Team Name:	Heart Protectors
Team Leader:	Van Ky Thien Nguyen
Team Members:	Marcos Villanueva Abreu, Kaitlyn Chan, Austin Choi
Meeting Date/Time:	April 23, 2025
Attendees:	All team members

1 Project Topic / Service Scenario

Abstract: Our project uses machine learning to predict heart failure risk from patient health data. Cardiovascular disease remains one of the leading causes of mortality worldwide, with heart failure affecting millions of patients and placing significant burden on healthcare systems. Traditional diagnostic approaches often fail to identify at-risk individuals until symptoms become severe, resulting in delayed intervention and poorer outcomes.

We apply machine learning algorithms and techniques to predict whether patients are at risk of heart disease based on their health indicators and demographic information. Our models provide simple status predictions that could help doctors identify high-risk patients who might benefit from preventive care. This early detection approach could help reduce heart failure cases, improve patient outcomes, and reduce healthcare costs.

2 Selected Dataset

Dataset Information

For this project, we utilize a single, comprehensive heart disease dataset from Kaggle to build and validate our predictive models:

1. **Heart Disease UCI** (10,000 entries) – `https://www.kaggle.com/datasets/oktayrdeki/heart-disease`

Table 1: Features Overview with Summary Statistics (Numerical Only)

Feature	Type	Mean	Std Dev	Min	Median	Max
Gender	Categorical					
Exercise Habits	Categorical					
Smoking	Categorical					
Family Heart Disease	Categorical					
Diabetes	Categorical					
High Blood Pressure	Categorical					
Low HDL Cholesterol	Categorical					
High LDL Cholesterol	Categorical					
Stress Level	Categorical					
Sugar Consumption	Categorical					
Age	Numerical	49.30	18.19	18.00	49.00	80.00
Blood Pressure	Numerical	149.76	17.57	120.00	150.00	180.00
Cholesterol Level	Numerical	225.43	43.58	150.00	226.00	300.00
BMI	Numerical	29.08	6.31	18.00	29.08	40.00
Sleep Hours	Numerical	6.99	1.75	4.00	7.00	10.00
Triglyceride Level	Numerical	250.73	87.07	100.00	250.00	400.00
Fasting Blood Sugar	Numerical	120.14	23.58	80.00	120.00	160.00
CRP Level	Numerical	7.47	4.34	0.00	7.47	15.00
Homocysteine Level	Numerical	12.46	4.32	5.00	12.41	20.00

3 Problems and Challenges

3.1 Data Quality and Preprocessing Challenges

Data Quality Issues Overview

Our initial analysis of the Heart Disease UCI dataset revealed:

- Missing values in most columns (19-30 entries per column)
- Significant missing data in 'Alcohol Consumption' (2,586 of 10,000 entries missing)
- Mix of numerical features (9) and categorical features (12)
- Need for proper data type conversion and standardization

Missing Values by Column:

Age	29	High Blood Pressure	26
Gender	19	Low HDL Cholesterol	25
Blood Pressure	19	High LDL Cholesterol	26
Cholesterol Level	30	Alcohol Consumption	2,586
Exercise Habits	25	Stress Level	22
Smoking	25	Sleep Hours	25
Family Heart Disease	21	Sugar Consumption	30
Diabetes	30	Triglyceride Level	26
BMI	22	Fasting Blood Sugar	22
		CRP Level	26
		Homocysteine Level	20

3.1.1 Missing Values

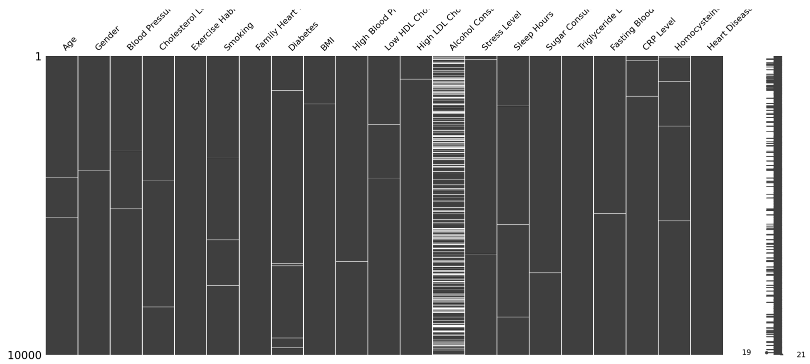

Figure 1: Missing Values Matrix Visualization

We able to identify missing data patterns through the matrix above.

