Enhanced Heart Disease Prediction using Decision Tree Classification

1. **Objective**: The goal of this assignment is to predict the presence of heart disease using medical attributes. The analysis focuses on rigorous data preparation, exploratory data analysis (EDA), and model optimization through hyperparameter tuning.

2. Comprehensive Exploratory Data Analysis (EDA): To understand the underlying patterns and distributions of the medical data, several visualization techniques were employed:

- **Feature Distributions (Histograms):** Numerical features such as `age`, `trestbps` (blood pressure), and `chol` (cholesterol) were plotted using histograms. This revealed that age is relatively normally distributed, while cholesterol shows a significant right skew with several high-value entries.
- **Relationship Analysis (Box Plots):** Box plots were generated for each numerical feature against the target variable (`num`). This highlighted how features like `thalch` (max heart rate) tend to be lower in patients with advanced stages of heart disease.
- **Correlation Matrix:** A heatmap was used to visualize the correlation between all numerical features. A notable negative correlation was observed between `age` and `thalch`, suggesting that maximum heart rate typically decreases with age.

3. Advanced Data Preprocessing & Outlier Analysis

- **Outlier Detection:** An explicit outlier analysis was performed using the Interquartile Range (IQR) method. Features like `chol` and `trestbps` contained values significantly higher than the 1.5*IQR threshold. These outliers were capped to the upper and lower bounds to prevent them from skewing the model's decision boundaries.
- **Feature Scaling:** Although Decision Trees are generally invariant to feature scaling, **StandardScaler** was applied to all numerical features. This ensures data consistency and prepares the pipeline for potential future comparisons with distance-based algorithms like KNN or SVM.
- **Categorical Encoding:** Categorical variables (e.g., `sex`, `cp`, `slope`) were transformed using Label Encoding to be compatible with the Scikit-Learn implementation.

4. Model Implementation and Results: The dataset was split into an 80% training set and a 20% test set.

- **Base Model:** The initial Decision Tree achieved an accuracy of approximately 58%.
- **Hyperparameter Tuning:** Using `GridSearchCV`, parameters such as `max_depth`, `min_samples_split`, and `criterion` were optimized.
- **Final Metrics:** The tuned model showed improved generalization, balancing precision and recall across the different stages of heart disease.

1. Interview Questions & Answers

Q: Why are histograms and box plots essential before training a Decision Tree model?

- **Answer:** While Decision Trees can handle raw data, these visualizations are critical for data understanding. **Histograms** help identify the distribution of features like `age` or `cholesterol`, revealing if the data is skewed or has narrow ranges. **Box plots** are vital for comparing features against the target (e.g., heart disease stages); they visually confirm if a feature like `thalch` (max heart rate) actually differs between healthy and ill patients, which justifies its use as a splitting criterion in the tree.

Q: What information does a correlation matrix provide for this medical dataset?

- **Answer:** It identifies linear relationships between features. For example, a strong correlation between `age` and `thalch` suggests they might provide redundant information. Identifying highly correlated features helps in understanding feature importance and potential multicollinearity, although Decision Trees are generally robust to the latter.

---

2. Outlier Analysis

Q: You used the IQR method to cap outliers. Why is outlier handling important even for "robust" models like Decision Trees?

- **Answer:** While Decision Trees are less sensitive to outliers than linear models, extreme outliers can still force the algorithm to create unnecessary "deep" splits to isolate a single abnormal data point (like a `cholesterol` value of 500). This can lead to **overfitting**, where the model learns noise rather than general patterns. Capping (Winsorization) ensures the model focuses on the bulk of the patient population.

---

3. Feature Scaling

Q: Since Decision Trees are scale-invariant, why did you apply `StandardScaler`?

- **Answer:** Practically, there are three reasons:
    1. **Pipeline Consistency:** It is best practice to include scaling in a preprocessing pipeline so that if you decide to compare the Decision Tree with distance-based models (like **KNN** or **SVM**), the data is already prepared.
    2. **Interpretation:** Standardizing features to a mean of 0 and a standard deviation of 1 makes it easier to compare the relative "spread" of different medical units (e.g., mmHg for blood pressure vs. mg/dl for cholesterol).
    3. **Optimization:** Some implementations of related ensemble methods (like Gradient Boosting) can converge faster when features are on a similar scale.

---

4. General Model Concepts

Q: What is the difference between Label Encoding and One-Hot Encoding in the context of your medical features? (As seen in your previous report)

- **Answer:** * **Label Encoding:** Assigns a unique integer to each category (e.g., `Sex`: Male=1, Female=0). It is efficient but can imply a false mathematical order.
    - **One-Hot Encoding:** Creates new binary columns for each category (e.g., `ChestPain_Type1`, `ChestPain_Type2`). This is safer for nominal data where no order exists, preventing the model from thinking "Type 3" is "greater than" "Type 1".

Q: How does `max_depth` help in preventing the tree from simply "memorizing" the training data?

- **Answer:** `max_depth` limits how many levels the tree can grow. Without it, a tree will continue splitting until every leaf is "pure" (contains only one class), which often captures random noise in the training set that won't exist in new patient data. Restricting depth forces the model to find more general, high-level rules.