Predicting Coronary Artery Disease Using Machine Learning

Keshab Raj Dahal¹, Nawa Raj Pokhrel², Pramesh Subedi³, Santosh Gaire⁴, Ramesh Bhandari⁵, Mimansa Dahal⁶, and Modesola Giwa⁷

¹ Department of Mathematics, State University of New York at Cortland, Cortland, NY USA
² Department of Physics and Computer Science, Xavier University of Louisiana, New Orleans, LA USA
³ Department of Mathematics, Ohio University, Athens, OH USA
⁴ Department of Physics, The Catholic University of America, Washington, DC, USA
⁵ Department of Physics, Central Michigan University, Mount Pleasant, MI, USA
⁶ Department of Mathematics and Statistics, Binghamton University, Binghamton, NY USA
Correspondence: Keshab Raj Dahal, Department of Mathematics, State University of New York at Cortland, 22 Graham Ave, Cortland, NY, USA. E-mail: keshabraj.dahal@cortland.edu

Received: March 8, 2024       Accepted: May 23, 2024       Online Published: May 29, 2024
doi:10.5539/ijsp.v13n2p1       URL: https://doi.org/10.5539/ijsp.v13n2p1

Abstract

Developing a predictive model for detecting Coronary Artery Disease (CAD) is crucial due to its high global fatality rate of approximately 17.9 million people annually. With the advancements in artificial intelligence, the availability of large-scale data, and increased access to computational capability, it is feasible to create robust models that can detect CAD with high precision. This study aims to build a predictive model that can assist health workers in the timely detection of CAD and ultimately reduce mortality. This study performs a comparative analysis of four supervised classification machine learning algorithms- Logistic regression (LR), Support vector machine (SVM), Extreme gradient boosting (XGBoost), and Artificial neural network (ANN), in predicting the case-control status of the patient. Chi-squared and lasso criteria are employed to select the most relevant ones from the available features. The performance of the employed models is compared using sensitivity, specificity, accuracy, and the area under the receiver operating characteristic (ROC) curve (AUC). The experimental results indicate that the LR model is the most effective and accurate among the models tested, and its implementation can improve the detection of CAD in clinical settings.

Keywords: machine learning, coronary artery disease, classification, neural network, XGBoost, support vector machine

1. Introduction

Coronary artery disease, commonly called CAD, is a widespread cardiovascular disease affecting many individuals worldwide (Benjamin et al., 2019). This condition damages the coronary arteries, increasing the risk of a heart attack or stroke. The arteries that supply oxygen and nutrient-rich blood to the heart muscle become hardened and narrowed, leading to CAD (Fuster et al., 1992). It is one of the leading causes of mortality among both males and females globally. In 2018, the American Heart Association reported that CAD alone was responsible for 13 percent of fatalities in the United States (Akella and Akella, 2021; Ling et al., 2021). Clinical symptoms of CAD are on the rise everywhere, regardless of geography (Mozaffarian et al., 2015). That’s why the scientific community is focused on developing and utilizing cutting-edge early detection techniques to handle this issue.

CAD is a severe condition that requires careful attention and effective management (Herzog, 2003). Fortunately, incorporating classification models can lead to more effective and tailored treatment strategies, optimizing patient care (Ahmed et al., 2020). Research studies have demonstrated the utility of classification models in early detection and diagnosis of CAD. A study published in the European Heart Journal revealed that classification algorithms for risk prediction in CAD better identified high-risk patients than traditional risk assessment methods (Hippisley-Cox et al., 2017). By embracing classification models, we can better identify high-risk patients and improve decision-making for individuals, hospital systems, and doctors. Moreover, classification outcomes have been proven to aid doctors in predicting disease progression and prognosis. A study published in the Journal of the American College of Cardiology indicated that machine learning-based classifiers accurately predicted adverse cardiac events in CAD patients, helping doctors anticipate potential outcomes and adjust management plans accordingly (Schlett et al., 2011). By incorporating classification results into treatment planning, doctors can enhance personalized treatment recommendations for CAD patients. This approach optimizes patient care and leads to more effective and tailored treatment strategies.
With significant advancements in medical treatment, several procedures can now detect or monitor cardiac problems. These diagnostic procedures include the electrocardiogram (ECG or EKG), echocardiogram, cardiac catheterization and angiography, computed tomography scan (CT scan), and others. They provide high-quality visualization of the coronary arteries with better accuracy. However, there is a present shortage of cardiac imagers who have obtained proper training, which can make detecting CAD more complicated than it needs to be. Machine learning (ML) can solve this issue by enabling an automatic method to assess examples and draw consistent and accurate conclusions. Additionally, machine learning techniques can lead to various subsequent diagnoses.

