Task 3: Supervised Learning - Heart Disease Prediction

Objective:

Your task is to build a supervised machine learning model that predicts the severity of heart disease using the provided dataset. This task requires predicting one of four possible classes of heart disease severity. You will preprocess the data, train multiple models, compare their accuracies using an ensemble approach, and save the final model.

Dataset Information

The **UCI Heart Disease Dataset** contains clinical records from the **Cleveland database** with 14 key attributes used to predict heart disease severity:

- **age**: Patient's age in years
- **sex**: Patient's gender (Male/Female)
- **cp**: Chest pain type (4 categories: typical angina, atypical angina, non-anginal, asymptomatic)
- **trestbps**: Resting blood pressure in mm Hg
- **chol**: Serum cholesterol in mg/dl
- **fbs**: Fasting blood sugar > 120 mg/dl (TRUE/FALSE)
- **restecg**: Resting electrocardiographic results (normal, ST-T wave abnormality, LV hypertrophy)
- **thalch**: Maximum heart rate achieved
- **exang**: Exercise-induced angina (TRUE/FALSE)
- **oldpeak**: ST depression induced by exercise relative to rest
- **slope**: Slope of the peak exercise ST segment (upsloping, flat, downsloping)
- **ca**: Number of major vessels colored by fluoroscopy (0-3)
- **thal**: Thalassemia (normal, fixed defect, reversible defect)
- **num**: **Target variable indicating the severity of heart disease** (values **0, 1, 2, 3** representing increasing severity)

Instructions:

🔹 **Part A: Preprocessing**

- Load the dataset and inspect its structure.
- Apply appropriate preprocessing techniques:
    - Handle missing values
    - Encode categorical data
    - Scale numerical features
- Ensure the dataset is clean and ready for model training.

🔹 **Part B: Data Visualization**

- Generate a **correlation heatmap** to analyze feature relationships.
- Create at least one visualization to explore the dataset distribution.

🔹 **Part C: Splitting the Data**

- Divide the dataset into **training (80%)** and **testing (20%)** sets.

🔹 **Part D: Model Training & Accuracy Comparison**

- Train the following four models:
  - **Logistic Regression**
  - **Decision Tree**
  - **Random Forest**
  - **AdaBoost**
- Use an **ensemble approach** (such as a **Voting Classifier**) to combine multiple models and improve accuracy.
- Evaluate all models using **accuracy, precision, recall, and F1-score** on the test set.
- Compare the individual models against the ensemble classifier and determine the best-performing model.
- Save the final model using **joblib** in a `.pkl` file.

🔹 **Bonus Task: Model Deployment on Streamlit (Optional, But Recommended)**

If you want to go further, deploy the model using **Streamlit** by following these steps:

1. Create a **Streamlit application (`app.py`)** that allows users to:
   - Input patient details.
   - Load the saved model (`.pkl` file) and predict the **severity** of heart disease.
   - Display the prediction results clearly.
2. Ensure the **Streamlit app** is **user-friendly and visually appealing**.

Expected Deliverables:

A **Jupyter Notebook** containing **preprocessing, model training, and evaluation**.
A saved **model file (`.pkl`)** for future deployment.
A **comparison table or graph** showcasing the accuracy of different models.
A **brief explanation** of the approach and findings.
(Bonus: Streamlit app script if attempted.)