Team RLA - Final Project Report

Anjali Ohri (903463903), Richie Phan (903651897), Lance Wilhelm (903597440)

March 26, 2022

Author Roles: Anjali generated the problem statement and all three authors were involved in researching usable data sources. Richie and Lance conducted an in depth exploration of the data with modeling and visualization for the final report. Anjali refined the document and generated the conclusion.

1 Problem Statement

According to the Centers for Disease Control and Prevention (CDC), heart disease is the leading cause of death for men and women in the United States [1]. Each year 805,000 Americans have a heart attack and 655,000 Americans die from heart disease, which accounts for 1 in every 4 total deaths. For 605,000 of those having a heart attack, it is their first time [1]. Many risk factors influence the likelihood a person will have a heart attack, such as: age, sex, medical history, the presence of other medical conditions, and lifestyle factors. By assigning specific risk factors as variables, a classification model can be developed to determine if an individual has a risk of developing heart disease. This information can be utilized in the clinical setting to determine the need for rapid evaluation of an individual experiencing chest pain based on their risk profile. First, exploratory analysis of the risk factors could be used to understand the correlation between risk factors and each feature's impact on the classification results. Then a properly trained logistic regression model could classify patients as either having a low or high risk of a heart attack. The resulting classification model developed with a supervised learning method would allow a medical professional to then predict the likelihood that chest pain in an individual is a result of underlying heart disease. With a proper classification of at-risk individuals, preventative measures such as dietary changes, lifestyle changes, and exercise can be prescribed to reduce the risk of a heart attack.

2 Data Source

For this project, we used a data set taken from Kaggle.com, a repository for published data. The main difficulty encountered was finding a data set with a substantial number of data points that also included a label. We were able to find the Cardiovascular Study Dataset which was taken from an ongoing cardiovascular study on residents of Framingham, Massachusetts, and and also publicly available on Kaggle [2]. The Cardiovascular Study Dataset includes 4000 data points (patients) with 14 features and 1 label column. The data set has been split into a train and test set. The columns include `Sex`, `Age`, `is_smoking` (Yes or No), `Cigs Per Day` (average number of cigarettes smoked per day), `BP meds` (whether on blood pressure medication), `Prevalent Stroke` (whether a patient had a previous stroke), `Prevalent Hyp` (whether a patient is hypertensive), `Diabetes` (whether a patient has diabetes), `Tot Chol` (cholesterol level), `Sys BP` (systolic blood pressure), `Dia BP` (diastolic blood pressure), `BMI` (body mass index), `Heart Rate`, and `Glucose` (glucose level). The training dataset has labels in the column called `TenYearCHD` that indicate whether or not the individual has a 10 year risk for heart disease. The test set does not include a label column, thereby reducing our usable number of datapoints.

Before the data can be used it must be cleaned to account for missing values and categorical values that may

cause issues when fitting the models later. The training data set contained missing values for `education`, `cigsPerDay`, `BPMeds`, `totChol`, `BMI`, `heartRate`, and `glucose`. Upon observation of the distribution of each data point, we decided to replace each missing value with the average value for that given attribute and for the respective sex of the data point. Lastly, we also plotted each predictor variable via box plots to visually observe any outliers that stray far away from their respective groupings.

3 Exploratory Data Analysis

Exploratory data analysis was done to determine if patterns exist in any of the features. First we review the general characteristics of the data.

Figure 1 gives us an understanding of the proportion of actual cases of coronary heart disease risk that are present in the data set. We see that actual cases of risk account for $\sim 1/7$ of the data. It will be important to remember that we have a relatively small proportion of positive risk cases to work with.