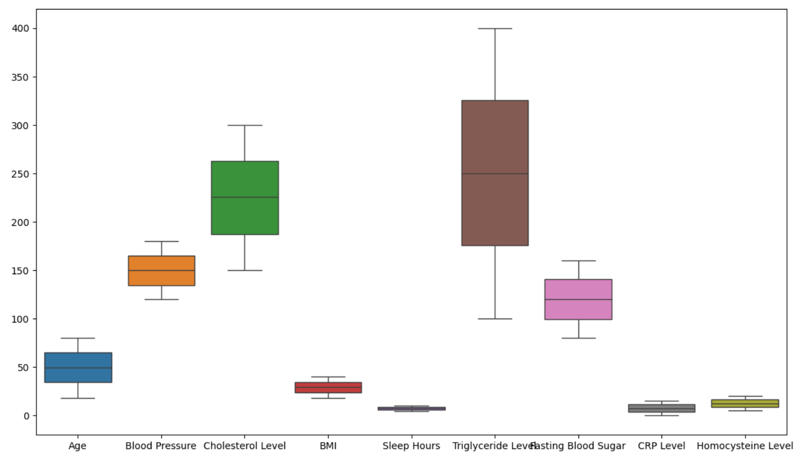

Figure 2: Boxplot Analysis for Outlier Detection

Next, we performed outlier analysis using boxplots. The analysis showed no significant outliers in the numerical features that would require special treatment. This allowed us to proceed with the preprocessing without additional outlier handling steps.

To summarize, our preprocessing approach for missing values involved these steps:

- **Feature removal**: We completely removed the 'Alcohol Consumption' feature due to excessive missing data (25.9%).
- **Numerical imputation**: For numerical features, we replaced missing values with column means (affecting 19-30 entries per column).
- **Categorical imputation**: For categorical features, we replaced missing values with the most frequent value (mode) in each column.

Results: After preprocessing, all missing values were successfully handled, creating a complete dataset with 10,000 entries and 20 features (the original 21 minus Alcohol Consumption).

3.1.2 Feature Encoding and Transformation

For effective model training, we applied these transformations:

- **Categorical encoding**: Converted categorical variables to numerical form using one-hot encoding
- **Feature scaling**: Standardized numerical features to have zero mean and unit variance
- **Data type conversion**: Optimized data types (e.g., converted Gender to category type)

Results: These transformations produced a machine learning-ready dataset with properly scaled numerical features and encoded categorical variables, enabling efficient model training.

3.2 Methodological Challenges

3.2.1 Feature Selection

Determining the most predictive features for heart failure risk presented a significant challenge. Through visualizations and statistical analysis, we identified several key predictors:

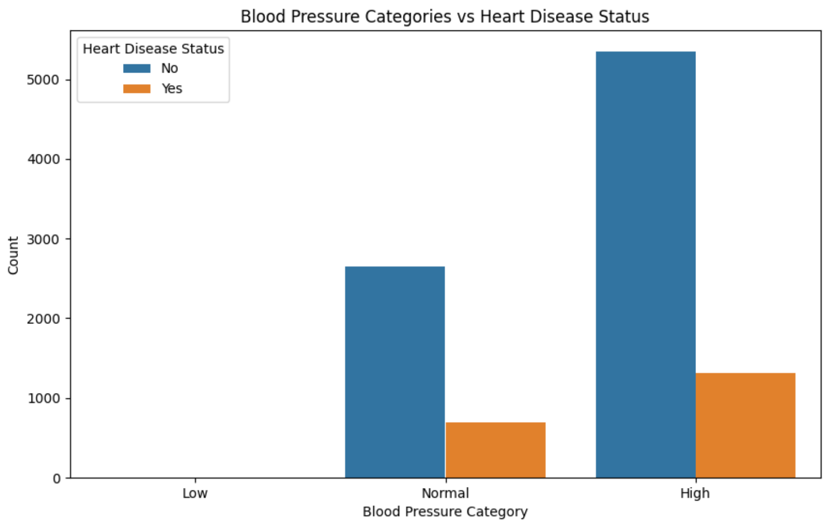

Figure 3: Blood Pressure Categories vs Heart Disease Status

Blood Pressure: Analysis shows that individuals with higher blood pressure tend to have a higher incidence of heart disease.

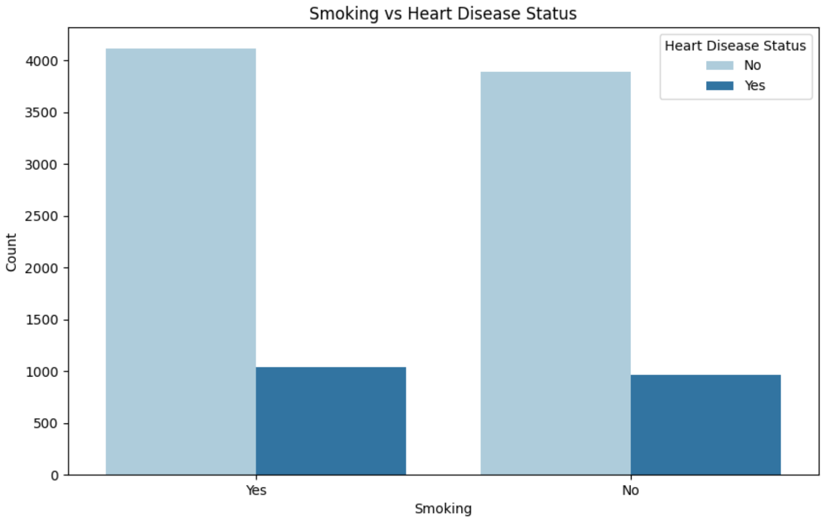

Figure 4: Smoking vs Heart Disease Status

Smoking: The visualization above reveals a clear relationship between smoking habits and heart disease status.

Our initial approach included using ANOVA F-tests to identify statistically significant features, but the effectiveness of this method varied across different subsets of our data. The interrelationships between various health indicators complicated the feature selection process, as some features that appeared individually insignificant could become important when considered in combination with others.

3.2.2 Model Selection and Optimization

Selecting the most appropriate machine learning models for heart failure prediction required balancing accuracy, interpretability, and computational efficiency. Our project implements Neural Network and Support Vector Machine models, each presenting unique challenges:

- ✔ **Neural Network (NN)**: Our deep learning approach required careful architecture design, including determining the optimal number of hidden layers, neurons per layer, and activation functions. Additionally, preventing overfitting through techniques like dropout and early stopping required extensive experimentation.
- ✔ **Decision Tree Classifier**: While effective for high-dimensional classification, optimizing SVM required extensive hyperparameter tuning, particularly for the regularization parameter (C) and kernel selection (linear, polynomial, RBF). Finding the right balance between model complexity and generalization capability was challenging.

3.3 Implementation Challenges

- **Neural Network (NN)** - During development, our neural network model encountered a very common issue, overfitting. The initial implementation would have 94% accuracy on average with the training data, and 70% accuracy on average with the testing data. To battle this overfitting issue, a dropout layer and regularization was implemented.

3.3.1 Cross-Validation Strategy

To ensure robust model evaluation and prevent overfitting, we implemented **k-fold cross-validation** + **bootstraping** with **k** values ranging from 5 to 20. This standardized approach was consistently applied across all models to facilitate fair comparison of performance metrics.

