Group 3 Final Report:
Heart Disease Prediction using Machine Learning

ZhiChong Lin, ZiDi Yao, Ke Ma

`yaoz25@mcmaster.ca`, `mak11@mcmaster.ca`, `lin281@mcmaster.ca`

1 Introduction

This project aims to use ML method to predict the likelihood of heart disease based on eleven clinical and demographic features. Heart disease continues to be a major global health concern, and early detection is crucial for reducing severe outcomes. As outlined in our project proposal, the goal of this work is to develop a reliable classification model that can identify high-risk patients using routinely collected medical measurements.

Building on prior studies, our project uses a supervised learning pipeline based on Principal Component Analysis (PCA) and an RBF-kernel SVM classifier. We preprocess the data with a column transformer that standardizes numerical features and applies one-hot encoding to categorical features, then apply PCA and keep the number of components according to the Kaiser criterion [3]. The resulting low-dimensional representations are fed into an RBF-SVM, and the full pipeline is evaluated on the Kaggle Heart Failure Prediction dataset [1] using stratified 5-fold cross validation with standard metrics (accuracy, precision, recall, F1-score, and ROC–AUC).

2 Dataset and Preprocessing

This section will introduce the raw dataset we used, and how we clean and process it.

2.1 Dataset Description

The dataset we used in this project is the *Heart Failure Prediction Dataset* published by Fedesoriano on Kaggle [1]. It contains **918 patient observations** and **12 attributes**, including 11 clinical predictor variables and one binary target label indicating the presence of heart disease. This dataset was designed to support research on early detection of cardiovascular risks, particularly heart failure, which remains one of the leading causes of global mortality.

The dataset includes a mixture of demographic features (Age, Sex), physiological measurements (like RestingBP, Cholesterol, MaxHR), and exercise-induced ECG-related metrics (ExerciseAngina, Oldpeak, ST_Slope). Those attributes show common risk factors used in medical diagnostics for cardiovascular disease and have been widely adopted in machine learning models for clinical prediction tasks.

A complete list of raw features and their corresponding descriptions is provided in Table 2.

Originally, the dataset contains:

$$\textbf{918 samples} \quad \times \quad \textbf{11 features}.$$

Feature	Description
Age	Age of patient (years)
Sex	Biological sex (M/F)
ChestPainType	Chest pain type (ATA, NAP, ASY, TA)
RestingBP	Resting blood pressure (mm Hg)
Cholesterol	Serum cholesterol (mg/dL)
FastingBS	Fasting blood sugar (0/1)
RestingECG	Resting ECG results (Normal, ST, LVH)
MaxHR	Maximum heart rate achieved
ExerciseAngina	Exercise-induced angina (Y/N)
Oldpeak	ST depression value induced by exercise
ST_Slope	Slope of ST segment (Up, Flat, Down)
HeartDisease	Target label (1 = disease, 0 = healthy)

Table 1: Raw dataset features.

2.2 Feature Preprocessing

The feature matrix X was constructed by removing the target column and retaining the remaining 18 input attributes. Since the dataset includes both numerical and categorical variables, several preprocessing steps were required to convert all values into a machine-learning- ready numeric form.

Binary Encoding   Three features, namely Sex, ExerciseAngina, and FastingBS contain only two possible values and were mapped directly to 0/1 following our preprocessing script:

- **Sex:** $M \to 1$, $F \to 0$

- **ExerciseAngina:** $Y \to 1$, $N \to 0$

- **FastingBS:** preserved as integer $0/1$

Ordinal Mapping of Multi-Class Features   Three categorical features contain more than two categories. In the stored processed dataset (X_encoded.csv), they were converted to integer codes according to predefined mappings:

$$\text{ChestPainType: } \{\text{ATA}, \text{NAP}, \text{ASY}, \text{TA}\} \to \{0, 1, 2, 3\},$$

$$\text{RestingECG: } \{\text{Normal}, \text{ST}, \text{LVH}\} \to \{0, 1, 2\},$$

$$\text{ST\_Slope: } \{\text{Up}, \text{Flat}, \text{Down}\} \to \{0, 1, 2\}.$$

These mappings avoid string-based ambiguity and ensure that all feature columns are numeric at the preprocessing stage.

In contrast from progress report, we adopted feedback from TA to change our way of presenting One-Hot Encoding, instead of present them in one single column to split them across the columns for every single category.

Therefore, after splitting them into one-hot encoding, the final dataset shape changes to 918 x 18.