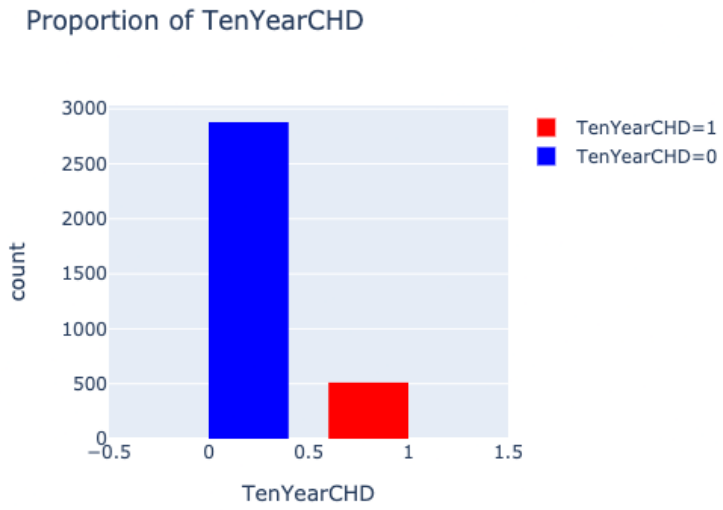


Figure 1: Proportion of CHD risk in data. 1 = positive CHD risk

Figures 2 and 3 outlines the distribution of data points by sex overall and also by presence or absence of ten year heart disease risk.

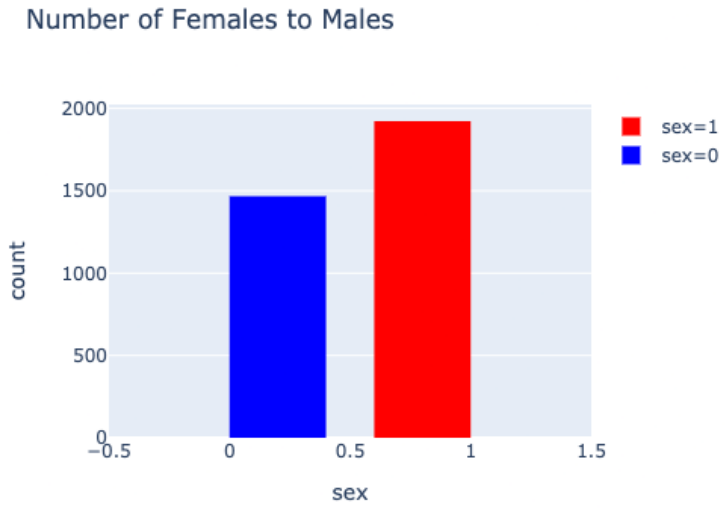


Figure 2: Proportion of sex in data. 1 = female

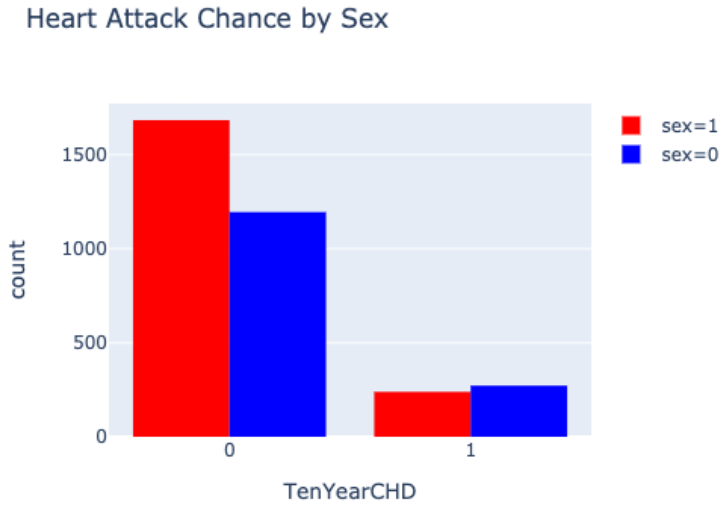


Figure 3: CHD risk by sex. Sex = 1 indicates female. `TenYearCHD` = 1 indicates positive CHD risk

Box plots were used on the features to visualize any potential outliers in the data set, see Figure 4. There seems to be possible outlier in totChol and glucose. Each feature was further analyzed by splitting them by CHD classification. The box plot range is similar, with the interquartile range for CHD = 1 being slightly higher, which is to be expected. Interestingly enough, the totChol outlier (see Figure 5) that is around 700 is from the CHD = 0 classifier. This is interesting because high cholesterol is usually considered an indicator for CHD, but since we have such a large sample of CHD = 0 in our data, we may see a larger range outliers

for CHD = 0 than CHD = 1. Glucose seems to have the same amount of upper bound outliers (see Figure 6) for both CHD classifiers, but again, since we have such a large CHD = 0 sample, it is difficult to make solid inference. It makes sense for CHD = 1 to have more upper bound outliers since high blood sugar is known to also increase risk of CHD. CHD = 0 having such large values may be due to the large sample. After all this analysis of the features, it was concluded that outliers should not be removed since the data set is already small, we are limited in CHD = 1 data points, and the outliers may have insights in our data that we do not want to remove.

