

# Project Report

---

Classification Based on Heart Disease Dataset

Group 06: Kefan Yu, Wendi Chu, Zifeng Xu  
Georgetown University | Data Science and Analytics

# Table of Contents

<b>1. Project Charter.....</b>	<b>3</b>
1.1 Project Description.....	3
1.2 Limitation and Success Criteria.....	3
1.3 Methods.....	4
1.4 Deliverables.....	5
1.5 Out of Scope.....	5
<b>2. Introduction.....</b>	<b>5</b>
<b>3. Exploratory Analysis and Visualization.....</b>	<b>6</b>
<b>4. Model Training and Selection.....</b>	<b>8</b>
4.1 Model Selection.....	8
4.2 Stacked Model and Model Performance.....	9
4.3 Hyperparameter Tuning.....	10
4.4 Resampling.....	11
4.5 Feature Importance.....	12
<b>5. App Deployment.....</b>	<b>15</b>
5.1 PCA and Visualization.....	15
5.2 Screen.....	17
<b>6. Conclusion.....</b>	<b>17</b>

# 1. Project Charter

## 1.1 Project Description

In this project, we are going to apply classification models on a heart disease dataset with target variable “HeartDiseaseorAttack”. This project aims to classify whether people have heart attacks or not. After evaluating individual models, we will stack them into one stacked model for deployment.

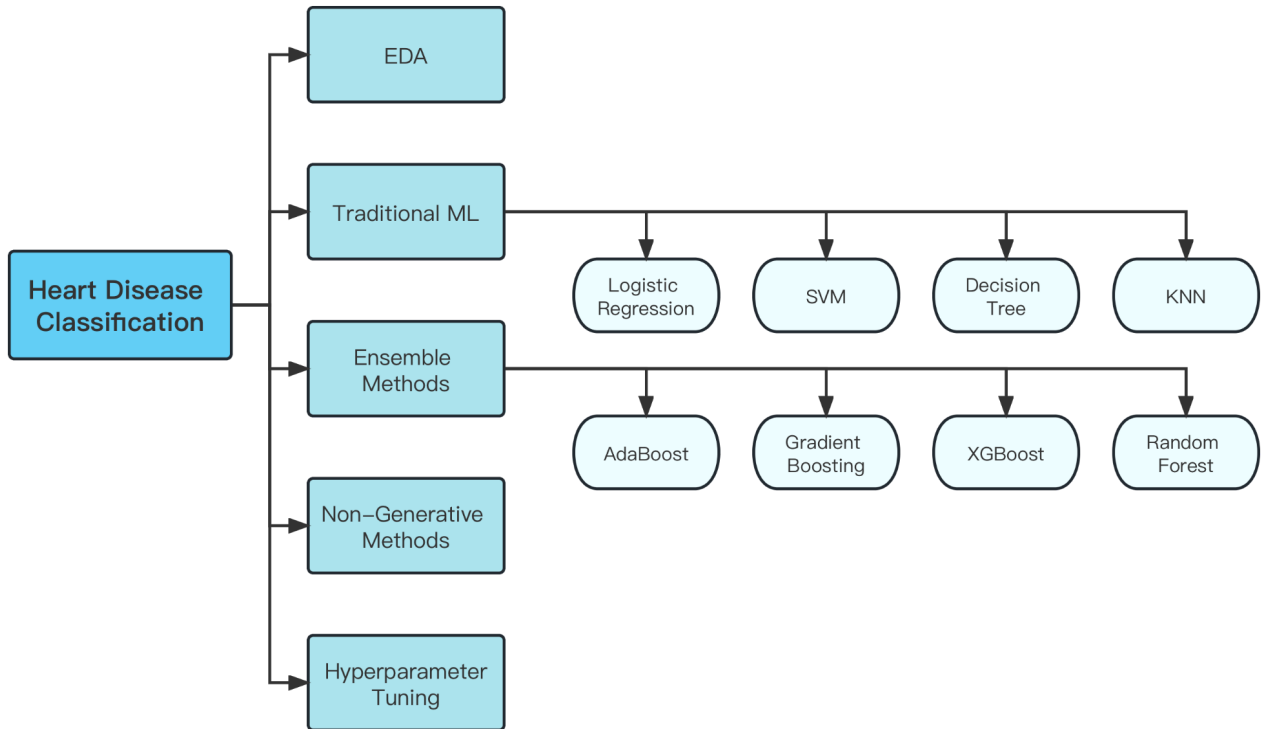
## 1.2 Limitation and Success Criteria

Although the Heart disease indicators dataset contains variables in various aspects encompassing body diseases indicators and lifestyle indicators, which is suitable for classification in the way of predicting heart disease, some features might not be useful to analyze. Specifically, variable “education” represents the level of education a person has attained, ranging from kindergarten to a 4-year university degree. While education is undoubtedly important in various aspects of life, including socioeconomic status and access to healthcare, it may not have a direct and immediate impact on the physiological or lifestyle factors that are typically associated with heart disease. Therefore, this kind of irrelevant variable might need to be removed using feature selection.

Similarly, the non-relevant variables that might need to be removed using feature selection encompass “MentHlth” and “PhysHlth”, standing for mental health issues and physical health issues that measure how many days during the past 30 days is patient’s mental health or physical health not good. These are not good variables because they only measure past 30 days status. Intuitively, heart disease is a long-term disease and past 30-day body statuses play insignificant roles. Also from the distribution of these two columns, the majority of the patients have 0 in these two variables, indicating that they don’t have health issues in the past 30 days, which is heavily biased and might need to be removed.

However, this dataset is a successful candidate for predicting target variables. Specifically, the heart disease indicator dataset is a cleaned dataset from Behavioral Risk Factor Surveillance System (BRFSS), which is a health-related telephone survey that collects responses from over 400,000 Americans on health-related risk behaviors, chronic health conditions, and the use of preventative services. Therefore the data is reliable and can be trusted for making decisions or drawing conclusions. Also with numerous attributes that are highly related to the target variables, the performance of classification would be much better.

## 1.3 Methods



*Figure 1: Flow chart of the methods in this project*

1. Exploratory data analysis
  - 1.1. Preliminary visual exploration of the data set
  - 1.2. Simple analysis based on visualizations, such as finding out relationship of certain independent variables according to the correlation heatmap
2. Traditional machine learning methods
  - 2.1. Use Logistic Regression to conduct classification for this dataset
  - 2.2. Use SVM to find the optimal hyperplane that best separates the data
  - 2.3. Use K-Nearest Neighbors to classify the heart disease data
  - 2.4. Use decision tree to conduct classification task
3. Ensemble methods
  - 3.1. Use Adaboost to evaluate the dataset.  
Parameters need to be considered: `n_estimators`, `learning_rate`
  - 3.2. Use Gradient Boosting to evaluate the dataset.  
Parameters need to be considered: `n_estimators`, `learning_rate`
  - 3.3. Use XGBoost to evaluate the dataset.  
Parameters need to be considered: `n_estimators`, `learning_rate`
  - 3.4. Use Random Forest to evaluate the dataset.

