Predictive Analytics for Heart Disease Risk Based on Key Indicators

Ruihan Dou (rd593)

Anqi Li (al2555)

Cornell University

ORIE 5741

Instructor: Haiyun He

GitHub Link: https://github.com/Angela6Li/Predictive-Analytics-for-Heart-Disease-Risk.git

Wednesday, May 8, 2024

1 Introduction

As of 2023, the CDC reports that heart disease was the top cause of death in the US. The severity of heart disease and the possibility of dying quickly if treatment is delayed highlight how crucial early risk assessment is. The CDC has identified several significant contributors to heart disease, including high blood pressure, smoking, diabetes, alcohol drinking, and high BMI. Since there are so many things that might cause heart disease, it is natural for us to ask the question, "How do these different factors affect heart disease?" We also want to know the following: "What model can provide the most accurate prediction?" To address these two questions, we conducted this project.

This project aims to develop a proper model to make accurate predictions about a patient given their medical history and to identify critical indicators contributing to heart disease risk. The model developed in this project can enable healthcare institutions to provide heart disease prevention measures to patients at high risk of heart disease by making accurate predictions. Through early prevention and treatment, not only can the incidence of sudden heart disease be reduced, but government medical expenses can also be saved. What's more, this model empowers insurance companies to correctly price commercial health insurance plans by providing them with a tool to assess the risk of heart disease in policyholders.

This project followed the steps of data collection, processing, building statistical models, and making predictions, all of which were completed in Python. In this course project, we extensively applied the knowledge learned in ORIE 5741. In the first two steps, we applied exploratory data analysis and feature engineering; in the latter two steps, we built classification models, including linear regression, logistic regression, random forest, gradient boosting, SVM, and neural network. We improved model robustness by splitting the data into training and test sets and leveraging regularization.

2 Dataset Description

The dataset is retrieved from Kaggle, "Indicator of Heart Disease (2022 UPDATE)". This dataset initially comes from the Centers for Disease Control and Prevention ("CDC") and is a significant part of the Behavioral Risk Factor Surveillance System ("BRFSS"). Established in 1984, BRFSS collects data in the United States by conducting annual phone surveys. There are 445,132 rows and 37 variables in the dataset. Variable "HadHeartAttack" is our response variable, and we will treat it as binary ("Yes" – respondent had heart disease; "No" - respondent did not have heart disease). Other variables are treated as explanatory variables.

As shown in Figure 1, this dataset contains personal information and several factors that may influence the occurrence of heart disease among the respondents. After examining the data, we decided not to consider some variables in data analysis and modeling, as they are not directly relevant or unclear in the definition. We exclude RemovedTeeth because this is not directly related to the occurrence of heart disease. We exclude HIVTesting because it does not clearly define whether "Yes" and "No" in the variable refer to having taken HIV tests or having tested positive for HIV. We exclude HighRiskLastYear because we have no idea what risks are involved in this variable. Among the rest of the 33 explanatory variables, they are categorized as Exhibit 1 shows.

3 Exploratory Data Analysis

3.1 Data Processing

Since the dataset was collected through a phone survey, it inevitably contains missing values. Additionally, the presence of nominal, Boolean, and ordinal variables makes subsequent predictive modeling difficult. Therefore, we carried out a series of data cleaning and processing steps.

Our approach to handling the dataset was guided by its nature. We understood that missing values in the phone survey data do not provide any useful information. Therefore, we decided to drop all rows with missing values.

3.2 Feature Engineering

For qualitative data, we chose to use one-hot encoding for Boolean variables and real encoding for ordinal variables. This decision was based on the need to ensure these variables are easily interpretable in the model.

In the process of developing our models, transforming continuous variables into categorical ones is a crucial step. This approach helps simplify the relationship between variables and the response, making it easier to interpret and manage within machine learning models. We transformed BMI and sleep hours data into categories and later applied one-hot encoding. Categorizing BMI allows models to treat different risk levels distinctly, improving the model's ability to identify patterns associated with higher risks of heart disease. Therefore, we categorized BMI into Underweight, Severe Thinness, Normal weight, Obese, etc. Then, we apply one-hot encoding to transform these categories into binary variables for each category. This encoding is essential for machine learning models as it removes any ordinal relationship that might be misinterpreted by the algorithms, allowing the model to treat each category as a separate entity without any inherent order.

Similar to BMI, sleep hours are also a continuous variable. However, the risk related to sleep duration is not linear. Both too little and too much sleep can indicate health issues. Categorizing sleep hours and applying one-hot encoding improve the model's interpretation of these non-linear relationships.

Besides, we calculated the heart disease rate among respondents in each state and replaced the State variable with the HeartAttackStateLevel variable.

3.3 Questions in Data Analysis

During the exploratory data analysis stage, to understand the individual impact of each variable on the heart disease prevalence rate and facilitate subsequent modeling, we proposed several sub-questions and presented our findings accordingly.

Firstly, given that the data was collected from across the United States, we raise the first sub-question: **Does the probability of having heart disease vary based on the state of residence?** From Figure 2, we find that among all respondents, the states with the highest probability of having a history of heart disease are West Virginia (WV) and Arkansas (AR), which is generally consistent with the data posted on ValuePenguin (Black, 2022). Thus, the probability does vary across different states.

Moreover, we wondered **if the birth sex affects the probability of having heart disease**. From Figure 3, we know that among all respondents, 4% of females have had heart disease, which is significantly lower than the 7% observed in their male counterparts. Thus, we conclude that males are more susceptible to heart disease.