A proper model developed with an optimal set of risk indicators can help detect CAD in earlier stages and reduce the likelihood of developing it in the near future. A plethora of research has been published to study how certain variables correlate with CAD. The prominent factors include but are not limited to dietary behaviors, a lack of physical activity, smoking, obesity, consumption of alcoholic drinks, diabetes, high blood pressure, and high blood cholesterol levels (Mozaarian et al., 2008; Poirier, 2008). A varying degree of success is seen concerning the accuracy and robustness of the models. One possible reason for not achieving the expected outcome could be in the variable selection process. The developed model performs reasonably better if a good combination of features is considered. One of the contributions of this study is selecting the variables by employing established statistical criteria to pinpoint the most influencing variables from available alternatives.

This study considers the comparative computational framework by utilizing supervised classification machine-learning models to predict coronary artery disease (CAD). The birds-eye view of the proposed research framework is expressed in Figure 1 via the schematic diagram. As the diagram outlines, the proposed study utilizes the Z-Alizadeh Sani dataset to build the model. After the data collected, it has been normalized using the minmax normalization technique. From classical machine learning to sophisticated ANN models are employed to serve the purpose of our study. Once the hyperparameters are tuned, the input data is fed into the respective models for classification. The identification of the optimal model is determined by comparing the performances of these distinct algorithms, with a particular focus on evaluating metrics such as sensitivity, specificity, accuracy, and the area under the receiver operating characteristic (ROC) curve (AUC).

The rest of the paper is organized as follows. Section 2 explains the related work in this field. Section 3 briefly illustrate the analytical structure of the implemented model architectures in the context. The standard assessment metrics are explained in Section 4. Model outcomes are discussed in Section 5. It also explains the predictive capability of the model, the hyperparameter tuning process, and variable selection criteria. Finally, Section 6 presents the conclusion and future work, followed by a list of references.

2. Related Work

Various works have explored machine learning algorithms for predicting Coronary Artery Disease (CAD). The results of these studies have consistently shown machine learning’s potential for accurate CAD prediction.

Akella and Akella (2021) used different methods such as a Generalized linear model, decision tree, random forest, support vector machine, neural network, and K-nearest neighbor on a UC Irvine Cleveland dataset to predict CAD. All models achieved accuracy greater than 79%. 93% was the highest accuracy obtained through neural networks among the other models. Higher accuracies seen in these ML models may provide predictive value in CAD.

A few more machine learning algorithms were used by Naushad et al. (2018) to predict CAD. The study applied ensemble machine learning algorithms (EMLA), multifactor dimensionality reduction (MDR), and recursive partitioning (RP). Among these models, EMLA shows better performance in accuracy with 89.3%. The EMLA model results show hyper-
tension and alcohol intake as the key predictors of CAD. Controlling the blood pressure and avoiding alcohol intake was shown to reduce the risk for CAD.

Dahal and Gautam (2020) performed logistic regression, random forest, support vector machine, and K-nearest neighbors algorithms on the Z-Alizadeh sani dataset to detect CAD. The results indicate that the support vector machine is the most efficient technique and shows a better performance with 89.47% accuracy in comparison to other tested methods. Dipto et al. (2020) studied a dataset that includes several features such as age, sex, weight, body mass index, etc. They built an Artificial neural network (ANN) model to predict CAD and obtained an accuracy of around 93.5%. The dataset contains class imbalance, so the synthetic minority oversampling technique (SMOTE) technique has been used to balance the dataset.

In addition, different classifiers have been chosen to predict CAD according to their performance. Reports from previous studies done by Hsu et al. (2021) show that random forest can be used as an effective algorithm to predict CAD based on the area under the receiver operating characteristic (ROC) curve (AUC) around 0.94. Muhammad et al. (2021) used machine learning predictive models for CAD prediction, and among the used models, random forest stands as the best model with an AUC of 0.92. Another study that finds the random forest as the best CAD prediction model is by Sharma et al. (2022) having an AUC of 0.85.

Studies conducted by Jiang et al. (2021) compared the predictive performance of seven machine learning models for cardiovascular disease risk prediction in a Kazakh Chinese population and reported logistic regression as the best classifier having an AUC of 0.872 with the lowest biased score of 0.078 and the highest sensitivity of 97.1%.

