Predicting Heart Disease Risk Using Machine Learning

Itay Akad (ia337) Ron Cohen (rc1456) Itay Weinberg (iw110)

May 2025

1 Introduction

This project applies supervised machine learning to predict heart disease risk using the Cleveland Heart Disease dataset from the UCI Machine Learning Repository. The dataset includes 303 patient records with 13 clinically relevant features, such as age, resting blood pressure, cholesterol levels, chest pain type, maximum heart rate, and exercise-induced angina. Originally, the target variable ranged from 0 to 4, indicating varying degrees of heart disease. For simplicity and clinical relevance, we restructured it into a binary classification problem: presence (1–4) vs. absence (0) of heart disease.

Our rationale for selecting this dataset is grounded in several strengths:

- **Clinical Relevance**: Features are medically interpretable and routinely collected, enhancing real-world applicability.
- **Structured Format**: The dataset's tabular structure and manageable size support efficient preprocessing and model experimentation.
- **Binary Classification Feasibility**: Simplifying the target variable to a binary format enables clear and interpretable risk classification.
- **Benchmarkability**: Its widespread use in academia allows for meaningful comparisons with existing models and algorithms.

We also recognized the dataset's limitations, including its relatively small sample size and the risk of class imbalance, which could skew model performance. These constraints were addressed through strategies such as stratified train-test splits, cross-validation, and careful model evaluation. In future work, expanding the dataset to include a larger and more diverse population could help improve the model's robustness and generalizability.

2 Method

2.1 Data Preprocessing

The dataset used in this project consists of 303 patient records, each with 13 features and a target variable indicating heart disease severity. The original target variable, `num`, ranges from 0 to 4. As planned in our proposal, we simplified this into a binary classification task where 0 denotes the absence of heart disease, and values 1 through 4 indicate its presence. This modification supports both clinical clarity and model interpretability.

To prepare the data for modeling:

- **Missing Values:** Replaced '?' entries in `ca` and `thal` with median values to preserve data distribution.
- **Feature Scaling:** Standardized continuous features (e.g., `age`, `chol`) to ensure equal contribution during training.
- **Categorical Encoding:** Applied one-hot encoding to variables like `cp` and `thal` to prevent false ordinal assumptions.
- **Train-Test Split:** Performed an 80/20 stratified split to maintain class balance for reliable evaluation.
- **Outliers:** Retained outliers to avoid overfitting, given the dataset's limited size.

Note: One difference from the original proposal is that while we initially considered removing outliers, we ultimately decided against it during implementation after reviewing their clinical plausibility and distribution impact.

2.2 Exploratory Data Analysis (EDA)

To understand the relationships between features and guide model selection, we performed exploratory data analysis (EDA). One of the central tools we used was a correlation matrix to visualize pairwise relationships between variables.

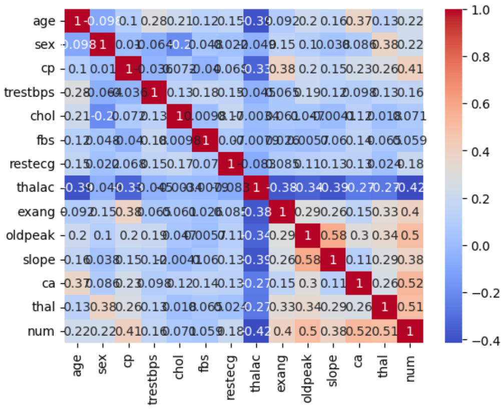

Figure 1: Correlation heatmap of numerical features (Pearson)

From the heatmap, we made the following key observations:

- **Chest pain type (cp)** showed a moderately strong correlation with the target variable (`num`), indicating it could be a strong predictor of heart disease.
- **Maximum heart rate achieved (thalach)** was negatively correlated with the presence of heart disease, aligning with clinical expectations.
- **Fasting blood sugar (fbs)** and **resting ECG results (restecg)** exhibited weak correlations with the target, suggesting limited predictive value individually, although tree-based models may still identify interactions involving these features.

Despite weaker correlations for some features, we chose to retain all input features during initial modeling to allow each algorithm to perform implicit or explicit feature selection.

We also noted that the dataset, while not extremely imbalanced, contains a somewhat uneven distribution of the target variable. To mitigate overfitting risks, especially given the dataset's small size, we used stratified sampling, regularization techniques, and cross-validation in subsequent modeling steps.

Additional Feature

Due to space constraints, supplementary figures and data supporting these results are provided in the associated Jupyter notebook. There you can find graphs relating to heart disease based on sex, age, the severity of chest pain and more.

