# Analysis on Coronary Artery Disease using Machine Learning

Isabel Pham
Department of Computer Science
San Jose State University
San Jose, USA
isabel.pham@sjsu.edu

Nahal Bagheri
Department of Computer Science
San Jose State Univeristy
San Jose, USA
nahal.bagheri@sjsu.edu

***Abstract*—Coronary artery disease is a common heart disease that is not only the leading cause of death in the United States but is also a progressive disease with no cure for it. While there is no defined cure for this disease, providing patients with treatment could help alleviate symptoms and slow the progression of the disease. This study aims to use machine learning techniques to create a diagnosis or predictive model that can predict whether a patient would have coronary artery disease so that the patient could apply treatment as soon as possible. This paper tested six different models on different sampling versions of data and discovered that the Random Forest algorithm performed the best among the other models in predicting the disease with accuracy and recall values.**

***Keywords—coronary artery disease, machine learning, controllable risk factor, uncontrollable risk factor***

## I. Introduction

Coronary Artery Disease (CAD) is one of the leading causes of death in the United States, accounting for approximately 610,000 deaths annually [1]. It is a heart disease caused by an inadequate supply of oxygen-rich blood to the heart. This is typically caused by plaque buildup of substances in the walls of the arteries. This condition could cause noticeable symptoms such as chest pain. However, it is entirely possible for a CAD patient to show no symptoms at all. Regardless of whether symptoms show, these patients are at risk of having a heart attack [2]. Currently, CAD is incurable and considered to be a progressive disease, but there are some treatments available to weaken the symptoms and the disease's progression. Treatments for CAD are not limited to just medication. It may include surgeries or lifestyle changes as well. It should be noted that these treatments will still require a fair amount of time and effort to show any effect [3]. Thus, there is a need for urgency to provide treatment to those who show signs of having CAD as soon as possible. This would lead to our study focusing on the detection of early-stage CAD.

To determine if someone has CAD, it is significant to consider the risk factors that correlate to having the disease. Risk factors can be divided into two main categories: controlled and uncontrolled. Controllable risk factors include modifiable aspects like a person's cholesterol levels, whether the person smoked or not, whether they are obese or not, and so forth. Some examples of uncontrollable risk factors for someone who might have CAD are their age, gender, family history, and race [1].

With many factors to consider and symptoms being an unreliable indicator, it may be difficult to predict if a person has CAD or not. Fortunately, machine learning (ML) is a popular tool that can be used to make a medical-based disease diagnosis model (MLBDD) in a time-efficient and inexpensive way to take and analyze data patterns for disease-related prediction [4]. For this study, we narrowed our scope to a binary classification of CAD. We will be utilizing machine learning to create a supervised medical-based predictive model that will classify or predict whether a patient has coronary artery disease based on a collection of past patients' data. We will be testing six different machine-learning models and using the best model to inform us what are the aspects or risk factors that are prevalent in having CAD.

## II. Related Works

In the exploration of machine learning applications for predicting coronary artery disease (CAD), several studies have leveraged datasets from diverse geographical locations, each encompassing health-related patient data with a spectrum of controllable and uncontrollable risk factors. For instance, Muhammad et al. [5] utilized a dataset primarily from Nigeria, providing a unique insight into the regional characteristics of CAD. On the other hand, Abdar et al. [6] leveraged a preprocessed heart disease dataset, focusing on advanced feature selection techniques using Genetic Algorithms (GA) and Particle Swarm Optimization (PSO), and introduced the innovative N2Genetic optimizer as part of their novel approach. This underscores the evolving nature of data mining techniques in healthcare.

A. Akella and S. Akella [7] used the 'Cleveland Dataset,' a comprehensive collection of patient data pivotal in heart disease research, to pursue their study on CAD. They implemented various machine-learning algorithms, achieving significant accuracy, especially with neural networks, underscoring the potential of machine learning in enhancing predictive accuracy in medical diagnostics. Wang et al. [8] combined datasets from three medical centers in China, illustrating the scalability of ML models across diverse data sources.

The first step unanimously taken by these studies was to perform data preprocessing. Data preprocessing is an essential phase in machine learning that includes manipulating, transforming, and converting a raw dataset into a clean, reliable, and usable dataset for the machine learning model to train from. Depending on how the preprocessing step was done, it can significantly impact the machine learning model performance. The approaches taken to preprocess data varied

among the studies. This reflects the diversity among the studies' datasets and the specific goals the authors wanted to achieve. While Muhammad et al. [5] and A. Akella and S. Akella [7] removed rows with missing values, the latter had also contemplated the use of the Synthetic Minority Over-sampling Technique (SMOTE) for addressing the data imbalance in their data. This is often a common challenge for medical datasets. However, they chose not to due to their concerns that it would not accurately represent their dataset despite the data set size being relatively small for their studies. This decision reflects the trade-offs in machine learning between presenting accurate data representation and model accuracy. Wang et al. [8] employed the least absolute shrinkage and selection operator (LASSO) technique for feature selection, a method known for its efficacy in enhancing model performance by reducing overfitting using both variable selection and regularization.

Training diverse machine-learning models with various algorithms was the next shared step among the studies. Algorithms like Random Forest, SVM variants, and logistic regression were commonly used among the studies. This aspect demonstrates the adaptability of machine learning techniques still being applicable to use on different types of medical data. The usage of other algorithms like Naive Bayes, KNN, and Artificial Neural Networks further emphasizes the exploratory nature of these studies in finding what would be the most effective model for CAD prediction in their dataset.

