Weather Data Analysis

Name: Vishal Singh
Roll No: 202401100300281
Date: 11 March 2025

Introduction

Understanding weather patterns is vital for predicting climate changes, preparing for extreme events, and supporting industries like agriculture, transportation, and disaster response. This report explores historical weather data, including temperature, rainfall, and humidity, to uncover trends and relationships that inform decision-making in climate-sensitive sectors.

Methodology

1. **Data Collection**
   The dataset comprises historical records with daily measurements of temperature (°C), rainfall (mm), humidity (%), and corresponding dates.
2. **Data Preprocessing**
   - **Handling Missing Values**: Gaps in the data were addressed by interpolation or removal to ensure accuracy.
   - **Date Formatting**: Dates were converted into a standardized format for time-based analysis.
   - **Data Type Checks**: Variables were verified for consistency (e.g., numerical values for temperature).
3. **Analysis & Visualization**
   - **Trend Analysis**: Line charts illustrated how temperature, rainfall, and humidity fluctuated over time.

- **Distribution Insights**: Histograms and box plots revealed data spread and highlighted outliers, such as unusually heavy rainfall days.
- **Relationship Mapping**: Heatmaps and scatter plots explored correlations between variables (e.g., temperature vs. humidity).

Code

```
import pandas as pd


# Load the CSV file
file_path = "weather_data.csv"
weather_df = pd.read_csv(file_path)


# Display basic information and the first few rows
weather_df.info(), weather_df.head()
# Summary statistics
summary_stats = weather_df.describe()


weather_df["Date"] = pd.to_datetime(weather_df["Date"])  #
Convert Date column to datetime


# Correlation matrix
correlation_matrix = weather_df.corr()


summary_stats, correlation_matrix
import matplotlib.pyplot as plt
import seaborn as sns


# Set style for better visuals
```