4 Models Implemented

The project implements the following machine learning models for heart failure risk prediction:

- **Neural Network (NN)**: A feed-forward multilayer perceptron trained on the UCI dataset with early stopping and dropout regularization.
- **Decision Tree Classifier**: A decision tree classifier wrapped in an `imblearn` pipeline

5 Each Member's Implementation and Evaluation

5.1 Member 1 (Marcos Villanueva Abreu, undergraduate)

Neural Network Implementation

Model and Implementation details:

The Neural Network model was implemented with the following key components:

- Two hidden layers, and one dropout layer that sets 5% of the inputs to 0.
- Each hidden layer has 512 nodes and used ReLU as their activation function.
- Output layer has 2 nodes (binary classification).
- Hidden layers use L1 and L2 regularization to reduce overfitting.
- Data is standardized, nominal values are one hot encoded and ordinal values are encoded with an ordinal encoder.
- Input layer has 26 nodes (7 additional nodes from the 19 features due to the encoding).

Evaluation Results:

- Accuracy: 80.43%
- Precision: 80.43%
- Recall: 80.43%
- F1 Score: 89.15%
- AUC-ROC: 80.43%

5.2 Member 2 (Van Ky Thien Nguyen, graduate)

Decision Tree Classifier Implementation

Model and implementation details:

- Decision-tree classifier wrapped in an `imblearn` pipeline
- Grid-search hyper-tuning for `max_depth` $\in \{3, 5, 7, \text{None}\}$ and `min_samples_leaf` $\in \{1, 5, 10\}$
- SMOTE oversampling to combat class imbalance
- One-hot encoding for any nominal categorical features (none in this run)
- Stratified k-fold cross-validation with $k \in \{5, \ldots, 20\}$, executed sequentially (`n_jobs=1`)

Evaluation results (k-fold, $k = 5$–20):

- Accuracy: **65.9 % – 73.3 %**
- Precision: **6.9 % – 15.1 %**
- Recall: **11.8 % – 25.8 %**
- F_1 score: **8.7 % – 19.0 %**
- AUC-ROC: **50.2 % – 51.9 %**

Train-on-full dataset snapshot:

- Accuracy: 60.5 %
- Precision: 21.4 %
- Recall: 36.5 %
- F_1 score: 27.0 %
- AUC-ROC: 52.2 %

5.3 Member 3 (Name, graduate/undergraduate)

Data Preprocessing and Feature Engineering

Implementation details:
This work focused on:

- a
- a
- a
- a
- a

Evaluation Results:

- Accuracy:
- Precision:
- Recall:
- F1 Score:
- AUC-ROC:

5.4 Member 4 (Name, graduate/undergraduate)

Visualization and Model Evaluation

Implementation details:

This work focused on:

- a
- a
- a
- a
- a

Evaluation Results:

- Accuracy:
- Precision:
- Recall:
- F1 Score:
- AUC-ROC:

6 Discussion

- **Neural Network (NN)** - As we can observe above from the evaluation results of the model, the accuracy, recall, and precision are the same values while the AUC is very similar (the last couple of decimal points are different) to the values of the aforementioned metrics. This could be that the model is prediction the samples correctly meaning there are no False Positives or False Negatives, this is really rare so it would be something our team would further need to investigate in order to improve this models performance. Another thing that caught our attention is that it was very common for the neural network to have the same training and testing accuracy average while validating the model using K-fold. The actual values for training and testing data accuracy were different in each fold but the overall accuracy tended to be the same for both.

- **Decision Tree Classifier** – While cross-validated accuracy is respectable $\approx$ **66–73%**, the model's precision and recall remain low ($\approx$ **7–26%**), and its AUC stays just above random chance ($\approx$ **50–52%**). These figures indicate the tree still struggles to isolate positive cases despite oversampling and tuning. Future work should investigate richer feature engineering, ensemble variants (e.g., balanced random forests, gradient boosting), or cost-sensitive learning to lift minority-class detection without sacrificing overall stability.