**CARDIOVASCULAR RISK PREDICTION  
 MACHINELEARNING(*CLASSIFICATION*)**

**By Md Amanatullah**

**Data Science Trainee**

**Alma better, Bangalore**

**Abstract:**

Cardiovascular diseases (CVD) are among the most common serious illnesses affecting human health. CVDs may be prevented or mitigated by early diagnosis, and this may reduce mortality rates. Identifying risk factors using machine learning models is a promising approach. We would like to propose a model that incorporates different methods to achieve effective prediction of heart disease. For our proposed model to be successful, we have used efficient Data Collection, Data Pre-processing and Data Transformation methods to create accurate information for the training model.

New hybrid classifiers like Decision Tree Bagging Method (DTBM), Random Forest Bagging Method (RFBM), K-Nearest Neighbours, Bagging Method (KNNBM), AdaBoost Boosting Method (ABBM), and Gradient Boosting Boosting Method (GBBM) are developed by integrating the traditional classifiers with bagging and boosting methods, which are used in the training process. We have also instrumented some machine learning algorithms to calculate the Accuracy (ACC), Sensitivity (SEN), Error Rate, Precision (PRE) and F1 Score (F1) of our model, along with the Negative Predictive Value (NPR), False Positive Rate (FPR), and False Negative Rate (FNR). The results are shown separately to provide comparisons. Based on the result analysis, we can conclude that our proposed model produced the highest accuracy while using RFBM and Relief feature selection methods (99.05%).

**Keywords:** *EDA, Correlation, Decision Tree*

*Random Forest, Bagging, Boosting, KNN, Logistic Regression, SVM Forecasting.*

**Introduction:-**

Cardiovascular disease has been regarded as the most severe and lethal disease in humans. The increased rate of cardiovascular diseases with a high mortality rate is causing significant risk and burden to the healthcare systems worldwide. Cardiovascular diseases are more seen in men than in women particularly in middle or old age , although there are also children with similar health issues . According to data provided by the WHO, one-third of the deaths globally are caused by the heart disease. CVDs cause the death of approximately 17.9 million people every year worldwide and have a higher prevalence in Asia.

**Approach:**

The approach followed here is to first check the sanctity of the data and then understand the features involved. The events followed were in our approach:

* **Understanding the business problem and the datasets.**
* **Data cleaning and pre-processing: -**finding null values and imputing them with appropriate values. Converting categorical values into appropriate data types and merging the datasets provided to get a final dataset to work upon.
* **Exploratory data analysis** of categorical and continuous variables against our target variable.
* **Data manipulation-** feature selection and engineering, feature scaling, outlier detection and treatment and encoding categorical features.
* **Modelling-** The baseline model Decision tree was chosen considering our features were mostly categorical with few having continuous importance.
* **Model Performance and Evaluation**
* **Conclusion and Recommendations**

**Problem Statement: -**

Predicting coronary heart disease in advance helps raise awareness for the disease. Preventive measurements like changing diet plans and exercise can slow down the progression of CHD. Early prediction can result in early diagnosis. So, we can treat the disease at an early stage and avoid more invasive treatment.

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

**Understanding the Data:**

First step involved is understanding the data and getting answers to some basic questions like; What is the data about? How many rows or observations are there in it? How many features are there in it? What are the data types? Are there any missing values? And anything that could be relevant and useful to our investigation. Let's just understand the dataset first and the terms involved before proceeding further. The data types were of integer, float and object in nature.

**Data Description:**

The dataset provides the patients’ information. It includes over 3390 records and 17 attributes. Variables Each attribute is a potential risk factor. There are both demographic, behavioural, and medical risk factors.

**Demographic:**

• Sex: male or female("M" or "F")

• Age: Age of the patient

**Behavioural:**

**• is\_smoking**: whether the patient is a current smoker ("YES" or "NO")

**• Cigs Per Day**: the number of cigarettes that the person smoked on average in one day

**Medical(History):**

**• BP Meds:** whether the patient was on blood pressure medication

**• Prevalent Stroke:** whether the patient had previously had a stroke

**• Prevalent Hyp**: whether the patient was hypertensive

**• Diabetes:** whether the patient had diabetes (Nominal) Medical(current)

**• Tot Chol:** total cholesterol level

**• Sys BP:** systolic blood pressure

**• Dia BP:** diastolic blood pressure

**• BMI:** Body Mass Index

**• Heart Rate:** heart rate

**• Glucose:** glucose level

**• CHD: 10-year** risk of coronary heart disease CHD(binary: “1”, means “Yes”, “0” means “No”)

**Data Cleaning and Pre-processing:**

Handling missing values is an important skill in the data analysis process. If there are very few missing values compared to the size of the dataset, we may choose to drop rows that have missing values. Otherwise, it is better to replace them with appropriate values. It is necessary to check and handle these values before feeding it to the models, to obtain good insights on what the data is trying to say and make great characterisation and predictions which will in turn help improve the business's growth. The real-world data often has a lot of missing values. If we want our model to work unbiased and accurately then we just can’t ignore the part of “missing value” in our data. One of the most common problems faced in data cleansing or pre-processing is handling missing values.

**What is Missing Data?**

Missing data means absence of observations in columns. It appears in values such as “0”, “NA”, “NaN”, “NULL”, “Not Applicable”, “None”.