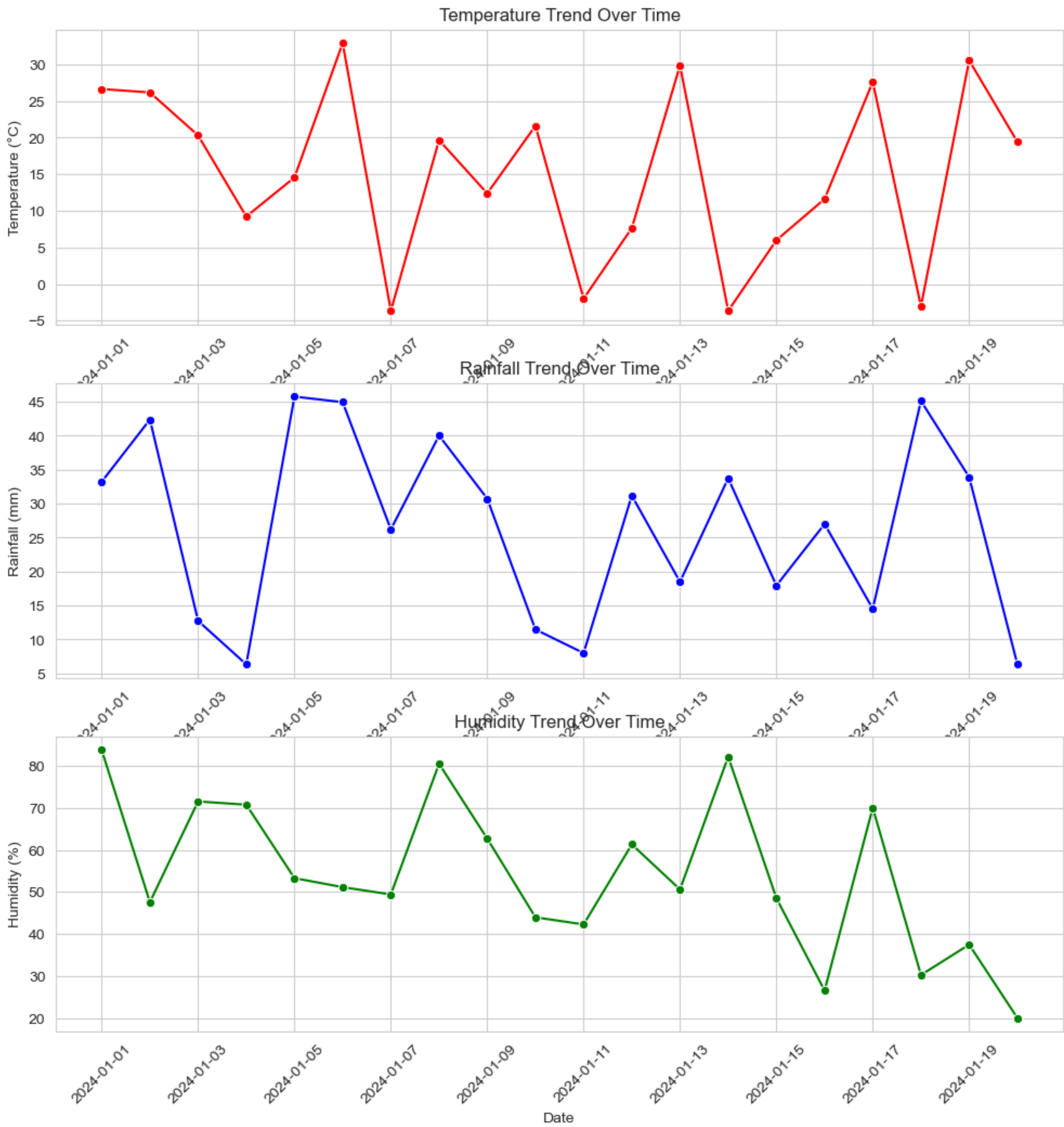
```
sns.set_style("whitegrid")

# Create subplots
fig, axes = plt.subplots(3, 1, figsize=(12, 12))

# Temperature trend
sns.lineplot(ax=axes[0], x=weather_df["Date"],
y=weather_df["Temperature"], marker="o", color="red")
axes[0].set_title("Temperature Trend Over Time")
axes[0].set_xlabel("Date")
axes[0].set_ylabel("Temperature (°C)")
axes[0].tick_params(axis='x', rotation=45)

# Rainfall trend
sns.lineplot(ax=axes[1], x=weather_df["Date"],
y=weather_df["Rainfall"], marker="o", color="blue")
axes[1].set_title("Rainfall Trend Over Time")
axes[1].set_xlabel("Date")
axes[1].set_ylabel("Rainfall (mm)")
axes[1].tick_params(axis='x', rotation=45)

# Humidity trend
sns.lineplot(ax=axes[2], x=weather_df["Date"],
y=weather_df["Humidity"], marker="o", color="green")
axes[2].set_title("Humidity Trend Over Time")
```

