**CHAPTER-1**

**INTRODUCTION**

* In the current era, Heart Failure (HF) is one of the common diseases that can lead to dangerous situation. Every year almost 26 million of patients are affecting with this kind of disease.
* The main cause of heart stroke is due to blockage in arteries. It has many other names such as cardiovascular disease and arterial hypertension.
* Therefore, this project has taken a small step towards saving the lives of Heart failure patients and describes a way to improve the performance of diagnosing the patients on the bases of their medical history.
* By proper analysis on the kind of dataset can improve the process of diagnosing and can assist the heart surgeons as well.
* Previously different researches used number of techniques to improve the Heart failure diagnosis process such as Extreme Learning Machine, heart disease classification , and machine learning classifiers.
* Therefore, this research attempts to improve the performance of the classifiers by doing experiments using multiple machine-learning models to make better use of the dataset collected from different medical databases.
* In the medical field, machine learning can be used for diagnosis, detection and prediction of various diseases. The main goal of this project is to provide a tool for doctors to detect heart disease as early stage . This in turn will help to provide effective treatment to patients and avoid severe consequences.
* Machine learning (ML) plays a very important role to detect the hidden discrete patterns and thereby analyse the given data. After analysis of data ML techniques help in heart disease prediction and early diagnosis.
* This project presents performance analysis of various ML techniques such as XGBM,LGBM, Decision Tree, SVM, and Random Forest for predicting heart disease at an early stage.

**CHAPTER-2**

**LITERATURE SURVEY**

[1] Title of paper: "Intelligent Heart Disease Prediction System Using Data Mining Techniques"

Authors: Sellappan Palaniappan, Rafiah Awang

Year of publication: 2008

Work carried:

* Intelligent Heart Disease Prediction System (IHDPS) using data mining techniques, namely, Decision Trees, Naive Bayes and Neural Networks.
* IHDPS can discover and extract hidden knowledge associated with heart disease from a historical heart disease database.
* This IHDPS is based on 15 attributes. A total of 909 records were obtained from the Cleveland Heart Disease database.
* The records were equally divided into two datasets, i.e. training dataset (455 records) and testing dataset (454 records)..
* It has been observed during the analysis that Naive Bayes appears to be most effective as it has the highest percentage of correct predictions (86.53%) for patients with heart disease, followed by Neural Network (85.53%) and Decision Trees. Decision Trees, however, appears to be most effective in case of predicting patients with no heart disease, i.e. (89%) as compared to other two models. Data mining Techniques Accuracy Naive Bayes 86.53% Decision Trees 89% ANN 85.53% ​

Drawbacks​

* The IHDPS is based on the 15 attributes. This list may need to be expanded to provide a more  comprehensive diagnosis system.​
* The drawback is that it divide dataset 50% for both train and test sets.
* ​Another drawbacks is that it only uses categorical data. For some diagnosis, the use of continuous data may be necessary.​
* Another drawbacks is that it only uses three data mining techniques. Additional data mining techniques can be incorporated to provide better diagnosis.

[2] Title of paper: Enhanced Prediction of Heart Disease with Feature Subset Selection using Genetic Algorithm

Authors: M. Anbarasi, E. Anupriya, N.Ch.S.N.Iyengar

Year of publication: 2010

Work carried:

* The objective of this work was to reduce the number of attributes which were used for heart disease diagnosis. Earlier, 13    attributes were used for this prediction but this research work reduced the number of attributes to six only using Genetic Algorithm and Feature Subset Selection.
* Genetic Algorithm incorporates natural evolution methodology. The genetic search started with zero attributes, and an initial population with randomly generated rules.
* Based on the idea of survival of the fittest, new population was constructed to match with fittest rules in the current population, as well as offspring of these rules. Offspring were generated by applying genetic operators; cross over and mutation.
* The process of generation continued until it evolved a population P where every rule in P satisfied the fitness threshold. With initial population of 20 instances, generation continued till the twentieth generation with cross over probability of 0.6 and mutation probability of 0.033. The genetic search resulted in six attributes out of thirteen attributes.
* CFS Evaluator is also used in addition to the genetic algorithm. Observations are conducted using Weka 3.6.0 tool. Initially, data set of 909 records with 13 attributes was used. All attributes were made categorical and inconsistencies were resolved for simplicity. After reduction of 13 attributes to 6 attributes, various classifiers are used on the dataset corresponding to these 6 attributes for heart disease prediction.
* It can be perceived from the table that Decision Tree has outperformed with highest accuracy and least mean absolute error. DM Techniques Accuracy Model Construction Time Mean Absolute Error Naive Bayes 96.5% 0.02s 0.044 Decision Tree 99.2% 0.09s 0.00016 Classification via Clustering 88.3% 0.06s 0.117

Drawbacks.

* the observations exhibit that the Decision Tree data mining technique outperforms other two data mining techniques after incorporating feature subset selection with relatively high model construction time. Naïve Bayes performs consistently before and after reduction of attributes with the same model construction time. Classification via clustering performs poor compared to other two methods. Inconsistencies and missing values were resolved before model construction but in real time, that is not the case. Also, the intensity of the disease based on the results was unpredictable. We intend to extend our work applying fuzzy learning models to evaluate the intensity of cardiac disease.

[3] Title of paper: Diagnosis of Heart Disease Using Datamining Algorithm

Authours:Asha Rajkumar, G. Sophia Reena

Year of publications: september 2010

Work carried:

* This research work has presented the data classification based on various supervised machine learning algorithms, namely, Naive Bayes, Decision List and KNN. TANAGRA tool is used to classify the data and the data is evaluated using 10- fold cross validation. TANAGRA  is a data mining tool for academic and research purposes.
* It proposes several data mining methods from exploratory data analysis, statistical learning, machine learning and databases area. It provides an easy-to-use interface by allowing the users to analyze either real or synthetic data. This tool also proposed an architecture to the users allowing them to easily add their own data mining methods, to compare their performances. It is a wide set of data sources, direct access to data warehouses and databases, data cleansing, interactive utilization.
* Experiments are conducted by using the training data set of 3000 instances with 14 different attributes. Depending upon the attributes, the dataset is classified into two parts, i.e. 70% of the data is used for training and rest 30% is used for testing. Performance of each algorithm is determined and comparison is made based on the accuracy and evaluation time of calculation for each algorithm . It has been observed that Naive Bayes algorithm performed better in comparison to Business Understanding Evaluation Data Preparation Deployment Modeling DATA Data Understanding other two algorithms.
* the performance study of various algorithms. Algorithm Used Accuracy Time Taken Naive Bayes 52.33% 609ms Decision List 52% 719ms KNN 45.67% 1000ms .

Drawbacks:

* Health care related data are voluminous in nature and they arrive from diverse sources all of them not entirely appropriate in structure or quality
* Attributes are fully classified by this algorithm and naïve bayes gives 52.33% of accurate results,it is found to be better using Naive Bayes algorithm compare to other algorithms. It does not give much accuracy

[4] Improved Study of Heart Disease Prediction System using Data Mining Classification Techniques ​

Authors: Chaitrali S. Dangare, Sulabha S. Apte​

Year of publication: 2012​

* An artificial neural network (ANN), often just called a "neural network" (NN), is a mathematical model or computational model based on biological neural network.
* The data mining tool Weka 3.6.6 is used for experiment. Initially, missing values were identified in the dataset and they were replaced with appropriate values using ReplaceMissingValues filter from 3.6.6 . Further, various data mining techniques have been analyzed on heart disease database. Confusion matrix is obtained for each classifier.
* depicts the outcomes of this research work and it shows that neural networks has outplayed over other data mining techniques. Classification Techniques Accuracy Naive Bayes 90.74% Decision Trees 99.62% Neural Networks 100%​

Drawbacks:​

* This system can be further expanded. It can use more number of input attributes . Other data mining techniques can also be used for predication e.g. Clustering, Time series, Association rules. The text mining can be used to mine huge amount of unstructured data available in healthcare industry database.

[5] Title Of Paper : Heart disease prediction using machine learning techniques : a survey.

Authors : V. V. Ramalingam, A. Dandapath, and M. Karthik Raja.

Year of Publication : Int. J. Eng. Technol., vol. 7, no. 2.8, pp. 684–687, 2018

Work carried :

* The data collected at Medical Organisations on various health issues can be exploited using various machine learning techniques to gain useful insights.. Thus, the algorithms and techniques such as NB, SVM, KNN, DT, RF are used that have become very useful, in recent times, to predict the presence or absence of heart related diseases accurately. In this survey, Dimensionality Reduction is used which involves selecting a mathematical representation such that one can relate the majority of, but not all, the variance athe given data, thereby including only most significant information.
* Dimensionality Reduction is a very important step considered while building any model. Dimensionality Reduction is generally achieved by two methods -Feature Extraction and Feature Selection.
* Feature Extraction In this, a new set of features is derived from the original feature set. Feature extraction involves a transformation of the features. Principal Component Analysis (PCA)is used for feature extraction. Principal Component Analysis is a popularly used linear transformation algorithm.
* Feature Selection In this, a subset of original feature set is selected. key features are selected by CFS(Correlation based Feature Selection) Subset Evaluation combined with Best First Search method to reduce dimensionality

Accuracy Obtained :

* Naive Bayes has achieved an accuracy of 84.1584%
* SVM classifies the pixel variation with an accuracy of 92.1%
* KNN gives an accuracy of 83.16%
* when Decision Tree is used witha boosting technique it performs better with an accuracy of 82.17%..
* Random forest is used to predict coronary heart disease and it obtains an accuracy of 97.7% than all the other methods.

Drawbacks :

* Feature extraction involves a transformation of the features which is often not reversible as few, or maybe many, useful information is lost in the process.
* the data collected is very massive and, many a times, this data can be very noisy. These datasets, which are too overwhelming for human minds to comprehend.
* Still there is a lot scope of research to be done on how to handle high dimensional data and overfitting. A lot of research can also be done on the correct ensemble of algorithms to use for a particular type of data

[6] Title Of Paper : A Hybrid Intelligent System Framework for the Prediction of Heart Disease Using Machine Learning Algorithms

Authors : Amin Ul Haq, Jian Ping Li,Muhammad Hammad Memon,Shah Nazir, and Ruinan sun

Year of publication : 2 dec 2018

Work carried:

* In the present study, various machines learning predictive models such as logistic regression, knearest neighbor, ANN, SVM, Decision Tree, Naive Bayes, and random forest have been used for classification of people with heart disease and healthy people. study used feature selection methods for selection of features. Therefore, the classification performance of the approach depends on selected features.
* The novelty of this research work is developing a diagnosis system for HD. The system used three FS algorithms, seven classifiers, one cross-validation method, and performance evaluation metrics for HD diagnosis. The system was tested on Cleveland heart disease dataset to classify HD and healthy subjects.
* This section of paper involved the discussion on the classification models and outcomes from different perspectives. First, they checked the performance of different machine learning algorithms such as logistic regression, k-nearest Neighbor, artificial neural network, support vector machine, Naive Bayes, and decision tree on Cleveland heart disease dataset on full features.
* In the second, They used feature selection algorithm Relief, mRMR, and LASSO for important features selection.
* In third classifiers, performances were checked on selected features. Also, the k-fold cross- validation method was used. In order to check the performance of classifiers performance evaluation metrics were applied. All features were normalized and standardized before applying to classifiers

Accuracy Obtained :

* Decision Tree : 74%, Logistic Regression : 84%, Random Forest : 83%, Naïve Bayes : 84%, SVM : 88%, ANN: 73%, KNN : 81%.

Drawbackas :

* More experiments are to be performed to increase the performance of these predictive classifiers for heart disease diagnosis by using others feature selection algorithms and optimization techniques.
* Designing a decision support system through machine-learning-based method will be more suitable for diagnosis of heart disease. Additionally, some irrelevant features reduced the performance of the diagnosis system and increased the computation time.

[7] Title Of Paper : Implementation of machine learning model to predict heart failure disease.

Author : Fahd Saleh Alotaibi

Year of publication : (IJACSA) International Journal of Advanced Computer Science and Applications, 10 (6) (2019)

MACHINE LEARNING CLASSIFIERS FOR HEART DISEASE PREDICTION:

* The identification of heart disease in a patient is complex and requires various details, laboratory tests, and equipment . This research is not for replacing the traditional approach use for diagnosing and predicting the chances of heart failure, rather the study attempted to support this process using advanced technologies such as Machine Learning (ML).
* This system used machine-learning methods for predicting the heart disease. The system was proposed to provide the assistance in affordable way, where the system have the capacity to integrate with existing system.
* ML approaches proved to be an effective in predicting the heart disease using historical data is further proved in a research conducted using Naïve Bayes, Decision Tree, support vector model and other models. The results indicated that the support vector machine provided the optimal results between other implemented approaches.
* This research used five machine learning models, using predictive approach to forecast the chances of heart failure in a patient admitted in the hospitals.

data overview :

* The dataset used in this research is collected from Kaggle platform, the dataset is known as Heart Disease Dataset.. Altogether, the data was the combination of four different database, but only Cleveland data used in this experiment.
* It is an open dataset, having number of attributes, but for this experiment only fourteen attributes selected as describedand suggested by different scholars that selected 14 attributes are most useful to predict the heart disease in a patient.
* In addition, the database file contains the record of 303 patients.

Accuracy obtain:

* Decision Tree : 82.22%, Logistic Regression : 82.56%, Random Forest : 84.17%, Naïve Bayes : 84.24%, SVM : 84.85%

Drawbacks:

* The main limitation in this work is the small size of the dataset.
* The dataset has limited number of patient’s records; therefore, the dataset was augmented using appropriate techniques.
* This research to emphasize onto overcome the vulnerable situation and proposed a computerized system, so that heart consultants cannot miss any information due to improper reading and understanding of the data.

[8] Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone

Authors : Davide Chicco, Giuseppe Jurman

Year of publication : BMC Med. Inf. Decis. Making, 20 (2020), p. 16

Work carried:

* In this paper, They analysed a dataset of 299 patients with heart failure collected in  2015.
* They  applied several machine learning classifiers to both predict the patients survival, and rank the features corresponding to the most important risk factors.
* They also perform an alternative feature ranking analysis by employing traditional biostatistics tests, and compare these results with those provided by the machine learning algorithms.
* Since both feature ranking approaches clearly identify serum creatinine     and ejection fraction as the two most relevant features, they then build the machine learning survival prediction models on these two factors alone.