In the study, Sinha and Sharma (2021) employed data mining techniques to develop a smart heart disease prediction (SHDP) algorithm for diagnosing heart disease, the algorithm utilized non-linear support vector classification (SVC) with a radial basis function (RBF) kernel. The dataset was divided into 80% for training and 20% for testing. The evaluation metrics, accuracy, specificity, sensitivity, precision, and F-score, were utilized to assess the algorithm’s performance. The proposed SHDP algorithm achieved an impressive average accuracy of 93.53%, specificity of 89.22%, sensitivity of 91.24%, precision of 86.98%, and F-score of 93.53%. These results highlight the potential of the SHDP algorithm for predicting cardiac arrest and its potential application in clinical settings.

The article by Sitar-tăuet et al. (2009) assessed the performance of several machine learning models, including decision trees, k-nearest neighbors, support vector machines, and naive Bayes models, for identifying patients with cardiovascular disease. The study used accuracy and AUC as performance metrics to evaluate these models. The results showed that the naive Bayes model achieved an accuracy of 0.87 and an AUC of 0.93 for identifying patients without stroke or peripheral artery disease. For identifying patients with coronary artery disease, the decision tree model achieved an accuracy of 0.81 and an AUC of 0.83. In contrast, the support vector machine model achieved an accuracy of 0.82 and an AUC of 0.85. These findings suggest that machine learning algorithms can effectively evaluate cardiovascular disease risk. They may offer advantages over traditional methods in specific contexts when evaluated using metrics such as accuracy and AUC.

3. Modeling Approach

3.1 A Brief Overview of Logistic Regression (LR)

Logistic regression (LR) is a versatile and widely used statistical approach for modeling dichotomous data. As a member of the generalized linear models family, it is a powerful tool for classifying binary dependent variables based on a set of independent variables. Extensive research (DeMaris, 1995; Field, 2009; Kleinbaum et al., 2002; LaValley, 2008; Menard, 2002; Pampel, 2020; Sperandei, 2014; Wright, 1995) has demonstrated the effectiveness and application of logistic regression in various fields, including medicine, social sciences, and economics. For instance, studies such as (Musa, 2013) have shown that LR outperforms other approaches for binary classification. Additionally, LR can be applied to predict outcomes in medical research, as demonstrated by (Hosmer Jr et al., 2013).

Consider a scenario where we have a dataset with $n$ observations and $k$ features represented by an input matrix $X$. Additionally, we have a binary outcome vector $Y$ where each response $y_i$ for every input $x_i, i = 1, 2, \ldots, k$, can take either 1 or 0. Based on their corresponding outcomes, we classify instances as belonging to either the positive or negative class. Let’s say instances with outcomes $y_i = 1$ belong to the positive class, and those with outcomes $y_i = 0$ belong to the negative class. We can use the LR model with a sigmoid activation function to classify an instance $x_i$ as positive or negative. The sigmoid function is defined as follows:

$$E(y_i|x_i) = \pi_i = \frac{e^{\beta x}}{1 + e^{\beta x}} = \frac{1}{1 + e^{-\beta x_i}}$$

where $\beta = (\beta_0, \beta_1, \beta_2, \cdots, \beta_k)^T$ is a vector of parameters and is estimated by maximum likelihood method. Instead of supplying the exact values 0 and 1, the fitted model provides the probabilistic values that lie between 0 and 1. So, the
predicted values are assigned one if \( \hat{\pi}_i \) is greater than or equal to some threshold and 0 otherwise.

Due to its simplicity and great interpretability, LR is widely used in binary classification problems in applied sciences such as medicine (Lemon et al., 2003; Peterson et al., 1995; Stoltzfus, 2011), biology and epidemiology (Abreu et al., 2009; Binder et al., 2019; Kleinbaum et al., 1982; Thompson et al., 2018). Moreover, research has shown that LR is competitive with others. For example, Jiang et al. have mentioned that a study designed to predict myocardial ischemia demonstrated that the predictive performance of LR was similar to that of SVM, which was consistent with their study (Jiang et al., 2021). A recently published systematic review also suggested that ML showed no performance benefit over LR for clinical prediction models (Budhathoki et al., 2023; Christodoulou et al., 2019).

3.2 A Brief Overview of Support Vector Machine (SVM)

Support vector machine (SVM) is a widely used, powerful, and robust machine learning algorithm that has made significant achievements in various fields due to its attractive features and promising empirical performance. It was first proposed by Vapnik and Chervonenkis in 1963 and used for both classification and regression. When used for classification, SVM is also known as support vector classifier (SVC) (Boser et al., 1992; Chervonenkis, 2013; Durgesh and Lekha, 2010; Li et al., 2009; Vapnik, 1995).

