ORIE 4741
Fall 2019
Project Midterm Report
Topic: Predicting Presence of Cardiovascular Disease in Patients
Team Member: *Adam Guo(hg426), Karan Maheswari(km857), Krishna Raju(kk992)*

Intro

In this project, we are concerned with what factors attribute to a person having heart disease. Heart disease is a chronic disease that takes a great amount of resources to treat, and if handled improperly could take away someone's life easily. By figuring out which factors have the most probabilities of leading to heart diseases, we could monitor those factors for patients, and thus prevent heart disease. For this project, we would seek to build a predictive model using data from the 'Cardiovascular Disease dataset' from Kaggle, which contains heart disease information of around 70,000 patients.

Goal

Our goal for the project is two-fold. First, we want to come up with a machine learning model to determine which factors (features) in the dataset contribute the most to the patient having cardiovascular disease. Second, we want to be able to predict the risk of a new patient having cardiovascular disease given his/her data, and maybe come up with a confidence interval for the prediction. To evaluate our model performance, we want to look at the test error for the model, and also by comparing the important factors we got from the model to some clinical proven results.

Data

We are using the '*Cardiovascular Disease Dataset*' from Kaggle. Our data set has 12 features and 70,000 entries. The features in the dataset are of 3 types -

Objective - Factual Information
Examination - Results of medical examination
Subjective - Information given by patient

It is important to distinguish between the three types, so that we can decide how much 'reliance' we can put on each feature. For example, subjective data like alcohol intake might not always be accurate because people might falsely report (knowingly or unknowingly) their medical conditions.

Below we list all our features with their type and some other useful properties:

1. Age | Objective Feature | age | int (days)
2. Height | Objective Feature | height | int (cm) |
3. Weight | Objective Feature | weight | float (kg) |

4. Gender | Objective Feature | gender | categorical code |
5. Systolic blood pressure | Examination Feature | ap_hi | int |
6. Diastolic blood pressure | Examination Feature | ap_lo | int |
7. Cholesterol | Examination Feature | cholesterol | 1: normal, 2: above normal, 3: well above normal |
8. Glucose | Examination Feature | gluc | 1: normal, 2: above normal, 3: well above normal |
9. Smoking | Subjective Feature | smoke | binary |
10. Alcohol intake | Subjective Feature | alco | binary |
11. Physical activity | Subjective Feature | active | binary |
12. Presence or absence of cardiovascular disease | Target Variable | cardio | binary |

Procedure

We plan to use a plethora of models to solve our problem accurately. So it is very important for us to identify overfitting and underfitting for our models. Our plan is to extensively use the training and validation sets to determine how well our models fit the data. A high training set error indicates underfitting, we would fix this by using more complex models and adding more features into our models. A low training set error but a high validation error would indicate overfitting. To tackle this, we would use less complex models, remove some of the features, and add more data (if possible). Also since we have only around 70,000 entries we would consider using less number of features to counter overfitting.

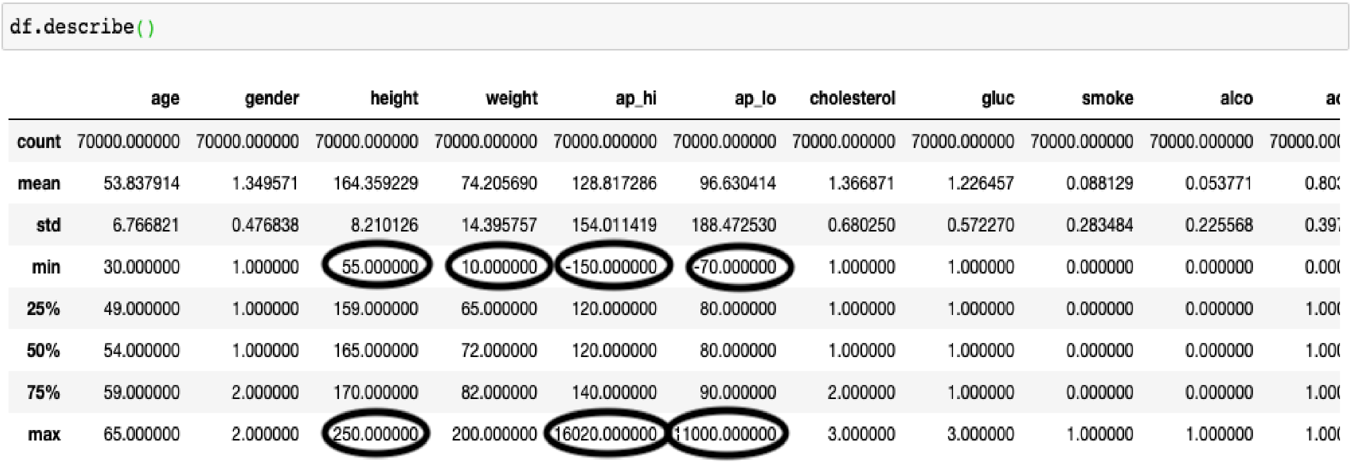
We plan to use two methods for evaluating our model –