Results :

* Their prediction results showed that Random Forests outperformed all the other methods, by obtaining the top MCC (+0.384), the top accuracy (0.740), and the top ROC AUC (0.800).
* The Decision Trees obtained the top results on the true positives (sensitivity = 0.532) and on the F1 score (0.554), and was the only classifier able to predict correctly the majority of deceased patients
* The linear Support Vector Machines achieved an almost perfect prediction score on the negative elements (specificity = 0.961).
* The Artificial Neural Network perceptron, instead, obtained the top value on the Precision-Recall AUC (0.750).

Drawbacks :

* As a drawback of this study, they have to report the small size of the dataset (299 patients): a larger dataset would have permitted us to obtain more reliable results.
* Additional information about the physical features of the patients (height, weight, body mass index, etc.) and their occupational history would have been useful to detect additional risk factors for cardiovascular health diseases.
* Also, if an additional external dataset with the same features from a    different geographical region had been available, They  would have used it as a validation cohort to verify our findings.z.

[9] Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques

Authors : Galla Siva Sai Bindhika, Munaga Meghana, ManchuriSathvika Reddy, Rajalakshmi

Year of publication : International Research Journal of Engineering and

* Technology (IRJET), 07 (2020)
* It is difficult to identify heart disease because of several contributory risk factors such as diabetes, high blood pressure, high cholesterol, abnormal pulse rate and many other factors.
* Various techniques in data mining and neural networks have been employed to find out the severity of heart disease among humans.
* The severity of the disease is classified based on various methods like K-Nearest Neighbour Algorithm (KNN), Decision Trees (DT), Genetic algorithm (GA), and Naive Bayes (NB).
* The nature of heart disease is complex and hence, the disease must be handled carefully. Not doing so may affect the heart or cause premature death.
* The perspective of medical science and data mining are used for discovering various sorts of metabolic syndromes.
* Data mining with classification plays a significant role in the prediction of heart disease and data investigation.
* Accuracy :
* They produce an enhanced performance level with an accuracy level of 88.7% through the prediction model for heart disease with the hybrid random forest with a linear model (HRFLM).
* Drawbacks :
* The drawback of this paper is that they have used only one algorithm. They  would have got better results if they had used more than one algorithms.

[10] Heart Disease Prediction using Machine Learning

Authors : Apurb Rajdhan, Milan Sai, Avi Agarwal, Dundigalla Ravi, Dr.Poonam Ghuli

Year of publications : International Journal of Engineering Research &

* Technology (IJERT), 9 (2020)
* The work proposed in this paper focus mainly on various data mining practices that are employed in heart disease prediction.
* Human heart is the principal part of the human body. Basically, it regulates blood flow throughout our body. Any irregularity to heart can cause distress in other parts of body.
* Any sort of disturbance to normal functioning of the heart can be classified as a Heart disease.
* In today’s contemporary world, heart disease is one of the primary reasons for occurrence of most deaths.
* According to the World Health Organization more than 10 million die due to Heart diseases every single year around the world.
* A healthy lifestyle and earliest detection are only ways to prevent the heart related diseases.
* They have used many algorithms in this paper such as , Decision Tree, Logistic Regression, Random forest, Naïve Bayes.
* Accuracy : Decision Tree : 81.97% , Logistic Regression : 85.25% ,Random Forest : 90.16% , Naive Bayes : 85.25% .

Drawbacks :

* They have not used KNN algorithm. They would have got better results if they had used KNN algorithm in addition to the other algorithms.
* The accuracy of Decision Tree is less.

**CHAPTER- 3**

**PROPOSED SYSTEM**

* The project aims at analyzing the machine learning algorithms based on percentage of various performance metrics (such as accuracy ,precision and recall).
* The better machine learning methodologies are implemented to achieve Accuracy
* Project proposes to explore more dataset .compare to the earlier researches. In order to get the better analysis of the performance metrics
* Analyzing using specific variables in the dataset by using python programming as well as different machine learning algorithms which includes, SVM, decision tree, random forest, XGBM, LGBM.

**CHAPTER-4**

**OBJECTIVES**

* To collect the data of heart patients from the portal.
* To pre-process the data to remove artifacts.
* To perform feature selection from the dataset.
* To classify the data of heart failure using Decision Tree algorithm.
* To Perform result analysis using different datasets.
* To make comparison using different algorithms like Random Forest, Decision Tree, XGBM , LGBM

**CHAPTER-5**

**METHODOLOGY**

This project aims to make the objective assessment of heart failure. This project collects the data from the portal and has 300 datasets which helps in predicting heart failure using ML algorithm Decision Tree. The below figure gives the methodology used to achieve the goal.

Diagram

Description automatically generated

Fig 5.1 Process block diagram

Data Description

A dataset is a collection of data. In case of tabular data a dataset corresponds to one or more database tables where every column of a table represents a particular variable and each row corresponds to given records of the dataset.

Data Pre-Processing

It is a process of modifying the required columns in a dataset (if the dataset contains raw data) for analyzing the data. For example, the dataset used in this paper had few string formats which represent labels, genders, etc. Data Pre-processing was applied to retrieve numerical data, string data, categorical data appropriately. In this project, the data preprocessing is carried out to remove the unnecessary values in the columns such as, missing values, false values and null values. Our dataset may be noisy sometimes, so to remove these unnecessary and missing columns, this step is carried out.

Feature Selection

Feature Selection is a process of selecting the necessary variables to increase the accuracy. The selection of variables during the feature selection can be manual or automatic. In this project, the necessary features are selected to raise the percentage of accuracy. This step involves selecting the relevant features and discarding the irrelevant features. In this project, the features regarding the heart failure are selected and access features are removed .

Train - Test Split

Train – Test Split is a technique to divide the given dataset into subsets and train further. This technique can be used to rate the performance of a machine learning algorithms. In this project, the dataset is split to fit into any of the models. The data is splitted into train and test subsets by taking the size of test. Train-split is 0.85 of total data and test-split is of 0.25.

Machine Learning Models

These models take request in the form of input data, analyze, process and then serve the response. The models can be trained over a set of data and the algorithms reason out and learn from this data. In this project, mathematical models are given by; SVM, Decision Tree, XGBM, LGBM and Random Forest are used for implementation.

**CHAPTER-6**

**ARCHITECTURE AND IMPLEMENTATION**

RAW DATA COLLECTED

DATA IS PROCESSED

CLEAN DATASET

EXPLORATOR--Y DATA ANALYSIS

MODELS AND ALGORITHM

CONSOLID-ATED REPORT

Fig 6.1 Architecture design

**6.1 Required Libraries**

**6.1.1 Pandas:**

Pandas is an open-source library that is made mainly for working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library. Pandas is fast and it has high performance & productivity for users**.**

* Fast and efficient for manipulating and analyzing data.
* Data from different file objects can be loaded.
* Easy handling of missing data (represented as NaN) in floating point as well as non-floating point data
* Size mutability: columns can be inserted and deleted from DataFrame and higher dimensional objects
* Data set merging and joining
* Data set merging and joining
* Provides time-series functionality
* Powerful group by functionality for performing split-apply-combine operations on data sets

**6.1.2 matplot lib.pyplot:**

Matplotlib is a plotting library for creating static, animated, and interactive visualizations in Python. Matplotlib can be used in Python scripts, the Python and IPython shell, web application servers, and various graphical user interface toolkits like Tkinter, awxPython, etc.

Pyplot is a Matplotlib module which provides a MATLAB-like interface. Matplotlib is designed to be as usable as MATLAB, with the ability to use Python and the advantage of being free and open-source. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc. The various plots we can utilize using Pyplot are Line Plot, Histogram, Scatter, 3D Plot, Image, Contour, and Polar.

**6.1.3 seaborn**

Seaborn is an amazing visualization library for statistical graphics plotting in Python. It provides beautiful default styles and color palettes to make statistical plots more attractive. It is built on the top of matplotlib library and also closely integrated to the data structures from pandas.

Seaborn aims to make visualization the central part of exploring and understanding data. It provides dataset-oriented APIs, so that we can switch between different visual representations for same variables for better understanding of dataset.

**6.2 Data collection and preprocessing**

**6.2.1 Data collection**

**Data collection** is the process of acquiring, collecting, extracting, and storing the voluminous amount of data which may be in the structured or unstructured form like text, video, audio, XML files, records, or other image files used in later stages of data analysis.

“Data collection” is the initial step before starting to analyze the patterns or useful information in data. The data which is to be analyzed must be collected from different valid sources.

We have collected data from kaggle.com “heart\_failure\_clinical\_record\_dataset.csv” kaggle, a subsidiary of Google LLC, is an online community of data scientists and machine learning practitioners. Kaggle allows user to find and publish datasets

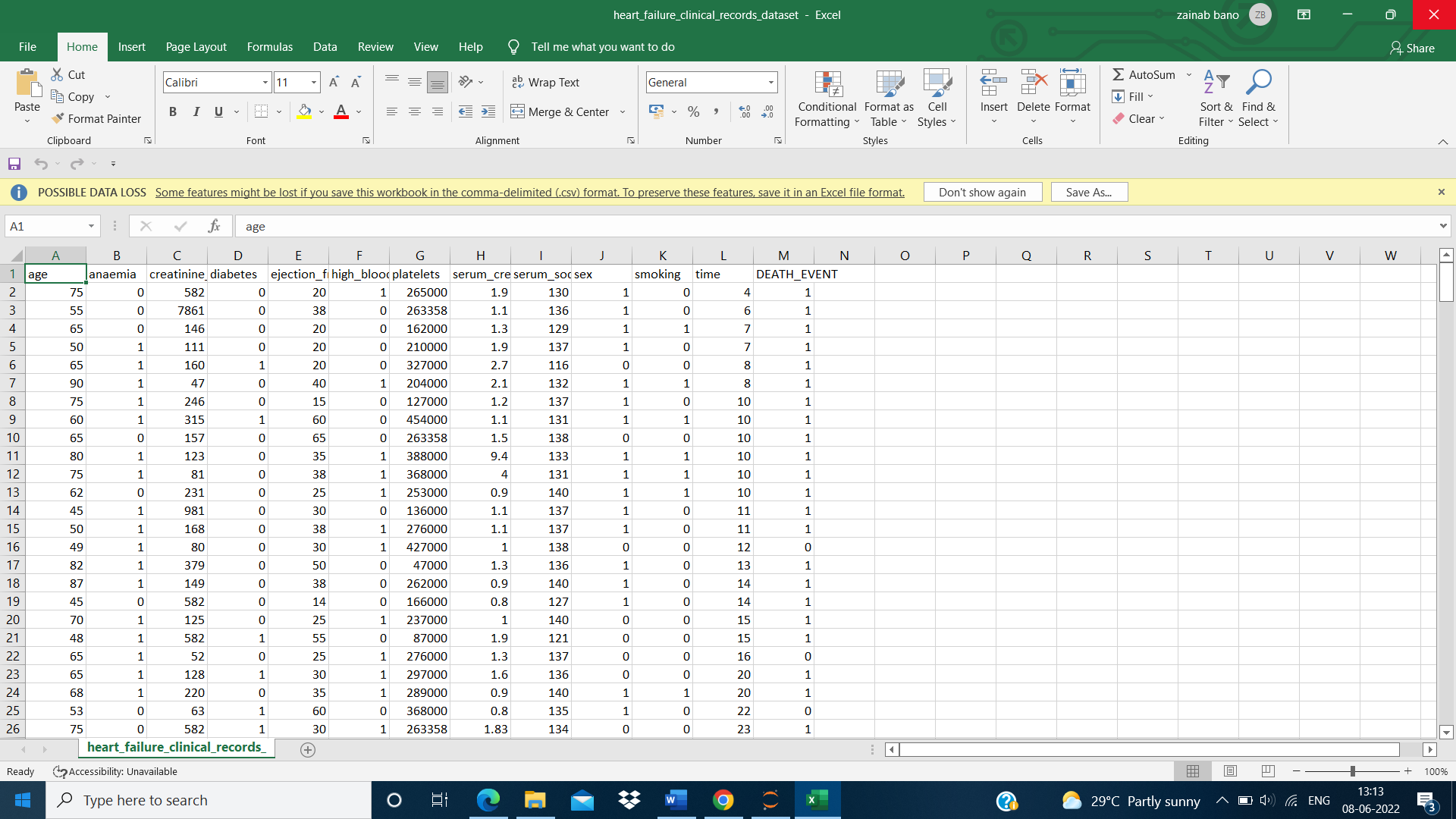


Fig 6.2 Data collection

**Preprocessing** simply refers to perform series of operations to transform or change data. It is transformation applied to our data before feeding it to algorithm.

**Data processing** refers to perform operations on data to retrieve, transform, or change data, especially by computer. It is technique that is used to convert raw data into clean data set.

**6.2.2 Data Description**

A **data set** is a collection of data, often presented in a table. A data set is a collection of related, discrete items of related data that may be accessed individually or in combination or managed as a whole entity.

A data set is organized into some type of [data structure](https://searchsqlserver.techtarget.com/definition/data-structure). In a [database](https://searchsqlserver.techtarget.com/definition/database), our data set contain a collection of Heart\_failure\_clinical\_record data (Age, Aneamia, High blood preasure, creatinine phosphokinase, diabetes, ejection fraction, sex, platelets, serum creatinine, serum sodium, smoking, Time, DEATH EVENT). The database itself can be considered a data set.