Furthermore, we are curious about **if age affects the probability of having heart disease**. Figure 4 shows that as age increases, the proportion of individuals with heart disease gradually rises.

In terms of other variables, we provide a summary here: angina, stroke, skin cancer, COPD, arthritis, kidney disease, alcohol drinking, lack of physical activity, smoking history, and diabetes history tend to significantly influence the probability of developing heart disease.

4 Model Training

4.1 Approach

We applied various machine learning methods discussed in the class to address the above two research questions. First, we identified the most significant features using the correlation plot, VIF, AIC/BIC, and linear regression analysis. Besides, we incorporated interaction and squared terms of numerical variables to enhance the model's complexity and uncover non-linear relationships.

We explored several machine-learning algorithms based on the training and test set to check our hypotheses. Candidates of algorithms include linear regression, diverse combinations of loss functions and regularization techniques, decision trees, ensemble methods derived from decision trees, and clustering approaches such as K-means combined with Principal Component Analysis.

Lastly, we compared the outcomes produced by different models. We assessed each model's performance and analyzed the reasons behind the differences. This provided insights into each model's predictive capabilities and highlighted their practical implications in real-world scenarios.

4.2 Exploration of Different Models

4.2.1 Linear Regression

We employed a correlation plot (Figure 8) and the Variance Inflation Factor (VIF) to find multicollinearity between explanatory variables prior to executing the linear regression model. In order to minimize multicollinearity and model complexity, we eliminated or merged strongly

correlated variables. We used PCA to remove one principle component from two explanatory variables whose VIF was above 10, bringing all of the variables' VIF below 10.

Next, we used the linear regression model to determine the selection of significant features. This step was based on the variables' P-values, followed by other feature selection methods such as AIC and BIC. We incorporated five most significant interaction terms (products of pairs of variables) and squared terms of numerical variables in regression to identify subtle patterns that might not be visible from initial findings. We obtained a strong set of explanatory factors with good predictive power by repeating this procedure.

We employed the standard Linear Regression Model (quadratic loss without regularizer) following the train-test split. Due to its sensitivity to outliers and tendency to overfit, the model is generally not the best for classification issues. Still, it helps us understand the model's basic predictive capability.

After running the linear regression model, the continuous prediction outputs are first converted into binary classifications using a threshold of 0.5, where predictions equal to or above this threshold are labeled as 1 and those below as 0. This translation allows us to determine the model's correctness in a classification context, giving us a clear indicator of how effectively the model distinguishes between the two classes. According to the dataset, only 5% of the respondents had a heart attack (y = 1). Therefore, we utilized the balanced error rate. In this way, the performance of both classes is fairly evaluated, and it avoided majority class from overshadowing the minority class when using the accuracy on the test set. **For quadratic loss with no regularizer, the test accuracy is 0.948865 and the balanced error rate is 0.416934.**

However, this setting gives us a very high balanced error rate and poor predictive performance in the minority group (with 2,698 incorrect predictions versus only 565 correct ones). To achieve a more balanced training dataset and improve model performance, especially for the minority class, we modified our training dataset by both downsampling the majority class (no heart disease) and upsampling the minority class (had heart disease) to each consist of 60,000 samples. This process balances the influence of each class during model training and reduces the bias towards the majority class. Besides, we evaluated the model performance using the original imbalanced test set to ensure that any improvements in model performance can be accurately assessed in an unbiased testing environment. **After the changes, for quadratic loss with no regularizer, the test accuracy is 0.853883, and the balanced error rate is 0.200055.**

4.2.2 Different Loss and Regularizers

We experimented with various loss functions and regularization techniques to enhance the model's performance. We first introduced an L1 regularizer that could output sparse solutions in the model parameters and highlight essential features in our quadratic loss model. This feature is valuable because it could provide fewer but more significant factors. **For quadratic loss with L1 regularizer and alpha = 0.1, the test accuracy is 0.940851, and the balanced error rate is 0.262343.** Following this, we explored the effects of an L2 regularizer, which helps shrink the

coefficients, prevent overfitting, and address multicollinearity. **For quadratic loss with L2 regularizer and alpha = 1, the test accuracy is 0.853883, and the balanced error rate is 0.200055.**

With the insights gained from quadratic loss models, we continued testing other loss functions more traditionally suited for classification tasks. We started with Hinge Loss, often used in Support Vector Machines (SVMs) with slackness. **For Hinge loss with no regularizer, the test accuracy is 0.569348, and the balanced error rate is 0.364286.** The low accuracy indicates potential underfitting or sensitivity to class imbalance. Given the limitation that scikit-learn does not support Hinge Loss with an L1 regularizer, we applied L2 regularization with Hinge Loss, which continued to show promise by preventing overfitting and maintaining a robust margin on the classification boundary. **For Hinge loss with an L2 regularizer, the test accuracy is 0.867850, and the balanced error rate is 0.204670.**

Lastly, we employed Logistic Loss, which is inherently suited for probabilistic frameworks and is commonly used in logistic regression. **For Logistic loss with no regularizer, the test accuracy is 0.844566, and the balanced error rate is 0.200064.** We enhanced this model with L1 and L2 regularizations to observe differences in performance. **For Logistic loss with L1 regularizer and c = 0.01, the test accuracy is 0.847320, and the balanced error rate is 0.199621. For Logistic loss with L2 regularizer and c = 0.01, the test accuracy is 0.844994, and the balanced error rate is 0.199693.** We can see that the logistic regression has satisfactory performance since we get high accuracy and balanced error rates that are minimally affected by the inclusion and type of regularization used.

