**PROFESSIONAL TRAINING REPORT**

**at**

**Sathyabama Institute of Science and Technology**

**(Deemed to be University)**

Submitted in partial fulfillment of the requirements for the award of

Bachelor of Engineering Degree in Computer Science and Engineering

By

**BODEPUDI LAKSHMI PRIYA**

**(Reg.No.39110174)**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

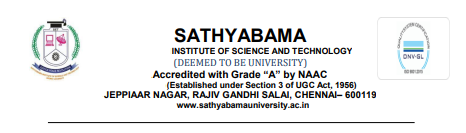
**SCHOOL OF COMPUTING**

**SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,**

**CHENNAI - 600119, TAMILNADU**

**NOVEMBER 2021**

****

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**BONAFIDE CERTIFICATE**

This is to certify that this Project Report is the Bonafide work of **BODEPUDI LAKSHMI PRIYA (Reg. No. 39110174)** who carried out the project entitled “**HEART DISEASE PREDICTION USING MACHINE LEARNING**” under my supervision from September 2021 to November 2021.

**Internal Guide**

**Mrs. KARTHIKA J**

**Head of the Department**

**Dr. L. Lakshmanan, M.E., PH.D.,**

**Submitted for Viva voce Examination held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Internal Examiner External Examiner**

**DECLARATION**

I, **BODEPUDI LAKSHMI PRIYA(Reg. No 39110174)** hereby declare that the Professional Training Report on “**HEART DISEASE PREDICTION USING MACHINE LEARNING**” done by me under the guidance **Mrs. KARTHIKA J** at **Sathyabama institute of science and technology** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

**DATE:**

**PLACE: SIGNATURE OF THE CANDIDATE:**

**ACKNOWLEDGEMENT**

I am pleased to acknowledge my sincere thanks to Board of Management

**SATHYABAMA** for their kind encouragement in doing this project and for

completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. SASIKALA., ME., Ph.D**., Dean, School of Computing **Dr. L.LAKSHMANAN,ME., Ph.D**., Head of the Department, Dept. of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project

Guide **Mrs. KARTHIKA J** for her valuable guidance , suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of

the Department of Computer Science and Engineering who were helpful in

many ways for the completion of the project.

**TRAINING CERTIFICATE**

**ABSTRACT**

The health care industries collect huge amounts of data that contain some hidden information, which is useful for making effective decisions. For providing appropriate results and making effective decisions on data, some advanced data mining techniques are used. In this study, a Heart Disease Prediction System (HDPS) is developed using Logestic Regression algorithm for predicting the risk level of heart disease. The system uses 15 medical parameters such as age, sex, blood pressure, cholesterol, and obesity for prediction. The HDPS predicts the likelihood of patients getting heart disease. It enables significant knowledge. E.g. Relationships between medical factors related to heart disease and patterns, to be established. We have employed the multilayer perceptron neural network with backpropagation as the training algorithm. The obtained results have illustrated that the designed diagnostic system can effectively predict the risk level of heart diseases.