To predict the class for every input \( x_i \), where \( i \) is from 1 to \( p \), the SVC model finds a hyperplane that creates a boundary between the types of data. This hyperplane is a flat subspace of dimension \( p - 1 \) in a \( p \)-dimensional space. The input matrix \( X \) is a \( nxp \) matrix, where \( n \) is the number of observations and \( p \) is the number of features. The binary outcome vector \( Y \) falls into one of the classes (−1, 1). The SVC model predicts the class by computing the hyperplane defined by:

\[
\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p = 0
\]

where \( \beta = (\beta_0, \beta_1, \beta_2, \cdots, \beta_p)^T \) is a vector of parameters and are estimated by optimization method with the property \((\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p) > 0 \) if \( y = 1 \) and \((\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p) < 0 \) if \( y = -1 \). SVC offers a variety of kernels including linear, radial, polynomial, and more, as well as hyperparameters such as cost (regularization parameter) and gamma (kernel coefficient for RBF). These factors all work together to control the performance of the model and enable it to deliver accurate results. For instance, if the boundary between two classes is linear, the SVC’s linear kernel performs well. Additionally, the tuning hyperparameters play a crucial role in balancing the bias and variance trade-off. To avoid underfitting or overfitting, it’s important to adjust these tuning parameters to the right level. The small values of these tuning parameters underfits, whereas the large values overfit the data (Claesen and De Moor, 2015; Géron, 2019; Wang et al., 2018).

The Support Vector Machines (SVMs) are powerful tools that use kernels for binary data classification and regression. They have been shown to outperform many other machine learning models like artificial neural networks in diverse areas such as pattern recognition, text categorization, biomedicine, and financial regression (Byvatov et al., 2003; Dahal and Gautam, 2020; Doniger et al., 2002; Shao et al., 2014).

3.3 A Brief Overview of Extreme Gradient Boosting (XGBoost)

Extreme gradient boosting (XGBoost) is a popular machine-learning framework that uses gradient-boosted decision trees and is known for its scalability and distribution capabilities. It is widely used for regression, classification, and ranking applications due to its ability to perform parallel tree boosting, which is more efficient than sequential tree boosting (Bekkerman et al., 2011). XGBoost leverages supervised learning, decision trees, ensemble learning, and gradient boosting, making it a reliable and accurate library for all applications (Chen and Guestrin, 2016).
The architecture of XGBoost, illustrated in Figure 2, utilizes a sequential process to progressively reduce residuals from previous trees to train subsequent trees until they reach an acceptable level. The final predicted output is obtained by summing up the results from all the trees in the ensemble. This iterative process improves the overall predictive power of the model and enhances its ability to capture complex relationships in the data. The mathematical formulation for XGBoost is presented in the following equation:

$$\hat{y}_i = \sum_{k=1}^{n} f_k(x_i), \quad f_k \in F,$$

where $F$ refers to the domain of classification tree, $f_k$ corresponds a particular tree, so $f_k(x_i)$ represents the result that tree $k$ produced, while $\hat{y}_i$ is the value that was predicted for the $i^{th}$ occurrence of $x_i$. A thorough mathematical derivation of the model is provided in (Chen and Guestrin, 2016).

Our specific motivation for employing this model in our analysis stemmed from its remarkable attributes, such as speed, accuracy, and adeptness in managing intricate relationships. These features have contributed to its extensive acceptance in competitive scenarios and practical applications. Moreover, we opted to leverage XGBoost’s capabilities, particularly due to the proven effectiveness of decision tree-based algorithms in handling moderate-sized structured and tabular datasets. Additionally, XGBoost offers a feature importance score, a valuable tool for comprehending the importance of different features in the prediction process.

3.4 A Brief Overview of Artificial Neural Networks (ANNs)

Artificial neural networks (ANNs), usually called neural networks (NNs), are computing systems inspired by the biological neural networks that constitute animal brains (Bland et al., 2020; Moayedi et al., 2020). They are composed of layers of interconnected simple processing units, called neurons, with a connection between different layers through the adjustable weights (Dahal et al., 2021; Mohandes et al., 1998). Each ANN consists of three main components: an input layer, one or more hidden layer(s), and an output layer. It contains many parameters such as weights, bias, optimizer, learning rate, number of hidden layers, number of neurons in each hidden layer, batch size, etc. Some of the parameters, such as weights and bias, are optimized (tuned) during the training process where as others, such as optimizer, learning rate, number of neurons in each hidden layer, and batch size, need to be selected properly using validation data. Moreover, a proper choice of the activation function improves the model’s accuracy. The most widely used activation functions are Sigmoid (Nwankpa et al., 2018; Turian et al., 2009), Rectified Linear unit (Nair and Hinton, 2010), and the SoftPlus (Glorot et al., 2011).

