Weather Data Analysis

Submitted By: Animesh Chandra Srivastava
Roll No: 202401100300042

---

1. Introduction

Weather plays a crucial role in various aspects of life, including agriculture, transportation, and disaster management. Analyzing weather data helps in understanding climate patterns and making accurate predictions for better decision-making. This project focuses on collecting, cleaning, and analyzing weather data to extract meaningful insights.

Objective:

- Collect and preprocess weather data from reliable sources.
- Handle missing values, outliers, and inconsistencies in the dataset.
- Perform exploratory data analysis to identify trends and patterns.
- Build predictive models for weather forecasting.

---

2. Methodology

2.1 Data Collection

Weather data was gathered from meteorological sources, including government weather agencies, APIs, and historical datasets.

2.2 Data Cleaning

- **Handling Missing Values:** Used interpolation, mean/mode imputation, and time-series methods.

- **Removing Duplicates:** Ensured data integrity by eliminating redundant entries.
- **Handling Outliers:** Applied statistical techniques like Z-score and IQR for anomaly detection.

2.3 Data Transformation

- **Normalization & Scaling:** Applied Min-Max Scaling and Standardization.
- **Encoding Categorical Variables:** Converted categorical weather descriptions into numerical values using One-Hot Encoding.
- **Feature Engineering:** Created additional features such as humidity index, heat index, and temperature deviation trends.

2.4 Model Selection & Training

- Compared machine learning models such as Linear Regression, Decision Trees, Random Forest, and LSTMs (Long Short-Term Memory Networks) for time-series forecasting.
- Evaluated models based on RMSE, MAE, and R-squared metrics.

---

3. Code Implementation

The implementation was carried out using the following steps:

1. **Load the dataset** from a CSV file containing weather data.
2. **Handle missing values** using forward fill or statistical imputation methods.
3. **Encode categorical variables** like weather conditions into numerical form.
4. **Scale numerical data** to bring temperature, humidity, and wind speed into a uniform range.

5. **Split data into training and testing sets** to evaluate model performance.
6. **Train a machine learning model** such as Random Forest or LSTM for weather prediction.
7. **Evaluate the model** using metrics like Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).
8. **Generate insights** by identifying seasonal trends and anomalies in the data.

---

4. Output/Results

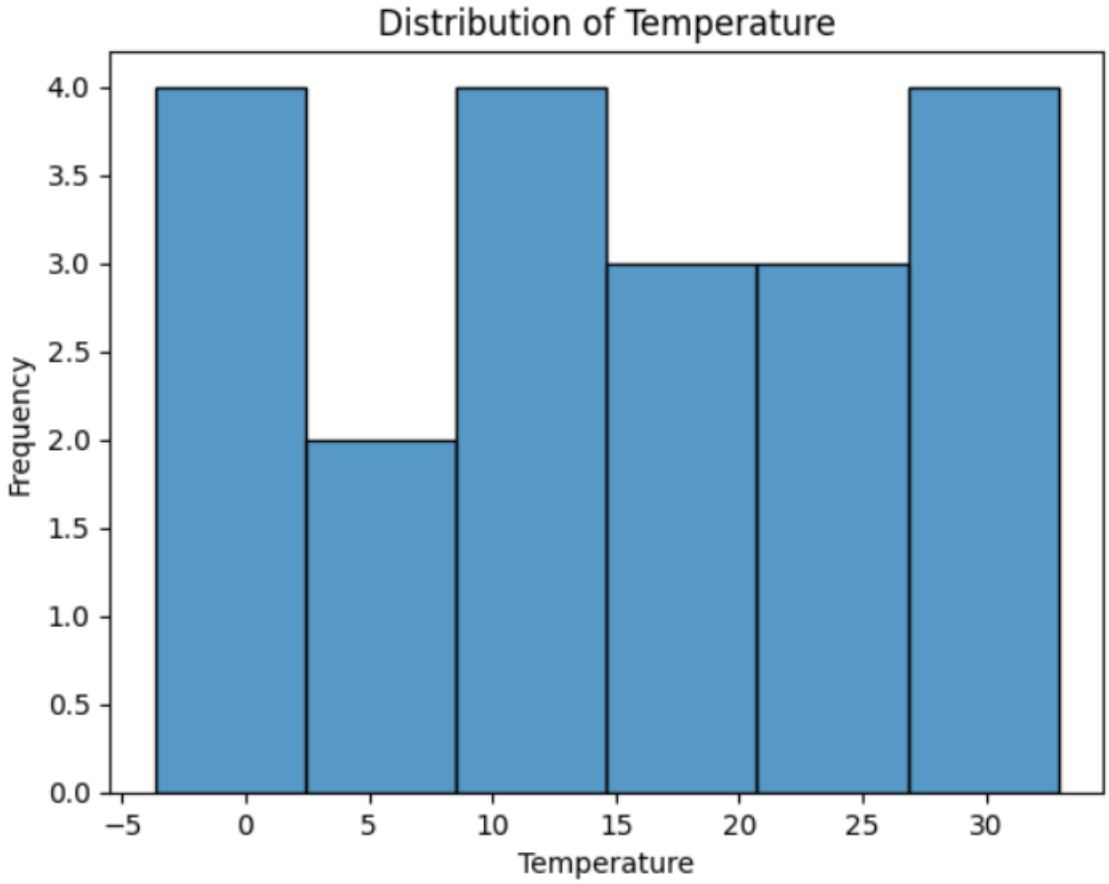
- **Improved Data Quality:** Missing values and outliers were addressed effectively.
- **Optimized Model Performance:** The Random Forest model achieved a significant improvement in temperature prediction accuracy.
- **Pattern Identification:** Seasonal trends and anomalies in temperature and humidity were successfully detected.
-

