Cardiovascular Disease Prediction

Mrinal Soni, Yanbing Lin, Sai Abhishikth Devabhaktuni

Abstract

Healthcare has always been an important sector; early detection or diagnosis is the key to early treatment and prevention of extreme cases. That was the motivation behind our project. Cardiovascular Disease refers to the conditions that affect the heart or blood vessels. It describes conditions ranging from peripheral artery disease and high blood pressure to heart attacks and strokes. For the scope of this project, we worked on medical data from Kaggle to predict if a potential patient has CVD or not. We implemented baseline Logistic Regression model, following it up with Naïve Bayes, Random Forrest, Ensemble Learning, KNN, Support Vector Machine and Gradient Boosting. We compared results between scaled and unsclaed data and it was observed that SVM, Logisitc Regression and KNN performed best.

Introduction

Cardiovascular disease (CVD) is a leading cause of death both in developing and developed nations. It accounted for 17.5 million deaths worldwide in 2012 (31% of the total deaths) and expected to increase up to 24.2 million by 2030. Identifying individuals at the highest risk of cardiovascular disease to target treatment and prevention strategies has been the focus of research for over 40 years. CVD risk prediction tools estimate the prob- ability of having a cardiovascular event within a defined time frame, based upon levels or presence of known risk factors. Given the historic data of deaths that CVDs have caused, it is essential that we build a model that predicts presence of CVD in a person based on their lifestyle. This could be effective in a few ways, a) early detection – early treatment, b) it could lead to people changing their lifestyle based on our analysis or prediction.

Technical Approach

For the scope of this project, we decided to compare baseline Logistic Regression with various other supervised machine learning techniques like Naïve Bayes, Random Forrest, KNN, Support Vector Machines and KNN.

Logistic Regression: Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable.

Random Forrest: Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification).

Naïve Bayes: classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naïve) independence assumptions between the features.

KNN: K-nearest neighbors algorithm (k-NN) is a non-parametric method proposed by Thomas Cover used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space.

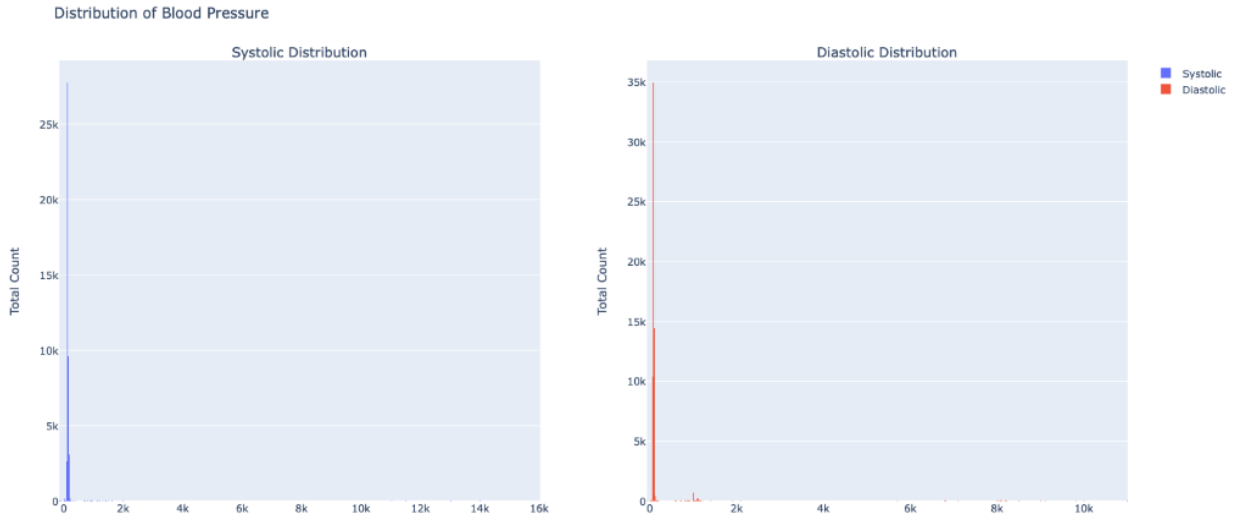
Decision Trees: Decision tree learning is one of the predictive modelling approaches used in statistics, data mining and machine learning. It uses a decision tree (as a predictive model)

