Predicting Coronary Heart Disease in Western Cape

Machine Learning for Quantitative Professionals Report

José Pedro Pessoa dos Santos

March 2025

Contents

1 Introduction

This report analyses various Machine Learning algorithms to predict Coronary Heart Disease (CHD) in male individuals from the Western Cape, South Africa. The dataset comprises individuals aged 15-64 and uses the presence or absence of CHD as the target variable, with 302 cases without CHD and 160 with CHD - indicating an unbalanced class distribution. Nine features are used for prediction. All except *famhist* are numerical, and their distributions were analysed for further processing. Detailed information is provided in table 1.

Feature	**Distribution**	**Description**
adiposity	Normal	
type A	Normal	type-A behaviour
obesity	Normal	
age	Normal	age at onset
sbp	Skewed	systolic blood pressure
tobacco	Skewed	cumulative tobacco (kg)
ldl	Skewed	low density lipoprotein cholesterol
alcohol	Skewed	current alcohol consumption
famhist	Categorical	family history of heart disease (Present, Absent)

Table 1: Feature distributions and description.

The following sections describe the entire machine learning pipeline, covering data processing, model building, evaluation, and presentation of results with a discussion of the findings.

2 Data Processing

After initially loading the data and removing rows with missing values, the data follows a preprocessing pipeline in order to prepare it for the model training phase. This pipeline encompasses the following steps:

1. Identification of Feature Types
    - Categorical Features
    - Numerical Features
2. Transformation Strategies
    - **Normally Distributed Features:** Processed with a *StandardScaler* to standardise the data.
    - **Skewed Features:** High skewness (> 1) features are log-transformed to stabilise variance and then scaled.

- **Bounded Features:** Features confined to the [0, 1] range are scaled using *MinMaxScaler*.
- **Categorical Data:** Application of one-hot encoding.

The appropriate transformation is applied to each subset of features. Optionally, PCA can be applied to reduce the dimensionality of the feature set while retaining 95% of the variance. The initial feature correlation can be observed in figure 1:

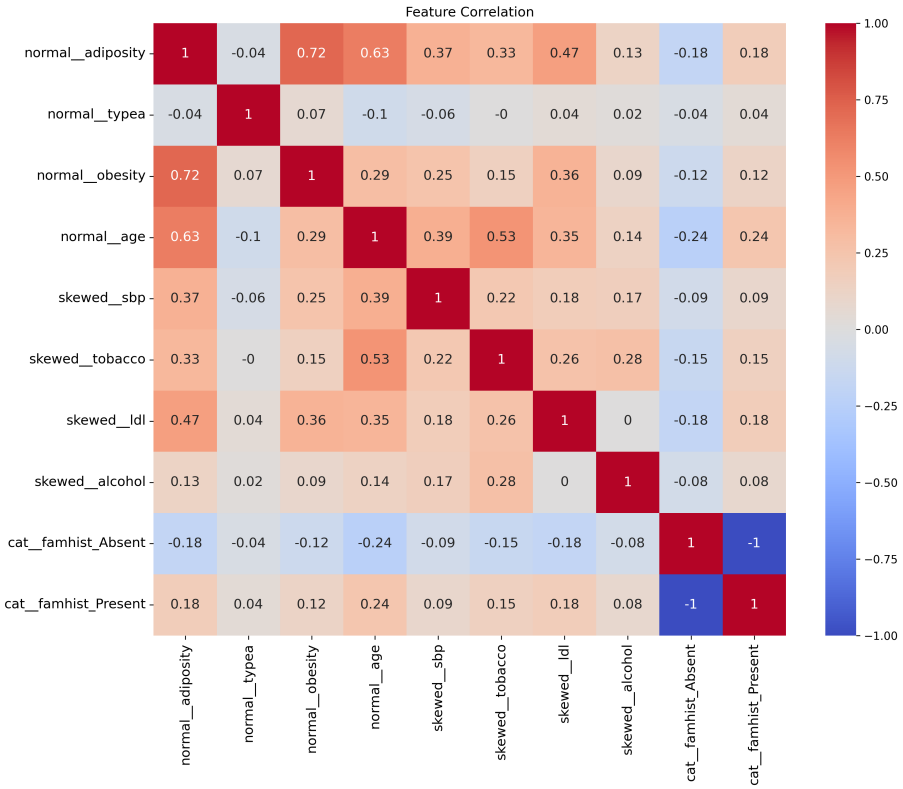


Figure 1: Feature correlation matrix

3 Model Building and Evaluation

After preprocessing, data was split into 70% training and 30% testing sets. Several machine learning models were trained and evaluated, some undergoing hyperparameter tuning via cross-validation to determine the best configuration. A list of the models, including a brief description and tuned hyperparameters, is provided below.

