Project on
Cardiovascular Diseases Prediction
using Machine Learning

1. Introduction

Cardiovascular diseases (CVDs) are most frequent reason of death nowadays. According to WHO, an estimated 17.9 million people died from CVDs in 2019, representing 32% of all global deaths. Of these deaths, 85% were due to heart attack and stroke. Cardiovascular diseases are conditions that affect the function of your heart and blood vessels that include coronary heart disease, cerebrovascular disease, rheumatic heart disease and other conditions. Most of the CVD deaths are due to heart attacks and strokes, and one third of these deaths occur in people who are under 70 years old. Most cardiovascular diseases are related to behavioral risk factors such as tobacco use, unhealthy diet, obesity, physical inactivity and harmful use of alcohol. In this project, one of the most reliable machine learning techniques, random forest, have been performed for CVD detection using a dataset from Kaggle, a public data repository. Outliers and unrelated observations have been removed to increase the model performance. Based on the results, we obtained 70 % accuracy in detection using the Random Forest model. All the analysis has been performed in Python 3.

2. Problem Definition and Algorithm

2.1 Task Definition

Based on The World health Organization estimation, nearly 31% of global deaths are related to Cardiovascular diseases. So, we can say that Cardiovascular Disease is one of the most dangerous disease in the whole world. So, it is very important to be able to diagnose it on time to prevent catastrophic consequences related to it, and also to find the proper factors that have an effect on these kinds of disease, because it is important to detect cardiovascular disease as early as possible so that the patients will be treated and fully cured. Some of the significant behavioral risk factors of these types of disease are unhealthy diet, physical inactivity, tobacco use and harmful use of alcohol. These factors may cause raised blood pressure, raised blood glucose, raised blood lipids,

and obesity. So, it is crucial to investigate these factors to find out a pattern in the relationship between the behavioral risk factors and Cardiovascular diseases.

2.2 Algorithm Definition

In this project, we used random forest algorithm to predict presence or absence of Cardiovascular diseases based on the provided features. There are different types of Machine Learning algorithms such as Supervised learning, Unsupervised learning, Semi-supervised learning, and Reinforcement learning. Depending on the nature of the data and the target response, we can decide which machine learning technique is suitable for our model. Here, we used classification technique which is a supervised learning method in machine learning. This technique is for predictive modeling, where a class label is predicted for a given example. Mathematically, it uses a function (f) to assign each input variables (X) to output variables (Y) as target response. To predict the class of given data points, it can be carried out on structured or unstructured data. Binary classification refers to the label classification into two levels like "true and false", "yes and no" or "0 and 1". In binary classification, one class is the normal state and the other class is abnormal state. Here, the target response which the presence or absence of cardiovascular disease is binary: "0" for absence and "1" for presence of cardiovascular disease. In binary classification problems, we consider (x_i, y_i) , $i = 1,2,\ldots,N$ as independent and identically distributed random variables of a p-dimensional vector x and a response variable y.

On the other hand, there are different types of machine learning algorithms for supervised technique. One of the most popular methods is random forest classifier which is a classification method. Random forest models are based on averaging over a large collection of decision trees, each trained on a separate bootstrap sample of the input set. Random decision forests correct for decision trees' habit of overfitting to their training set. The random forest learning model with multiple decision trees is more accurate than a single decision tree model. Meanwhile, random forest models are suitable for both classification and regression problems and can be used for both categorical and continuous values.

On the other hand, we used KNN algorithm in modeling to provide the performance comparison between these two algorithms. The K-Nearest Neighbors (KNN) algorithm is a supervised machine

learning algorithm used for classification and regression. It is a simple and versatile algorithm that works by identifying the K nearest data points to a given input data point, based on a specified distance metric. The algorithm then assigns a label or value to the input data point based on the labels or values of its nearest neighbors. In classification, the label with the highest frequency among the K nearest neighbors is assigned to the input data point, while in regression, the average or weighted average of the values of the K nearest neighbors is assigned. Moreover, we applied Logistic Regression model to check if we can improve the accuracy of them model or not.

3. Experimental Evaluation

3.1 Methodology

In this project, a dataset of 70000 patients has been analyzed which is included 11 attributes and the target variable. The dataset is obtained from Kaggle, a public data repository for datasets. We investigated this dataset to predict presence or absence of Cardiovascular diseases based on the provided features. The attributes are Age, Gender, Height, Weight, Systolic blood pressure, Diastolic blood pressure, Cholesterol (1: normal, 2: above normal, 3: well above normal), Glucose (1: normal, 2: above normal, 3: well above normal), Smoking, Alcohol intake, Physical activity. Our target variable is whether a person has CVD or does not. We used Random Forest method as the classification algorithm to comprise training of the model and then validating our model based on test data of patients. Finally, performance measurements are presented. The required steps for model building to predict CVD is as following:

Step 1: Data cleaning and pre-processing has been implemented by eliminating improper values and removing outliers from the observation.
Step 2: EDA (Exploratory Data Analysis) has been performed to analyze and summarize data sets in order to gain insights into the underlying patterns, distributions, and relationships between variables.
Step 3: feature engineering and feature selection have been performed to choose the proper attributes which are more helpful in CVD prediction.

Step 4: Random forest, KNN and Logistic Regression algorithms are chosen to classify the selected features.

Step 5: performance measures are evaluated to gain proper result from the models.