The evaluation and comparison of models across these studies were done using metrics like accuracy, specificity, sensitivity, and ROC. These varied metrics highlight the complexity of model assessment in healthcare. Different aspects of a model's performance such as its ability to correctly identify cases (sensitivity) and its overall accuracy are all crucial to consider. Muhammad et al. [5] found that heart rate was a significant feature in determining if a patient has CAD through their usage of a Random Forest model. Another significant feature was discovered by Wang et al. [8] as they identified HDL-C levels as the key factor in CAD risk. Findings like the identification of significant features of patients having CAD not only demonstrate the practical implications of machine learning in medical diagnostics but also illustrates the capability of machine learning models to uncover new insights in medical data.

Overall, these studies collectively highlight the transformative potential of machine learning in healthcare, particularly in predicting CAD. They showcase the advancements in algorithmic techniques, the importance of comprehensive data preprocessing, and the critical role of having nuanced model evaluation metrics. The synthesis of these elements within machine learning research plays an influential part in developing more accurate, efficient, and effective diagnostic tools, ultimately contributing to better patient outcomes and healthcare delivery.

## III. Methodology

The objective of this study is to create a predictive model that determines whether a person has CAD or not based on past patients' data. After surveying the literature reviews, the overall steps we will take to complete our objective are to obtain our dataset, investigate the data, perform data preprocessing, split the data into train and test sets, train different machine learning models, and evaluate and compare the models' results with each other. The overall process we took can be summarized and illustrated as a flow diagram as shown in Figure 1. In the following subsections, we will go into more detail about the process and reasoning behind our course of actions.

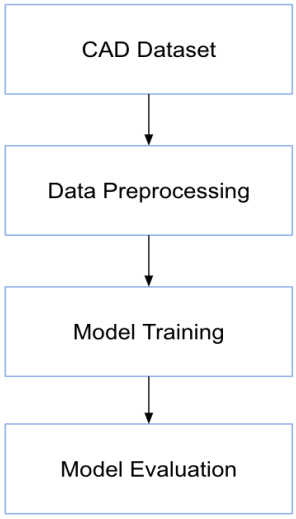



Fig. 1. Overall Method Flow

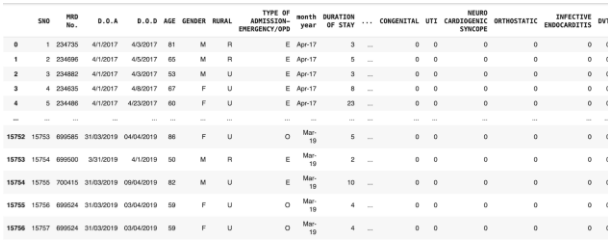
### A. *Dataset*

The dataset utilized for our predictive model is a Kaggle dataset called "Hospital Admission Data" [9, 10]. It is a comprehensive collection of patient data spanning two years starting from 2017 to 2019 at the Hero DMC Heart Institute, Unit of Dayanand Medical College and Hospital in Ludhiana, India.

Before performing any data preprocessing, we need to first observe the original data set to understand its content to then consider what preprocessing steps should be taken. Using Google Colab and the Pandas library, we can read the dataset CSV file as a Pandas data frame as shown in Figure 2. The dataset consists of about 15,575 rows and 56 columns, where each row represents a patient, and the columns are the health attributes or features of the patient.

To understand what kind of information the columns represent, we can utilize the CSV file that was packaged with the original dataset as a reference. Table 1 displays the columns' headings and the explanatory name for each of them. Based on Table 1, we can identify the dataset does include hospital administrative data such as the serial number, admission number, how long the patient stayed at the hospital, the date of their admission and discharge, type of the patient's admission type, and so forth.

The dataset has a general mix of both controllable and uncontrollable risk factors. It also has a general mix of discrete and continuous data, or in other words, categorical and numerical data. Upon checking the dataset's columns type in the snippet shown in Figure 3, there are noticeable incorrect data types for some of the columns including HB, TLC, PLATELETS, GLUCOSE, UREA, CREATININE, BNP, and



| | SNO | MRD No. | D.O.A | D.O.D | AGE | GENDER | RURAL | TYPE OF ADMISSION-EMERGENCY/OPD | month year | DURATION OF STAY | ... | CONGENITAL | UTI | NEURO CARDIOGENIC SYNCOPE | ORTHOSTATIC | INFECTIVE ENDOCARDITIS | DVT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 234735 | 4/1/2017 | 4/3/2017 | 81 | M | R | E | Apr-17 | 3 | ... | 0 | 0 | 0 | 0 | 0 | |
| 1 | 2 | 234696 | 4/1/2017 | 4/5/2017 | 65 | M | R | E | Apr-17 | 5 | ... | 0 | 0 | 0 | 0 | 0 | |
| 2 | 3 | 234882 | 4/1/2017 | 4/3/2017 | 53 | M | U | E | Apr-17 | 3 | ... | 0 | 0 | 0 | 0 | 0 | |
| 3 | 4 | 234635 | 4/1/2017 | 4/8/2017 | 67 | F | U | E | Apr-17 | 8 | ... | 0 | 0 | 0 | 0 | 0 | |
| 4 | 5 | 234486 | 4/1/2017 | 4/23/2017 | 60 | F | U | E | Apr-17 | 23 | ... | 0 | 0 | 0 | 0 | 0 | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 15752 | 15753 | 699585 | 31/03/2019 | 04/04/2019 | 86 | F | U | O | Mar-19 | 5 | ... | 0 | 0 | 0 | 0 | 0 | |
| 15753 | 15754 | 699500 | 3/31/2019 | 4/1/2019 | 50 | M | R | E | Mar-19 | 2 | ... | 0 | 0 | 0 | 0 | 0 | |
| 15754 | 15755 | 700415 | 31/03/2019 | 09/04/2019 | 82 | M | U | E | Mar-19 | 10 | ... | 0 | 0 | 0 | 0 | 0 | |
| 15755 | 15756 | 699524 | 31/03/2019 | 03/04/2019 | 59 | F | U | O | Mar-19 | 4 | ... | 0 | 0 | 0 | 0 | 0 | |
| 15756 | 15757 | 699524 | 31/03/2019 | 03/04/2019 | 59 | F | U | O | Mar-19 | 4 | ... | 0 | 0 | 0 | 0 | 0 | |

