HEART DISEASE PREDICTION SYSTEM

A Project Report submitted in the partial fulfillment of the

Requirements for the award of the degree

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by

K. Sampath Kumar (19471A0504)

Ch. Sai Ganesh (18471A05J2)

V. Raju (18471A05N5)

Under the esteemed guidance of

M. Seerisha M.Tech

Assoc. Professor.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NARASARAOPETA ENGINEERING COLLEGE

(AUTONOMOUS)

(Affiliated to JNTUK, Kakinada, Approved by AICTE &Thrice Accredited by NBA)

2020-2021

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project that is entitled with the name "**HEART DISEASES PREDICTION SYSTEM**" is a bonafide work done by the team **K. Sampath Kumar (16471A0584), Ch. Sai Ganesh (18471A05J2), V. Raju (18471A05N5)** in partial fulfillment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY** in the Department of **COMPUTER SCIENCE AND ENGINEERING** during 2020-2021.

PROJECT GUIDE

Ms Sireesha .M, M.Tech.,
Assoc. Professor

PROJECT CO-ORDINATOR

Ms Sireesha .M, M.Tech.,
Assoc. Professor

HEAD OF THE DEPARTMENT

Dr. S. N. TirumalaRao, M.Tech., Ph.D.,
Professor

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We wish to express my thanks to carious personalities who are responsible for the completion of the project. We are extremely thankful to our beloved chairman sri **M.V.Koteswara Rao,B.Sc.,** who took keen interest in us in every effort throughout this course. We owe out sincere gratitude to our beloved principal **Dr.M.Sreenivasa Kumar, M.Tech., Ph.D., MISTE., FIE(I).,** for showing his kind attention and valuable guidance throughout the course.

We express our deep felt gratitude towards **Dr.S.N.Tirumala Rao, M.Tech.,Ph.D.,** HOD of CSE department and also to our guide **Ms Seerisha. M,M.Tech.,** of CSE department whose valuable guidance and unstinting encouragement enable us to accomplish our project successfully in time.

We extend our sincere thanks towards **Ms Sireesha .M, M.Tech.,** Associate professor & Project coordinator of the project for extending her encouragement. Their profound knowledge and willingness have been a constant source of inspiration for us throughout this project work.

We extend our sincere thanks to all other teaching and non-teaching staff to department for their cooperation and encouragement during our B.Tech degree.

We have no words to acknowledge the warm affection, constant inspiration and encouragement that we received from our parents.

We affectionately acknowledge the encouragement received from our friends and those who involved in giving valuable suggestions had clarifying out doubts which had really helped us in successfully completing our project.

By

K. Sampath Kumar (16471A0584)

Ch. Sai Ganesh (18471A05J2)

V. Raju (18471A05N5)

ABSTRACT

Heart disease is one of the most significant causes of mortality in today's world. Heart disease proves to be the leading cause of death for both men and women. This affects the human life very badly. The diagnosis of heart disease in most cases depends on a complex combination and huge volume of clinical and pathological data. Accurate and on time diagnosis of heart disease is important for health failure prevention and treatment.

Machine learning has been shown to be effective assisting in making decisions and predictions from the large quantity of data produced by the health care industry. Various traditional machine learning algorithms were available that aims in improving the accuracy of heart disease prediction.

To address this issue, surrogate data is generated from kaggle dataset .The datasets used are classified in terms of medical parameters. The datasets are processed in python programming using Machine learning algorithms namely logistic regression , Decision Tree, Naïve Bayes. This project gives us significant Knowledge that can help us predict patients with heart disease.

INDEX

LIST OF FIGURES

1. INTRODUCTION

1.1 Introduction

Heart is one such organ which pumps blood throughout the body and if it does not do so, the human body can have fatal circumstances. Heart diseases have become a major concern to deal with as studies show that the number of deaths due to heart diseases have increased significantly over the past few decades in India, in fact it has become the leading cause of death in India.

Early detection and treatment of several heart diseases is very complex, especially in developing countries, because of the lack of diagnostic centers and qualified doctors and other resources that affect the accurate prognosis of heart disease. Thus, preventing Heart diseases has become more than necessary. Identification of any heart related illness at primary stage can reduce the death risk.

Good data-driven systems for predicting heart diseases can improve the entire research and prevention process, making sure that more people can live healthy lives. With this concern, in recent times computer technology and machine learning techniques are being used to make medical aid software as a support system for early diagnosis of heart disease.

This is where Machine Learning comes into play. Machine Learning helps in predicting the Heart diseases, and the predictions made are quite accurate. Various ML techniques are used in medical data to understand the pattern of data and making prediction from them. Healthcare data are generally massive in volumes and complex in structure.

ML algorithms are capable to handle the big data and mine them to find the meaningful information. Machine Learning can play an essential role in predicting presence/absence of Locomotor disorders, Heart diseases and more. Such information, if predicted well in advance, can

encourage cardiologists in taking quicker actions so more patients can get medicines within a shorter timeframe, thus saving large number of lives.

We have four main types of Machine learning Methods based on the kind of learning we expect from the algorithms: Here we going to use supervised machine learning algorithm.

Supervised learning is the types of machine learning in which machines are trained using well "labelled" training data, and on basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output. In supervised learning, the training data provided to the machines work as the supervisor that teaches the machines to predict the output correctly. It applies the same concept as a student learns in the supervision of the teacher.

Supervised learning is a process of providing input data as well as correct output data to the machine learning model. The aim of a supervised learning algorithm is to find a mapping function to map the input variable(x) with the output variable(y)**.**In the real-world, supervised learning can be used for Risk Assessment, Image classification, Fraud Detection, heart disease prediction, spam filtering, etc.

In supervised learning, models are trained using labelled dataset, where the model learns about each type of data. Once the training process is completed, the model is tested on the basis of test data, and then it predicts the output.

Supervised learning can be further divided into two types of problems:

1Regression: Regression algorithms are used if there is a relationship between the input variable and the output variable. It is used for the prediction of continuous variables, such as Weather forecasting, Market Trends, etc.

2. Classification: Classification algorithms are used when the output variable is categorical, which means there are two classes such as Yes-No, Male-Female, True-false, etc.

So, in our project Machine learning (ML) plays a significant role in heart disease prediction. It predicts whether the patient has a heart disease or not based on an efficient learning technique. Here, we are utilizing supervised learning techniques for predicting the early stage of heart disease. Now we are going to use some supervised machine learning algorithms such as a Logistic regression and k-nearest neighbor (KNN), algorithms to classify whether the people tested belong to the class of heart disease or healthy people.

In modern days, Machine learning algorithms are being the solution for different medical fields, for this case also we can use machine learning algorithms to predict the heart diseases. Here we are going to compare the accuracy of different machine learning technique over "heart.csv" dataset and conclude which algorithm gives the best result.

And we also will use Streamlit. Streamlit is an open source python library. It is used to create and share data web apps..

1.2 Existing System

Very few systems use the available clinical data for prediction purpose.

Diagnosis solely depends upon the doctors intuition and patients records.

Practical use of various collected data is time consuming.

Drawbacks:

It is time consuming and need a cardiac specialist.

It is not more accurate.

1.3 Proposed System

Cardiovascular disease (CVD) is increasing rapidly in the modern world. Millions of people die due to cardiovascular disease. The diagnosis of heart disease is a challenging task. The project aims to determine whether a person is diagnosed with a heart disease or not. We are implementing different machine learning classification algorithms to predict heart disease.

After evaluating the results from the existing methodologies, we have used python and pandas operations to perform heart disease classification for the data obtained from the UCI repository. It provides an easy-to-use visual representation of the dataset, working environment and building the predictive analytics. ML process starts from a preprocessing data phase followed by feature selection based on data cleaning, classification of modelling performance evaluation.

Advantages:

1. Generates accurate and efficient results
2. Computation time is greatly reduced
3. Easy maintenance of patient records
4. Reduces manual work
5. Efficient further treatment
6. Automated prediction

1.4 System Requirements

Hardware Requirements:

- SystemType : Intel Core i3 or above
- Cachememory : 4MB(Megabyte)
- RAM : 8 gigabyte (GB)
- BusSpeed : 5 GT/s DBI2
- Number of cores : 2
- Number of threads : 4

Software Requirements:

- Operating System : Windows 10 Home, 64 bit Operating System
- Coding Language : Python
- Python distribution : Google Colab, Pycharm (FrontEnd)

2. LITERATURE SURVEY

2.1 Machine Learning

Machine learning is one of the applications of artificial intelligence (AI) that provides computers, the ability to learn automatically and improve from experience instead of explicitly programmed. It focuses on developing computer programs that can access data and use it to learn from themselves. The main aim is to allow computers to learn automatically without human intervention and also adjust actions accordingly.

2.2 Some machine learning methods

Machine learning algorithms are often categorized as supervised and unsupervised.

- **Supervised machine learning algorithms** can apply what has been learned in the past to new data using labeled examples to predict future events. Starting from the analysis of a known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. The system is able to provide targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors in order to modify the model accordingly.
- In contrast, **unsupervised machine learning algorithms** are used when the information used to train is neither classified nor labeled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabeled data. The system doesn't figure out the right output, but it explores the data and can draw inferences from datasets to describe hidden structures from unlabeled data.
- **Semi-supervised machine learning algorithms** fall somewhere in between supervised and unsupervised learning, since they use both labeled and unlabeled data for training – typically a small amount of labeled data and a large amount of unlabeled data. The systems that use this method are able to considerably improve learning accuracy. Usually, semi-supervised learning is

chosen when the acquired labeled data requires skilled and relevant resources in order to train it / learn from it. Otherwise, acquiring unlabeled data generally doesn't require additional resources.