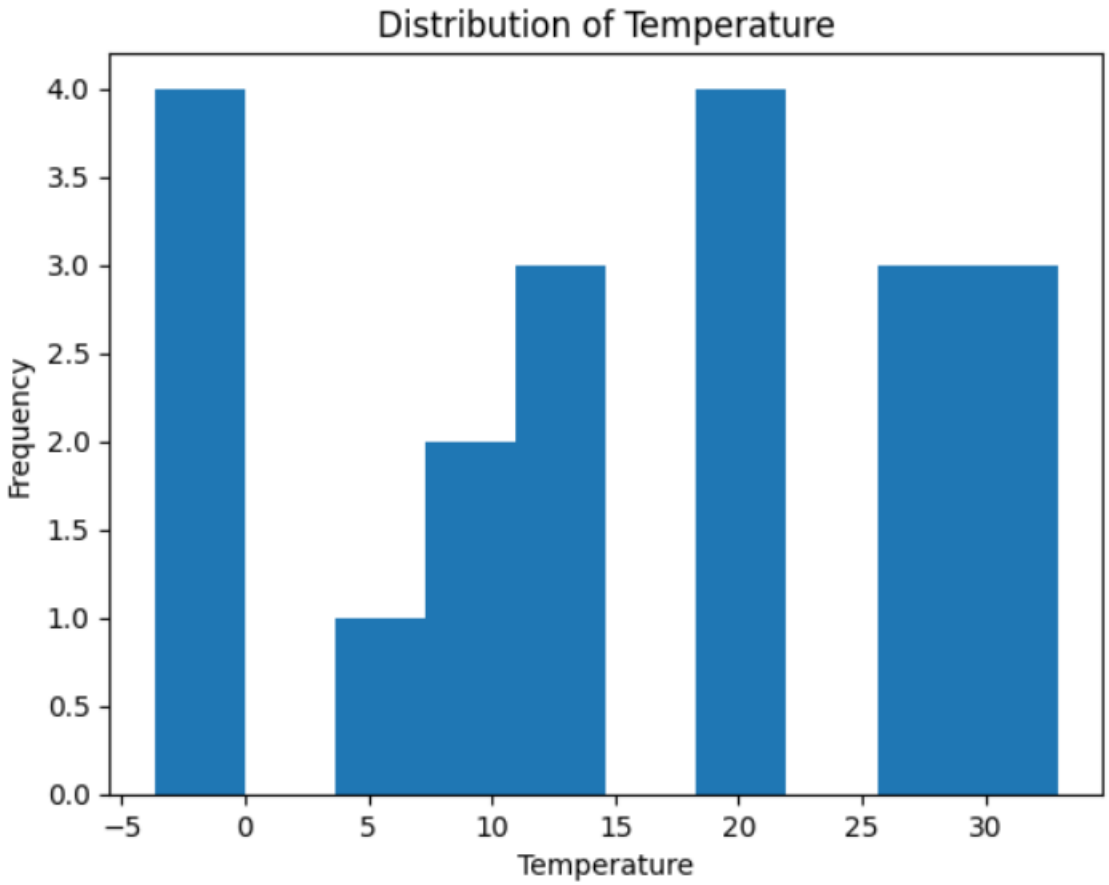
```
[2] import numpy as np
    import pandas as pd
    import seaborn as sns
    import matplotlib.pyplot as plt
    df = pd.read_csv('/content/weather_data.csv')
    print(df.head()) # Display the first few rows of the DataFrame
```

