Assignment: Develop a Machine Learning Web Application Using Streamlit

Objective

Build an interactive web application that allows users to input data and get machine learning model predictions using Streamlit, a Python framework for building data apps.

Assignment Tasks

Task 1: Setup Environment

- Install Python 3.8+ and create a virtual environment.
- Install required libraries: `streamlit`, `scikit-learn`, `pandas`, `numpy`.
- Verify installation by running a simple Streamlit "Hello World" app.

Task 2: Dataset and Model Preparation

- Choose a simple classification dataset (e.g., Iris Dataset from scikit-learn).
- Load the dataset using `pandas` or directly from `sklearn.datasets`.
- Train a basic ML model (e.g., Logistic Regression or Random Forest) for classification.
- Save the trained model to a file using `joblib` or `pickle`.

Task 3: Streamlit App Development

- Create a Python script `app.py`.
- Use Streamlit to:
    - Display the app title and description.
    - Provide input widgets (e.g., sliders, dropdowns) to capture user input features corresponding to the model.
    - Load the saved model.
    - Make predictions based on user input.
    - Display the prediction result and optionally the prediction probabilities.

Task 4: Add Data Exploration Features

- Add a sidebar with simple dataset exploratory visualizations such as:
    - Histogram of features.
    - Scatter plot of feature pairs.
- Allow toggling between data exploration mode and prediction mode.

Task 5: App Styling and UX

- Use Streamlit layout features (columns, containers).
- Add helpful tooltips or markdown explanations for inputs.
- Use conditional formatting or color coding for prediction results.

Task 6: Testing

- Run the app locally using `streamlit run app.py`.
- Test with different inputs.

Deliverables

- `app.py` with the full Streamlit app code.
- Model training script and saved model file.

Evaluation Criteria

- Correctness of ML model training and prediction.
- Functional and interactive Streamlit UI.
- Quality of data input handling and output presentation.
- Inclusion of dataset exploration features.
- Code organization, comments, and readability.