Fig 2. Dataset before any preprocessing

EF. These features or attributes were marked as object types when they should be continuous or numerical data types like float. Figure 4 showcases all the columns that have missing values or data that were marked as null.

While observing numerical data on the table may provide helpful information and statistics, visualizations could also provide more helpful insight and statistics that numbers may not cover or showcase. For instance, Figure 5 displays a collection of bar graphs representing the number of patients with CAD versus patients without CAD for each categorical feature. From observing this figure, we can identify that there is more male who has CAD in the dataset compared to males who do not. The amount of female who has CAD compared to those who do not is also more like males but does not have a drastic difference like male. The bar graph also depicts that dataset consists of more people who live in urban area compared to people who live in rural area. Considering how the data for this dataset was extracted from a hospital, which is in an urban area, it would make sense why there are more people with urban lifestyle than rural in the dataset. People who live in urban areas would have easier access to hospitals compared to those who live in rural areas.

Figure 6 is a heatmap that shows the relationship between the features or columns. Unfortunately, this heatmap only includes features that have a numerical type. Features like HB and GENDER would not be accounted for because they are of type "object." Another issue with this heatmap is that it includes hospital administrative data. These should not be in the heatmap because they do not count toward a patient's health status. An insight we can gain from this heatmap is how these features relate to CAD. From the dataset without any preprocessing, there seems to be a slight positive correlation between a patient having hypertension and CAD. After hypertension, ACS and age seem to have a positive correlation with CAD.

Figure 7 provides a distribution idea of how many patients have CAD versus how many do not in a pie chart visual. From this visual, we can see that the dataset is slightly unbalanced. About 33% of the patients do not have CAD whereas 67% of the patients do have CAD. This would mean we would need to balance the data to avoid having a skewed model.

## *B. Data Preprocessing*

The purpose of data preprocessing is to make the data ready for analysis and machine learning. This process could include tasks like cleaning, manipulating, and transforming the data. We mainly used Python and some libraries to do this preprocessing process. Based on what was examined about the dataset in the previous section, the issues that were present in the original dataset are the following: the dataset being unbalanced, dataset having columns and or features not related to patient's health, missing values for some features, and incorrect types for a subset of features.

The first step of our preprocessing phase was to remove all the columns that do not relate to or influence the patient's health status of getting CAD. This would include removing all the hospital administrative data such as the admission number, serial number, date of admission, date of discharge, column