**TABLE OF CONTENTS**

**CHAPTER NO. TITLE PAGE NO.**

**ABSTRACT 5**

**LIST OF FIGURES 8**

**1 INTRODUCTION 9**

* 1. PROBLEM STATEMENT 9

1.2 MOTIVATION 9

1.3 OBJECTIVE 10

1.3.1 PROPOSED SYSTEM 10

1. **TECHNOLOGIES LEARNT 11**
   1. PYTHON 11

2.1.1 ADVANTAGES OF PYTHON 11

2.1.2 DISADVANTAGES OF PYTHON 12

* 1. MACHINE LEARNING 12

2.2.1 CHALLENGES IN MACHINE 13

LEARNING

2.2.2 TYPES OF MACHINE LEARNING 14

2.2.3 ADVANTAGES OF MACHINE 15

LEARNING

2.2.4 DISADVANTAGES OF MACHINE 16

LEARNING

2.2.5 APPLICATIONS OF MACHINE 17

LEARNING

* 1. MODULES USED IN PROJECT 17

2.4 INSTALL PYTHON STEP-BY-STEP 19

IN WINDOWS

1. **EXPERIMENTAL OR MATERIALS 21**

**AND METHODS**

* 1. ALGORITHMS USED IN PROJECT 21

3.1.1 LOGISTIC REGRESSION 21

ALGORITHM

* 1. SYSTEM SPECIFICATION 24

3.2.1 SOFTWARE REQUIREMENTS 24

3.2.2 HARDWARE REQUIREMENTS 24

**4 DESIGN 25**

4.1 PYTHON IMPLEMENTATION OF

LOGISTIC REGRESSION (BINOMIAL) 25

4.1.1 DATA PRE-PROCESSING STEP 25

4.1.2 FITTING LOGISTIC REGRESSION

TO THE TRAINING SET 27

4.1.3 PREDICTING THE TEST RESULT 27

4.1.4 TEST ACCURACY OF THE RESULT 27

* + 1. VISUALIZING THE TRAINING SET RESULT 28

4.1.6 VISUALIZING THE TEST SET RESULT 28

4.1.7 FLOW CHART DIAGRAM 29

**5 SUMMARY AND CONCLUSION 30**

**REFERENCES 31**

**APPENDIX 32**

A. SCREENSHOTS 32

B. SOURCE CODE 33

**LIST OF FIGURES**

**FIGURE NO. FIGURE NAME PAGE NO.**

3.1 LOGISTIC FUNCTION 22

4.1 TRAINING SET 28

4.2 TEST SET 29

4.3 FLOWCHART 30

5.1 IMPORT DATA SET 32

5.2 DATA SET INFORMATION 32

5.3 ACCURACY 32

**CHAPTER 1**

**INTRODUCTION**

* 1. **PROBLEM STATEMENT**

Heart disease can be managed effectively with a combination of lifestyle changes, medicine and, in some cases, surgery. With the right treatment, the symptoms of heart disease can be reduced and the functioning of the heart improved. The predicted results can be used to prevent and thus reduce cost for surgical treatment and other expensive. The overall objective of my work will be to predict accurately with few tests and attributes the presence of heart disease. Attributes considered form the primary basis for tests and give accurate results more or less. Many more input attributes can be taken but our goal is to predict with few attributes and faster efficiency the risk of having heart disease. Decisions are often made based on doctors’ intuition and experience rather than on the knowledge rich data hidden in the data set and databases. This practice leads to unwanted biases, errors and excessive medical costs which affects the quality of service provided to patients.

* 1. **MOTIVATION**

The main motivation of doing this research is to present a heart disease

prediction model for the prediction of occurrence of heart disease. Further, this

research work is aimed towards identifying the best classification algorithm for

identifying the possibility of heart disease in a patient. This work is justified by

performing a comparative study and analysis using classification algorithm

named Logestic Regression isused at different levels of

evaluations. Although it is commonly used machine learning algorithm, the

heart disease prediction is a vital task involving highest possible accuracy. Hence, the

the algorithm is evaluated at numerous levels and types of evaluation strategies.

This will provide researchers and medical practitioners to establish a better.

* 1. **OBJECTIVE**

The main objective of this research is to develop a heart prediction system. The system can discover and extract hidden knowledge associated with diseases from a historical heart data set Heart disease prediction system aims to exploit data mining techniques on medical data set to assist in the prediction of the heart diseases.

* + 1. **PROPOSED SYSTEM**

The working of the system starts with the collection of data and selecting the

important attributes. Then the required data is preprocessed into the required format.

The data is then divided into two parts training and testing data. The algorithms are

applied and the model is trained using the training data. The accuracy of the system is

obtained by testing the system using the testing data. This system is implemented

using the following modules.

1.) Collection of Dataset

2.) Selection of attributes

3.) Data Pre-Processing

4.) Balancing of Data

5.) Disease Prediction

**CHAPTER 2**

**TECHNOLOGIES LEARNT**

**2.1 PYTHON**

Python is currently the most widely used multi-purpose, high-level programming

language. Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java. Programmers have to type relatively less and indentation

requirement of the language, makes them readable all the time. Python language is

being used by almost all tech-giant companies like Google, Amazon, Facebook,

Instagram, Dropbox, Uber… etc. The biggest strength of Python is huge collection

of standard library which can be used for the following –

* Machine Learning
* GUI Applications (like Kivy, Tkinter, PyQt etc. )
* Web frameworks like Django (used by YouTube, Instagram, Dropbox)
* Image processing (like OpenCV, Pillow)
* Web scraping (like Scrapy, BeautifulSoup, Selenium)
* Test frameworks
* Multimedia

**2.1.1 ADVANTAGES OF PYTHON**

* Extensive Libraries
* Extensible
* Embeddable
* Improved Productivity
* IOT Opportunities
* Readable
* Object-Oriented
* Free and Open Source
* Portable
* Interpreted

**2.1.2 DISADVANTAGES OF PYTHON**

So far, we’ve seen why Python is a great choice for your project. But if you choose

it, you should be aware of its consequences as well. Let’s now see the downsides

of choosing Python over another language.