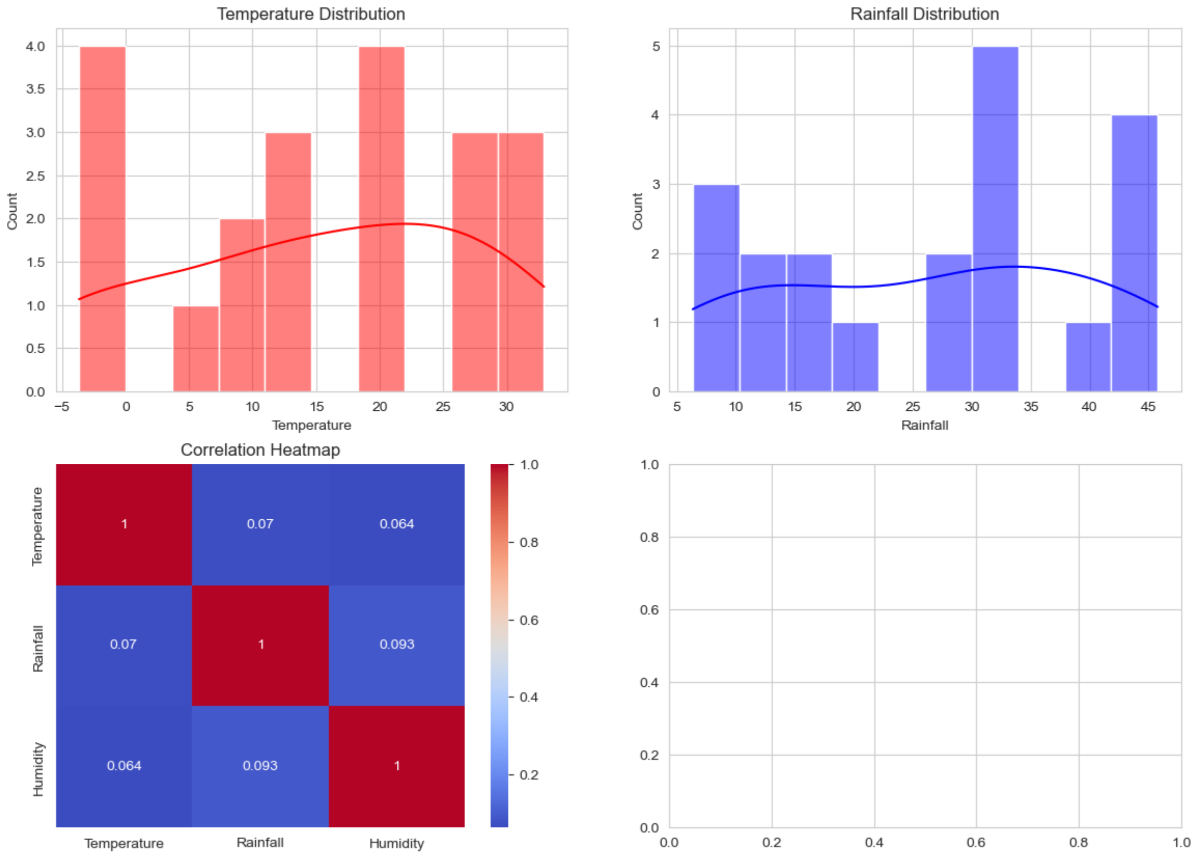
```
axes[2].set_xlabel("Date")
axes[2].set_ylabel("Humidity (%)")
axes[2].tick_params(axis='x', rotation=45)

plt.show()
# Create subplots for different types of visualizations
fig, axes = plt.subplots(2, 2, figsize=(14, 10))

# Histogram for Temperature
sns.histplot(weather_df["Temperature"], bins=10, kde=True,
color="red", ax=axes[0, 0])
axes[0, 0].set_title("Temperature Distribution")

# Histogram for Rainfall
sns.histplot(weather_df["Rainfall"], bins=10, kde=True,
color="blue", ax=axes[0, 1])
axes[0, 1].set_title("Rainfall Distribution")

# Heatmap for correlation
sns.heatmap(weather_df.drop(columns=["Date"]).corr(),
annot=True, cmap="coolwarm", ax=axes[1, 0])
axes[1, 0].set_title("Correlation Heatmap")

plt.show()
```

Outputs

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 4 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   Date         20 non-null     object 
 1   Temperature  20 non-null     float64
 2   Rainfall     20 non-null     float64
 3   Humidity     20 non-null     float64
dtypes: float64(3), object(1)
memory usage: 772.0+ bytes

(None,
          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)
```

```
(       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,
                  Date  Temperature  Rainfall  Humidity
 Date         1.000000    -0.159054 -0.185458 -0.544355
 Temperature -0.159054     1.000000  0.069556  0.064469
 Rainfall    -0.185458     0.069556  1.000000  0.092529
 Humidity    -0.544355     0.064469  0.092529  1.000000)
```


Temperature Trend Over Time
Temperature (°C)
30
25
20
15
10
5
0
−5
Rainfall Trend Over Time
Rainfall (mm)
45
40
35
30
25
20
15
10
5
Humidity Trend Over Time
Humidity (%)
80
70
60
50
40
30
20
2024-01-01
2024-01-03
2024-01-05
2024-01-07
2024-01-09
2024-01-11
2024-01-13
2024-01-15
2024-01-17
2024-01-19
Date


Temperature Distribution
Count
Temperature
Rainfall Distribution
Count
Rainfall
Correlation Heatmap
Temperature
Rainfall
Humidity
1
0.07
0.064
0.07
1
0.093
0.064
0.093
1

References

- **Python Documentation:** https://docs.python.org/
- **Pandas Library:** https://pandas.pydata.org/
- **Seaborn Library:** https://seaborn.pydata.org/
- **Matplotlib Library:** https://matplotlib.org/