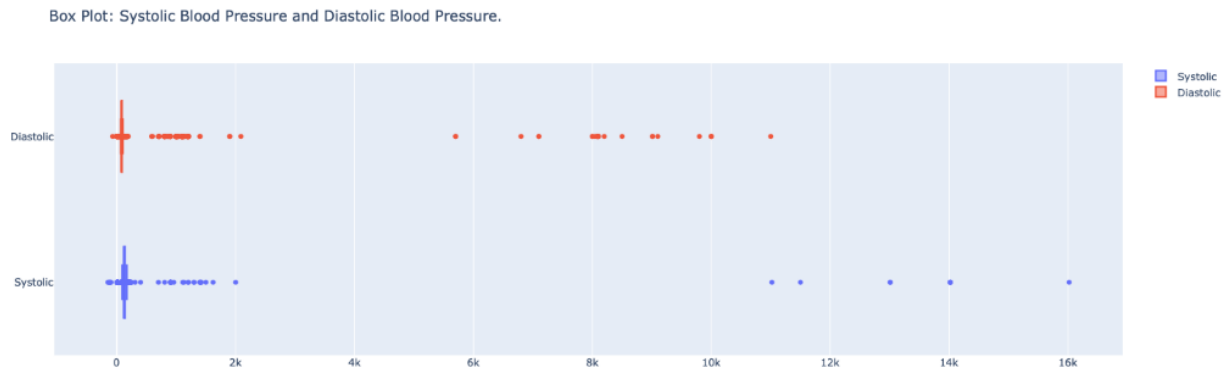
to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves).
Support Vector Machine: are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis.

Experimental Results

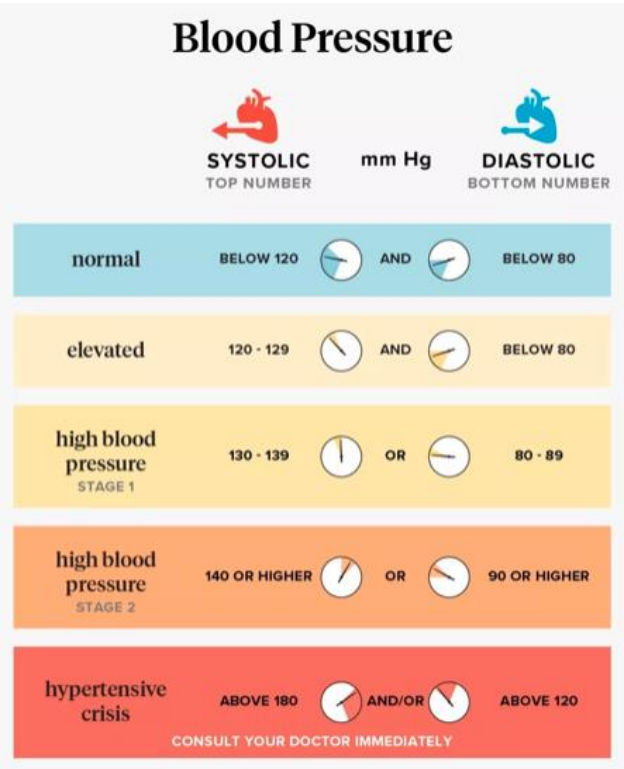
Data Collection: Initial goal was to obtain Electronic Health Records (EHR) and build a predictive model capturing the immense information that EHRs contain. That was a struggle as EHR are private and hard to obtain due to the anonymous nature of the data. We tried getting a hold of EHR data from Stanford and the UK Data Bank, but that was not a success. After this initial setback, we switched to more publicly available datasets. We obtained "Cardiovascular Disease" dataset on Kaggle, where we have 12 features (11 predictor features and 1 target feature). There are three types of features: **subjective**, **examination** and **subjective**. Objective features are records that are information about the patients. Subjective features contain information given by the patients about their habits / activities. Examination features contain results of medical tests performed by doctors. We have a mix of quantitative features and categorical features (which have either been transformed using label encoding or are represented in a binary representation). The dataset is well balanced as our predictor variable has 35,004 instances for cardio = 0 and 34,972 for cardio = 1. On preprocessing, we observed that our data is free of null values. Further, we observed that there were some duplicate values even before any preprocessing was performed (these were rows with identical data). Since the count of identical rows was only 24, we decided to drop these rows and we had significant amount of data.

Outlier Detection: On exploring our data, we observed that there were a few unusual values which might affect the performance of our model as they are **outliers**. To get a clearer understanding of such outliers, we focused on boxplots.





