CS 622 Project Proposal

Jiaqi Li and Shen C. Huang

University of Nevada – Las Vegas

Abstract

Research in detection and treatment for heart diseases plays a large role in advancing medicine. In this project proposal, we are to solve a binary classification problem for predicting heart diseases. We propose to apply three different methods: logistic regression, support vector machine, and K-Nearest Neighbors given some specific parameters described in section 3. In addition, we will preprocess the data before modeling. The dataset [1] was found on Kaggle.

1 Problem Definition

The task is to build a binary classification model to predict whether a patient has heart disease or not, based on their demographic and clinical information. The output variable is "output", which takes a value of 1 if the patient has heart disease and 0 otherwise. This is a supervised learning problem, where the target variable is known for the training set and the goal is to develop a model that can accurately predict the target variable for new, unseen data.

2 Proposed Work

List of the ML methods we will implement:
Based on the topics we discussed in class, the following machine-learning algorithms will be applied.

2.1 Logistic Regression

It's a simple and efficient algorithm used for binary classification tasks.

2.2 Support Vector Machines (SVM)

It's a powerful algorithm used for binary classification tasks that can handle both linear and non-linear decision boundaries.

2.3 K-Nearest Neighbors (KNN)

It's a non-parametric algorithm that can be used for both classification and regression tasks. It's based on the idea that similar data points tend to have similar labels.

3 Brief description of the data

The dataset [1] has a total of 303 samples or rows, with each row representing a different patient. There are 13 input features that describe various demographic and clinical symptoms, including age, sex, chest pain type, resting blood pressure, serum cholesterol level, fasting blood sugar level, resting electrocardiographic results, maximum heart rate achieved, ST depression, the slope of the peak exercise, number of major vessels (0-3) colored by fluoroscopy, and thalassemia.
The last column "output" indicates whether a patient has diabetes or not.

4 Description of Data settings or preprocessing

4.1 Handling Missing Values

The dataset may or may not contain missing values, but we need to prepare for this concern and see how to deal with it if there are missing values. The missing values can be represented as NaN or NULL values. In such cases, we need to decide how to handle these missing values, such as dropping the rows with missing values or filling them with a suitable value like the mean or median of the feature.

4.2 Encoding categorical variables

Some of the features in the dataset, such as gender and chest pain type, are categorical variables. To use these variables in machine learning algorithms, they need to be encoded as numerical values. One common approach is one-hot encoding, where a binary variable is created for each category in the original feature.

4.3 Normalization or scaling

Some machine learning algorithms, such as K-nearest neighbors and support vector machines, work better when the features are normalized or scaled to a specific range. For example, we can scale the features to have zero mean and unit variance or normalize them to have a range of 0 to 1.

4.4 Feature Selection

Since the dataset has a small number of samples and features, we can try to identify the most relevant features to the classification problem using feature selection techniques. This can help reduce overfitting and improve the accuracy and interpretability of the model.

5 Experiment Design

5.1 Data Preprocessing

Perform data preprocessing steps like handling missing values, encoding categorical variables, normalizing or scaling the features, and feature selection.

5.2 Train-validation-test split

Perform data preprocessing steps like handling missing values, encoding categorical variables, normalizing or scaling the features, and feature selection.

5.3 Model Selection with k-fold cross-validation

Use k-fold cross-validation (e.g., with $k = 5$ or $k = 10$) on the training set to compare the performance of different machine learning algorithms on this dataset. For each algorithm, use a grid search to tune the hyperparameters, and evaluate the performance using the average cross-validation score.

5.4 Evaluation metrics

Once the best-performing model is selected, evaluate its performance on the test set using evaluation metrics like accuracy, precision, recall, F1-score, and area under the receiver operating characteristic (ROC) curve

5.5 Inerpretation

Interpret the selected model to understand the features that are most important for predicting heart attacks. Visualize the decision boundaries of the model to see how it separates the positive and negative samples.

5.6 Deployment

Deploy the selected model in a production environment, where it can be used to predict heart attacks in new patients based on their symptoms.

6 Expected Results & Impact

The goal is to develop a model that can accurately predict the patient's potential heart attack risk based on the health condition.

References

[1] R. Rahman. (2023) Heart attack analysis & prediction dataset. Version 2. [Online]. Available: https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset