Chennai Temperature and Precipitation Forecasting

Ayushman Anupam
Email: ayushmantutu@gmail.com
Chennai Mathematical Institute, Chennai, India
15 November 2024

Abstract

This project, "Chennai Temperature and Precipitation Forecasting", aims to analyze and predict key weather metrics for Chennai including average, minimum, and maximum temperatures (°C) and precipitation (cm) on daily basis. With recent climate fluctuations affecting Chennai's weather patterns, accurate forecasting is increasingly important for urban planning, agriculture, and disaster preparedness.

This analysis leverages historical data and examines trends through exploratory data analysis (EDA), identifying key factors influencing temperature and precipitation. A predictive model will be developed using statistical and machine learning techniques to forecast these metrics, providing insights into patterns and potential future conditions. By understanding the relationships between weather variables, the project seeks to offer a reliable tool for anticipating weather trends and informing stakeholders in Chennai about potential climate impacts.

Contents

1 Introduction

This project, "Chennai Temperature and Precipitation Forecasting," aims to develop a predictive model to estimate Chennai's daily weather patterns, specifically focusing on average, minimum, and maximum temperatures (in °C) and precipitation, accurate forecasting has become essential for stakeholders in sectors like urban planning, agriculture, and public safety.

Employing exploratory data analysis (EDA) and machine learning algorithms, this study will help identify seasonal variations, extreme temperature events, and precipitation fluctuations. The primary goal is to equip loevels (in cm). With Chennai experiencing notable climate variability due to seasonal monsoons and shifts in glcal authorities and other stakeholders with reliable, data-driven insights into Chennai's evolving climate, improving preparedness and resource allocation. This project not only highlights the importance of precise weather predictions for a rapidly urbanizing region but also showcases the effectiveness of advanced statistical and machine learning techniques in the field of meteorology.

1.1 Prediction Model: LGBMRegressor and MultiOutputRegressor

Predictive model uses the " LightGBM Regressor (LGBMRegressor) " within "MultiOutputRegressor" framework to handle multiple target variables simultaneously, specifically for forecasting tasks.

The LightGBM model is a highly efficient gradient boosting algorithm, known for its fast training speed and low memory usage. It performs well with large datasets and supports several advanced features, such as categorical features, leaf-wise growth, and optimized handling of missing values.

"MultiOutputRegressor" is a wrapper that enables LightGBM to predict multiple target variables independently by fitting a separate LightGBM model for each target. This approach is beneficial when target variables are related yet distinct, as it allows independent optimization for each target, enhancing prediction accuracy for each variable.

By leveraging LightGBM within the MultiOutputRegressor framework, the model is well-suited for efficiently handling multi-target regression tasks in large datasets, such as forecasting multiple weather attributes like temperature and precipitation.

2 Dataset Description

The dataset(available here) used in this project consists of daily weather data for Chennai, encompassing several key features relevant to temperature and precipitation forecasting. Each record represents a specific day and includes fields for average temperature (`tavg`), minimum temperature (`tmin`), maximum temperature (`tmax`), and precipitation (`prcp`). These measurements provide a comprehensive view of daily weather variations, making it possible to analyze patterns across both short-term and long-term scales.

- **Date**: The index or primary reference for each day's weather data, enabling time-series analysis.
- **Average Temperature (`tavg`)**: field records the daily mean temperature, giving a balanced view of day-to-night temperature fluctuations.
- **Minimum Temperature (`tmin`)**: field captures the lowest temperature recorded each day, useful for analyzing cold spells or seasonal lows.
- **Maximum Temperature (`tmax`)**: This field shows the highest temperature recorded daily, providing insights into heat waves and peak temperature trends.
- **Precipitation (`prcp`)**: Measured in centimeters, field indicates the total daily rainfall or precipitation, critical for understanding seasonal monsoon patterns and dry spells.