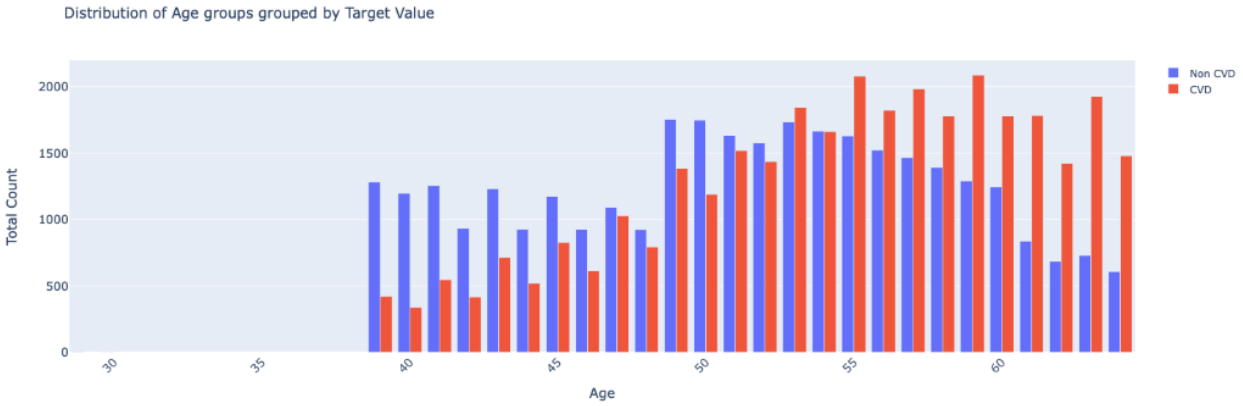
As seen in the plots above, features "ap_hi" and "ap_lo" (our primary quantitative features) had immense outliers. This may be due to an error in data entry or could be a result of misreading results after a test. Features "ap_hi" and "ap_lo" represent systolic blood pressure and diastolic blood pressure. **Systolic blood pressure** is the top number on blood pressure reading. It measures the force of blood against one's artery walls while the ventricles (the lower two chambers of your heart) squeeze, pushing blood out to the rest of your body. **Diastolic blood pressure** is the bottom number on the blood pressure reading. It measures the force of blood against your artery walls as your heart relaxes and the ventricles can refill with blood. Diastole — this period when your heart relaxes between beats — is also the time that your coronary artery can supply blood to your heart. The higher the blood pressure, the more risk one has for other health problems such as heart disease, heart attack, strokes... problem in focus cardiovascular diseases.



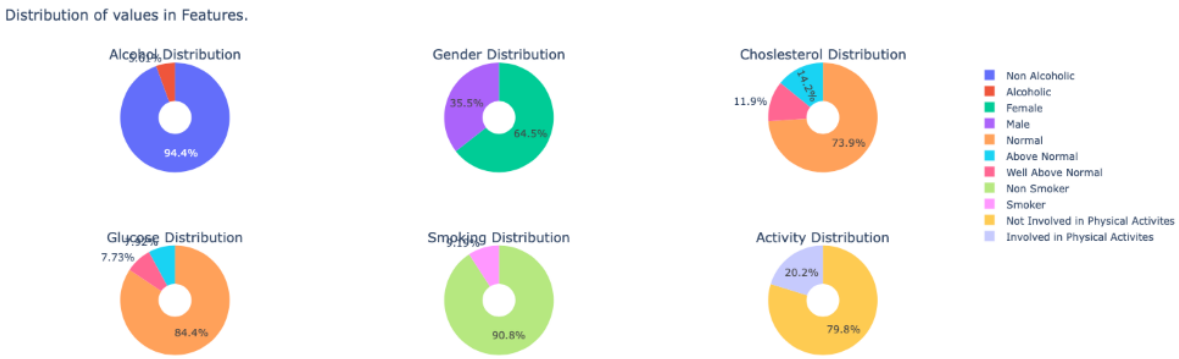
To handle outliers, we implemented Interquartile range. IQR is a measure of variability, based on dividing a data set into quartiles. Quartiles divide a rank-ordered data set into four equal parts. The values that separate parts are called the first, second, and third quartiles; and they are denoted by Q1, Q2, and Q3, respectively. The interquartile range is often used to find outliers in data. Outliers here are defined as observations that fall below **Q1 − 1.5 IQR** or above **Q3 + 1.5 IQR**. In a boxplot, the highest and lowest occurring value within this limit are indicated by whiskers of the box (frequently with an additional bar at the end of the whisker) and any outliers as individual points. In addition, we had to research more about blood pressure to understand our data well. As shown in the image above, CDC and other health organizations define that if systolic and diastolic are above 180 and / or 120 respectively then immediate medical attention should be taken. This played an important role in removal of outliers as we got rid of values which were greater that 250 and 200 for systolic and diastolic, respectively. After outliers were eliminated (dropped for the scope of our project), we were left with 65799 records.