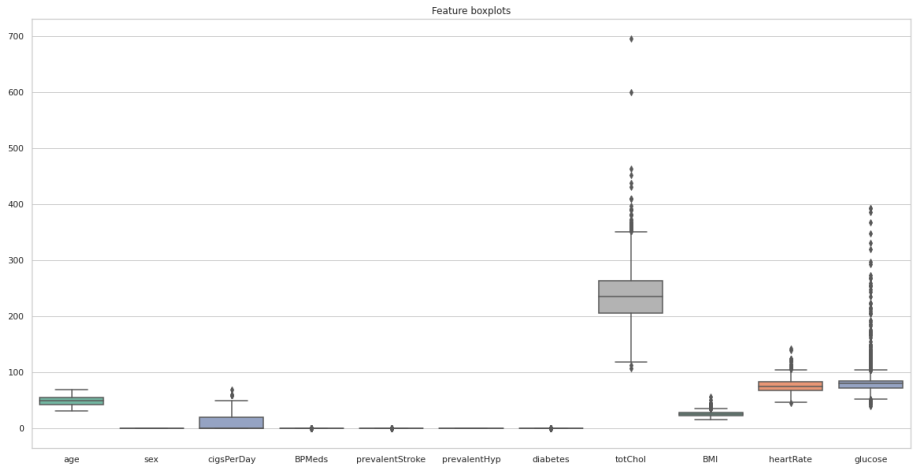


Figure 4: Visualization of outliers and distribution of data in features.

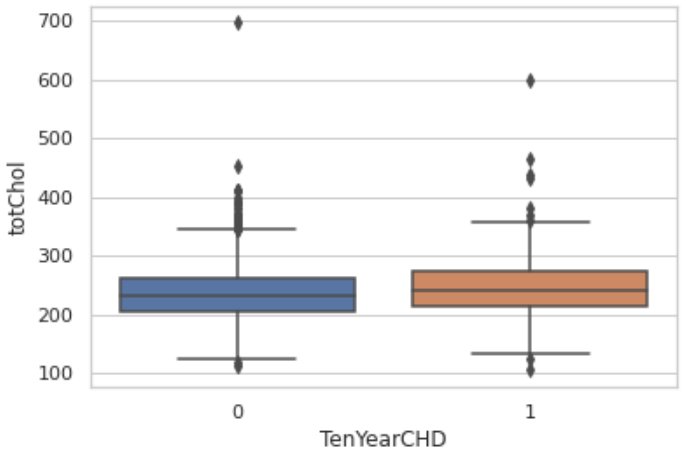


Figure 5: Box plot for totChol feature divided by classification.

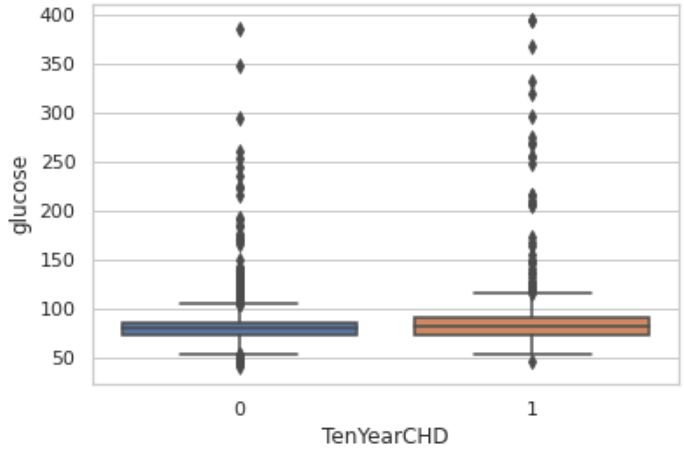


Figure 6: Box plot for glucose feature divided by classification.

Lastly, we can use the correlation matrix between predictor variables found in Figure 7 to observe which variables may need to be pruned. Having too many co-linear/correlated variables may skew model fitting. Based on the results of the correlation matrix, we set a threshold of 0.7 and chose to prune the `sysBP`, diaBP in favor for the `prevalentHyp` predictor. This makes sense given that certain systolic and diastolic blood pressures typically indicate hypertension in a patient.