- **Reinforcement machine learning algorithms** is a learning method that interacts with its environment by producing actions and discovers errors or rewards. Trial and error search and delayed reward are the most relevant characteristics of reinforcement learning. This method allows machines and software agents to automatically determine the ideal behaviour within a specific context in order to maximize its performance. Simple reward feedback is required for the agent to learn which action is best. This is known as there inforcement signal.

2.3 Applications of machine learning

1. Virtual Personal Assistants
2. Predictions while Commuting
3. Videos Surveillance
4. Social Media Services
5. Email Spam and Malware Filtering
6. Online Customer Support
7. Search Engine Result Refining
8. Product Recommendations
9. Online Fraud Detection

2.4 Importance of machine learning in healthcare

The importance of machine learning in healthcare is increasing because of its ability to process huge datasets efficiently beyond the range of human capability, and then dependably convert analysis of that data into clinical insights that assist physicians in planning and providing care, which ultimately leads to better outcomes, reduces the costs of care, and increases patients satisfaction. Using these types of

advanced analytics, we can provide better information to doctors at the point of patientcare.

2.5 Classification

- It is a process of categorizing data into given classes. Its primary goal is to identify the class of our new data.

2.6.1 Machine learning algorithms for classification

Research on data mining has led to the formulation of several data mining algorithms. These algorithms can be directly used on a dataset for creating some models or to draw vital conclusions and inferences from that dataset. Some popular data mining algorithms are Decision tree, Naïve Bayes, k-means, artificial neural network etc.

1. **Decision Tree**: Decision Tree Analysis is a general, predictive modelling tool that has applications spanning a number of different areas. In general, 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 supervised learning method used for both classification and regression tasks. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

The decision rules are generally in form of if-then-else statements. The deeper the tree, the more complex the rules and fitter the model.

2. **Naive Bayes (NB):** It is a simple technique for constructing classifiers. It is a probabilistic classifier based on Bayes' theorem. All Naive Bayes classifiers assume that the value of any particular feature is independent of the value of anyother feature, given the class variable. Bayes theorem is given as follows**:** **P(C|X) = P(X|C) * P(C)/P(X)**, where X is the data tuple and C is the class such that P(X) is constant for all classes. Though it assumes an unrealistic condition that attribute values are conditionally independent, it performs surprisingly well on large datasets where this condition is assumed and holds.

3.**Random Forest:** Random Forests are an ensemble learning method (also thought of as a form of nearest neighbour predictor) for classification and regression techniques. It builds multiple decision trees and then merges them together in-order to get more accurate and stable predictions. It constructs a number of Decision trees at training time and outputs. The class that is the mode of the classes output by individual trees. It also tries to minimize the problems of high variance and high bias by averaging to find a natural balance between the two extremes. Both R and Python have robust packages to implement this algorithm.

4. KNN: KNN algorithm is one of the simplest classification algorithms and it is one of the most used learning algorithms. KNN is a non-parametric, lazylearning algorithm. Its purpose is to use a dataset in which the data points are separated into several classes to predict the classification of a new sample point. A KNN algorithm uses a data and classifies new data points based on a similarity measures (e.g. distance function, error rate). Classification is done by a majority vote to its neighbours. The data is assigned to the class which has the most nearest neighbours. As we increase the number of nearest neighbours, the value of k, accuracy may increase.

When we say a technique is non-parametric, it means that it does not make any assumptions on the underlying data distribution. In other words, the model structure is determined from the data. If you think about it, it's pretty useful, because in the "real world", most of the data does not obey the typical theoretical assumptions made (as in linear regression models, for example). Therefore, KNN could and probably should be one of theirst choices for a classification study when there is little or no prior knowledge about the distribution data.

5. Logistic Regression: Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1. Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems.

2.6 Implementation of machine learning using Python

Python is a popular programming language. It was created in 1991 by Guido van Rossum. It is used for:

- web development(server-side),
- software development,
- mathematics,
- system scripting.

The most recent major version of Python is Python 3. However, Python 2, although not being updated with anything other than security updates, is still quite popular.

It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse, Anaconda which are particularly useful when managing larger collections of Python files.Python was designed for its readability. Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose. In the older days, people used to perform Machine Learning tasks manually by coding all the algorithms and mathematical and statistical formula. This made the process time consuming, tedious and inefficient. But in the modern days, it is become very much easy and efficient compared to the olden days by various python libraries, frameworks, and modules. Today, Python is one of the most popular programming

languages for this task and it has replaced many languages in the industry, one of the reason is its vast collection of libraries. Python libraries that used in Machine Learning are:

- Numpy
- pandas
- seaborn
- matplotlib.pyplot

NumPy: is a very popular python library for large multi-dimensional array and matrix processing, with the help of a large collection of high-level mathematical functions. It is very useful for fundamental scientific computations in Machine Learning. It is particularly useful for linear algebra, Fourier transform, and random number capabilities. High-end libraries like TensorFlow uses NumPy internally for manipulation of Tensors.

Pandas: is a popular Python library for data analysis. It is not directly related to Machine Learning. As we know that the dataset must be prepared before training. In this case, Pandas comes handy as it was developed specifically for data extraction and preparation. It provides high-level data structures and wide variety tools for data analysis. It provides many inbuilt methods for groping, combining and filtering data.

Matpoltlib: is a very popular Python library for data visualization. Like Pandas, it is not directly related to Machine Learning. It particularly comes in handy when a programmer wants to visualize the patterns in the data. It is a 2D plotting library used for creating 2D graphs and plots. A module named pyplot makes it easy for programmers for plotting as it provides features to control line styles, font properties, formatting axes, etc. It provides various kinds of graphs and plots for datavisualization, histogram, errorcharts, barchats, etc.

2.7 Need of Data Pre-processing

Pre-processing refers to the transformations applied to our data before feeding it to the algorithm. Data Pre-processing is a technique that is used to convert the raw data into a clean dataset. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis.

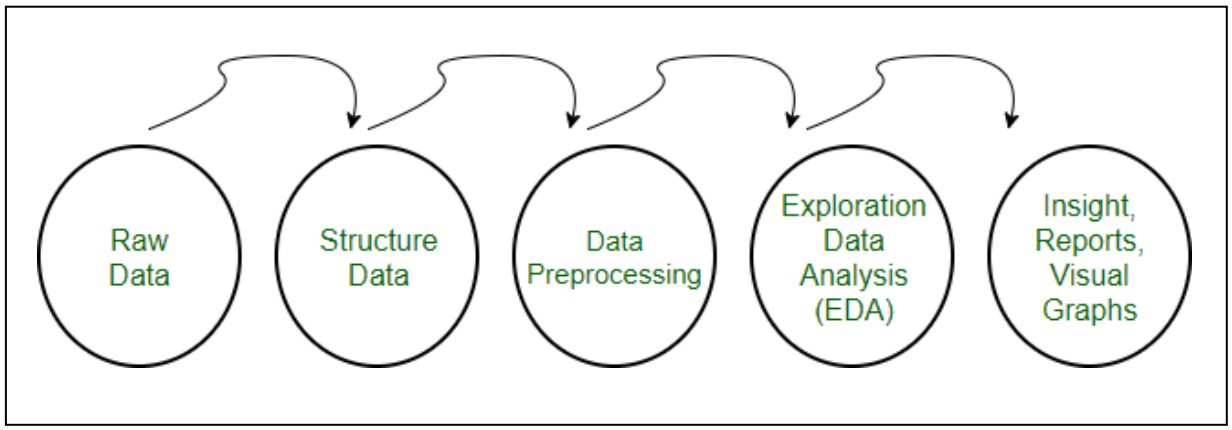
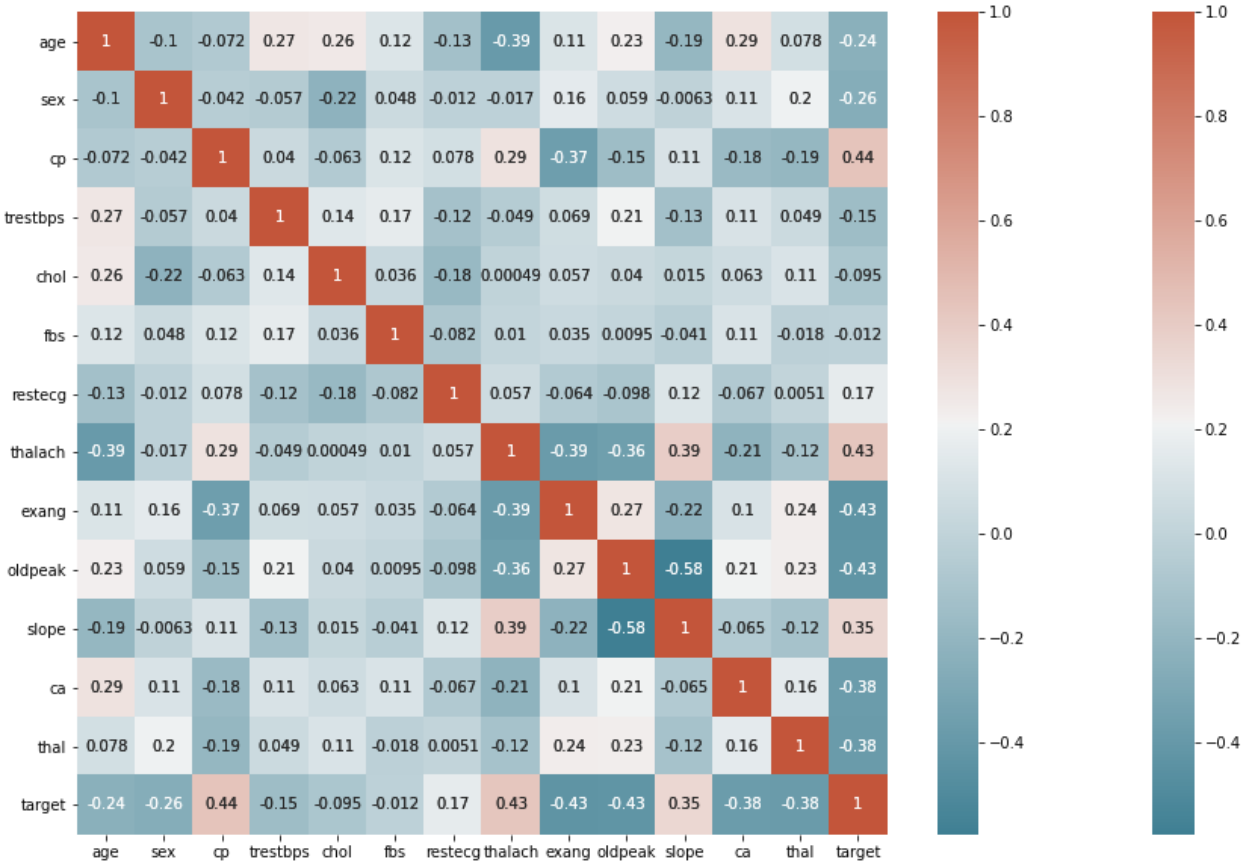