**Why has dataset Missing values?**

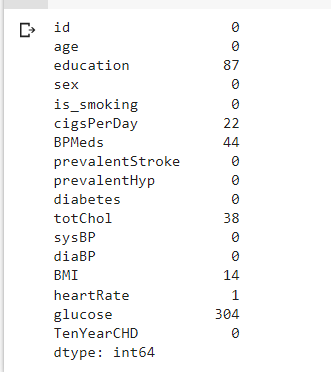
The cause of it can be data corruption ,failure to record data, lack of information, incomplete results ,person might not provide the data intentionally ,some system or equipment failure etc. There could any reason for missing values in your dataset.

**Why to handle Missing values?**

One of the biggest impact of Missing Data is, It can bias the results of the machine learning models or reduce the accuracy of the model. So, It is very important to handle missing values.

**How to check Missing Data?**

The first step in handling missing values is to look at the data carefully and find out all the missing values. In order to check missing values in Python Pandas Data Frame, we use a function like isnull() and notnull() which help in checking whether a value is “NaN”(True) or not and return Boolean values.



**How to handle Missing data?**

Missing values can be handled in different ways depending on, if the missing values are continuous or categorical. Because method of handling missing values are different between these two data type .By using “dtypes” function in python we can filter our columns from dataset.

THREE WAYS to treat missing values in dataset are as follows:

* [DROPPING](https://medium.com/p/98e47f025b9c/edit/#10db)
* [IMPUTION](https://medium.com/p/98e47f025b9c/edit/#7da3)
* [PREDICTIVE MODEL](https://medium.com/p/98e47f025b9c/edit/#0b54)

## Dropping missing values

This method is commonly used to handle null values. It is easy to implement and there is no manipulation of data required. This varies from case to case on the amount of information we think the variable has. If dataset information is valuable or training dataset has a smaller number of records then deleting rows might have negative impact on the analysis. Deletion methods works great when the nature of missing data is missing completely at random(MCAR) but for non-random missing values can create a bias in the dataset, if a large amount of a particular type of variable is deleted from it.

**General problem**: Method of handling missing values between two data type such as continuous data and categorical data are different.

1. Missing values in continuous data can be solved by imputing with mean ,median ,mode or with multiple imputation
2. Missing values in categorical data can be solved by mode ,multiple imputation
3. **Other Imputation Methods:**

There are many other imputation techniques to impute missing values, few are given below in addition to above methods.

**Multiple imputation** is flexible and essentially an iterative form of stochastic imputation. It is statistical technique for handing missing data. It preserves sample size and statistical power

**KNN Imputer** or **Iterative Imputer** classes to impute missing values considering the multivariate approach. In a multivariate approach, more than one feature is taken into consideration.

**Arbitrary Value Imputation** is an important technique used in Imputation as it can handle both the Numerical and Categorical variables.

# **Handling Skewed data:-**

# Data is skewed when its distribution curve is asymmetrical (as compared to a normal distribution curve that is perfectly symmetrical) and skewness is the measure of the asymmetry. The skewness for a normal distribution is 0.There are 2 different types of skews in data, left(negative) or right(positive) skew. Chart Description automatically generated with medium confidence

**Effects of skewed data:** Degrades the model’s ability (especially regression-based models) to describe typical cases as it has to deal with rare cases on extreme values. i.e. right skewed data will predict better on data points with lower value as compared to those with higher values. Skewed data also does not work well with many statistical methods. However, **tree-based models are not affected**.

To ensure that the machine learning model capabilities is not affected, skewed data must be transformed to approximate to a normal distribution. The method used to transform the skewed data depends on the characteristics of the data.

**Dealing with skew data:**

**1**.log transformation: transform skewed distribution to a normal distribution

Not able to log 0 or negative values (add a constant to all value to ensure values > 1)

**2**.Remove outliers

**3**.Normalize (min-max)

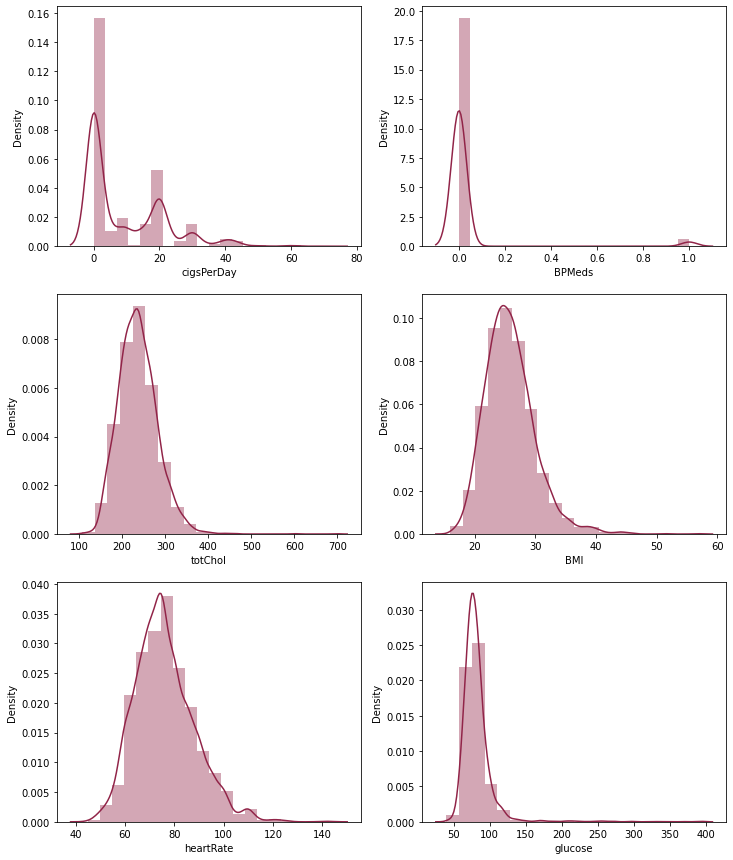
**4**.Cube root: when values are too large. Can be applied on negative values