4.2.3 Decision Tree and Ensemble Methods

We also tried decision tree classifiers, which uncover non-linear relationships. Starting with a basic decision tree model, we utilized grid search for hyperparameters fine-tuning. **For the basic decision tree, the test accuracy is 0.8242995, and the balanced error rate is 0.219300.** As both the accuracy and balanced error rate are not very good, we want to utilize ensemble methods with hyperparameter fine-tuning, which combine the predictions of several weak learners to improve the predictive power.

First, we implemented the Random Forest, which aggregates multiple simple decision trees to improve prediction accuracy and reduce overfitting. **For Random Forest, the test accuracy is 0.880250, and the balanced error rate is 0.211990.**

Second, we implemented Gradient Boosting. We effectively reduce both bias and variance by sequentially building trees on residuals. **For Gradient Boosting, the test accuracy is 0.835414, and the balanced error rate is 0.196085.**

Thirdly, we implemented Bagging (Bootstrap Aggregating), which aggregates predictions from multiple decision trees using sub-sampling. The benefit of Bagging is that it can increase stability

and accuracy by reducing variance without significantly increasing bias. **For Bagging, the test accuracy is 0.884438, and the balanced error rate is 0.260359.**

4.2.4 PCA and K-means

Here, we combined PCA with K-means clustering, a method suitable for pattern recognition and data reduction in large datasets. PCA reduces dimensionality and enhances K-means performance by focusing on the most significant features and minimizing noise. K-means clustering partitions observations into k clusters in which each observation belongs to the cluster with the nearest mean. K-means is helpful for its simplicity and efficiency in processing large datasets quickly.

We applied PCA to our feature set to extract ten principal components, which were then combined with the explanatory variables. After balancing the dataset, we trained the K-means model. **For PCA and K means, the test accuracy is 0.855614, and the balanced error rate is 0.405803.**

The result is not as good as other models because of the limitations of K-means. It assumes clusters are spherical and often of similar size, which may not be accurate for our datasets. What's more, our dataset has a high dimension. K-means clustering may fail due to the "curse of dimensionality."

5 Model Comparison and Conclusion

5.1 Model Comparison

After completing the evaluation of various machine learning models, we observed differences in performance metrics. Please refer to Exhibit 2 and Figure 5 in the Appendix for the summary.

Among all the models, Quadratic Loss (L1 Regularizer) has the highest test accuracy. But it also has a higher test balanced error rate, indicating potential overfitting to the majority class. It has lower false positives, but higher false negatives compared to no regularization. Quadratic Loss with L2 has similar results as without regularizer, due to the coefficients are already small.

For Hinge Loss without Regularizer, the low accuracy indicates potential underfitting or sensitivity to class imbalance. Hinge Loss with L2 Regularizer has much better test accuracy and balanced error rate, enhancing margin maintenance while preventing overfitting.

Logistic Loss is consistently strong across all variants (no regularizer, L1, L2) and it also has a good balance between accuracy and balanced accuracy. It has similar false positive and true positive rates among different regularization methods.

The simple decision tree exhibits overfitting as it has poor test balanced accuracy and comparatively high train accuracy. Random Forest has good train and test accuracy, but also has a high test-balanced error rate. Although bagging demonstrated the best train accuracy, its test-balanced accuracy was mediocre. Gradient Boosting offered a better balance but did not excel in test accuracy.

The PCA and K-means model demonstrates good accuracy but an extremely high balanced error rate in the test set, indicating that it only accurately predicts the majority class (no heart disease). Then, this model is not helpful as the test set itself is imbalanced.

5.2 Feature Importance Analysis

From the Random Forest model based on the processed data (shown in Figure 6), the most influential feature in predicting the likelihood of a heart attack is Interaction_4, which is a product of HadAngina and PC1(the principal component from GeneralHealth and HeartAttackStateLevel), with a relative importance of approximately 8.1%. Other significant features include PC1, PC1^2, HadAngina, Interaction_2(product of ChestScan and HadAngina), and PhysicalHealthDays^2.

We also ran Random Forest model on original data to detect the feature importances of single variables (shown in Figure 7). HadAngina, GeneralHealth, PhysicalHealthDays, HadStroke, ChestScan, and MentalHealthDays are the most significant features. Knowing the features' importance can help individuals identify risks earlier, help healthcare providers tailor interventions more effectively, and provide directions for model refinement and validation.

Meanwhile, note that the significant contributors to heart disease that the CDC identified, such as high blood pressure, smoking, diabetes, alcohol drinking, and high BMI, do not appear as the top important features. The top features identified are directly linked to clinical symptoms and general wellness indicators. For instance, "HadAngina" tops the list because angina is a direct symptom of coronary artery disease, which is a precursor to heart attacks.

5.3 Model Summary

According to our experiments, Logistic Regression with L1 Regularizer, Random Forest, and Gradient Boosting have the best performance in different aspects when predicting whether a person is at high risk of having heart disease. When the priorities of the prediction task are interpretability and generalization, Logistic Regression with L1 Regularizer is recommended. It can provide a clear decision boundary and feature selection through regularization. The ability to capture non-linear relationships in Random Forest and Gradient Boosting makes them more suitable for situations requiring higher balanced accuracy and when dealing with complex data patterns and feature interactions.