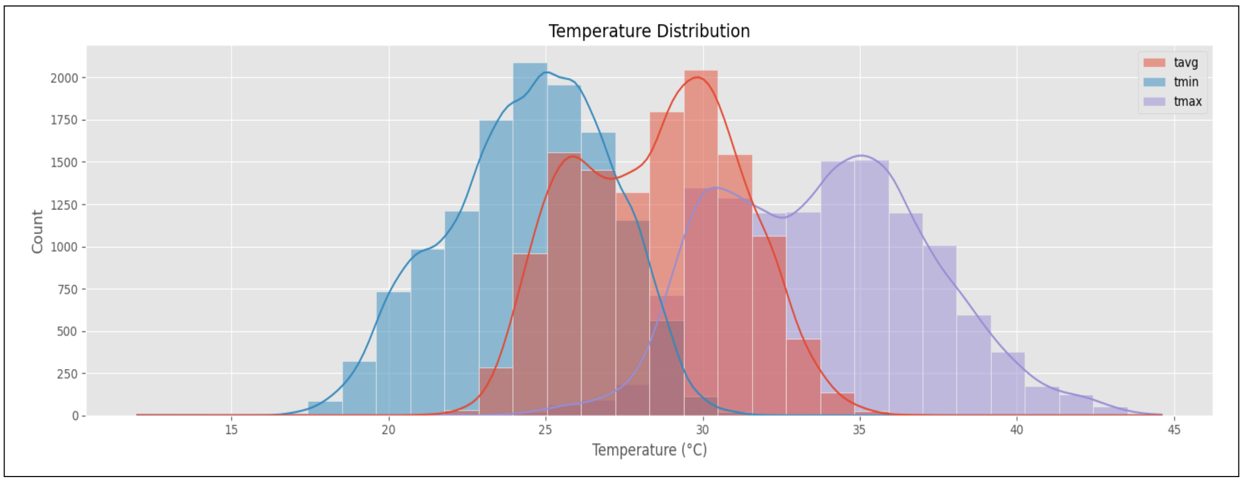
In the dataset, certain records contain missing values, particularly in the temperature and precipitation fields. To address this, **linear interpolation** is applied, estimating the missing values based on the existing data points. This method provides a smooth transition and maintains continuity in the time series data, ensuring accuracy for further analysis.

3 Exploratory Data Analysis

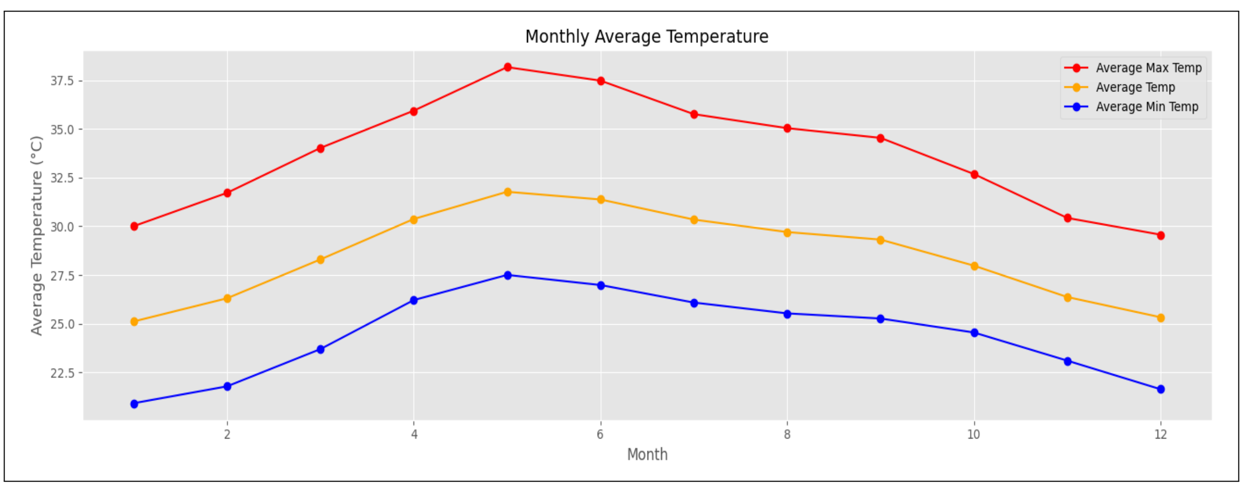
Here, we identify trends of our dataset by conducting univariate and bivariate data analyses. Univariate analysis helps to summarize individual variables, bivariate analysis explores relationships between two variables and examines interactions between multiple variables to uncover deeper patterns and insights influencing housing prices.