**5**.Square root: applied only to positive values

**6**.Reciprocal

**7**.Square: apply on left skew

**Let’s check the skewness in cardiovascular datasets.**



the skewness of the column cigsPerDay is: 1.2230053709053774

the skewness of the column BPMeds is: 5.524325007968017

the skewness of the column totChol is: 0.9406357047700903

the skewness of the column BMI is: 1.0222520011438563

the skewness of the column heartRate is: 0.6764897223370003

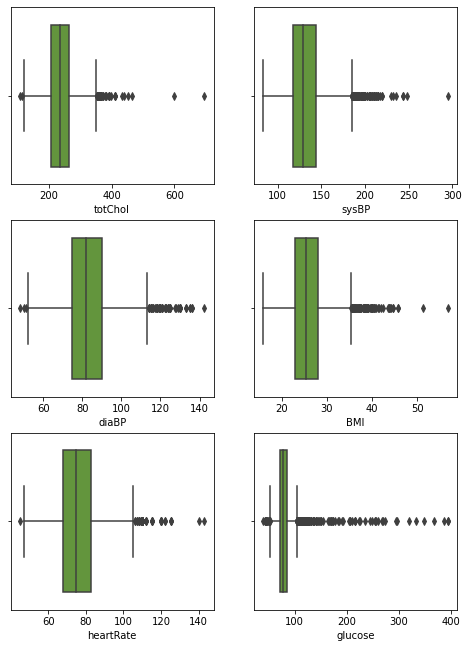
the skewness of the column glucose is: 6.1443896544049394

If skewness value is between:-0.5 and 0.5, the distribution of the value is almost symmetrical

-1 and -0.5, the data is negatively skewed, and if it is between 0.5 to 1, the data is positively skewed.

If the skewness is lower than -1 (negatively skewed) or greater than 1 (positively skewed), the data is highly skewed.

It seems like most of the values of the Columns are towards the left and the distribution is skewed on the right. Median is more robust to outlier effect hence median was imputed in the null values. Right skewed distributions occur when the long tail is on the right side of the distribution also called as positive skewed distribution which essentially suggests that there are positive outliers far along which influences the mean. Boxplot is an effective method to detect outliers.



# **Exploratory Data Analysis(EDA)**

**Exploratory Data Analysis (EDA)**is an approach to analyse the data using visual techniques. It is used to discover trends, patterns, or to check assumptions with the help of statistical summary and graphical representations.

## Getting insights about the dataset

* Let’s see the shape of the data using the shape.

df.shape (3390, 17)

* df.info()

0 id 3390 non-null int64

1 age 3390 non-null int64

2 education 3303 non-null float64

3 sex 3390 non-null object

4 is\_smoking 3390 non-null object

5 cigsPerDay 3368 non-null float64

6 BPMeds 3346 non-null float64

7 prevalentStroke 3390 non-null int64

8 prevalentHyp 3390 non-null int64

9 diabetes 3390 non-null int64

10 totChol 3352 non-null float64

11 sysBP 3390 non-null float64

12 diaBP 3390 non-null float64

13 BMI 3376 non-null float64

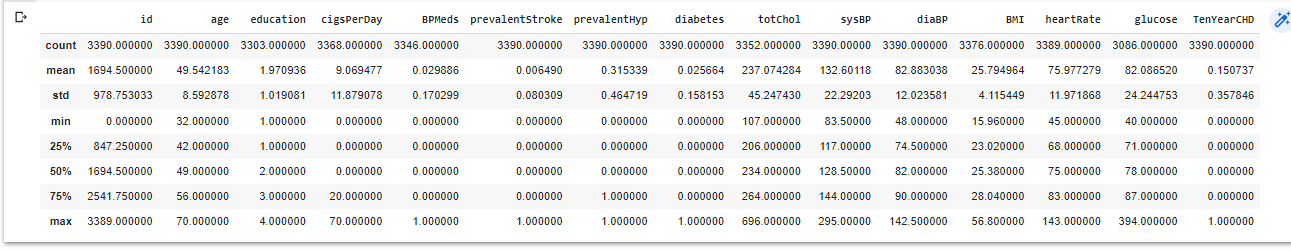
14 heartRate 3389 non-null float64

15 glucose 3086 non-null float64

16 TenYearCHD 3390 non-null int64

dtypes: float64(9), int64(6), object(2)

memory usage: 450.4+ KB

* df.describe()

The describe() function applies basic statistical computations on the dataset like extreme values, count of data points standard deviation, etc. Any missing value or NaN value is automatically skipped. describe() function gives a good picture of the distribution of data.