Feature Engineering:
Feature “age” was measured in days, which led to an unusual distribution of values in the dataset which could have very well affected the performance of any models. To eliminate this, we calculated age in years (from days) and reassigned it to the feature “age”, giving us a more consistent distribution of values with most people belonging to the age group 49 to 60. Since the dataset had features like “weight” and “height”, we decided to use both said features in creating “BMI” (Body Mass Index) to experiment if BMI could have any significant effect on various model’s performance.

Data Analysis:
Once our data was clean and preprocessed, we decided to do some basic Exploratory Data Analysis to understand our data better, using a lot of visualizations. We used “matplotlib.pyplot” for plotting results (discussed later) and “plotly.graph_objects” for EDA.
The data is analyzed to gain further insights. Bivariate analysis is performed on categorical variables to examine their relationship with the target class.

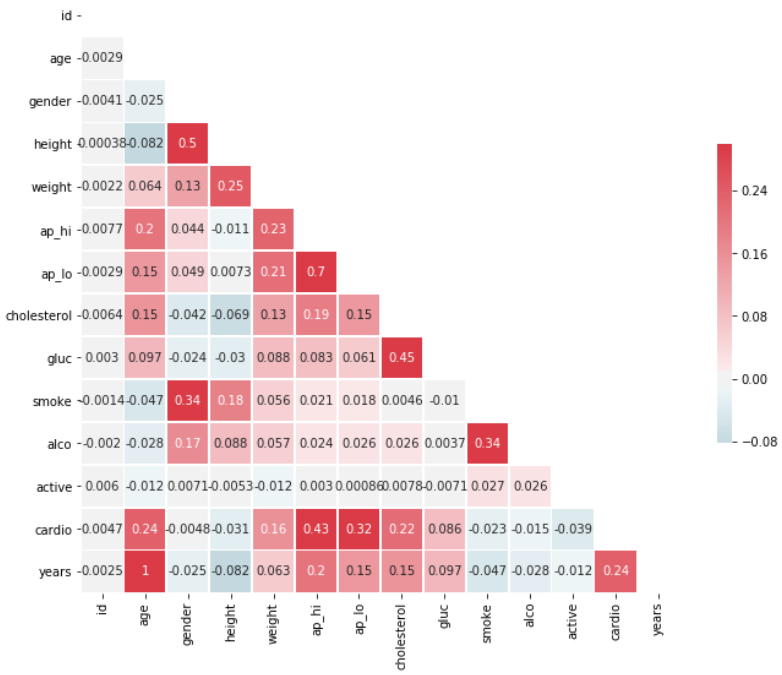


The plot above represents distribution of age groups grouped by target variable. It can be observed that till the age of 52, there is lesser risk of diagnosing CVD. From the age of 53 there is a clear increase in the number of people who get CVD which gives us an insight into age groups who might be prone to CVD. This could be helpful in preventing extreme cases in patients.



The plot above represents the distribution of values in categorical variables which were transformed and provided in the dataset. Many interesting insights can be drawn, we see that only 5.6% people consume alcohol. We also note that females comprise of more patients (64.5%) when compared to men (35.5%). It is also observed that 73.9% of the patients have a cholesterol level of normal, which could be a good indicator of their good eating habits.

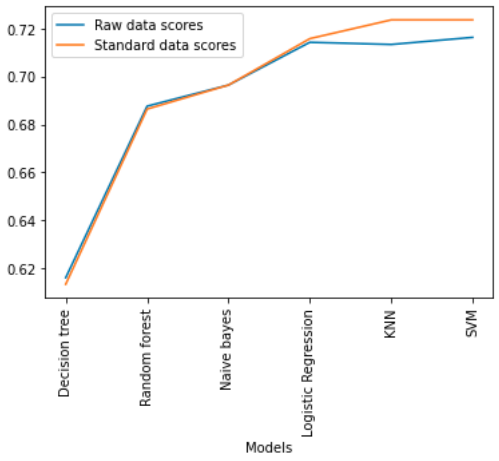
To further examine the relationships among variables, we plot a correlation matrix (fig. below). As we can see age and cholesterol have significant impact, but not very highly correlated with target class.



Following EDA, we moved forward to standardize data to ensure similar scale for our quantitative values. To ensure that there is no testing data leaked before the model is trained, we split our data before standardizing the features. Since we had a mixed bag of quantitative and encoded features, we only scaled numerical features. For this task

we used StandardScaler() from sklearn to transform our data, following it up by splitting data into training and testing data, 80% and 20% respectively. Once we had our data ready for modeling, we decided to compare a few machine learning methods to see which approach would be the best to help with our task. We wanted to compare results on unscaled data and standardized data (as shown in the plot and table below).

	Raw data scores	Standard data scores
Decision tree	0.616033	0.613298
Random forest	0.687614	0.686398
Naive bayes	0.696429	0.696429
Logistic Regression	0.714286	0.715805
KNN	0.713374	0.723632
SVM	0.716337	0.723632



Almost all models performed to a similar extent, but SVM, KNN and Logistic regression outperformed Naïve Bayes, Random Forrest and Decision Trees. When comparing results between unscaled and scaled data, we observed that performances went up for SVM, KNN and Logistic Regression.

Hyperparameter tuning:
To further improve our models the hyperparameters were tuned to achieve maximum performance. GridsearchCV() function from the sklearn package was used to run the model through a range of parameters and find the best values. This led to a small but noticeable improvement in our accuracy score. Through Gridsearch the number of estimators for Random forest algorithm was found to be 300. The updated model with this parameter improved its score from 0.6863 to 0.6903. KNN model showed an optimal 'n_neighbors' value of 150. Care should be taken when using hyperparameter tuning with grid search as the function bases its decisions on the training data and not the testing data. There is a chance for overfitting of the model.

Participants Contribution

Mrinal Soni: Researched into the problem statement, contacted Stanford / UK Data Bank for getting hold of Electronic Health Records, handled preprocessing (missing values, removal of outliers), used plotly to create interactive plots for Exploratory Data Analysis, implemented Logistic Regression, KNN, Support Vector Machines, edited final presentation, compared performance of models on unscaled and standardized data.
Yanbing Lin: Researched problem background and introduction, experimenting data preprocessing methods, implementation of Logistic Regression model and Naive Bayes. Managed project presentation as the presenter and edited PowerPoint for efficient visualization.
Sai Abhishikth Devabhaktuni: Researched problem background, handling duplicates, plots, implementing new features, removed outliers, KNN,

SVM, Naïve Bayes (Gaussian), Decision Trees, Random Forest, Neural Nets, Gradient Boosting. Created visualizations for the final report and edited it.

Future Work: There were a few challenges faced in the scope of this project, starting with acquiring the right data and then encountering the possibility that our data does not support our models properly. Which is why, one primary step in the future is to look for a more refined data that holds better features (hopefully more quantitative features) which would contribute greatly to our model. Next step would be to investigate Ensemble Learning again, we felt we could've done ensemble learning effectively if we had more time and probably built stronger model based on weak models. Lastly, we would like to see if other diseases or medical situations play a crucial role in determining a better prediction rate of predicting Cardiovascular Diseases.

References:

[1] Know the Differences: Cardiovascular Disease, Heart Disease, Coronary Heart Disease
[2] Study of cardiovascular disease prediction model based on random forest in eastern China
[3] Cardiovascular Disease dataset
[4] Heart Disease Prediction. Cleveland Heart Disease(UCI Repository)… | by Shubhankar Rawat
[5] https://datascience.stackexchange.com/questions/45900/when-to-use-standard-scaler-and-when-normalizer
[6] https://www.nature.com/articles/s41598-020-62133-5
[7] https://pubmed.ncbi.nlm.nih.gov/32023562/
[8] https://www.cdc.gov/heartdisease/index.htm
[9] https://towardsdatascience.com/preventing-data-leakage-in-your-machine-learning-model-9ae54b3cd1fb
[10]https://www.acc.org/latest-in-cardiology/articles/2016/08/03/13/47/a-comprehensive-review-of-predictive-risk-models-for-cardiovascular-disease
[**11**]https://www.healthline.com/health/high-blood-pressure-hypertension/blood-pressure-reading-explained