36-662 Final Project

Prediction for Heart Disease

May 1, 2023

Group Name: 2_duck

Yiting Wang (yitingwa@andrew.cmu.edu)

Yicheng Wang (yicheng6@andrew.cmu.edu)

1. Introduction

In recent years, heart diseases have become an increasingly pressing concern in the world that affect the health situation of adults, especially the elderly. According to the research conducted by Joseph, J.J. et al. published in American Heart Association's journals, "Cardiovascular disease is multifactorial, and control of the cardiovascular risk factors leads to substantial reductions in cardiovascular events". In this report, we focus on analyzing how the heart disease status is related to 11 collected demographic and biomedical features. By exploring the binary prediction model with various machine learning algorithms, we aim to gain deep understanding about how heart disease is related to some part of or all of these factors, which may provide some valuable insights about early detection and management of cardiovascular diseases.

2. Exploration

This dataset contains a total of 735 patients in the row and 12 predictor variables in the column. There is no missing value in this dataset. As we can see from Table 1, `HeartDisease` is a binary response variable about whether each patient has heart disease or not. These two classes have relatively balanced samples as we check in Figure 1's bottom right bar plot, where the difference in the amount of samples between two classes is less than 10% of the total sample. In this regard, there is no need to perform downsampling or upsampling.

Looking into the independent variables, we firstly plot the boxplot to examine the outlier for each numerical variable. From Figure 1's rightmost column, we can see that `RestingBP`, `Cholesterol` and `Oldpeak` have substantial outliers, which would impact the performance of our model, say, to be biased or overfit due to the outliers. In this regard, it is necessary to transform these variables to reduce the impact of the outliers in feature engineering, as well as using robust modeling techniques that are less sensitive to outliers, such as decision trees, random forest and support vector machines.

To investigate the relationship between numerical variables and `HeartDisease` further, we can see the correlation matrix in the upper triangle of the pair plot (Figure 1). It is noticeable that `Age` and `MaxHR` are the pair that has the comparably the most significant negative correlation, with a value of -0.368, followed by `Age` & `Oldpeak`, `MaxHR` & `Cholesterol`, and `Age` & `RestingBP`. Therefore, it may be worth considering variable selection techniques to identify which variable is more important for the model, such as regularization techniques like Lasso or Ridge regression, or principal component analysis (PCA) to identify linear combinations of variables that explain the most variation in the data. Also, we can add some interaction variables for these highly correlated pairs. In addition to the correlation, we are able to find the boxplots for two labels that don't overlap in Figure 1's rightmost column, given `Age` and `MaxHR`. This implies that there is a significant difference between groups for these two variables, so that we'd at least keep one of them in the feature selection.

While we get some common sense about numerical variables, we are also interested in exploring categorical variables more. We primarily plot the distribution of samples in each categorical variable respectively (Figure 2), in order to investigate the balance for each category. We can observe that all six categorical variables are imbalanced, for example, `FastingBS`, `Sex`, and `down` in `ST_Slope` have the most explicit imbalance distribution where the sample in one category is much lower than others. Such imbalances are highly likely to affect the performance of the models as well. Therefore, it may be beneficial to balance the classes by using techniques such as oversampling, undersampling, or synthetic data generation, and to prevent using models that are sensitive to class imbalance such as logistic regression or decision trees.

Variable Name	Data Type	Description
Age	Numerical	Age of the patient
Sex	Categorical	Sex of the patient: ● M: Male ● F: Female
ChestPainType	Categorical	Chest pain type: ● TA: Typical Angina ● ATA: Atypical Angina ● NAP: Non-Anginal Pain ● ASY: Asymptomatic
RestingBP	Numerical	Resting blood pressure in mmHg
Cholesterol	Numerical	Serum cholesterol in mm/dl
FastingBS	Categorical	Fasting blood sugar: ● 1: if FastingBS > 120 mg/dl ● 0: otherwise
RestingECG	Categorical	Resting electrocardiogram results: ● Normal: Normal ● ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV) ● LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria
MaxHR	Numerical	Maximum heart rate achieved in the range between 60 and 202
ExerciseAngina	Categorical	exercise-induced angina: ● Y: Yes ● N: No
Oldpeak	Numerical	ST, Numeric value measured in depression
ST_Slope	Categorical	The slope of the peak exercise ST segment: ● Up: upsloping ● Flat: flat ● Down: downsloping
HeartDisease	Categorical	Output class: ● 1: heart disease ● 0: Normal