1) Test set error – We expect our test set error to be a strong indication as to how well our model predicts the presence of heart disease. Therefore, a low test set error would mean a more effective model and vice versa.

2) Trends from Previous Studies – Since this problem is well studied, we would validate findings from our model by comparing them with results of past credible studies. This would not only help us determine effectiveness but can also act as a sanity check.

Data Exploration

There are 70,000 examples and 11 features which describe the target variable.
There isn’t any missing data but there are some outliers. We observed this in exploratory data analysis.



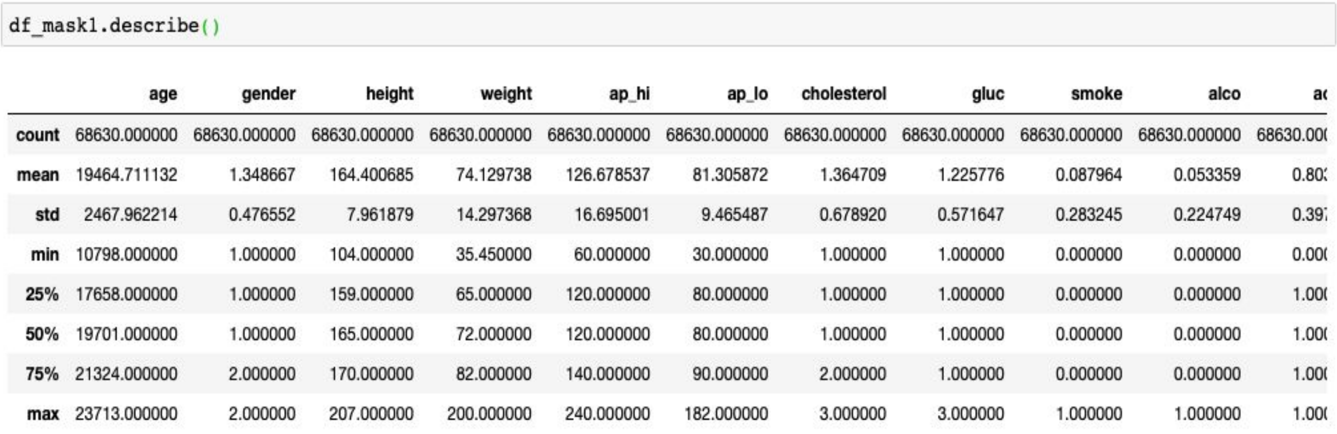
```
df.describe()
```

	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	ac
count	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.00
mean	53.837914	1.349571	164.359229	74.205690	128.817286	96.630414	1.366871	1.226457	0.088129	0.053771	0.80
std	6.766821	0.476838	8.210126	14.395757	154.011419	188.472530	0.680250	0.572270	0.283484	0.225568	0.39
min	30.000000	1.000000	55.000000	10.000000	-150.000000	-70.000000	1.000000	1.000000	0.000000	0.000000	0.00
25%	49.000000	1.000000	159.000000	65.000000	120.000000	80.000000	1.000000	1.000000	0.000000	0.000000	1.00
50%	54.000000	1.000000	165.000000	72.000000	120.000000	80.000000	1.000000	1.000000	0.000000	0.000000	1.00
75%	59.000000	2.000000	170.000000	82.000000	140.000000	90.000000	2.000000	1.000000	0.000000	0.000000	1.00
max	65.000000	2.000000	250.000000	200.000000	16020.000000	11000.000000	3.000000	3.000000	1.000000	1.000000	1.00

- Here we can see that weight feature (kgs) has a minimum value of 10. For a 30-65 year old person having a weight of 10 kgs is not possible.
- Similarly we see in height a person with a height of 55 cm. This indicates presence of corrupt values
- Another set of features which have corrupt values are ap_hi (Systolic Blood Pressure) and ap_lo (Diastolic Blood Pressure). After carefully studying the data we saw that there are garbage values of negative and very high blood pressure.

There are two possible solutions to this problem: either we can drop the samples with corrupt value or we can replace the value with median/mean values.

Since our dataset is large enough and our application is health critical we chose to drop the examples with corrupt entries.

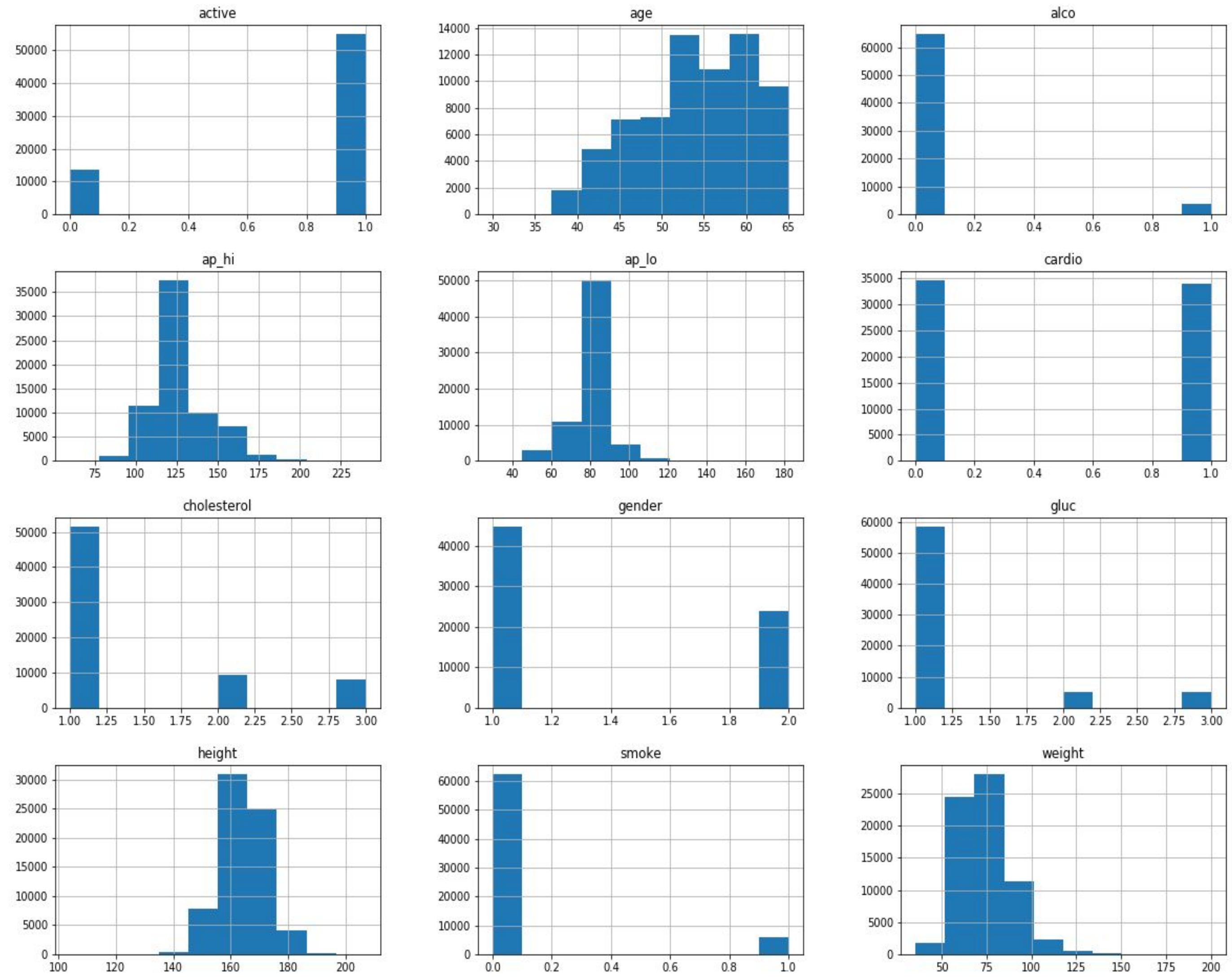


```
df_mask1.describe()
```

	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	ac
count	68630.000000	68630.000000	68630.000000	68630.000000	68630.000000	68630.000000	68630.000000	68630.000000	68630.000000	68630.000000	68630.00
mean	19464.711132	1.348667	164.400685	74.129738	126.678537	81.305872	1.364709	1.225776	0.087964	0.053359	0.80
std	2467.962214	0.476552	7.961879	14.297368	16.695001	9.465487	0.678920	0.571647	0.283245	0.224749	0.39
min	10798.000000	1.000000	104.000000	35.450000	60.000000	30.000000	1.000000	1.000000	0.000000	0.000000	0.00
25%	17658.000000	1.000000	159.000000	65.000000	120.000000	80.000000	1.000000	1.000000	0.000000	0.000000	1.00
50%	19701.000000	1.000000	165.000000	72.000000	120.000000	80.000000	1.000000	1.000000	0.000000	0.000000	1.00
75%	21324.000000	2.000000	170.000000	82.000000	140.000000	90.000000	2.000000	1.000000	0.000000	0.000000	1.00
max	23713.000000	2.000000	207.000000	200.000000	240.000000	182.000000	3.000000	3.000000	1.000000	1.000000	1.00

After looking at the filtered data we realized age is in number of days. Age in number of days can not model the relationship between heart disease and age properly. So we transformed it into years.

Then we drew the histograms of all the features, which are useful for us to understand each feature and its distribution.