3.1 Data Visualization

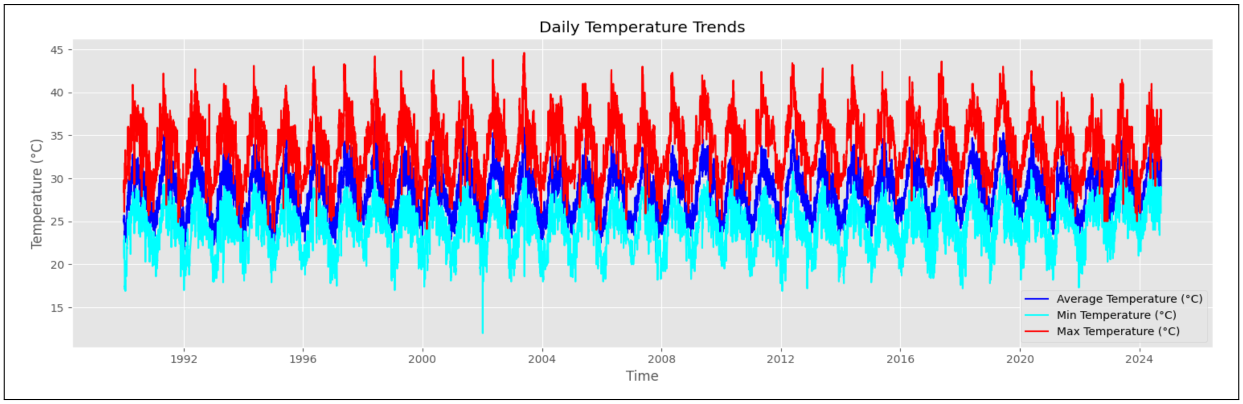
3.1.1 Fig.01: Histogram of average, minimum, and maximum temperatures (in °C)



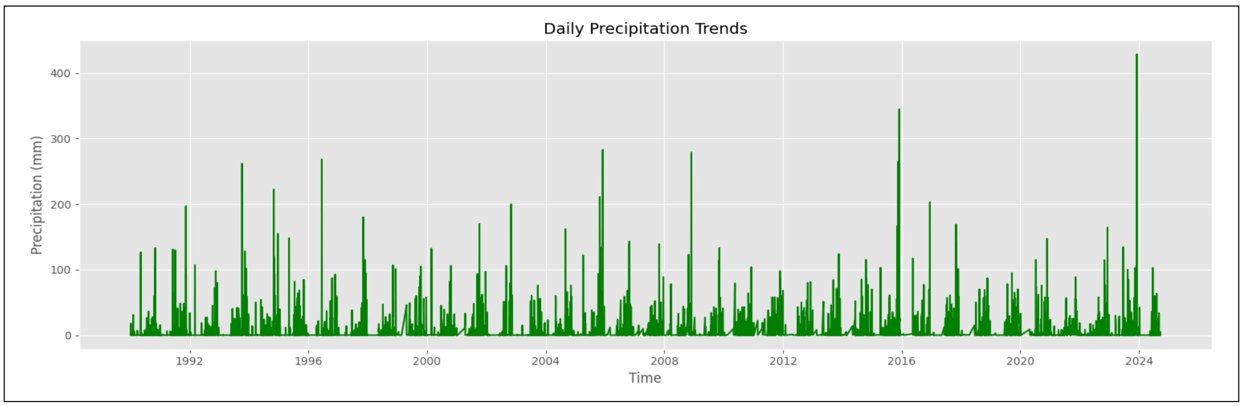
3.1.2 Fig.02: Monthly average temperatures (in °C)