Table 1: Variable Descriptions

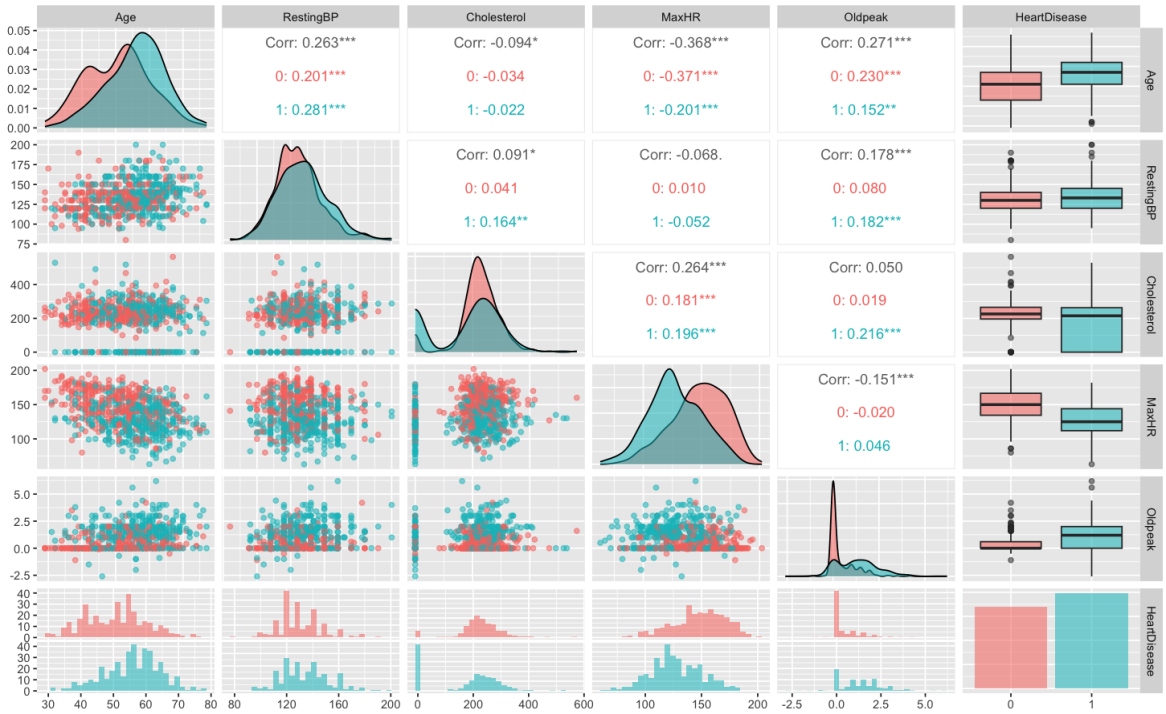

Figure 1: Correlation between pairs of numerical variables and `HeartDisease`.

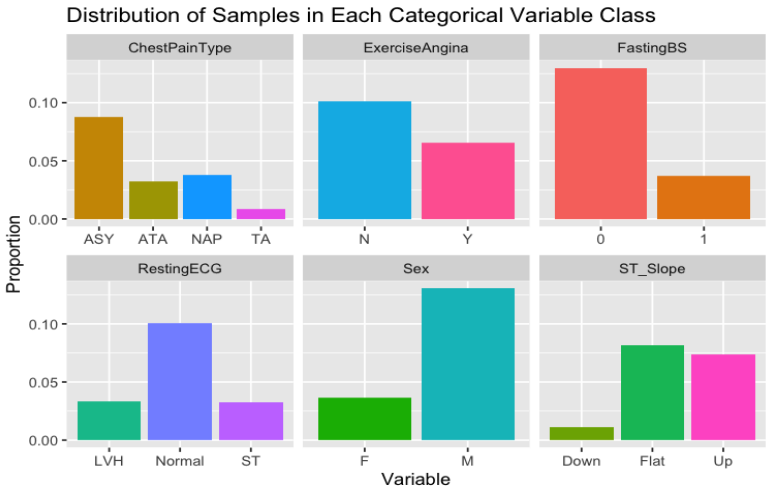

Figure 2: Most categorical variables have imbalance samples.