Feature	Description
Age	Age of patient (years)
Sex	Biological sex (M/F)
ChestPainType_0	ATA
ChestPainType_1	NAP
ChestPainType_2	ASY
ChestPainType_3	TA
RestingBP	Resting blood pressure (mm Hg)
Cholesterol	Serum cholesterol (mg/dL)
FastingBS	Fasting blood sugar (0/1)
RestingECG_0	Normal
RestingECG_1	ST
RestingECG_2	LVH
MaxHR	Maximum heart rate achieved
ExerciseAngina	Exercise-induced angina (Y/N)
Oldpeak	ST depression value induced by exercise
ST_Slope_0	Up
ST_Slope_1	Flat
ST_Slope_2	Down
HeartDisease	Target label (1 = disease, 0 = healthy)

Table 2: One-Hot Encoding Table.

2.3 Target Label Extraction

The target label `HeartDisease` is a binary indicator representing whether a patient shows signs of heart disease. We extract this column and converted it to integer form using:

$$y = \mathrm{df['HeartDisease'].astype(int)}.$$

The resulting is a one-dimensional vector , which was saved as `processed/y.csv` for all downstream training and evaluation.

The cleaned dataset was saved to `processed/X_encoded.csv`, and the corresponding feature names were exported to `processed/feature_names.txt` for reproducibility.

3 Model Inputs (Features)

During our data preprocessing phase, we use `scikit-learn Pipeline` and `ColumnTransformer` to transform the raw attributes before training. Numerical features (Age, RestingBP, Cholesterol, MaxHR, Oldpeak, FastingBS) are imputed with the median and standardized, while categorical features (Sex, Chest-PainType, RestingECG, ExerciseAngina, ST_Slope) are converted into one-hot vectors. After this step each patient is represented by an 18-dimensional feature vector, matching the one-hot layout described in Table 2.

Motivated by our progress report and TA feedback, in the final system we use PCA as our only feature-engineering method, instead of the Chi-square + PCA combination proposed in [2].

- **PCA.** We apply Principal Component Analysis (PCA) to the standardized and one-hot encoded features to reduce dimensionality. PCA finds orthogonal directions (principal components) that capture the

largest variance in the data and projects each sample onto these directions. To choose the number of components we adopt the Kaiser criterion [3], keeping only components with eigenvalues greater than 1. In our experiments the first five components satisfy this rule, so we retain five principal components for the final representation.

Hence, each patient sample is represented as a compact 5-dimensional feature vector summarizing the most informative physiological and categorical characteristics, which is then used as input to our machine learning models.

4 Model Implementation

We have implemented a supervised learning pipeline for binary classification of heart disease presence. Our main model is a **Support Vector Machine (SVM)** with a **Radial Basis Function (RBF)** kernel, implemented using the `scikit-learn` library. This kernel choice allows the decision boundary to be nonlinear, which is important given the heterogeneous mixture of categorical and numerical medical features in the dataset.

Loss Function:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{i=1}^{n} \max(0, 1 - y_i(\mathbf{w}^\top \phi(\mathbf{x}_i) + b))$$

where C is the penalty parameter controlling the trade-off between the margin size and misclassification tolerance, and $\phi(\cdot)$ denotes the nonlinear mapping induced by the **RBF Kernel:**

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|^2)$$

The model was optimized using the `libsvm` implementation, which employs a coordinate descent solver with kernel caching for efficiency.

With SVM-RBF, we evaluate it's accuracy by using cross-validation, and we was only able to achieve 87% of accuracy.

5 Evaluation

In turns of method evalution, we first split the dataset into training and validaion. Training dataset with 80% of the data, while validation dataset with 20% of the data.

5.1 Evaluation Method

In our evaluation stage, we use stratified K-fold cross-validation. The reason for this choice is that our dataset contains only 918 patient records, and using a fixed train/validation/test split may lead to overfitting due to the small number of data points.

5.2 Metrics

Same as progress report, we have in total six metrics to evaluate our model performance. The metrics used in this project include **accuracy, recall, precision, F1-score, and AUC–ROC and Confusion Matrix**. Our rationale for choosing these metrics is as follows. First, in disease prediction, precision (closely related to Type II error) is especially important. Hospitals cannot afford to misclassify a patient with the disease as healthy. Therefore, minimizing false negatives is crucial. In addition, we use the AUC–ROC curve because it is an effective way to evaluate the overall performance of a binary classifier.

We decided to keep using the metrics reported in our progress report, as these six metrics provide a sufficient and well-rounded understanding of our model's performance for this task.

6 Process

In our project proposal we planned to follow the Chi-square + PCA + RBF–SVM framework from Gárate-Escamila et al. [2]. In the early stage of the project (before the progress report), we implemented a first version of this pipeline: nominal features such as `ChestPainType`, `RestingECG`, and `ST_Slope` were encoded as integers (e.g., ATA $\rightarrow$ 0, NAP $\rightarrow$ 1, ASY $\rightarrow$ 2), and we evaluated an RBF–SVM using a simple 80%/20% train–test split. This gave us an initial accuracy of about 85%, which confirmed that the approach was reasonable but left room for improvement.

In the progress report, the TA gave us two concrete suggestions: (1) replace label encoding of multi-class categorical variables with one-hot encoding, and (2) experiment with a different models like a neural network. After that feedback we revised our preprocessing: we switched to one-hot encoding (either via `pandas.get_dummies` or `OneHotEncoder` in a `ColumnTransformer`), which expanded the feature space to 18 columns and removed the artificial ordering implied by the integer codes.

Our original plan to strictly reproduce a full Chi-square + PCA pipeline changed slightly as we test more with the Kaggle dataset. Because this dataset has only 11 original features and relatively clean structure, we found that PCA alone already captures most of the variance, and chi-square filtering did not noticeably increase performance. So, we simplified the feature engineering stage to focus on PCA and foces our comparison on three models: Neural Network (as a simple baseline), RBF–SVM (our main model), and LightGBM + SHAP (The method found online). Overall, we followed the spirit of our original plan (PCA + SVM with additional baselines), but refined the details of preprocessing and evaluation in response to TA feedback and empirical results.

7 Results (Error Anaysis) and Baseline

7.1 Results

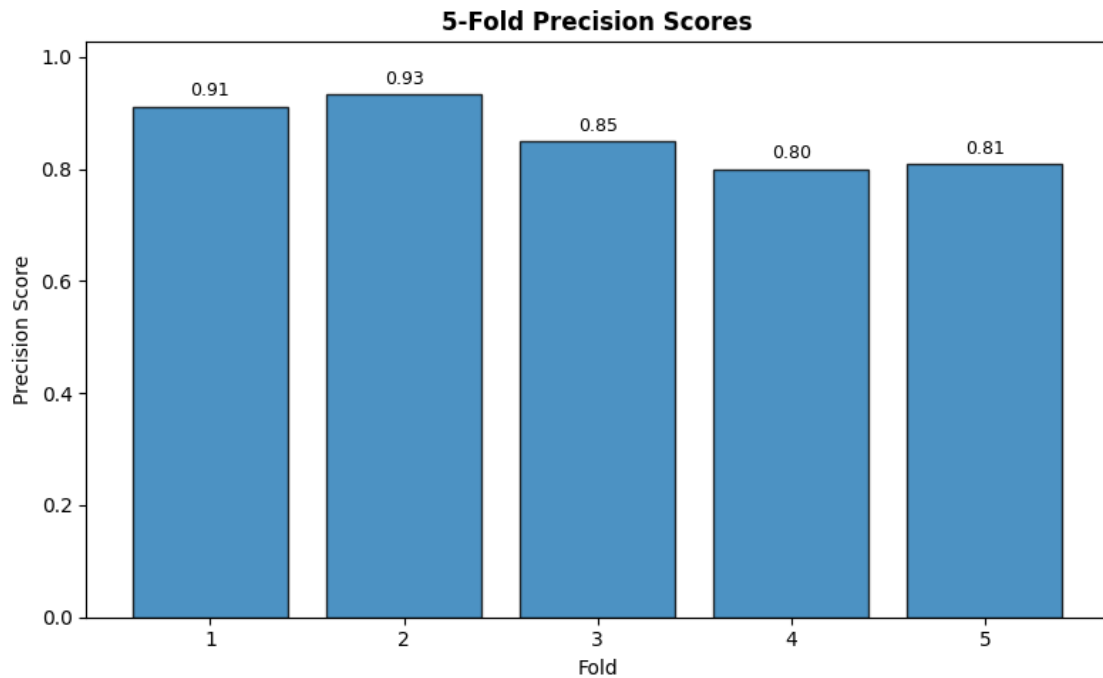The figures are listed in the last page, please check them out