Fig 2.8.1: Need of Data Preprocessing

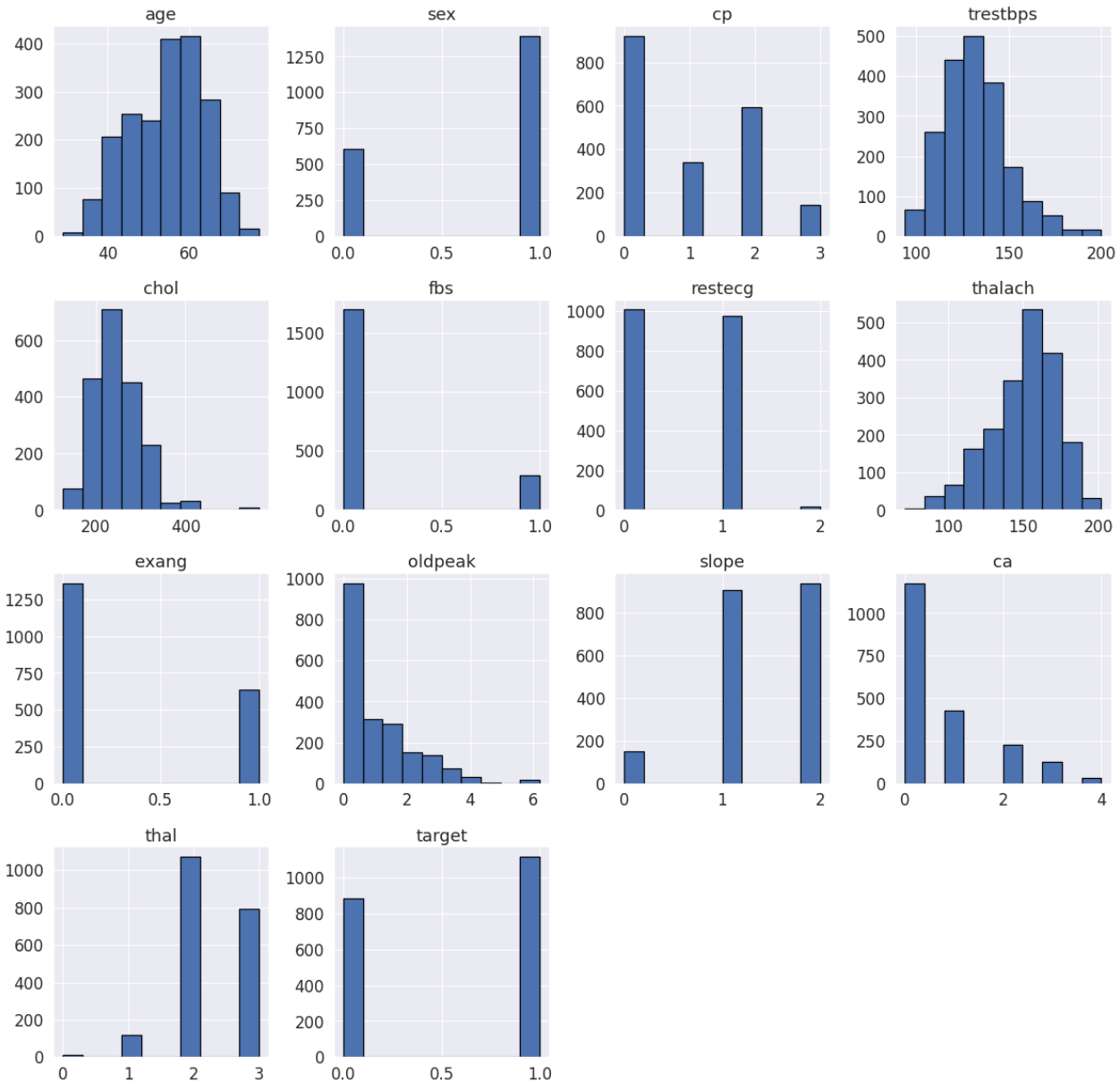
For achieving better results from the applied model in Machine Learning projects the format of the data has to be in a proper manner. Some specified Machine Learning model needs information in a specified format. For example, Random Forest algorithm does not support null values, therefore to execute random forest algorithm null values have to be managed from the original raw data set.

Another aspect is that dataset should be formatted in such a way that more than one Machine Learning and Deep Learning algorithms are executed in one dataset, and best out of them is chosen.

2.8 Machine learning products



2.8.2 Correlation matrix



2.8.3 Categorical values

3. SYSTEM ANALYSIS

3.1 Scope of the project

The scope of this system is to maintain patient details in datasets, train the model using the large quantity of data present in datasets and predict whether presence or absence of disease on new data during testing.

3.2 Analysis

The dataset that we use for the prediction of Heart disease was taken from the kaggle reposotary which contain the details of following attributes to train and test the system for the prediction

1. age –age
2. Gender
3. Chest Pain
4. Trustbps
5. Chol
6. Fbs
7. Restecg
8. Thalach
9. Exang
10. Old peak
11. Slop
12. Ca
13. Thal
14. Target

3.3 Data Preprocessing

1. **age** (#)

2. **sex** : 1= Male, 0= Female (*Binary*)

3. (**cp**)chest pain type (4 values -*Ordinal*):Value 1: typical angina ,Value 2: atypical angina, Value 3: non-anginal pain , Value

4. asymptomatic

4. (**trestbps**) resting blood pressure (#)

5. (**chol**) serum cholesterol in mg/dl (#)

6. (**fbs**)fasting blood sugar > 120 mg/dl(*Binary*)(1 = true; 0 = false)

7. (**restecg**) resting electrocardiography results(values 0,1,2)

8. (**thalach**) maximum heart rate achieved (#)

9. (**exang**) exercise induced angina (*binary*) (1 = yes; 0 = no)

10. (**oldpeak**) = ST depression induced by exercise relative to rest (#)

11. (**slope**) of the peak exercise ST segment (*Ordinal*) (Value 1: up sloping , Value 2: flat, Value 3: down sloping)

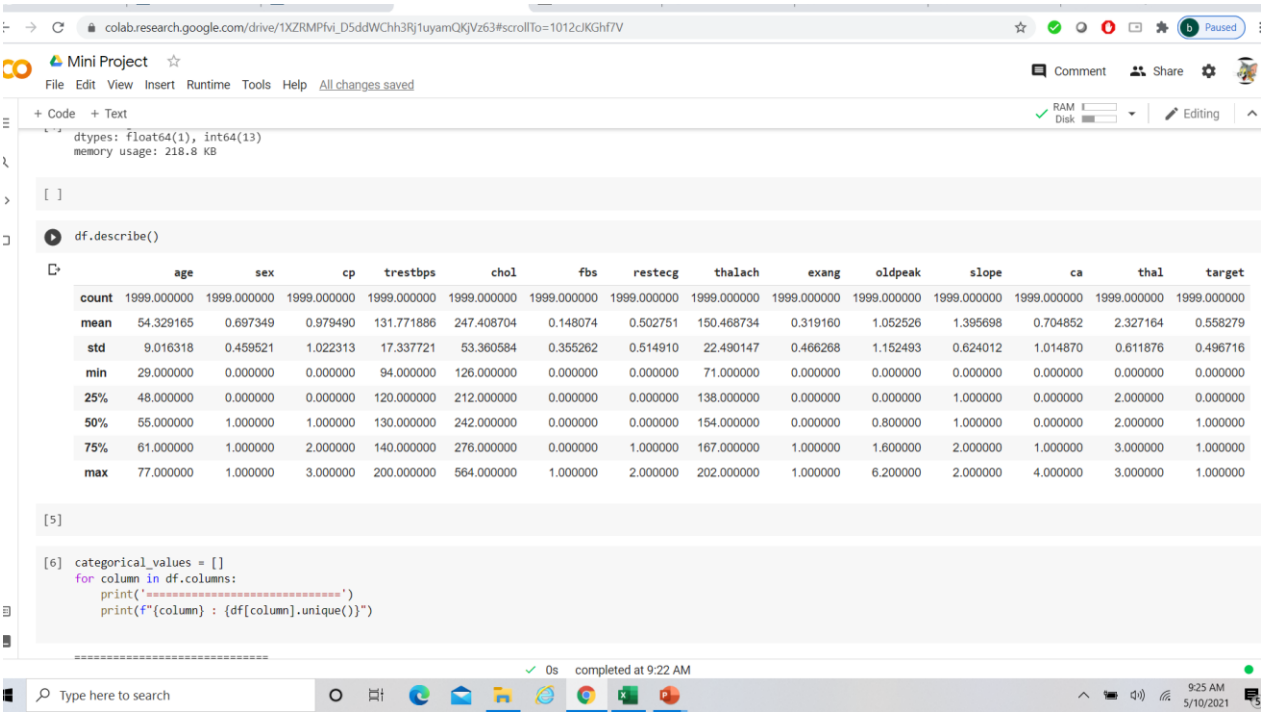
12. (**ca**) number of major vessels (0–3, *Ordinal*) colored by fluoroscopy