Parameters need to be considered: n\_estimators, max\_depth, min\_samples\_split, min\_samples\_leaf

4. Non-generative methods  
Repeat the procedures for the Non-generative method.
5. Hyperparameter tuning  
Use methods like grid search for hyperparameter tuning based on results of 2 and 3.

## 1.4 Deliverables

Deliverable ID#	Description
1	Project Charter
2	Traditional and ensemble trained models with preliminary results of the training and testing
3	Hyperparameter tuning process based on step 2, and final optimal parameters results
4	Machine learning model deployment
5	Final report of the project

## 1.5 Out of Scope

This project will not accomplish or include the following:
1. This project will not include any factors other than the independent variables in the dataset.
2. This project will not include any pathological analysis in biology.

# 2. Introduction

Nowadays, heart disease remains a significant global health concern, responsible for a substantial number of deaths. Understanding the intricate factors that contribute to heart disease is crucial for improving public health outcomes. This machine learning project seeks to unravel some potential factors that affect heart disease, aiming to uncover insights that can lead to better prevention.

The interest in understanding the factors affecting heart disease lies in its complexity and its far-reaching impact on public health. Heart disease doesn't have a single cause; it's influenced by a wide array of factors such as genetics, lifestyle choices, diet, physical activity, and more. These intricate interconnections present an exciting challenge for machine learning models, as unraveling these relationships can potentially find approaches for heart disease prevention.

The proposed solution to comprehending the factors influencing heart disease carries profound benefits. First, it can be used as a predictive tool to identify high-risk individuals, enabling timely

interventions. Moreover, a deeper understanding of these factors can inform public health policies, leading to more effective preventive measures on a broader scale.

In this project, we are primarily dealing with a classification problem. The dataset contains 253,680 survey responses that are collected annually by the CDC to be used primarily for the binary classification of heart disease. The independent variables include blood pressure, blood cholesterol, smoking, and so on. The goal is to develop a model that can accurately assess an individual's likelihood of developing heart disease, thereby aiding in early intervention and prevention efforts.

### 3. Exploratory Analysis and Visualization

This dataset comprises 253680 data points encompassing 22 features, each representing a public's health condition, with the target variable being the presence of heart disease. Notably, all features are of numeric nature, and there are no instances of missing values. Prior to the model application, our intent is to conduct exploratory data visualizations to glean fundamental insights into the dataset's characteristics and interrelationships.



Figure 2: Histograms of all 22 variables

These histograms reveal that the majority of features, such as "HighCol" or "ChoiCheck," predominantly exhibit binary values, signifying the presence or absence of specific medical conditions or lifestyle indicators, with 0 indicating "No" and 1 meaning "Yes". Notably, the target variable "HeartDiseaseAttack" displays a significant imbalance, with 229787 respondents indicating the absence of heart disease, while 23893 respondents have reported a history of heart disease. Although such imbalance is intuitive given the low prevalence of heart attacks among the public, it may significantly impact model performance. Moreover, similar imbalances are observed in other features, including "CholCheck", "Strokes", "Diabetes", etc, where one label substantially outweighs the other, might cast influences on classification outcomes.

For variables like “BMI”, we can see that its distribution is skewed to the left away from the normal distribution. Some people have extremely high BMI values compared to the majority of them, leading to such skewness.

