Title: Crop Recommendation Based on Temperature and Weather Parameters

Problem Statement

In regions with dynamic weather conditions, selecting the right crop based on temperature, humidity, and rainfall is essential for maximizing agricultural productivity. This project aims to recommend the most suitable crop using machine learning techniques by analyzing environmental factors like temperature**,** humidity**,** rainfall**,** and soil pH—excluding soil nutrient content (N, P, K)**.**

Dataset Description

The dataset used is Crop_recommendation.csv with the following selected features:

Feature	Description
temperature	Average temperature (°C)
humidity	Relative humidity (%)
ph	Acidity or alkalinity of soil
rainfall	Annual rainfall (mm)
label	Recommended crop

Steps Followed

1. **Data Collection**
   The dataset for crop prediction was sourced from the ICFA India dataset, which includes features like Nitrogen (N), Phosphorus (P), Potassium (K), temperature, humidity, pH, and rainfall, with crop labels.
2. **Data Preprocessing**
   a. Handled missing values (if any) and removed inconsistencies.
   b. Encoded the categorical crop labels using `LabelEncoder`.
   c. Performed train-test split to prepare for model evaluation.
3. **Exploratory Data Analysis (EDA)**
   a. Visualized the feature distributions using heatmaps and interactive widgets.

    b. Analyzed environmental conditions associated with each crop type to understand feature relationships.
4. **Model Selection and Training**
    a. Chose Logistic Regression as the base classification model for its simplicity and efficiency.
    b. Trained the model using the training dataset.
    c. Logged training steps and model creation for traceability.
5. **Model Evaluation**
    a. Evaluated model performance using accuracy, precision, recall, and F1-score.
    b. Generated a confusion matrix to understand class-level predictions.
    c. Saved the evaluation metrics in a structured `metrics.json` file.
6. **Prediction System**
    a. Created a prediction script (`predict.py`) that loads the trained model and label encoder.
    b. Accepts new input values for N, P, K, temperature, humidity, pH, and rainfall and returns the recommended crop.
7. **Logging and Documentation**
    a. Maintained a detailed `training_log.txt` for all key stages during model building.
    b. Modularized code into reusable Python scripts for scalability and readability.

Tools & Libraries Used

1. **Language**: Python 3.8+
2. **Libraries**:
    - `pandas` – data handling
    - `scikit-learn` – modeling & evaluation
    - `matplotlib` – data visualization
    - `joblib` – model persistence
    - `numpy` – numerical computations
    - `json` – saving metrics
3. **Development Environment**:
    - Jupyter Notebook (`crop-prediction.ipynb`)
    - Python scripts (`train.py`, `predict.py`, `data_preprocessing.py`, `model_utils.py`)
    - Text logs (`training_log.txt`)
    - Metrics (`metrics.json`)