13. (**thal**) maximum heart rate achieved — (*Ordinal*): 3 = normal; 6 = fixed defect; 7 =reversible defect

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1999 entries, 0 to 1998
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   age       1999 non-null   int64  
 1   sex       1999 non-null   int64  
 2   cp        1999 non-null   int64  
 3   trestbps  1999 non-null   int64  
 4   chol      1999 non-null   int64  
 5   fbs       1999 non-null   int64  
 6   restecg   1999 non-null   int64  
 7   thalach   1999 non-null   int64  
 8   exang     1999 non-null   int64  
 9   oldpeak   1999 non-null   float64
 10  slope     1999 non-null   int64  
 11  ca        1999 non-null   int64  
 12  thal      1999 non-null   int64  
 13  target    1999 non-null   int64  
dtypes: float64(1), int64(13)
memory usage: 218.8 KB
```

```
[ ]
```



	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
count	1999.000000	1999.000000	1999.000000	1999.000000	1999.000000	1999.000000	1999.000000	1999.000000	1999.000000	1999.000000	1999.000000	1999.000000	1999.000000	1999.000000
mean	54.329165	0.697349	0.979490	131.771886	247.408704	0.148074	0.502751	150.468734	0.319160	1.052526	1.395698	0.704852	2.327164	0.558279
std	9.016318	0.459521	1.022313	17.337721	53.360584	0.355262	0.514910	22.490147	0.466268	1.152493	0.624012	1.014870	0.611876	0.496716
min	29.000000	0.000000	0.000000	94.000000	126.000000	0.000000	0.000000	71.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	48.000000	0.000000	0.000000	120.000000	212.000000	0.000000	0.000000	138.000000	0.000000	0.000000	1.000000	0.000000	2.000000	0.000000
50%	55.000000	1.000000	1.000000	130.000000	242.000000	0.000000	0.000000	154.000000	0.000000	0.800000	1.000000	0.000000	2.000000	1.000000
75%	61.000000	1.000000	2.000000	140.000000	276.000000	0.000000	1.000000	167.000000	1.000000	1.600000	2.000000	1.000000	3.000000	1.000000
max	77.000000	1.000000	3.000000	200.000000	564.000000	1.000000	2.000000	202.000000	1.000000	6.200000	2.000000	4.000000	3.000000	1.000000

```
categorical_values = []
for column in df.columns:
    print('==============================')
    print(f"{column} : {df[column].unique()}")
```

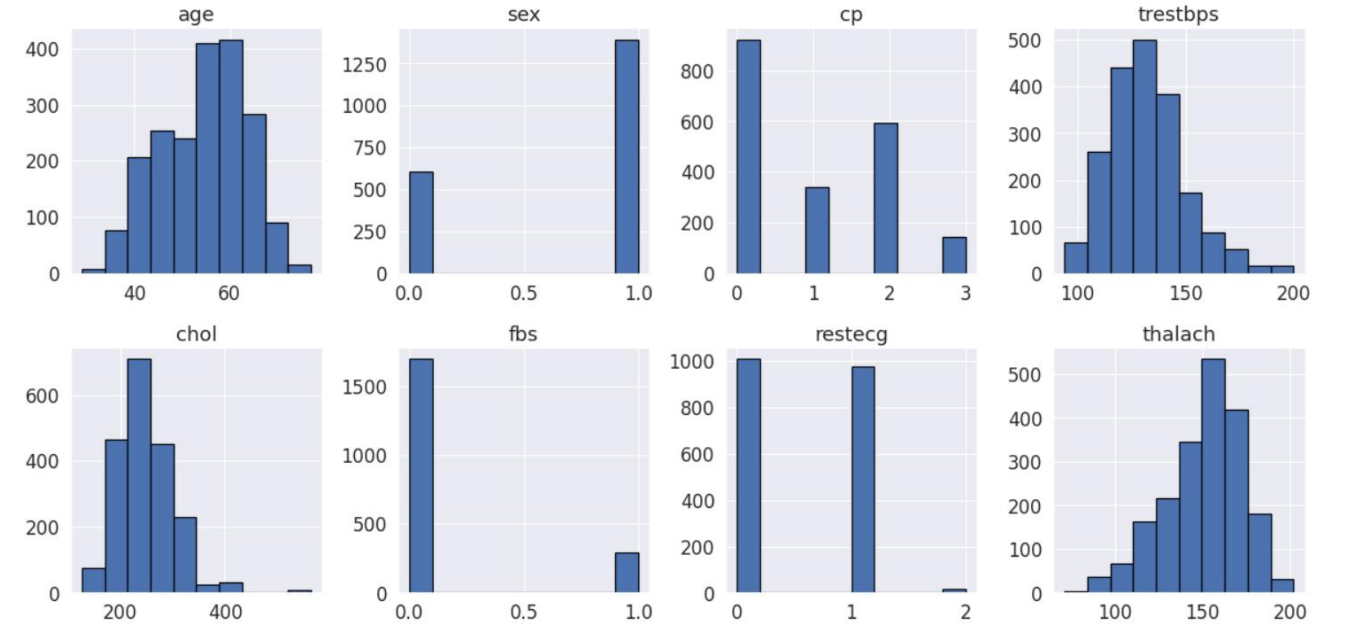
```
==============================
age : [63 37 41 56 57 44 52 54 48 49 64 58 50 66 43 69 59 42 61 40 71 51 65 53
 46 45 39 47 62 34 35 29 55 60 67 68 38 70 77 74 76]
==============================
sex : [1 0]
==============================
cp : [3 2 1 0]
==============================
trestbps : [145 130 120 140 172 150 110 135 160 105 125 142 155 104 138 128 108 134
 122 115 118 100 124  94 112 102 152 101 132 170 146 117 180 200 165 126
 174 192 164 144 148 178 129 136 106 156 123 154 114]
==============================
chol : [233 250 204 236 354 192 294 263 199 168 239 275 266 211 283 219 340 226
 247 234 243 302 212 175 417 197 198 177 273 213 304 232 269 360 308 245
 208 264 321 325 235 257 216 256 231 141 252 201 222 260 182 303 265 309
 186 203 183 220 209 258 227 261 221 205 240 318 298 564 277 214 248 255
 207 223 288 160 394 315 180 228 149 278 253 342 157 286 229 268 254 284
 224 206 167 230 335 276 353 225 330 290 172 305 188 282 185 326 267 270
 274 164 307 249 341 407 217 174 281 289 246 322 299 300 293 184 409 187
 176 241 193 131 244 195 196 126 313 262 215 271 210 295 306 178 242 259
 200 327 237 218 319 166 311 169]
==============================
fbs : [1 0]
==============================
restecg : [0 1 2]
==============================
thalach : [150 187 172 178 163 148 153 173 162 174 160 139 171 144 158 114 151 161
 179 137 157 123 152 168 140 188 125 170 165 142 180 143 182 156 115 149
 146 175 186 185 159 130 190 132 147 154 202 166 164 184 122 169 138 111
 155 131 108 129 120 112 128 109 113  99 177 141 145 136  97 127 133 126
 103 124  96  88 105 195  90 194 167 192 121 181 116 106  95 117  71 118
 134]
==============================
exang : [0 1]
==============================
oldpeak : [2.3 3.5 1.4 0.8 0.6 0.4 1.3 0.  0.5 1.6 1.2 0.2 1.8 1.  2.6 1.5 3.  2.4
 0.1 1.9 3.6 3.1 3.2 2.  2.5 2.2 2.8 3.4 6.2 4.  5.6 2.9 2.1 4.2 1.1 0.7
 0.3 0.9 3.8 4.4]
==============================
slope : [0 2 1]
==============================
ca : [0 2 1 3 4]
==============================
thal : [1 2 3 0]
==============================
target : [1 0]
```

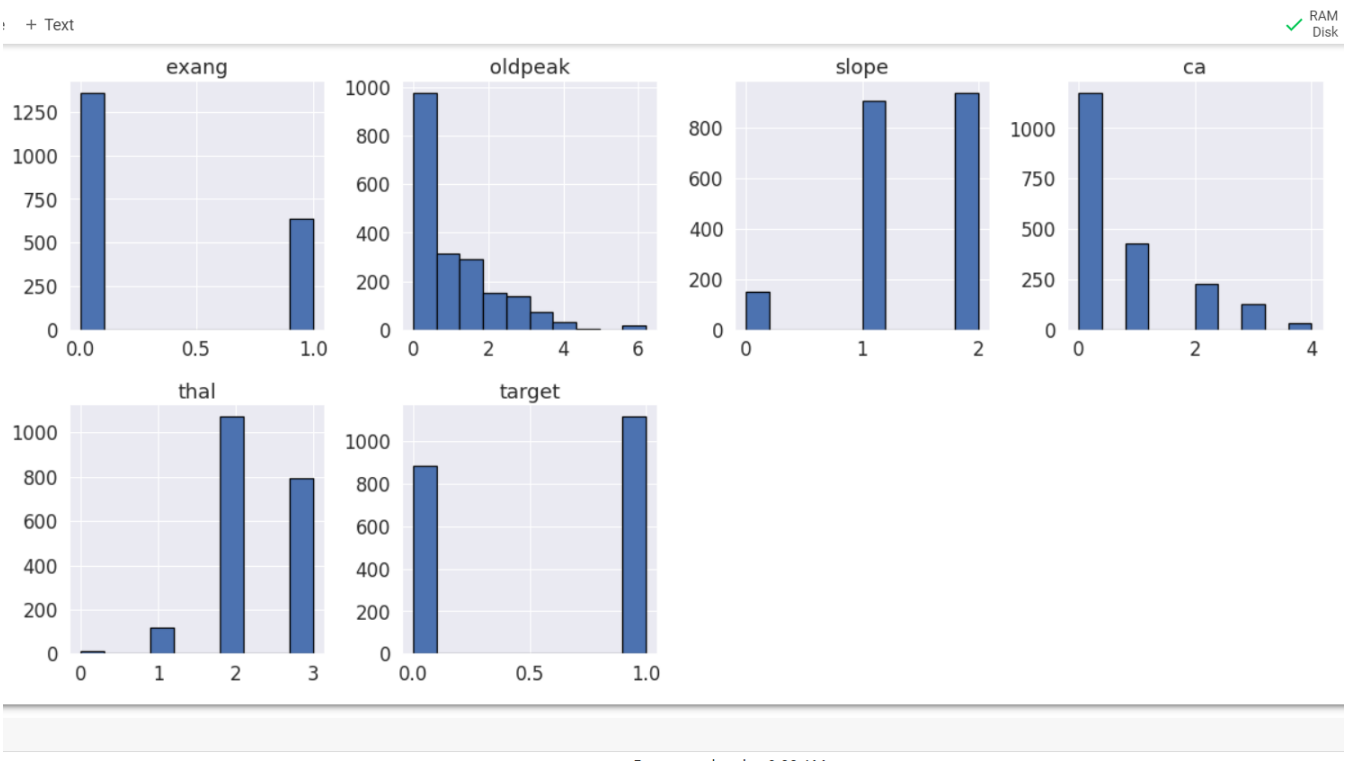
Edit View Insert Runtime Tools Help

+ Text

RAM Disk

```
sns.set(font_scale=1.5)
df.hist(edgecolor='black', linewidth=1.2, figsize=(20, 20));
```






```
from sklearn.model_selection import train_test_split

predictors = df.drop("target",axis=1)
target = df["target"]

X_train,X_test,Y_train,Y_test = train_test_split(predictors,target,test_size=0.20,random_state=0)
```

```
[30] X_train.shape
```

```
(1599, 13)
```

```
[31] X_test.shape
```

```
(400, 13)
```

```
[32] Y_train.shape
```

```
(1599,)
```

```
[33] Y_test.shape
```

```
(400,)
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
2	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
3	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
4	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
5	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
6	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
7	57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
8	56	0	1	140	294	0	0	153	0	1.3	1	0	2	1
9	44	1	1	120	263	0	1	173	0	0	2	0	3	1
10	52	1	2	172	199	1	1	162	0	0.5	2	0	3	1
11	57	1	2	150	168	0	1	174	0	1.6	2	0	2	1
12	54	1	0	140	239	0	1	160	0	1.2	2	0	2	1
13	48	0	2	130	275	0	1	139	0	0.2	2	0	2	1
14	49	1	1	130	266	0	1	171	0	0.6	2	0	2	1
15	64	1	3	110	211	0	0	144	1	1.8	1	0	2	1
16	58	0	3	150	283	1	0	162	0	1	2	0	2	1
17	50	0	2	120	219	0	1	158	0	1.6	1	0	2	1
18	58	0	2	120	340	0	1	172	0	0	2	0	2	1
19	66	0	3	150	226	0	1	114	0	2.6	0	0	2	1
20	43	1	0	150	247	0	1	171	0	1.5	2	0	2	1
21	69	0	3	140	239	0	1	151	0	1.8	2	2	2	1
22	59	1	0	135	234	0	1	161	0	0.5	1	0	3	1
23	44	1	2	130	233	0	1	179	1	0.4	2	0	2	1
24	42	1	0	140	226	0	1	178	0	0	2	0	2	1
25	61	1	2	150	243	1	1	137	1	1	1	0	2	1
26	40	1	3	140	199	0	1	178	1	1.4	2	0	3	1
27	71	0	1	160	302	0	1	162	0	0.4	2	2	2	1
28	59	1	2	150	212	1	1	157	0	1.6	2	0	2	1
29	51	1	2	110	175	0	1	123	0	0.6	2	0	2	1
30	65	0	2	140	417	1	0	157	0	0.8	2	1	2	1

Fig 3.2.1: Dataset of CVD

4. Design

4.1 Introduction:

System design is the process of designing the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system.

System Analysis is the process that decomposes a system into its component pieces for the purpose of defining how well those components interact to accomplish the set requirements. The purpose of the System Design process is to provide sufficient detailed data and information about the system and its system elements to enable the implementation consistent with architectural entities as defined in models and views of the system architecture.

The purpose of the design phase is to plan a solution of the problem specified by the requirement document. This phase is the first step in moving from problem domain to the solution domain. The design of a system is perhaps the most critical factor affecting the quality of the software, and has a major impact on the later phases, particularly testing and maintenance. The output of this phase is the design document. This document is similar to a blueprint or plan for the solution, and is used later during implementation, testing and maintenance.

The design activity is often divided into two separate phase-system design and detailed design. System design, which is sometimes also called top-level design, aims to identify the modules that should be in the system, the specifications of these modules, and how they interact with each other to produce the desired results. At the end of system design all the major data structures, file formats, output formats, as well as the major modules in the system and their specifications are decided.

A design methodology is a systematic approach to creating a design by application of set

of techniques and guidelines. Most methodologies focus on system design. The two basic principles used in any design methodology are problem partitioning and abstraction. A large system cannot be handled as a whole, and so for design it's partitioned into smaller systems. Abstraction is a concept related to problem partitioning. When partitioning is used during design, the design activity focuses on one part of the system at a time. Since the part being designed interacts with other parts of the system, a clear understanding of the interaction is essential for property designing the part.

4.2 UML Diagrams:

UML Diagrams is a rich visualizing model for representing the system architecture and design. These diagrams help us to know the flow of the system.

Some of them are;

- Use case diagram
- Sequence diagram
- State chart diagram

USECASE DIAGRAM:

A Use Case Diagram in the Unified Modelling Language (UML) is a type of behavioural diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted. Interaction among actors is not shown on the use case diagram. If this interaction is essential to a coherent description of the

desired behavior, perhaps the system or use case boundaries should be re-examined. Alternatively, interaction among actors can be part of the assumptions used in the use case.

Use cases:

A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse.

Actors:

An actor is a person, organization, or external system that plays a role in one or more interactions with the system.

System boundary boxes:

A rectangle is drawn around the use cases, called the system boundary box, to indicate the scope of system. Anything within the box represents functionality that is in scope and anything outside the box is not.

Four relationships among use cases are used often in practice.

Include:

In one form of interaction, a given use case may include another. "Include is a Directed Relationship between two use cases, implying that the behaviour of the included use case is inserted into the behaviour of the including use case.

The first use case often depends on the outcome of the included use case. This is useful for extracting truly common behaviours from multiple use cases into a single description. The

notation is a dashed arrow from the including to the included use case, with the label "«include» “. There are no parameters or return values. To specify the location in a flow of events in which the base use case includes the behaviour of another, you simply write include followed by the name of use case you want to include, as in the following flow for track order.

Extend:

In another form of interaction, a given use case (the extension) may extend another. This relationship indicates that the behaviour of the extension use case may be inserted in the extended use case under some conditions. The notation is a dashed arrow from the extension to the extended use case, with the label "«extend»". Modelers use the «extend» relationship to indicate use cases that are "optional" to the base use case.

Generalization:

In the third form of relationship among use cases, a generalization/specialization relationship exists. A given use case may have common behaviours, requirements, constraints, and assumptions with a more general use case. In this case, describe them once, and deal with it in the same way, describing any differences in the specialized cases. The notation is a solid line ending in a hollow triangle drawn from the specialized to the more general use case (following the standard generalization notation

Associations:

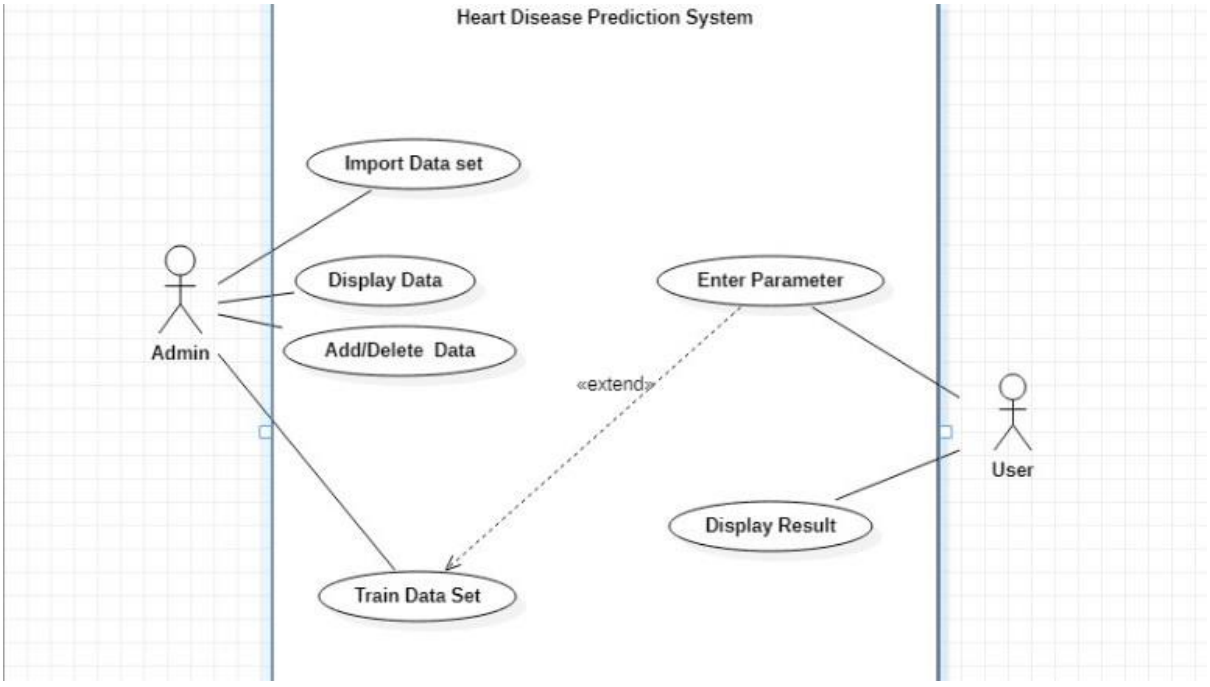
Associations between actors and use cases are indicated in use case diagrams by solid lines. An association exists whenever an actor is involved with an interaction described by a use case. Associations are modelled as lines connecting use cases and actors to one another, with an

optional arrowhead on one end of the line. The arrowhead is often used to indicating the direction of the initial invocation of the relationship or to indicate the primary actor within the use case.

Identified Use Cases

The “user model view” encompasses a problem and solution from the preservative of those individuals whose problem the solution addresses. The view presents the goals and objectives of the problem owners and their requirements of the solution. This view is composed of “use case diagrams”. These diagrams describe the functionality provided by a system to external integrators. These diagrams contain actors, use cases, and their relationships.

Use Case Diagram for Heart Disease Prediction System:



4.2.1. Use case diagram

Sequence Diagram:

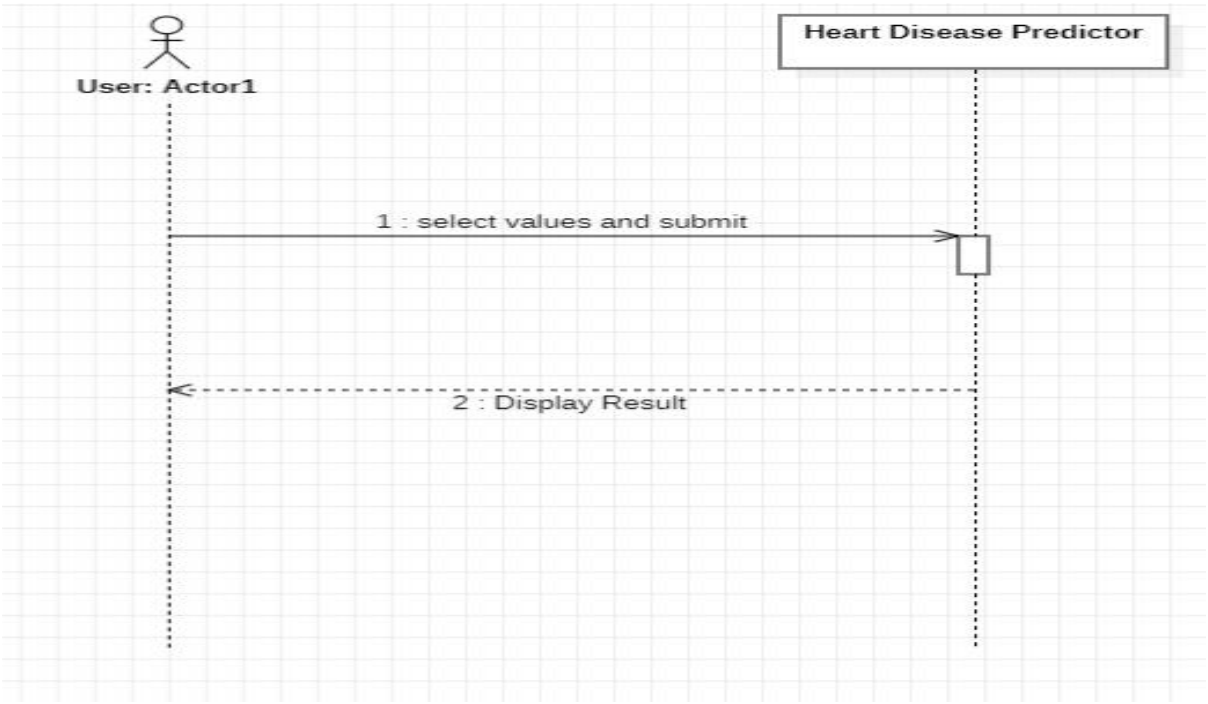
A **sequence diagram** shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called **event diagrams** or **event scenarios**.

A sequence diagram shows, as parallel vertical lines (*lifelines*), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

Purpose of Sequence Diagram:

- Model high-level interaction between active objects in a system.
- Model the interaction between object instances within a collaboration that realizes a use case.
- Model the interaction between objects within a collaboration that realizes an operation.
- Either model generic interactions (showing all possible paths through the interaction) or specific instances of a interaction (showing just one path through the interaction).

Sequence Diagram for Heart Disease Prediction System



4.2.2 Sequence diagram

STATE CHART DIAGRAM:

Statechart diagram is one of the five UML diagrams used to model the dynamic nature of a system. They define different states of an object during its lifetime and these states are changed by events. Statechart diagrams are useful to model the reactive systems. Reactive systems can be defined as a system that responds to external or internal events.

Statechart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered. The most important purpose of Statechart diagram is to model lifetime of an object from creation to termination.

Statechart diagrams are also used for forward and reverse engineering of a system. However, the main purpose is to model the reactive system.

Following are the main purposes of using Statechart diagrams −

- To model the dynamic aspect of a system.
- To model the lifetime of a reactive system.
- To describe different states of an object during its lifetime.
- Define a state machine to model the states of an object.

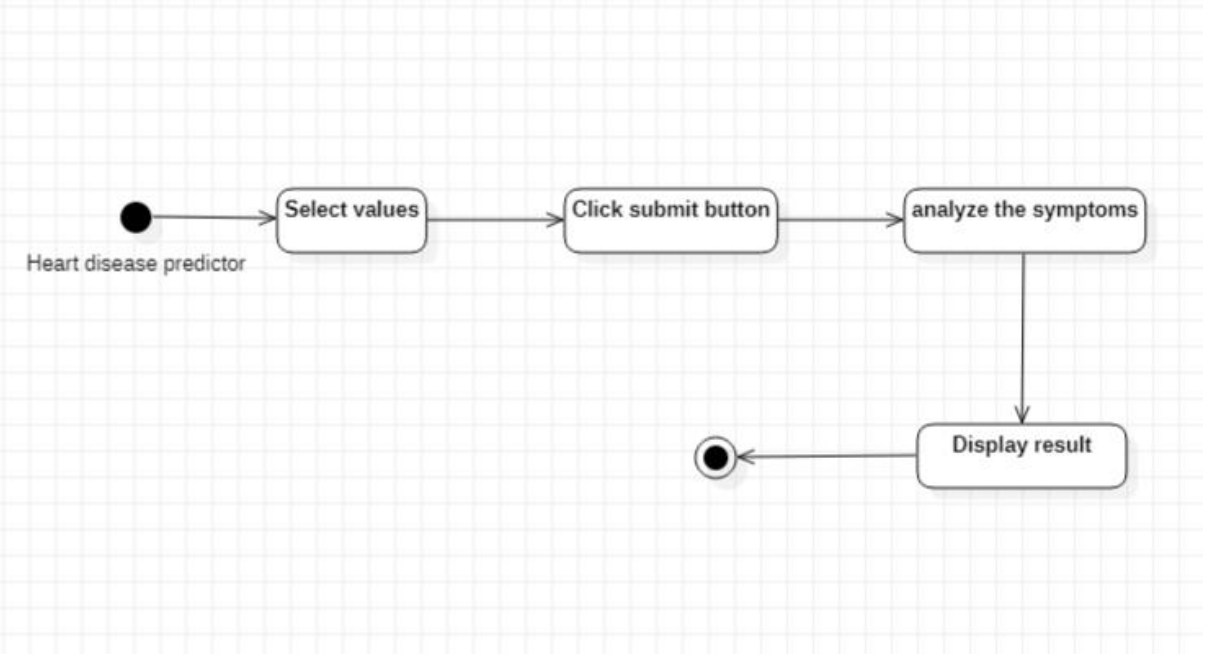
Statechart diagram is used to describe the states of different objects in its life cycle. Emphasis is placed on the state changes upon some internal or external events. These states of objects are important to analyze and implement them accurately.

Statechart diagrams are very important for describing the states. States can be identified as the condition of objects when a particular event occurs.

Before drawing a Statechart diagram we should clarify the following points −

- Identify the important objects to be analyzed.
- Identify the states.
- Identify the events.

State chart Diagram for Heart Disease Prediction System



4.2.3 State chart Diagram

5. Implementation

5.1 Introduction

5.2 Project Modules

5.3 Algorithms(optional)

5.4 Screens

5.1 Introduction:

Heart disease affects millions of people, and it remains the chief cause of death in the world. Medical diagnosis should be proficient, reliable, and aided with computer techniques to reduce the effective cost for diagnostic tests. Machine learning is a software technology that helps computers to build and classify various attributes. Here we are using classification techniques to predict heart disease. This section gives a portrayal of the machine learning and its methods with brief descriptions, data pre-processing, evaluation measurements and description of the dataset used here.

