Human Heart Disease Prediction Using Data Mining Techniques

Sukriti Mishra
SJSU ID - 014580696

I. Introduction

Why I chose this topic?

To start with this project, I really had to explore lots of datasets. The main goal of exploring various datasets was that I wanted to work on a dataset with which I can get to apply the concepts/approaches suggested in Data Mining course. Another important aspect that I wanted to cover while exploring the numerous datasets was to find a dataset which is of my own area of interest and in spite of that closer to the real-world problem. I released this consideration to be important as with this project, I was getting a chance to understand the real-world problem from a closer look and to get real-truth and useful answers from the dataset.

The main idea of this project is to plan, explore, and analyze the data obtained based on each of the attributes provided in the dataset to determine whether those features contribute to any patterns in the data. I come from a biology background and am inclined to work as data scientist in medical sector in future. so, dataset on "Human Heart Disease Prediction" instantly caught my attention as heart disease is a very common disease which results in serious health problems such as heart failures. The cost due to heart disease in the US was about $219 billion from 2014 to 2015 each year. There are several types of heart conditions that are grouped under the same category and named as "heart disease". "Amongst those, coronary artery disease (CAD) occurs more frequently and is the most common type of heart disease that causes heart attack. Around 18.2 million adults having age more than 20 years have CAD and around 2 in 10 deaths happen due to CAD in adults less than 65 years old".[6] In spite of the significant development in medical technology such as bioinformatics, computational biology, medical imaging, etc. the healthcare sector in the world continues to face complications and challenges such as insufficient funding, rising drug costs, slow diffusion of medical knowledge, lack of personal and professional skills, medication errors, etc.

Why does this topic need a data mining method?

In the modern and progressive world, treatment decisions have shifted towards evidence-based medicine that involves invariably analysis of clinical data and systematically taking best treatment decisions from the available information. At present, data mining helps in healthcare related companies to regulate healthcare challenges by transforming data into meaningful information to improve healthcare. The data mining techniques can improve the treatment for some of the most common diseases such as Heart disease and Cancer. In this project, I used Machine learning classification algorithms such as support vector machines, Naïve Bayes Theorem, Logistic Regression and Random Forest first to prognosticate the disease and classify the human heart disease.

In the United States, Heart Disease causes the majority of deaths. "According to the heart disease and stroke statistics 2019 report, approximately 6,47000 US citizens die from heart disease each year. That means 1 out of 4 deaths is caused by heart disease". While cardiovascular disease can refer to different heart or blood vessel problems, the term is often used to mean damage to our heart or blood vessels by atherosclerosis, a buildup of fatty plaques in your arteries. Plaque buildup thickens and stiffens artery walls, which can inhibit blood flow through your arteries to our organs and tissues.

Overview of Dataset:

This dataset is procured from the UCI (University of California, Irvine) Center for machine learning and intelligent systems.[5] It contains four databases from four hospitals. Specifically, the Cleveland database is the only one dataset that has been used mostly by all ML researchers till now, due to containing fewer missing attributes and having more records. This dataset has 13

independent variables and one dependent variable(target). Independent variables include the age, sex, their serum cholesterol level, insulin level, chest pain type, and so on. Layout of the dataset:

Data Set Characteristics:	Multivariate	**Number of Instances:**	303
Attribute Characteristics:	Categorical, Integer, Real	**Number of Attributes:**	75
Associated Tasks:	Classification	**Missing Values?**	Yes

Fig 1: Overview of the Dataset

Numbers of Instances: 303
Number of features: 13 (independent variables) + target feature (dependent variable)
Attributes: as shown in Table 1

Age	Age (years)
Sex	(1 = male; 0 = female)
Cp	chest pain type
trestbps	resting blood pressure (in mm Hg on admission to the hospital)
chol	serum cholesterol in mg/dl
fbs	(fasting blood sugar > 120 mg/dl) (1 = true; 0 = false)
restecg	resting electrocardiographic results
thalach	maximum heart rate achieved
exang	exercise induced angina (1 = yes; 0 = no)
thal	(Thallium heart scan) Thallium heart rate
oldpeak	ST depression induced by exercise relative to rest
slope	the slope of the peak exercise ST segment
ca	number of major vessels (03 = normal; 6 = fixed defect; 7 = reversible defect) colored by fluoroscopy
target	have disease or not (1= yes, 0 = no)”

