Project Proposal: Predicting Heart Failure

Dhiren Patel

February 24, 2025

Abstract

This is a supervised learning project that intends to train a model predicting the presence of heart failure using a neural network. In this proposal, we discuss the problem, dataset, and methodology, including preprocessing of data, model architecture, training, and model evaluation.

1 Introduction

Cardiovascular (heart) diseases are among the most prevalent causes of death in the world. The ability to detect and predict heart failure early can help provide patients an increased chance of survial and even an extended lifespan.

Machine Learning techniques like neural networks can help identify the complex relationships between health factors that lead to heart failure. In this project, we will train a predictive model that can understand relationships between these variables at a more complex degree than can be understood by humans.

2 Methodology

2.1 Data Collection and Description

For this project, we are utilizing the publicly available UCI Heart Failure Prediction Dataset. The dataset is a compilation of 5 independent Heart Failure datasts from Cleveland, Hungary, Switzerland, Long Beach (VA), and Stalog. There are a total of 918 unique observations with the following features:

- **Age:** in years

- **Sex:**
  - **M**: Male
  - **F**: Female
- **ChestPainType:**
  - **TA**: Typical Angina
  - **ATA**: Atypical Angina
  - **NAP**: Non-Anginal Pain
  - **ASY**: Asymptomatic
- **RestingBP:** in millimeters of mercury (mm Hg)
- **Cholesterol:** in milligrams per deciliter (mg/dl)
- **FastingBS:**
  - **1**: $>$ 120 mg/dl
  - **0**: $<$ 120 mg/dl
- **RestingECG:**
  - **Normal**
  - **ST**: ST-T wave abnormalities
  - **LVH**: Left Ventricular Hypertrophy
- **MaxHR:** in bpm
- **ExerciseAngina**
  - **Y**: chest pain during exercise
  - **N**: no chest pain during exercise
- **Oldpeak:** numerical, representing ST depression of exercise relative to rest
- **ST_Slope:** the slope of the peak exercise ST segment
  - **Up**: (normal)
  - **Flat**

  - **Down**

- **Heart Disease**, in this case, the target variable. HeartDisease is binary with 0 representing the lack of heart failure and 1 representing the presence of heart failure in a patient.

2.2 Preprocessing

Prior to training our model, it will be necessary for us to preprocess the data for ease of interpretability by our model, leading to greater efficiency and lower loss.

1. **Data Cleaning**: Impute or remove missing / incomplete data
2. **Encoding Categorical Features**: Use one-hot encoding to convert categorial data to numerical (e.g., ChestPainType, RestingECG).
3. **Scaling**: Standardize numeric features (Age, Cholesterol, Oldpeak) for improved training stability.
4. **Feature Engineering**: Apply formulas to engineer further features from the data to optimize training.
5. **Train-Test Split**: Split data into training / testing / validation. We will use the training data to train our model on, validation data to test our training data on during training, and testing to test our model on after training.

2.3 Neural Network Architecture

We will start with a standard neural network architecture as described below and fine-tune our model emperically, adding, subtracting, and modifying layers, activation functions, optimizers, and regularization functions.

- **Input Layer**: Send input features through a first layer with number of nodes equal to features after preprocessing.
- **Hidden Layers**: Send the output of the input layer into multiple fully connected layers with an activation function (possibly ReLU). We will emperically determine the specifics of these layers.
- **Output Layer**: The output layer will consist of one node with a sigmoid activation function that returns a probability from 0 to 1.

- **Loss Function**: We will use a binary cross-entropy loss function to inform our model training due to the existance of only 2 classes.

$$L(y, \hat{y}) = -\frac{1}{N}\sum_{i=1}^{N} \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right]$$

  where:

  - y_i is the true class (0 or 1)
  - $\hat{y}_i$ is the predicted probability for the value of 1

- **Optimizer**: We anticipate using the Adam optimizer for its dynamic step size handling.

2.4 Model Training and Evaluation

- **Training Procedure**:

  1. Initialize weights randomly.
  2. Train the network over a series of epochs.
  3. Use validation loss as a signal for stopping training.

- **Evaluation Metrics**: We anticipate using the following evaluation metrics to interpret the effectiveness of our model:

  - Accuracy
  - Precision
  - Recall
  - F1-score
  - Confusion Matrix (for visual / interpretive purposes)

3 Conclusion

We hope that this projct will lead to an accurate neural network that can predict heart failure in patients prior to their deaths based on measurable health factors. We also hope that this exercise can reveal which features are influential in contributing to heart failure. The primary purpose of this project is to become comfortable working with large datasets and using ML tools to help optimize the task of learning through data.

4 References

1. fedesoriano. (September 2021). Heart Failure Prediction Dataset. Retrieved [February 24, 2025] from

   `https://www.kaggle.com/fedesoriano/heart-failure-prediction`.