Table 1. Dataset Table Headings

table_headings

| Table Heading | Explanatory Name |
|---|---|
| **SNO** | Serial Number |
| **MRD No.** | Admission Number |
| **D.O.A** | Date of Admission |
| **D.O.D** | Date of Discharge |
| **AGE** | AGE |
| **GENDER** | GENDER |
| **RURAL** | RURAL(R) /Urban(U) |
| **TYPE OF ADMISSION-EMERGENCY/OPD** | TYPE OF ADMISSION-EMERGENCY/OPD |
| **month year** | month year |
| **DURATION OF STAY** | DURATION OF STAY |
| **duration of intensive unit stay** | duration of intensive unit stay |
| **OUTCOME** | OUTCOME |
| **SMOKING** | SMOKING |
| **ALCOHOL** | ALCOHOL |
| **DM** | Diabetes Mellitus |
| **HTN** | Hypertension |
| **CAD** | Coronary Artery Disease |
| **PRIOR CMP** | CARDIOMYOPATHY |
| **CKD** | CHRONIC KIDNEY DISEASE |
| **HB** | Haemoglobin |
| **TLC** | TOTAL LEUKOCYTES COUNT |
| **PLATELETS** | PLATELETS |
| **GLUCOSE** | GLUCOSE |
| **UREA** | UREA |
| **CREATININE** | CREATININE |
| **BNP** | B-TYPE NATRIURETIC PEPTIDE |
| **RAISED CARDIAC ENZYMES** | RAISED CARDIAC ENZYMES |
| **EF** | Ejection Fraction |
| **SEVERE ANAEMIA** | SEVERE ANAEMIA |
| **ANAEMIA** | ANAEMIA |
| **STABLE ANGINA** | STABLE ANGINA |
| **ACS** | Acute coronary Syndrome |
| **STEMI** | ST ELEVATION MYOCARDIAL INFARCTION |
| **ATYPICAL CHEST PAIN** | ATYPICAL CHEST PAIN |
| **HEART FAILURE** | HEART FAILURE |
| **HFREF** | HEART FAILURE WITH REDUCED EJECTION FRACTION |
| **HFNEF** | HEART FAILURE WITH NORMAL EJECTION FRACTION |
| **VALVULAR** | Valvular Heart Disease |
| **CHB** | Complete Heart Block |
| **SSS** | Sick sinus syndrome |
| **AKI** | ACUTE KIDNEY INJURY |
| **CVA INFRACT** | Cerebrovascular Accident INFRACT |
| **CVA BLEED** | Cerebrovascular Accident BLEED |
| **AF** | Atrial Fibrilation |
| **VT** | Ventricular Tachycardia |
| **PSVT** | PAROXYSMAL SUPRA VENTRICULAR TACHYCARDIA |
| **CONGENITAL** | Congenital Heart Disease |
| **UTI** | Urinary tract infection |
| **NEURO CARDIOGENIC SYNCOPE** | NEURO CARDIOGENIC SYNCOPE |
| **ORTHOSTATIC** | ORTHOSTATIC |
| **INFECTIVE ENDOCARDITIS** | INFECTIVE ENDOCARDITIS |
| **DVT** | Deep venous thrombosis |
| **CARDIOGENIC SHOCK** | CARDIOGENIC SHOCK |
| **SHOCK** | SHOCK |
| **PULMONARY EMBOLISM** | PULMONARY EMBOLISM |
| **CHEST INFECTION** | CHEST INFECTION |
| | |
| **Other Abbreveiations** | |
| **DAMA** | Discharged Against Medical Advice |

```
Data columns (total 56 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   SNO                                   15757 non-null  int64
 1   MRD No.                               15757 non-null  object
 2   D.O.A                                 15757 non-null  object
 3   D.O.D                                 15757 non-null  object
 4   AGE                                   15757 non-null  int64
 5   GENDER                                15757 non-null  object
 6   RURAL                                 15757 non-null  object
 7   TYPE OF ADMISSION-EMERGENCY/OPD       15757 non-null  object
 8   month year                            15757 non-null  object
 9   DURATION OF STAY                      15757 non-null  int64
 10  duration of intensive unit stay       15757 non-null  int64
 11  OUTCOME                               15757 non-null  object
 12  SMOKING                               15757 non-null  int64
 13  ALCOHOL                               15757 non-null  int64
 14  DM                                    15757 non-null  int64
 15  HTN                                   15757 non-null  int64
 16  CAD                                   15757 non-null  int64
 17  PRIOR CMP                             15757 non-null  int64
 18  CKD                                   15757 non-null  int64
 19  HB                                    15505 non-null  object
 20  TLC                                   15471 non-null  object
 21  PLATELETS                             15472 non-null  object
 22  GLUCOSE                               14894 non-null  object
 23  UREA                                  15516 non-null  object
 24  CREATININE                            15510 non-null  object
 25  BNP                                   7316 non-null   object
 26  RAISED CARDIAC ENZYMES                15757 non-null  int64
 27  EF                                    14252 non-null  object
 28  SEVERE ANAEMIA                        15757 non-null  int64
```

Fig. 3. Snippet List of Features and Dtypes

"month year," outcome, and all columns that contain how long the patient stayed at the hospital.

The column with the date of birth was also removed because while it was related to the patient's age, the date was not a good candidate as an attribute for comparison. Keeping track of date of birth of patients would be useful to compare the months and days, but we assume that it would not provide a significant enough difference as compared to the differences in years, which is can be accomplished by using age. In addition, keeping track of dates would take up more space and computing resources while introducing unnecessary complexity in our comparisons.

Columns "Heart Failure," HFREF" and HFRNEF were removed because it would not make sense to heart failure as a feature to determine whether someone has CAD or not. Having CAD would mean there is more risk of having heart failure, but not vice versa. In addition to that, if a patient already has heart failure, it would be already too late to treat that patient.

The second step of preprocessing would be to fix the types of some of the features. Columns like HB and BNP are supposed to be continuous or numerical values. However, in

```
HB                              252
TLC                             286
PLATELETS                       285
GLUCOSE                         863
UREA                            241
CREATININE                      247
BNP                            8441
RAISED CARDIAC ENZYMES            0
EF                             1505
```

Fig. 4. Features with missing values


Fig. 5. Bar Graphs of Categorical Features vs CAD count

the original dataset, the reason why they were marked as type "object" instead of numerical was due to the column having invalid data. For some values with missing data, instead of a NaN or null value, a string "EMPTY" was placed instead, converting the type from numeric to object. To resolve this issue, we would convert any invalid data to NaN before promptly removing all rows with null values. Figures 8 and 9 show the distribution of all numerical or continuous features after fixing the types. Note that the scale among each of these features is quite vastly different, so this would require the values to be converted to the same scale.

Figure 10 shows the heat map of the features after the second preprocessing step. It now includes the features that were marked as objects before. However, this heat map does not include gender or the rural yet. We will need to convert them into their binary categorization representation using OneHotEncoder.

The last three preprocessing steps are applying OneHotEncoder to categorical columns, applying normalization to scale the data, and balancing the dataset. However, it seems that it is recommended to apply these steps after splitting the data into train and test sets to avoid data leakage and to keep the test set "unseen." Thus, the order in which we complete the process is as follows: balancing the data, applying OneHotEncoder, and then applying normalization.