II. Methodology

1.Support Vector Machine: SVM is a supervised machine learning model that analyzes the large amount of data. It uses classification and regression analysis to identify the patterns in the sample. In this method, we aim to find the best hyperplane that separates the dataset into two classes, as shown in Fig. 10. SVM consists of a datapoint that is a p-dimensional vector, a list of p numbers, and a (p − 1) dimensional hyperplane that separates the data points. We can derive many hyperplanes from the dataset that can classify the data. However, we have to find the hyperplane that represents the largest separation, or margin, between the two classes. This means that the hyperplane that has the maximum distance from it to the nearest data point on each side of the dataset. If such a hyperplane exists, then it is known as the maximum-margin hyperplane(w) and the linear classifier" as shown in Fig. 1

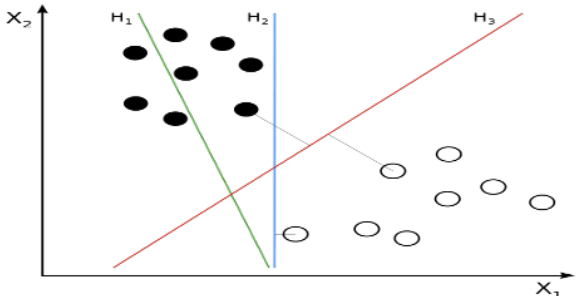


Fig.2: H_1 does not divide the classes, H_2 does, but only with a small margin, H_3 separates them with the maximum margin. So, H_3 will be considered as the best hyperplane with maximum margin.

In the SVM algorithm, we are looking to maximize the margin between the data points and the hyperplane. The cost is 0 if the predicted value and the actual value are of the same sign. If they are not, we then calculate the loss value. We also add a regularization parameter the cost function. The objective of the regularization parameter is to balance the margin maximization and loss. When there is no misclassification, we only have to update the gradient from the regularization parameter. When there is a misclassification, we include the loss along with the regularization parameter to perform gradient update. Additionally, SVM has a technique known as the kernel trick. These functions take low dimensional input space and convert it to a higher dimensional space i.e., it converts non transformable problem to transformable problem, these functions are called kernels. It is mostly common in non-linear separation problems. It does some extremely complex data transformations, then finds out the process to separate the data based on the labels or outputs we have defined.[1]

Model Suitability: SVM can be used to solve both classification and regression problems. SVM algorithm is suitable for small data sets. And performs very well when the data set has no noise i.e., target classes are not overlapping and relatively memory efficient. In my dataset, target values are not overlapped to each other and there is a clear margin also my dataset is also not very large. SVM works relatively well when there is a clear margin of separation between classes. SVM has L2 Regularization feature. So, it has good generalization capabilities which prevent it from over-fitting.

2. Naive Bayes Classifier: A Naive Bayes classifier is a probabilistic machine learning model that's used for classification task. The crux of the classifier is based on the Bayes theorem. We make two assumptions here, that predictors are independent. Naïve Bayes is a supervised learning

algorithm that is a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naïve) independence assumptions between the features and all the predictors have an equal effect on the outcome. The term 'Naive' here suggests that the algorithm naively believes that all the features in the dataset are "conditionally independent". Naïve Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression, which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers. Bayes' theorem states the following relationship, given class variable y and dependent feature vector x1 through xn. Below is the simplified relationship formula assuming conditional probability is

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y) \prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)} \qquad \text{Eqn (1)}$$

In spite of the oversimplified assumptions, naive Bayes classifiers worked quite well in many real-world situations. The main advantage of this algorithm is it requires a small amount of training data to estimate the necessary parameters. Naive Bayes learners and classifiers can be extremely fast compared to more sophisticated methods. The decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one-dimensional distribution. This in turn helps to alleviate problems stemming from the curse of dimensionality.