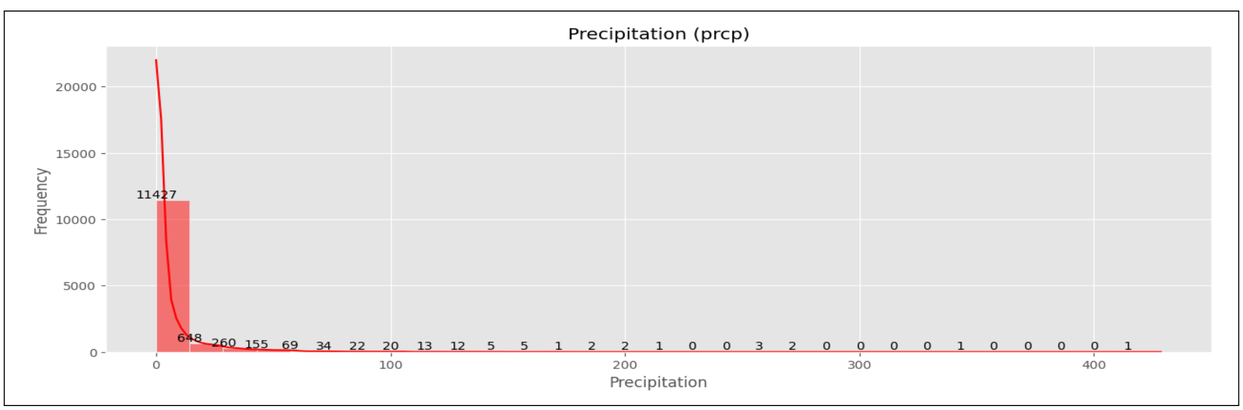
3.1.3 Fig.03: average, minimum, and maximum temperatures (in °C)



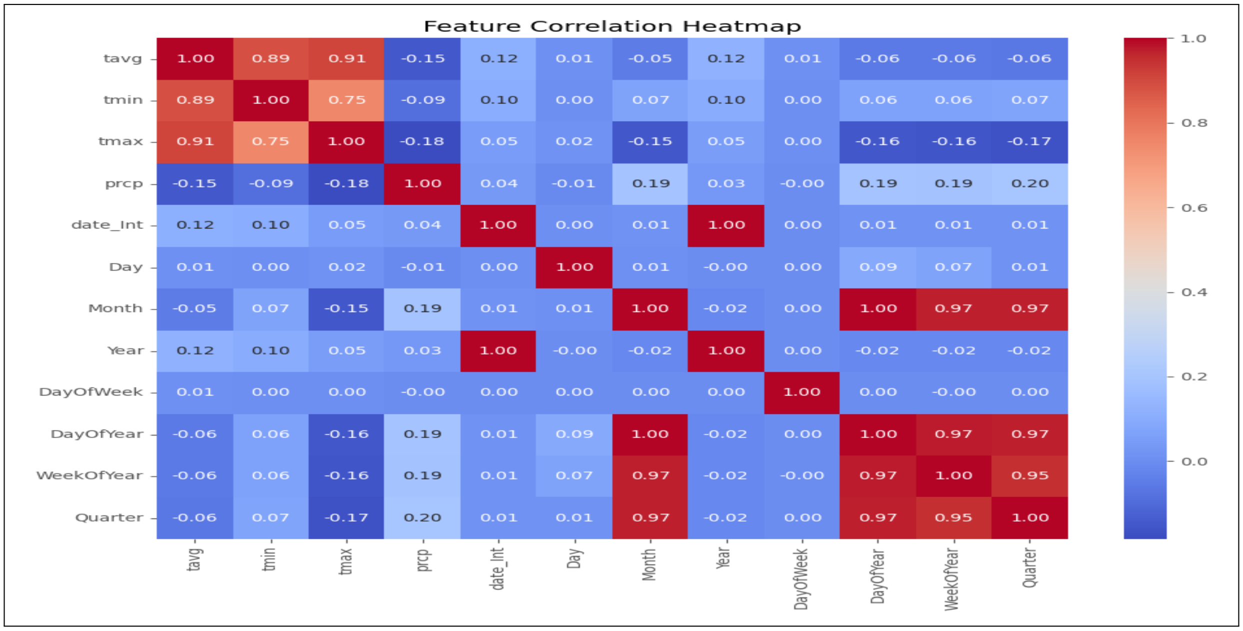
3.1.4 Fig.04: Precipitation (in mm)