**Data visualization**

Data Visualization is the process of analysing data in the form of graphs or maps, making it a lot easier to understand the trends or patterns in the data. There are various types of visualizations –

* **Univariate analysis:** This type of data consists of only one variable. The analysis of univariate data is thus the simplest form of analysis since the information deals with only one quantity that changes. It does not deal with causes or relationships and the main purpose of the analysis is to describe the data and find patterns that exist within it.
* **Bi-Variate analysis:** This type of data involves two different variables. The analysis of this type of data deals with causes and relationships and the analysis is done to find out the relationship among the two variables.
* **Multi-Variate analysis:** When the data involves three or more variables, it is categorized under multivariate.

***We will use Matplotlib and Seaborn library for the data visualization****.*

**Approach:**

There are two kinds of features in the dataset: Categorical and Non-Categorical Variables.

Categorical- A categorical variable is a variable that can take on one of a limited, and usually fixed, number of possible values putting a a particular category to the observation.

Non-Categorical- A non-categorical or continuous variable is a variable whose value is obtained by measuring, i.e. ., one which can take on an uncountable set of values. Both are analysed separately. Categorical data is usually analysed through count plots and bar plots in accordance with the target variable and that is what is done here too. On the other hand, Numeric or Continuous variables were analysed through distribution plots, box plots and scatterplots to get useful insights.

### **Histogram**

It can be used for both univariate and bivariate analysis.Calendar

Description automatically generated

From the above distribution plot it can be observed that there are some columns which are not good indicator of CHD as they have imbalanced data like 'BPMeds', 'prevalentStroke' , 'diabetes'...so it’s better to drop these columns.

### **Boxplot**

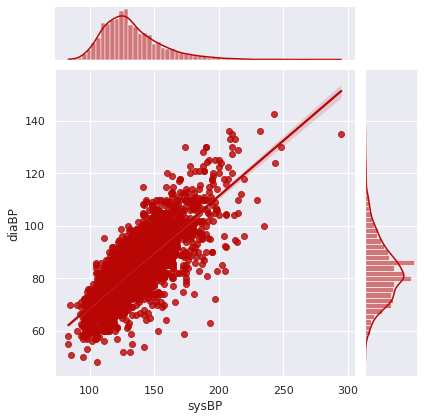
It can also be used for univariate and bivariate analysis.Chart, box and whisker chart

Description automatically generated

It can be observed that whether male and female with higher age are more prone to CHD

### **Joint Plot**

It can be used for bivariate analyses.



We can clearly see that sysBP and diaBP are positively correlated and there are many people who have higher value of sysBP and diaBP than normal range indicating more prone to chronic heart diseases. And also the value of diaBP is lower than sysBP for a particular person which is a generalised behaviour.

**Count Plot:-**

Show the counts of observations in each categorical bin using bars. A count plot can be thought of as a histogram across a categorical, instead of quantitative, variable.

#counting the no. of people whether they are at risk of CHD or not given that they are diabetic or not.

**Chart, bar chart

Description automatically generated**

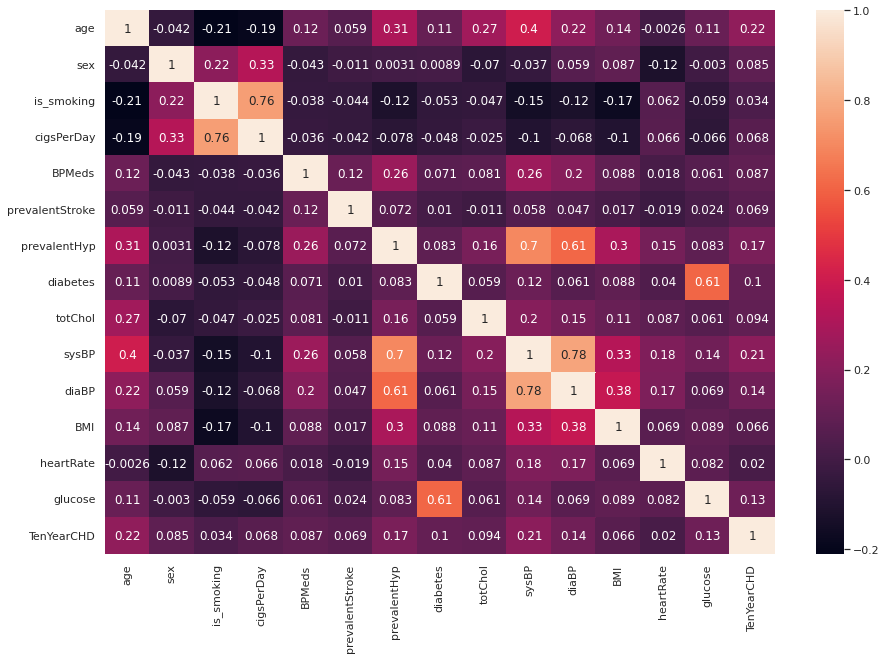
From the above plot it can be interpreted that non-diabetic person is less prone to CHD although there are about 500 people who are non-diabetic but are at risk of CHD indicating the effect of some other factors.