Model Suitability: It is basically used for classification with a high dimensional training dataset. Some examples where we can apply Naive Bayes Theorem are spam filtration, sentimental analysis, and classification of news articles. This algorithm is simple and effective. It is very easy and convenient to build models and make predictions with Naive Bayes algorithms. We should consider Naïve Bayes as the first algorithm for solving classification problems.[3]

3. Logistic Regression: Logistic Regression is used when the dependent variable(target) is categorical. For example, to predict whether an email is spam or not whether the tumor is malignant or not. Data is fit into linear regression model, which then be acted upon by a logistic function predicting the target categorical dependent variable. To predict which class a data belongs, a threshold can be set. Based upon this threshold, the obtained estimated probability is classified into classes. Say, if predicted_ value ≥ 0.5, then classify email as spam else as not spam. Decision boundary can be linear or non-linear. Polynomial order can be increased to get complex decision boundary. The logistic distribution constrains the estimated probabilities to lie between 0 and 1. The estimated probability is: $\ln[p/(1-p)] = \beta_0 + \beta_1 X$ where, p is the probability that the event Y occurs, p(Y=1), p/(1-p) is the "odds ratio" and ln[p/(1-p)] is the log odds ratio, or "logit".[4]

Model suitability: Binary logistic regression estimates the probability that a characteristic is present (e.g., estimate probability of "success") given the values of explanatory variables. In my dataset Let Y be a binary response variable (Y= 1 if heart disease in present and 0 if not). The data Y_1, Y_2, ..., Y_n are independently distributed, i.e., cases are independent. The dependent variable does not need to be normally distributed. It does not assume a linear relationship between the dependent variable and the independent variables, but it does assume linear relationship between the logit of the response and the explanatory variables; $logit(\pi) = \beta_0 + \beta X$. It uses maximum likelihood estimation (MLE) rather than ordinary least squares (OLS) to estimate the parameters.

4. Random Forest: Random forest is an ensemble tree-based learning algorithm where model using bagging as the ensemble method and decision tree as the individual model. Ensemble algorithms are those which combines more than one algorithm of same or different kind for classifying objects. The Random Forest Classifier is a set of decision trees from randomly selected subset of training set. It aggregates the votes from different decision trees to decide the final class of the test object. In figure 2, Random Forest Prediction for a classification problem is given as f(x) = majority vote of all predicted classes over B trees. In Random Forest, results are aggregated, through model votes or averaging, into a single ensemble model that ends up outperforming any individual decision tree's output.[2]

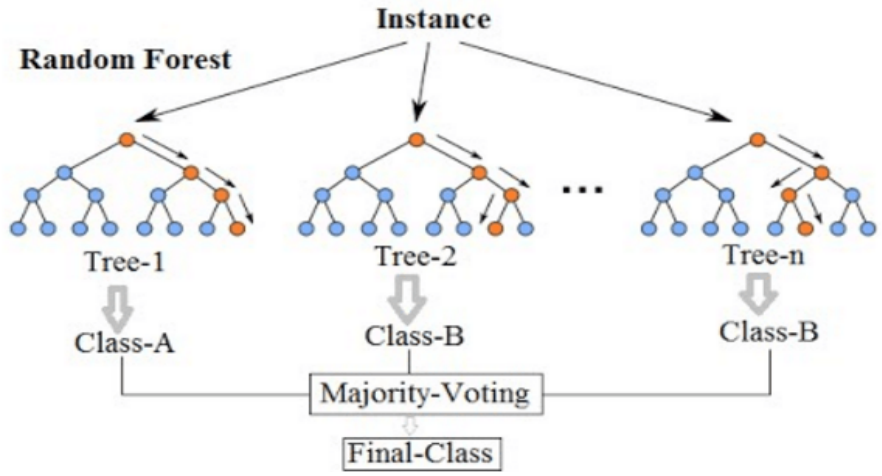


Fig 3: Random Forest Algorithm

Model Suitability: It is one of the most accurate learning algorithms available. For many datasets, it produces a highly accurate classifier. It runs efficiently on dataset. It can handle thousands of input variables without variable deletion. So, the reason behind to select this model is that Random forest classifier also performs feature selection. It gives estimates of what variables that are important in the classification. It generates an internal unbiased estimate of the generalization error as the forest building progresses. As my dataset has few missing values, this model has an effective method for estimating missing data and maintains accuracy when proportion of the data are missing.