There is also a strong correlation between `glucose` and `diabetes` which also follows good logical reasoning, but we decided to leave both variables in the data given our arbitrarily set threshold of 0.7.

With the dropped predictors we are left with 12 predictors and 1 response variable. A sample of the first 5 data points from this data set can be found in Table 1.

	age	education	sex	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	BMI	heartRate	glucose	TenYearCHD
0	64	2.0	1	3.0	0.0	0	0	0	221.0	25.48135318704286	90.0	80.0	1
1	36	4.0	0	0.0	0.0	0	1	0	212.0	29.77	72.0	75.0	0
2	46	1.0	1	10.0	0.0	0	0	0	250.0	20.35	88.0	94.0	0
3	50	1.0	0	20.0	0.0	0	1	0	233.0	28.26	68.0	94.0	1
4	64	1.0	1	30.0	0.0	0	0	0	241.0	26.42	70.0	77.0	0

Table 1: Sample data with correlated predictors dropped.

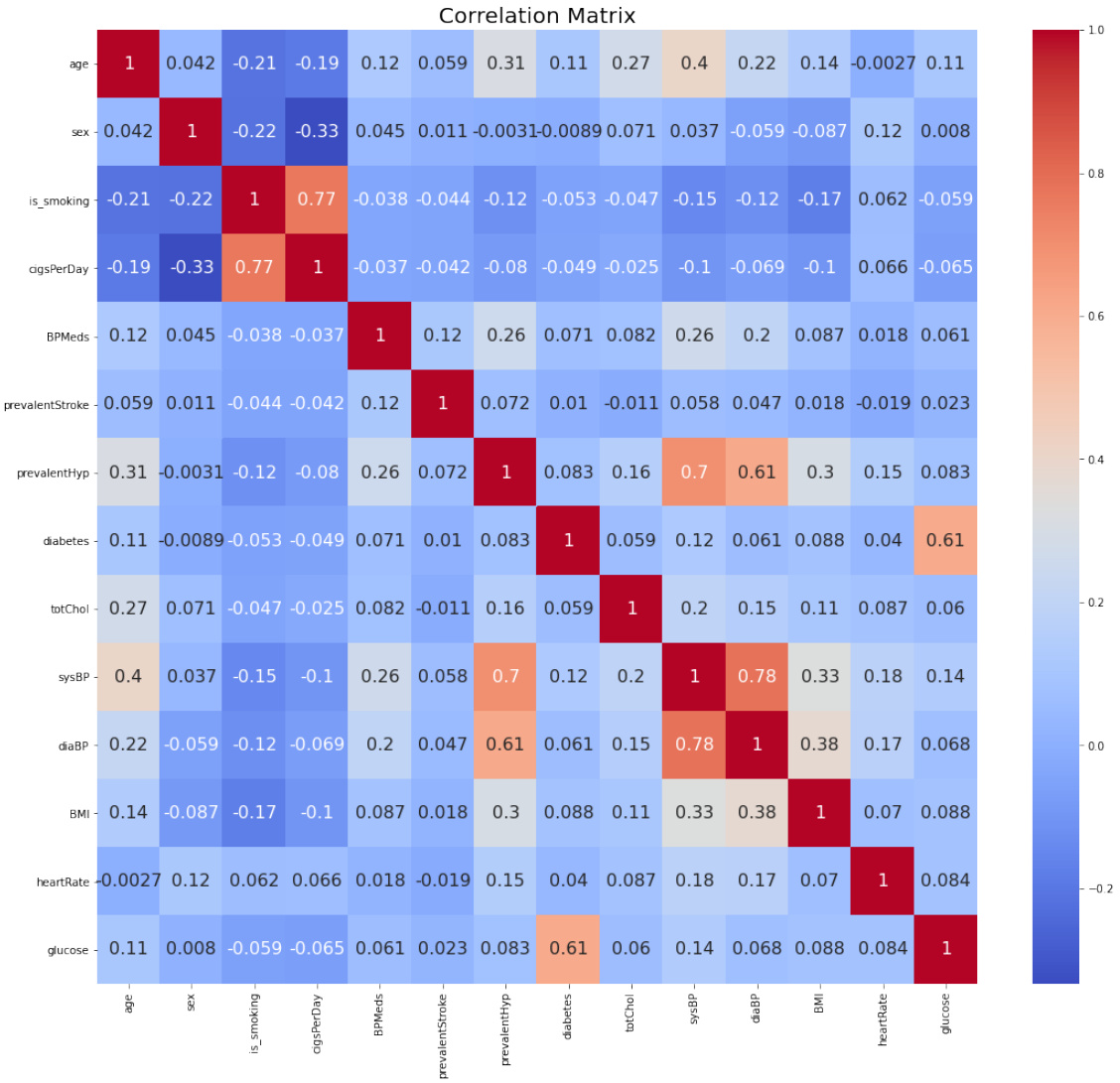


Figure 7: Correlation matrix for the data predictors

4 Methodology

We conducted our analysis on the training portion of the supplied data set. We then further split that portion of the data into a training set and a testing set using a standard 80 percent to 20 percent train-test split method. We then optimized various classification models using cross-validation and evaluated each models performance on the test data set.

The models studied include:

- K-Nearest Neighbors
- Naive Bayes
- Support Vector Classification - Linear and Radial Kernels
- Logistic regression

- Classification Tree
- Random Forest
- Neural Network
- AdaBoost