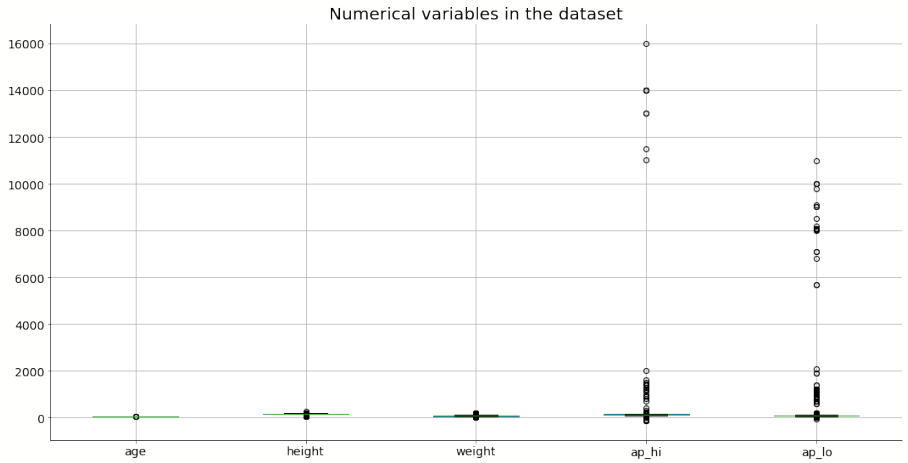
2.2 Data Cleaning

First, we implemented the data pre-processing and data cleaning. The process of detecting and investigating the data to find and remove inaccurate and improper observation from the data set is called as data cleaning. It is an important step in providing the proper data to find the best model. By looking at the statistical description of the data, we can get some information about the dataset. For example, we can say that there are no missing values in the dataset and the "age" variable is in days which have been changed to years. Besides, by looking at the statistical description of numerical variables, we recognized that Systolic blood pressure(ap_hi) and Diastolic blood pressure (ap_lo) has negative values which is not acceptable. So, we filtered out the unrealistic data of Systolic blood pressure and Diastolic blood pressure from the dataset. Also, these two variables cannot be negative, so we removed the negative values too.

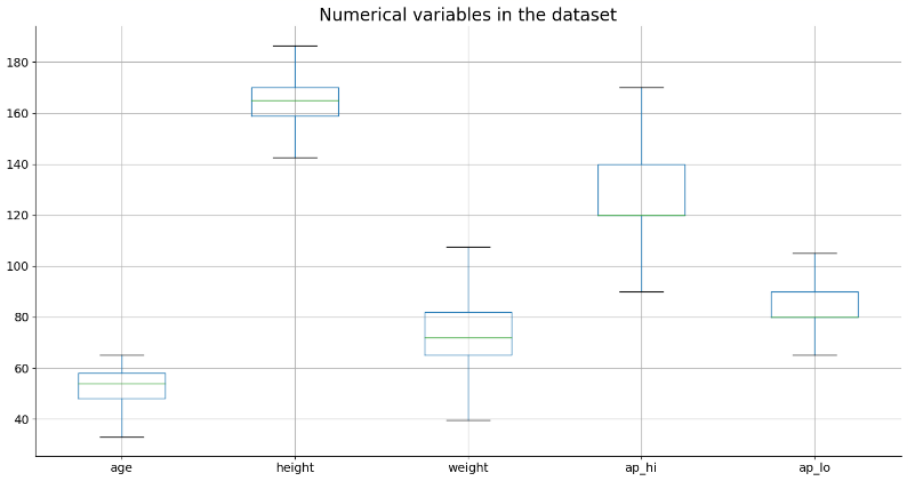
According to the American Heart Association, if the Systolic blood pressure and Diastolic blood pressure values exceed 180/120 mm Hg, it will be a hypertensive crisis. So, for safety we considered the values greater than 250 Systolic blood pressure and 200 for Diastolic blood pressure as outliers and they need to be removed. Meanwhile, we checked the number of duplicate observations, we removed 24 duplicated observation from the dataset.

	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio
count	69976.000000	69976.000000	69976.000000	69976.000000	69976.000000	69976.000000	69976.000000	69976.000000	69976.000000	69976.000000	69976.000000	69976.000000
mean	53.338945	1.349648	164.359152	74.208371	128.820453	96.636261	1.366997	1.226535	0.088159	0.053790	0.803718	0.499771
std	6.765633	0.476862	8.211218	14.397283	154.037729	188.504581	0.680333	0.572353	0.283528	0.225604	0.397187	0.500004
min	30.000000	1.000000	55.000000	10.000000	-150.000000	-70.000000	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	48.000000	1.000000	159.000000	65.000000	120.000000	80.000000	1.000000	1.000000	0.000000	0.000000	1.000000	0.000000
50%	54.000000	1.000000	165.000000	72.000000	120.000000	80.000000	1.000000	1.000000	0.000000	0.000000	1.000000	0.000000
75%	58.000000	2.000000	170.000000	82.000000	140.000000	90.000000	2.000000	1.000000	0.000000	0.000000	1.000000	1.000000
max	65.000000	2.000000	250.000000	200.000000	16020.000000	11000.000000	3.000000	3.000000	1.000000	1.000000	1.000000	1.000000

Next step is to investigate the outliers, we have visualized the numerical quantities in the dataset as boxplots, to have a better sense of the outliers. The boxplot of numerical variables like "age", "height"," weight","ap_hi" and "ap_lo" is as below:

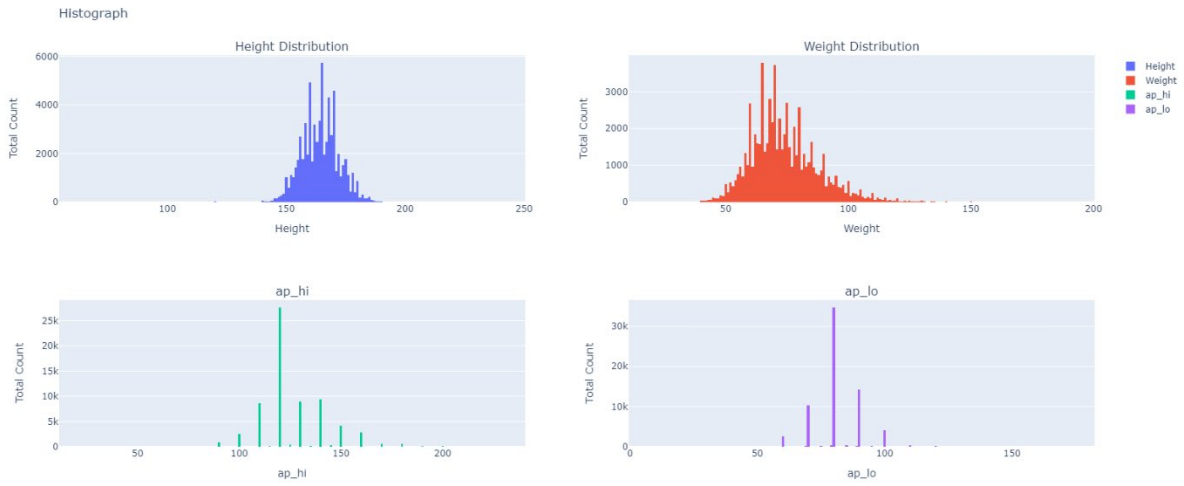


As can be seen from the boxplot, there are too many outliers and if we remove all the outliers, then almost 10% of the data will be missed. So, we decided to use another method and replace the outliers with the first and third quartile. Clamping is a statistical process used to replace extreme values or outliers with more representative values. One method of clamping involves using the first and third quartile of a data set to identify potential outliers. The first quartile is the value below which 25% of the data falls, while the third quartile is the value below which 75% of the data falls. Any value outside of the interquartile range (IQR), which is defined as the difference between the third and first quartiles, multiplied by a factor (typically 1.5 or 3), is considered an outlier. To clamp outliers using the quartile method, values outside of the IQR are replaced with the closest value within the range of the IQR. This process helps to reduce the influence of extreme values on statistical analysis and modeling. After removing unnecessary observation and fixing the outliers, data, we recognized that 1.86% of the observations has been missed. Now, we have 68699 observation in the dataset. The boxplot after fixing the outliers is as below:

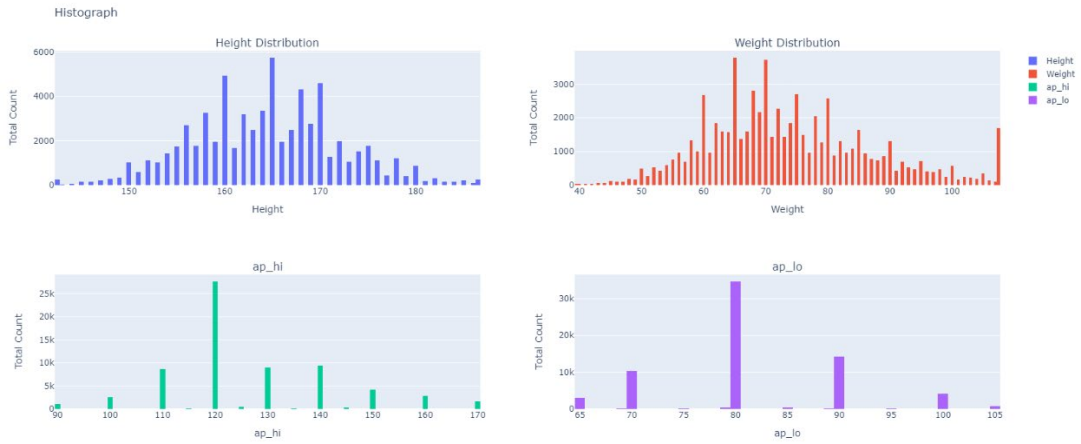


After clamping outliers using the quartile method, it is important to check that the resulting data set is still representative of the underlying population and that the clamping did not introduce any unintended biases. One way to check this is by examining the distribution of the data before and after clamping.

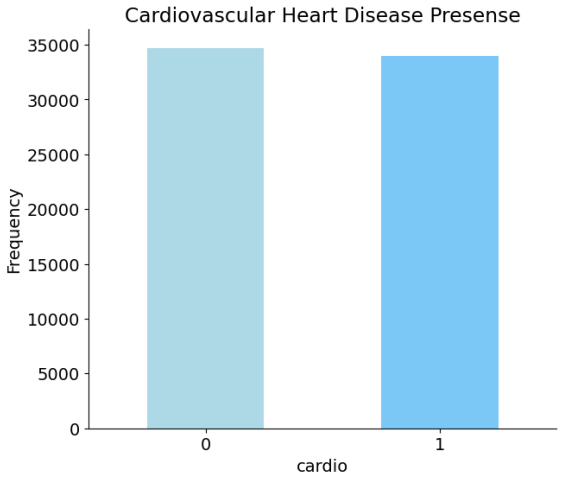
Distribution before fixing the outliers:



Distribution after fixing the outliers:

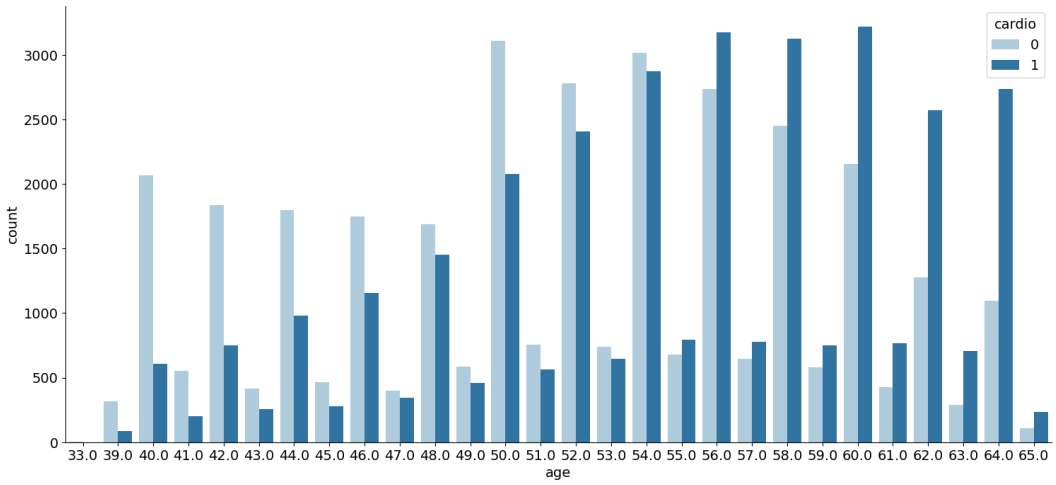


So, we can see that that fixing the outliers with clamping method did not change the distribution and statistical description of the data. Meanwhile, We have plotted the bar chart of target variable and as can be seen; the data set was balanced as 33989 patients had CVD and 34710 patients had not CVD.

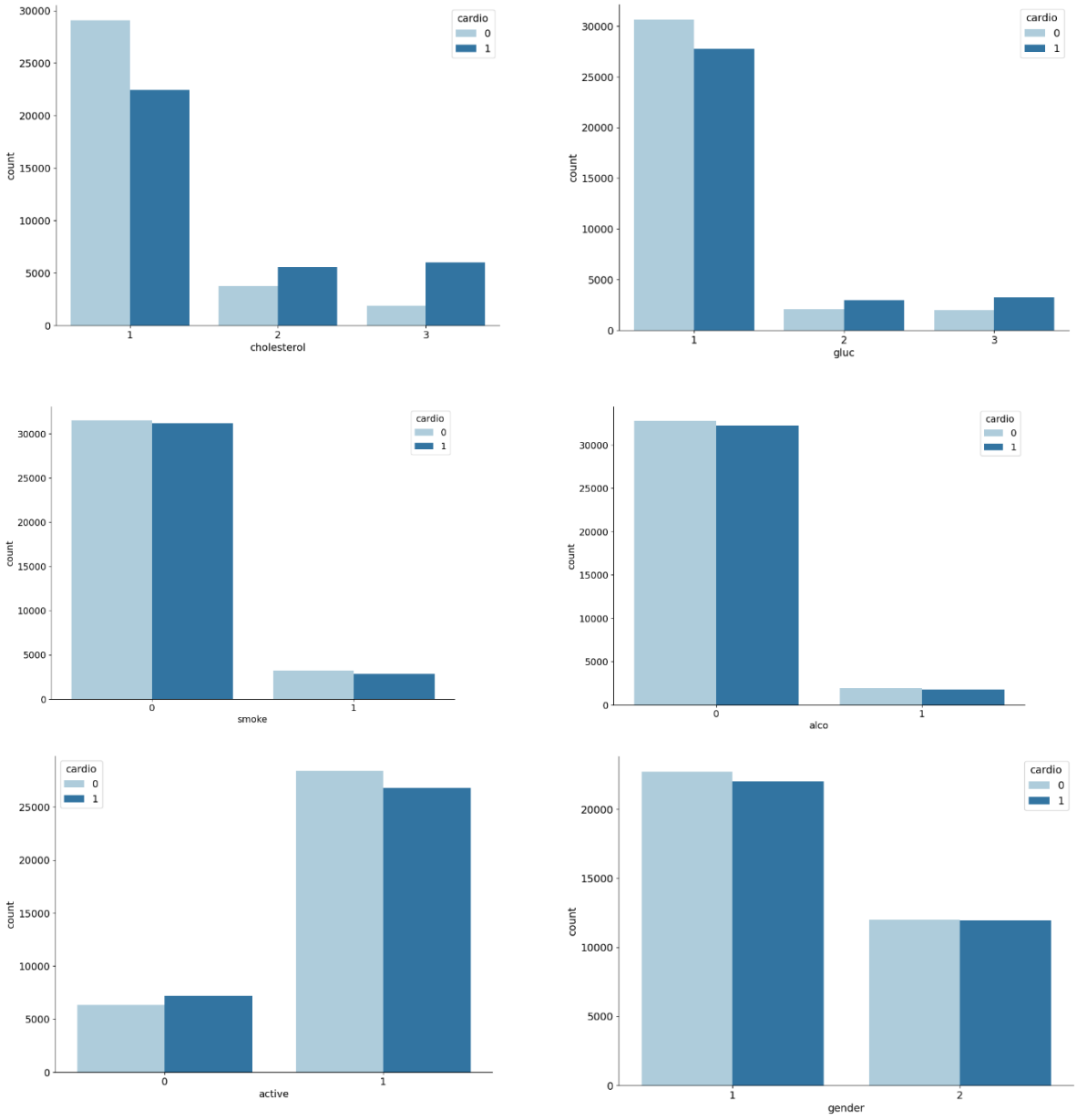


2.3 EDA (Exploratory Data Analysis)

First, we investigated the relationship of age variable with the target variable. One can notice that individuals who are older than 55 years of age have a higher risk of being affected by cardiovascular disease (CVD). Individuals who belong to younger age groups have a reduced likelihood of developing cardiovascular disease (CVD). The plot indicates a decrease in the incidence of non-cardiovascular disease (CVD) and an increase in the incidence of CVD after reaching the highest point at the age group of 55. It is evident that individuals who belong to older age groups are more susceptible to developing cardiovascular disease (CVD).