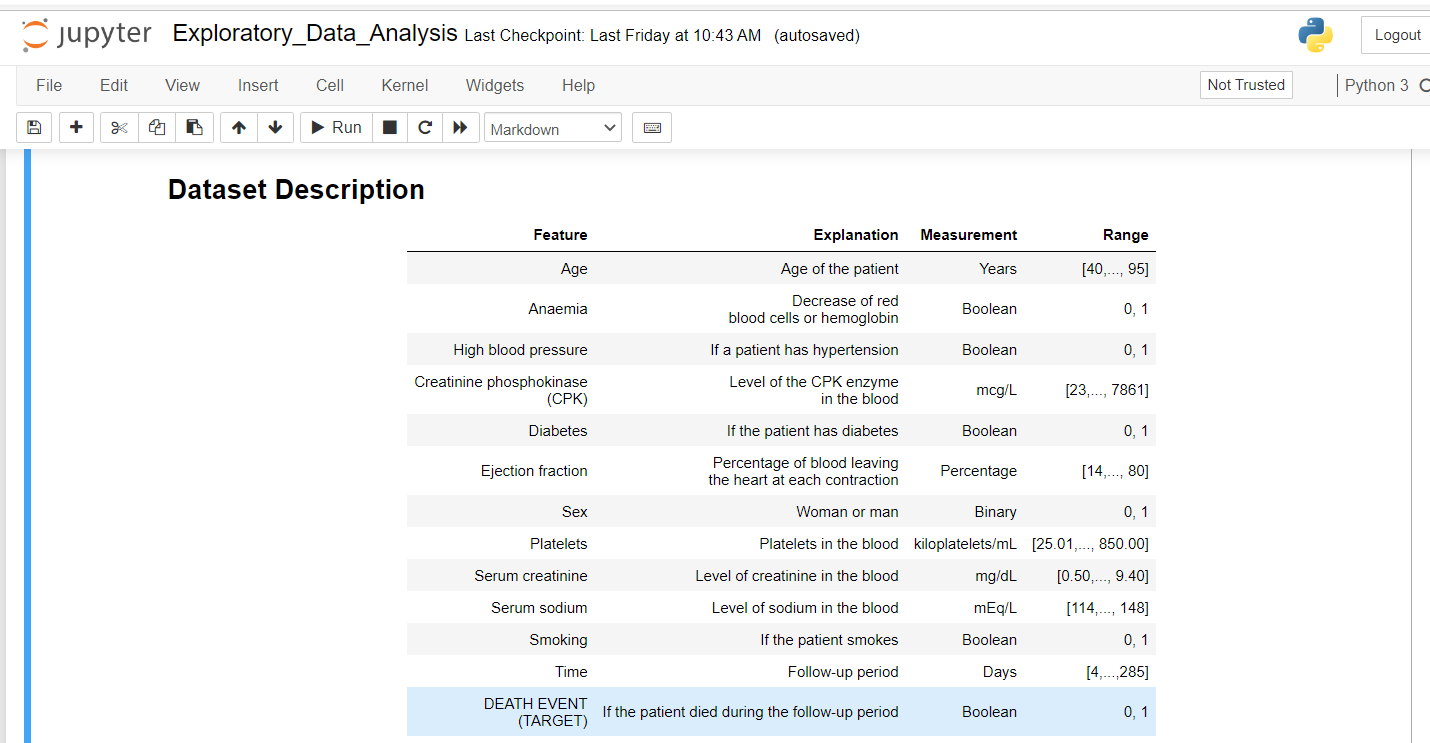


Fig 6.3 Data Description

**NOTE: mcg/L: micrograms per liter. mL: microliter. mEq/L: milliequivalents per litre**

**6.2.3 Import Data**

How to read from a file in Python

Python provides inbuilt functions for creating, writing and reading files. There are two types of files that can be handled in python, normal text files and binary files (written in binary language, 0s and 1s).

Text files: In this type of file, Each line of text is terminated with a special character called EOL (End of Line), which is the new line character (‘\n’) in python by default.

Binary files: In this type of file, there is no terminator for a line and the data is stored after converting it into machine-understandable binary language.

Access mode

Access modes govern the type of operations possible in the opened file. It refers to how the file will be used once it’s opened. These modes also define the location of the File Handle in the file. File handle is like a cursor, which defines from where the data has to be read or written in the file. Different access modes for reading a file are –

Read Only (‘r’) : Open text file for reading. The handle is positioned at the beginning of the file. If the file does not exists, raises I/O error. This is also the default mode in which file is opened.

Read and Write (‘r+’) : Open the file for reading and writing. The handle is positioned at the beginning of the file. Raises I/O error if the file does not exists. Append and Read (‘a+’) : Open the file for reading and writing. The file is created if it does not exist. The handle is positioned at the end of the file. The data being written will be inserted at the end, after the existing data.

Checking total data count:

We have 299 rows and 13 columns of data

Checking the total missing values

Total NULL Values in each columns

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

age 0

anaemia 0

creatinine\_phosphokinase 0

diabetes 0

ejection\_fraction 0

high\_blood\_pressure 0

platelets 0

serum\_creatinine 0

serum\_sodium 0

sex 0

smoking 0

time 0

DEATH\_EVENT 0

dtype: int64

***This looks to be a clean dataset without any missing values...***

**6.2.4 What is Exploratory Data Analysis (EDA) ?**

EDA is a phenomenon under data analysis used for gaining a better understanding of data aspects like:

– main features of data

– variables and relationships that hold between them

– identifying which variables are important for our problem

We shall look at various exploratory data analysis methods like:

Descriptive Statistics, which is a way of giving a brief overview of the dataset we are dealing with, including some measures and features of the sample

Grouping data [Basic grouping with group by]

ANOVA, Analysis Of Variance, which is a computational method to divide variations in an observations set into different components.

**6.2.4.1** **Univariate Analysis of Categorical variables**

1. anaemia
2. high\_blood\_pressure
3. diabetes
4. sex
5. smoking
6. DEATH\_EVENT

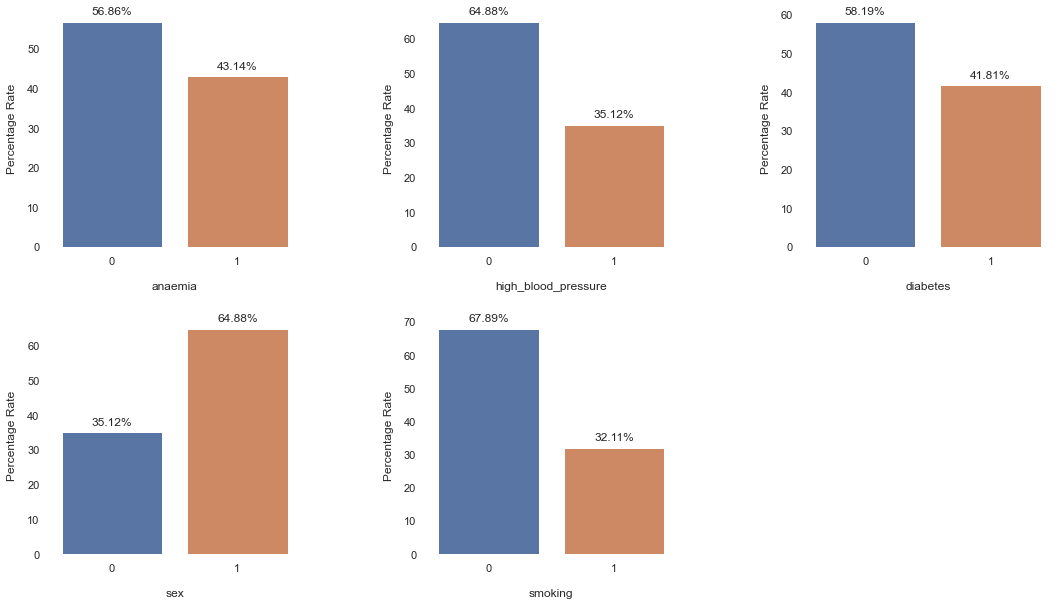


Fig 6.4 univariate analysis of categorical variables

**Data Insight 1:**

1. ~57% of the population under study have aneamic symptoms while ~43% are non-aneamic.

2. ~65% of the population have hypertension or high blood pressure while ~35% have normal blood pressure.

3. ~58% of the population are diabetic while ~42% are non-diabetic.

4. ~65% of the population are male while ~35% are female.

5. ~68% of the population have smoking habits while ~32% are non-smokers.

**6.2.4.2** **Distribution Of Age**

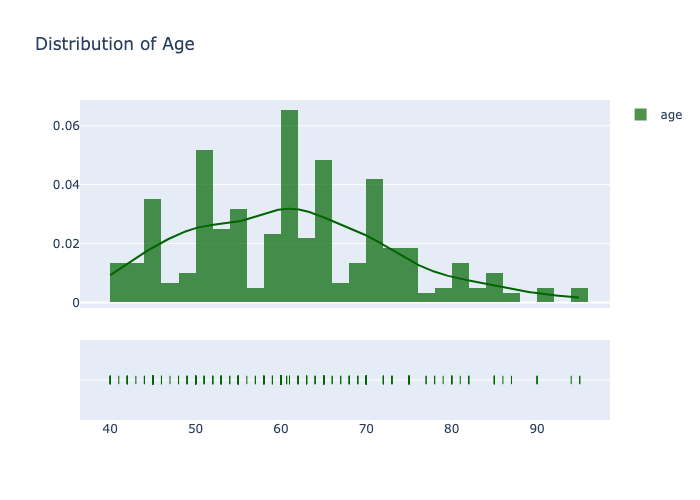


Fig 6.5 Distribution of Age

**Data Insight 2:**

1. The age group of people ranges from 40 - 95 with visible spike in the population density at certain age intervals around [44-46], [50-52], [60-62] (highest density), [64-66], [70-72].

**6.2.4.3 Distribution of Age w.r.t Gender**

****

Fig 6.6 Distribution of age w.r.t gender

**Data Insight 3:**

1. The minimum age is 40 for both Males and Female.

2. The maximum age is 95 for Male population and for female population the maximum age is 90 for the population being studied**.**

**6.2.4.4 Effect of Age on Survival Rate**

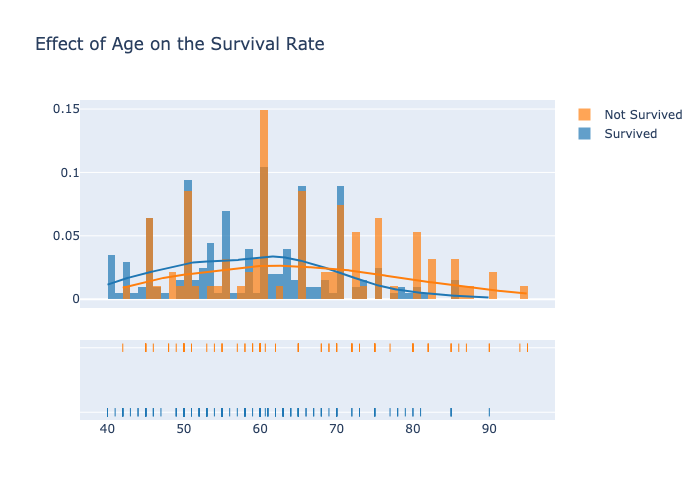
****

Fig 6.7 Effect of Age on survival rate

**Data Insight 4:**

1. The survival rate is more within the age group 50 to 70.

2. The chances of not surviving happens around all the age group in the population being studied. This is highest around the 60's age group. Above the age of 80, the chances of survival decreases drastically.

3. The above numbers could be mostly due the lifestyle and the eating habits of the population. A better active lifestyle would increase the chances of surviving the event of a heart failure.

**6.2.4.5 Survival Rate w.r.t Gender**

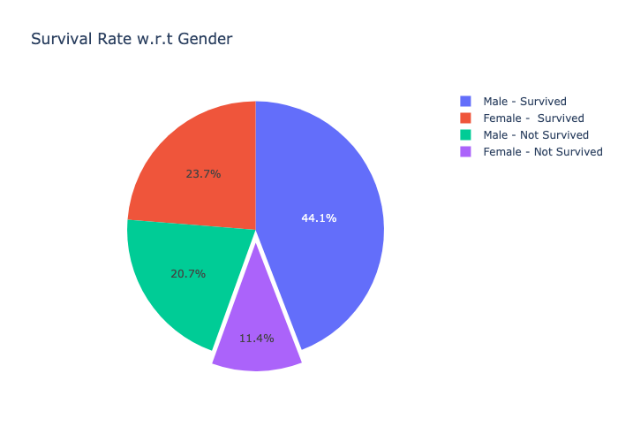
****

Fig 6.8 Survival Rate w.r.t gender

**Data Insight 5:**

1. From the above pie chart it is evident that 44.1% (132) of the male population have survived while 20.7% (62) succumbed to the heart failure.

2. Coming to the female population, 23.7% (71) of the population survived the heart failure while 11.4% (34) succumbed to the heart failure**.**

3. Male patients are more at risk of heart failure than a female.

**6.2.4.6 Effect of High Blood Pressure (Hypertension) on Survival Rate**

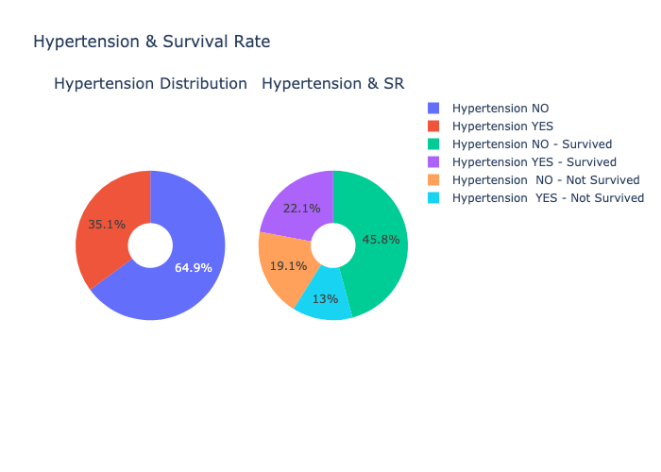
****

Fig 6.9 Effect of high blood pressure

**Data Insight 6:**

1. Out of the total population, around 35% of then suffer of high blood pressure or hypertension.

2. Amongst these 35% population, 22% survived the event of a heart failure while 13% succumbed to the condition.

3. Amongst the 65% population that do not have hypertension, 45.8% survived the event of a heart failure while 19% succumbed to the condition.

