Heart Disease Prediction

1: Introduction

Heart disease is the leading cause of death in America, accounting for approximately 25% of all deaths annually[1]. Unfortunately, the first indication many people have that they have heart disease when they have a heart attack or other similar incident. Better screening could save lives, but tests can be expensive and doctors can have difficulty gaining patient compliance if the procedures are inconvenient or risky.

The goal of this project is to create a system to flag patients for probable presence of heart disease. This could be of use to family practitioners with limited access to results from more expensive or rare tests, hospitals looking to ensure the best possible patient outcomes given more complete data, and healthcare networks or insurance providers who are attempting to move patients toward screening and preventative care instead of more expensive treatment or hospitalization for an acute cardiac incident.

2: Materials

2.1 Data Acquisition

For this project, I used the Heart Disease Data Set from the University of California (Irvine) machine learning repository[2]. The data came in four parts, containing patient information from the Hungarian Institute of Cardiology, two university hospitals in Switzerland, a V.A. Medical center in Long Beach, and the Cleveland Clinic. The original dataset had 76 attributes, but on the recommendation of the dataset description, I used a subset of 14 attributes as seen below.

age	Age in years
sex	Sex: 0 for female, 1 for male
cp	Chest pain type: typical angina, atypical angina, non-anginal pain, asymptomatic (coded 1-4)
trestbps	Resting blood pressure
chol	Cholesterol
fbs	Fasting blood sugar: 0 for under 120 mg/dl, 1 for over 120 mg/dl
restecg	Resting ECG: normal, having ST-T wave abnormality, left ventricular hypertrophy (coded 0-2)
thalach	Maximum heartrate achieved on a stress test
exang	Exercise induced angina: 0 for no, 1 for yes
oldpeak	ST depression induced by exercise relative to rest
slope	Slope of the peak exercise ST segment: upsloping , flat, or downsloping (coded 1-3)
ca	Number of major vessels (0-3) colored by fluoroscopy
thal	Thallium stress test results: 3 = normal; 6 = fixed defect; 7 = reversable defect
num	Diagnosis of heart disease (0 for no, 1-4 for yes)

1 https://www.cdc.gov/nchs/data/hus/hus16.pdf#019

2 http://archive.ics.uci.edu/ml/datasets/Heart+Disease

2.2 Data Acquisition/Wrangling

I started my data wrangling by loading the four datasets into python and combining them into one pandas dataframe. Many patients within the data had incomplete information, with a "?" used as a placeholder to indicate unknown measurements. I changed those to a NaN, along with any zeroes in columns where a zero was meaningless (a 0 in 'ca' could be a real measurement, but a 0 in 'chol' could not). I also changed the column that indicates the presence of heart disease in four categories (labeled as 0 for no heart disease, and 1, 2, 3, or 4 for different types of heart disease) into a simpler 0 or 1 for heart disease absent or present.

Four of the columns contained categorical data, such as the chest pain type column. This column used integer labels (1=typical angina, 2 = atypical angina, 3 = non anginal pain, 4 = asymptomatic), but the data itself is not ordinal. The distance from 1 to 4 is not greater than the distance from 3 to 4. I broke each of those columns up into multiple columns that gave a 0 or 1 value for each label in the original column using sklearn.preprocessing.OneHotEncoder, so that any calculations done on that information would weight each classification equally.

Of the 920 patient records in the data, 621 have one or more values missing, which is to be expected, since medical records are not reasonably expected to have information for every test possible. So that the data could be used in machine learning, the missing values needed to be filled in. I used KNN imputation to fill in for NaNs in the dataframe, with K=9.

Feature	Missing Values
age	0
sex	0
cp	0
trestbps	59
chol	202
fbs	90
restecg	2
thalach	55
exang	55
oldpeak	62
slope	309
ca	611
thal	486
pred	0

Finally, I standardized the non-categorical data so that it scaled from 0-1. This was for uniformity, so that I didn't have blood pressures in the 200s being weighed against other numeric data that only ranged from 0-3, leaving me with some heavily skewed coefficients.