From previous parts, our dataset is highly unbalanced, with a significantly higher proportion of cases of non-heart disease than cases of heart disease. Therefore, it is essential to find a model that provides a stable and reliable performance while considering this imbalance. With this in mind, we select Gradient Boosting as our final model.

Gradient Boosting has a favorable balance between test accuracy and balanced error rate, which makes it successful in this imbalanced dataset even if it does not have the best test accuracy. As Gradient Boosting handles complicated, imbalanced datasets well, we are confident in the robustness

and applicability of our results.

This model's advantage in reducing error rates, especially in the minority class, shows its potential to provide reliable predictions in real-world applications. Therefore, we are willing to see that healthcare institutions and insurance companies can use this model in production. This model's strong prediction power, especially in the imbalanced scenarios, gives these companies a powerful tool to better assess people's risk of having heart disease to improve the decision-making process.

5.4 Future Improvement

In the future, we can do a more comprehensive grid search across a broader range of hyperparameters. In addition, by looking further into the correlations between the variables, feature engineering may yield even more informative features. Moreover, a larger data set will yield a better model performance, particularly from the minority class, if we have greater processing power. The model's bias towards the majority class can also be mitigated by using a larger dataset.

Furthermore, the significant characteristics derived by predictive modeling are not the same as the risk factors that the CDC has historically highlighted. In the future, we could preprocess our dataset to reduce the impact of co-occurring conditions that might mask the effects of primary risk factors. This can help better align our model with known heart disease predictors. As an alternative, to increase relevance to more general public health issues, we can find a dataset that is explicitly about these conventional risk factors.

5.5 Weapon of Math Destruction and Fairness

The concept of Weapon of Math Destruction is invented by Cathy O'Neil in her book *Weapon of Math Destruction*. The author introduced several negative impacts on social fairness caused by poorly designed mathematical models. Due to the model's failure to consider false positives or false negatives, the model may exhibit unfairness and harm the interests of certain groups, such as the poor. In her book, the author also suggests that one should avoid using information such as race and residence to replace individual information with group information. It is true that our model does incorporate the respondent's state of residence, race, gender, and other personal information into the prediction. However, based on our answers to the sub-questions in Section 3.3 and calculations when exploring the dataset, gender, state of residence, and race do indeed influence the risk of heart disease. Therefore, we believe that our model does not create a weapon of math destruction.

Due to the limitations of computing power and the imbalanced dataset, we resampled from 80% of the original dataset to generate a balanced training set and tested the model performance on the imbalanced test set (20% of the original dataset). According to our training approach, we did not consider the information of all respondents, which may lead to some unfairness in the model. However, the use of a balanced error rate when measuring model performance aligns with the concept of Equalized Odds. Besides, as I mentioned above, the model doesn't penalize individuals based on their race, gender, or location. Thus, we believe our model is fair.

Contribution

Ruihan Dou is responsible for Section 1 Introduction, Section 2 Dataset Description, Section 3 Exploratory Data Analysis (code + report + presentation), and Section 5.5 Weapon of Math Destruction and Fairness (report). Ruihan Dou is also responsible for the overall polishing of the report.

Anqi Li is responsible for Section 4 Model Training and Section 5.1 – 5.4 (code + report + presentation). Anqi Li is also responsible for building and maintaining the Github repository.

The overall contribution of Ruihan Dou and Anqi Li to this report is 50% and 50%, respectively.

Appendix

State: The U.S. state where the individual resides.
Sex: Gender of the individual (Male or Female).
GeneralHealth: Self-reported general health status of the individual.
PhysicalHealthDays: Number of days in the past 30 days that physical health was not good.
MentalHealthDays: Number of days in the past 30 days that mental health was not good.
LastCheckupTime: Time since the last routine checkup or health examination.
PhysicalActivities: Frequency of engaging in physical activities or exercises.
SleepHours: Average number of hours of sleep per night.
RemovedTeeth: Number of permanent teeth removed due to dental issues.
HadHeartAttack: Whether the individual has had a heart attack.
HadAngina: Whether the individual has experienced angina (chest pain or discomfort).
HadStroke: Whether the individual has had a stroke.
HadAsthma: Whether the individual has had asthma.
HadSkinCancer: Whether the individual has had skin cancer.
HadCOPD: Whether the individual has had Chronic Obstructive Pulmonary Disease (COPD).
HadDepressiveDisorder: Whether the individual has had a depressive disorder.
HadKidneyDisease: Whether the individual has had kidney disease.
HadArthritis: Whether the individual has had arthritis.
HadDiabetes: Whether the individual has had diabetes.
DeafOrHardOfHearing: Whether the individual is deaf or hard of hearing.
BlindOrVisionDifficulty: Whether the individual has blindness or vision difficulty.
DifficultyConcentrating: Self-reported difficulty in concentrating.
DifficultyWalking: Self-reported difficulty in walking.
DifficultyDressingBathing: Self-reported difficulty in dressing or bathing.
DifficultyErrands: Self-reported difficulty in running errands.
SmokerStatus: Current smoking status (smoker, former smoker, non-smoker).
ECigaretteUsage: Whether the individual uses e-cigarettes.
ChestScan: Whether the individual has had a chest scan.
RaceEthnicityCategory: Categorized race or ethnicity of the individual.
AgeCategory: Categorized age group of the individual.
HeightInMeters: Height of the individual in meters.
WeightInKilograms: Weight of the individual in kilograms.
BMI: Body Mass Index calculated from height and weight.
AlcoholDrinkers: Whether the individual consumes alcohol.
HIVTesting: Whether the individual has undergone HIV testing.
HighRiskLastYear: Whether the individual has been considered at high risk for the past year.
CovidPos: Whether the individual tested positive for COVID-19. ═══分节符(连续)═══