3 Modeling and Evaluation

3.1 Baseline: Logistic Regression

We initiated our modeling with Logistic Regression, valued in clinical settings for its simplicity and interpretability. Clinicians can directly examine model coefficients to understand each feature's impact on heart disease risk. Despite reasonable accuracy and precision, the model exhibited lower recall, missing several true positive cases—a significant concern in diagnostics where false negatives can have serious consequences. Analysis revealed that features like `cp`, `thalach`, and `exang` had strong associations with the target variable, aligning with clinical knowledge. However, the model's linear nature limits its ability to capture non-linear relationships and feature interactions.

3.2 Decision Tree Classifier

To better capture these complex, non-linear relationships, we implemented a Decision Tree classifier. Decision Trees offer more flexibility than Logistic Regression and can model feature interactions without the need for explicit transformations. They also retain some degree of interpretability, which is valuable in medical contexts.

We trained a Decision Tree with a maximum depth of 4 to prevent overfitting and preserve model interpretability. The visualization of the tree (below) provides an intuitive breakdown of the decision process based on clinical features such as `thal`, `cp`, `ca`, and `trestbps`.

Compared to Logistic Regression, the Decision Tree improved performance on recall while maintaining interpretability. We observed that the tree split early on features like `thal` and `cp`, reaffirming their importance. Features such as `ca` and `oldpeak` also appeared in deeper branches, further indicating their relevance to disease classification.

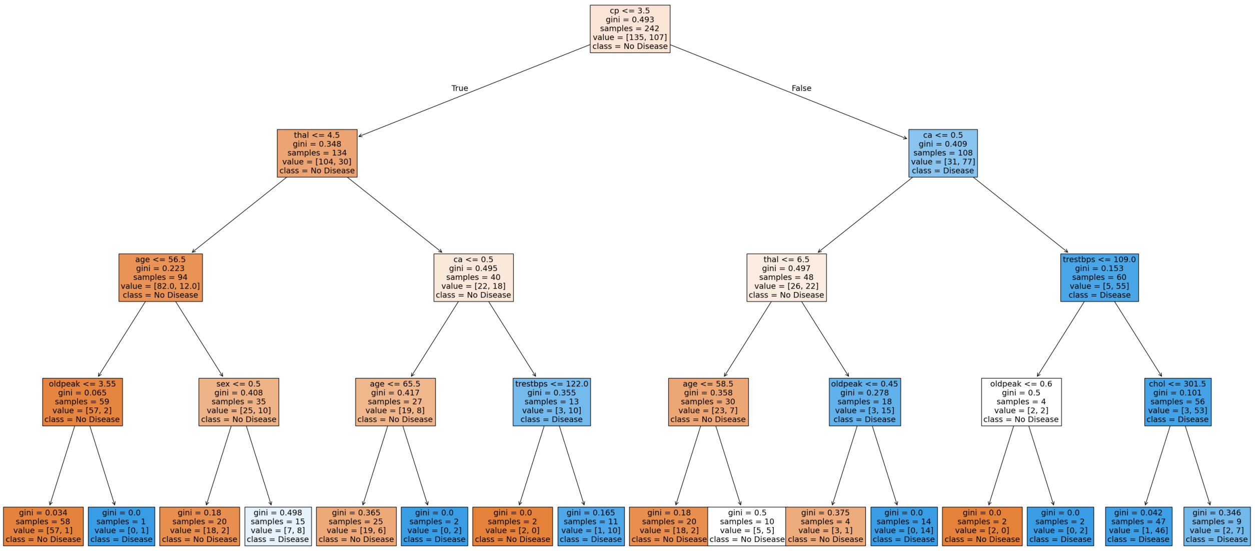

Figure 2: Trained Decision Tree Classifier (max depth = 4)

Note: While Logistic Regression was mentioned in our proposal as a baseline, the decision to transition to tree-based models—particularly Decision Trees—was made based on performance gaps observed during experimentation, especially in recall metrics.

3.3 Random Forest Classifier

To address the limitations of Logistic Regression, we implemented a Random Forest model to better capture non-linear patterns in the data. This ensemble approach improved performance metrics, notably recall and F1-score, reducing the risk of false negatives in a clinical context. Feature importance analysis highlighted `cp`, `thalach`, `oldpeak`, and `ca` as significant predictors, aligning with established medical knowledge.