**Correlation:**

Correlation is a statistical term used to measure the degree in which two variables move in relation to each other. A perfect positive correlation means that the correlation coefficient is exactly 1. This implies that as one variable moves, either up or down, the other moves in the same direction. A perfect negative correlation means that two variables move in opposite directions, while a zero correlation implies no linear relationship at all.

## Correlation heatmap

A correlation heatmap is a heatmap that shows a 2D correlation matrix between two discrete dimensions, using coloured cells to represent data from usually a monochromatic scale. The values of the first dimension appear as the rows of the table while of the second dimension as a column. The colour of the cell is proportional to the number of measurements that match the dimensional value. This makes correlation heatmaps ideal for data analysis since it makes patterns easily readable and highlights the differences and variation in the same data. A correlation heatmap, like a regular heatmap, is assisted by a colorbar making data easily readable and comprehensible.



* **Conclusion from Heatmap: -**

**1.sysBP and diaBP is positively correlated**

* **Systolic Blood Pressure.** The normal range of systolic blood pressure should be 90 – 120 mm Hg.
* **Diastolic Blood Pressure.** The normal range of diastolic blood pressure should be 60 – 80 mm Hg.

IF the systolic pressure is high the diastolic pressure should be low.

**2.Glucose and diabetes are positively correlated.**

* Diabetes is a disease that occurs when your blood glucose, also called blood sugar, is too high.

**3.is\_smoking and age is negatively correlated** indicating older age People have less habit of smoking.

**4.prevalentHyp and sysBP is positively correlated.**

**Feature Scaling:**

Feature Scaling is a technique to standardize the independent features present in the data in a fixed range. It is done to prevent biased nature of machine learning algorithms towards features with greater values and scale. The two techniques are:

**Normalization**: is a scaling technique in which values are shifted and rescaled so that they end up ranging between 0 and 1. It is also known as Min-Max scaling. [0,1]

**Standardization**: is another scaling technique where the values are centred around the mean with a unit standard deviation. This means that the mean of the attribute becomes zero and the resultant distribution has a unit standard deviation. [-1,1] .

Normalization of the continuous variables was done further.

**Modelling:-**

# **Classification Algorithm in Machine Learning**

As we know, the Supervised Machine Learning algorithm can be broadly classified into Regression and Classification Algorithms. In Regression algorithms, we have predicted the output for continuous values, but to predict the categorical values, we need Classification algorithms.

## What is the Classification Algorithm?

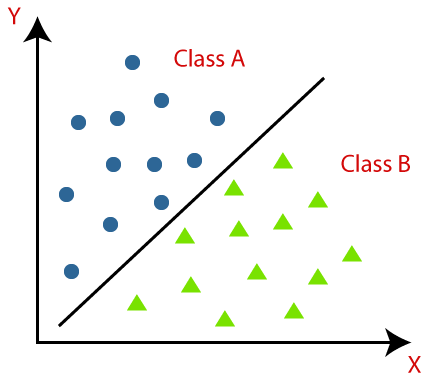
The Classification algorithm is a Supervised Learning technique that is used to identify the category of new observations on the basis of training data. In Classification, a program learns from the given dataset or observations and then classifies new observation into a number of classes or groups. Such as, **Yes or No, 0 or 1, Spam or Not Spam, cat or dog,** etc. Classes can be called as targets/labels or categories.

Unlike regression, the output variable of Classification is a category, not a value, such as "Green or Blue", "fruit or animal", etc. Since the Classification algorithm is a Supervised learning technique, hence it takes labelled input data, which means it contains input with the corresponding output.

In classification algorithm, a discrete output function(y) is mapped to input variable(x).

The main goal of the Classification algorithm is to identify the category of a given dataset, and these algorithms are mainly used to predict the output for the categorical data.

Classification algorithms can be better understood using the below diagram. In the below diagram, there are two classes, class A and Class B. These classes have features that are like each other and dissimilar to other classes.



The algorithm which implements the classification on a dataset is known as a classifier. There are two types of Classifications:

* **Binary Classifier:** If the classification problem has only two possible outcomes, then it is called as Binary Classifier.  
  **Examples:** YES or NO, MALE or FEMALE, SPAM or NOT SPAM, CAT or DOG, etc.
* **Multi-class Classifier:** If a classification problem has more than two outcomes, then it is called as Multi-class Classifier.  
  **Example:** Classifications of types of crops, Classification of types of music.

Learners in Classification Problems:

In the classification problems, there are two types of learners:

1. **Lazy Learners:** Lazy Learner firstly stores the training dataset and wait until it receives the test dataset. In Lazy learner case, classification is done on the basis of the most related data stored in the training dataset. It takes less time in training but more time for predictions.  
   **Example:** K-NN algorithm, Case-based reasoning
2. **Eager Learners:** Eager Learners develop a classification model based on a training dataset before receiving a test dataset. Opposite to Lazy learners, Eager Learner takes more time in learning, and less time in prediction. **Example:** Decision Trees, Naïve Bayes, ANN.