Figure 1: Dataset's Features

Variable Type	Count
Numerical	6
Nominal	3
Ordinal	5
Boolean	19
Total	33

Exhibit 1: Type Counts of All Variables

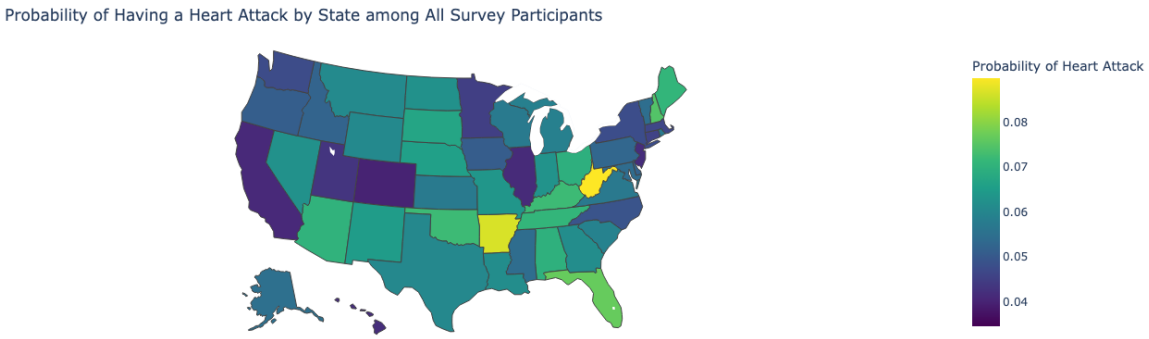

Figure 2: Probability of Having a Heart Attack by State among All Survey Participants

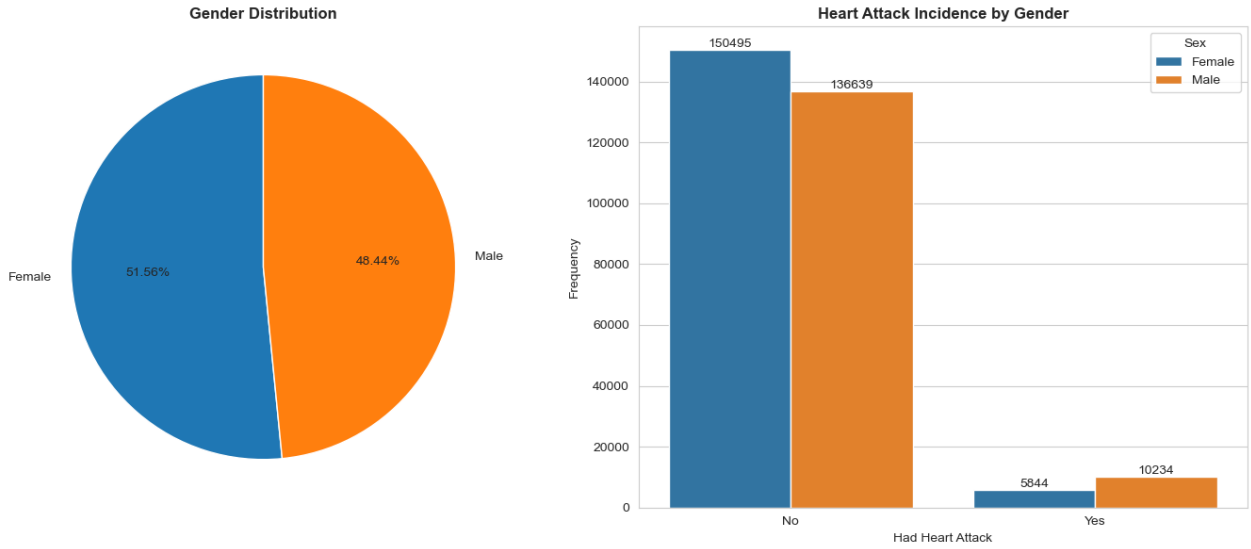

Figure 3: Gender Distribution (Left) and Probability of Having a Heart Attack by Gender (Right)