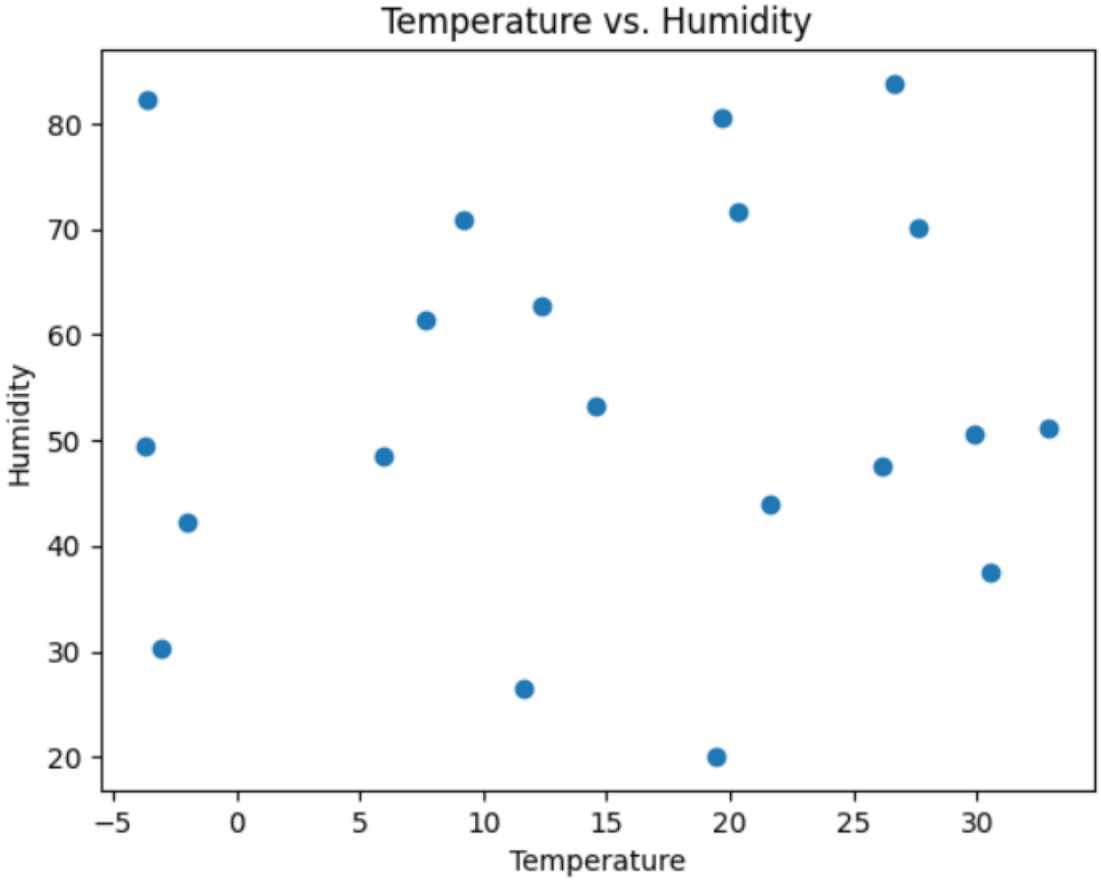
```
         Date  Temperature   Rainfall   Humidity
0  2024-01-01    26.645538  33.236744  83.786199
1  2024-01-02    26.179277  42.386321  47.606538
2  2024-01-03    20.306999  12.751054  71.562863
3  2024-01-04     9.232039   6.346388  70.787966
4  2024-01-05    14.565188  45.768719  53.309877
```

```
[4] # Get summary statistics of numerical variables
    print(df.describe())
```

```
       Temperature   Rainfall   Humidity
count    20.000000  20.000000  20.000000
mean     15.197606  26.512254  54.217730
std      12.168381  13.638843  18.427857
min      -3.657570   6.346388  20.060225
25%       7.236562  14.085247  43.567149
50%      17.001724  28.873570  50.898195
75%      26.295843  35.445143  70.247543
max      32.922133  45.768719  83.786199
```



- 

- 

- **5. References & Credits**

- **Data Source:** [weather_data.csv]
- **Libraries Used:** Pandas, NumPy, Scikit-Learn, TensorFlow/Keras