3. Supervised Analysis

As our goal is to build the best classification model to predict the heart disease for patients, we would like to perform some feature engineering based on preliminary exploratory data analysis findings, to compare three supervised learning algorithms, and finally to make predictions.

Firstly, we transform three variables that have explicit outliers, `RestingBP`, `Cholesterol` and `Oldpeak`, by taking log since they are right skewed. In order to capture potential non-linear relationships among variables, the top four correlated pairs of variables are added as the interaction terms, where the absolute correlation values are larger than 0.25. To mitigate the imbalance property of the categorical variables, we applied synthetic data generation by substituting "`ATA`" and "`TA`" in `ChectPainType` as "`AP`" which means Anginal Pain. In this way, this new variable would become more representative with more samples, resulting in a more balanced class distribution in `ChectPainType`. However, for the sake of preserving the most realistic information, we don't upsample or downsample other variables at this time, unless they bias our model too much by bringing us the erroneous predictive result. Moreover, we apply one-hot encoding for all categorical variables, making up a total of 23 predictor variables while we scale them as well to increase the stability and accuracy of our later modeling.

To start with the modeling, we split 80% of the dataset into a training set and 20% of that into a test set. As our response variable is binary, we would like to try the simple model first, logistic regression, which can tell us the predicted probability for the discrete response variable about whether the patient has heart disease or not, by mixing and matching discrete and continuous features to make discrete classification. In order to prevent overfitting and provide a more generalized model, we use Lasso to perform variable selection to select the most important variables. In this way, the bias increases and variance decreases. Here, we tune the parameter `lambda`, which determines the power of shrinkage, by using 10-fold cross-validation to pick the `lambda` that minimizes the mean cross-validation error. This regularization approach leaves us with 17 variables with coefficients shown in Figure 3. We can infer that as `age`, `oldpeak`, `male`, `higher fastingBS`, `ExerciseAngina` and `asymptomatic` chest pain type increase by 1 unit, the estimated odds that Y=1 would increase by e^beta respectively (i.e. beta means coefficient), holding other variables constant. All other variables would negatively related the heart disease, meaning less likely to have heart disease (i.e. Y=0). By setting the decision boundary as 0.5, we finally obtain the accuracy of 0.8776, which is the percentage of samples being correctly identified as having HeartDisease or not. Notice that the decision boundary here is tuned by iterating from 0.1 to 0.9 and getting the one that could result in the best prediction accuracy.

```
Call:  glm(formula = HeartDisease ~ ., family = "binomial", data = lasso_xtrain)

Coefficients:
          (Intercept)                   Age               Oldpeak       inter.maxhr_age        inter.rest_age                 Sex.M
               0.2066                0.5689                0.5271               -0.2522               -0.1087                0.7009
     FastingBS.Higher       FastingBS.Lower       RestingECG.LVH      ExerciseAngina.Y         ST_Slope.Down           ST_Slope.Up
               0.6333                    NA               -0.1614                0.3641               -0.4259               -1.2503
 new_ChestPainType.AP  new_ChestPainType.ASY  new_ChestPainType.NAP
              -0.2242                0.7790                    NA
```

Figure 3: Logistic Regression Model Coefficients

It is acknowledged that logistic regression doesn't work well with large feature space. Other than using Lasso to select variables, we also try Principal Component Analysis (PCA) to reduce the dimension of variables and fit the logistic regression with selected PCs. Figure 4 plots the percentage of variation explained by each principal component in decreasing order. We find the "elbow" point in the plot where the bar starts to level off is in 14th PC. This indicates 14 principal components could capture the most variance in the data. Then, we use these 14 PCs to fit the logistic regression, leading to the accuracy of 0.8503.

