**Cardiovascular Risk Prediction**

**Ruchika Nayak and Om Prakash Pradhan**

**Data science trainees,**

**AlmaBetter, Bangalore**

**Abstract:**

Coronary heart disease (CHD) also known as coronary artery disease (CAD) is the most common type of heart disease in the United States. CHD is caused by plaque buildup in the walls of the arteries that supply blood to the heart. Overweight, physical inactivity, unhealthy eating, and smoking tobacco are some risk factors for CHD.

Our goal is to present a model that is able to automatically predict whether patients have risk of coronary heart disease based on their medical records. To achieve this, we have fitted various classification models to the given data and compare them. Data used include demographic, behavioral and medical risk factors and the dependent variable ten-year risk of CHD. From simple logistic regression model to different complex models like boosting are used in this project. To fine tune the results different hyper parameter combinations also tested using GridSearchCV.

***Keywords:* *Classification, Coronary heart disease(CHD), GridSearchCV***

**1. Problem Statement**

Cardiovascular risk dataset is taken from an ongoing cardiovascular study on residents of the town of Framingham, Massachusetts. The dataset provides the patients information. It includes over 3000 records and 15 attributes. Each attribute is a potential risk factor. There are both demographic, behavioral and medical risk factors.

The main classification goal is to predict whether the patient has a 10-year risk of future coronary heart disease (CHD)

The main objective is to model the data using different classification models and come up with more accurate model which will help in prediction of 10 year CHD risk.

**2.Introduction**

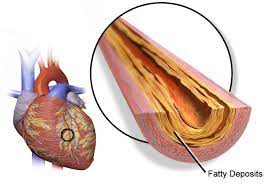
### Coronary Heart Disease is a type of heart disease that develops when the arteries of the heart cannot deliver enough oxygen rich blood to the heart. Cardiovascular risk is defined as the probability of suffering in the future from a clinical cardiovascular event. In our project there are many features which makes the cardiovascular risk, some features like age, cigsPerDay are very much important for 10 year risk of CHD.

### Here, for our features visualization and analyzation we have done EDA. From our EDA we came to know that our dataset contains many null values so for our null values treatment we have done imputation using KNN imputer. Later on in this project we have different models just to know which features greatly affect the patient and which patient have a risk of 10 year Coronary Heart Disease or CHD

## **3.Coronary Heart Disease**

### Coronary heart disease(CHD) is a narrowing or blockage of coronary arteries usually caused by the buildup of fatty material called plaque. Coronary heart disease is also called coronary artery disease, ischemic heart disease and heart disease.

### In some cases, when plaque breaks, a blood clot may block the supply to your heart muscle. This causes a heart attack. The damage may be caused by various factors including smoking, high blood pressure, high cholesterol, diabetes or insulin resistance, not being active (sedentary lifestyle) etc.



**4.Steps involved:**

* **Reading and understanding data**

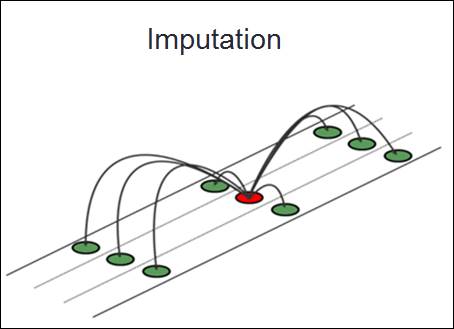
After reading the dataset which is in csv format we inspected the rows and data type of each columns. We have also explored number of unique values of each column and viewed some basic statistical details like percentile, mean, standard deviation etc. of numerical columns.

* **Null values and Duplicates**

Our dataset contains null values but there are no duplicate rows.

* **Null values treatment**

As there are null values in ‘education’, ‘cigsPerDay’, ‘BPMeds’, ‘totchol’, ‘BMI’, ‘heartrate’ and ‘glucose’ features, instead of dropping those columns we have used KNN imputer to impute those null values and replace it with the nearest values.



* **Exploratory Data Analysis (EDA)**

After loading the dataset, in EDA we have checked all the assumptions required for model fitting like correlation between variables, Gaussian distribution etc. We have also found out the features affecting the dependent variable by using various visualization tools.

* **Encoding of categorical columns**

We have applied label encoding for ‘sex’ and ‘is\_smoking’ features as it comprises of only two distinct labels.

* **Feature engineering**

We have created a new feature ‘BP’ by taking average of two features i.e. ‘diaBP’ and ‘sysBP’. Then we have applied log transformation on all the continuous features as all of them are right skewed. To select the most relevant features we have used correlation matrix. We have also dropped some features like ‘id’, ‘education’, ‘is\_smoking’, ‘sysBP’ and ‘diaBP’.

* **Data preparation for model building**

We have used Standard Scalar (z-score) which will standardize features by removing the mean and scaling to unit variance. For modelling we split our data into two parts. The first part will be used for fitting the model, which is 80% of our data and second part is used for testing the model.

* **Fitting different models:**

For modelling we have used various models like:

1. Logistic Regression
2. Decision Tree
3. Random Forest
4. XGBoost
5. KNN
6. Support Vector Machine

* **Tuning the hyper parameters for better accuracy**

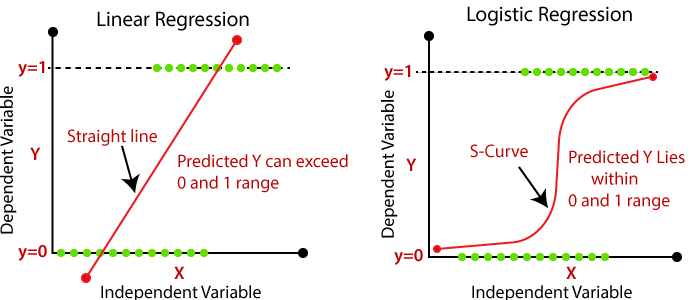
Tuning the hyper parameters of respective algorithms is necessary for getting better accuracy and to avoid overfitting specially in case of tree based models.

**5.1 Algorithms:**

For our data visualization and analysis, we have used the following visualization tools.

* **Logistic Regression**:

The logistic regression is a process of modelling the probability of a discrete outcome given an input variable. Logistic regression is a useful analysis method for classification problems, where you are trying to determine if a new sample fits best into a category.

****

* **Decision Tree:**

Decision trees are a type of Supervised Machine learning where the data continuously split according to a certain parameter. Decision tree can be used for classification as well as regression problems. The name itself suggest that it uses a flowchart like a tree structure to show the predictions that result from a series of feature based splits. It starts with a root node and ends with a decision made by leaves.



* **Random Forest:**

Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. It performs better results for classification problems. It builds decision trees on different samples and take their majority vote for classification and average in case of regression.



* **XGBoost:**

XGBoost stands for Extreme Gradient Boosting, is a scalable, distributed gradient-boosted decision tree machine learning library. It provides parallel tree boosting and is the leading machine learning library for regression, classification and ranking problems. It dominates structured or tabular datasets on classification and regression predictive modelling problems.

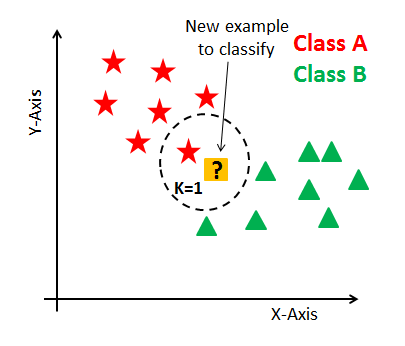
* **Support Vector Machine:**

A support vector machine is a supervised machine learning model that uses classification algorithms for two group classification problems. After giving an SVM model sets of labeled training data for each category, they are able to categorize new text. Its goal is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.



* **K-nearest neighbor(KNN):**

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. KNN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.



**5.2. Model performance:**

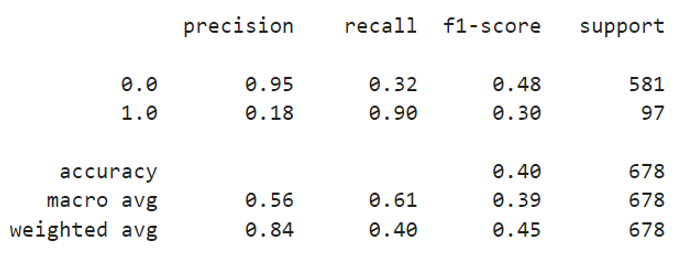
Classification model can be evaluated by using following metrics

* **Confusion Matrix:** Confusion matrices are used to visualize important predictive analytics like recall, specificity, accuracy, and precision. Confusion Matrices are useful because they give direct comparisons of values like True Positive, False Positives, True Negatives and False Negatives.



1. **True Positive:** It is an outcome where the model correctly predicts the positive class.
2. **True Negative:** It is an outcome where the model correctly predicts the negative class.
3. **False Positive:** It is an outcome where the model incorrectly predicts the positive class.
4. **False Negative:** It is an outcome where the model incorrectly predicts the negative class.

* **Classification Report:** A classification report is used to measure the quality of predictions from a classification algorithm. The report shows the main classification metrics precision, recall and f1-Score on a per class basis. The metrics are calculated by using true and false positives, true and false negatives.



1. **Precision:** It is the ability of a classifier not to label an instance positive that is actually negative. It is defined as the ratio of true positives to the sum of a true positive and false positive.
2. **Recall:** It is the ability of a classifier to find all positive instances. For each class it is defined as the ratio of true positives to the sum of true positives and false negatives.
3. **F1-Score:** It is a weighted harmonic mean of precision and recall such that the best score is 1.0 and the worst score is 0.0.
4. **Support:** Support is the number of actual occurrences of the class in the specified dataset. Imbalanced support in the training data may indicate structural weakness in the reported scores of the classifier and could indicate the need for stratified sampling or rebalancing.

**5.3. Hyper-parameter Tuning**

Hyper-parameter optimization or tuning is the problem of choosing a set of optimal hyper-parameters for a learning algorithm. A hyper-parameter is a parameter whose value is used to control the learning process. Hyper-parameters alter the way a model learns to trigger this training algorithm after parameters to generate outputs. We have used Grid Search CV for hyper-parameter tuning.

**GridSearchCV:**

Grid Search combines a selection of hyper-parameters established by the scientist and runs through all of them to evaluate the model’s performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific region of the parameter space and cannot understand which movement or which region of the space is important to optimize the model.

**6.Conclusion:**

So, that’s it. We reached the end of our exercise. Starting from the data loading and reading the data so far we have done EDA, encoding of categorical columns, KNN imputation for null values treatment, feature selection and then different model building.

From all these process we came to know that age and cigsPerDay greatly affect all the models. Logistic regression, random forest and support vector machine gave us a good balanced result.

**7. Challenges Faced:**

* Handling null values
* Dealing with missing values
* Selecting most relevant features.
* Selecting relevant set of hyper parameters for tuning.
* Computation time during GridSearchCV.

**References-**

1. GeeksforGeeks
2. Wikipedia
3. Kaggle
4. Javatpoints