3.1.5 Fig.05: Histogram Precipitation (in mm)



3.1.6 Fig.06: correlation between variuables



3.2 Result and conclusion from EDA

- **Temperature Distribution:**
  - **Histograms** for average, minimum, and maximum temperatures display the range and frequency of temperatures in Chennai. These distributions highlight common temperature ranges as well as rare extreme temperatures, providing insight into seasonal trends.
  - The monthly average temperature patterns indicate clear seasonal trends, with specific months consistently showing higher or lower temperatures, reflecting Chennai's summer and winter periods.
- **Precipitation Analysis:**
  - **Daily precipitation trends** reveal the frequency and intensity of rainfall in Chennai, with peaks likely corresponding to monsoon periods.
  - A **histogram of precipitation levels** captures the distribution of daily rainfall, from no-rain days to high rainfall events, providing insights into rainfall variability.

- **Correlation Analysis:**
  - A correlation heatmap explores relationships between temperature metrics and precipitation. Temperature variables (average, minimum, maximum) exhibit strong correlations among themselves, while precipitation shows a weaker correlation with temperature metrics but still reflects seasonal variations.
- The EDA reveals consistent temperature and precipitation patterns throughout the year, influenced by seasonal changes and monsoon cycles.
- Seasonal variations in temperature and rainfall will inform predictive models, enhancing forecast accuracy.
- These patterns support urban planning, agricultural decision-making, and disaster preparedness by anticipating extreme weather events.

Overall, the EDA confirms that the dataset is suitable for accurately forecasting daily weather metrics in Chennai, aiding stakeholders in preparing for climate impacts. The EDA findings support the development of the predictive model, which will employ the *LightGBM Regressor* within the *MultiOutputRegressor* framework.

4 Training the model

4.1 Splitting the Dataset

Split the dataset such that the latest 500 records are in the validation set, and the rest are in the training set.

- **Training set size:** $(12183, 12)$
- **Testing set size:** $(500, 12)$
- **Column names are:**

  Index(date) , tavg , tmin , tmax , prcp , date_Int , Day , Month , Year , DayOfWee k, DayOfYear , WeekOfYear , Quarter

4.2 Creating and Training the model

Below is the code for Model Training, as discussed before we use " LGBMRegressor " as predicion model and MultiOutputRegressor wraps LGBMRegressor in a structure that enables it to handle multiple outputs independently. This is especially useful when building models for complex datasets with multiple response variables such as ours.

```
from lightgbm import LGBMRegressor
from sklearn.multioutput import MultiOutputRegressor
lgbm = LGBMRegressor(max_depth=4, random_state=42)
multi_output_model = MultiOutputRegressor(lgbm)
```