**6.2.4.7 Effect of Diabetes on Survival Rate**

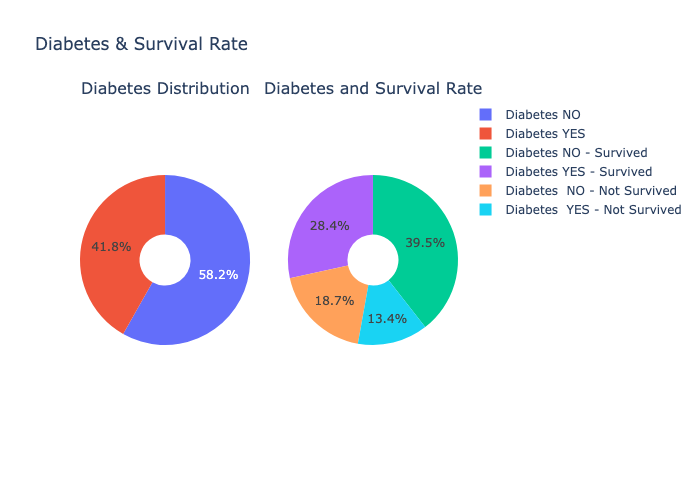
****

Fig 6.10 Effect of Diabetes on Survival Rate

**Data Insight 7:**

1. Out of the total population, ~42% of the people have deibetes while ~58% do not have diabetes.

2. Of the people who have deibetes, 28.4% of the population survived the event of a heart failure while 13.4% people succumbed to the condition.

3. Of the people who do not have deibetes, 39.5% of the population survived the event of a heart failure while 18.7% people succumbed to the condition**.**

**6.2.4.8 Effect of Smoking on Survival Rate**

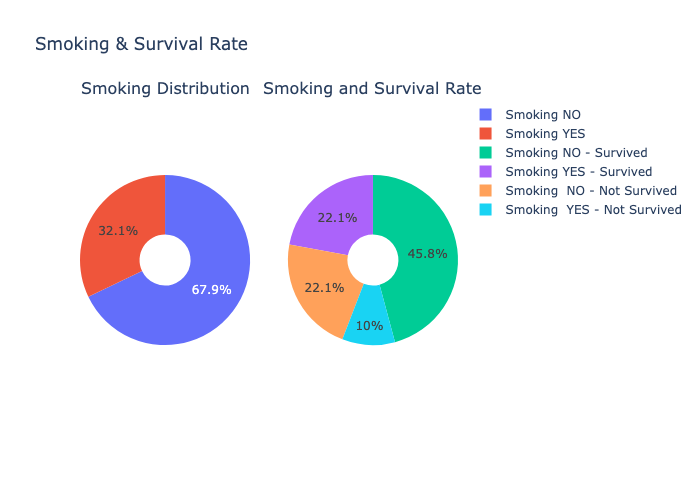
****

Fig 6.11 Effect of Smoking on Survival Rate

**Data Insight 8:**

1. Out of the total population, ~32% of the people have smoking habits while ~68% do not have any smoking habits.

2. Of the people who smoke, 22.1% of the population survived the event of a heart failure while 10% people succumbed to the condition.

3. Of the people who do not have smoking habits, 45.8% of the population survived the event of a heart failure while 22.1% people succumbed to the condition.

**6.2.4.9 Effect of Anaemia on Survival Rate**

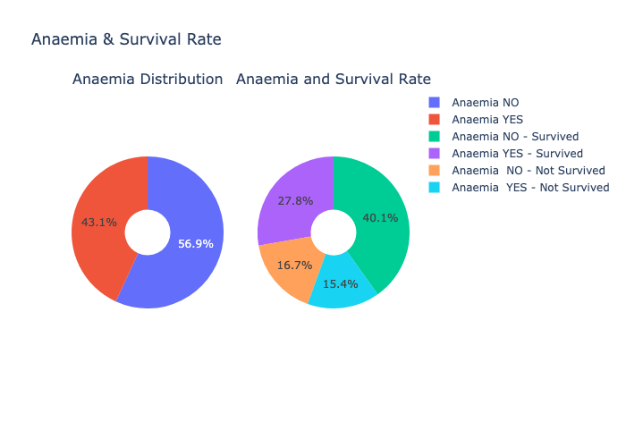
****

Fig 6.12 Effect of Anaemia on Survival rate

**Data Insight 9:**

1. Out of the total population, ~43.1% of the people have Anaemia observed while ~56.9% do not have any Anaemia synmtoms.

2. Of the people who have anaemia, 27.8% of the population survived the event of a heart failure while 15.4% people succumbed to the condition.

3. Of the people who do not have anaemia, 40.1% of the population survived the event of a heart failure while 16.7% people succumbed to the condition.

**6.2.4.10. Creatinine Phosphokinase w.r.t Survival Rate**

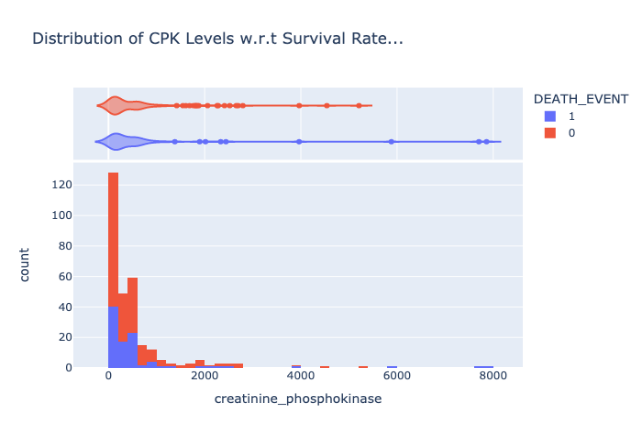
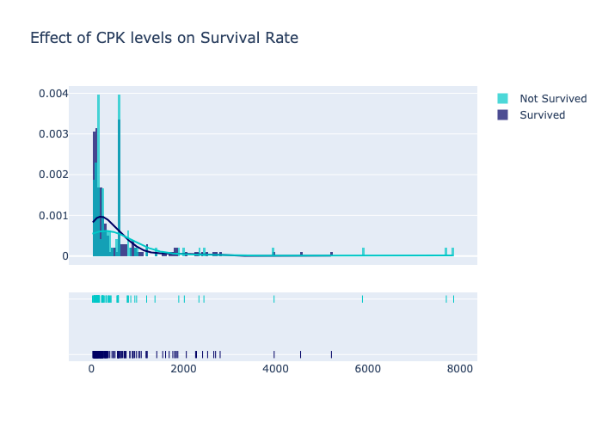
**a) CPK Enzyme Levels – Histogram b) CPK Enzyme Levels - Dist Plot  **

Fig 6.13 CPK Enzyme level-Histogram Fig 6.14 CPK Enzyme Levels-Dist plot

**Data Insight 10:**

1. The CPK levels for patients who did not survive the heart failure are on the higher side. Some patients show an abnormally high levels of the CPK enzyme.

2. There are some clear outliers (on the higher side) as it is observed in the violin histogram above. However, there very high values of CPK levels are present in both the events of SURVIVAL and DEATH.

**6.2.4.11. Ejection Fraction w.r.t Survival Rate**

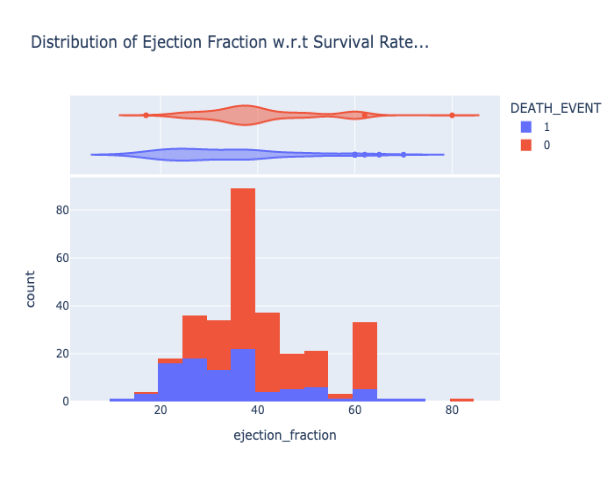
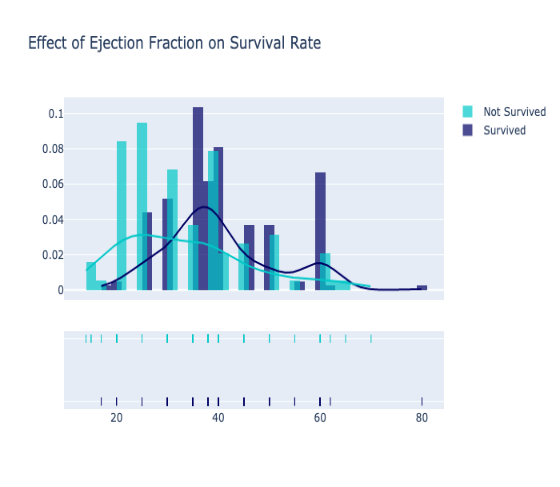
**a) Ejection Fraction – Histogram b) Ejection Fraction - Dist Pl **

Fig 6.15 Ejection Fraction – Histogram Fig 6.16 Ejection Fraction – Dist PI

**Data Insight 11:**

1. Majority of the person who succumbed to the heart failure condition showed lower than normal ejection fraction values. Only 25-45 (%) of the blood was being pumped out of the heart.

**6.2.4.12 Platelets w.r.t Survival Rate**

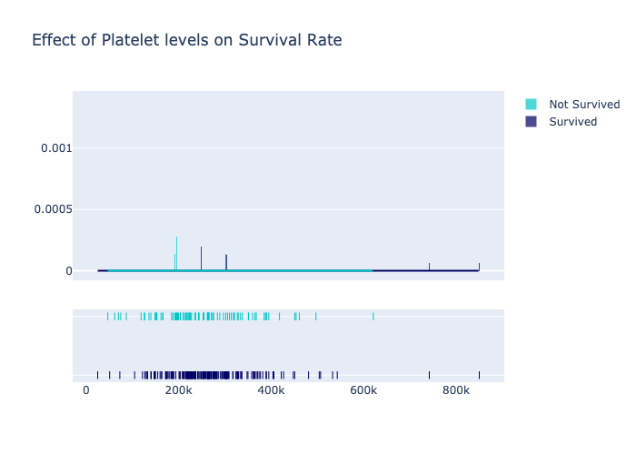
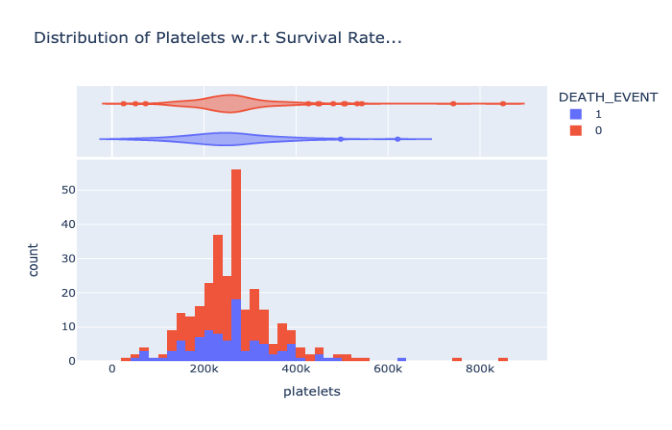
**a)** **Platelets and Death Event – Histogram b)** **Platelets and Death Event -DistPlot**

Fig 6.17 Platelets and Death Event – Histogram Fig 6.18 Platelets and Death Event -DistPlot

**c)** **Platelets and Death Event - Notched Box Plot**

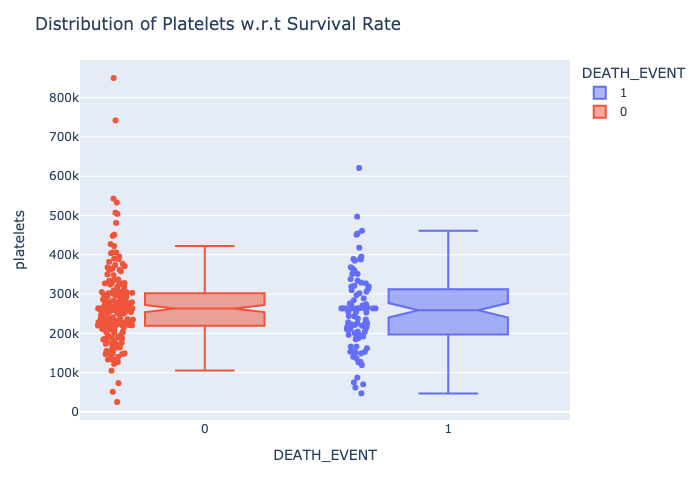
****

Fig 6.19 Platelets and Death Event - Notched Box Plot

**Data Insight 12:**

1. Majority of the person who succumbed to the heart failure condition have platelets count within the mormal range. There are a few cases of death where the platelet count hovers arund the lower boundary limit.

2. Some cases report abnormally high count of platelets in both the scenarios - SURVIVAL and DEATH\_EVENT.

3. Thus as mentioned in [2] above, Platelet abnormalities have been well described in heart failure (which can be seen in the plots above) but the significance of platelets in contributing to the thromboembolic complications of heart failure remains uncertain.

4. There are some clear outliers as it is observed in the box plot above. These outlying values are mostly observed when the patient survived the event of a heart failure.

