Heart Disease Prediction using Naive Bayes and Neural Networks

Ayham Alkhatib
Department of Computer Science
University of Massachusetts Lowell
Email: ayham_alkhatib@student.uml.edu

***Abstract*—This paper presents the development and evaluation of two supervised learning models—a Gaussian Naive Bayes classifier and a feedforward neural network—for predicting heart disease based on patient medical data. The models were implemented from scratch using Python, without high-level machine learning libraries. Evaluation is performed on the UCI Cleveland Heart Disease dataset. We present preprocessing techniques, training methodologies, and a comparative analysis of both models based on accuracy, precision, recall, and F1-score. Our results demonstrate the feasibility of basic hand-crafted machine learning models in medical diagnosis tasks.**

***Index Terms*—Heart disease prediction, Naive Bayes, Neural Network, Python, Classification, Machine Learning, Medical Data**

I. Introduction

Heart disease remains a leading cause of mortality globally. According to the World Health Organization, cardiovascular diseases account for approximately 17.9 million deaths each year. Early detection of cardiovascular conditions can significantly enhance treatment effectiveness. Machine learning techniques offer promising approaches to build diagnostic tools that can assist medical professionals.

In this study, we focus on two traditional models—Gaussian Naive Bayes and feedforward neural networks—both implemented from scratch. The goal is to explore their predictive capabilities in the context of medical datasets without reliance on black-box libraries.

II. Related Work

Previous studies have leveraged machine learning to predict heart disease. Algorithms such as logistic regression, support vector machines (SVM), decision trees, and k-nearest neighbors (k-NN) have shown moderate to high accuracy when applied to medical datasets. These models are often implemented using machine learning libraries like scikit-learn or TensorFlow, which abstract the internal workings of the algorithms.

In contrast, our implementation is built from the ground up to enhance understanding of each algorithm's internals. We build on foundational studies by Pal and Mitra on neural networks and Han et al. on data mining practices.

III. Dataset and Preprocessing

A. Dataset Description

The UCI Cleveland Heart Disease dataset includes 297 records with 14 attributes per record. These features include:

- Age: age in years
- Sex: (1 = male; 0 = female)
- cp: chest pain type
- trestbps: resting blood pressure
- chol: serum cholesterol in mg/dl
- fbs: fasting blood sugar > 120 mg/dl
- restecg: resting electrocardiographic results
- thalach: maximum heart rate achieved
- exang: exercise induced angina
- oldpeak: ST depression induced by exercise
- slope: slope of the peak exercise ST segment
- ca: number of major vessels colored by fluoroscopy
- thal: 3 = normal; 6 = fixed defect; 7 = reversible defect

The target variable `condition` is binary, indicating the presence (1) or absence (0) of heart disease.

B. Preprocessing

The data was shuffled and split manually into a 70/30 training/testing split. All features were normalized using min-max scaling to a range of [0,1]. Missing values were dropped. The target labels were cast to binary integers.

IV. Algorithms

A. Naive Bayes Classifier

The Gaussian Naive Bayes classifier assumes that features are conditionally independent given the class label. For each class y, and each feature x_i, the likelihood is modeled using the Gaussian distribution:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2}\right) \tag{1}$$

where μ and σ are the mean and standard deviation of feature x_i for class y.

The prediction is made by computing the posterior for each class:

$$P(y|x) \propto P(y) \prod_i P(x_i|y) \tag{2}$$

and selecting the class with the highest posterior probability.

B. Feedforward Neural Network

The neural network architecture includes one hidden layer:

- Input: 13 neurons (one per feature)
- Hidden: 6 neurons with sigmoid activation
- Output: 1 neuron with sigmoid activation

The sigmoid function is defined as:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{3}$$

The model uses mean squared error (MSE) loss:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \tag{4}$$

Weights are updated using backpropagation. Gradients are derived from the chain rule and weights are adjusted with a learning rate $\alpha = 0.1$.