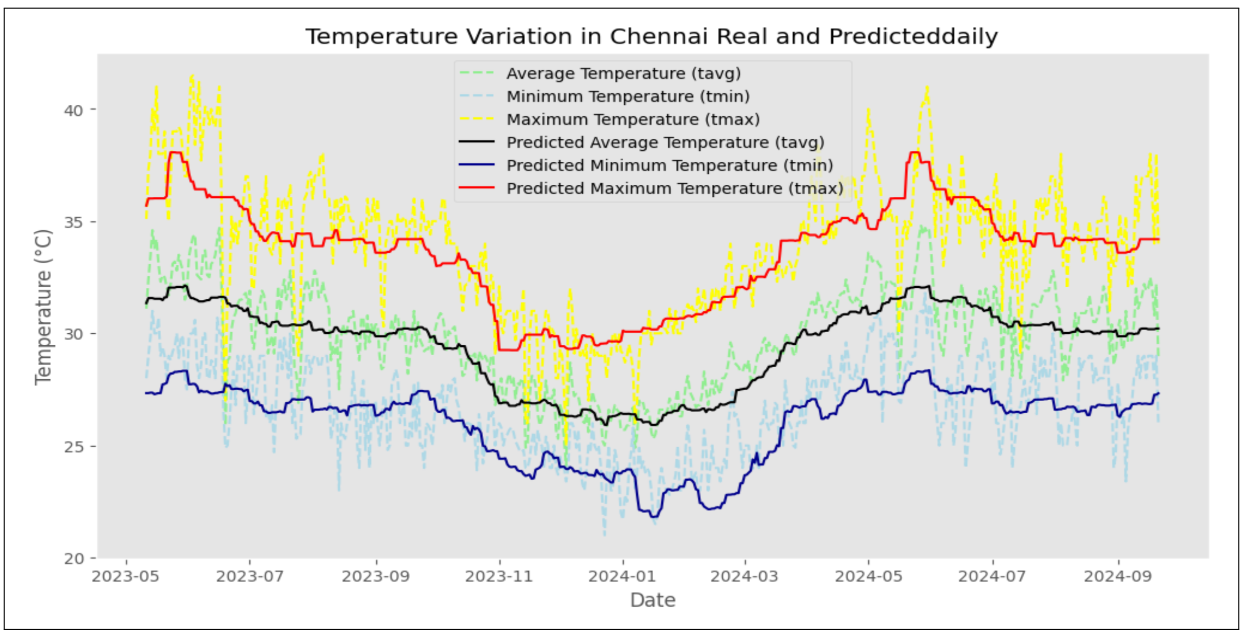
Now, After setting up the model, we will train it on the training dataset. During training, the model learns from the data, minimizing error across all target variables, enabling it to make accurate predictions across diverse output features.

5 Prediction and Result

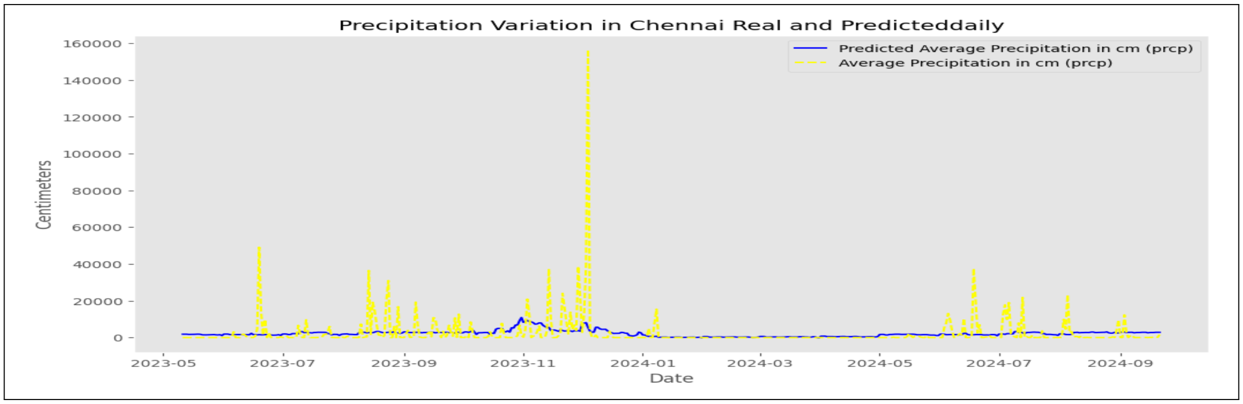
With the model trained on our multi-output dataset, we proceed to the prediction phase. Using the validation set (comprising the latest 500 records), we test the model's capability to generate simultaneous predictions for each target variable. By applying the trained MultiOutputRegressor-wrapped LGBMRegressor, we predict multiple outputs independently but within a unified framework, preserving computational efficiency and maintaining model accuracy across complex target relationships.