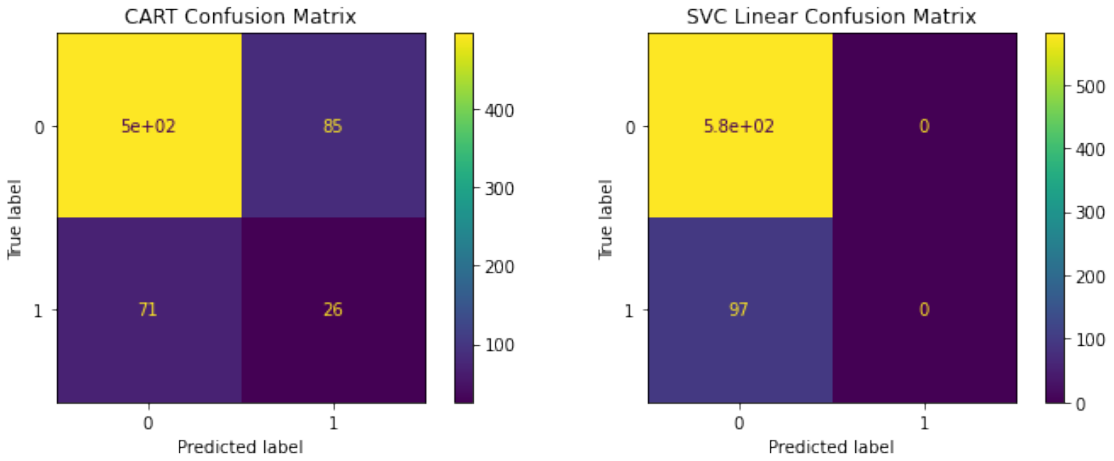
Each model came from the `scikit-learn` family of machine learning models. This allowed for streamline model fitting as well as performance analysis. Each model was fit using `GridSearchCV` to optimize any hyperparameters using 5-fold cross-validation whenever possible. The scoring method used to find the optimal parameter was the default accuracy metric. As an example, the KNN model fit models with `n_neighbors` = 1 to 10 which ultimately resulted the best model when `n_neighbors` = 8.

The results from the first round of model fitting are given in Table 2.

Model	**Train Score**	**Test Score**	**Pos Precision**	**Pos Recall**
KNN	0.845	0.848	0.286	0.041
NB	0.83	0.845	0.395	0.155
SVC_Lin	0.847	0.857	0	0
SVC_RBF	0.85	0.855	0.4	0.021
Log	0.853	0.858	0.571	0.041
CART	1	0.77	0.234	0.268
RF	0.848	0.853	0.435	0.103
NN	0.862	0.85	0.353	0.062
Ada	0.852	0.86	0.571	0.082

Table 2: Initial results from model fitting using accuracy as the scoring metric

Each model appears to perform well when classifying the data points in the test data set. But we argue that using basic accuracy for this problem is not the correct performance metric to use. Figure 8 shows the confusion matrices for the classification tree and the SVC - linear kernel models. From the results in Table 2 we know that the SVC model performs much better in overall classification accuracy, but upon closer inspection we see that it improperly classifies every single actual positive case as a negative.