A general fully connected neural network with $n$ predictors, one hidden layer with $m$ neurons, and an output layer containing one neuron is illustrated in Figure 3. In input layer, the $n$ predictor is represented by an input vector, $\bar{x} = (x_1, x_2, \ldots, x_n)^T$. $W_f$ and $W_g$ are weight matrices of order $n \times 1$ and $m \times 1$ respectively. $b_f$ and $b_g$ are bias whereas $f$ and $g$ are activation functions. The vector, $\bar{o} = (o_1, o_2, \ldots, o_m)^T$, is the hidden layer’s output, thus input for the output layer. Finally, $g(\bar{o} * W_g + b_g)$ is the network’s final output.

![Figure 3. General architecture of neural network with n predictors, single hidden layer with m neurons, and an output layer](image-url)
methods that adjust to the data without explicitly specifying the underlying model’s functional or distributional form (Bhandari et al., 2022a; Dahal et al., 2023; Pokhrel et al., 2022, 2024). Neural networks have been applied to various real-world classification problems, including medical diagnosis, crime classification, fault detection, and speech recognition, and are superior to classical models (Baxt, 1990, 1991; Bourlard and Morgan, 1993; Burke, 1994; Burke et al., 1997; Dahal et al., 2020; Dahal and Gautam, 2020; Hoskins et al., 1990; Lippmann, 1989).

4. Model Performance Metrics

Coronary Artery Disease (CAD) prediction from the constructed model is assessed through the four different performance metrics; sensitivity, specificity, accuracy, and area under the receiver operating characteristic (ROC) curve (AUC). The stated matrices help us determine the best model in terms of accuracy and reliability. The elements of the confusion matrix are utilized to find three important metrics; sensitivity, specificity, and accuracy. The analytical form of a confusion matrix is defined in Table 1.

![Table 1. Confusion matrix](image)

The proportion of the actual positive cases (patients suffering from CAD) that is correctly predicted as positive is called sensitivity. Similarly, the proportion of the actual negative cases (patients not suffering from CAD) that are correctly predicted as negative is called specificity, and the proportion of the cases that are predicted accurately (predicted positive for the patients suffering from CAD and predicted negative for the patients not suffering from CAD) is called the accuracy.

The analytical form of these metrics is defined by

\[
\text{Sensitivity} = \frac{\text{True positive (TP)}}{\text{True positive (TP)} + \text{False negative (FN)}}
\]

\[
\text{Specificity} = \frac{\text{True negative (TN)}}{\text{True negative (TN)} + \text{False positive (FP)}}
\]

\[
\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP+FN+TN+FP}}
\]

Sensitivity and specificity are inversely proportional to each other. Receiver operating characteristic (ROC) curve is commonly used to characterize the sensitivity/specificity trade-offs for a binary classifier. The ROC curve is obtained by plotting the false positive rate (1-specificity) on x-axis against the sensitivity on y-axis at various threshold settings. Area under the ROC curve (AUC) is one of the most important matrices to measure the performance of the model. Its value lies between 0 and 1. A model is said to be an excellent if its AUC is close to 1. The higher the AUC, the better the model, and vice-versa. The model with the highest statistics: sensitivity, specificity, accuracy, and AUC is considered the best model.

5. Experimental Design

The study compares the performance of LR, SVM, XGBoost, and ANN models in CAD prediction. The experiment is divided into five phases: data preprocessing, variable selection, data normalization and partition, hyperparameter tuning, and model comparison, as illustrated in Figure 4.
5.1 Data Preprocessing

This study uses the publicly available Z-Alizadeh Sani dataset obtained from the UCL Machine Learning Repository. The dataset includes the records of 303 random patients who visited Shaheed Rajaei Cardiovascular, Medicine, and Research Centre of Tehran, Iran. Out of these patients, 216 have been diagnosed with CAD, while the remaining 87 are normal. Each dataset entry contains information about the patient, such as age, sex, etc. The dataset comprises 56 features arranged into four groups: demographic, symptoms and examinations, ECG, and laboratory and echo features. The target variable “Cath” is binary with labels “Cad” and “Normal”. “Cad” refers to the presence of CAD, while “Normal” refers to normal patients (Alizadehsani et al., 2012, 2016; Arabasadi et al., 2017; UCL-MachineLearning-Repository, 2022).

