Group X Progress 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.

In this progress report, we will discuss the steps completed so far, including dataset preprocessing, feature encoding, and the development of a Chi-Square + PCA feature-engineering pipeline. We also present initial results from models such as SVM, Logistic Regression, and Random Forest, and outline feedback from the TA along with our planned next steps.

Building on these prior studies, our project adopts a supervised learning pipeline that combines Chi-square feature selection and Principal Component Analysis (PCA) with an RBF-kernel SVM classifier. We first apply a column transformer with median imputation and standardization for numerical features, and most-frequent imputation with one-hot encoding for categorical features; Then we use a MinMaxScaler followed by `SelectKBest` with the Chi-square statistic to keep the top-k informative features, and apply PCA with the number of components chosen using the Kaiser criterion [5, 6]. The low-dimensional representation is fed into an RBF-SVM, and the full pipeline is evaluated on the Kaggle Heart Failure Prediction dataset [4] using stratified k-fold cross validation, reporting accuracy, precision, recall, F1-score, and AUC as in previous work.

2 Related Work

Our progress in this project builds directly on the foundation built in Milestone01, where we considered both the clinical motivation and the technical approaches. In that proposal, we discussed the limitations of traditional diagnostic methods and highlighted the importance of ML models that can identify multi-feature interactions such as age, cholesterol level, chest pain type, and ECG-related measurements, which observations align with prior work showing that heart disease prediction is well suited to ML due to its multivariate and non-linear nature.

As we talked about in M1, several existing studies have explored this predictive task using classical and modern machine learning techniques. Logistic regression remains a common baseline method, as demonstrated by Awan [3], who achieved approximately 85% accuracy on the Kaggle using minimal feature engineering. More advanced pipelines integrate supervised feature selection and dimensionality reduction. The Chi-Square + PCA framework proposed by Gárate-Escamila et al. [5] achieved up to 99% accuracy on multiple UCI heart disease datasets, motivating our decision in Milestone 1 to incorporate both Chi-Square filtering and PCA into our own preprocessing pipeline.

Moreover, foundational statistic work such as Kaiser factor analysis criterion [6] provides justification for retaining only principal components with eigenvalues greater than one, a rule we apply in our dimensionality reduction stage. Other work has examined optimization strategies for medical classification models, including scalable L1-regularized training [2] and multi-task predictive structure learning [1], which highlight broader approaches to improving generalization when datasets are small—one of the challenges noted in our proposal.

3 Dataset and Preprocessing

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

3.1 Dataset Description

The dataset we used in this project is the *Heart Failure Prediction Dataset* published by Fedesoriano on Kaggle [4]. 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 1.

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.

3.2 Target 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:

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

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

3.3 Feature Preprocessing

The feature matrix X was constructed by removing the target column and retaining the remaining 11 raw 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:** $\texttt{M} \to 1,\ \texttt{F} \to 0$
- **ExerciseAngina:** $\texttt{Y} \to 1,\ \texttt{N} \to 0$
- **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:

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

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

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

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

3.4 Final Processed Dataset

After applying the above processing , the final processed feature matrix contains:

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

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

This processed dataset serves as the input to the feature selection (Chi-square) and dimensionality reduction (PCA) procedures described in the next section.

4 Model Inputs (Features)

During our data preprocessing phase, we utilize `scikit-learn` built-in function `pipeline` to transform our data before feeding into Machine Learning Model The dataset has both numerical and categorical features, in the previous part we mentioned that for each numerical part we standardized them, while for categorical part we used one-hot encoding to transform them into vectors. In result, we have total 22 features after preprocessing, including 6 numerical features and 16 categorical features "(One-Hot Encoding)". We then went ahead to further transform our data by using both features selection and dimensionality reduction techniques. Which is so called Chi-PCA method [5].

- **Chi-Square:** Since in Mathematically, the Chi-square statistic is defined as:

  $$\chi^2 = \sum_i \sum_j \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

  , where O_{ij} is the observed frequency and E_{ij} is the expected frequency.

  Therefore, It is expecting a positive value for frequency, so we use `MinMaxScaler` to scale all numerical features to [0,1] range. So that all numerical features and categorical features are non-negative.

  Then we have a Hyperparameters 'k' to select top k features with highest Chi-square statistic with respect to the target label. In here we set k=9, as we found that 9 is the best parameter after conducting grid-search.

- **PCA:** After feature selection, we then applied Principal Component Analysis (PCA) to reduce the dimensionality of the selected features. PCA works by identifying the directions (principal components) in which the data varies the most, and projecting the data onto these directions.