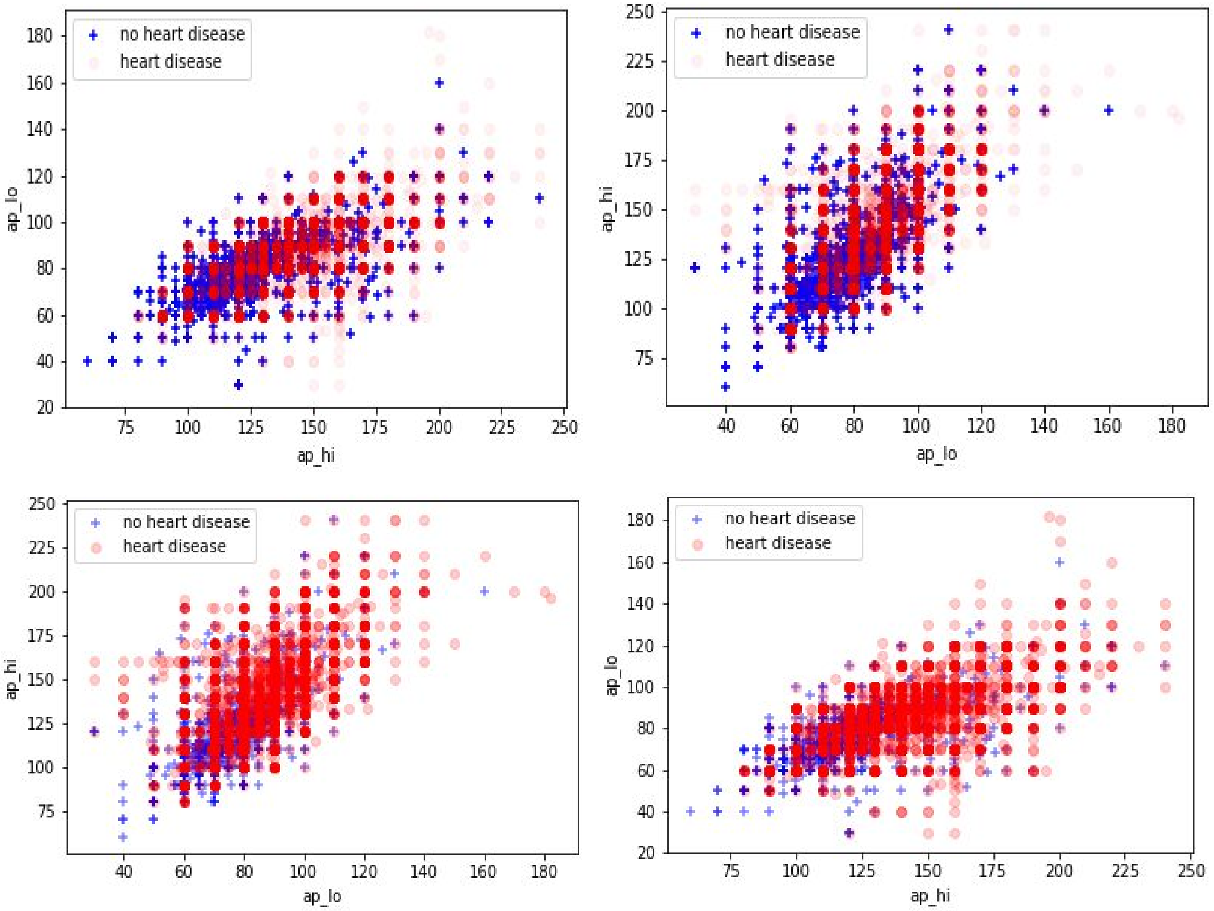


A few observations about the dataset from the histogram

Looking at the target we see that dataset is balanced (equal number of examples for both ‘heart disease’ and ‘no heart disease’ and not skewed.