(a) Classification Tree Confusion Matrix

(b) SVC - Linear Kernel Confusion Matrix

Figure 8: Confusion Matrix Comparison for initial models

Given the medical nature of this problem, the consequences of misdiagnosing a patient as "not at risk" for coronary heart disease could prove fatal if not caught later. Therefore, any false negative (FN) should be penalized heavily. Overall accuracy is not the correct method of measuring a models accuracy in our case. We argue that recall, specifically for the positive cases (labeled as `TenYearCHD` = 1), is a better metric of scoring the model's performance.

5 Dimensionality Reduction

Some classification models can be improved through the use of dimensionality reduction in order to highlight the largest variance in the data set. Our trimmed data set only contains 12 predictors, but we nonetheless performed a principal component analysis (PCA) and observed the results. Figure 9 gives a visual representation of the first two principal components along with each point's respective class indicated by their color. Unfortunately, the figure shows that there is no grouping and geometric separation between the two classes which means that the two-dimensional data is likely no more useful to us than the original data set with 12 predictors.

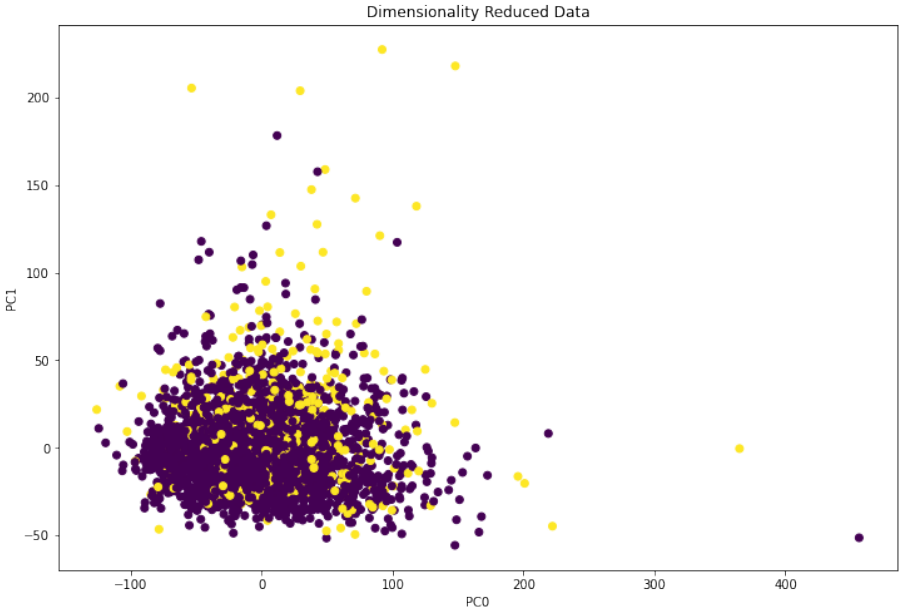


Figure 9: First two principal components of the reduced data set

Figure 10 gives us the scree plot of the PCA where we can perform a rudimentary analysis to observe when increasing the number of principal components has little to no effect. An argument could be made that 4 principal components explains a majority of the variation in the data and any more components yields no additional benefit. Given this fact, we fit a few of the models from the initial analysis, using accuracy as the scoring method, to the first four principal components. Table 3 indicates no increase in performance when compared to the same model's performance on the data set with 12 predictors.

6 Evaluation and Final Results

In order to create a more appropriate model for our given problem, we need to address the impact of false negatives (FN). One way that we can do this is by obtaining the classification probabilities for each model instead of the predicted values. This will then allow us to create our own custom evaluator that