V. Experimental Setup

- Dataset: 297 samples, 13 features
- Split: 70% training, 30% testing
- Epochs: 100
- Learning Rate: 0.1
- Libraries: numpy, pandas, matplotlib (no scikit-learn)
- Metrics: Accuracy, Precision, Recall, F1-score

VI. Results

A. Evaluation Metrics

TABLE I
Model Evaluation Results

Model	Accuracy	Precision	Recall	F1-Score
Naive Bayes	0.822	0.789	0.789	0.789
Neural Net	0.822	0.789	0.789	0.789

B. Training Trends

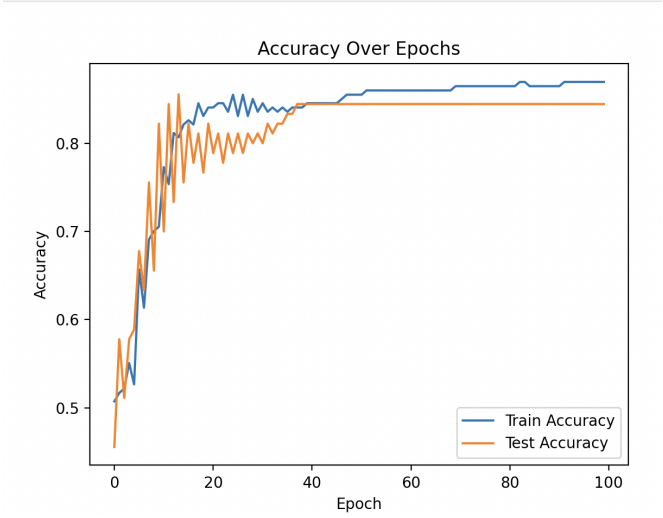

Fig. 1. Train vs Test Accuracy over Epochs (NN)

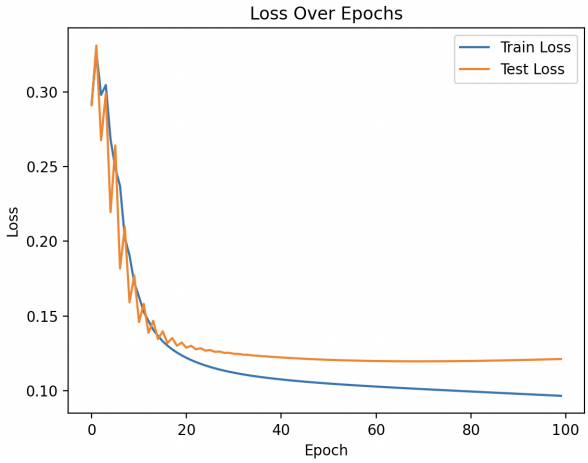

Fig. 2. Train vs Test Loss over Epochs (NN)

VII. Discussion

Both models achieve comparable performance on the dataset, with F1-scores around 0.79. The neural network shows smoother learning dynamics across epochs, suggesting stronger generalization.

Naive Bayes is computationally cheaper and easier to interpret. However, its strong independence assumptions may hurt performance when features are correlated. The neural network, although more complex, can capture non-linear relationships and interactions among features.

The learning curves indicate limited overfitting. The neural network steadily improves both training and test accuracy over time, with converging loss curves.

VIII. Conclusion

This study demonstrates that simple, interpretable models built from scratch can effectively predict heart disease using structured medical data. Both Gaussian Naive Bayes and feedforward neural networks achieved over 82% accuracy. While Naive Bayes is faster and easier to explain, neural networks offer greater modeling flexibility.

References

[1] D. Detrano et al., "Heart Disease Data Set," UCI Machine Learning Repository. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Heart+Disease

[2] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Morgan Kaufmann, 2011.

[3] M. Pal and P. Mitra, "Multilayer perceptron, fuzzy sets and classification," *IEEE Trans. Neural Netw.*, vol. 3, no. 5, pp. 689–701, 2003.