Then, we presented the relationship between categorical variables with target variable as following:

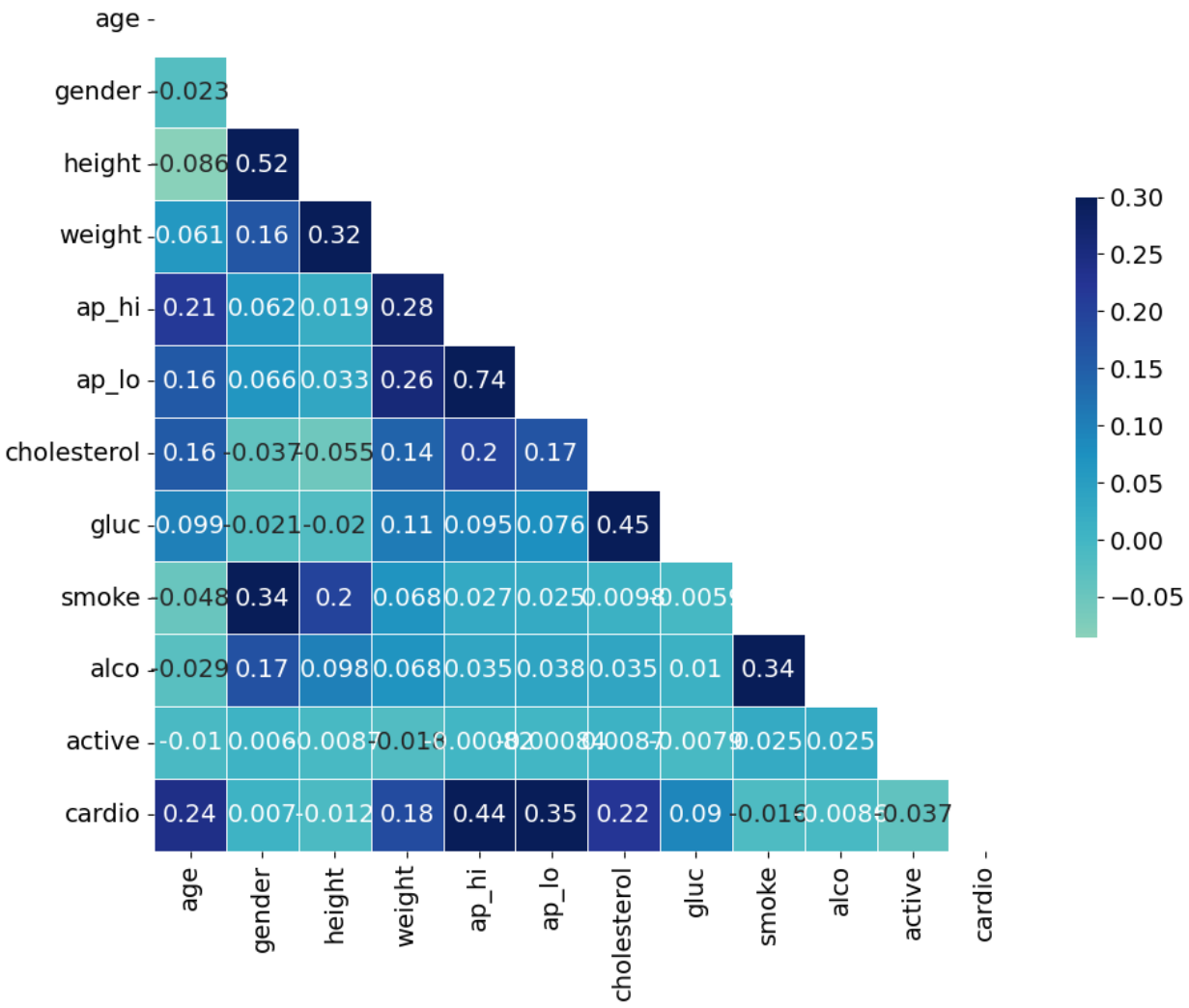


It is apparent that individuals who have cardiovascular disease (CVD) tend to exhibit elevated levels of cholesterol and blood glucose, as well as a lower level of physical activity overall. In the dataset, 44742 are women and 23957 are men and below is the crosstab presentation of how the target class is distributed among men and women;

Cardio/gender	Female	Male
0	0.33	0.17
1	0.32	0.17

3.3.1 Multivariate Analysis

It could be beneficial to take into account the correlation matrix. We can observe that Systolic blood pressure and Diastolic blood pressure are most correlated variables with the target variable.