The importance of data preprocessing cannot be overstated, as it lays the foundation for accurate modeling and analysis. In Figure 5, we showcase the initial data preprocessing phase, where out of the 56 features, 35 are character variables, and 21 are numeric variables. Some of the character variables had extremely unbalanced categories, which could significantly affect the accuracy of our analysis. For example, External CP had very few categories, and the binary variables CHF and Low TH Ang had very few Y (CAD patients) categories and a high number of N (normal patients) categories. For this reason, we decided to remove these three variables during the initial data preprocessing phase, ensuring that our analysis is based on the most accurate and reliable data possible.

5.2 Variable Selection

When dealing with a large number of variables during data preprocessing, it’s possible that some might be irrelevant. However, it’s important to ensure that only relevant variables are used to minimize the errors in the analysis. We employed two powerful methods to address this issue: the Chi-squared test for independence and lasso regression for feature selection. These methods allow us to identify and remove irrelevant variables effectively, ensuring the model’s robustness and reliability.

We conducted Chi-square test of independence, developed by Karl Pearson and Stephen M. Stigler (Franke et al., 2012; Magnello, 1998; Stigler, 2002), to examine the relationship between Cath and all 52 predictors. By calculating the test statistic, degrees of freedom (df), and p-values, we identified 17 predictors that strongly correlate with Cath. These results are presented in Table 2. By applying a cut-off p-value of 0.05, we have ensured that the findings are statistically significant and reliable.

We have also implemented the Least Absolute Shrinkage and Selection Operator (LASSO) (Ranstam and Cook, 2018) for the variable selection. we applied this method to the full data set with Cad as the response variable and discovered that for certain values of $\lambda$, the coefficient estimates for some features were zero. We performed 10-fold cross-validation to find the best value of $\lambda$ and discovered it to be 0.028, which minimizes cross-validation error. Only 16 variables had non-zero regression coefficients after applying LASSO regression on the full data set using $\lambda = 0.028$. These 16 variables, including Age, DM, HTN, FH, BP, PR, Typical.Chest.Pain, Dyspnea, Nonanginal, Tinversion, TG, EF.TTE, and Region.RWMA, were selected as predictors for our analysis.

It is worth noting that the Chi-Squared and LASSO feature selection criteria use a different number of predictors, which consist of both numerical and categorical variables. However, when it comes to training ANN and SVM models, numerical features are required. The ordinal encoding technique is implemented to transform the categorical variables into numerical values, enabling them to be used in the model training process. (Choong and Lee, 2017; Jackson and Agrawal, 2019).
Table 2. Chi-square test of independence: Cath vs predictors. The predictors with p-value smaller than 0.05 are shown in the table.

<table>
<thead>
<tr>
<th>Variables</th>
<th>Chi-squared value</th>
<th>df</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Typical.Chest.Pain</td>
<td>86.936</td>
<td>1</td>
<td>≈ 0.0000</td>
</tr>
<tr>
<td>Lymph</td>
<td>67.932</td>
<td>49</td>
<td>0.0379</td>
</tr>
<tr>
<td>Age</td>
<td>66.38</td>
<td>45</td>
<td>0.0207</td>
</tr>
<tr>
<td>Atypical</td>
<td>50.442</td>
<td>1</td>
<td>≈ 0.0000</td>
</tr>
<tr>
<td>EF.TTE</td>
<td>43.100</td>
<td>10</td>
<td>≈ 0.0000</td>
</tr>
<tr>
<td>Region.RWMA</td>
<td>34.901</td>
<td>4</td>
<td>≈ 0.0000</td>
</tr>
<tr>
<td>BP</td>
<td>28.497</td>
<td>16</td>
<td>0.02755</td>
</tr>
<tr>
<td>HTN</td>
<td>23.813</td>
<td>1</td>
<td>≈ 0.0000</td>
</tr>
<tr>
<td>Nonanginal</td>
<td>20.149</td>
<td>1</td>
<td>≈ 0.0000</td>
</tr>
<tr>
<td>DM</td>
<td>18.175</td>
<td>1</td>
<td>≈ 0.0000</td>
</tr>
<tr>
<td>VHD</td>
<td>16.198</td>
<td>3</td>
<td>0.0010</td>
</tr>
<tr>
<td>T.inversion</td>
<td>15.883</td>
<td>1</td>
<td>≈ 0.0000</td>
</tr>
<tr>
<td>St.Depression</td>
<td>5.589</td>
<td>1</td>
<td>0.0181</td>
</tr>
<tr>
<td>Q.Wave</td>
<td>5.4033</td>
<td>1</td>
<td>0.0201</td>
</tr>
<tr>
<td>Diastolic.Murmur</td>
<td>4.7566</td>
<td>1</td>
<td>0.0292</td>
</tr>
<tr>
<td>St.Elevation</td>
<td>4.5328</td>
<td>1</td>
<td>0.0333</td>
</tr>
<tr>
<td>Dyspnea</td>
<td>4.2095</td>
<td>1</td>
<td>0.0402</td>
</tr>
</tbody>
</table>