Types of ML Classification Algorithms:

Classification Algorithms can be further divided into the Mainly two category:

* **Linear Models**
  + Logistic Regression
  + Support Vector Machines
* **Non-linear Models**
  + K-Nearest Neighbours
  + Kernel SVM
  + Naïve Bayes
  + Decision Tree Classification
  + Random Forest Classification

## Evaluating a Classification model:

Once our model is completed, it is necessary to evaluate its performance; either it is a Classification or Regression model. So for evaluating a Classification model, we have the following ways:

1. **Log Loss or Cross-Entropy Loss**: (ylog(p)+(1-y)log(1-p))

Where y= Actual output, p= predicted output.

* It is used for evaluating the performance of a classifier, whose output is a probability value between the 0 and 1.
* For a good binary Classification model, the value of log loss should be near to 0.
* The value of log loss increases if the predicted value deviates from the actual value.
* The lower log loss represents the higher accuracy of the model.

**2. Confusion Matrix:**

* The confusion matrix provides us a matrix/table as output and describes the performance of the model.
* It is also known as the error matrix.
* The matrix consists of predictions result in a summarized form, which has a total number of correct predictions and incorrect predictions. The matrix looks like as below table:

|  |  |  |
| --- | --- | --- |
|  | **Actual Positive** | **Actual Negative** |
| Predicted Positive | True Positive | False Positive |
| Predicted Negative | False Negative | True Negative |

Classification Algorithm in Machine Learning

**3. AUC-ROC curve:**

* ROC curve stands for **Receiver Operating Characteristics Curve** and AUC stands for **Area Under the Curve**.
* It is a graph that shows the performance of the classification model at different thresholds.
* To visualize the performance of the multi-class classification model, we use the AUC-ROC Curve.
* The ROC curve is plotted with TPR and FPR, where TPR (True Positive Rate) on Y-axis and FPR(False Positive Rate) on X-axis.
* **Train-Test Split:**
* In machine learning, train/test split splits the data randomly, as there's no dependence from one observation to the other. That's not the case with time series data. Here, it's important to use values at the rear of the dataset for testing and everything else for training.

**Baseline Model (Decision Tree):-**

A decision tree is a flowchart-like structure in which each internal node represents a test on a feature (e.g. whether a coin flip comes up heads or tails) , each leaf node represents a class label (decision taken after computing all features) and branches represent conjunctions of features that lead to those class labels. The paths from root to leaf represent classification rules. Below diagram illustrate the basic flow of decision tree for decision making with labels (Rain(Yes), No Rain(No)).

**Diagram

Description automatically generated**

Decision trees are constructed via an algorithmic approach that identifies ways to split a data set based on different conditions. It is one of the most widely used and practical methods for supervised learning. Decision Trees are a [non-parametric](https://machinelearningmastery.com/parametric-and-nonparametric-machine-learning-algorithms/) **supervised learning** method used for both **classification**and **regression**tasks.

Tree models where the target variable can take a discrete set of values are called **classification trees**. Decision trees where the target variable can take continuous values (typically real numbers) are called **regression trees**. Classification And Regression Tree (CART) is general term for this.

**How does the Tree decide which variable to branch out at each level?**

**Variable selection criterion**

Here is where the true complexity and sophistication of decision lies. Variables are selected on a complex statistical criterion which is applied at each decision node. Now, variable selection criterion in Decision Trees can be done via two approaches:

1. **Entropy and Information Gain**

2**. Gini Index**

Both criteria are broadly similar and seek to determine which variable would split the data to lead to the underlying child nodes being most homogenous or pure. Both are used in different Decision Tree algorithms. To add to the confusion, it is not clear which one is the preferred approach. So, one must understand both.

Let us begin with Entropy and Information Gain criterion

**What is Entropy?**

Entropy is a term that comes from physics and means a measure of disorder.

Entropy is measured by the formula:

Shape

Description automatically generated with medium confidence

Where the pi is the probability of randomly selecting an example in class i.

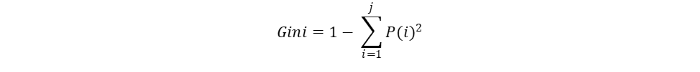
The change in entropy is termed Information Gain and represents how much information a feature provides for the target variable.



**Gini Index**

The other way of splitting a decision tree is via the Gini Index. The Entropy and Information Gain method focuses on purity and impurity in a node. The Gini Index or Impurity measures the probability for a random instance being misclassified when chosen randomly. The lower the Gini Index, the better the lower the likelihood of misclassification.

The formula for Gini Index



Where j represents the no. of classes in the target variable — Pass and Fail in our example

P(i) represents the ratio of Pass/Total no. of observations in node.