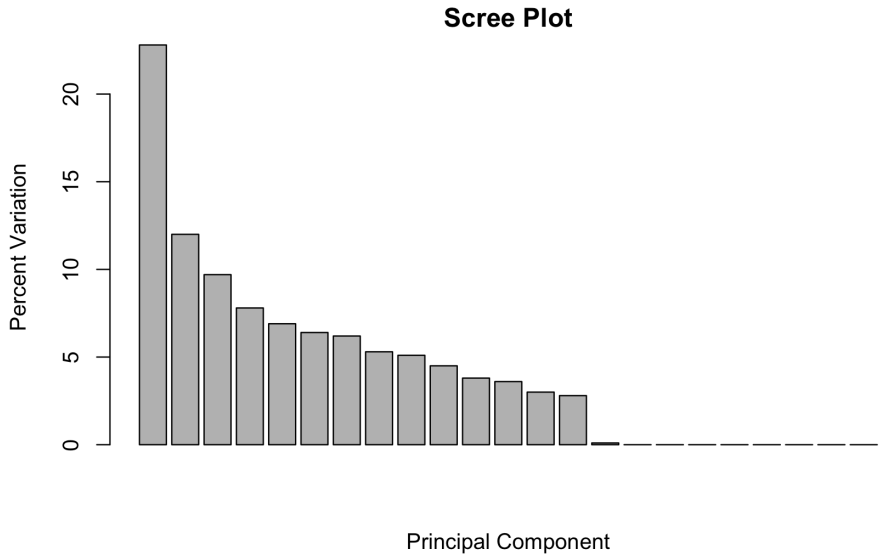

Figure 4: Top 14 principal components capture the most information of the data.

Although we did some feature engineering to mitigate the effect of outliers and imbalanced data, in Figure 3, we can see some variables with large amounts of samples, such as `ASY` and `Male`, have very abnormal large effects on the response variable. This corroborates the property that logistic regression is sensitive to the balance of data and would lead to overfitting, even though we performed Lasso regularization to generalize the data. Apart from logistic regression, random forest uses bagging, which is a type of ensemble method where multiple models are trained on different subsamples of the training data and their predictions are combined to produce the final prediction. Such a random subsampling could effectively reduce the impact of imbalanced data, could handle outliers and work well with non-linear relationships. These benefits align quite well with our dataset so that we are interested in training this model. We tune `ntree` in the `randomForest` function in R that specifies the number of trees to grow in the random forest. It controls the trade-off between model accuracy and computation time. Increasing the number of trees can improve the accuracy of the model, but it also increases the computation time. So we try a range of values for `ntree` from 100 to 1500, and select `ntree=230` that gives the best performance on the validation set, which returns the accuracy of 0.8844.

Support Vector Machine (SVM) is also a very handy model to handle outliers in the dataset with light assumption on the data distribution, since it allows more misclassification rate, which in turn increases the bias, in order to decrease variance. We use Radial Kernel to systematically find Support Vector Classifiers in higher dimensions. We create a grid of values for the `C` and `gamma` parameters of the SVM, and then use the `tune()` function to search over this grid using 5-fold cross-validation. The `svm()` function from the `e1071` package is used as the model to be tuned, and the ranges argument specifies the parameter grid to search over. The `tunecontrol` argument specifies the cross-validation settings. The best model is obtained from the `tune`() output using the `$best.model` attribute. Specifically, `C` is a regularization parameter that controls the misclassification of training examples by allowing the creation of a soft margin, where we find the best SVM model has `C=0.003`. It balances the desire for the decision boundary to fit the data well while also being generalized to new data. Besides, `gamma` scales how much influence neighboring points have on classification, where we find the best SVM model has `gamma=0.001`. With these hyperparameters being determined, this model leads to an accuracy of 0.8639.

4. Discussion

4.1 Lasso Logistic Regression and PCA Logistic Regression

In this part, we explored two techniques - PCA and Lasso regularization - to reduce the dimension of our data or perform feature selection. The benefit of doing this before building a logistic model is due to the fact that logistic regression doesn't work well with large feature space. By applying them respectively to the original dataset and building a logistic regression model on transformed or selected variables, we found out that the accuracy by using Lasso is higher than that by using PCA, obtaining 0.8776 by Lasso and 0.8503 by PCA. As the original performance with featured data without Lasso or PCA is 0.8707, Lasso seems to improve the performance a little bit, while PCA performs worse than the original. Since PCA is a linear method and assumes that the relationship between variables is linear, we might extrapolate that there are some important nonlinear relationships that PCA may not capture. On the other hand, the improvement of accuracy by lasso logistic regression implies that our preliminary transformation to mitigate numerical variables' outliers' impact works well, since logistic regression is sensitive to outliers if we don't remove outliers. However, we notice that there are still some categorical variables' coefficients being abnormally large due to its large sample size on that typical class, this suggest that we didn't do enough feature engineering to handle imbalance categorical variables, which would mislead the results by bringing biased or overfitting result.