**6.2.4.13 Serum Creatinine w.r.t Survival Rate**

**a)** **Elevated Serum Levels in Pie Charts b)** **Serum CreatinineHistogram**

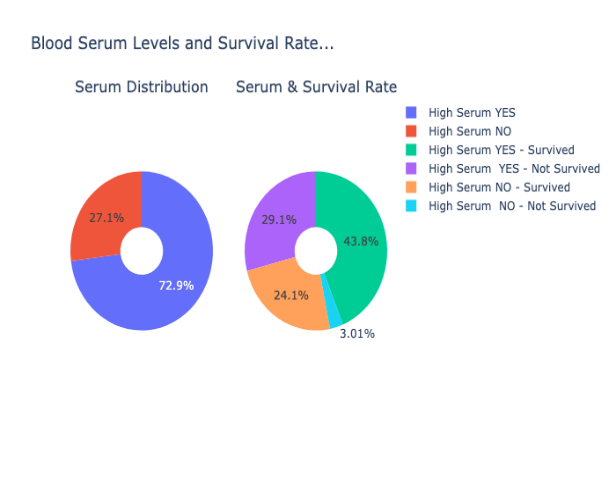
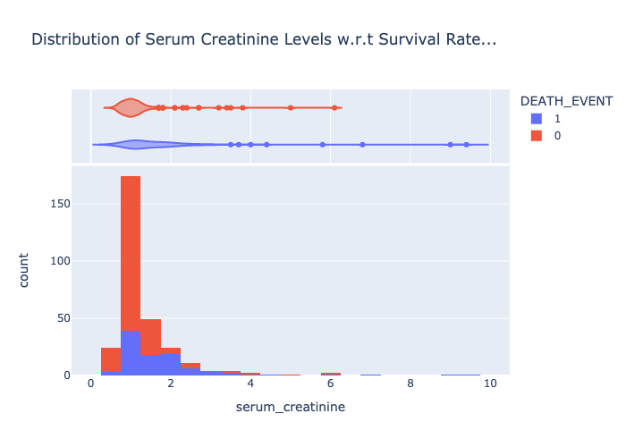
** **

Fig 6.20 Elevated Serum Levels in Pie Charts Fig 6.21 Serum Creatinine Histogram

**c) Serum Creatinine Dist Plot**

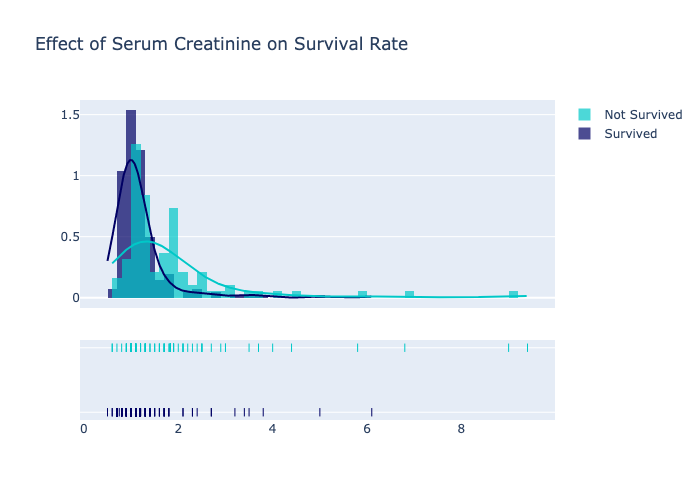
****

Fig 6.22 Serum Creatinine Dist Plot

**Data Insight 13:**

1. Of the entire population being studied, there are 96 cases who have succumbed to the heart failure condition. Out of these 96 cases, 87 cases have reported serum levels either around the higher boundary limit or higher than the normal range. This observation is inline with the research studies cited.

2. Some cases with much elevated levels of serum creatinine survived the heart failure condition.

3. Of the entire population 72.9% cases have reported elevated levels of blood serum. While 27.1% cases have serum levels in the normal range.

4. Out of the 72.9% that have reported high serum levels, 43.8% cases survived the heart failure condition while the remaining 29.1% succumbed to the condition.

5. Of the cases that have serum levels in the acceptable range 24.1% cases survived the heart failure condition while a very small number of 3.01% cases succumbed to the condition.

**6.2.4.14 Serum Sodium w.r.t Survival Rate**

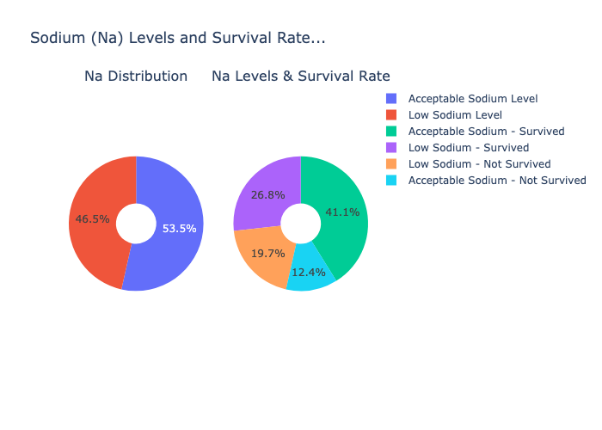
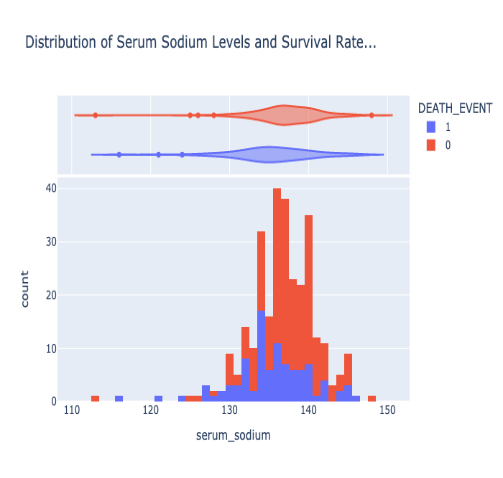
**a) serum sodium levels in pie charts b)** **Serum Sodium Histogram  **

Fig 6.23 serum sodium levels in pie charts Fig 6.24Serum Sodium Histogram

**c)** **Serum Sodium Dist Plot**

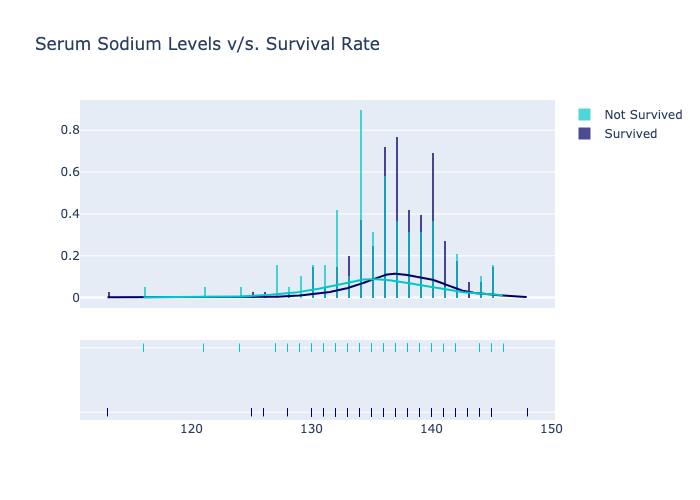
****

Fig 6.25 Serum Sodium Dist Plot

**Data Insight 14:**

1. Of the entire population being studied, there are 96 cases who have succumbed to the heart failure condition. Out of these 96 cases, 59 cases have reported lower than normal sodium levels. This observation is inline with the research studies cited.

2. Of the entire population 46.5% cases have reported lower levels of blood sodium levels. While 53.5% cases have serum levels in the acceptable range.

3. Out of the 46.5% that have reported low sodium levels, 26.8% cases survived the heart failure condition while the remaining 19.7% succumbed to the condition.

4. Of the cases that have sodium levels in the acceptable range, 41.1% cases survived the heart failure condition while a very small number of 12.4% cases succumbed to the condition.

**6.2.4.15 Distribution of DEATH\_EVENTS**

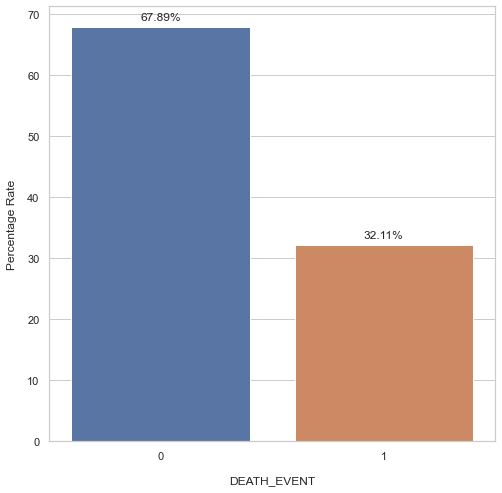


Fig 6.26 Distribution of death\_events

**Data Insight 15:**

1. In the population being studied for the condition of heart failure out of the 299 cases, 96 have succumbed to the condition while 203 cases survived.

2. In percentages, 32.11% of the cases succumbed (positive examples) while 67.89% cases survived (negative examples) the condition.

**6.3 Feature Selection Using Data correlation**

**What do you mean by feature selection in dataset?**

Feature selection, one of the main components of [feature engineering](https://www.heavy.ai/technical-glossary/feature-engineering), is the process of selecting the most important features to input in machine learning algorithms. Feature selection techniques are employed to reduce the number of input variables by eliminating redundant or irrelevant features and narrowing down the set of features to those most relevant to the machine learning model.

The main benefits of performing feature selection in advance, rather than letting the machine learning model figure out which features are most important, include:

* **simpler models**: simple models are easy to explain - a model that is too complex and unexplainable is not valuable
* **shorter training times**: a more precise subset of features decreases the amount of time needed to train a model
* **variance reduction**: increase the precision of the estimates that can be obtained for a given simulation
* **avoid the curse of high dimensionality**: dimensionally cursed phenomena states that, as dimensionality and the number of features increases, the volume of space increases so fast that the available data become limited - PCA feature selection may be used to reduce dimensionality

The most common input variable data types include: Numerical Variables, such as Integer Variables and Floating Point Variables; and Categorical Variables, such as Boolean Variables, Ordinal Variables, and Nominal Variables. Popular libraries for feature selection include sklearn feature selection, feature selection Python, and feature selection in R.

What makes one variable better than another? Typically, there are three key properties in a feature representation that makes it most desirable: easy to model, works well with regularization strategies, and disentangling of causal factors.

**‍6.3.1** **Data Correlation:**

Correlation is used to test relationships between quantitative variables or categorical variables. In other words, it's a measure of how things are related. The study of how variables are correlated is called correlation analysis. Some examples of data that have a high correlation: Your caloric intake and your weight. Correlations are useful for describing simple relationships among data. For example, imagine that you are looking at a dataset of campsites in a mountain park. You want to know whether there is a relationship between the elevation of the campsite (how high up the mountain it is), and the average high temperature in the summer.

For each individual campsite, you have two measures: elevation and temperature. When you compare these two variables across your sample with a correlation, you can find a linear relationship: as elevation increases, the temperature drops. They are negatively correlated.

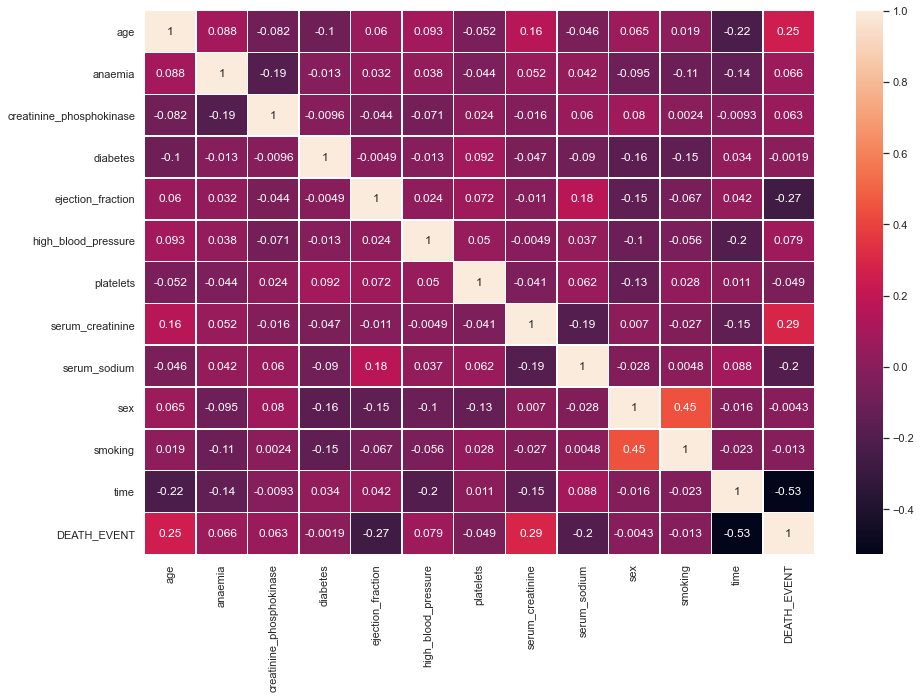


Fig 6.27 data correlation

**Features most likely related to DEATH\_EVENT**

age 0.253729

ejection\_fraction -0.268603

serum\_creatinine 0.294278

time -0.526964

DEATH\_EVENT 1.000000

Name: DEATH\_EVENT, dtype: float64

**Data Insight 16:**

1. The Correlation matrix shows the correlation among the features and their correlation with the DEATH\_EVENT.

2. Five features - 'age', 'ejection\_fraction', 'serum\_creatinine', 'serum\_sodium', 'creatinine\_phosphokinase', 'time' seem to be the most correlated to the death event when compared to the other features.

**6.4 Train – Test split**

What is Training Data?

Machine learning uses algorithms to learn from data in datasets. They find patterns, develop understanding, make decisions, and evaluate those decisions.

In machine learning, datasets are split into two subsets.

The first subset is known as the training data - it’s a portion of our actual dataset that is feed into the machine learning model to discover and learn patterns. In this way, it trains our model.

The other subset is known as the testing data

Once your machine learning model is built (with your training data), you need unseen data to test your model. This data is called testing data, and you can use it to evaluate the performance and progress of your algorithms’ training and adjust or optimize it for improved results.

**6.4.1 SMOTE**

Data Insight 17:

1. As we see in the training data, the number of 1s (80 - Death Event) are much less comapred to the number of 0s (174 - Survival Case). This is also confirmed in the Data Insight 15.

2. With this imbalance dataset (67.89% survival cases & 32.11% death events), the model could become more inclined towards learnign and predicting the negative examples than the positive (death) cases.

3. To avoid this scenario and create an un-biased model, the dataset should be balanced where we need to generate more examples of the minority class. We will apply the Synthetic Minority Oversampling Technique (SMOTE) algorithm on the training data such that more examples of the death events are generated and we have a balanced dataset.

# **6.5 Machine Learning Models**

# **6.5.1 Support Vector Machine Algorithm**

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm 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.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:

**6.5.2 Decision Tree Classification Algorithm**

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.

In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.

The decisions or the test are performed on the basis of features of the given dataset.

It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.

# A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

# Below diagram explains the general structure of a decision tree:

# Note: A decision tree can contain categorical data (YES/NO) as well as numeric data.

# Decision Tree Classification Algorithm

Fig 6.28 Decision tree structure

Why use Decision Trees?

There are various algorithms in Machine learning, so choosing the best algorithm for the given dataset and problem is the main point to remember while creating a machine learning model. Below are the two reasons for using the Decision tree:

Decision Trees usually mimic human thinking ability while making a decision, so it is easy to understand.

The logic behind the decision tree can be easily understood because it shows a tree-like structure.

Decision Tree Terminologies

Root Node: Root node is from where the decision tree starts. It represents the entire dataset, which further gets divided into two or more homogeneous sets.

# Leaf Node: Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.

# Splitting: Splitting is the process of dividing the decision node/root node into sub-nodes according to the given conditions.

# Branch/Sub Tree: A tree formed by splitting the tree.

# Pruning: Pruning is the process of removing the unwanted branches from the tree.

# Parent/Child node: The root node of the tree is called the parent node, and other nodes are called the child nodes.

**6.5.3 Random Forest Algorithm**

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "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." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

# The below diagram explains the working of the Random Forest algorithm:

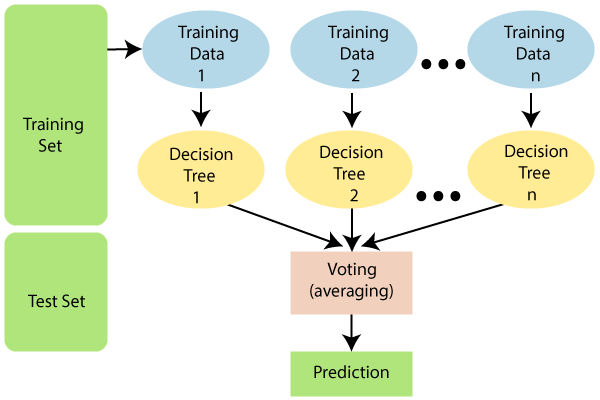


Fig 6.29 Random forest

Note: To better understand the Random Forest Algorithm, you should have knowledge of the Decision Tree Algorithm.

Assumptions for Random Forest

Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. But together, all the trees predict the correct output. Therefore, below are two assumptions for a better Random forest classifier:

There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result.

The predictions from each tree must have very low correlations.Why use Random Forest?

Below are some points that explain why we should use the Random Forest algorithm:

It takes less training time as compared to other algorithms.

# It predicts output with high accuracy, even for the large dataset it runs efficiently.

# It can also maintain accuracy when a large proportion of data is missing.

# **6.5.4 GBM in Machine Learning**

# Machine learning is one of the most popular technologies to build predictive models for various complex regression and classification tasks. Gradient Boosting Machine (GBM) is considered one of the most powerful boosting algorithms.

# Although, there are so many algorithms used in machine learning, boosting algorithms has become mainstream in the machine learning community across the world. Boosting technique follows the concept of ensemble learning, and hence it combines multiple simple models (weak learners or base estimators) to generate the final output. GBM is also used as an ensemble method in machine learning which converts the weak learners into strong learners. In this topic, "GBM in Machine Learning" we will discuss gradient machine learning algorithms, various boosting algorithms in machine learning, the history of GBM, how it works, various terminologies used in GBM, etc. But before starting, first, understand the boosting concept and various boosting algorithms in machine learning.

# What is Boosting in Machine Learning?

# Boosting is one of the popular learning ensemble modeling techniques used to build strong classifiers from various weak classifiers. It starts with building a primary model from available training data sets then it identifies the errors present in the base model. After identifying the error, a secondary model is built, and further, a third model is introduced in this process. In this way, this process of introducing more models is continued until we get a complete training data set by which model predicts correctly.

# AdaBoost (Adaptive boosting) was the first boosting algorithm to combine various weak classifiers into a single strong classifier in the history of machine learning. It primarily focuses to solve classification tasks such as binary classification.

**EXTREME GRADIENT BOOSTING MACHINE (XGBM)**

XGBM is the latest version of gradient boosting machines which also works very similar to GBM. In XGBM, trees are added sequentially (one at a time) that learn from the errors of previous trees and improve them. Although, XGBM and GBM algorithms are similar in look and feel but still there are a few differences between them as follows:

XGBM uses various regularization techniques to reduce under-fitting or over-fitting of the model which also increases model performance more than gradient boosting machines.

XGBM follows parallel processing of each node, while GBM does not which makes it more rapid than gradient boosting machines.

XGBM helps us to get rid of the imputation of missing values because by default the model takes care of it. It learns on its own whether these values should be in the right or left node.

**6.5.5 Light Gradient Boosting Machines (Light GBM)**

Light GBM is a more upgraded version of the Gradient boosting machine due to its efficiency and fast speed. Unlike GBM and XGBM, it can handle a huge amount of data without any complexity. On the other hand, it is not suitable for those data points that are lesser in number.

Instead of level-wise growth, Light GBM prefers leaf-wise growth of the nodes of the tree. Further, in light GBM, the primary node is split into two secondary nodes and later it chooses one secondary node to be split. This split of a secondary node depends upon which between two nodes has a higher loss.

Hence, due to leaf-wise split, Light Gradient Boosting Machine (LGBM) algorithm is always preferred over others where a large amount of data is given.

# **Advantages of Boosting Algorithms:**

# Boosting algorithms follow ensemble learning which enables a model to give a more accurate prediction that cannot be trumped.

# Boosting algorithms are much more flexible than other algorithms as can optimize different loss functions and provides several hyperparameter tuning options.

# It does not require data pre-processing because it is suitable for both numeric as well as categorical variables.

# It does not require imputation of missing values in the dataset, it handles missing data automatically.

# **Disadvantages of Boosting Algorithms:**

# Below are a few disadvantages of boosting algorithms:

# Boosting algorithms may cause overfitting as well as overemphasizing the outliers.

# Gradient boosting algorithm continuously focuses to minimize the errors and requires multiple trees hence, it is computationally expensive.

# It is a time-consuming and memory exhaustive algorithm.

# Less interpretative in nature, although this is easily addressed with various tools.

**6.5.6 Hyperparameter tuning**

A Machine Learning model is defined as a mathematical model with a number of parameters that need to be learned from the data. By training a model with existing data, we are able to fit the model parameters. However, there is another kind of parameters, known as Hyperparameters, that cannot be directly learned from the regular training process. They are usually fixed before the actual training process begins. These parameters express important properties of the model such as its complexity or how fast it should learn.

GridSearchCV  
 In GridSearchCV approach, machine learning model is evaluated for a range of hyperparameter values. This approach is calledGridSearchCV, because it searches for best set of hyperparameters from a grid of hyperparameters values.

For example, if we want to set two hyperparameters C and Alpha of Logistic Regression Classifier model, with different set of values. The gridsearch technique will construct many versions of the model with all possible combinations of hyerparameters, and will return the best one.

Randomized SearchCV solves the drawbacks of GridSearchCV, as it goes through only a fixed number of hyperparameter settings. It moves within the grid in random fashion to find the best set hyperparameters. This approach reduces unnecessary computation.

**6.6 Performance Metrics in Machine Learning**

Evaluating the performance of a Machine learning model is one of the important steps while building an effective ML model. To evaluate the performance or quality of the model, different metrics are used, and these metrics are known as performance metrics or evaluation metrics***.*** These performance metrics help us understand how well our model has performed for the given data. In this way, we can improve the model's performance by tuning the hyper-parameters. Each ML model aims to generalize well on unseen/new data, and performance metrics help determine how well the model generalizes on the new dataset.

In machine learning, each task or problem is divided into **classification** and **Regression**. Not all metrics can be used for all types of problems; hence, it is important to know and understand which metrics should be used. Different evaluation metrics are used for both Regression and Classification tasks. In this topic, we will discuss metrics used for classification and regression tasks.

## Performance Metrics for Classification

In a classification problem, the category or classes of data is identified based on training data. The model learns from the given dataset and then classifies the new data into classes or groups based on the training. It predicts class labels as the output, such as *Yes or No, 0 or 1, Spam or Not Spam,* etc. To evaluate the performance of a classification model, different metrics are used, and some of them are as follows:

* **Accuracy**
* **Confusion Matrix**
* **Precision**
* **Recall**
* **F-Score**
* **AUC(Area Under the Curve)-ROC**

### **I. Accuracy**

The accuracy metric is one of the simplest Classification metrics to implement, and it can be determined as the number of correct predictions to the total number of predictions.

It can be formulated as:

To implement an accuracy metric, we can compare ground truth and predicted values in a loop, or we can also use the scikit-learn module for this.

Firstly, we need to import the *accuracy\_score* function of the scikit-learn library as follows:

1. from sklearn.metrics import accuracy\_score
2. Here, metrics is a class of sklearn.
3. Then we need to pass the ground truth and predicted values in the function to calculate the accuracy.
4. print(f'Accuracy Score is {accuracy\_score(y\_test,y\_hat)}')

Although it is simple to use and implement, it is suitable only for cases where an equal number of samples belong to each class.

**When to Use Accuracy?**

It is good to use the Accuracy metric when the target variable classes in data are approximately balanced. For example, if 60% of classes in a fruit image dataset are of Apple, 40% are Mango. In this case, if the model is asked to predict whether the image is of Apple or Mango, it will give a prediction with 97% of accuracy.

**When not to use Accuracy?**

It is recommended not to use the Accuracy measure when the target variable majorly belongs to one class. For example, Suppose there is a model for a disease prediction in which, out of 100 people, only five people have a disease, and 95 people don't have one. In this case, if our model predicts every person with no disease (which means a bad prediction), the Accuracy measure will be 95%, which is not correct.

### **II. Confusion Matrix**

A confusion matrix is a tabular representation of prediction outcomes of any binary classifier, which is used to describe the performance of the classification model on a set of test data when true values are known.

The confusion matrix is simple to implement, but the terminologies used in this matrix might be confusing for beginners.

A typical confusion matrix for a binary classifier looks like the below image(However, it can be extended to use for classifiers with more than two classes).

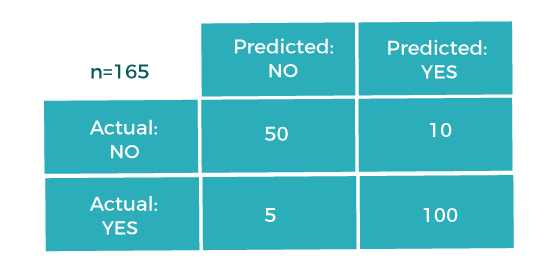


Fig 6.30 Confusion matrix

We can determine the following from the above matrix:

* In the matrix, columns are for the prediction values, and rows specify the Actual values. Here Actual and prediction give two possible classes, Yes or No. So, if we are predicting the presence of a disease in a patient, the Prediction column with Yes means, Patient has the disease, and for NO, the Patient doesn't have the disease.
* In this example, the total number of predictions are 165, out of which 110 time predicted yes, whereas 55 times predicted No.
* However, in reality, 60 cases in which patients don't have the disease, whereas 105 cases in which patients have the disease.

In general, the table is divided into four terminologies, which are as follows:

1. **True Positive(TP):** In this case, the prediction outcome is true, and it is true in reality, also.
2. True Negative(TN): in this case, the prediction outcome is false, and it is false in reality, also.
3. False Positive(FP): In this case, prediction outcomes are true, but they are false in actuality.
4. False Negative(FN): In this case, predictions are false, and they are true in actuality.

### **III. Precision**

The precision metric is used to overcome the limitation of Accuracy. The precision determines the proportion of positive prediction that was actually correct. It can be calculated as the True Positive or predictions that are actually true to the total positive predictions (True Positive and False Positive).

### **IV. Recall or Sensitivity**

It is also similar to the Precision metric; however, it aims to calculate the proportion of actual positive that was identified incorrectly. It can be calculated as True Positive or predictions that are actually true to the total number of positives, either correctly predicted as positive or incorrectly predicted as negative (true Positive and false negative).

The formula for calculating Recall is given below:

**When to use Precision and Recall?**

From the above definitions of Precision and Recall, we can say that recall determines the performance of a classifier with respect to a false negative, whereas precision gives information about the performance of a classifier with respect to a false positive.

So, if we want to minimize the false negative, then, Recall should be as near to 100%, and if we want to minimize the false positive, then precision should be close to 100% as possible.

In simple words, *if we maximize precision, it will minimize the FP errors, and if we maximize recall, it will minimize the FN error.*

### V. F-Scores

F-score or F1 Score is a metric to evaluate a binary classification model on the basis of predictions that are made for the positive class. It is calculated with the help of Precision and Recall. It is a type of single score that represents both Precision and Recall. So, ***the*** F1 Score can be calculated as the harmonic mean of both precision and Recall, assigning equal weight to each of them.

The formula for calculating the F1 score is given below:

**When to use F-Score?**

As F-score make use of both precision and recall, so it should be used if both of them are important for evaluation, but one (precision or recall) is slightly more important to consider than the other. For example, when False negatives are comparatively more important than false positives, or vice versa.

### **VI. AUC-ROC**

Sometimes we need to visualize the performance of the classification model on charts; then, we can use the AUC-ROC curve. It is one of the popular and important metrics for evaluating the performance of the classification model.

Firstly, let's understand ROC (Receiver Operating Characteristic curve) curve. *ROC represents a graph to show the performance of a classification model at different threshold levels*. The curve is plotted between two parameters, which are:

* **True Positive Rate**
* **False Positive Rate**

TPR or true Positive rate is a synonym for Recall, hence can be calculated as:

FPR or False Positive Rate can be calculated as:

To calculate value at any point in a ROC curve, we can evaluate a logistic regression model multiple times with different classification thresholds, but this would not be much efficient. So, for this, one efficient method is used, which is known as AUC.

### AUC: Area Under the ROC curve

AUC is known for **Area Under the ROC curve**. As its name suggests, AUC calculates the two-dimensional area under the entire ROC curve, as shown below image:

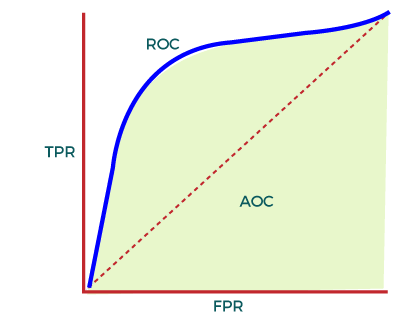


Fig 6.31 AOC – ROC Curve

AUC calculates the performance across all the thresholds and provides an aggregate measure. The value of AUC ranges from 0 to 1. It means a model with 100% wrong prediction will have an AUC of 0.0, whereas models with 100% correct predictions will have an AUC of 1.0.

**When to Use AUC**

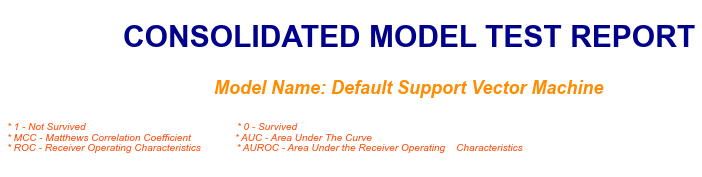
AUC should be used to measure how well the predictions are ranked rather than their absolute values. Moreover, it measures the quality of predictions of the model without considering the classification threshold.

**When not to use AUC**

As AUC is scale-invariant, which is not always desirable, and we need calibrating probability outputs, then AUC is not preferable.

Further, AUC is not a useful metric when there are wide disparities in the cost of false negatives vs. false positives, and it is difficult to minimize one type of classification error

**6.7 RESULTS**



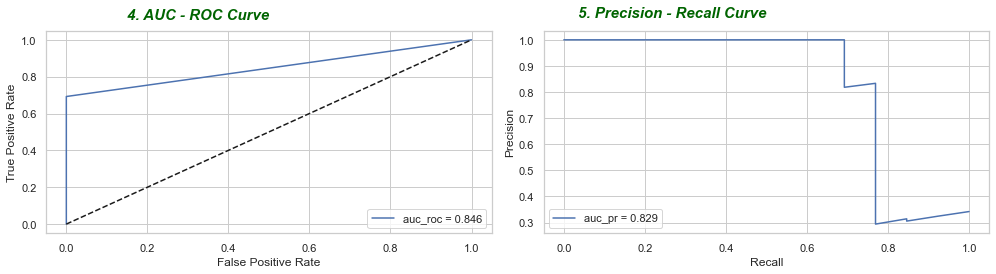
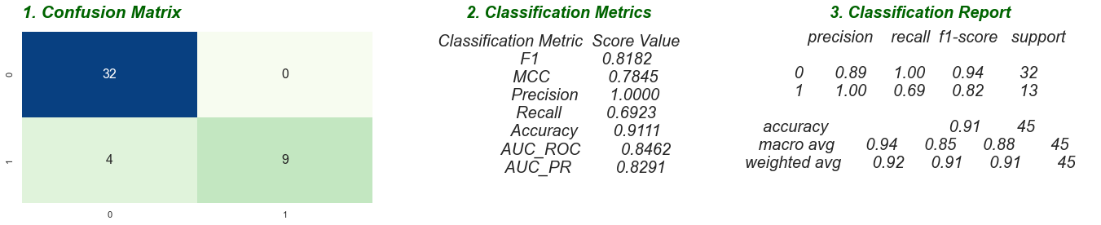


Fig 6.32 Consolidated model test report on default SVM

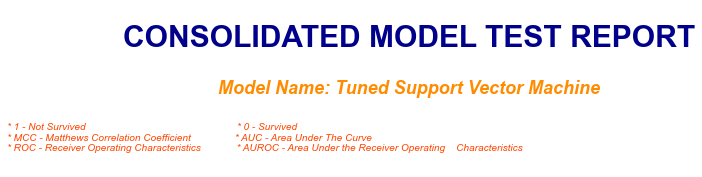
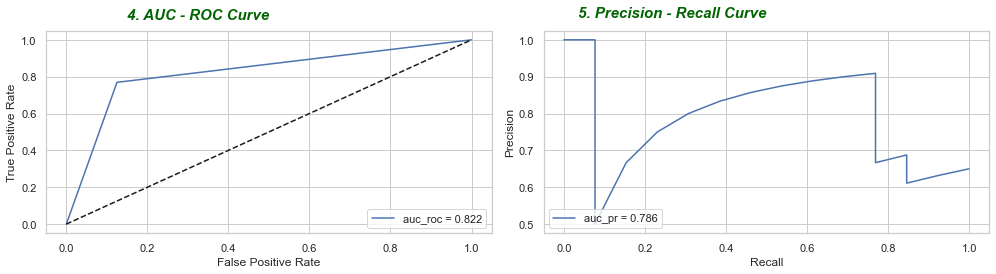
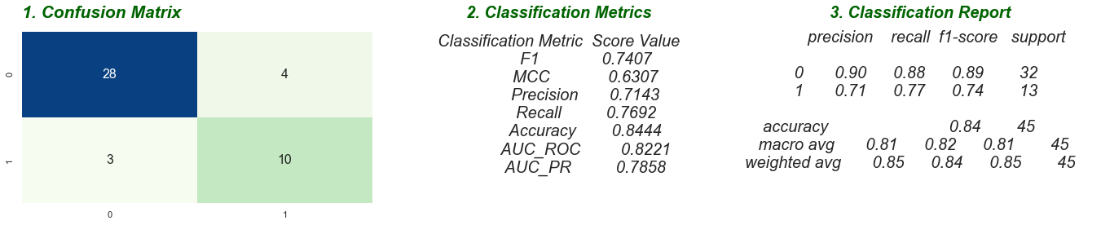
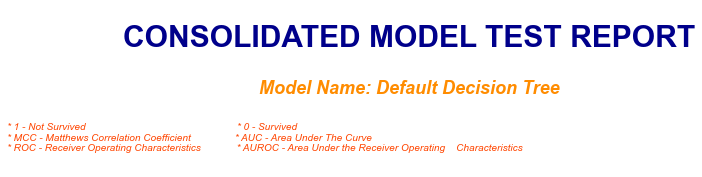
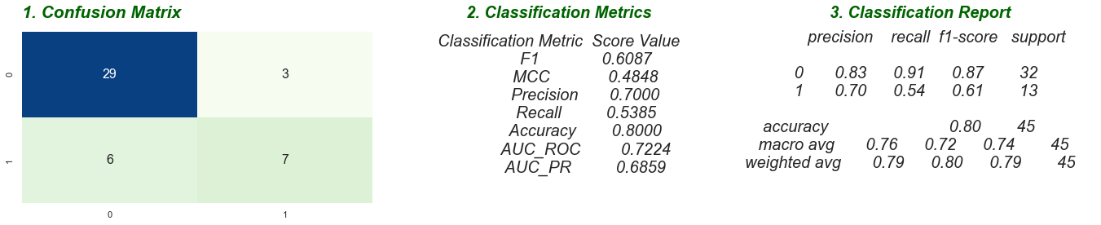
  


Fig 6.33 Consolidated model test report on tuned SVM





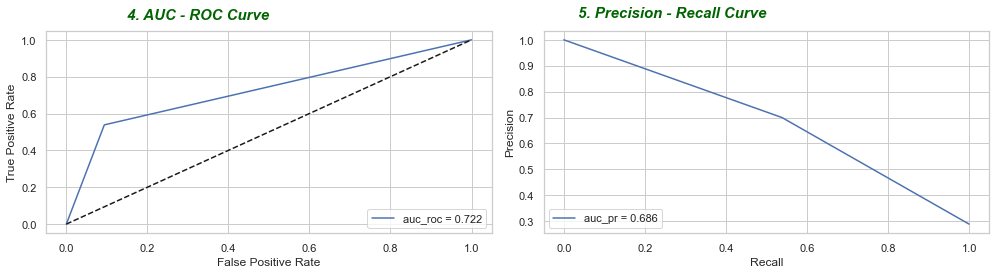
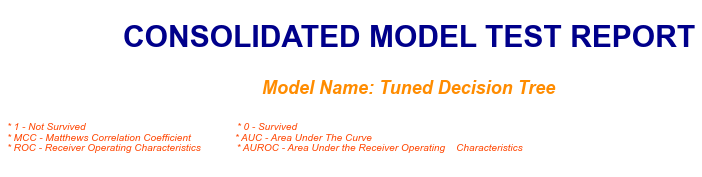
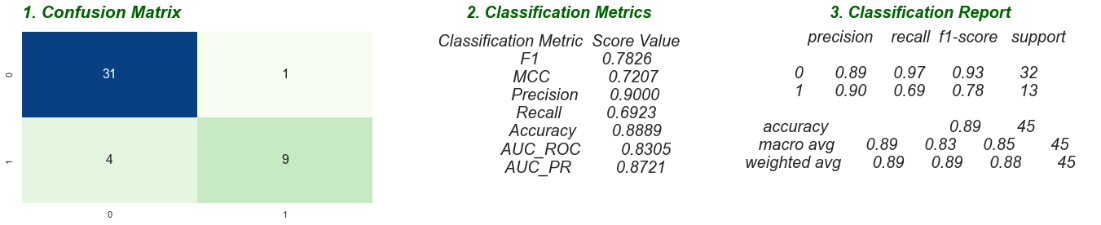


Fig 6.34 Consolidated model test report on default Decision Tree





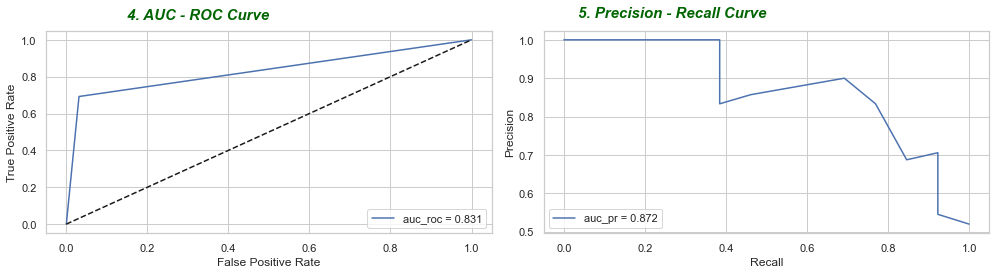
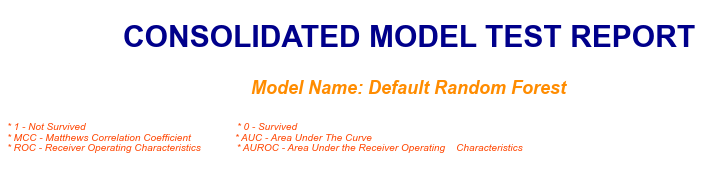
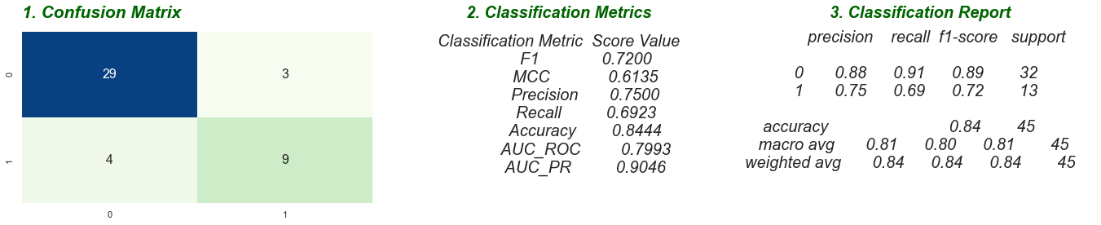


Fig 6.35 Consolidated model test report on tuned Decision Tree





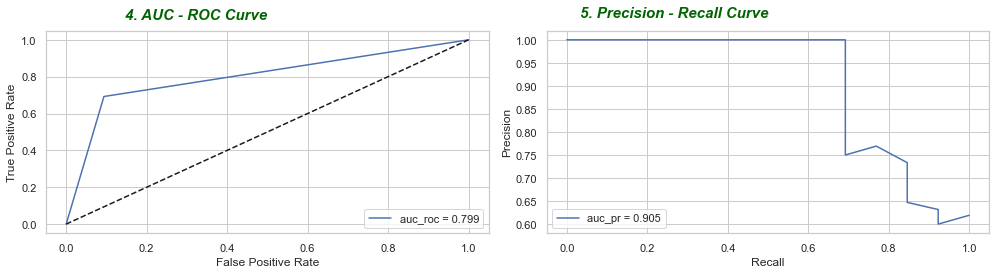
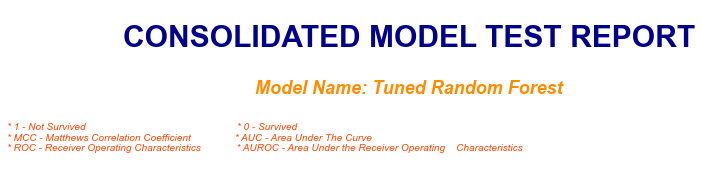
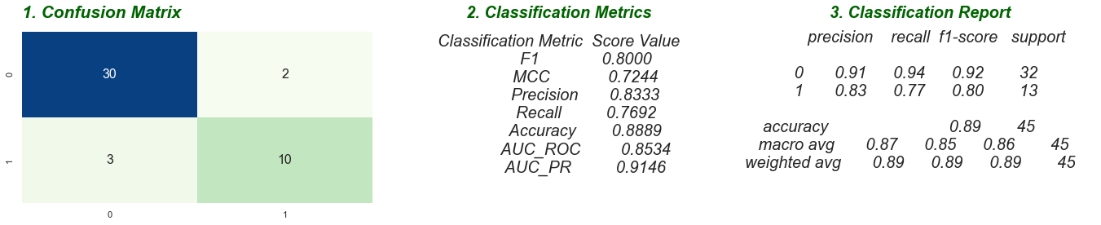