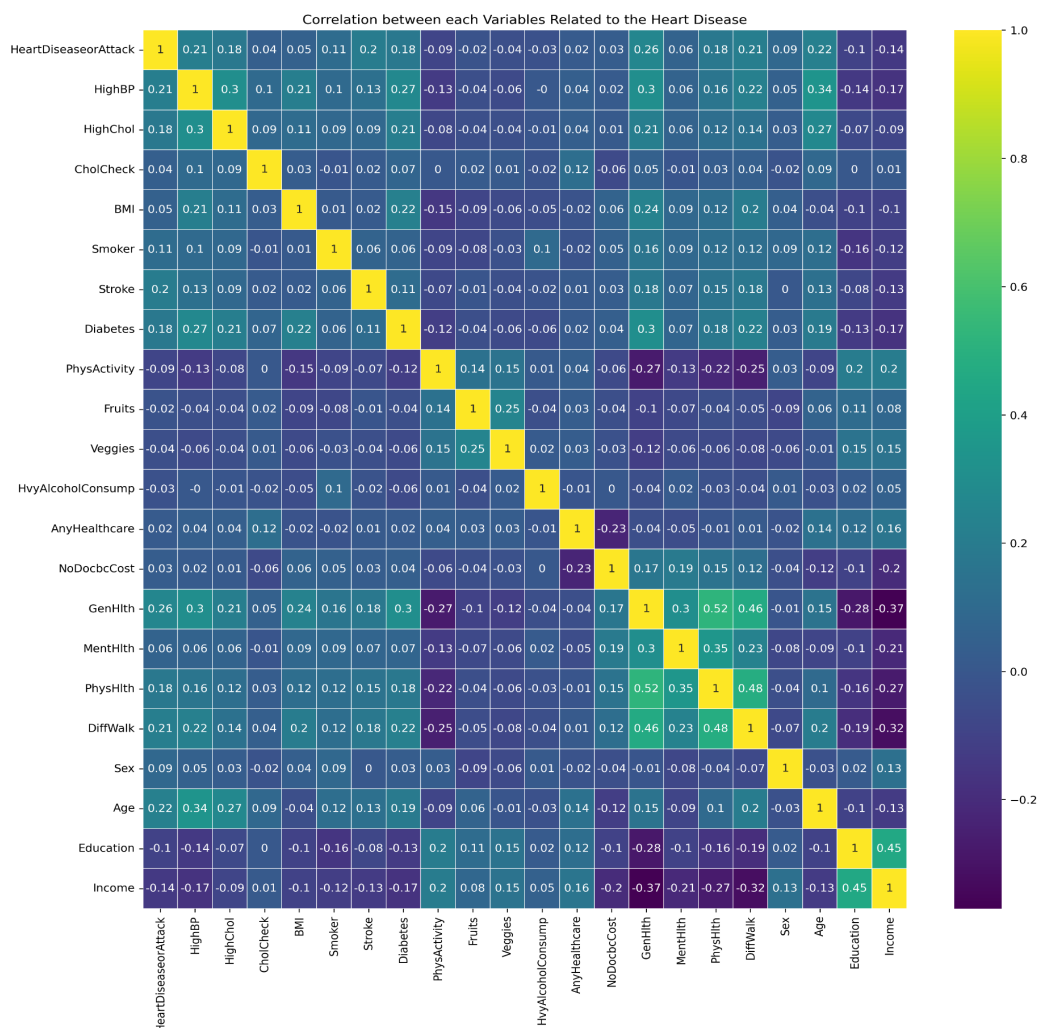


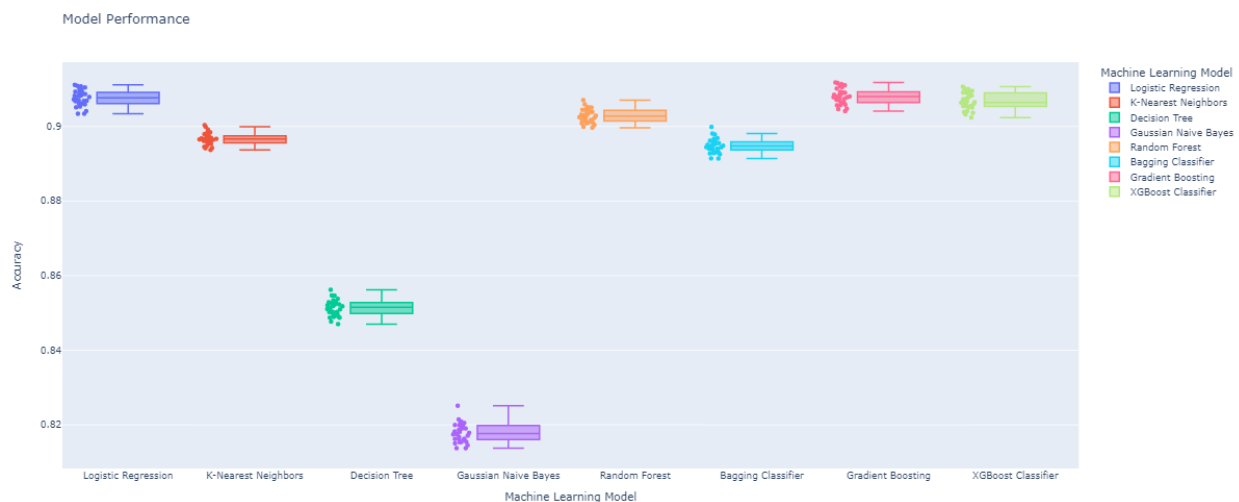
Figure 3: Correlation heatmap

The above heatmap presents the relationships among these features. In detail, we can see that the correlation between the target variable and other features are not very strong with many values that are close to 0. Some relatively large correlation values are between features like “GenHlth” (Adults’ general health condition) and “Age”, which are around 0.2. For other features’ correlations, we have the value between “GenHlth” and “PhysHlth” be the highest, which is 0.52. It makes sense since both of them are related to health conditions.

## 4. Model Training and Selection

### 4.1 Model Selection

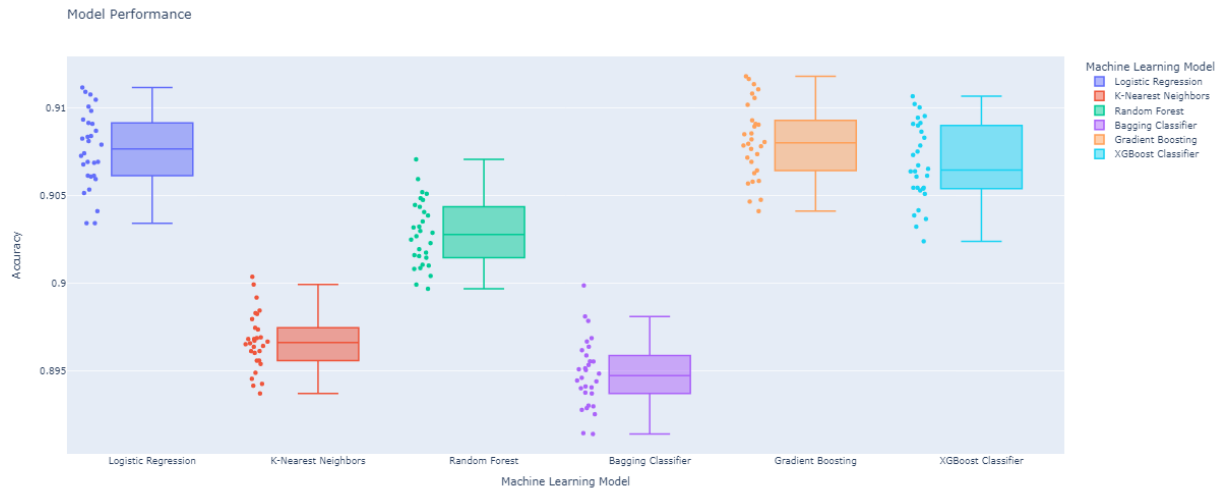
Since this study focuses on a classification problem, we are going to perform the analysis using different machine learning classifiers, including Logistic Regression, K-Nearest Neighbors (KNN) Classifier, Decision Tree, Gaussian Naive Bayes Classifier, Random Forest, Bagging Classifier, Gradient Boosting, and XGBoost Classifier as possible candidate classifiers. In order to evaluate the performance of these models, we leverage 10-fold cross validation and compute the average accuracy scores. The boxplot below demonstrates the performance of all these machine learning models.



*Figure 4: Machine learning model result (all 8 models)*

According to Figure 4, we can see that the Decision Tree and Gaussian Naive Bayes perform the worst, with accuracy scores below 0.86. Thus we remove these two models and check the accuracy scores again.



