humidity, wind speed, atmospheric pressure, cloud cover, and sea surface temperature data.
3. **Seasonal Analysis:** Develop monthly or seasonal models rather than annual models to capture intra-annual variability.
4. **Real-time Updates:** Update temperature projections using actual observed trends for better future forecasts.
5. **Ensemble Methods:** Use ensemble approaches that combine multiple models and incorporate uncertainty estimates.
6. **Extreme Event Modeling:** Specialized models or anomaly detection techniques for extreme rainfall events.

7 LEARNING OUTCOMES

The internship provided extensive practical experience in various aspects of Machine Learning and data science, specifically applied to climate prediction.

7.1 Technical Skills Acquired

- **Data Science**
 - Data preprocessing and cleaning for time series data
 - Feature engineering (lag features, rolling averages, derived features)
 - Exploratory data analysis for climate data
 - Correlation analysis and feature selection
- **Machine Learning**
 - Implementation of Random Forest, XGBoost, LightGBM algorithms
 - LSTM networks for time series forecasting
 - Model hyperparameter tuning
 - Cross-validation techniques
- **Model Evaluation**
 - MAE, RMSE for regression evaluation
 - Feature importance analysis
 - Confidence interval generation
 - Model validation with observed data
- **Programming**
 - Python and essential libraries (Pandas, NumPy, Scikit-Learn)
 - TensorFlow/Keras for deep learning
 - Data visualization with Matplotlib

7.2 Domain Knowledge Gained

- Understanding of Puducherry's monsoon patterns
- Relationship between temperature and rainfall in tropical climates
- Importance of accurate rainfall prediction for agriculture and water management
- Time series analysis for climate data
- Climate change impacts on regional weather patterns

7.3 Challenges Faced and Solutions

Table 12: Challenges and Solutions

Challenge	Solution
Limited historical data (10 years)	Used feature engineering to create additional predictors; applied data augmentation techniques
Time series complexity	Implemented LSTM networks; created lag features
Model overfitting	Used cross-validation; applied regularization
Temperature-rainfall relationship variability	Incorporated rolling averages; used ensemble methods
Validation discrepancies	Identified need for additional predictors and extended training data

8 CONCLUSION AND FUTURE WORK

8.1 Summary

This internship project successfully developed a Rainfall Prediction Model for the Puducherry region using Machine Learning and Deep Learning techniques, leveraging historical temperature and rainfall data from 2011 to 2020.

The key achievements of the project include:

1. Successful implementation of four ML/DL models (Random Forest, XGBoost, LightGBM, LSTM)
2. Comprehensive feature engineering for time series prediction
3. Accurate rainfall forecasting for 2021-2030 with confidence intervals
4. Identification of temperature as a significant predictor of rainfall patterns
5. Generation of valuable insights for Puducherry's climate
6. Validation against observed 2021–2025 data, which yielded an MAE of 419.79 mm and highlighted the model's limitations in extreme years
7. Identification of a significant warming trend in observed temperatures (2023-2025) that was not captured in the projected data
8. Comprehensive comparison of predicted and observed temperature and rainfall values

8.2 Significance of the Study

The developed model provides:

- **Agricultural Benefits:** Helps farmers plan cultivation seasons
- **Water Management:** Assists in reservoir and groundwater planning
- **Disaster Preparedness:** Supports flood prediction and management
- **Policy Making:** Aids government agencies in resource allocation

8.3 Future Work

The following enhancements are proposed based on validation findings:

8.3.1 Data Enhancement

1. Integration of additional weather parameters:
 - Humidity data
 - Wind speed and direction
 - Cloud cover
 - Sea surface temperature
2. Incorporation of daily/monthly data for seasonal analysis
3. Inclusion of other meteorological indices (El Niño, IOD)
4. Extension of training data beyond 2011-2020 to capture more variability
5. Regular updates using the latest observed data

8.3.2 Model Improvements

1. Hybrid models combining different algorithms
2. Attention mechanisms in LSTM networks
3. Transformer-based architectures
4. Ensemble learning techniques
5. Specialized models for extreme event prediction
6. Uncertainty quantification methods

8.3.3 Application Development

1. Web-based dashboard for visualization
2. Mobile application for farmer alerts
3. Real-time data integration from IMD
4. Geographic Information System (GIS) integration

8.4 Acknowledgments

I express my heartfelt gratitude to:

- The faculty members of the Department of Computer Science and Engineering, PKIET, for their invaluable guidance and support
- The research scholars and staff who provided assistance with data collection
- The India Meteorological Department for making weather data available
- My family and friends for their encouragement throughout this internship

The practical exposure gained during this project has been invaluable for my academic and professional development. It has deepened my understanding of applying machine learning to real-world problems, particularly in the critical domain of climate prediction.

9 REFERENCES

1. India Meteorological Department (IMD). Available: <https://mausam.imd.gov.in/>
2. Puducherry State Government Data. Available: <https://www.py.gov.in/>
3. Scikit-Learn Documentation. Available: <https://scikit-learn.org/stable/>
4. TensorFlow Documentation. Available: <https://www.tensorflow.org/>
5. XGBoost Official Documentation. Available: <https://xgboost.readthedocs.io/>
6. LightGBM Official Documentation. Available: <https://lightgbm.readthedocs.io/>
7. Pandas Documentation. Available: <https://pandas.pydata.org/docs/>
8. NumPy Documentation. Available: <https://numpy.org/doc/>
9. Brownlee, J., "Deep Learning for Time Series Forecasting", Machine Learning Mastery. Available: <https://machinelearningmastery.com/>
10. Wang, H., Liu, Y., and Chen, J., "Rainfall Prediction Using Machine Learning Techniques," International Journal of Advanced Computer Science and Applications, 2023.
11. World Meteorological Organization (WMO). Available: <https://wmo.int/>
12. Goodfellow, I., Bengio, Y., and Courville, A., "Deep Learning," MIT Press, 2016.
13. Géron, A., "Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow," O'Reilly Media, 2019.
14. Murphy, K.P., "Machine Learning: A Probabilistic Perspective," MIT Press, 2012.
15. Hyndman, R.J., and Athanasopoulos, G., "Forecasting: Principles and Practice," OTexts, 2021.
16. Chollet, F., "Deep Learning with Python," Manning Publications, 2017.
17. Raschka, S., and Mirjalili, V., "Python Machine Learning," Packt Publishing, 2019.
18. Kumar, A. and Singh, R., "Analysis of Rainfall Patterns in Coastal Regions of India," Journal of Climate Research, 2022.
19. Rao, S., "Northeast Monsoon Rainfall Variability in South India," International Journal of Climatology, 2021.
20. Saha, S., et al., "The NCEP Climate Forecast System Version 2," Journal of Climate, 2014.
21. Rajeevan, M., and Nair, R., "Monsoon Variability and Climate Change," Springer, 2021.
22. Chattopadhyay, S., "Machine Learning in Meteorology," Springer, 2022.