Fig 6.36 Consolidated model test report on default Random Forest





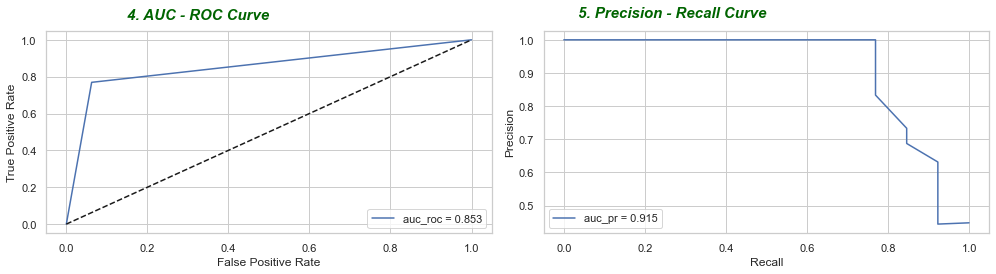
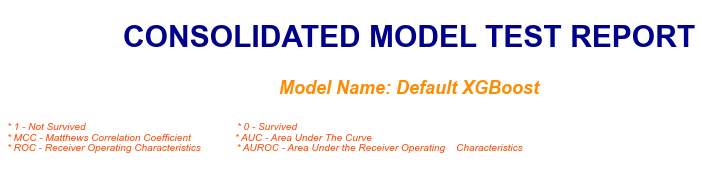
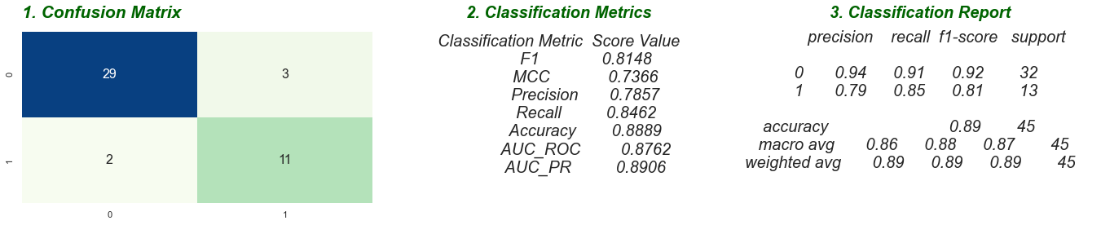


Fig 6.37 Consolidated model test report on tuned Random Forest





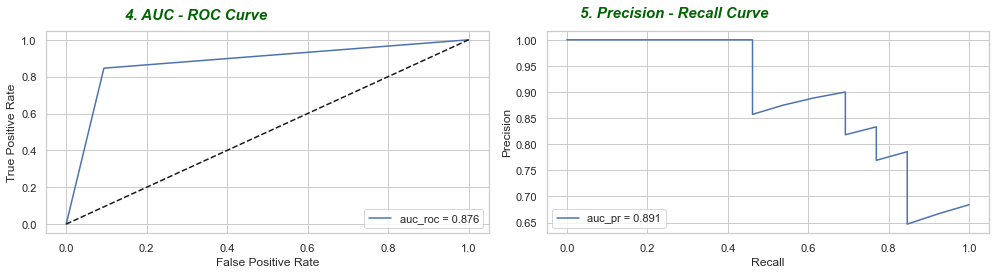
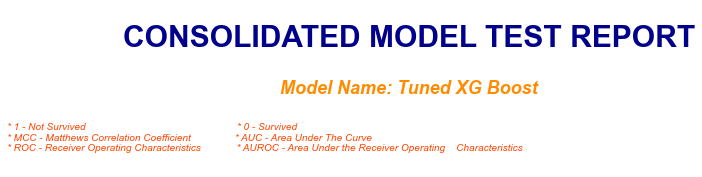
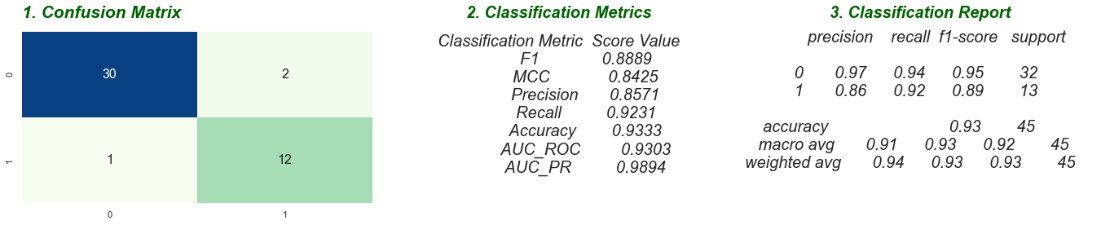


Fig 6.38 Consolidated model test report on default XGBoost





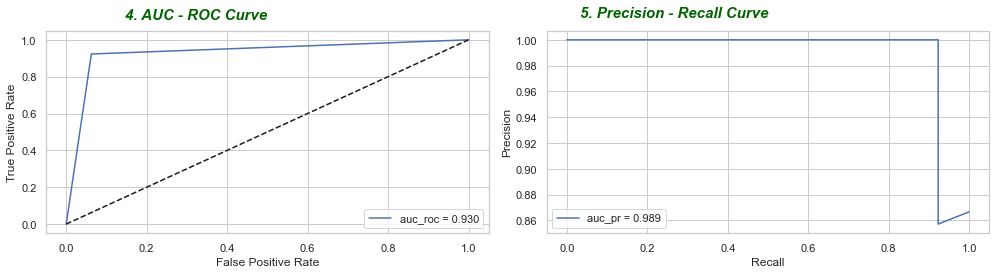
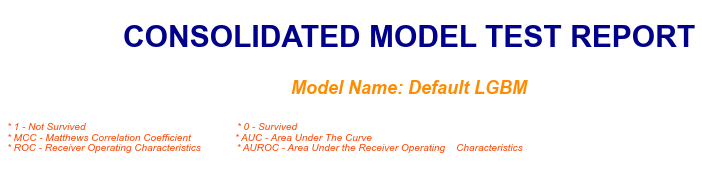
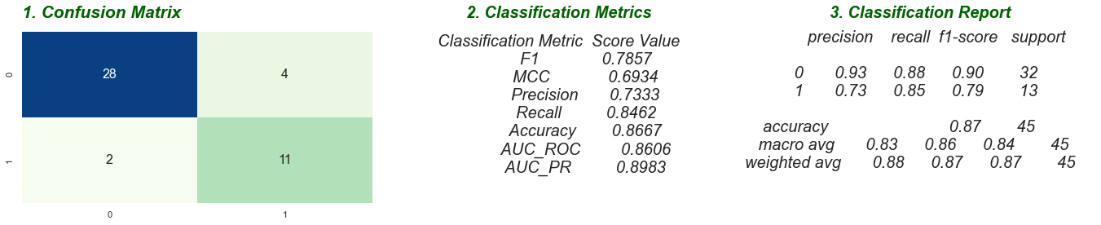


Fig 6.39 Consolidated model test report on tuned XGBoost





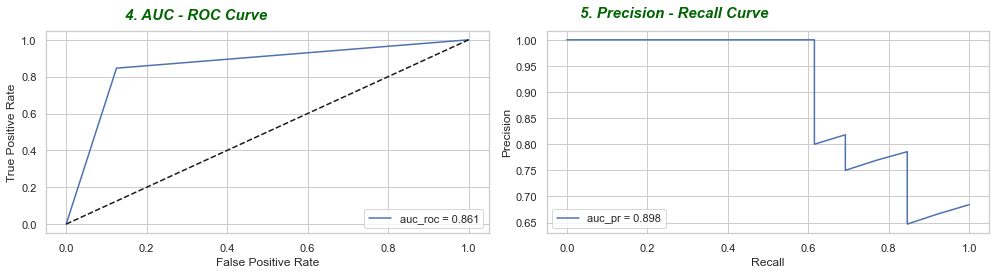
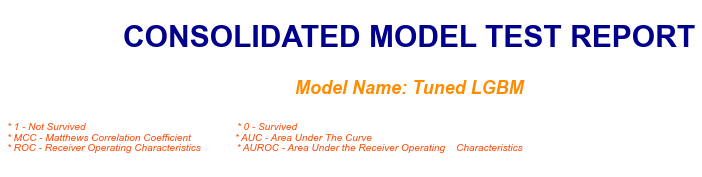
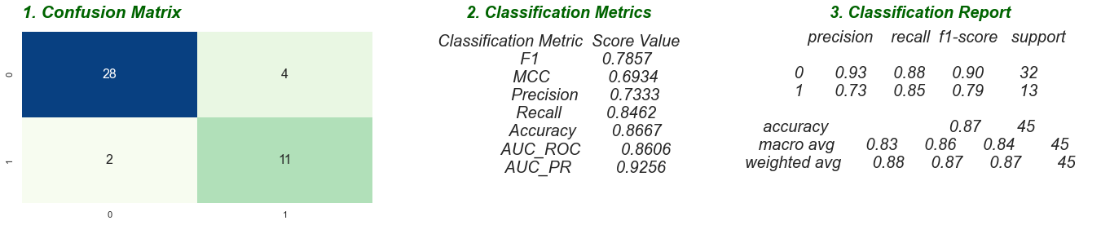


Fig 6.40 Consolidated model test report on default LGBM





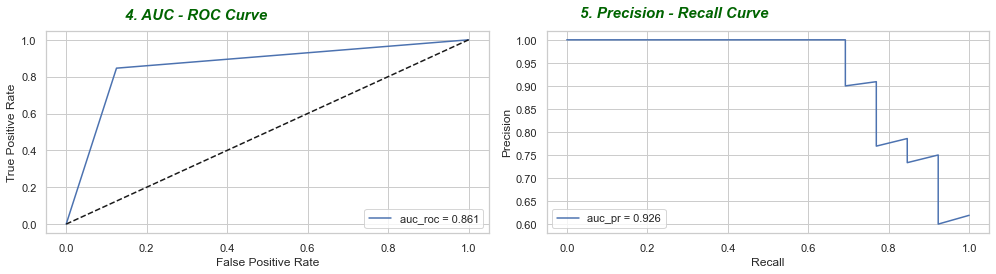


Fig 6.41 Consolidated model test report on tuned LGBM

**6.8 MODELS COMPARISION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ML algorithms | Accuracy | F1 score | precision | recall |
| SVM (d) | 0.91 | 0.84 | 1.00 | 0.69 |
| SVM(t) | 0.84 | 0.74 | 0.71 | 0.77 |
| Decision tree(d) | 0.80 | 0.61 | 0.70 | 0.54 |
| Decision tree(t) | 0.89 | 0.78 | 0.90 | 0.69 |
| Random forest(d) | 0.84 | 0.72 | 0.75 | 0.69 |
| Random forest(t) | 0.88 | 0.80 | 0.83 | 0.77 |
| XGBM(d) | 0.89 | 0.81 | 0.79 | 0.85 |
| XGBM(t) | 0.93 | 0.89 | 0.86 | 0.92 |
| LGBM(d) | 0.87 | 0.79 | 0.73 | 0.85 |
| LGBM(t) | 0.87 | 0.79 | 0.73 | 0.85 |

Table 6.8 Comparison of modules

**SVM, Decision Tree, XGBM, LGBM turn out to be the better models in predicting the event of death given**

**CHAPTER 7**

**CONCLUSION**

Heart failure is a serious condition and there is no cure for this disease. It is a situation in which the patient's heart is not pumping the blood well as the normal heart pumps. Heart Failure prediction is a complex task in the medical field The features are 'age', 'ejection\_fraction', 'serum\_creatinine', 'serum\_sodium', 'time' seem to be the more closely related to the death event. The event of a heart failure in the old age could be fatal. The levels of 'ejection\_fraction', 'serum\_creatinine', 'serum\_sodium' play an important role in the event of heart failure or the abnormalities in the levels of these become an important factor in triggerring heart failure condition.

SVM, XGBoost and Light GBM turn out to be the better models in predicting the event of death given the vital parameters that effect heart failure of Cardiovascular conditions.

Machine Learning models can prove to be a time saving factor in predicting the event of death and can help the doctors take additional precautionary measures with critical care.

The imbalance in the dataset had been dealt using the SMOTE oversampling of the minority class. SMOTE balanced the no. of instances for each class but the amount of data is still limited.

The accuracy and model performance can be increased as more data is collected and made available for the model to train on. This data limitation, limits the model performance to some extent. As more and more data becomes available, the prediction accuracy and model's ability to act on other similar real-time data could be enhanced.

XGBM, LGBM, SVM, Decision Tree turn out to be the better models in predicting the event of death given.

**REFERENCE:**

[1]   Sellappan Palaniappan, Rafiah Awang,**―**Intelligent Heart Disease Prediction System Using DataMining Techniques**‖**; 978-1-4244-1968-5/08/$25.00©2008 IEEE.

[2]  M. Anbarasi, E. Anupriya, N.Ch.S.N.Iyengar,**―**Enhanced Prediction of Heart Disease with Feature Subset Selection using Genetic Algorithm**‖;** International Journal of Engineering Science and Technology, Vol. 2(10), 2010.

[3]  Asha Rajkumar, G. Sophia Reena, **―**Diagnosis of Heart Disease Using Datamining Algorithm**‖;** Global Journal of Computer Science and Technology, Page | 38 Vol. 10 Issue 10 Ver. 1.0 September, 2010.

[4]  Chaitrali S. Dangare, Sulabha S. Apte**, ―**Improved Study of Heart Disease Prediction System using Data Mining Classification Techniques**‖;** International Journal of Computer Applications (0975 – 888) Volume 47– No.10, June 2012.

[5] V. V. Ramalingam, A. Dandapath, and M. Karthik Raja, ―Heart disease prediction using machine learning techniques : a survey,‖ Int. J. Eng. Technol., vol. 7, no. 2.8, pp. 684–687, 2018.

[6] Amin ul haq, Jian ping li, Muhammed Hammad Memon, Shah nazir and Ruinan sun—A Hybrid Intelligent System Framework for the Prediction of Heart Disease using Machine Learning Algorithms, volume 2018, Article id 3860146

[7] Fahd Saleh Alotaibi,"Implementation of machine learning model to predict heart failure disease"(IJACSA) International Journal of Advanced Computer Science and Applications, 10 (6) (2019)A

[8]  Davide Chicco, Giuseppe Jurman: Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. BMC Medical Informatics and Decision Making 20, 16 (2020).

[9]  Apurb Rajdhan, Milan Sai, Avi Agarwal, Dundigalla Ravi and Dr. Poonam Ghuli, “Heart Disease Prediction using Machine Learning”, International Journal of Engineering Research & Technology (IJERT) http://www.ijert.org ISSN: 2278-0181 IJERTV9IS040614 (This work is licensed under a Creative Commons Attribution 4.0 International License.) Published by : www.ijert.org Vol. 9 Issue 04, April-2020.

[10]  Galla Siva Sai Bindhika, Munaga Meghana, Manchuri Sathvika Reddy and Rajalakshmi, “Heart Disease Prediction Using Machine Learning Techniques”, International Research Journal of Engineering and Technology (IRJET) Volume: 07 Issue: 04 | Apr 2020 [www.irjet.net](http://www.irjet.net).