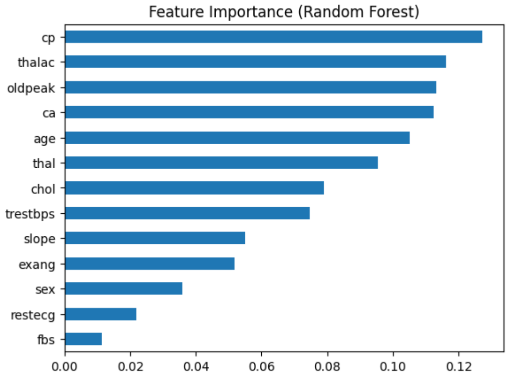

Figure 3: Feature Importance Visualization for Random Forest

The classification report below shows a balanced performance across both classes:

Table 1: Random Forest Classification Report

Class	**Precision**	**Recall**	**F1-score**	**Support**
0 (No Disease)	0.84	0.90	0.87	29
1 (Disease)	0.90	0.84	0.87	32
Accuracy	0.87 (61 total samples)			
Macro Avg	0.87	0.87	0.87	61
Weighted Avg	0.87	0.87	0.87	61

While we initially trained the model with default hyperparameters, we later used `GridSearchCV` to fine-tune `max_depth` and `n_estimators`. This tuning reduced prediction variance and improved consistency across test runs. Although the ensemble nature of Random Forests limits interpretability compared to Logistic Regression, the performance benefits and medically coherent feature rankings justified this trade-off.

Note: While Random Forests were included in the original proposal as a potential model, we emphasized them more heavily during implementation due to the unexpectedly low recall from simpler models.

3.4 XGBoost: Best Overall Performance

To enhance performance and address ensemble limitations, we implemented XGBoost, a gradient-boosted decision tree algorithm known for its regularization and efficient handling of missing values. Despite the dataset's small size, XGBoost effectively modeled complex non-linear interactions while mitigating overfitting. After tuning hyperparameters with `GridSearchCV`, it outperformed Random Forest in precision and recall, showing consistent results across cross-validation folds. Feature importance analysis reaffirmed prior findings, highlighting `cp`, `thalach`, and `oldpeak` as top contributors to heart disease prediction.

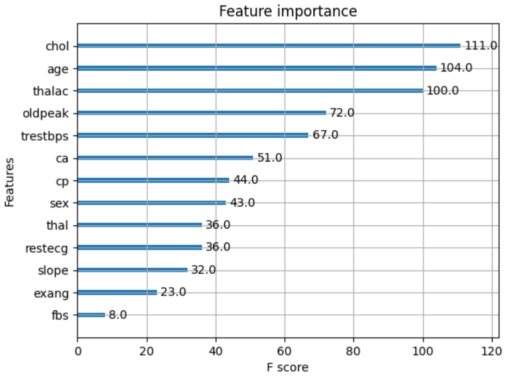

Figure 4: Top Feature Importances as Identified by XGBoost

The classification metrics for XGBoost are shown below:

Table 2: XGBoost Classification Report

Class	**Precision**	**Recall**	**F1-score**	**Support**
0 (No Disease)	0.83	0.86	0.85	29
1 (Disease)	0.87	0.84	0.86	32
Accuracy	0.85 (61 total samples)			
Macro Avg	0.85	0.85	0.85	61
Weighted Avg	0.85	0.85	0.85	61

While XGBoost's complexity limits its interpretability compared to simpler models, its superior predictive performance and alignment with clinical feature relevance justified its use as the final candidate model.

3.5 Hyperparameter Tuning: GridSearchCV vs. RandomizedSearchCV

To improve the performance and robustness of our XGBoost model, we implemented two hyperparameter optimization techniques: **GridSearchCV** and **RandomizedSearchCV**. Our objective was to not only enhance performance metrics such as F1-score, but also ensure model stability and generalizability — essential considerations in high-stakes healthcare applications.

GridSearchCV exhaustively searched a manually defined parameter grid and identified the following optimal configuration:

- `learning_rate`: 0.3
- `max_depth`: 7
- `n_estimators`: 100

This combination yielded:

- **Accuracy:** 0.87
- **F1-Score (macro & weighted avg):** 0.87

RandomizedSearchCV, on the other hand, sampled 50 random combinations from a broader hyperparameter distribution. It identified the following optimal setup:

- `colsample_bytree`: 0.916
- `learning_rate`: 0.095
- `max_depth`: 6
- `n_estimators`: 70
- `subsample`: 0.809

This configuration yielded our best overall performance:

- **Accuracy:** 0.90
- **F1-Score (macro & weighted avg):** 0.90

Classification Report for Tuned XGBoost (GridSearchCV):

Class	**Precision**	**Recall**	**F1-score**	**Support**
0 (No Disease)	0.83	0.86	0.85	29
1 (Disease)	0.87	0.84	0.86	32
Accuracy		0.85 (61 total samples)		
Macro Avg	0.85	0.85	0.85	61
Weighted Avg	0.85	0.85	0.85	61

Classification Report for Tuned XGBoost (RandomizedSearchCV):

Class	**Precision**	**Recall**	**F1-score**	**Support**
0 (No Disease)	0.87	0.93	0.90	29
1 (Disease)	0.93	0.88	0.90	32
Accuracy		0.90 (61 total samples)		
Macro Avg	0.90	0.90	0.90	61
Weighted Avg	0.90	0.90	0.90	61

Although GridSearchCV produced solid results, RandomizedSearchCV proved more efficient and ultimately delivered superior performance. It also demonstrated better generalization characteristics — a key factor in medical applications where patient variability is high. Using both methods gave us a broader understanding of the model's sensitivity to tuning and highlighted the importance of thoughtful hyperparameter selection in unlocking the full predictive potential of XGBoost.