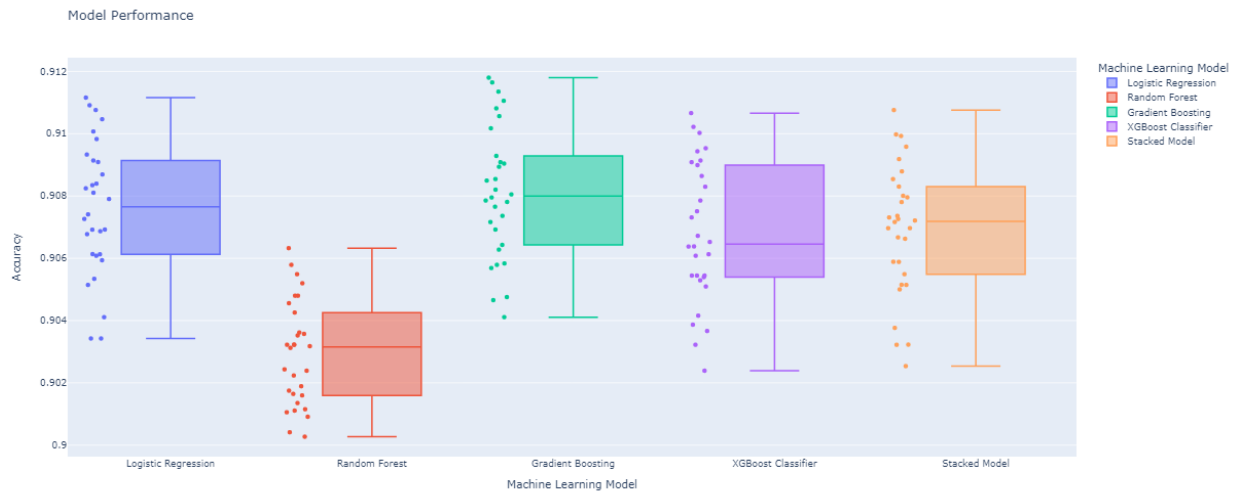


*Figure 5: Machine learning model result (6 models)*

Figure 5 displays the results of these 6 models. Based on the results, Logistic Regression, Gradient Boosting Classifier, and XGBoost Classifier all have accuracy scores above 0.905. We can also notice that Bagging Classifier and K-Nearest Neighbors Classifier perform the worst among these selected models, which can be further removed.

## 4.2 Stacked Model and Model Performance

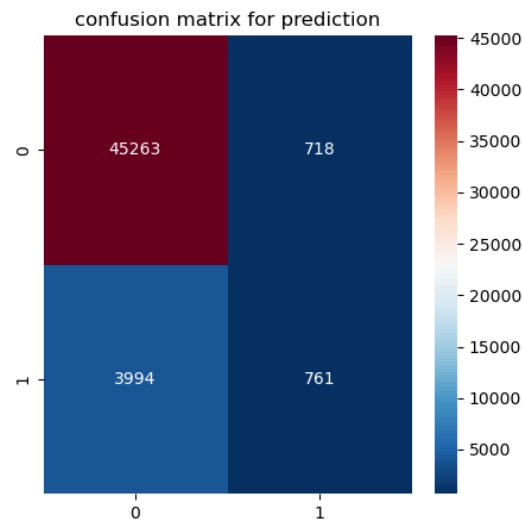
According to the model performance, we choose Random Forest Classifier, Gradient Boosting Classifier and XGBoost Classifier as the candidate models for level 0 in the stacking classification. And we use Logistic Regression as the level 1 model for this stacked classifier. Then we also utilize 10-fold cross validation to evaluate the models.



*Figure 6: Candidate model and stacked model performance*

The above boxplot showcases the performance of the stacked model compared with the selected standalone models. We can see that the average accuracy score for the stacked model is better than Random Forest and XGBoost, but worse than Logistic Regression and Gradient Boosting. Now we will continue to focus on the performance of the stacked model on the test dataset.

We split the whole dataset into the training set and the test set based on an 80-20 ratio, where the test set contains 50736 observations. Then we export the stacked classifier to a pickle model and use this model to predict whether a patient has heart disease on the test data.



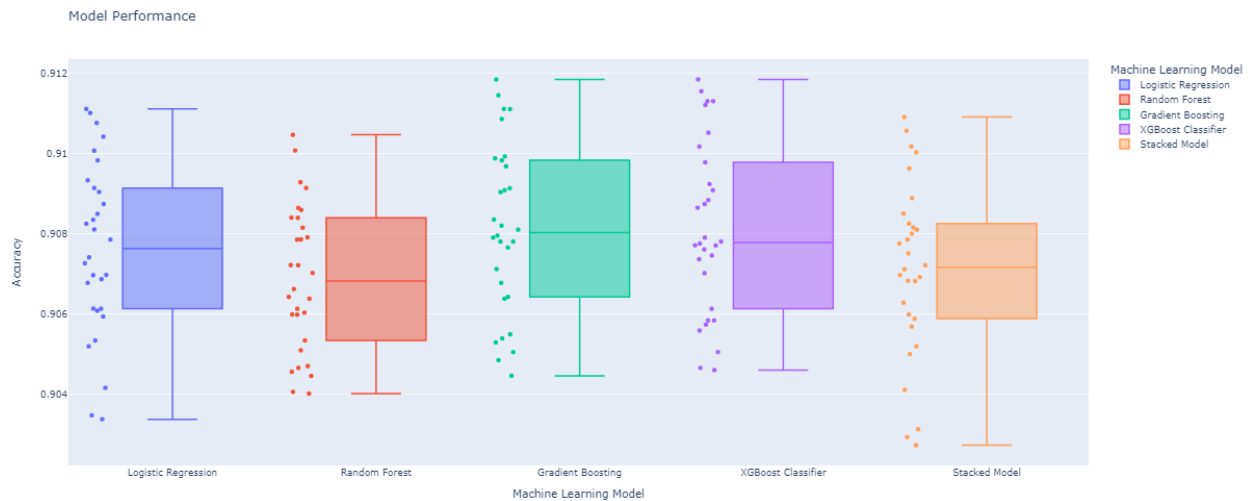
*Figure 7: Confusion matrix of the prediction*

The accuracy of this model on the test dataset is 0.91, which is acceptable. However, according to the confusion matrix above, we can observe that this model struggles with those positive patients, and the recall score is not ideal. The performance of this stacked model can be further improved by hyperparameter tuning, which will be manifested in subsequent sections.

### 4.3 Hyperparameter Tuning

In this part, we mainly focus on optimizing the performance of the model selected above including Logistic Regression, RandomForest, GradientBoosting, and XGBoost. For each classifier, a grid of hyperparameter values is defined, which will be explored to find the best combination of hyperparameters. Then, we perform hyperparameter tuning using GridSearchCV, which exhaustively searches through the defined hyperparameter space. The grid search is performed using cross-validation to ensure robust evaluation of the models. After hyperparameter tuning is complete for all classifiers, we evaluate the performance of the best models on a test dataset and the accuracy increases for all the models. However, when we use the

stack model to stack these models together, the general performance of the stack model does not increase.