4.2 Random Forest Performs the Best Prediction

The prediction accuracy of the random forest is the best that has accuracy around 0.8844. We believe that there are several explanations for the superior performance of the random forest.

Firstly, it can help to handle high-dimensional data, which is suited as our data contain 23 features, so that there is no need to do dimensional reduction. Secondly, it can decide the importance of variables and determine the interaction between them. Thirdly, this algorithm allows to balance the error for unbalanced variables, which is pretty needed for our data set as we didn't handle the imbalance categorical data to preserve the information. The only concern for this model is that random forest only decreases the variance without decreasing the bias, it is likely to cause bias to fit the predicted data.

For improvement to this algorithm specifically, it would be beneficial to perform other ensemble methods like boosting which combine the outputs of multiple weak learners into a single strong learner. Boosting is effective at reducing both bias and variance errors, while bagging is mainly useful in reducing variance.

4.3 Support Vector Machine

Generally speaking, the strength of SVM is to mitigate the negative effects of outliers. But it seems like we effectively removed the outlier, this algorithm performs not that outstanding with the accuracy of 0.8639. Moreover, it is helpful to handle interactions of non-linear features which are not captured by PCA, so that it obtains an accuracy that is higher than the PCA logistic regression model. In the perspective of bias-variance tradeoff, SVM allows for certain amounts of misclassification, which is controlled by the hyperparameter C, we think although the accuracy is not that outstanding, it is likely to achieve better generalization by decreasing the variance, at the expense of increasing the bias. However, in order to make sure this is a valid inference, we are willing to perform 5-fold cross-validation in the future to increase the credibility of this well-generalized model.

4.4 Future Development

To address all limitations we have identified, it is always important to consider the bias-variance tradeoff. For a model with high bias, it means that the model pays limited or too much attention to the training model and may lead to underfitting or overfitting. In contrast, for a model with high variance, it means that the model pays lots of attention to the training model, which may lead to overfitting and poor generalization to test data. Both high variance or bias will account for the high error rates of the prediction model. Thus, we can adopt three main strategies that could apply to all algorithms to improve our unexpected prediction results.

The first strategy we are considering is to implement both ridge and lasso regularization to the training dataset. This combined approach is more effective compared to solely using lasso regularization. By adding the ridge method, it will introduce a penalty term to the cost function of our model, which can help to shrink the coefficients of features towards zero. Through the

combined usage of ridge of lasso regularization, we are able to make our model more robust with more generalization. While lasso regularization can help to identify and remove less important features that may be considered as noise to our model, ridge regularization can help to reduce the impact of multicollinearity and stabilize our model by preventing overfitting,

The second strategy that might be helpful is cross validation, which is a powerful technique that can help to check and compare models. By applying a 5-cross validation, we can separate data into 5 subsets, while using one of them for testing and others for training. Parallelly comparing the model accuracy by cross-validation, we are likely to get a more stable and trustworthy accuracy. For our random-forest model, this is an essential step to avoid overfitting and improve the accuracy of generalization performance.

Last but not least, we need to discuss with stakeholders to decide whether we can apply synthetic data generation to any other variables in order to balance the whole dataset, or if it is appropriate to perform upsampling or downsampling, since making variables balanced and eliminating outliers would always be our priority to build a high performance model that could work well on prediction.

Reference

Joseph, J. J., Deedwania, P., Acharya, T., Aguilar, D., Bhatt, D. L., Chyun, D. A., Di Palo, K. E., Golden, S. H., Sperling, L. S., & American Heart Association Diabetes Committee of the Council on Lifestyle and Cardiometabolic Health; Council on Arteriosclerosis, Thrombosis and Vascular Biology; Council on Clinical Cardiology; and Council on Hypertension (2022). Comprehensive Management of Cardiovascular Risk Factors for Adults With Type 2 Diabetes: A Scientific Statement From the American Heart Association. Circulation, 145(9), e722–e759. https://doi.org/10.1161/CIR.0000000000001040