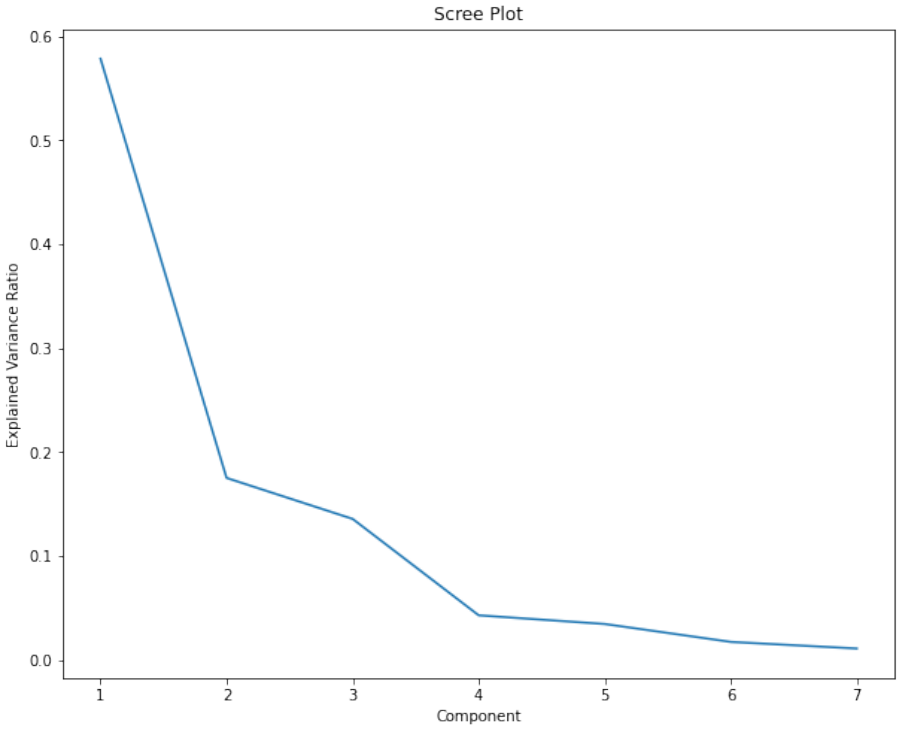


Figure 10: Scree plot of the PCA

Model	**Train Score**	**Test Score**	**Pos Precision**	**Pos Recall**
PCA Ada	0.847	0.857	0.000	0.000
PCA Ada n=50	0.854	0.860	0.600	0.062
PCA NN	0.855	0.857	0.500	0.113

Table 3: Results from basic model fitting on the reduced data set

predicts and optimizes by adjusting the threshold to reduce FNs as much as possible. Some models, such as logistic regression, lend themselves to this type of optimization, but others may not give such straightforward probabilities.

Luckily we can take advantage of `scikit-learn`'s built in scoring functions. Based on our observations earlier, we want to reward models with higher recall scores for 1-labeled data. By simply changing the scoring function within `GridSearchCV` we can optimize our models and hyperparameters by "recall" instead of overall accuracy. The results from fitting models based on recall scores are given in Table 4.

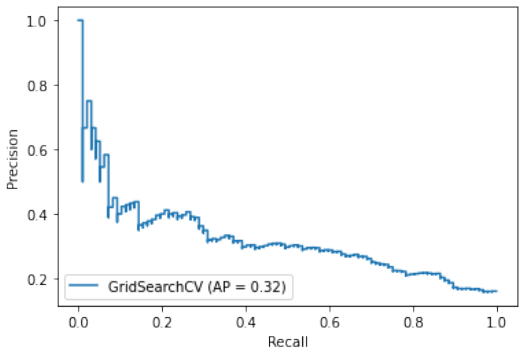
There is a large precision vs. recall trade off when it comes to a model's performance on our data set. If we optimize our models based on recall score, then the precision is lowered, which ultimately results in more false positives. However, misdiagnosing someone as having a risk for coronary heart disease when they are not at risk may be the necessary trade off given the nature of the problem.

A precision-recall curve can also give us more insight into a model's performance when adjusting the threshold for classifying a data point. By modifying the threshold you can observe the precision vs. recall trade off. Examples of precision-recall curves for some of the models fitted using recall as the scoring method are given in Figures 11 and 12.

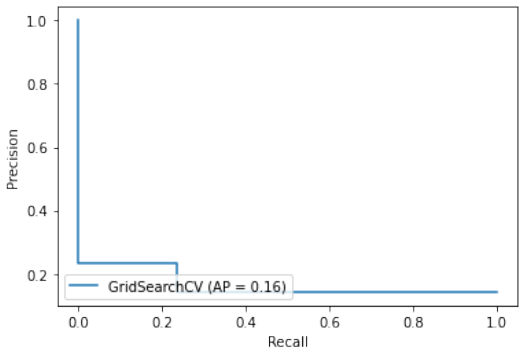
The results of these curves show that there is more to the precision-recall trade off than can be seen from a