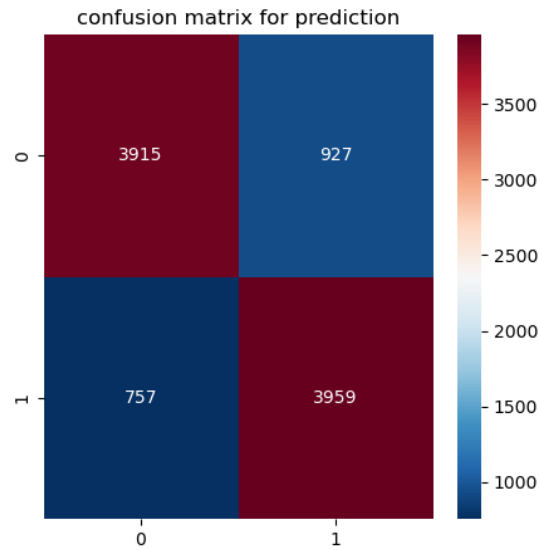
*Figure 8: Candidate model and stacked model performance after hyperparameter tuning*

## 4.4 Resampling

As mentioned in 4.2, we identify a significant challenge in our models' ability to accurately classify positive patients, evidenced by notably low recall values. This is due to a substantial imbalance in the target variable, where the number of negative cases is almost 10 times larger than the positive ones. Additionally, since the goal of these machine learning models is to classify the heart attack or disease problem, where false negative errors are highly costly, it is essential to further increase the recall value and improve the performance for positive patient identification.

To address this, we employ resampling techniques to create a more balanced dataset. There are two primary resampling strategies: oversampling and undersampling. The former increases the number of instances in the minority class by replicating or by using synthetic data generation methods like SMOTE (Synthetic Minority Over-sampling Technique), while the latter decreases the number of the instances in the majority class. Taking into account the computational demands and the size of exported models, we opt for the undersampling method. This approach allows us to equalize the number of instances in the negative class with those in the positive class.

Subsequent to the resampling process, we conduct hyperparameter tuning using GridSearchCV for Logistic Regression, RandomForest, GradientBoosting, and XGBoost models. Also, a stack model is constructed based on all of these four models.

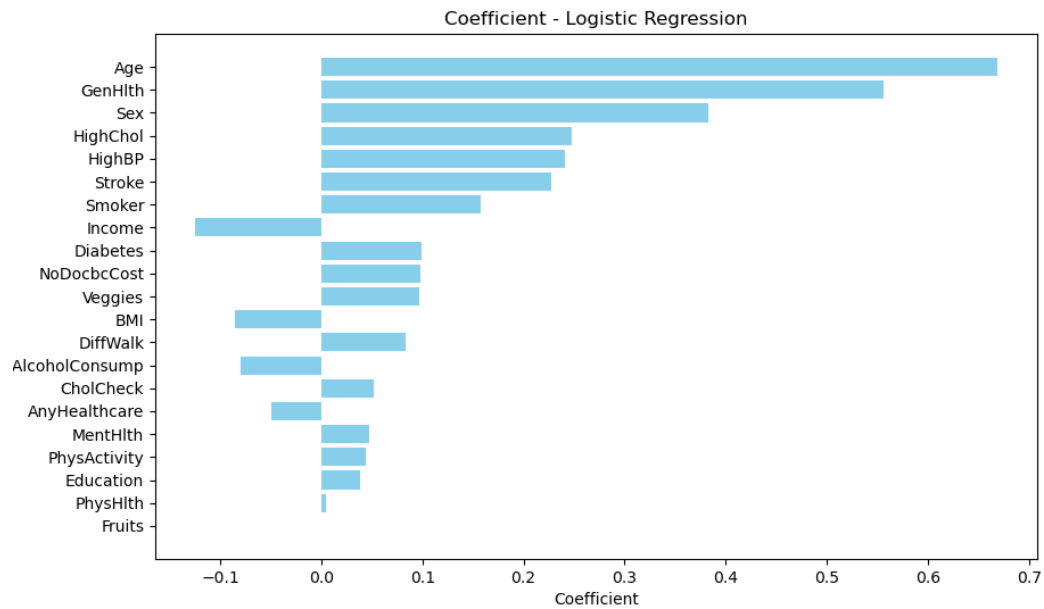


*Figure 9: Confusion matrix of the prediction after undersampling*

The model result, as evidenced by the above confusion matrix, demonstrates a substantial improvement in identifying positive patients. Although there is a decrease in overall accuracy to 0.82, the recall value has impressively surged from 0.16 to 0.84. This improvement is a significant stride in our objective to accurately identify positive cases of heart attack or disease, emphasizing the efficacy of our resampling and modeling approach.

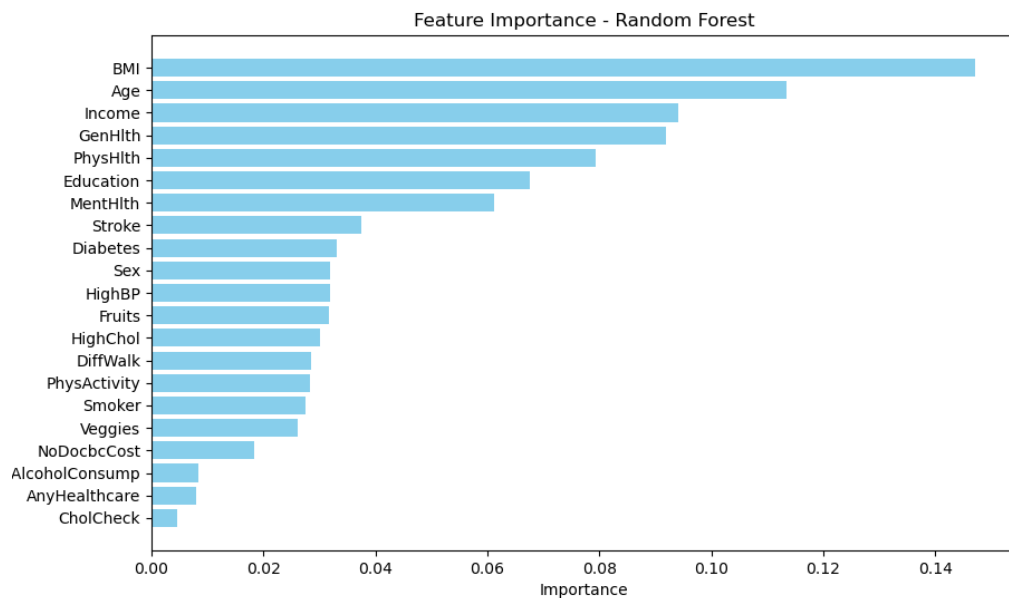
## 4.5 Feature Importance

In our project, we mainly utilize 4 classification models: Logistic Regression, Random Forest, GradientBoosting and XGBoost, and we have combined them to establish a stacked model in the preceding sections. Now we will take a look at the coefficients of Logistic Regression and the feature importance of other three models respectively.