Machine learning is an emerging subdivision of artificial intelligence. Its primary focus is to design systems, allow them to learn and make predictions based on the experience. It trains machine learning algorithms using a training dataset to create a model. The model uses the new input data to predict heart disease. Using machine learning, it detects hidden patterns in the input dataset to build models. It makes accurate predictions for new datasets. The dataset is cleaned and missing values are filled. The model uses the new input data to predict heart disease and then tested for accuracy.

5.2 Project Modules:

Data Analysis

Data pre-processing

Applying ML algorithms

Deploy ML model using streamlit.

Data Analysis:

The dataset is collected from Kaggle. It consists of 14 attributes.

Our dataset consists of 3 types of data.

- Continuous
- Ordinal data
- Binary data

Attributes in dataset:

Age	thalach
Gender	exang
Chest pain	oldpeak
Trestbps	slope
Chol	ca
Fbs	thal
Restecg	target

Data pre-processing:

Data preprocessing is required tasks for cleaning the data.

And to make the data suitable for a machine learning model.

It also increases the accuracy and efficiency of a machine learning model.

Applying ML algorithms:

We have applied the following ml algorithms:

1. Logistic Regression
2. K-nearest neighbors (KNN) algorithm

Deploy ML model using Streamlit:

Streamlit is an open-source python library.
It is used to create and share data web apps.

5.3 Algorithms:

Logistic Regression Algorithm:

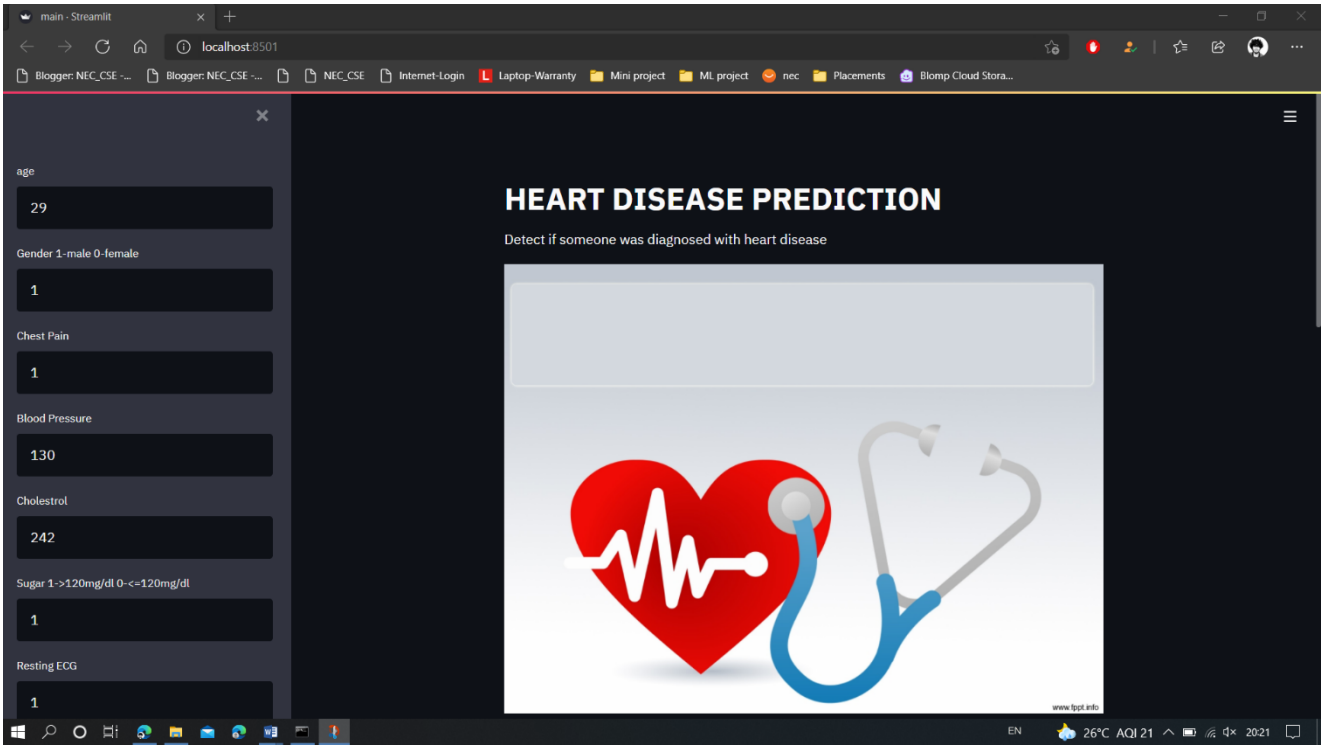
Logistic regression is basically a supervised classification algorithm. It is essentially used to predict a binary outcome based on a set of independent variables.
A **binary outcome** is one where there are only two possible scenarios - either the event happens (1) or it does not happen (0). **Independent variables** are those variables or factors which may influence the outcome.
So, Logistic regression is the correct type of analysis to use when we are working with binary data.

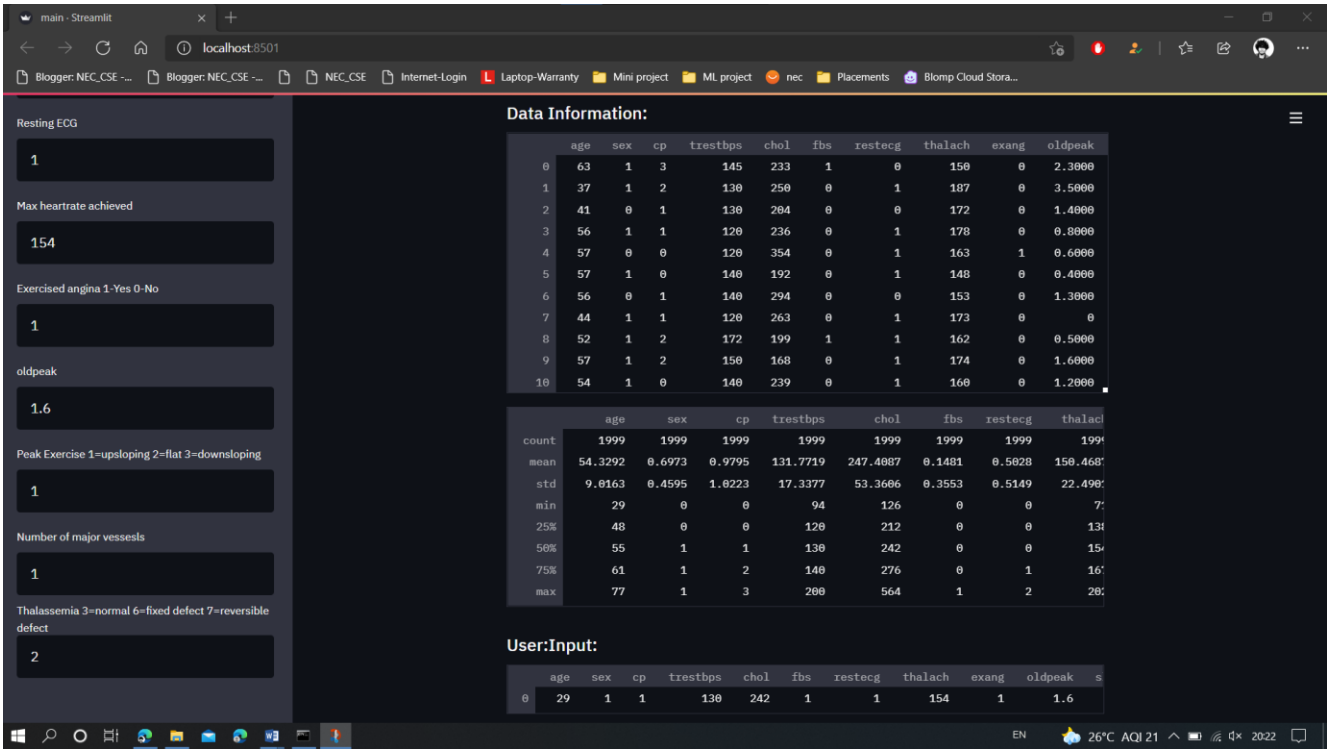
Different types of Logistic regression:
1. Binary logistic regression
2. Multinomial logistic regression
3. Ordinal logistic regression