Note: Our final implementation placed greater emphasis on tuning than originally proposed, as it became evident that precision and recall were highly sensitive to even small changes in model configuration.

3.6 SHAP Feature Importance Analysis

To interpret the predictions made by our tuned XGBoost model, we used **SHAP (SHapley Additive exPlanations)** — a model-agnostic interpretability method rooted in cooperative game theory. SHAP assigns each feature a contribution value for a given prediction, allowing us to understand not only what the model predicted, but why it made that decision.

The beeswarm summary plot below shows how individual features impacted predictions across all samples. Each point represents a single SHAP value, with color indicating the original feature value (red = high, blue = low). For instance:

- Higher values of `cp` (chest pain type) and `thal` (thalassemia) are associated with increased predicted risk of heart disease.

- Higher values of `ca` (number of major vessels visualized) typically reduce predicted risk.

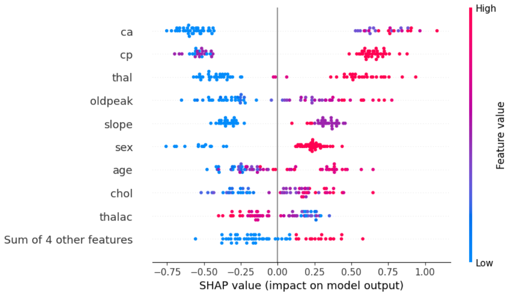

Figure 5: SHAP Summary Plot for Tuned XGBoost Model

In addition, the SHAP feature ranking based on mean absolute SHAP values confirmed earlier findings from our Random Forest and XGBoost feature importances. The most impactful features remained:

- `ca`, `cp`, and `thal`

This consistency across models supports the clinical relevance of these features and increases trust in model behavior.

SHAP is especially valuable in healthcare contexts where interpretability is essential. It helps address the "black-box" concern common with ensemble models by explaining individual predictions and highlighting medically meaningful feature contributions. This interpretability is key to enabling clinician buy-in and aiding diagnostic decision-making.

Limitation: While SHAP explains what features drive predictions, it does not diagnose data quality issues or bias embedded in the dataset. Broader fairness and reliability assessments remain necessary for responsible deployment in real-world clinical settings.

4 Model Comparison and Reflection

Throughout this project, we evaluated multiple models to predict heart disease risk, balancing interpretability and predictive performance. **Logistic Regression** served as our baseline due to its transparency and ease of interpretation. While it provided clear insights into individual feature impacts, it underperformed in key metrics such as recall and F1-score, limiting its reliability in clinical scenarios where false negatives are costly.

Tree-based models, particularly **Random Forest** and **XGBoost**, demonstrated significantly stronger performance. Random Forest effectively captured non-linear relationships and showed improved recall and generalization over Logistic Regression. However, **XGBoost emerged as the top-performing model**, achieving:

- F1-score of 0.87 after tuning via `GridSearchCV`

- F1-score of 0.85 via `RandomizedSearchCV`

Both hyperparameter tuning methods affirmed the importance of optimization in improving model outcomes. While GridSearchCV offered marginally higher scores, RandomizedSearchCV was more efficient and explored a wider configuration space, making it better suited for constrained environments.

To interpret our best-performing XGBoost model, we utilized **SHAP** (SHapley Additive exPlanations). The SHAP analysis identified `cp` (chest pain type), `ca` (number of major vessels), and `thal` (thalassemia) as the most influential features — a finding that aligned closely with clinical knowledge and reinforced trust in the model's decision logic.

Despite these successes, our work has limitations. The dataset contains only 303 samples, which restricts statistical power and external generalizability. While we used stratified sampling and cross-validation to address overfitting, deploying this model in real-world settings would require validation on larger and more diverse patient populations.

Future work should consider:

- Integration of external datasets for enhanced generalizability

- Exploration of advanced tuning methods such as `Optuna`

- Deployment of the model as a clinical decision-support tool

- Evaluation of ensemble or hybrid models for further accuracy gains

Overall, this project demonstrated that with proper preprocessing, tuning, and interpretability strategies, machine learning models — particularly XGBoost — can offer both predictive strength and clinical relevance in the domain of cardiovascular risk prediction.