3.3.2 Feature Engineering

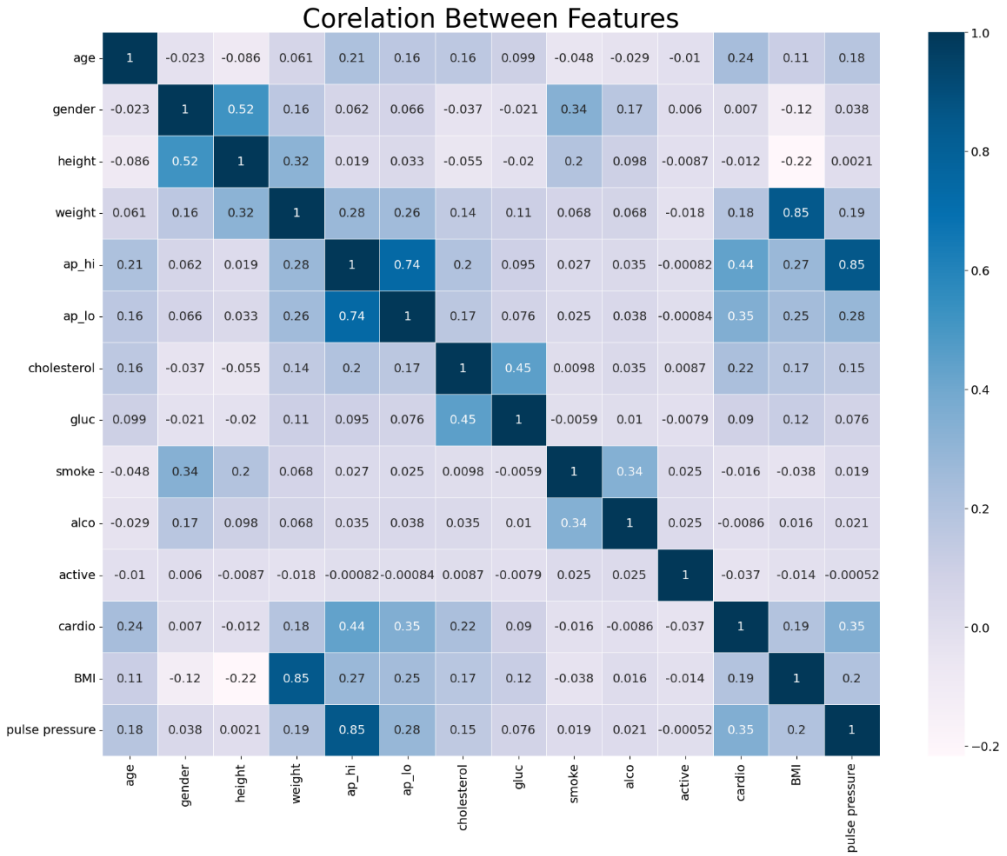
Whenever height and weight measurements are available, it is possible to compute the body mass index (BMI). It may be advantageous to create an additional feature for BMI, as it could potentially yield more valuable insights. The body mass index (BMI) is a frequently used measurement for

assessing medical health and cardiovascular wellness. BMI can be calculated by the following: BMI = weight(kg) / height (cm) / height (cm) x 10,000

Pulse pressure is an additional indicator of cardiovascular wellbeing. Pulse Pressure can be calculated by the following: Pulse Pressure = systolic – diastolic. Usually, a pulse pressure that exceeds 60 can be a beneficial predictor of the likelihood of experiencing heart attacks or other cardiovascular ailments.

3.3.3 Feature Selection

The presence of extraneous features within a dataset may result in a decrease in the precision of the models, and can lead to the model learning based on features that are not relevant to the problem being studied. Therefore, we employ feature selection methods to choose relevant data for our analysis. There are various techniques for feature selection, but in this case, we will use a more conventional approach, which involves utilizing a correlation matrix. Now that we have additional features, let us construct a heatmap to investigate the correlations between the variables.



The plot of the correlation matrix indicates that the variable "ap_hi" has a positive correlation with the target output, with a coefficient of 0.44. This implies that the presence of "ap_hi" increases the likelihood of developing cardiovascular disease. Likewise, there are negative correlations between some variables, such as "active" and the target variable. The correlation coefficient of -0.037 suggests that if a person has high level of physical activity, they are less likely to suffer from cardiovascular disease (CVD). It is preferable to choose those that have a strong positive correlation with the target variable. In order to streamline the data and improve accuracy, one of these features can be eliminated during the pre-processing stage.

Essentially, you want to select features that have a strong correlation with the target feature, but a weak correlation with any independent feature. In other words, if two independent features have high correlation, it means they are both trying to represent the same thing. So, if one of the features is dropped, there won't be a significant loss of quality data. The feature "ap_hi" is correlated with several other features, but it has the highest correlation with the target value, so it cannot be ignored. The same is true for "pulse" and "ap_lo".

The feature alcohol has the lowest correlation with the target feature. Also, features such as 'height', 'smoke', and 'active' have relatively low correlation values with the target feature. To ensure the quality of our data, we will remove some features such as ‘height’ and 'alco'. Although 'age' and 'cholesterol' have a significant impact, their correlation with the target class is not very high. The feature 'ap_hi' has the highest correlation with the target value, indicating that it has a significant impact on the model. Similarly, 'ap_lo' also has a strong correlation with the target value and is important for the model.

3.4 Modeling

Train-test split is a commonly used method in machine learning to evaluate the performance of a predictive model. The basic idea behind this technique is to split the available data into two sets: one for training the model and the other for testing its performance. The training set is used to build the model by learning the patterns and relationships between the features and the target variable. The test set is then used to evaluate the model's ability to generalize to new, unseen data. The goal is to develop a model that performs well on both the training and test sets, which indicates

that it has learned the underlying patterns in the data and can make accurate predictions on new data. The ratio of the training set to the test set can vary depending on the size and complexity of the dataset, but a common ratio is 80% for training and 20% for testing. By using train-test split, we can avoid overfitting the model to the training data, which would result in poor performance on new data. In this scenario, we will be using the commonly used split ratio of 80:20, which means that 80% of the dataset will be used for training the model, and the remaining 20% of the dataset will be used for testing the model. In this project, we used random forest and KNN algorithms to model the data.