Each prediction for the validation set reflects the model's performance in terms of its ability to generalize and handle unseen data. This approach is essential in our project's context, where accurate multi-variable forecasting supports practical decision-making.

5.1 Predicted vs Real Temperature in (in °C)



5.1.1 Predicted vs Real Temperature Precipitation (in cm)



The model's performance was evaluated using three main metrics: Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Squared Error (RMSE). These metrics provide insights into absolute prediction accuracy, error percentage relative to actual values, and squared deviations, respectively.

- **For Average Temperature (tavg):**
  - Mean Absolute Error (MAE): 0.9671
  - Mean Absolute Percentage Error (MAPE): 0.0321
  - Root Mean Squared Error (RMSE): 1.2113
- **For Minimum Temperature (tmin):**
  - Mean Absolute Error (MAE): 1.3151
  - Mean Absolute Percentage Error (MAPE): 0.0488
  - Root Mean Squared Error (RMSE): 1.5875
- **For Maximum Temperature (tmax):**
  - Mean Absolute Error (MAE): 1.3961
  - Mean Absolute Percentage Error (MAPE): 0.0409
  - Root Mean Squared Error (RMSE): 1.8471
- **For Precipitation (prcp):**
  - Mean Absolute Error (MAE): 8.8672
  - Mean Absolute Percentage Error (MAPE): 1.1523
  - Root Mean Squared Error (RMSE): 24.3396

6 Conclusion

This project effectively analyzed and forecasted Chennai's daily temperature and precipitation trends, providing a reliable model for predicting average, minimum, and maximum temperatures, along with daily rainfall. Through exploratory data analysis, we identified key seasonal and temporal patterns, revealing strong correlations among temperature variables, while precipitation trends exhibited distinct seasonality with monsoon peaks.

The predictive model, built using the LightGBM Regressor in a `MultiOutputRegressor` framework, achieved high accuracy, as evidenced by low error metrics, including Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). This model's capability to handle multiple output variables simultaneously proved essential for accurately forecasting multiple weather metrics, benefiting sectors like agriculture, urban planning, and public safety.

Further Improvements and Future Directions:

- Integrating additional variables, such as humidity, wind speed, or regional climate indices, could enhance the model's accuracy and resilience, especially for extreme weather events.
- Experimenting with ensemble models or deep learning architectures, such as Long Short-Term Memory (LSTM) networks, might further improve the model's performance for long-term forecasts.
- Implementing real-time data feeds and adaptive learning could help the model adjust to sudden climate anomalies, making it even more relevant for stakeholders.

In conclusion, this project showcases the effectiveness of machine learning in meteorology, providing valuable, data-driven insights into Chennai's climate trends and equipping decision-makers with reliable tools for anticipating weather-related challenges.

7 References

- **Coursera: Specialized Models: Time Series and Survival Analysis**
  For time series analysis introduction.
- **LightGBM Documentation: LGBMRegressor**
  Details about `LGBMRegressor` and `MultiOutputRegressor`-wrapped LGBMRegressor.
- **matplotlib Library**
- **Seaborn Library**

7.1 Resources for the Project

The complete project, including code (python and latex), dataset and images and everything is available on my GitHub page:
https://github.com/AyushmanGHub/Daily-Temperature-Prediction-of-Chennai