2.3 Data Limitations

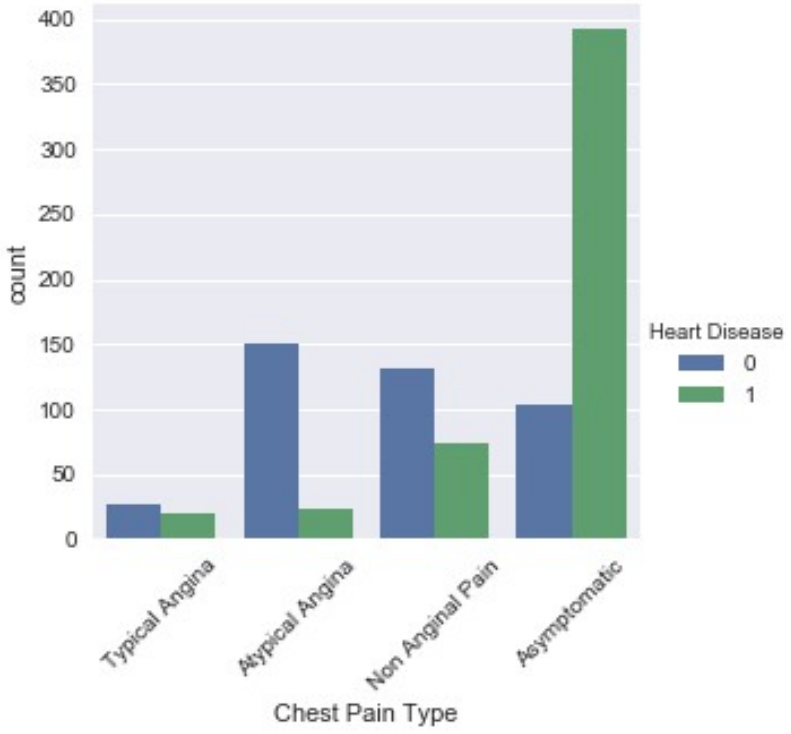
While exploring and cleaning the data, two important limitations of this particular data set emerged that need to be considered when interpreting the results of the model.

The majority of the missing data comes from the cardiac catheterization (611/920 missing) and thallium stress tests (486/920 missing). Part of the motivation for this project is to be able to identify at-risk patients who have not had those more powerful screening tests done, so that missing data is not too much of a hindrance. However, the full model, with all features considered, would certainly perform better with more of that data available.

This dataset is only of people who were selected by doctors as heart disease patients. This means the data has a heavy sampling bias towards patients with heart disease. According to the CDC[3], about 11.7% of adults have some form of diagnosed heart disease, but 55% of this dataset does. This

3 https://www.cdc.gov/nchs/fastats/heart-disease.htm

leads to the situation displayed in the figure below. Within this data, persons who have no chest pain (asymptomatic) are far more likely to have heart disease than not. This is because the the only asymptomatic patients in this data are there because a doctor had some other reason to suspect heart disease. This sample is not reflective of all patients.



Within the general population, the asymptomatic category should look more like the proportions in the atypical-angina (which would be enough to get a patient referred over to a cardiac clinic, but is not a strong predictor of heart disease itself). To use this data on the average patient walking into an annual physical, we would need general population data. As it stands now, the data makes it look like experiencing angina makes you less likely to have heart disease, which I think most doctors would not agree with.

3. Model

I chose to use logistic regression model. The decision boundary should be linearly separable, since there is no reason to believe that between any of the features a complex logical relationship exists. Logistic regression has the additional benefit of being able to handle missing inputs (where a patient has not had a specific test or procedure done) fairly smoothly, which is a realistic problem for the medical industry.

Since many of the numerical variables might be non-linear (the difference between 25 and 30 years old is likely to be less significant than the difference between 65 and 70), I added columns to the dataframe for the square of those features.

Using sklearn.model_selection.train_test_split, I split the data with %70 of the samples going into the training set. On that training data, I ran a grid search with a k=5 kfold cross validation to find the optimal regularization parameter for the logistic classifier.

The performance of the classifier depends somewhat on the random draw for the initial train/test split, but a typical output can be seen below.

Train set accuracy: 0.829192546584
Test set accuracy: 0.804347826087

Confusion Matrix on test set:

	Predict:0	Predict:1
Observed:0	98	29
Observed:1	25	124

age	agesq	sex	trestbps	trestbpssq	cp_1	cp_2	cp_3	cp_4
-0.11269	0.033499	0.637673	-0.171018	-0.029574	0.002587	-0.806272	-0.308267	0.74436
restecg_0	restecg_1	restecg_2	chol	cholsq	fbs	exang	oldpeak	slope_1
-0.177381	-0.013415	-0.239368	-0.074606	-0.023192	0.315433	0.755142	0.4754	-0.32586
slope_2	slope_3	thalach	thalachsq	thal_3	thal_6	thal_7	ca	casq
0.339298	0.085929	-0.555648	-0.578421	-0.359188	0.119399	0.170911	0.509163	0.148364

The coefficients above show the relative importance of the feature to the prediction. As I had stated in section 2.3 above, the cp_4 variable (asymptomatic of chest pain) is erroneously one of the strongest indicators of heart disease. I've debated whether to include chest pain as a feature due to this bad behavior, but ultimately decided that any final model should take it into account once some better data can be fed into the program.