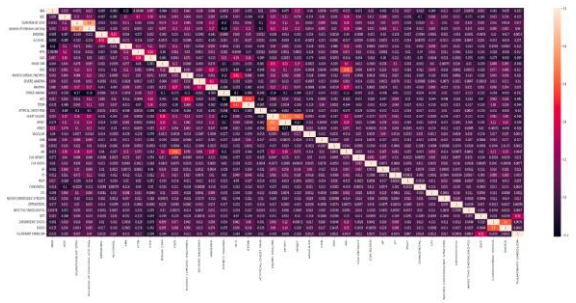


Fig. 6. Heat map before preprocessing

There are two ways to balance the dataset: undersample and oversample. We chose to have both the undersampled and oversampled versions of the processed dataset. We kept the undersampled version to keep true to the original dataset. However, because we have an insufficient amount of data, we could lose data if we did undersampling. With the risk of data loss when using the undersample method, we decided to also consider having an oversampled version. To generate these balanced datasets, we use random sampling for the undersample version and SMOTE to generate synthetic data for the oversampled version.

For each sampled version, we apply OneHotEncoder on the two categorical features: GENDER and RURAL. This would result in making the categorical columns represented as binary categorization (i.e. value would be 1 if the feature is present and 0 when it is not). GENDER would be split into female and male while RURAL would be rural and urban. This step was so that machine learning models would understand the data more readily. After applying OneHotEncoder, we normalized each version. As a result of preprocessing, we would have a total of 45 features to consider.

### *C. Model Training*

In this section, we delve into the models selected for the CAD prediction model, informed by our literature review. The chosen models – Random Forest, Naive Bayes, Support Vector Machine, Logistic Regression, Decision Trees, and K-nearest neighbors – are all supervised machine-learning algorithms, suitable for our dataset which includes a target label for training.

- Logistic Regression

  This statistical model calculates the probability of a specific event based on independent variables. It's particularly useful for classification and predictive analysis due to its ability to handle binary outcomes and provide probabilities for each class[11].

- Decision Tree

  A Decision Tree algorithm creates a tree-like structure of decisions, derived from historical data, to predict the class or value of the target variable. It's beneficial for understanding the impact of various decisions, as it breaks down the decision-making process into a series of straightforward choices.

- Random Forest

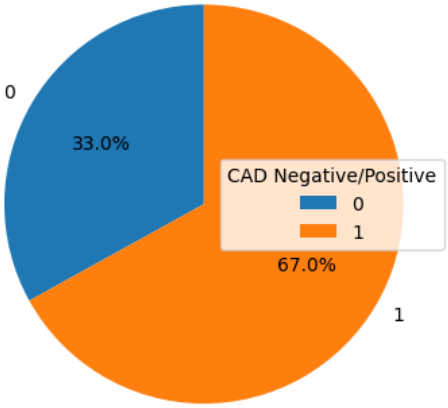



Fig. 7 Pie chart of CAD versus no CAD

  An ensemble method, Random Forest combines multiple decision trees to make more accurate predictions than a single tree could. By aggregating the predictions from numerous trees, it reduces the risk of overfitting and improves overall model accuracy.

- K-nearest neighbors

  KNN operates on the principle that similar things are near to each other. It classifies a data point based on how its neighbors are classified, making it a non-parametric and instance-based learning method. This algorithm is particularly effective when there's a significant amount of data.

- Support Vector Machines

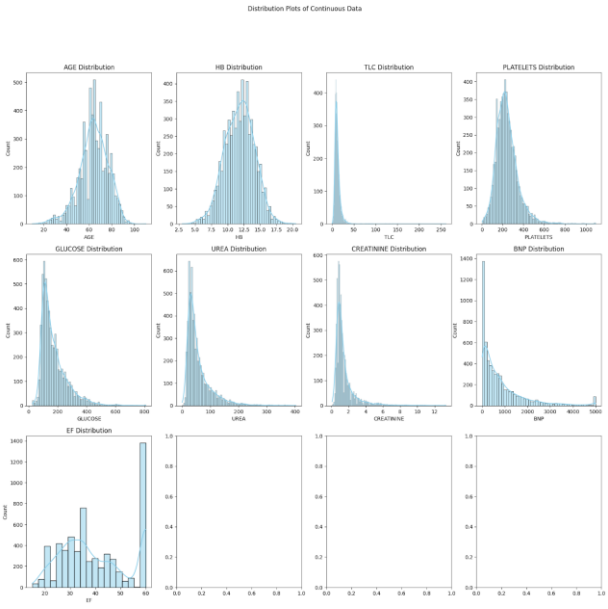



Fig. 8 Oversample Accuracies Among Models

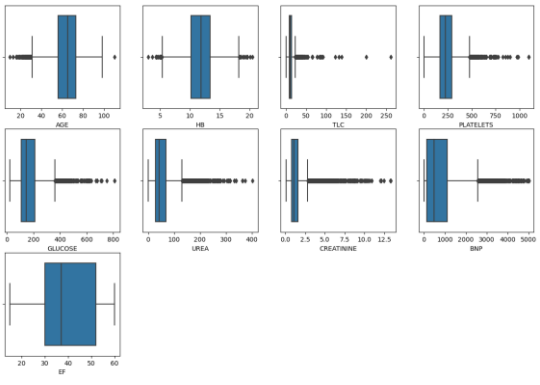



Fig. 9 Box plots of continuous values

SVM is a robust algorithm used for both classification and regression tasks. It works by finding the hyperplane that best divides a dataset into classes, which is particularly useful in high-dimensional spaces.

- Navies Bayes

  Based on Bayes' theorem, this algorithm assumes independence between predictors and is highly scalable. It's known for its simplicity and efficiency in large datasets, making it well-suited for classification tasks involving a large number of features.