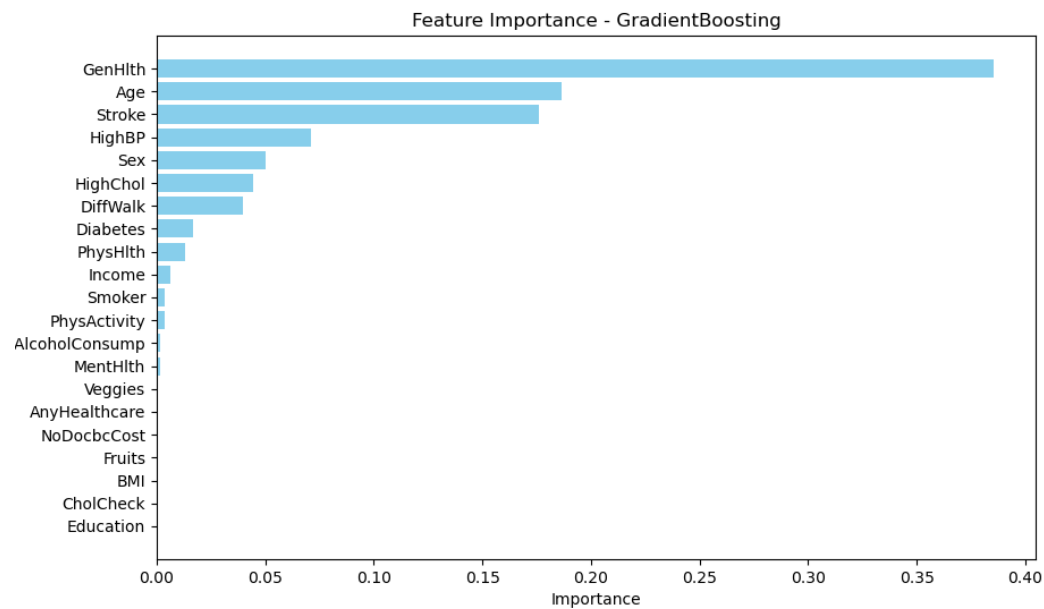
*Figure 10: Coefficient of Logistic Regression*

Age, GenHlth and sex are the features with the top three highest absolute values of Logistic Regression coefficient.



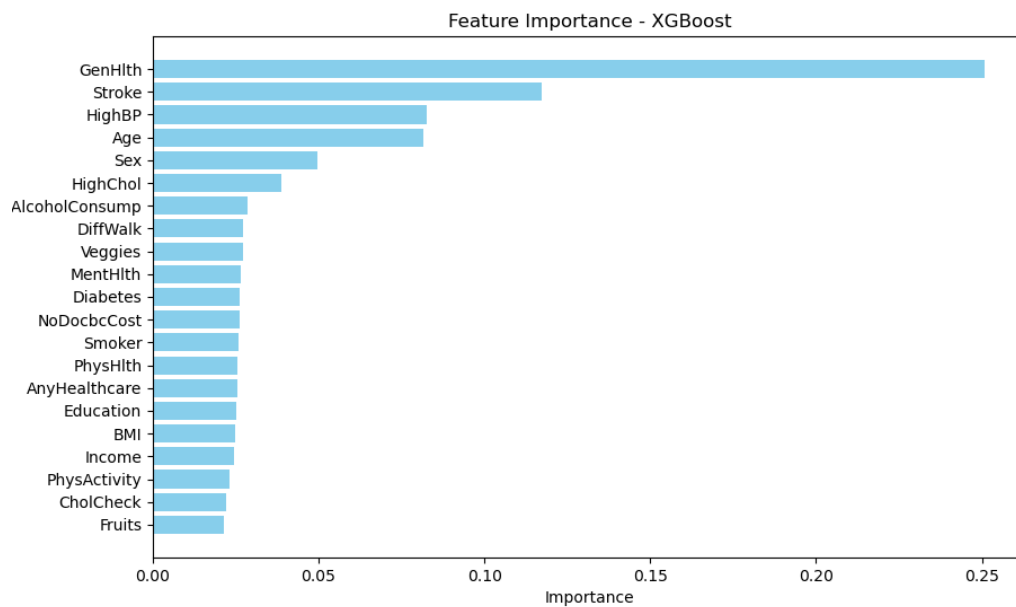
*Figure 11: Feature importance of Random Forest*

The top three features with highest importance based on the RandomForest model are BMI, Age and Income.



*Figure 12: Feature importance of GradientBoosting*

The top three features with highest importance based on the GradientBoosting model are GenHlth, Age and Stroke.



*Figure 13: Feature importance of XGBoost*

The top three features with highest importance based on the XGBoost model are GenHlth, Stroke and HighBP. Actually, the top four features with the highest importance are consistent across

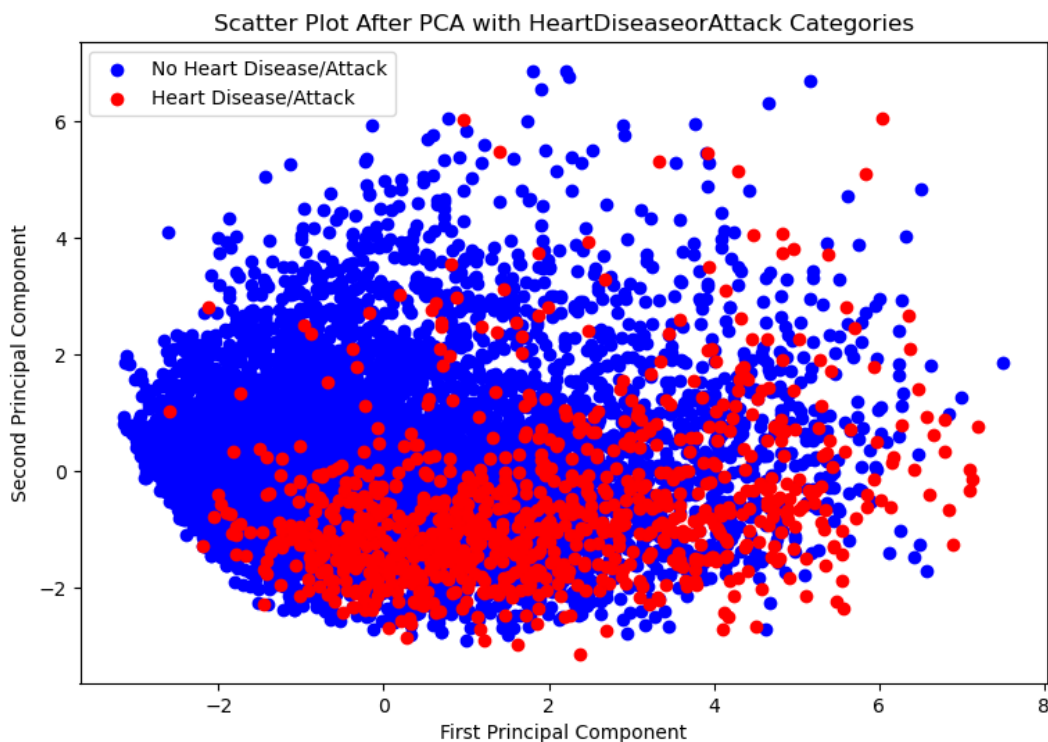
both the Gradient Boosting and XGBoost models, indicating a strong agreement between the two models regarding the most influential predictors in the dataset.