The ecg slope and thal variables behave appropriately, with slope_1 and thal_3 (both corresponding to a "normal" result) having a negative coefficient of moderate magnitude, but the other two options that represent abnormalities having a positive coefficient. Sex (men are more at risk), exang (chest pain during exercise), and ca (vessels illuminated by fluoroscopy) were the most effective predictors of heart disease, and a strong indicator of no heart disease was thalach (heartrate achieved in a stress test)

An oddity of the regression outcome is that all the restecg variables have a negative coefficient. The contribution to the result is mostly in how negative it is (restecg_1 being more associated with heart disease than the other two). It does, at first glance, make it appear as if simply having an ecg performed makes patients less likely to have heart disease.

Since I don't want that behavior in the final model, I separated the features into dataframes based on test categories: basic vitals, rest ecg, routine labwork, stress ecg, thallium stress test, and cardiac catheterization. These dataframes can be combined based on what patient information is available to generate a custom classifier, such as this one I made using just basic vitals and routine bloodwork, which is what a family practitioner doing an annual physical would likely have access to.

Train set accuracy: 0.795031055901
Test set accuracy: 0.778985507246

Confusion Matrix on test set:

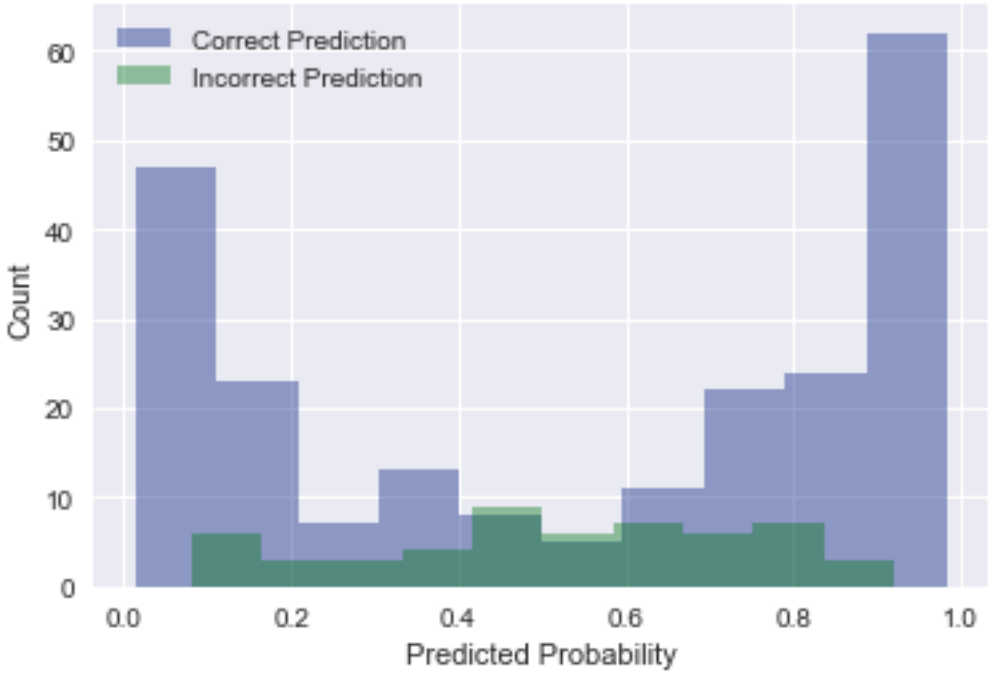
	Predict:0	Predict:1
Observed:0	90	35
Observed:1	26	125

Variable	age	agesq	sex	trestbps	trestbpssq	cp_1
Coefficient	0.468739	2.091463	1.480556	-0.698125	0.829354	-0.727871
	cp_2	cp_3	cp_4	chol	cholsq	fbs
	-2.166455	-1.061086	0.825597	2.499624	0.6009	0.347611

Interestingly, the overall accuracy only took a relatively minor 3-4 percent dip, despite not having any of the advanced diagnostics available. The algorithm puts much more emphasis here on

age, blood pressure, and cholesterol, the more traditional heart disease risk factors.

Since the raw output of the logistic regression is a probability prediction for each patient, I created a histogram of the prediction accuracy for gives levels of confidence.



The histogram shows how the prediction is very accurate when it has a confident probability score, and it struggles in the middle where p is between .4 and .6.

4. Conclusions

4.1 Recommendations

This model could run behind the scenes of a healthcare network, flagging patients as high risk for heart disease. This would allow a healthcare coordinator or doctor's office to contact the patient to suggest further screening. Future research into which of the advanced diagnostics does the best at explaining the uncertainty in this model would be interesting, potentially eliminating some expense and inconvenience by making sure the most information is gained from each new test.

The model in its current state shows good promise in predicting the presence of heart disease, but it needs more robust data for best results. Modern healthcare systems usually have digital records already, so feeding general population data into the system will alleviate some of the shortcomings of this model, particularly where chest pain is concerned.