For model training and evaluation, both undersampled and oversampled versions of the dataset will be used. The Sci-kit and Pandas libraries are instrumental in this process. To fine-tune the models, GridsearchCV was used. This is a robust method for determining the optimal hyperparameters. Tables 2 to 7 in the paper detail the specific hyperparameters selected for each model. It's noteworthy that some models did not adhere strictly to the hyperparameters suggested by GridSearchCV from the Sci-kit library, indicating a customized approach to model optimization.

.

### *D. Model Evaluation*

While accuracy is the typical evaluation metric for model training, it is also important to consider other metrics such as recall, precision, and F1 score into account as well.

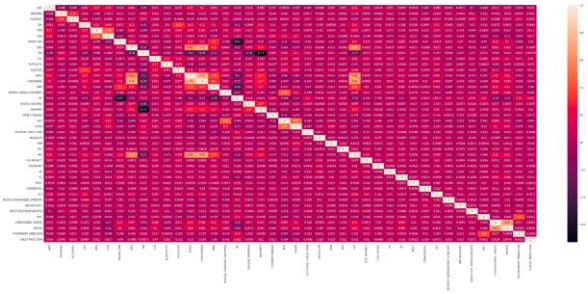


Fig. 10 Heat map after preprocessing

Table 2 Hyperparamters for Random Forest

| Random Forest | n_estimators | max_depth | min_samples_split | min_samples_leaf | max_features |
|---|---|---|---|---|---|
| Undersample | 100 | None | 2 | 1 | sqrt |
| Oversample | 100 | None | 2 | 1 | sqrt |

Table 3 Hyperparamters for SVM

| SVM | C | Kernel | Gamma |
|---|---|---|---|
| Undersample | 0.1 | linear | 0.1 |
| Oversample | 0.1 | linear | 0.1 |

Table 4 Hyperparamters for Naïve Bayes

| Naive Bayes | Var_smoothingfloat |
|---|---|
| Undersample | 1e-9 |
| Oversample | 1e-9 |

Table 5 Hyperparamters for Decision Tree

| Decision Trees | max_depth | min_sample_split | min_sample_leaf | max_feature |
|---|---|---|---|---|
| Undersample | 10 | 10 | 2 | None |
| Oversample | 20 | 2 | 1 | None |

Table 6 Hyperparamters for Logisitic Regression

| Logistic Regression | C | splver | Penalty |
|---|---|---|---|
| Undersample | 100 | liblinear | l1 |
| Oversample | 100 | liblinear | l1 |

Table 7 Hyperparamters for KNN

| K-nearest neighbors | n_neighbors | weights | metric |
|---|---|---|---|
| Undersample | 15 | distance | Manhattan |
| Oversample | 9 | distance | Manhattan |

- *Recall*

  Recall is a metric that measures the proportion of actual positives (people with CAD) that are correctly identified by the model.

- *Precision*

  Precision is another metric that measures the proportion of positive classification results that were actually correct.

- *F1 score*

F1 Score is a metric that serves as a harmonic mean of precision and recall. It is particularly useful when you need to find a balance between precision and recall.

In the context of a CAD prediction or diagnosis model, the amount of people the model correctly classifies with CAD is called true positive while the amount of people that do have CAD but were classified as having no CAD is called false negative.

A high recall is often more significant than high accuracy because the cost of a false negative (failing to diagnose a patient with CAD) can be life-threatening. It is vital to have and create models that prioritize minimizing the number of false negatives from occurring, even if it means tolerating more false positives (wrongly diagnosing CAD when it is not present).

Having a high precision in the context of CAD could be as important as high recall to avoid unnecessary anxiety or treatment. However, for our study, we put more emphasis on recall.

For each model, we plan to record the accuracies, precision, recall, and F1 score. We will also be generating a confusion matrix to provide a visualization of the model's performance and accuracies in terms of identifying the number of true positives, false positives, true negatives, and false negatives.

In conclusion, evaluating a CAD prediction model requires a multifaceted approach. Recall is significant in reducing the risk of undiagnosed cases, but it must be considered alongside the precision, accuracy, and all the other trade-offs inherent in dealing with imbalanced datasets. The choice of model and the approach to data sampling play a crucial role in determining the effectiveness of the diagnostic tool. Having a continuous evaluation using metrics like the confusion matrix is essential for refining the model and ensuring its reliability in real-world scenarios.

## IV. Experimental Evaluation

The results of the accuracies for both undersample and oversample versions of the dataset can be seen in Figure 11