III. Feature Engineering

Raw data comprises thousands of numbers of features but all that features will not be relevant to our project work. Irrelevant or partially relevant features can negatively impact model performance. So, the selection of desirable parameters should be done by using the feature engineering concept. During data collection, the databases have 76 raw attributes, only 14 of them are actually used. So, for this project I have taken the heart disease dataset that has only 14 attributes only those who are mainly responsible for the heart disease. This is called feature selection. The benefits of feature selection are reducing overfitting and training time and improves accuracy. The following four feature selection techniques that I have used as they are easy to implement and also gives good results.

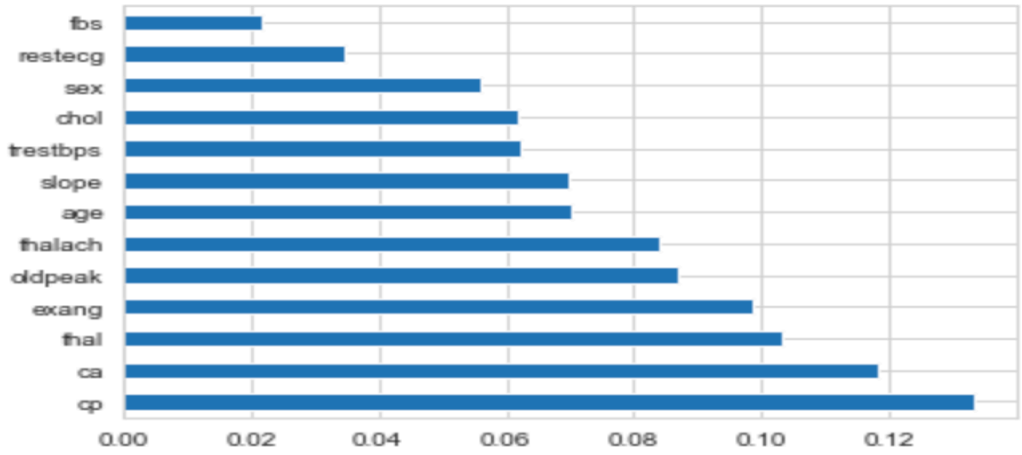
1.Univariate Feature Selection: My dataset is a classification predictive modeling problem with categorical input variables. So, in this case, the most common correlation measure for categorical

data is the chi-squared test. So, for univariate feature selection, we could use Statistical Chi-squared test to select certain features that have the best relationship to the performance variable. The scikit-learn library provides the *SelectKBest* class that can be used to select a specific number of features in a suite of different statistical tests.

	Features	Score
7	thalach	650.008493
9	oldpeak	253.653461
2	cp	217.823922
11	ca	210.625919
8	exang	130.470927
4	chol	110.723364
0	age	81.425368
3	trestbps	45.974069
10	slope	33.673948
1	sex	24.373650
12	thal	19.373465
6	restecg	9.739343

Fig 4: The chi-squared (statistical test score for the best feature's selection

2. Feature Importance: We can the significance of each feature of our dataset by using the Model Characteristics property. Feature value gives you a score for every function of your results, the higher the score the more significant or appropriate the performance variable is. Feature importance is the built-in class that comes with Tree Based Classifiers, we will use the Extra Tree Classifier to extract the top features for



the dataset.[7] An extra-trees classifier implements a meta estimator that fits a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

Fig 5: Extra Tree Classifier to extract the top features for the dataset.

3.Correlation Matrix: It indicates how the features are related to each other or to the target variable. The correlation may be positive means increase in one value of the feature increases the value of the target variable or negative (increase in one value of the feature decreases the value of the target variable) To show correlation, I plotted heatmap using seaborn library. It is an easy way to classify the features are most relevant to the target variable, from this heatmap in the last row (target vs other features), we can observe that the 'cp' chest pain is highly related to the target

variable having highest correlation value. We can say that chest pain contributes the most in prediction of presences of a heart disease and that is true as well because usually a cardiac attack occurs when blood clot blocks blood flow to the heart and tissues loses oxygen without blood and causing chest pain. The other strong relationship is maximum heart rate received with target variable and chest pain. Additionally, age is highly correlated with resting blood pressure (in mm Hg on admission to the hospital).