3.5 Performance Evaluation

A collection of performance metrics is used to evaluate the capability of ML algorithms. Obtaining a confusion matrix for both actual and predicted data, which includes true positive (TP), false positive (FP), true negative (TN), and false negative (FN), is necessary to assess the parameters. A confusion matrix is a table that is often used in machine learning to evaluate the performance of a classification model. It presents a summary of the model's predictions versus the actual values for a given set of data, and displays the number of true positives, true negatives, false positives, and false negatives. These metrics are useful for assessing the accuracy, precision, recall, and other performance measures of a classifier.

3.5.1 Confusion Matrix Parameters

Confusion matrix is a table that summarizes the number of true positive, true negative, false positive, and false negative predictions made by a model. These metrics can be used to compare different models and to optimize the hyperparameters of a model. The choice of which metric to use depends on the specific problem being solved. Four parameters or metrics that are used to evaluate the performance of the model are as following:

Confusion Matrix Parameters	
Confusion matrix parameters	**Definition**
True positives (TP)	The number of instances that belong to the positive class and are correctly classified as positive by the model

False positives (FP)	The number of instances that belong to the negative class but are incorrectly classified as positive by the model
False negatives (FN)	The number of instances that belong to the positive class but are incorrectly classified as negative by the model
True negatives (TN)	The number of instances that belong to the negative class and are correctly classified as negative by the model

These four parameters can be used to calculate various performance metrics, such as precision, recall, F1-score, and accuracy.

3.5.2 Definition of performance metrics

Performance metrics in machine learning modeling are used to evaluate the effectiveness of a model in predicting outcomes based on input data. These metrics are used to measure the accuracy, precision, recall, and other properties of a model's predictions. Here are some common performance metrics used in machine learning:

Performance Metrics Definition	
Performance metrices	**Definition**
Accuracy	The proportion of correct predictions made by the model. $\frac{True\ positive\ +\ True\ nagative}{True\ positive\ +\ True\ nagative\ +\ False\ positive\ +\ False\ negative}$
Precision	The proportion of true positive predictions out of all positive predictions made by the model. $\frac{True\ positive}{True\ positive\ +\ False\ positive}$
Recall	The proportion of true positive predictions out of all actual positive cases. $\frac{True\ positive}{True\ positive\ +\ False\ negative}$
F1 score	A weighted average of precision and recall $\frac{2 * (\text{precision} * \text{recall})}{\text{precision} + \text{recall}}$
Support	The number of actual occurrences of a class in the provided data set

Macro average	This means that each class contributes equally to the final score, regardless of its size or frequency in the dataset
Weighted avg	The weight of each class's contribution to the average

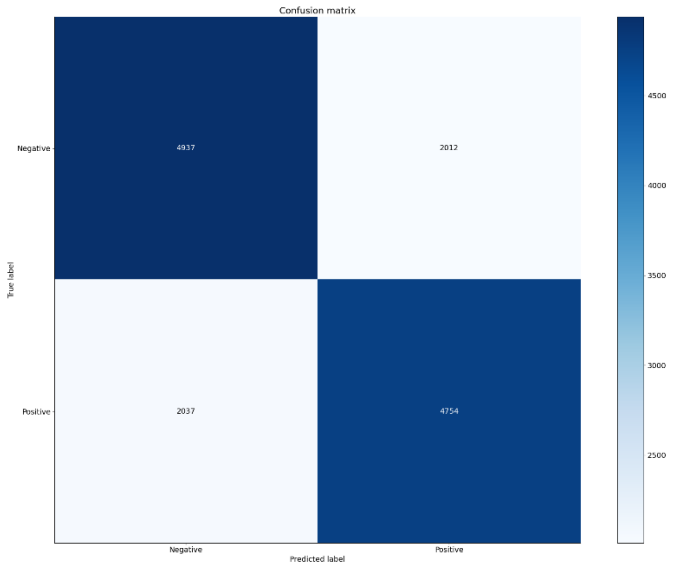
3. Results

3.1 Random Forest Performance

After making predictions on the test dataset, we obtained the performance metrics which are presented in the following classification report and the confusion matrix.

Table 1 Classification Report of Random Forest Algorithm

Classification Report				
	precision	**recall**	**f1-score**	**support**
0	0.71	0.71	0.71	6949
1	0.70	0.70	0.70	6791
accuracy			0.71	13740
macro avg	0.71	0.71	0.71	13740
weighted avg	0.71	0.71	0.71	13740



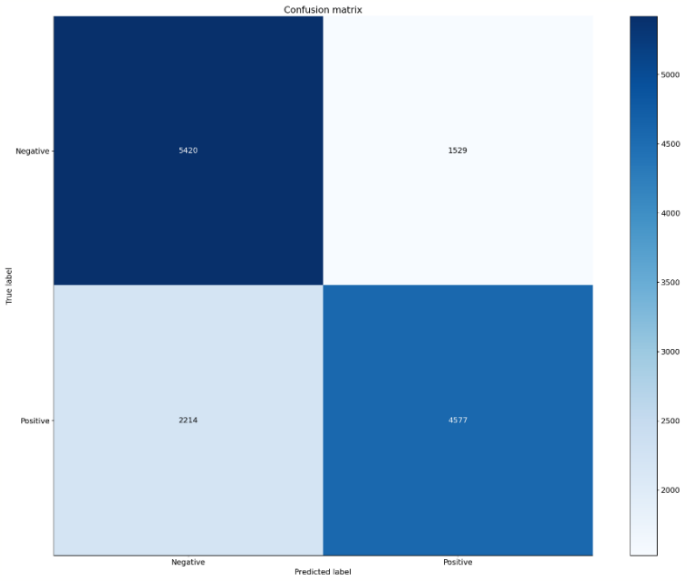
Based on the accuracy, we can say that the model correctly predicted the outcome for 71% of the patient's CVD in the test dataset. The precision says that of the patients that the model predicted