- **Decision Tree:** Uses a decision tree classifier. Fine-tuned on `ccp_alpha`, which sets the pruning threshold to control tree complexity and prevent overfitting.
- **Random Forest:** Ensemble of decision trees. Fine-tuned on `max_features`, which determines how many features are considered for splitting at each node, affecting model generalisation.

- **AdaBoost:** Boosts a series of shallow decision trees (with a maximum depth of 3 as base learners). Fine-tuned on `learning_rate`, which scales the contribution of each weak learner to improve convergence speed and performance.
- **Gradient Boosting:** Builds trees sequentially, where each new tree corrects the errors of the previous ones. Fine-tuned on `learning_rate`, which controls the impact of each new tree on the final prediction.
- **Logistic Regression (L2):** Linear classifier that models probabilities using the logistic function. Fine-tuned on `C`, where lower values impose stronger regularisation to reduce overfitting.
- **k-Nearest Neighbors (kNN):** Non-parametric classifier that assigns class labels based on the majority vote of the k closest data points. Fine-tuned on the number of neighbours (`k`), which influences the decision boundary.
- **Support Vector Machine (SVM):** Finds the optimal hyperplane to separate classes. Fine-tuned on `gamma`, affecting the influence of single data points in the kernel space, and `C`, defining margin maximisation with misclassification tolerance.
- **Gaussian Naive Bayes:** Probabilistic classifier that assumes features follow a Gaussian distribution.
- **Linear Discriminant Analysis (LDA):** Projects data onto a lower-dimensional space to maximize class separability using a linear combination of features.
- **Quadratic Discriminant Analysis (QDA):** Similar to LDA but allows for quadratic decision boundaries.

4 Analysis of Model Performance

After model training and fine-tuning, performance was evaluated on the testing set. Due to the initial class imbalance, the ROC curve was chosen as the primary evaluation metric. It allows for a clear visualization of the results and ease of comparison. Below is a brief explanation of how to compute it.

Confusion Matrix: The confusion matrix provides a summary of model performance by comparing actual and predicted class labels. It helps identify the types of errors the model makes and is the basis for calculating the ROC Curve.

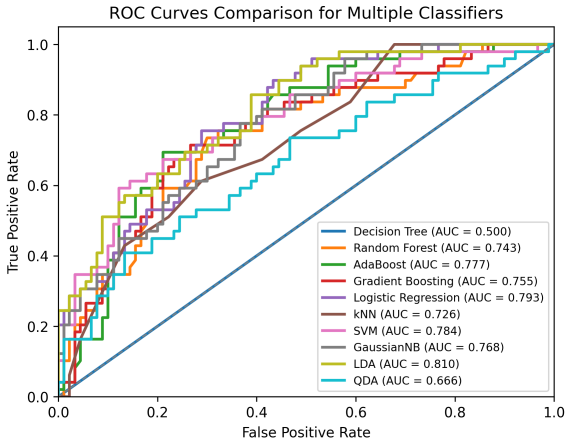
	Predicted Positive	Predicted Negative
Actual Positive	*True Positive* (TP)	*False Negative* (FN)
Actual Negative	*False Positive* (FP)	*True Negative* (TN)

ROC Curve and AUC: The Receiver Operating Characteristic (ROC) curve is a graphical representation of a classifier's performance across different thresholds. It plots the True Positive Rate (TPR) against the False Positive Rate (FPR). The Area Under

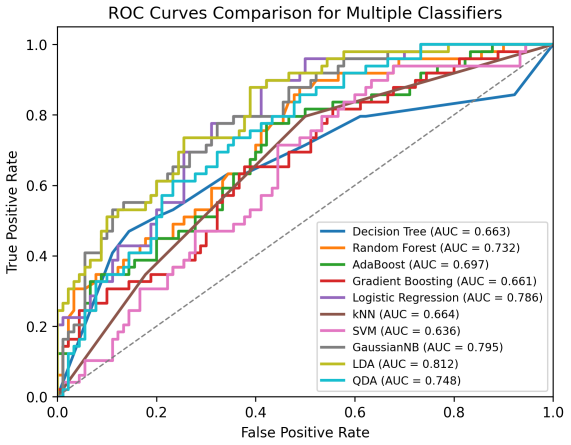
the Curve (AUC) is a single value that reflects the model's ability to distinguish between classes. A model with an AUC of 1.0 has perfect discrimination, while an AUC of 0.5 is similar to random guessing. It is defined as:

$$TPR = \frac{TP}{TP + FN} \quad \text{and} \quad FPR = \frac{FP}{FP + TN}.$$

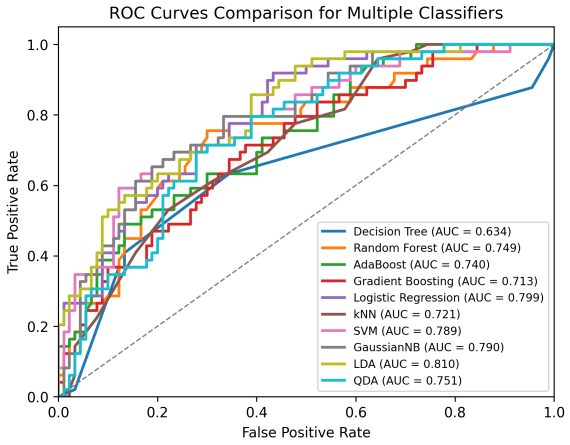
Given the class imbalance observed during data analysis and the feature correlations in Table 1, four evaluation setups were tested. The first used only the preprocessed data (no SMOTE or PCA; Figure 2a). The second applied both SMOTE, balancing the distribution from 302 negatives and 106 positives to 212 each, and PCA (Figure 2b). The third used only PCA (Figure 2c), and the fourth applied only SMOTE (Figure 2d).



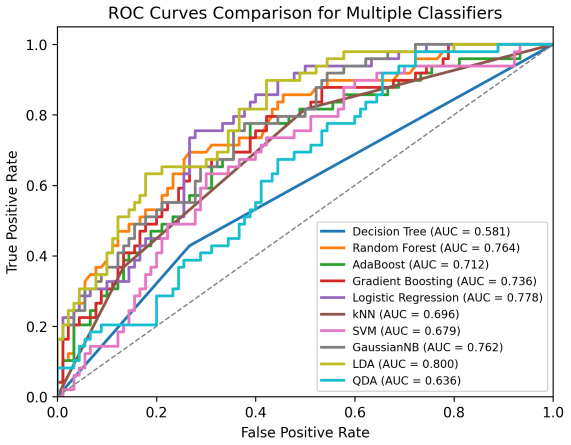
(a) ROC Curves



(b) ROC Curves with SMOTE and PCA



(c) ROC Curves with PCA



(d) ROC Curves with SMOTE

Across the four experimental setups, LDA consistently achieves the highest AUC ($\sim$ 0.81), closely followed by Logistic Regression (AUC $\sim$ 0.78–0.80). Their strong performance likely results from two factors: (1) the dataset's features clearly distinguish between CHD-positive and CHD-negative individuals, allowing linear decision boundaries to be effective; (2) these models manage overfitting - LDA through pooled covariance

estimates and Logistic Regression via L2 regularization - which is particularly beneficial given the limited sample size and class imbalance.

Ensemble classifiers, including Random Forest, AdaBoost, and Gradient Boosting, also demonstrate strong predictive performance (AUC $\sim$ 0.66–0.78). These models effectively capture non-linear relationships in the data. Random Forest benefits from averaging multiple decision trees to reduce variance, while boosting algorithms incrementally improve predictions on misclassified cases. When SMOTE is applied (balancing the minority CHD-positive class) and PCA is introduced (addressing feature redundancy), these models show modest performance improvements.

Support Vector Machine (SVM) shows variable performance across setups (AUC ranging from 0.64 to 0.79). While capable of capturing complex boundaries with appropriate kernels, its effectiveness depends on hyperparameter tuning and sufficient representation of the minority class. Similarly, k-Nearest Neighbors (kNN) performs relatively poorly (AUC around 0.66–0.73), as distance-based classification struggles in higher-dimensional spaces and imbalanced scenarios.

The weakest model is the standalone Decision Tree (AUC around 0.50–0.66) as it tends to overfit training data and generalise poorly without ensemble aggregation. Quadratic Discriminant Analysis (QDA) also underperforms (AUC approximately 0.63–0.75), likely due to its strong assumption that each class has a distinct covariance structure.

Overall, the top-performing models effectively predict the correct class without overfitting, with LDA delivering the strongest performance. In contrast, models that performed poorly struggled due to assumptions that don't fit the dataset and the class imbalance present in this task that even SMOTE couldn't completely fix.