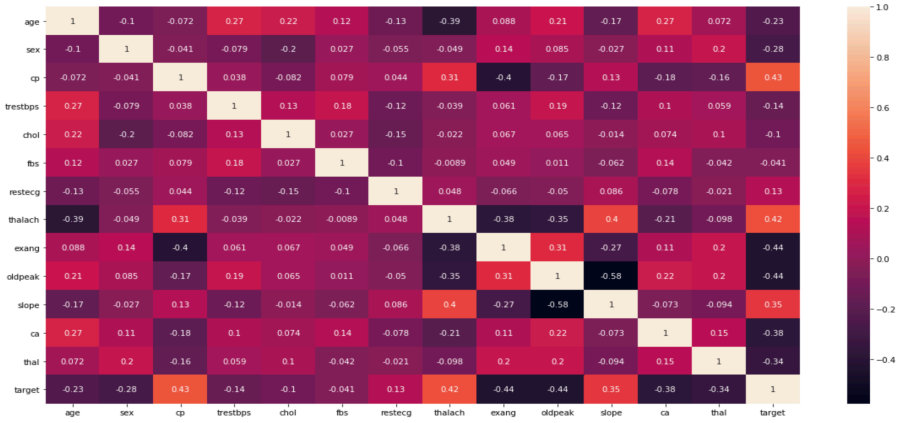


Fig 6: Correlation matrix using heatmap

4. Regression Coefficient for Binary Logistic regression: A regression coefficient describes the size and direction of the relationship between a predictor and the response variable. Coefficients are the numbers by which the values of the term are multiplied in a regression equation. The interpretation uses the fact that the odds of a reference event are P(event)/P (not event) and assumes that the other predictors remain constant. The greater the log odds, the more likely the reference event is. Therefore, positive coefficients indicate that the event becomes more likely and negative coefficients indicate that the event becomes less likely. [7]

IV. Result

Table 1: Accuracy rate for all selected models with and without feature selection

	Accuracy rate	
Models	**Before Feature Selection**	**After Feature Selection**
SVM	**92.2**	**74.15**
Naïve Bayes	**87.8**	**83.41**
Random Forest	**90.24**	**91.22**
Logistic Regression	**84.3**	**83.90**

Which data-mining methods are suitable for this problem?

For all features, SVM performed really very well as compared to other three models. For classification problem, SVM algorithm is suitable for small data sets and performed very well when the data set has no noise i.e., target classes are not overlapping. For the selected features, Random forest performed very well. Random Forest is better than the Naïve Bayes Classifier, Support Vector Classifier and Logistic Regression because RF is one of the most accurate learning algorithms can handle thousands of input variables without variable deletion and gives estimates of what variables that are important in the classification. The Area under the ROC curve of RF is 91% which is very satisfactory. So, the selected model for the human heart disease prediction is the Random Forest Classifier.

V. Discussion

Difference between all features and selected features:

The difference between all the features and selected feature is that with all the features all the models really performed very well without overfitting and underfitting. On the basis of feature engineering results, I dropped some features who have large negative correlation coefficient values. However, during the feature selection, I noticed that serum cholesterol has large negative coefficient values, but I considered cholesterol in selected features because having high blood cholesterol is one risk factor for heart disease. The features that I dropped are typical angina chest pain type, non-anginal chest pain type, normal electrocardiographic results, normal thalassemia, and number of major vessels. There are four types of chest pain, asymptomatic, atypical angina, non-anginal pain and typical angina. Most of the Heart Disease patients are found to have asymptomatic chest pain so I dropped other types besides having high correlation coefficient values. Along with the model accuracy rate, I considered the confusion matrix often used to describe the performance of classifiers. For all features, RF model has the less accuracy but more sensitive due to low false negatives (FN) values whereas for selected features, RF has lowest FN values with highest accuracy. FN means we predicted no, but they actually do have the disease. (Also known as a "Type II error."). In case of medical data, FN should be very low as much as possible. For selected feature, RF model has the highest accuracy rate and satisfactory sensitivity and specificity also also due to low false negatives (FN) and False Positive (FP) values respectively as shown in fig 7.