  To determine the number of principal components to retain, we adopt the **Kaiser criterion** [6]. This rule suggests keeping only components with eigenvalues greater than 1.0.

  After doing a experiments of calculating eigenvalues for every single increase of principal components, we found that the first five components have eigenvalues greater than 1.0. Thus, we decided to retain five principal components for our final feature representation.

Hence, each patient sample is represented as a compact (data, 5) feature vector summarizing the most informative physiological and categorical characteristics. This final feature set is then used as input to our machine learning model.

5 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.

To evaluate our implementation, we compared SVM–RBF against two baseline models:

- **Logistic Regression:** This baseline model was chosen from kaggle [3], where the author didn't use features selection or dimensionality reduction, and he was able to achieve 85% accuracy.
- **Random Forest:** From this paper [5], the author has 98% accuracy by using Random Forest with Chi-PCA method. However, he used a 74 features and around 1000 datapoint of heartdiease dataset from UCL.

With SVM-RBF, we evaluate it's accuracy by using cross-validation, and we was only able to achieve 86% of accuracy.

We then use Random Forest and Logistic Regression as our model, Randomforest was able to achieve 87% accuracy, while Logistic Regression was able to achieve 85% accuracy.

Varies reason can be introduces in here, such as different dataset, the quality of dataset, or minor changes in the preprocessing phase that effect the data values meaning.

In future iterations, we plan to explore a **neural network architecture** (e.g., a multi-layer perceptron) to capture more complex feature interactions and potentially improve generalization performance. We also intent to change the detail in our preprocessing phase, such as using different order, or different preprocessing tools to present the data in a better way.

6 Evaluation Strategy and Results

6.1 Evaluation Method

Explain why you used stratified K-fold cross validation (e.g., small dataset size, need for robust evaluation).

6.2 Metrics

Explain why accuracy, precision, recall, F1, and AUC-ROC are important in medical diagnosis.

6.3 Results

Insert your figures:

7 Feedback and Future Plans

Summarize TA feedback and your improvements:

- Replace label encoding with one-hot encoding
- Consider switching from SVM to neural networks for performance gains
- Create a new GitHub branch for experiments

Team Contributions

Zhicong Lin: Background Research, Data preprocessing, feature engineering, model implementation, report writing.

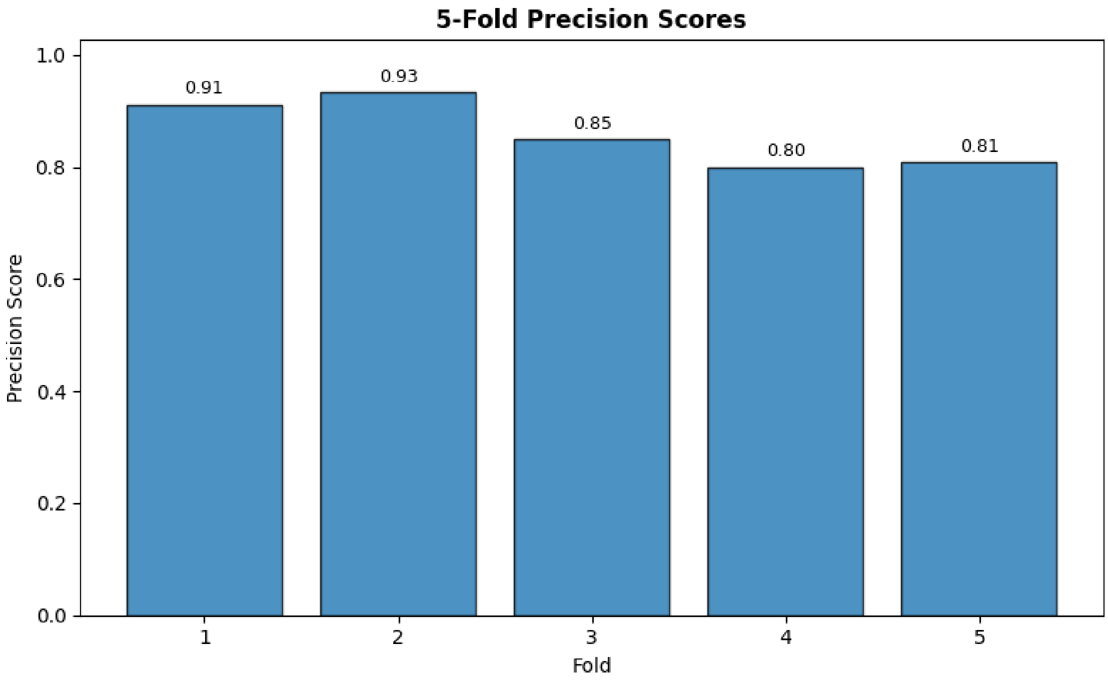


Figure 1: 5-fold precision scores.