## ****Decision Tree Classifiers In Cardiovascular datasets:-****

dt\_classifier=DecisionTreeClassifier(criterion='entropy', max\_leaf\_nodes=20, random\_state=0)

dt\_classifier.fit(X\_train,y\_train)

DecisionTreeClassifier(criterion='entropy', max\_leaf\_nodes=20, random\_state=0)

accuracy\_score(y\_test,pred\_y\_test)

0.6689814814814815

Chart, treemap chart

Description automatically generated

## ****Decision Tree with Hyperparameter Tuning****

dt\_hp = DecisionTreeClassifier(random\_state=43)

params = {'max\_depth':[3,5,7,10,15],

          'min\_samples\_leaf':[3,5,10,15,20],

          'min\_samples\_split':[8,10,12,18,20,16],

          'criterion':['gini','entropy']}

GS = GridSearchCV(estimator=dt\_hp,param\_grid=params,cv=5,n\_jobs=-1, verbose=True, scoring='accuracy')

dt\_hp1 = DecisionTreeClassifier(criterion='gini',

 max\_depth=10,

 min\_samples\_leaf=3,

 min\_samples\_split=8,random\_state=43)

# **Logistic Regression:-**

Logistic regression is basically a supervised classification algorithm. In a classification problem, the target variable(or output), y, can take only discrete values for a given set of features(or inputs), X.  The model builds a regression model to predict the probability that a given data entry belongs to the category numbered as “1”. Just like Linear regression assumes that the data follows a linear function, Logistic regression models the data using the sigmoid function.

Chart

Description automatically generated

Logistic regression becomes a classification technique only when a decision threshold is brought into the picture. **.**

**Low Precision/High Recall:** In applications where we want to reduce the number of false negatives without necessarily reducing the number of false positives, we choose a decision value that has a low value of Precision or a high value of Recall. For example, in a cancer diagnosis application, we do not want any affected patient to be classified as not affected without giving much heed to if the patient is being wrongfully diagnosed with cancer. This is because the absence of cancer can be detected by further medical diseases, but the presence of the disease cannot be detected in an already rejected candidate.  
**2. High Precision/Low Recall:** In applications where we want to reduce the number of false positives without necessarily reducing the number of false negatives, we choose a decision value that has a high value of Precision or a low value of Recall. For example, if we are classifying customers whether they will react positively or negatively to a personalized advertisement, we want to be absolutely sure that the customer will react positively to the advertisement because otherwise, a negative reaction can cause a loss of potential sales from the customer.

## Based on the number of categories, Logistic regression can be classified as:

1. **binomial:** target variable can have only 2 possible types: “0” or “1” which may represent “win” vs “loss”, “pass” vs “fail”, “dead” vs “alive”, etc.
2. **multinomial:** target variable can have 3 or more possible types which are not ordered(i.e. types have no quantitative significance) like “disease A” vs “disease B” vs “disease C”.
3. **ordinal:** it deals with target variables with ordered categories. For example, a test score can be categorized as: “very poor”, “poor”, “good”, “very good”. Here, each category can be given a score like 0, 1, 2, 3.

## In Logistic Regression, the hypothesis we used for prediction is:

## 

## 

## is called **logistic function** or the **sigmoid function**.

We can infer from the above equation that: 

* g(z) tends towards 1 as z tends to infinity
* g(z) tends towards 0 as z tends to -infinity
* g(z) is always bounded between 0 and 1

So, now, we can define conditional probabilities for 2 labels(0 and 1) for observation as:



We can write it more compactly as:



Now, we define another term, **likelihood of parameters** as:

Diagram, text, schematic

Description automatically generated

And for easier calculations, we take**log-likelihood:**

Text

Description automatically generated

The **cost function** for logistic regression is proportional to the inverse of the likelihood of parameters. Hence, we can obtain an expression for cost function, J using log-likelihood equation as:

A picture containing text, device, gauge

Description automatically generated

and our aim is to estimate beta so that cost function is minimized !!

**Logistic Regression in Cardiovascular datasets:-**

logit= LogisticRegression(fit\_intercept=True, max\_iter=10000)

logit.fit(X\_train, y\_train)

LogisticRegression(max\_iter=10000)

The accuracy on train data is 0.6595533498759305

The accuracy on test data is 0.6707175925925926

Treemap chart

Description automatically generated

**LogisticRegression with hyperparameter tuning.**

param\_grid=[{'penalty' : ['l1', 'l2', 'elasticnet', 'none'],

    'C' : np.logspace(-4, 4, 20),

    'solver' : ['lbfgs','newton-cg','liblinear','sag','saga'],

    'max\_iter' : [100, 1000,2500, 5000]

    }

]

clf=GridSearchCV(logit,param\_grid=param\_grid,cv=10)

clf.fit(X\_train,y\_train)

creating new logisticregression model with best hyperparameters and fitting the data into it

clf1=LogisticRegression(C=0.004832930238571752,max\_iter=100,penalty= 'l2',solver='lbfgs')

clf1.fit(X\_train,y\_train)

accuracy\_score(y\_test,pred\_test\_cv)

0.6736111111111112

**Bagging:-**

## What Is Ensemble Learning?

Ensemble learning is a widely used and preferred machine learning technique in which multiple individual models, often called base models, are combined to produce an effective optimal prediction model. The [Random Forest algorithm](https://www.simplilearn.com/tutorials/machine-learning-tutorial/random-forest-algorithm) is an example of ensemble learning.

**What Is Bagging in Machine Learning?**

Bagging, also known as Bootstrap aggregating, is an ensemble learning technique that helps to improve the performance and accuracy of machine learning algorithms. It is used to deal with bias-variance trade-offs and reduces the variance of a prediction model. Bagging avoids overfitting of data and is used for both regression and classification models, s

Diagram

Description automatically generated

## What Is Bootstrapping?

Bootstrapping is the method of randomly creating samples of data out of a population with replacement to estimate a population parameter.