5.3 Data Normalization and Partition

The range of values for input features can vary significantly, negatively impacting our results’ accuracy. Thus, we have implemented a min-max normalization technique that scales each feature to the [0,1] range. The mathematical formulation of the normalization procedure is defined as:

\[
    z = \frac{x - x_{\min}}{x_{\max} - x_{\min}}
\]  

(7)

where \( x \) is the original value, and \( z \) is the normalized value of the input variable. Similarly, \( x_{\min} \) and \( x_{\max} \) are the minimum and the maximum values, respectively. The normalized data is split into training and test sets in the 4:1. Within the training set, 25% of the data is separated for validation, which corresponds to 20% of the total data.

5.4 Hyperparameter Tuning

The standard LR model requires no hyperparameters, but the ten-fold cross-validation method is used for the SVM model to determine the best hyperparameters. The optimal values for cost and gamma are 100 and 0.01, respectively, for either case. The sigmoid kernel is employed for binary target variables. In case of XGBoost model, the trial and check method is used to identify the best hyperparameters. The hyperparameters that yielded the best results in either case are gamma=0, learning rate=0.3, n_estimators=100, max_depth=6, and min_child_weight=1. The ANN model employed a dataset divided into three parts: a training set, a validation set, and a test set, respectively, in the ratio 60:20:20. The hyperparameters included the number of hidden layers, the number of neurons on each hidden layer, batch sizes, learning rates, optimizers, and the number of epochs. Due to the size of the data, a single hidden layer was used. The remaining hyperparameters were selected using a trial and check approach on both data sets. The epochs were set to 500, and an early stopping criterion was applied to stop the model training if there was no improvement in the validation loss. The best hyperparameters included a number of neurons in the hidden layer of 50, optimizer as Adam, learning rate as 0.01, and batch size as 4. Furthermore, a 20% dropout was implemented between the hidden and fully connected output layers to prevent overfitting. The binary cross-entropy was used as the loss function during the training process. During the training of the ANN, the loss function behaved on the number of epochs, as illustrated in Figure 6. The model overfits after about 25 epochs on Chi-Squared data, whereas it overfits after 20 epochs on Lasso data.
5.5 Model Comparison

The model was tested on the test data set with the optimized hyperparameters in place. The performance scores achieved by the model are presented in Table 3 and Table 4, demonstrating the approach’s effectiveness.

Table 3. Model performance matrices obtained using test Chi-Squared data

<table>
<thead>
<tr>
<th>Models</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Accuracy</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR</td>
<td>0.9592</td>
<td>0.9167</td>
<td>0.9508</td>
<td>0.9847</td>
</tr>
<tr>
<td>SVM</td>
<td>0.9592</td>
<td>0.9167</td>
<td>0.9508</td>
<td>0.9813</td>
</tr>
<tr>
<td>XGBoost</td>
<td><strong>0.9778</strong></td>
<td>0.8125</td>
<td>0.9344</td>
<td>0.9278</td>
</tr>
<tr>
<td>ANN</td>
<td>0.9268</td>
<td>0.8500</td>
<td>0.9016</td>
<td>0.9451</td>
</tr>
</tbody>
</table>

The performance metrics from the four models on the test Chi-Squared data are impressive, with all four models achieving a minimum of 0.9016 for sensitivity, accuracy, and AUC. The XGBoost model had a lower specificity of 0.8125 than the SVM and LR models. However, the LR model outperformed the others with the highest specificity of 0.9167, accuracy of 0.9508, and AUC of 0.9847. The XGBoost model had the highest sensitivity of 0.9778, while the LR model had the second highest sensitivity of 0.9592. Upon careful analysis of the metrics, the LR model is undoubtedly considered the best model with a sensitivity of 0.9592, specificity of 0.9167, accuracy of 0.9508, and AUC of 0.9847, followed by the SVM model with a sensitivity of 0.9592, specificity of 0.9167, accuracy of 0.9508, and AUC of 0.9813. The summary of the performance matrices is highlighted in Table 3.