## 5. App Deployment

In this section, we use Flask to deploy our model locally. The local site will print out the message showing the classification result (True or False).

### 5.1 PCA and Visualization

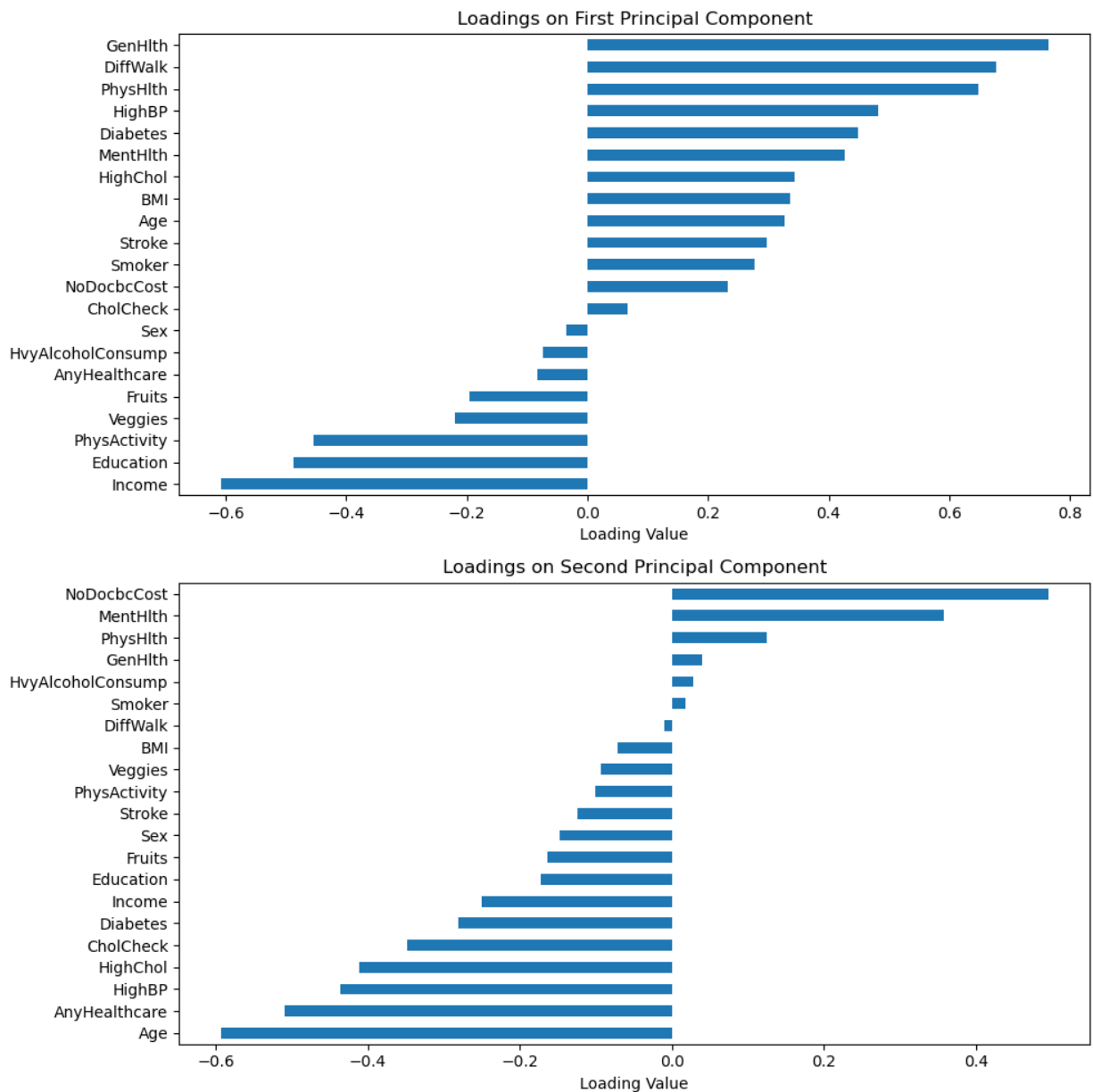
There are 21 independent variables in the dataset. We first apply PCA on this data and choose the first two principal components for visualization.



*Figure 14: Scatter plot based on first two principal components*

This is a scatter plot of a random sample with 10000 records from the whole dataset, where x-axis and y-axis respectively refer to the first and second principal component under PCA analysis. In the visualization, red dots depict patients who have experienced heart disease or a heart attack, whereas blue dots denote individuals without heart problems. According to the plot, although a distinct demarcation between the two groups is absent, there's a noticeable trend

where blue points predominantly gather towards the lower left and middle left, while red points seem to amass more on the lower right side.



