Weather data analysis

Owner	Aditya Sikarwar
Instructor	BIKKI kumar
Subject	AI(artificial introduction)

Date

Introduction

- Weather data analysis plays a crucial role in understanding climate patterns, predicting future weather conditions, and making informed decisions in agriculture, transportation, and disaster management. This project aims to analyze weather data using Python to identify trends, patterns, and correlations in meteorological parameters.

Date

Methodology

The methodology followed in this project consists of the following steps:

4.1 Data Acquisition

- The dataset was collected from reliable meteorological sources such as OpenWeatherMap, Kaggle, or government databases.
- Data was extracted in CSV format for further processing.

4.2 Data Preprocessing

- filled with appropriate statistics (mean, median, etc.).
- **Converting Data Types:** The 'date' column was converted to datetime format.
- **Outlier Detection:** Box plots were used to detect anomalies in temperature and humidity values.
- **Normalization:** Data was normalized where necessary to ensure uniformity.

4.3 Exploratory Data Analysis (EDA)

- Data was summarized using descriptive statistics to understand distributions and trends.
- Visualization techniques such as histograms, scatter plots, and time series analysis were used to identify patterns.

4.4 Data Analysis and Visualization

Several analytical techniques and visualizations were used to explore the dataset:

4.4.1 Temperature Trends Over Time

A line chart was used to observe temperature fluctuations over a specific period. The analysis revealed seasonal variations and possible long-term trends.

4.4.2 Correlation Between Weather Parameters

A heatmap of the correlation matrix was generated to analyze relationships between variables. Key observations:

- Temperature and humidity showed a moderate negative correlation.
- Wind speed had little to no correlation with temperature.
- Precipitation was positively correlated with humidity.

4.4.3 Seasonal Analysis

The dataset was grouped by months to identify seasonal trends. Summer months showed higher temperatures, while winter months exhibited lower temperatures and increased precipitation.

4.4.4 Anomaly Detection

Using statistical methods such as Z-score and IQR, extreme values were identified, indicating potential unusual weather events.

References and credits

- **Handling Missing Values:** Rows with missing values were either removed or
- OpenWeatherMap API: https://openweathermap.org/api
- Kaggle Datasets: https://www.kaggle.com/
- Python Pandas Documentation: https://pandas.pydata.org/docs/
- Seaborn Documentation: https://seaborn.pydata.org/
- Matplotlib Documentation: https://matplotlib.org/stable/contents.html

---

Code

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv("weather_data.csv")
print(df.head())  # Display first few rows

print(df.isnull().sum())  # Count missing values
df = df.dropna()  # Remove missing values if necessary
print(df.describe())
print(df.columns)
df.columns = df.columns.str.strip()  # Remove leading/trailing spaces
df.columns = df.columns.str.lower()  # Convert to lowercase
print(df.columns)  # Check again
df.rename(columns={"Date": "date"}, inplace=True)
plt.figure(figsize=(10, 5))
sns.lineplot(x=df.index, y=df['temperature'], label="Temperature")
plt.xlabel("Date")
plt.ylabel("Temperature (°C)")
plt.title("Temperature Trends Over Time")
plt.legend()
plt.show()
```

```
print(df.describe())  # Display mean, min, max, std deviation, etc.
plt.figure(figsize=(8, 6))
sns.heatmap(df.corr(), annot=True, cmap="coolwarm")
plt.title("Correlation Matrix")
plt.show()
```