Figure 1: 5-fold precision scores.

```
===== K-Fold Cross Validation Results =====
Accuracy : 0.8627
Recall   : 0.9001
Precision: 0.8610
F1-score : 0.8785

============================================
```
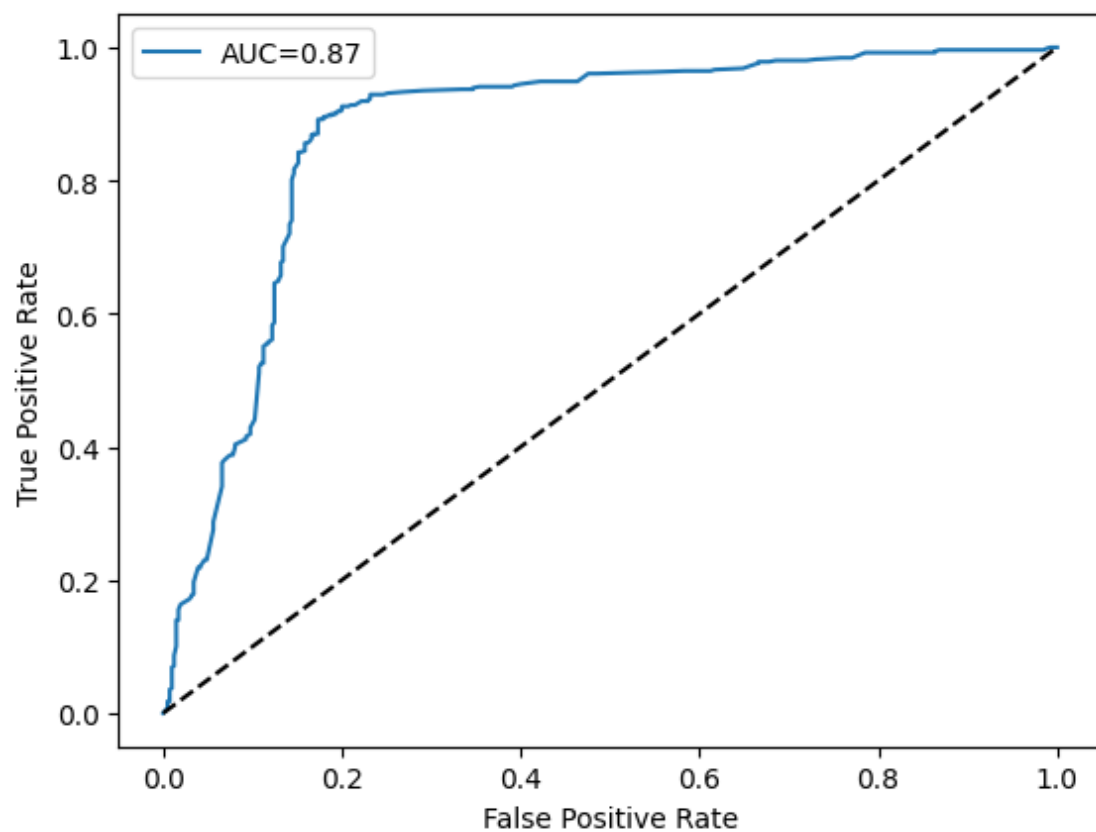
Figure 2: Summary metrics across folds.
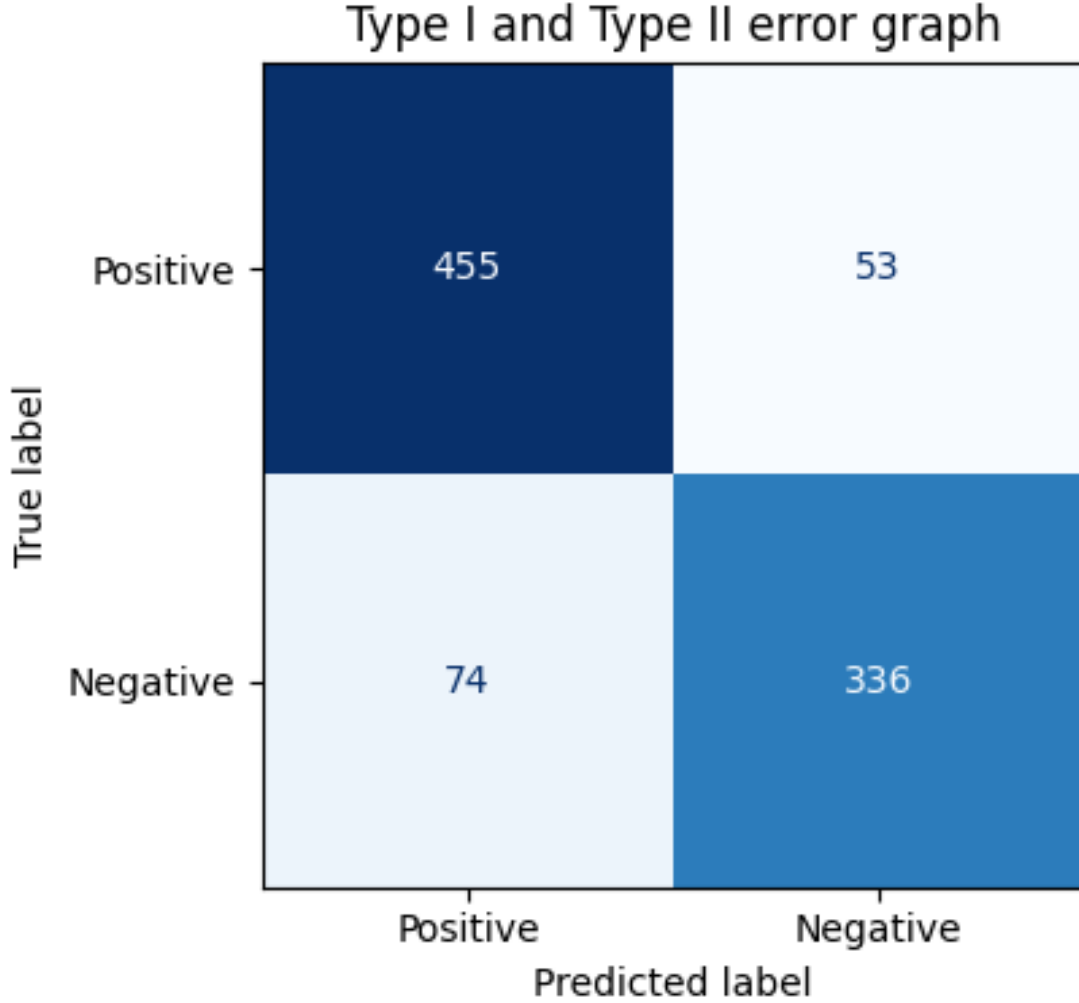
Figure 3: AUC–ROC curves.