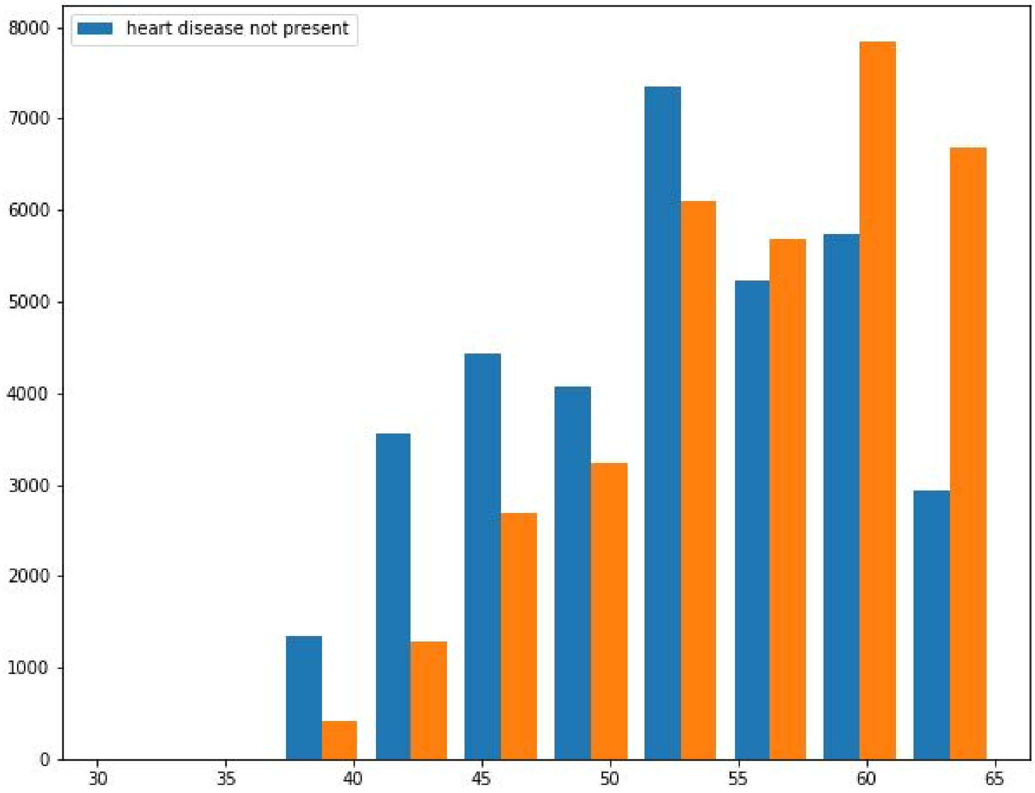
Next we go on to realise that the number of men and women isn’t uniform and thus before making any conclusion about correlation of heart disease with gender we should take this fact in account.

After this as commonly known that BP is correlated to heart disease, we wanted to model the relationship.
After drawing some scatter plots



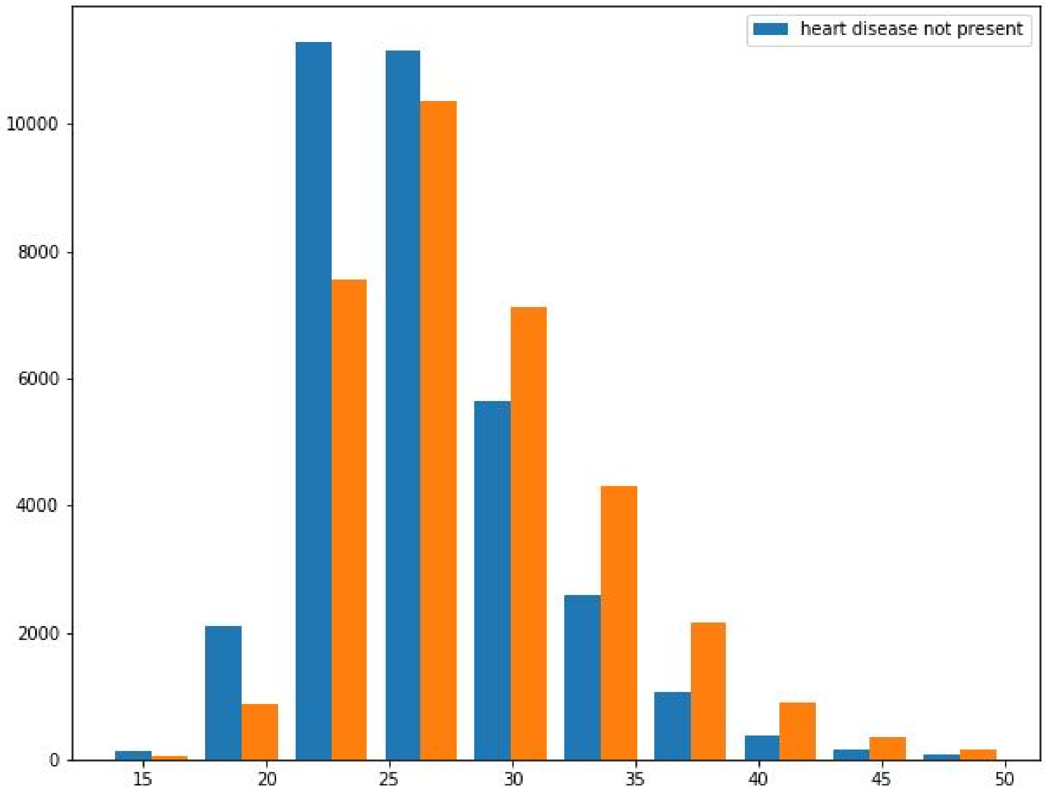
We can see a trend here that high blood pressure is somewhat correlated to the presence of heart disease.

Also we drew a histogram for age with presence of heart disease



We can clearly see a trend here, “as age is increasing people with heart disease increases” Also it can be seen that at lower age relatively fewer people have heart disease, and at higher age relatively few people do not have heart disease.

We couldn’t find any relation between height and presence of disease or weight and presence of disease so we did a polynomial transformation of height and weight which is Body Mass Index (height/weight^2). Here again we can very well see a trend in the figure below. At lower BMI fewer people have heart diseases and as obesity increases number of heart diseases also increases.



Project Plan

First, we still need to explore more feature engineering methods to optimize our features to use for our models. For example, we need to look for potential interaction between the features, like Age x smoking, and whether that can be a feature. Also, we need to look for possible dependencies or correlations between variables (e.g. Systolic blood pressure vs. Diastolic blood pressure) and whether we can use methods like Principal Component Analysis to eliminate the correlations.

Second, we need to determine the model we will be implementing, and how we are evaluating the performance of each. Because this is a binary classification task, we can try linear models like perceptron, SVM, etc. For those models, we need to think about adding a regularizer, maybe using an absolute loss function, etc. We can also try out more complex models, in the form of decision trees or random forest, and those methods could give us more direct insights into which factors play a more important role. In the case that the data is not linearly separable, and therefore we cannot use methods like perceptron and SVM, we can think of adding a hidden layer and building a neural net model. The weights or importance of each variable would be harder to explain, but it would guarantee good performance.

The next step would be to evaluate our models. For evaluation, we are thinking about either splitting our dataset into training, validation, and test dataset, or use K-fold cross validation. The benefit for using cross validation is that we can mimic testing our model on the real dataset. We also previously mentioned getting the variance of our prediction and generating a confidence interval, and we believe using bootstrap could help us achieve that. The test error we obtain should help us determine which model will hypothetically perform the best on the real dataset.