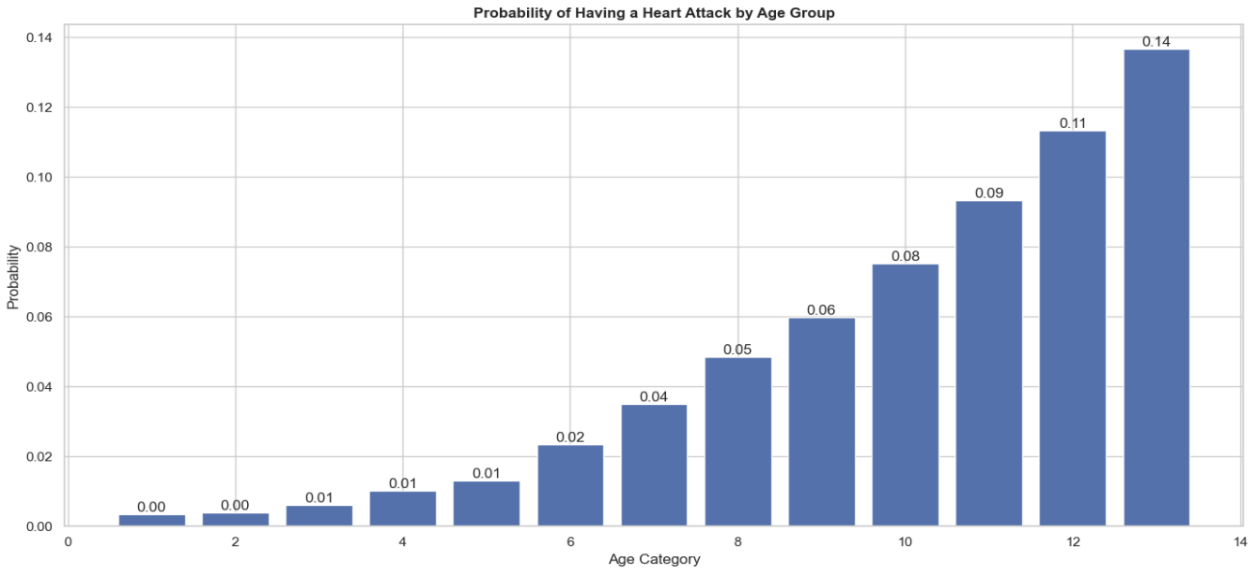

Figure 4: Probability of Having a Heart Attack by Age Group

	Train Accuracy	Test Accuracy	Train Balanced Error Rate	Test Balanced Error Rate	Test Confusion Matrix
Quadratic loss, no regularizer	0.795992	0.853883	0.204008	0.200055	[[49369 8011] [850 2413]]
Quadratic loss, L1 regularizer	0.733300	0.940851	0.266700	0.262343	[[55392 1988] [1599 1664]]
Quadratic loss, L2 regularizer	0.795992	0.853883	0.204008	0.200055	[[49369 8011] [850 2413]]
Hinge loss, no regularizer	0.639733	0.569348	0.360267	0.364286	[[32210 25170] [946 2317]]
Hinge loss, L2 regularizer	0.791475	0.867850	0.208525	0.204669	[[50299 7081] [933 2330]]
Logistic loss, no regularizer	0.797233	0.844566	0.202767	0.200064	[[48770 8610] [816 2447]]
Logistic loss, L1 regularizer	0.797225	0.847320	0.202775	0.199621	[[48944 8436] [823 2440]]
Logistic loss, L2 regularizer	0.797358	0.844994	0.202642	0.199693	[[48795 8585] [815 2448]]
Basic Decision Tree	0.808575	0.824300	0.191425	0.219300	[[47600 9780] [875 2388]]
Random Forest	0.888283	0.880250	0.111717	0.211990	[[51147 6233] [1029 2234]]
Gradient Boosting	0.817417	0.835414	0.182583	0.196085	[[48154 9226] [755 2508]]
Bagging	0.979275	0.884438	0.020725	0.260359	[[51751 5629] [1379 1884]]
PCA+K means	0.501358	0.855614	0.498642	0.405803	[[50904 6476] [2280 983]]