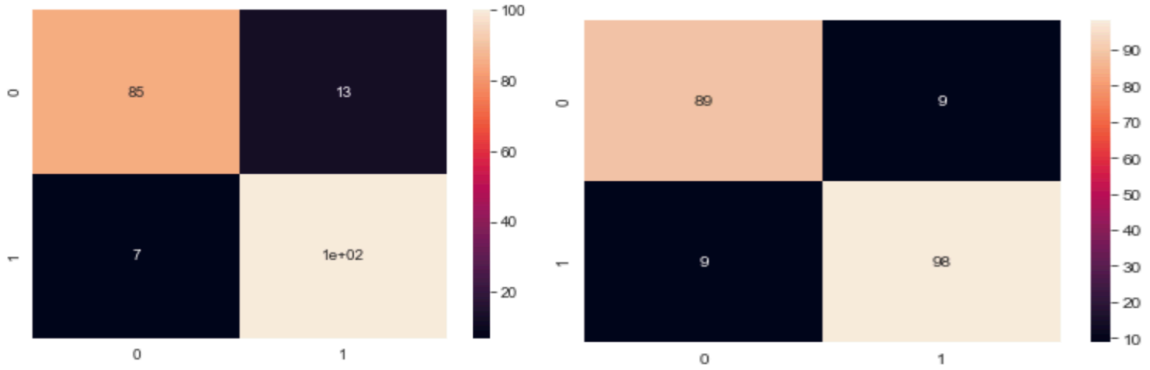


Fig7: left to right: Random Forest Confusion matrix before and after feature selection

Result (interpretability) based on my domain knowledge:

The major features contributing to precision of predicting model are maximum heart rate achieved and chest pain types. Most of the Heart Disease patients are found to have asymptomatic chest pain. An asymptomatic attack, like any heart attack, involves, blockage of blood flow to your heart and possible damage to the heart muscle. The risk factors for asymptomatic heart attacks are same as those with heart symptoms. I expected there to be a strong positive relationship between the all-chest types and heart disease. Then, after research, I found that asymptomatic chest pain usually occurs in heart patients. Asymptomatic Heart attack puts you at a greater risk of having another heart attack which could be deadly. The only way to tell If you had asymptomatic attack is by an electrocardiogram. These tests can reveal changes that signal a heart attack. The resting electrocardiographic results show having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV) and definite left ventricular hypertrophy. For these electrocardiographic results, the correlation coefficient values are negative still I did not drop this feature. In normal type of rest ECG proves to be important for the predictive analysis along with the down sloping and upsloping ST slope. The patient with these two features usually has cholesterol level between 170 to 225 mg/dl. The fasting blood sugar (fasting blood sugar > 120 mg/dl) also proves to be important factor the heart disease prediction. Optimal blood pressure is defined as 120 mm Hg(systolic) which is the pressure as our heart beats over 80 mm Hg (diastolic) which is the pressure as our heart relaxes. For our resting heart rate, the target is between 60 and 100 beats per minute (BPM). Fasting can help lower blood pressure but also result in an electrolyte imbalance. High blood pressure (hypertension) increases the risk of heart attack and stroke. Low blood pressure (hypotension) causes weakness. So, for healthy heart, blood pressure should be in normal range i.e., blood pressure below 120/80 is normal. In data analytics, domain knowledge plays a significant role for feature selection and feature extraction. The benefits of feature selection are reducing overfitting and training time and improves accuracy.

References

[1] "towards data science," [Online]. Available: https://archive.ics.uci.edu/ml/datasets/heart+disease.

[2] "towards data science," [Online]. Available: https://towardsdatascience.com/understanding-random-forest-58381e0602d2 .

[3] "towards data science," [Online]. Available: https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c.

[4] "scikit learn," [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html.

[5] "wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Support_vector_machine.

[6] "Human heart disease," [Online]. Available: "Centers for Disease Control and Prevention," 2 December 2019. [Online]. Available: https://www.cdc.gov/heartdisease/facts.html. .

[7] H. Deshmukh, "towards data science," [Online]. Available: https://towardsdatascience.com/heart-disease-uci-diagnosis-prediction-b1943ee835a7.