to have CVD, 70% actually had CVD. This is a moderate precision score but it suggests that the model may be missing some patient's CVD history. The Recall indicates that of the patients that actually had CVD, the model correctly identified 70%. Moreover, The F1-score is a balance between precision and recall, and is 0.70 in this case. This indicates that the model is performing well overall, with a good balance between precision and recall. According to these performance metrics, we can conclude that the Random Forest model is performing well overall and is correctly identifying a high proportion of patient's CVD. However, the model may need to be further tuned to improve its recall score and identify more patients who had CVD.

3.2 The K-Nearest Neighbors (KNN) algorithm Performance

Table 2 Classification Report of KNN Algorithm

Classification Report				
	precision	**recall**	**f1-score**	**support**
0	0.71	0.78	0.74	6949
1	0.75	0.67	0.71	6791
accuracy			0.73	13740
macro avg	0.73	0.73	0.73	13740
weighted avg	0.73	0.73	0.73	13740

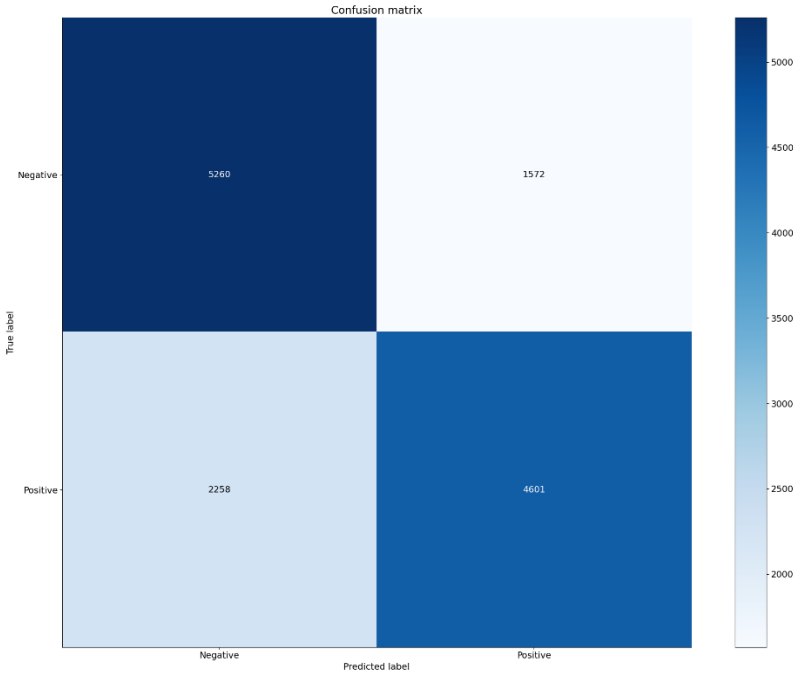


Based on the accuracy, we can say that the model correctly predicted the outcome for 73% of the patient's CVD in the test dataset. The precision says that of the patients that the model predicted to have CVD, 75% actually had CVD. So, we can conclude that KNN provides better performance than random forest for this dataset.

3.3 Logistic Regression Algorithm Performance

Table 3 Classification Report of Logistic Regression Algorithm

Classification Report				
	precision	**recall**	**f1-score**	**support**
0	0.70	0.77	0.73	6832
1	0.75	0.67	0.71	6859
accuracy			0.72	13691
macro avg	0.72	0.72	0.72	13691
weighted avg	0.72	0.72	0.72	13691



As can be seen from the results, the accuracy of logistic regression is almost the same as KNN algorithm.

4. Discussion:

In this project, we have investigated the behavior and patterns in a dataset of cardiovascular disease to provide a prediction of presence of CVD among the patients. In this regard, we used Random Forest, KNN algorithms and Logistic Regression to perform the prediction. While all these algorithms have their strengths and weaknesses, studies have shown that KNN can provide better accuracy. It is important to carefully consider the characteristics of the dataset and the requirements of the task when selecting an algorithm for classification.

5. References:

1- https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)

2- https://www.heart.org/en/health-topics/consumer-healthcare/what-is-cardiovascular-disease

3- Aamir Javaid, Fawzi Zghyer, Chang Kim, Erin M. Spaulding, Nino Isakadze, Jie Ding, Daniel Kargillis, Yumin Gao, Faisal Rahman, Donald E. Brown, Suchi Saria, Seth S. Martin, Christopher M. Kramer, Roger S. Blumenthal, Francoise A. Marvel,Medicine 2032: The future of cardiovascular disease prevention with machine learning and digital health technology,American Journal of Preventive Cardiology,Volume 12,2022,100379,ISSN 2666-6677

4- Sarker, I.H. Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN COMPUT. SCI.* **2**, 160 (2021)

5- Pal M, Parija S, Panda G, Dhama K, Mohapatra RK. Risk prediction of cardiovascular disease using machine learning classifiers. Open Med (Wars). 2022 Jun 17;17(1):1100-1113. doi: 10.1515/med-2022-0508. PMID: 35799599; PMCID: PMC9206502.