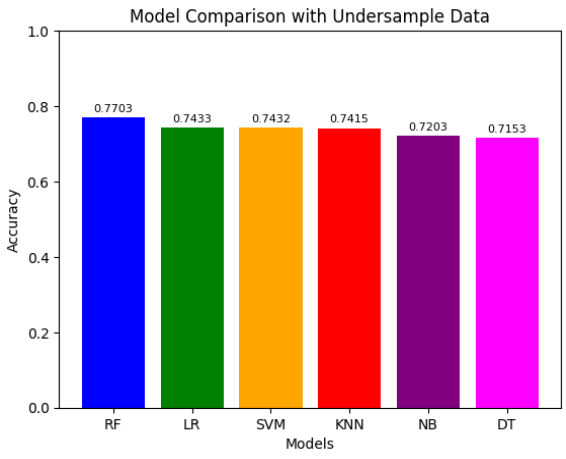



Fig. 11. Undersample Accuracies Among Models

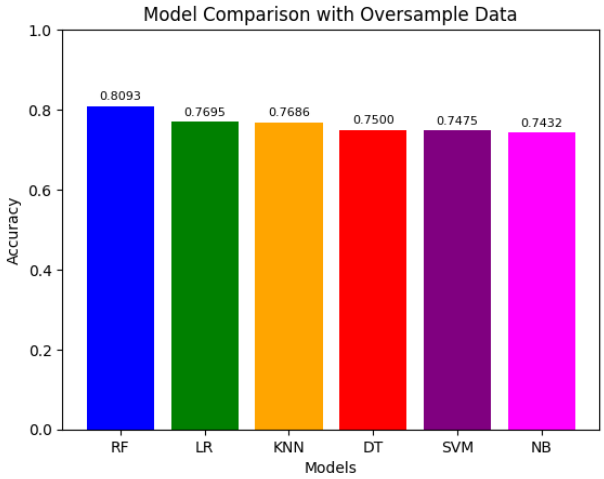



Fig. 12. Oversample Accuracies Among Models

and Figure 12. Random Forest had the best accuracies out of the other five models with both undersample and oversample data. In general, the oversampled data had higher accuracies compared to the undersampled. This could be due to there being more training data for the models to learn from. The order of accuracy seems to uphold the same manner when dealing with undersample or oversample version of the data.

Figures 13 show the confusion matrix for each model. The y-axis represents the actual condition of the patient. Class 0 means that the patient does not have CAD whereas Class 1 means that the patient does have CAD. The horizontal axis represents the prediction made by the model. The bottom right corner of the confusion matrix is the true positive, in other words, the number of patients that have been correctly diagnosed or predicted to have CAD. The bottom left corner is the number of patients that have been misdiagnosed as not having CAD by the model.

Figure 14 and Figure 15 display the recall results of the models. The results showed that the Random Forest algorithm has a better performance in determining who has CAD versus those who do not. Compared to the accuracy graph, the order of who has the highest recall differs. After Random Forest, SVM and Logistic Regression had the next highest recall and not Naive Bayes. As you observed, Random Forest performed better in terms of recall. This indicates its effectiveness in handling complex, non-linear relationships in data, which is often the case in medical diagnostics.

## V. Conclusion and Future Work

In conclusion, our study found that the Random Forest algorithm emerged as the most effective model for developing a predictive model of coronary artery disease (CAD). Among the various machine-learning algorithms tested, Random Forest consistently outperformed others in terms of accuracy and recall, regardless of whether the dataset was oversampled or undersampled.

Interestingly, oversampling generally improved the accuracy and recall values, but it did not significantly alter the relative performance of the models against each other. This

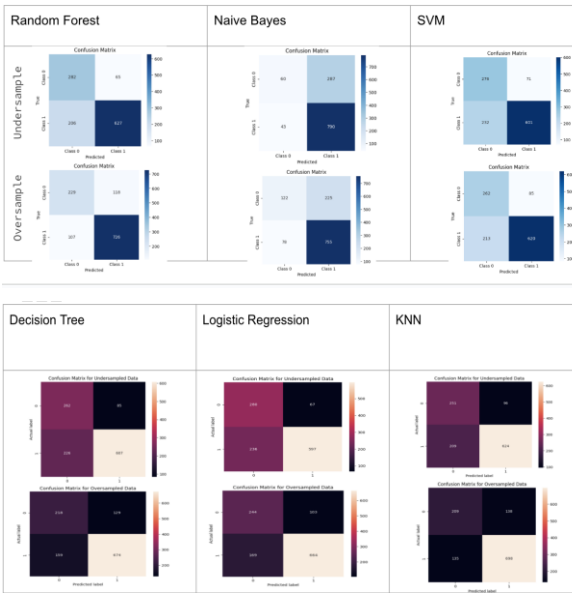



Fig. 13 Confusion Matrices

consistency in model ranking between undersampled and oversampled datasets underscores the robustness of our findings.

A notable observation was the performance of the Naive Bayes algorithm. Despite its higher accuracy in some cases, it exhibited the lowest recall values, casting doubts on its reliability for this specific application. This discrepancy between accuracy and recall highlights the importance of considering multiple performance metrics when evaluating machine-learning models.

The feature importance analysis from the Random Forest model revealed insightful predictors for CAD. Hypertension was identified as a primary indicator, followed by age and Brain Natriuretic Peptide (BNP) levels. These findings could guide clinicians in early diagnosis and risk assessment for CAD.

Looking ahead, there are several avenues for future work. Expanding the study to include a broader range of machine-learning algorithms could provide deeper insights into model efficacy. Additionally, further fine-tuning of the models and incorporating datasets from diverse regions with different characteristics could enhance the accuracy and generalizability of the predictive models.

One limitation of our study was the exclusion of data regarding patients' cholesterol levels due to missing or invalid entries. Future experiments could explore the impact of imputing these missing values on model performance. While we opted against this approach due to concerns about data integrity, investigating the effects of different data imputation strategies could be beneficial. This exploration would be particularly relevant given the known significance of cholesterol levels in CAD risk.

In summary, our study contributes to the growing body of research on machine learning in medical diagnostics, particularly in the context of CAD prediction. The insights gained from this research not only advance our understanding of algorithmic efficacy in healthcare applications but also underscore the importance of comprehensive feature analysis and the careful consideration of data integrity in model training. The potential for further research in this domain is vast, with opportunities to enhance predictive accuracy and broaden the applicability of these models in diverse clinical settings.

## REFERENCES

[1] J. C. Brown, T. E. Gerhardt, and E. Kwon, "Risk Factors for Coronary Artery Disease," ncbi.nlm.nih.gov. Available: https://www.ncbi.nlm.nih.gov/books/NBK554410/#. (accessed: Sept. 23, 2023). [Online].