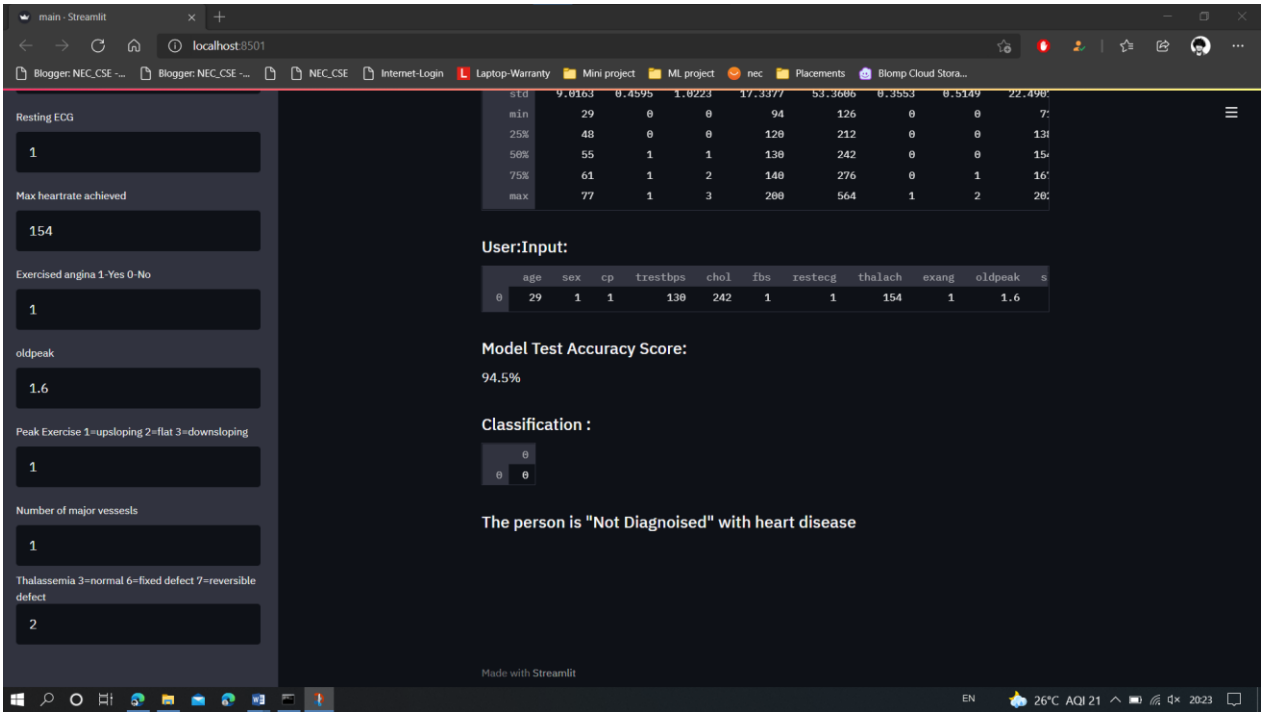
K-nearest neighbors (KNN):

K-Nearest Neighbors is one of the essential classification algorithms in Machine Learning.
The following two properties would define KNN well:
1. Lazy learning algorithm
2. Non-parametric learning algorithm
Using the k-nearest neighbor algorithm we fit the historical data and predict the future.
K-nearest neighbor or K-NN algorithm basically creates an imaginary boundary to classify the data.
When new data points come in, the algorithm will try to predict that to the nearest of the boundary line.
Therefore, larger k value means smother curves of separation resulting in less complex models. Whereas, smaller k value tends to overfit the data and resulting in complex models.

5.4 Screens:


main - Streamlit
localhost:8501
Blogger: NEC_CSE -...
Blogger: NEC_CSE -...
NEC_CSE
Internet-Login
Laptop-Warranty
Mini project
ML project
nec
Placements
Blomp Cloud Stora...
age
29
Gender 1-male 0-female
1
Chest Pain
1
Blood Pressure
130
Cholestrol
242
Sugar 1->120mg/dl 0-<=120mg/dl
1
Resting ECG
1
HEART DISEASE PREDICTION
Detect if someone was diagnosed with heart disease
www.fppt.info
EN
26°C AQI 21
20:21


main - Streamlit
localhost:8501
Blogger: NEC_CSE -...
Blogger: NEC_CSE -...
NEC_CSE
Internet-Login
Laptop-Warranty
Mini project
ML project
nec
Placements
Blomp Cloud Stora...
Resting ECG
1
Max heartrate achieved
154
Exercised angina 1-Yes 0-No
1
oldpeak
1.6
Peak Exercise 1=upsloping 2=flat 3=downsloping
1
Number of major vessels
1
Thalassemia 3=normal 6=fixed defect 7=reversible defect
2
Data Information:
age sex cp trestbps chol fbs restecg thalach exang oldpeak
0 63 1 3 145 233 1 0 150 0 2.3000
1 37 1 2 130 250 0 1 187 0 3.5000
2 41 0 1 130 204 0 0 172 0 1.4000
3 56 1 1 120 236 0 1 178 0 0.8000
4 57 0 0 120 354 0 1 163 1 0.6000
5 57 1 0 140 192 0 1 148 0 0.4000
6 56 0 1 140 294 0 0 153 0 1.3000
7 44 1 1 120 263 0 1 173 0 0
8 52 1 2 172 199 1 1 162 0 0.5000
9 57 1 2 150 168 0 1 174 0 1.6000
10 54 1 0 140 239 0 1 160 0 1.2000
age sex cp trestbps chol fbs restecg thalac
count 1999 1999 1999 1999 1999 1999 1999 199
mean 54.3292 0.6973 0.9795 131.7719 247.4087 0.1481 0.5028 150.468
std 9.0163 0.4595 1.0223 17.3377 53.3606 0.3553 0.5149 22.498
min 29 0 0 94 126 0 0 7
25% 48 0 0 120 212 0 0 13
50% 55 1 1 130 242 0 0 15
75% 61 1 2 140 276 0 1 16
max 77 1 3 200 564 1 2 20
User:Input:
age sex cp trestbps chol fbs restecg thalach exang oldpeak s
0 29 1 1 130 242 1 1 154 1 1.6
EN
26°C AQI 21
20:22


main - Streamlit
localhost:8501
Blogger: NEC_CSE -...
Blogger: NEC_CSE -...
NEC_CSE
Internet-Login
Laptop-Warranty
Mini project
ML project
nec
Placements
Blomp Cloud Stora...
Resting ECG
1
Max heartrate achieved
154
Exercised angina 1-Yes 0-No
1
oldpeak
1.6
Peak Exercise 1=upsloping 2=flat 3=downsloping
1
Number of major vessesls
1
Thalassemia 3=normal 6=fixed defect 7=reversible defect
2
std 9.0163 0.4595 1.0223 17.3377 53.3606 0.3553 0.5149 22.490
min 29 0 0 94 126 0 0 7
25% 48 0 0 120 212 0 0 13
50% 55 1 1 130 242 0 0 15
75% 61 1 2 140 276 0 1 16
max 77 1 3 200 564 1 2 20
User:Input:
age sex cp trestbps chol fbs restecg thalach exang oldpeak s
0 29 1 1 130 242 1 1 154 1 1.6
Model Test Accuracy Score:
94.5%
Classification :
0
0 0
The person is "Not Diagnoised" with heart disease
Made with Streamlit
EN
26°C AQI 21
20:23

6. System Testing

6.1 Introduction:

Software Testing is an important element of the software quality assurance and represents the ultimate review of specification, design and coding. The increasing feasibility of software as a system and the cost associated with the software failures are motivated forces for III planned through testing.

TESTING OBJECTIVES:

These are several rules that can save as testing objectives:

- Testing is a process of executing program with the intent of finding an error.
- A good testcase is one that has a high probability of finding an undiscovered error.

Test Levels:

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or darkness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.2 TESTING METHODS:

6.2.1 Unit Testing

Unit testing involves the design of test cases that validate that the internal program logic is

functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application.

6.2.2 Integration Testing

Integration tests are designed to test integrated software components to determine if they run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields.

6.2.3 Functional Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases.

6.2.4 System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test.

6.2.5 White Box Test

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

6.2.6 Black Box Test

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document.

6.2.7 Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

6.2.8 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

6.2.9 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user.

7 CONCLUSION

The early prognosis of cardiovascular diseases can aid in making decisions on lifestyle changes in high-risk patients and in turn reduce the complications, which can be a great milestone in the field of medicine. This project resolved the existing system drawbacks and successfully predict the heart disease, with 91% accuracy. The models used are Logistic Regression and K-nearest neighbors (KNN) algorithm. In these both we got more accuracy with K-nearest neighbors (KNN) algorithm compared to logistic regression. Further for its enhancement, we can train on models and predict the types of cardiovascular diseases providing recommendations to the users, and also use more enhanced models.

8 FUTURE SCOPE

This project further can be developed as Android application to overcome the limitation of accessing the system by only desktop and also suggest them about their stage in chronic kidney diseases. And also suggest the required treatment that to be taken by the person to cure from the chronic kidney disease at that particular stage.

9 REFERENCES

[1] Alotaibi, F. S. (2019). Implementation of machine learning model to predict heart failure disease. *International Journal of Advanced Computer Science and Applications, 10* (6), 261-268.

[2] Amin, M. S., Chiam, Y. K., & Varathan, K. D. (2018). Identification of significant features and data mining techniques in predicting heart disease. *Telematics and Informatics*. doi: 10.1016/J.TELE.2018.11.007.

[3] Anitha, S., & Sridevi, N. (2019). Heart disease prediction using data mining techniques. *Journal of Analysis and Computation, 8* (2), 48-55.

[4] Annepu, D., & Gowtham, G. (2019). Cardiovascular disease prediction using machine learning techniques. *International Research Journal of Engineering and Technology, 6* (4), 3963-3971.

[5] Ayatollahi, H., Gholamhosseini, L., & Salehi, M. (2019). Predicting coronary artery disease: a comparison between two data mining algorithms. *BMC Public Health*. doi: 10.1186/S12889-019-6721-5.

[6] Banu, G. R., & Jamala, J. H. (2015). Heart attack prediction using data mining technique. *International Journal of Modern Trends in Engineering and Research, 2* (5), 428-432.

[7] Benjamin, H., David, F., & Belcy, S. A. (2018). Heart disease prediction using data mining techniques. *ICTACT Journal of Soft Computing, 9* (1), 1824-1830.

[8] Chaithra, N., & Madhu, B. (2018). Classification models on cardiovascular disease prediction using data mining techniques. *Journal of Cardiovascular Diseases and Diagnosis*. doi: 10.4172/2329-9517.1000348.

[9] D'Souza, A. (2015). Heart disease prediction using data mining techniques. *International Journal of Research in Engineering and Science, 3* (3), 74-77.

[10] Devi, S. K. (2016). Prediction of heart disease using data mining techniques. *Indian Journal of Science and Technology*. doi: 10.17485/ijst/2016/v9i39/102078.