Table 4. Model performance matrices obtained using test Lasso data

<table>
<thead>
<tr>
<th>Models</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Accuracy</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR</td>
<td><strong>0.9796</strong></td>
<td><strong>0.9167</strong></td>
<td><strong>0.9672</strong></td>
<td><strong>0.9864</strong></td>
</tr>
<tr>
<td>SVM</td>
<td><strong>0.9796</strong></td>
<td><strong>0.9167</strong></td>
<td><strong>0.9672</strong></td>
<td>0.9830</td>
</tr>
<tr>
<td>XGBoost</td>
<td>0.9333</td>
<td>0.6250</td>
<td>0.8525</td>
<td>0.8861</td>
</tr>
<tr>
<td>ANN</td>
<td>0.9024</td>
<td>0.8500</td>
<td>0.8852</td>
<td>0.9451</td>
</tr>
</tbody>
</table>

Table 4 presents a clear summary of the performance metrics from the four models on the Lasso data. All models achieved a sensitivity of more than 0.90, which is a reasonable result. Based on the best performance metrics highlighted in the table, it is evident that all four models’ performances are competitive. However, LR emerged as the clear winner with the highest sensitivity of 0.9796, specificity of 0.9167, accuracy of 0.9672, and AUC of 0.9864. SVM came in a close second with a sensitivity of 0.9796, specificity of 0.9167, accuracy of 0.9672, and AUC of 0.9830. Therefore, we confidently say that LR is the best model for the given Lasso dataset.
The comparison of the performances of different machine learning algorithms using ROC curves is shown in Figure 7. The ROC plot provides an excellent visualization of the trade-off between sensitivity and specificity, and based on the plot, it is evident that there is no significant difference between the AUC of all models within the two datasets, except for the XGBoost model. The XGBoost model has an AUC of 0.9278 on Chi-Squared data and of 0.8861 on Lasso data.

6. Conclusion

CAD is a leading cause of death globally. Detecting it early can significantly improve patient outcomes by enabling timely interventions and preventive measures. Modeling techniques can help to identify risk factors and patterns associated with CAD, allowing healthcare professionals to intervene before the disease progresses to a more critical stage. This study demonstrates the comparative analysis of the effectiveness of logistic regression, support vector machine, extreme grading boosting, and artificial neural network models in detecting coronary artery disease (CAD) using the Z-Alizadeh Sani dataset. Most significant variables from the dataset are selected based on Chi-squared and lasso criteria. The performance of the model is evaluated using sensitivity, specificity, accuracy, and the area under the Receiver Operating Characteristic (ROC) curve (AUC). The experimental results show that the logistic regression model is the most effective in predicting with a sensitivity of 0.9592, specificity of 0.9167, accuracy of 0.9508, and AUC of 0.9847 followed by the SVM model with a sensitivity of 0.9592, specificity of 0.9167, the accuracy of 0.9508, and AUC of 0.9813. The results also demonstrate that increasing the dataset size may improve the performance of the ANN model.

In the near future, we want to further investigate the role of lifestyle factors, such as diet and exercise, in the development and progression of CAD. Another potential direction is utilizing novel imaging modalities and techniques, such as coronary computed tomography angiography (CCTA), cardiac magnetic resonance imaging (MRI), and intravascular imaging by employing deep learning techniques. We also plan to develop personalized risk assessment models that integrate genetic, clinical, lifestyle, and environmental factors to identify high-risk individuals for targeted interventions.

References


Poirier, P. (2008). Healthy lifestyle: even if you are doing everything right, extra weight carries an excess risk of acute coronary events.


**Acknowledgments**

Not applicable

**Authors contributions**


**Funding**

This research received no external funding.
Competition of interests
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Informed consent
Obtained.

Ethics approval
The Publication Ethics Committee of the Canadian Center of Science and Education. The journals policies adhere to the Core Practices established by the Committee on Publication Ethics (COPE).

Provenance and peer review
Not commissioned; externally double-blind peer reviewed.

Data availability statement
The data are publicly available on the UCL Machine Learning Repository.

Data sharing statement
No additional data are available.

Open access
This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/4.0/).

Copyrights
Copyright for this article is retained by the author(s), with first publication rights granted to the journal.