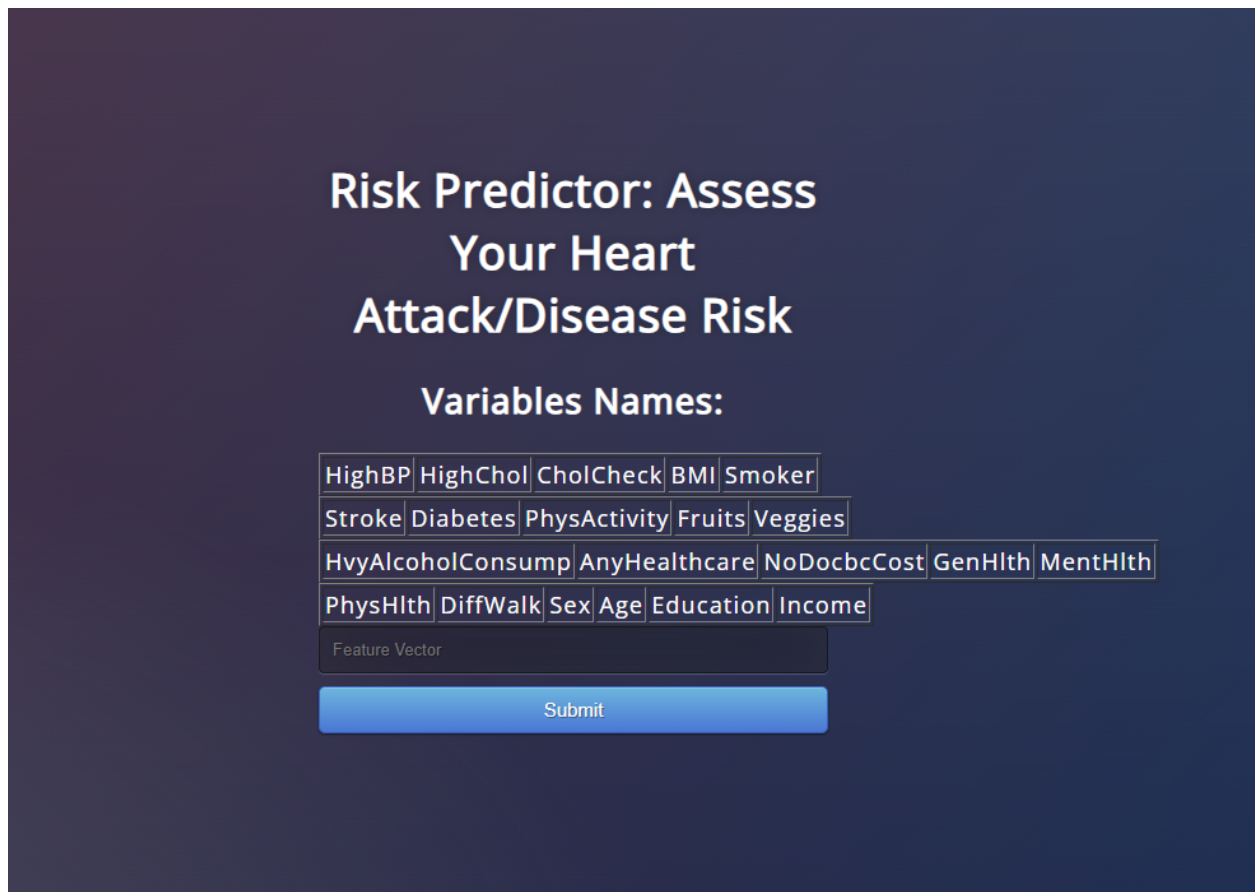
*Figure 15: Loadings on first and second principal components*

According to the plot above, GenHlth and DiffWalk contribute most to the first principal component, while Age and AnyHealthcare are the major contributors to the second principal component. Interestingly, the features significantly influencing the first principal component



exhibit very low absolute loadings on the second principal component and vice versa. This observation highlights a distinct separation in the influence of these features across the two principal components.

## 5.2 Screen



**Risk Predictor: Assess Your Heart Attack/Disease Risk**

**Variables Names:**

HighBP	HighChol	CholCheck	BMI	Smoker	
Stroke	Diabetes	PhysActivity	Fruits	Veggies	
HvyAlcoholConsump	AnyHealthcare	NoDocbcCost	GenHlth	MentHlth	
PhysHlth	DiffWalk	Sex	Age	Education	Income

Feature Vector

Submit

*Figure 16: Web APP*

## Risk Predictor: Assess Your Heart Attack/Disease Risk

**Variables Names:**

HighBP	HighChol	CholCheck	BMI	Smoker	
Stroke	Diabetes	PhysActivity	Fruits	Veggies	
HvyAlcoholConsump	AnyHealthcare	NoDocbcCost	GenHlth	MentHlth	
PhysHlth	DiffWalk	Sex	Age	Education	Income

[1.0,1.0,1.0,47.0,1.0,0.0,0.0,0.0,1.0,1.0,0.0,1.0,1.0,2.0,0.0,1.0,0.0,1.0,12.0,4.0,2.0]

Submit

Heart Attack/Disease Risk Assessment:  
High Risk

*Figure 17: Feature input and result*

The feature vector for this screen is

[1.0,1.0,1.0,47.0,1.0,0.0,0.0,0.0,1.0,1.0,0.0,1.0,1.0,2.0,0.0,1.0,0.0,1.0,12.0,4.0,2.0]. This patient is categorized into a high risk group of patients.

Below are some examples of patients categorized into high risk and low risk of heart disease:

High Risk:

[1.0,1.0,1.0,47.0,1.0,0.0,0.0,0.0,1.0,1.0,0.0,1.0,1.0,2.0,0.0,1.0,0.0,1.0,12.0,4.0,2.0]

[1.0,1.0,1.0,34.0,0.0,0.0,2.0,0.0,1.0,1.0,0.0,1.0,1.0,4.0,0.0,14.0,1.0,0.0,9.0,5.0,1.0]

[1.0,1.0,1.0,45.0,0.0,0.0,2.0,1.0,1.0,0.0,0.0,1.0,1.0,4.0,15.0,10.0,0.0,0.0,8.0,5.0,3.0]

Low Risk:

[0.0,0.0,1.0,22.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,1.0,0.0,1.0,0.0,3.0,0.0,1.0,2.0,4.0,1.0]

[1.0,0.0,1.0,28.0,1.0,0.0,0.0,0.0,1.0,1.0,0.0,1.0,0.0,2.0,0.0,1.0,0.0,1.0,5.0,5.0,8.0]

[1.0,0.0,1.0,28.0,1.0,0.0,0.0,0.0,1.0,1.0,0.0,1.0,0.0,2.0,0.0,1.0,0.0,1.0,5.0,5.0,8.0]

## 6. Conclusion

In conclusion, our project involved the selection and implementation of a stacked classification model to enhance the accuracy of our predictions. After careful evaluation, we chose Random Forest Classifier, Gradient Boosting Classifier, and XGBoost Classifier as the candidate models for the first level of our stacked approach, while utilizing Logistic Regression as the second-level model. The results of our study indicate that this stacked classifier achieved a commendable accuracy of 0.91, however, the recall of this model is pretty low, only around 0.3, which means we cannot successfully predict the positive patients. Even though we try to optimize the model using the hyperparameter tuning, unfortunately, the general performance of the stack model is still not good enough, the recall is still pretty low due to the dataset being very unbalanced. Therefore, we choose to use the method of undersampling to make the dataset more balanced. As a result, the recall improved to 0.84 which is pretty high. Furthermore, an important aspect of our analysis involved identifying the most influential features in our model. From the feature importance analysis, it is evident that BMI, Age, Income, GenHlth, Stroke, and HighBP are the features with the highest importance. Additionally, our principal component analysis (PCA) provided insights into the underlying structure of our data. GenHlth and DiffWalk were identified as primary contributors to the first principal component, while Age and AnyHealthcare played pivotal roles in shaping the second principal component. This knowledge can guide future efforts to focus on these critical variables when designing interventions or refining the model further.