Figure 4: Confusion matrices.

7.2 Comparison with the Logistic Regression Baseline

To put the performance of our proposed RBF–kernel SVM pipeline into context, we trained a simple Logistic Regression model as a baseline on the same dataset of 918 patients. For the baseline we used the integer–encoded features described in Section **??**: sex, chest pain type, resting ECG, exercise–induced angina, and ST slope were mapped to small integer codes, and no additional feature selection or dimensionality reduction was applied. The model was trained with an 80/20 train–test split and evaluated using the same 5-fold stratified cross-validation protocol as our main model.

The Logistic Regression baseline achieved a hold-out test accuracy of approximately **0.87** and a mean 5-fold cross-validation accuracy of **0.85**. Our full RBF–SVM pipeline, which combines standardisation, PCA (5 components) and an RBF kernel, achieved a higher mean cross-validation accuracy of **0.86**, with a mean recall of **0.90**, precision of **0.86**, and F1-score of **0.88** across the five folds. The corresponding ROC–AUC of the SVM model is around **0.87**, as shown in Figure **??**.

Table 3 summarises the key metrics. Although the absolute gain in accuracy over Logistic Regression is modest (about 1–2 percentage points), the increase in recall and F1-score for the positive (heart disease) class is clinically important, because missing a patient with heart disease is more costly than producing an

additional false alarm. This justifies the additional complexity of the SVM-based pipeline over the linear baseline.

Model	CV Accuracy	Recall (positive)	Precision	F1-score
Logistic Regression (baseline)	0.85	–	–	–
RBF–SVM pipeline (ours)	0.86	0.90	0.86	0.88

Table 3: Comparison between the simple Logistic Regression baseline and our proposed RBF–SVM pipeline (5-fold cross-validation).

Team Contributions

Jeffrey Lin:
ZiDi Yao:
Ke Ma:

References

[1] Fedesoriano. Heart failure prediction dataset. https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction, 2020. Accessed 2025-11-14.

[2] Anna Karen Gárate-Escamila, Amir Hajjam El Hassani, and Emmanuel Andrès. Classification models for heart disease prediction using feature selection and pca. *Informatics in Medicine Unlocked*, 19:100330, 2020.

[3] Henry F. Kaiser. The application of electronic computers to factor analysis. *Educational and Psychological Measurement*, 20(1):141–151, 1960. Original work published 1960.