Model	Train Recall	Test Recall	Pos Precision	Pos Recall
KNN	0.195	0.237	0.235	0.237
KNN n=10	0.075	0.031	0.300	0.031
NB	0.140	0.155	0.395	0.155
SVC_RBF	0.133	0.155	0.349	0.155
Log	0.068	0.041	0.571	0.041
CART	1.000	0.268	0.232	0.268
RF	0.217	0.237	0.217	0.237
NN	0.157	0.082	0.308	0.082
Ada	0.106	0.072	0.389	0.072

Table 4: Model fitting results using recall as the scoring method

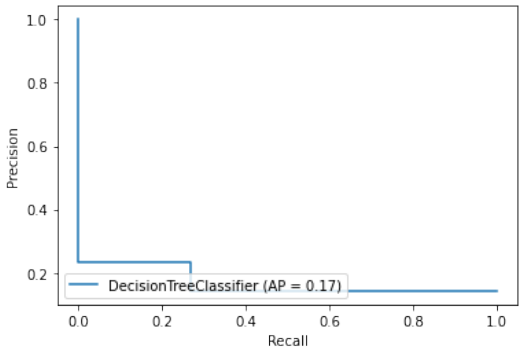


(a) Precision-Recall Curve - Logistic Regression

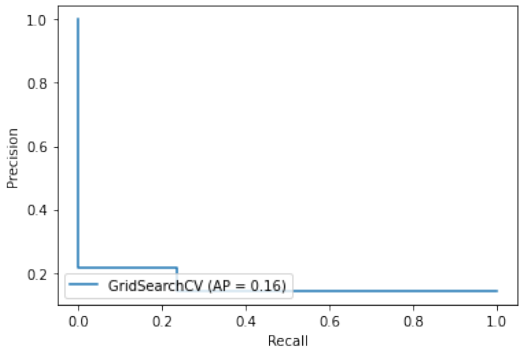


(b) Precision-Recall Curve - KNN

Figure 11: Precision-Recall Curve Comparison #1



(a) Precision-Recall Curve - Classification Tree



(b) Precision-Recall Curve - Random Forest

Figure 12: Precision-Recall Curve Comparison #2

quick optimization using the recall score. While it appears that the CART and random forest models provide the best recall scores, their precision-recall trade off is large when compared to the logistic regression model.

Lastly, taking the results from these curves we can observe what threshold it would take to obtain a recall = 1.0 and the resulting precision of such a threshold. These results are given in table 5.

While the best model appears to be the Naive Bayes, the resulting threshold is 0.2%. This translates to the possible case that if a patient has even a 0.2% chance of being at risk for coronary heart disease we would

Model	Precission	Recall	Threshold
KNN	0.143	1.000	0.000
NB	0.164	1.000	0.002
Log	0.161	1.000	0.043
CART	0.143	1.000	0.000
RF	0.143	1.000	0.000
NN	0.143	1.000	0.005
Ada	0.143	1.000	0.328

Table 5: Thresholds when recall = 1.0 for each model

classify them as such. Some consideration should be given to how reasonable a threshold is.

7 Conclusions

We have determined several conclusions through our analysis relating to the relationship between machine learning modeling and its potential applications to the healthcare setting. First, as we have seen here, it is critical to choose the correct metric for model evaluation in relation to the nature of the problem at hand. For example, the trade off between sensitivity and specificity is a well known conundrum in health care. Through our analysis we have mirrored this with our description of precision-recall tradeoff. Second, when addressing missing values in the data, we replaced these missing values with average scores. This could have had an unforseen impact on our results and requires further analysis. Lastly, is the importance of a very careful analysis of the relationships between variables and their effects on modeling results. For example, in an effort to improve the precision and recall scores of our modeling the first intuitive thought was to explore the effects of removing data points with outlying values; however, this had a further adverse impact on our results and so ultimately outlying data points were retained. This was most likely a result of underfitting. Ultimately, this shows us that having vastly large amounts of usable data will be essential to creating accurate usable models and to further determine the unseen and possibly counterintuitive relationships between features, labels, and classification results.

References

[1] "Heart Disease Facts — cdc.gov". *Centers For Disease Control and Prevention*, 2021, `https://www.cdc.gov/heartdisease/facts.htm`.

[2] Ganteng, Christofel. "Cardiovascular Study Dataset". *Kaggle.com*, 2020, `https://www.kaggle.com/christofel04/cardiovascular-study-dataset-predict-heart-disea/activity`.