Diagram

Description automatically generated

\

Steps to Perform Bagging

* Consider there are n observations and m features in the training set. You need to select a random sample from the training dataset without replacement
* A subset of m features is chosen randomly to create a model using sample observations
* The feature offering the best split out of the lot is used to split the nodes
* The tree is grown, so you have the best root nodes
* The above steps are repeated n times. It aggregates the output of individual decision trees to give the best prediction

Advantages of Bagging in Machine Learning

* Bagging minimizes the overfitting of data
* It improves the model’s accuracy
* It deals with higher dimensional data efficiently

**Random Forest(Bagging) on Cardiovascular datasets:-**

rf=RandomForestClassifier()

rf.fit(X\_train,y\_train)

accuracy\_score(y\_test,rf\_predicted\_y\_test)

0.8344907407407407

accuracy\_score(y\_train,rf\_predicted\_y\_train)=1.0

print(classification\_report(y\_test,rf\_predicted\_y\_test))

A picture containing text, receipt

Description automatically generated

**Confusion matrix:-**

**Application, PowerPoint

Description automatically generated**

**Boosting:-**

**Ad boost:-**

AdaBoost is a part of bigger set of algorithms that belong to methods called Ensemble learning. The entire idea behind Ensemble learning is to create multiple learning models instead of one model and predict the values.

AdaBoost does this process of predicting the value by creating multiple weak learners, associating a weight to each data point based on the predictions of that point in each learner and finally giving the output.

The way AdaBoost works is as follows:

1.) Primarily each data point is initialized with a weight ‘alpha’ that is equal to (1/number of data points).

2.) Then for each weak learner or model, we iterate to calculate the predicted value, then go onto compute the weight ‘w’(weighted error) and based on that re-adjust the ‘alpha’ term for next iteration until the very end.

3.) After all the iterations are done, we calculate the prediction based on the equation.



Equation of final prediction. L number of stumps.

**Adaboost on Cardiovascular datasets:-**

accuracy\_score(y\_test,ad\_test\_pred)

0.7146990740740741

accuracy\_score(y\_train,ad\_train\_pred)

0.7384615384615385

print(classification\_report(y\_test,ad\_test\_pred))

A picture containing table

Description automatically generated

# **#XGBoost**

XGBoost is an implementation of Gradient Boosted decision trees.

In this algorithm, decision trees are created in sequential form. Weights play an important role in XGBoost. Weights are assigned to all the independent variables which are then fed into the decision tree which predicts results. The weight of variables predicted wrong by the tree is increased and these variables are then fed to the second decision tree. These individual classifiers/predictors then ensemble to give a strong and more precise model. It can work on regression, classification, ranking, and user-defined prediction problems.

**XGBoost on Cardiovascular datasets:-**

**Table

Description automatically generated**

**K-Nearest Neighbors:-**

K-Nearest Neighbours is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection.  
It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as [GMM](https://en.wikipedia.org/wiki/Mixture_model), which assume a Gaussian distribution of the given data).

**KNN in Cardiovascular datasets:-**

**Table

Description automatically generated**

**RESULTS**: Test Accuracies obtained on various ML algorithms: Logistic Regression= 0.6707175925925926, KNN=0.7511574074074074,   
Decision Tree Classifier()=0.6689814814814815,

Random Forest Classifier(at no. of estimators:100)=0.8344907407407407,

AdaBoost=0.7384615384615385,XGBoost=0.8429280397022333

**Among all XGBoost Classifier has best accuracy rate as it is a boosting algorithm.**

# **Findings and Applications**

**1.**The elderly are more likely to suffer from various types of cardiovascular diseases.

**2.**Cholesterol level is a very important determinant in leading to cardiovascular diseases. The risks of getting cardiovascular diseases climb significantly when the cholesterol content in human body rises above the normal.

**3.**People with high alcohol consumption have a lower likelihood to have cardiovascular diseases.

**4.**Physical activities help people become less susceptible to cardiovascular diseases.

**5.**A female non-smoker is less likely to have cardiovascular diseases, while a male smoker is less likely to get cardiovascular diseases.

**6.**As long as people have systolic blood pressure ≥ 126, they are classified as those susceptible to cardiovascular diseases. Therefore, high systolic blood pressure plays an important role in predicting that one experiences greater risks of suffering from cardiovascular diseases.

**7.**Among the three splitting factors, two of them are age. This suggests that apart from systolic blood pressure, age is the second most significant factor in deciding whether a person will have a chance of getting cardiovascular disease.

## ****CONCLUSION****

**1.**As age increases the risk of getting diagnosed with heart disease also increases.

**2.**Cigarette consumption is also a major factor that causes CHDs.

**3**.Patients having Diabetes and cholesterol problems show a higher risk of CHDs.

**4.**Patients having high glucose levels are more prone to CHDs.

**5.**Patients with a history of “strokes” have a higher chance of developing CHDs.

**6.**Patients with high BMI(Body Mass Index) are at more risk of getting diagnosed with CHDs.

7.Finally we can say that, XGBoost Classifier has performed best among all the models with the accuracy of 76% & f1-score of 0.71. It is by far the second highest score we have achieved. So, It’s safe to say that XGBoost Classifier provides an optimal solution to our problem.