* Speed Limitations
* Weak in Mobile Computing and Browsers
* Design Restrictions
* Underdeveloped Database Access layers
* Simple

**2.2 MACHINE LEARNING**

Before we take a look at the details of various machine learning methods, let's start

by looking at what machine learning is, and what it isn't. Machine learning is often

categorized as a subfield of artificial intelligence, but I find that categorization can

often be misleading at first brush. The study of machine learning certainly arose

from research in this context, but in the data science application of machine

learning methods, it's more helpful to think of machine learning as a means

of building models of data.

Fundamentally, machine learning involves building mathematical models to help

understand data. "Learning" enters the fray when we give these models tunable

parameters that can be adapted to observed data; in this way the program can be

considered to be "learning" from the data. Once these models have been fit to

previously seen data, they can be used to predict and understand aspects of newly

observed data. I'll leave to the reader the more philosophical digression regarding

the extent to which this type of mathematical, model-based "learning" is similar to

the "learning" exhibited by the human brain.

**2.2.1 CHALLENGES IN MACHINE LEARNING**

While Machine Learning is rapidly evolving, making significant strides with

cybersecurity and autonomous cars, this segment of AI as whole still has a long

way to go. The reason behind is that ML has not been able to overcome number of

challenges. The challenges that ML is facing currently are –

**Quality of data**

Having good-quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data preprocessing and feature extraction.

**Time-Consuming task**

Another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.

**Lack of specialist persons**

As ML technology is still in its infancy stage,availability of expert resources is a tough job.

**No clear objective for formulating business problems**

Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.

**Issue of overfitting & underfitting**

If the model is overfitting or underfitting, it cannot be represented well for the problem.

**Curse of dimensionality**

Another challenge ML model faces is too many features of data points. This can be a real hindrance.

**Difficulty in deployment**

Complexity of the ML model makes it quite difficult to be deployed in real life.

**2.2.2 TYPES OF MACHINE LEARNING**

**Supervised Learning**

This involves learning from a training dataset with labeled data using classification and regression models. This learning process continues until the required level of performance is achieved.

**Unsupervised Learning**

This involves using unlabelled data and then finding the

underlying structure in the data in order to learn more and more about the data itself

using factor and cluster analysis models.

**Semi-supervised Learning**

This involves using unlabelled data like Unsupervised Learning with a small amount of labeled data. Using labeled data vastly increases the learning accuracy and is also more cost-effective than Supervised Learning.

**Reinforcement Learning**

This involves learning optimal actions through trial and error. So the next action is decided by learning behaviors that are based on the current state and that will maximize the reward in the future.

**2.2.3 ADVANTAGES OF MACHINE LEARNING**

**Easily identifies trends and patterns**

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

**No human intervention needed (automation)**

With ML, you don’t need to babysit your project every step of the way. Since it

means giving machines the ability to learn, it lets them make predictions and also

improve the algorithms on their own. A common example of this is anti-virus softwares, they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

**Continuous Improvement**

As ML algorithms gain experience, they keep improving in accuracy and efficiency.

This lets them make better decisions. Say you need to make a weather forecast

model. As the amount of data you have keeps growing, your algorithms learn to

make more accurate predictions faster.

**Handling multi-dimensional and multi-variety data**

Machine Learning algorithms are good at handling data that are multi-dimensional

and multi-variety, and they can do this in dynamic or uncertain environments.

**Wide Applications**

You could be an e tailer or a healthcare provider and make ML work for you. Where

it does apply, it holds the capability to help deliver a much more personal experience

to customers while also targeting the right customers.

**2.2.4 DISADVANTAGES OF MACHINE LEARNING**

**Data Acquisition**

Machine Learning requires massive data sets to train on, and these should be

inclusive/unbiased, and of good quality. There can also be times where they must

wait for new data to be generated.