References

[1] Rie Kubota Ando and Tong Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853, December 2005.

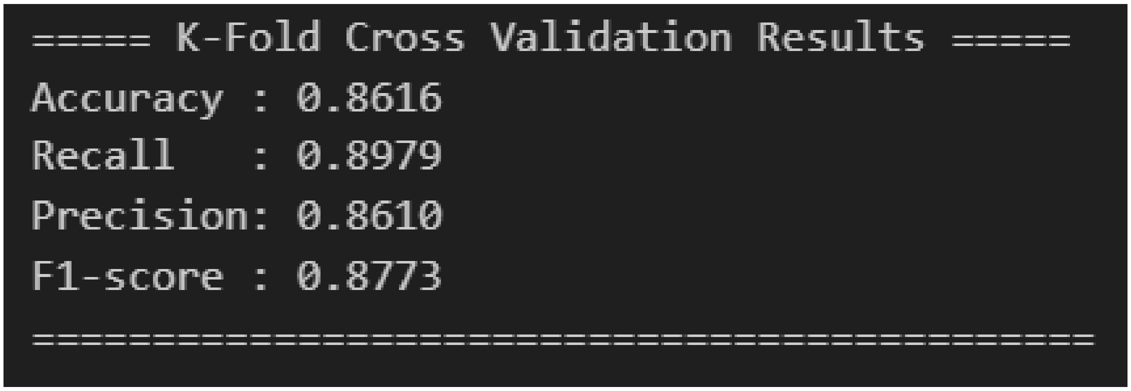
[2] Galen Andrew and Jianfeng Gao. Scalable training of L1-regularized log-linear models. In *Proceedings of the 24th International Conference on Machine Learning*, pages 33–40, 2007.

[3] M.Usman Aslam Awan. Heart disease prediction using logistic regression. Kaggle Notebook, 2020. Online; accessed: 2025-11-11.

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

[5] 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.

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



```
===== K-Fold Cross Validation Results =====
Accuracy : 0.8616
Recall   : 0.8979
Precision: 0.8610
F1-score : 0.8773
============================================
```

Figure 2: Summary metrics across folds.

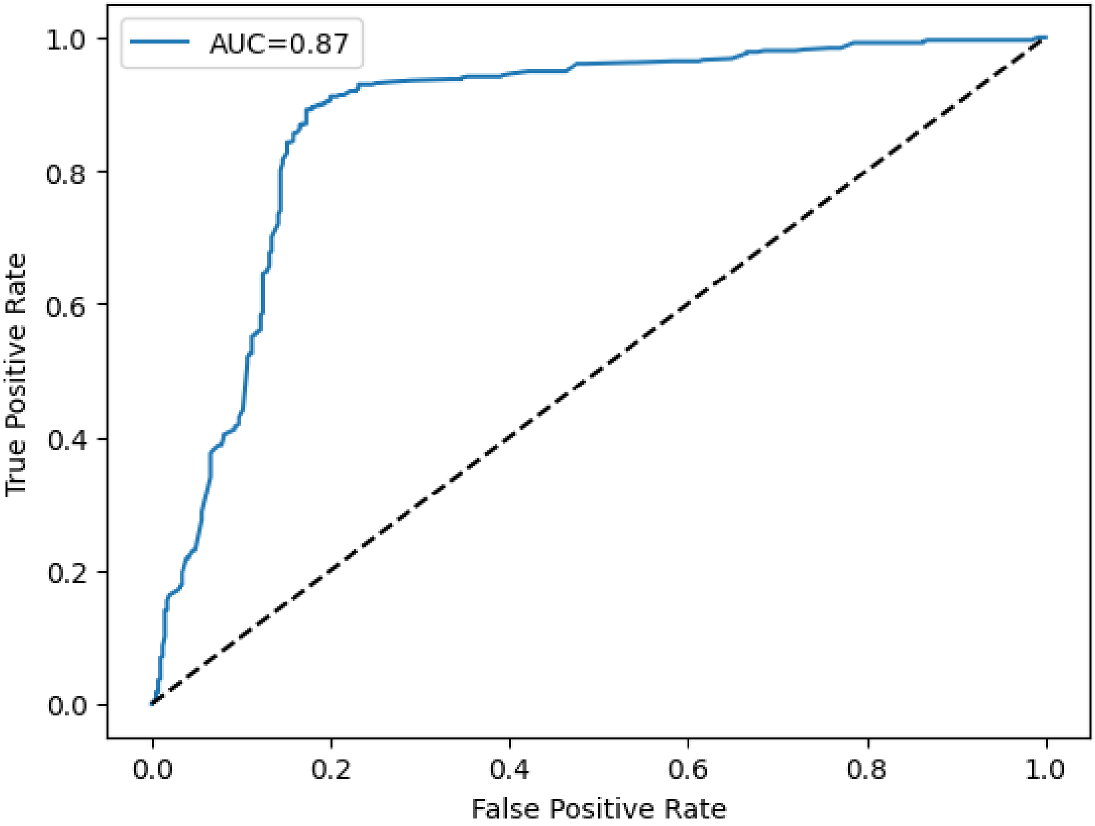


Figure 3: AUC–ROC curves.

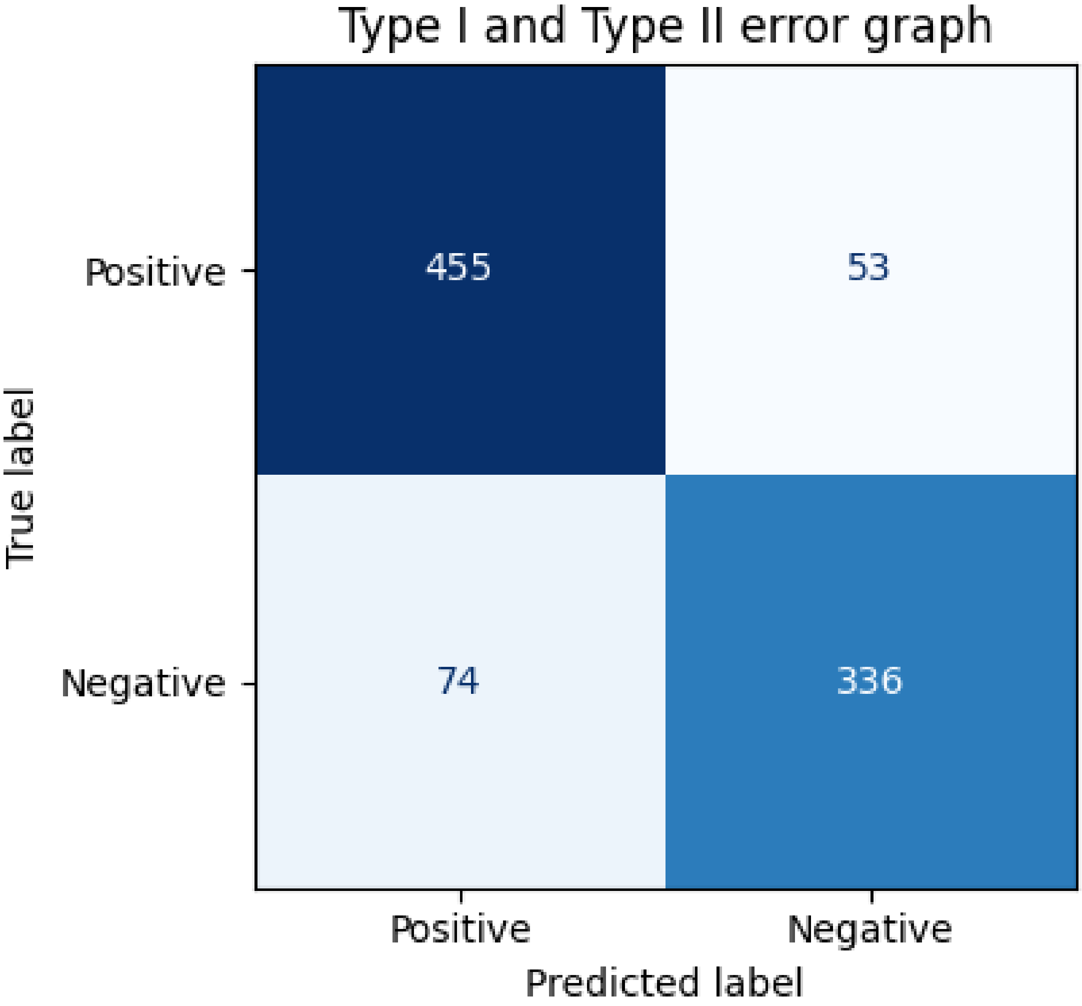


Figure 4: Confusion matrices.