Exhibit 2: Summary Table of All Different Models

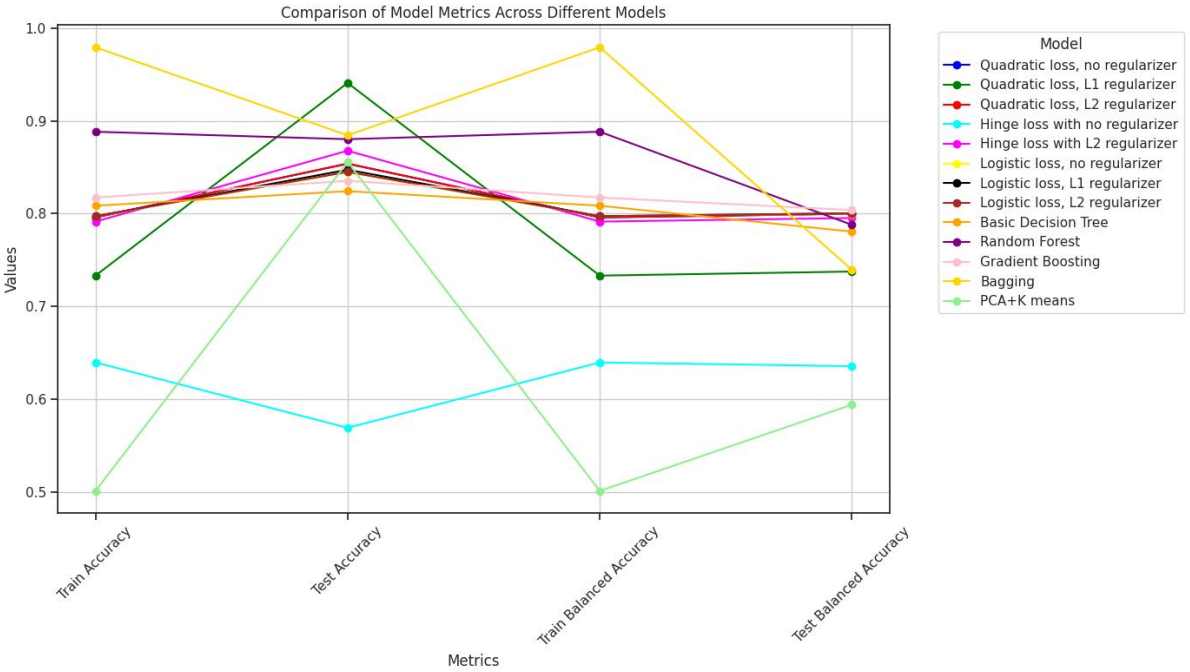

Figure 5: Comparison of Model Metrics Across Different Models

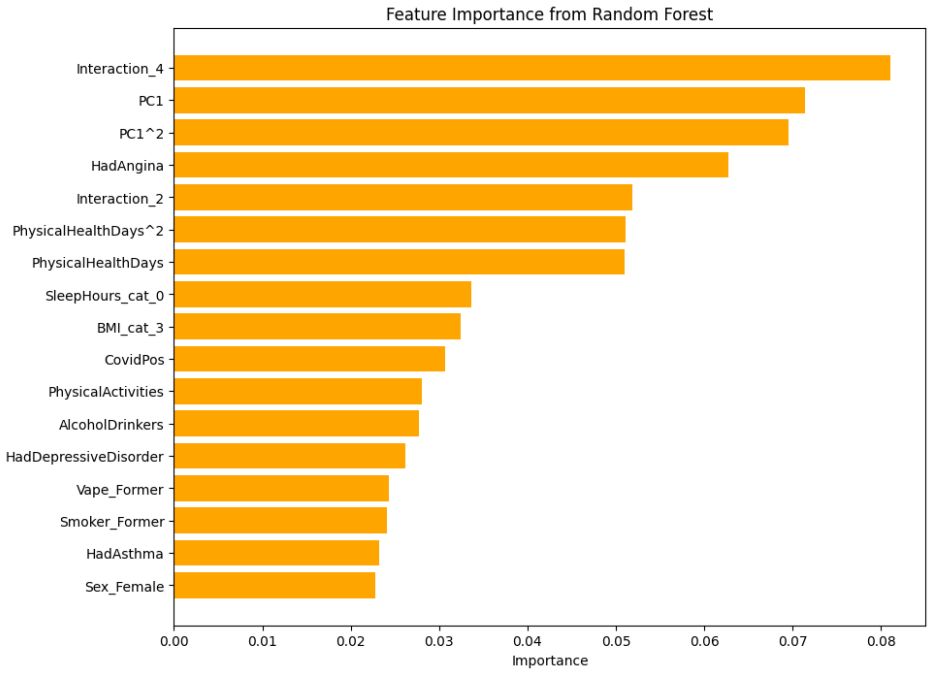

Figure 6: Feature Importance from Random Forest (Using Processed Data)