**Time and Resources**

ML needs enough time to let the algorithms learn and develop enough to fulfill their

purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer

power for you.

**Interpretation of Results**

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

**High error-susceptibility**

Machine Learning is autonomous but highly susceptible to errors. Suppose you train

an algorithm with data sets small enough to not be inclusive. You end up with biased

predictions coming from a biased training set. This leads to irrelevant advertisements being displayed to customers. In the case of ML, such blunders can

set off a chain of errors that can go undetected for long periods of time. And when

they do get noticed, it takes quite some time to recognize the source of the issue,

and even longer to correct it.

**2.2.5 APPLICATIONS OF MACHINE LEARNING**

Machine Learning is the most rapidly growing technology and according to

researchers we are in the golden year of AI and ML. It is used to solve many real-world complex problems which cannot be solved with traditional approach. Following are some real-world applications of ML.

* Emotion analysis
* Sentiment analysis
* Error detection and prevention
* Weather forecasting and prediction
* Stock market analysis and forecasting
* Speech synthesis
* Speech recognition
* Customer segmentation
* Object recognition
* Fraud detection
* Fraud prevention
* Recommendation of products to customer in online shopping

**2.3 MODULES USED IN PROJECT**

**NUMPY**

Numpy is a general-purpose array-processing package. It provides a high performance multidimensional array object, and tools for working with these arrays.It

is the fundamental package for scientific computing with Python. It contains various

features including these important ones:

* A powerful N-dimensional array object
* Sophisticated (broadcasting) functions
* Tools for integrating C/C++ and Fortran code
* Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using Numpy which allows Numpy to seamlessly and speedily integrate with a wide variety

of databases.

**PANDAS**

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance,

economics, Statistics, analytics, etc.

**MATPLOTLIB**

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms.

Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail

gallery.For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.

**SCIKIT – LEARN**

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD

license and is distributed under many Linux distributions, encouraging academic and commercial use.

**2.4 INSTALL PYTHON STEP-BY-STEP IN WINDOWS**

There have been several updates in the Python version over the years. The question is how to install Python? It might be confusing for the beginner who is willing to start

learning Python but this tutorial will solve your query. The latest or the newest version of Python is version 3.7.4 or in other words, it is Python 3. Before you start with the installation process of Python. First, you need to know about your system Requirements. Based on your system type i.e. operating system and based processor, you must download the python version. My system type is a Windows 64-bit operating system. So the steps below are to install python version 3.7.4 on Windows 7 device or to install Python 3. Download the Python Cheat sheet here.The

steps on how to install Python on Windows 10, 8 and 7 are divided into 4 parts to help understand better.Download the Correct version into the system.

**Step 1**: Go to the official site to download and install python using Google Chrome or any other web browser. OR Click on the following link**:** [**https://www.python.org**](https://www.python.org)

Now, check for the latest and the correct version for your operating system.

**Step 2**: Click on the Download Tab.

**Step 3**: You can either select the Download Python for windows 3.7.4 button in Yellow Color or you can scroll further down and click on download with respective to their version. Here, we are downloading the most recent python version for windows 3.7.4

**Step 4**: Scroll down the page until you find the Files option.

**Step 5**: Here you see a different version of python along with the operating system.

Here we will install Windows x86-64 web-based installer. Here your first part regarding.

Here we will install Windows x86-64 web-based installer. Here your first part regarding which version of python is to be downloaded is completed. Now we move ahead with the second part in installing python i.e. Installation.

**Note**: To know the changes or updates that are made in the version you can click on the Release Note Option.

**Installation of Python**

**Step 1:** Go to Download and Open the downloaded python version to carry out the

installation process.

**Step 2:** Before you click on Install Now, make sure to put a tick on Add Python 3.7 to path.

**Step 3:** Click on Install now, after the installation is successful. Click on Close.

With these above three steps on python installation, you have successfully and correctly installed Python. Now is the time to verify the installation.

**Note:** The installation process might take a couple of minutes.

**CHAPTER 3**

**EXPERIMENTAL OR MATERIALS AND METHODS**

**3.1 ALGORITHMS USED IN THIS PROJECT**

**3.1.1 LOGISTIC REGRESSION ALGORITHM**

Logistic regression is basically a supervised classification algorithm. In a classification problem, the target variable(or output), y, can take only discrete values for a given set of features(or inputs), X.