[2] Cleveland Clinic, “Coronary Artery Disease,” my.clevelandclinic.org. Available: https://my.clevelandclinic.org/health/diseases/16898-coronary-artery-disease. (accessed: Sept. 23, 2023). [Online].

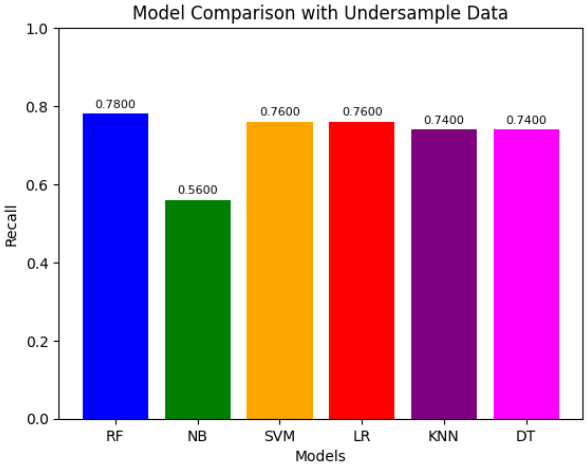



Fig. 14. Undersample Recall Among Models

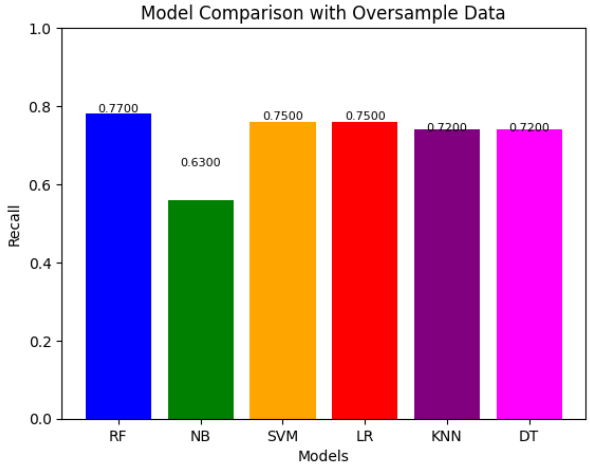



Fig. 15. Oversample Recall Among Models

[3] NIH, “Coronary Heart Disease - Treatment | NHLBI, NIH,” nhlbi.nih.gov, Mar. 24, 2022. https://www.nhlbi.nih.gov/health/coronary-heart-disease/treatment. (accessed: Sept. 25, 2023). [Online].

[4] M. M. Ahsan, S. A. Luna, and Z. Siddique, “Machine-Learning-Based Disease Diagnosis: A Comprehensive Review,” Healthcare, vol. 10, no. 3, p. 541, Mar. 2022, doi: https://doi.org/10.3390/healthcare10030541.

[5] L. J. Muhammad, I. Al-Shourbaji, A. A. Haruna, I. A. Mohammed, A. Ahmad, and M. B. Jibrin, “Machine Learning Predictive Models for Coronary Artery Disease,” SN Computer Science, vol. 2, no. 5, Jun. 2021, doi: https://doi.org/10.1007/s42979-021-00731-4.

[6] M. Abdar, W. Książek, U. R. Acharya, R.-S. Tan, V. Makarenkov, and P. Pławiak, “A new machine learning technique for an accurate diagnosis of coronary artery disease,” Computer Methods and Programs in Biomedicine, vol. 179, p. 104992, Oct. 2019, doi: https://doi.org/10.1016/j.cmpb.2019.104992.

[7] A. Akella and S. Akella, “Machine learning algorithms for predicting coronary artery disease: efforts toward an open source solution,” *Future Science OA*, p. FSO698, Mar. 2021, doi: https://doi.org/10.2144/fsoa-2020-0206.

[8] C. Wang et al., “Development and Validation of a Predictive Model for Coronary Artery Disease Using Machine Learning,” Frontiers in Cardiovascular Medicine, vol. 8, Feb. 2021, doi: https://doi.org/10.3389/fcvm.2021.614204.

[9] Hospital Admissions Data, kaggle.com, 2021. Available: https://www.kaggle.com/datasets/ashishsahani/hospital-admissions-data. [Online].

[10] S.C. Bollepalli, A.K. Sahani, N. Aslam, B. Mohan, K. Kulkarni, A. Goyal, B. Singh, G. Singh, A. Mittal, R. Tandon, S.T. Chhabra, G.S. Wander, A.A. Armoundas, An Optimized Machine Learning Model Accurately Predicts In-Hospital Outcomes at Admission to a Cardiac Unit, Diagnostics, 2022. Available: https://doi.org/10.3390/diagnostics12020241.(accessed Nov 20, 2023). [Online].

[11] IBM, “What is Logistic Regression?,” ibm.com. Available: https://www.ibm.com/topics/logistic-regression. (accessed Nov. 24, 2023). [Online].

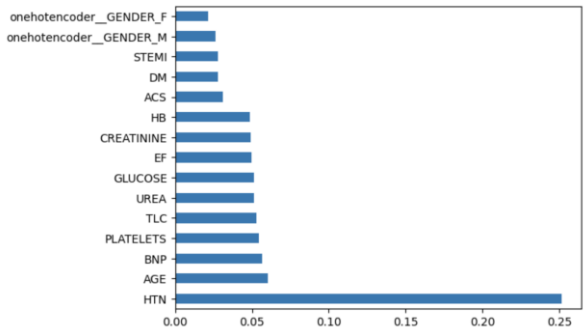



Fig. 16. Best Model Top Features