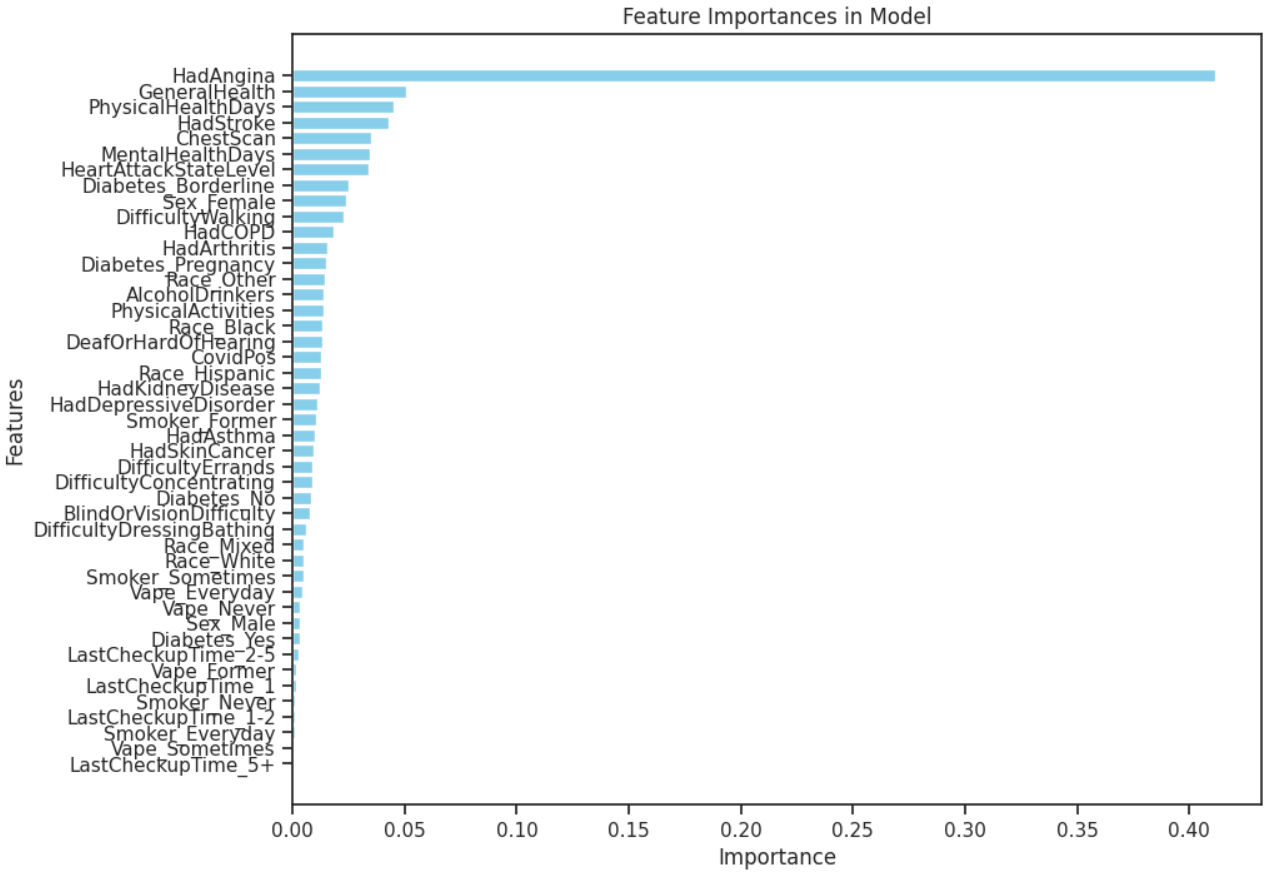

Figure 7: Feature Importance from Random Forest (Using Original Data)

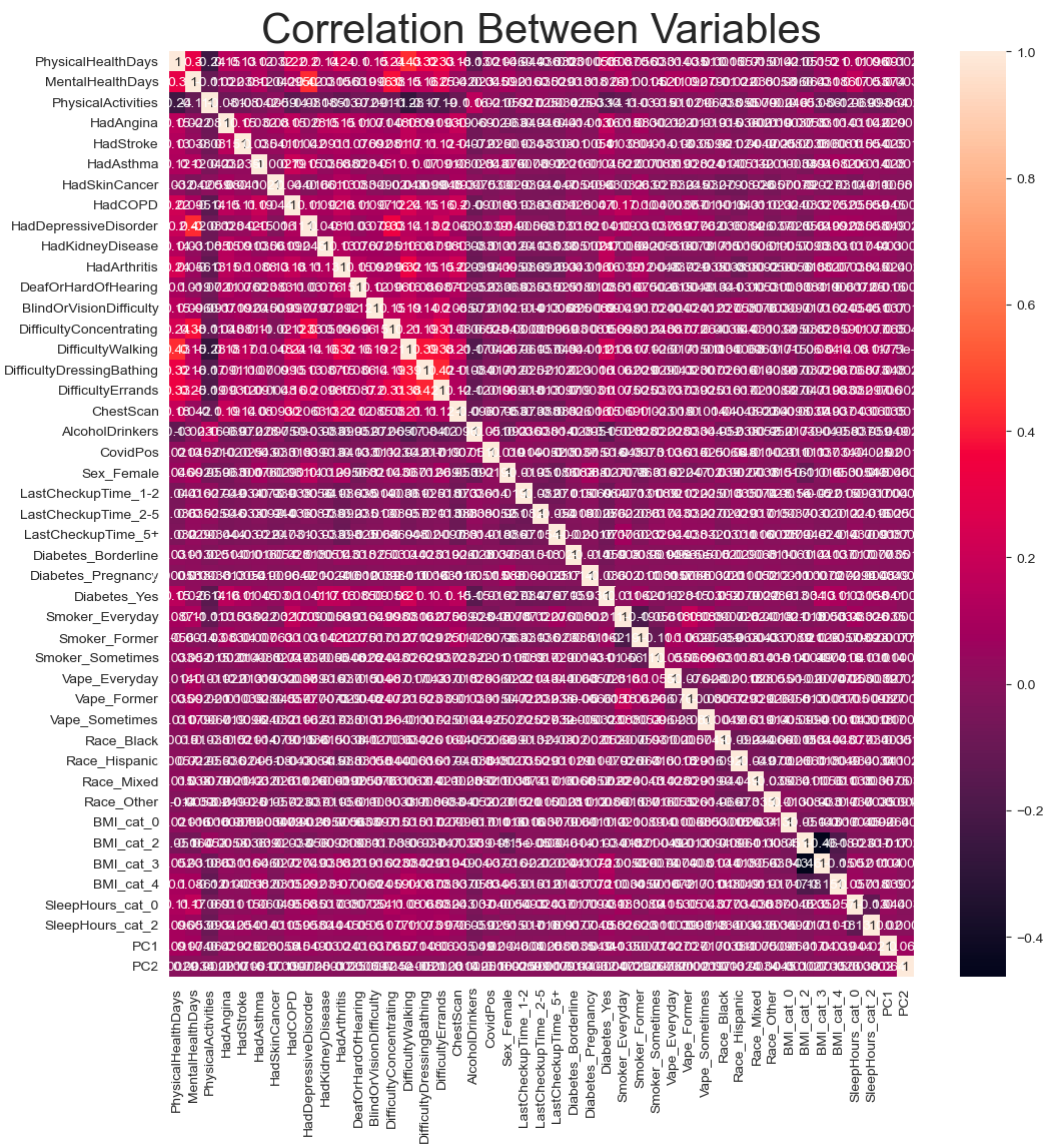

Figure 8: Correlation Plot of Features After Treatment

Reference

Black, M. L. (2022). Most heart-healthy states in the U.S. ValuePenguin. Retrieved from https://www.valuepenguin.com/heart-healthy-states-study

Dijkinga, F. (2024). Explaining L1 and L2 Regularization in Machine Learning. Medium. Retrieved from https://medium.com/@fernando.dijkinga/explaining-l1-and-l2-regularization-in-machine-learning-2356ee91c8e3

Towards Data Science. (2020). Interaction Effect in Multiple Regression. Retrieved from https://towardsdatascience.com/interaction-effect-in-multiple-regression-3091a5d0fadd