Heart Disease

Team: Mohamed Elsabah, Bassam Kamal
Heart disease – Dataset: Heart Disease (UCI), https://archive.ics.uci.edu/dataset/45/heart+disease

Problem and Motivation

Cardiovascular disease is one of the leading causes of mortality globally. Early detection can help clinicians treat patients more effectively, prioritize diagnostic testing, and communicate risk to patients early. Our project investigates classic and neural models on the UCI Cleveland Heart Disease dataset to classify whether heart disease is present, and predict maximum heart rate achieved (thalach, bpm). This comparison will clarify which approach would be best based on clinical data.

Dataset Description

We use the UCI Machine Learning Repository "Heart Disease" dataset containing 303 patient records with 14 commonly used attributes. The original database has 70 attributes, but published experiments typically use the 14 attribute subset. The dataset includes mixed numeric and categorical variables; for example, age, sex, chest pain type, resting blood pressure, cholesterol, fasting blood sugar, resting ECG, max heart rate, exercise-induced angina, ST depression, ST slope, number of major vessels, and thalassemia test result.

Our source: Janosi, A., Steinbrunn, W., Pfisterer, M., & Detrano, R. (1989). Heart Disease [Dataset]. UCI Machine Learning Repository. https://doi.org/10.24432/C52P4X.

Tasks

Classification: (binary) label = 1 if original num > 0, else 0 (if disease is present vs not).

Explanation: Feasible because the label is provided for all patients and the dataset contains a mix of categorical and continuous clinical features (age, chest pain type, chol, etc).

Regression: (continuous) thalach (maximum heart rate achieved: bpm).

Explanation: Feasible because this feature is numeric, measured consistently across patients, and closely tied to cardiovascular health.

Metrics Plan

Classification: Report Accuracy and F1-score (macro).

Regression: MAE and RMSE were used for validation and testing.

Data split: Single stratified 60/20/20 train/validation/test split with a fixed random seed, reused for both tasks for fair comparisons.

Baseline Plan (Classical ML)

Classification baselines: Logistic Regression, Decision Tree Classifier.

Regression baselines: Linear Regression (OLS), Decision Tree Regressor.

Reproducibility Plan (Dependencies + MLflow)

- **Pinned deps:** numpy, pandas, scikit-learn, matplotlib, mlflow.
- **Tracking:** MLflow local tracking with parameters, metrics, and artifacts.
- **Entry points:**
    - python src/train_baselines.py (trains/logs classical models for both tasks).
    - python src/train_nn.py (simple feed-forward neural network to train/log for both tasks).
    - python src/evaluate.py (loads artifacts and produces metrics table).

Risks & Limitations

The dataset is small (303 rows), which increases the chance of overfitting, especially with neural networks. There is also a slight class imbalance, so accuracy alone may be misleading, making F1-score and ROC-AUC important. Finally, missing values in a few columns need careful handling to avoid biased results.

Table 1 - Dataset Snapshot

Item	Value
Rows	303
Columns	14 (13 features + 1 target)
Targets	Classification: Heart disease present vs not Regression: thalach (maximum heart rate achieved)
% missing	ca: 4 rows (1.32%) thal: 2 rows (0.66%) others: 0 rows
Class distribution	Present = 1: 165 (54.5%) Absent = 0: 138 (45.5%)

Table 2 - Planned Models and Metrics

Task	Baseline Models	Metrics (Val/Test)
Classification	Logistic Regression; Decision Tree	Accuracy; F1 (macro)
Regression	Linear Regression; Decision Tree Regressor	MAE; RMSE

GitHub Repo Link: https://github.com/MBELSABAH/Cs-4120-Project.git