Contrary to popular belief, logistic regression IS a regression model.

The model builds a regression model to predict the probability that a given data entry belongs to the category numbered as “1”. Just like Linear regression assumes that the data follows a linear function, Logistic regression models the data using the sigmoid function.

**Working of Logistic Regression Algorithm**

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.

In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).

The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.

Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification. The below image is showing the logistic function:

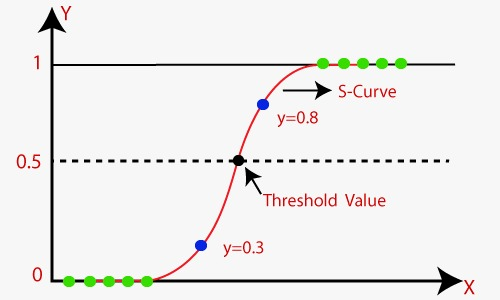


Fig 3.1 Logistic Function

**Logistic Function (Sigmoid Function):**

The sigmoid function is a mathematical function used to map the predicted values to probabilities.

It maps any real value into another value within a range of 0 and 1.

The value of the logistic regression must be between 0 and 1, which cannot go beyond this limit, so it forms a curve like the "S" form. The S-form curve is called the Sigmoid function or the logistic function.

In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.

**Assumptions for Logistic Regression:**

The dependent variable must be categorical in nature.

The independent variable should not have multi-collinearity.

**Logistic Regression Equation:**

The Logistic regression equation can be obtained from the Linear Regression equation. The mathematical steps to get Logistic Regression equations are given below:

* We know the equation of the straight line can be written as:
* In Logistic Regression y can be between 0 and 1 only, so for this let's divide the above equation by (1-y):



* But we need range between -[infinity] to +[infinity], then take logarithm of the equation it will become:



The above equation is the final equation for Logistic Regression.

**Type of Logistic Regression:**

On the basis of the categories, Logistic Regression can be classified into three types:

* **Binomial**: In binomial Logistic regression, there can be only two possible types of the dependent variables, such as 0 or 1, Pass or Fail, etc.
* **Multinomial**: In multinomial Logistic regression, there can be 3 or more possible unordered types of the dependent variable, such as "cat", "dogs", or "sheep"
* **Ordinal**: In ordinal Logistic regression, there can be 3 or more possible ordered types of dependent variables, such as "low", "Medium", or "High".

**3.2 SYSTEM SPECIFICATION**

**3.2.1 SOFTWARE REQUIREMENTS**

Functional requirements for a secure cloud storage service are straight forward:

* The service should be able to store the user’s data;
* The data should be accessible through any devices connected to the Internet;
* The service should be capable to synchronize the user’s data between multiple
* devices (notebooks, smart phones, etc.);
* The service should preserve all historical changes (versioning);
* Data should be shareable with other users;
* The service should support SSO; and
* The service should be interoperable with other cloud storage services, enabling
* data migration from one CSP to another.
* **Operating System:** Windows
* **Coding Language:** Python 3.7

**3.2.2 HARDWARE REQUIREMENTS**

* **Processor** – Intel core i5
* **Speed –** 1.19 GHz
* **RAM -** 512 MB (min)
* **Hard Disk** - 20 GB
* **Key Board -** Standard Keyboard
* **Monitor –** 15 VGA Colo

**CHAPTER 4**

**DESIGN**

**4.1** **PYTHON IMPLEMENTATION OF LOGISTIC REGRESSION (BINOMIAL)**

We will build a Machine Learning model using the Logistic regression algorithm and the dataset is uploaded.

**Steps in Logistic Regression**: To implement the Logistic Regression using Python, we will use the same steps as we have done in previous topics of Regression. Below are the steps:

* Data Pre-processing step
* Fitting Logistic Regression to the Training set
* Predicting the test result
* Test accuracy of the result(Creation of Confusion matrix)
* Visualizing the test set result.

**4.1.1 DATA PRE-PROCESSING STEP**

In this step, we will pre-process/prepare the data so that we can use it in our code efficiently. It will be the same as we have done in Data pre-processing topic. The code for this is given below:

#Data Pre-procesing Step

# importing libraries

import numpy as nm

import matplotlib.pyplot as mtp

import pandas as pd

#importing datasets

data\_set= pd.read\_csv('user\_data.csv')

* Now, we will extract the dependent and independent variables from the given dataset. Below is the code for it:

#Extracting Independent and dependent Variable

x= data\_set.iloc[:, [2,3]].values

y= data\_set.iloc[:, 4].values

* Now we will split the dataset into a training set and test set. Below is the code for it:

# Splitting the dataset into training and test set.

from sklearn.model\_selection import train\_test\_split

x\_train, x\_test, y\_train, y\_test= train\_test\_split(x, y, test\_size= 0.25, random\_state=0)

* In logistic regression, we will do feature scaling because we want accurate result of predictions. Here we will only scale the independent variable because dependent variable have only 0 and 1 values. Below is the code for it:

#feature Scaling

from sklearn.preprocessing import StandardScaler

st\_x= StandardScaler()

x\_train= st\_x.fit\_transform(x\_train)

x\_test= st\_x.transform(x\_test)

**4.1.2 FITTING LOGISTIC REGRESSION TO THE TRAINING SET**

We have well prepared our dataset, and now we will train the dataset using the training set. For providing training or fitting the model to the training set, we will import the LogisticRegression class of the sklearn library.

After importing the class, we will create a classifier object and use it to fit the model to the logistic regression. Below is the code for it:

#Fitting Logistic Regression to the training set

from sklearn.linear\_model import LogisticRegression

classifier= LogisticRegression(random\_state=0)

classifier.fit(x\_train, y\_train)

**4.1.3 PREDICTING THE TEST RESULT**

Our model is well trained on the training set, so we will now predict the result by using test set data. Below is the code for it:

#Predicting the test set result

y\_pred= classifier.predict(x\_test)

**4.1.4 TEST ACCURACY OF THE RESULT**

Now we will create the confusion matrix here to check the accuracy of the classification. To create it, we need to import the confusion\_matrix function of the sklearn library. After importing the function, we will call it using a new variable cm. The function takes two parameters, mainly y\_true( the actual values) and y\_pred (the targeted value return by the classifier). Below is the code for it:

#Creating the Confusion matrix

from sklearn.metrics import confusion\_matrix

cm= confusion\_matrix()

**4.1.5 VISUALIZING THE TRAINING SET RESULT**

Finally, we will visualize the training set result. To visualize the result, we will use ListedColormap class of matplotlib library.

We have imported the ListedColormap class of Matplotlib library to create the colormap for visualizing the result.

To create a filled contour, we have used mtp.contourf command, it will create regions of provided colors (purple and green). In this function, we have passed the classifier.predict to show the predicted data points predicted by the classifier.



Fig 4.1 Logistic Regression training set

**4.1.6 VISUALIZING THE TEST SET RESULT**

Our model is well trained using the training dataset. Now, we will visualize the result for new observations (Test set). The code for the test set will remain same as above except that here we will use x\_test and y\_test instead of x\_train and y\_train.

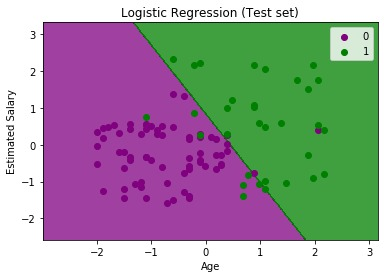


Fig 4.2 Logistic Regression test set

The above graph shows the test set result. As we can see, the graph is divided into two regions (Purple and Green). And Green observations are in the green region, and Purple observations are in the purple region. So we can say it is a good prediction and model.

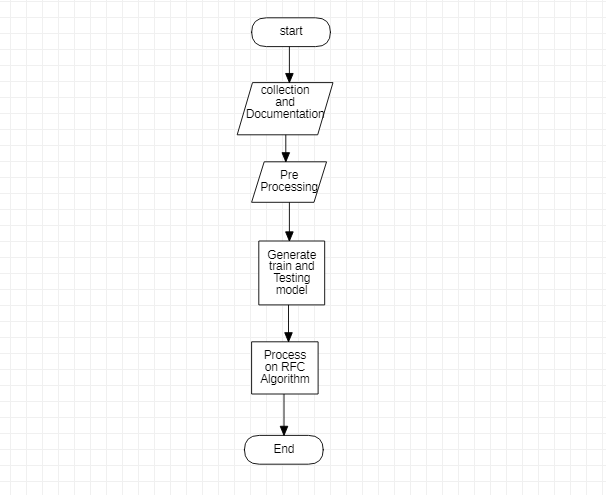
**4.1.7 FLOW CHART DIAGRAM**

A flowchart is simply a graphical representation of steps. It shows steps in

sequential order and is widely used in presenting the flow of algorithms, workflow

or processes. Typically, a flowchart shows the steps as boxes of various kinds, and

their order by connecting them with arrows.



**Fig 4.3 Flow chart diagram**

**CHAPTER 5**

**SUMMARY AND CONCLUSION**

With the increasing number of deaths due to heart diseases, it has become mandatory to develop a system to predict heart diseases effectively and accurately. The motivation for the study was to find the most efficient ML algorithm for detection of heart diseases. This study sees the accuracy score of Logistic Regression algorithm for predicting heart disease using UCI machine learning repository dataset. The result of this study indicates that the Logestic regression algorithm is the efficient algorithm with accuracy score of 85% for prediction of heart disease. In future the work can be enhanced by developing a web application based on the Logestic Regression algorithm as well as using a larger dataset as compared to the one used in this analysis which will help to provide better results and help health professionals in predicting the heart disease effectively and efficiently.

**REFERENCES**

[1] Avinash Golande, Pavan Kumar T, Heart Disease Prediction Using Effective Machine Learning Techniques, International Journal of Recent Technology and Engineering, Vol 8, pp.944-950,2019.

[2] T.Nagamani, S.Logeswari, B.Gomathy, Heart Disease Prediction using Data Mining with Mapreduce Algorithm, International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-3, January 2019.

[3] Fahd Saleh Alotaibi, Implementation of Machine Learning Model to Predict Heart Failure Disease, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 10, No. 6, 2019.

[4] Theresa Princy R,J. Thomas,Human heart Disease Prediction System using Data Mining Techniques, International Conference on Circuit Power and Computing Technologies,Bangalore,2016.

[5] Nagaraj M Lutimath,Chethan C,Basavaraj S Pol.,Prediction Of Heart Disease using Machine Learning, International journal Of Recent Technology and Engineering,8,(2S10), pp 474-477, 2019.

**APPENDIX**

1. **SCREENSHOTS**

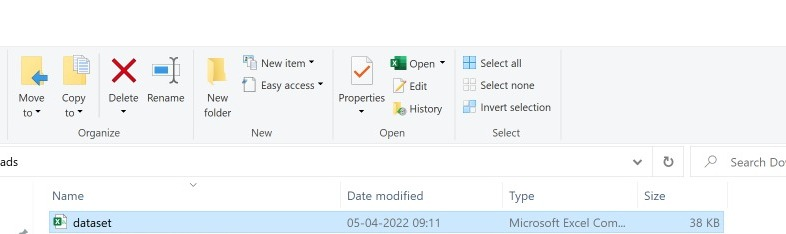
****

Fig 5.1 Import the data set

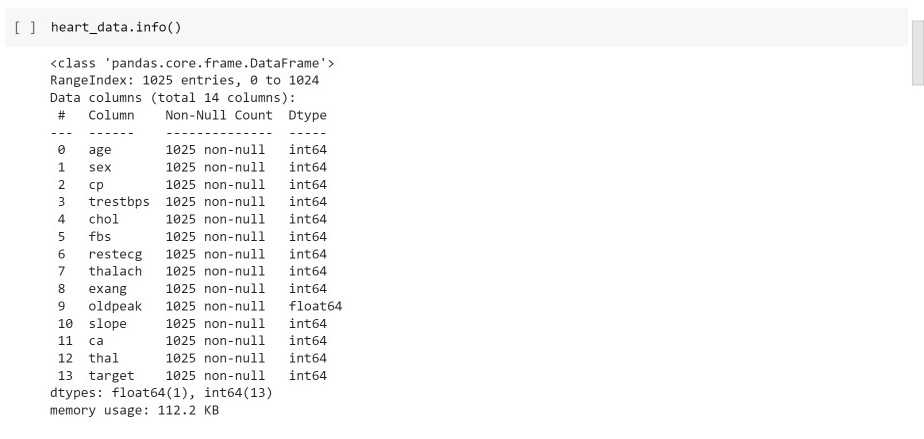
****

Fig 5.2 Dataset Information



Fig 5.3 Accuracy

1. **SOURCE CODE**

import numpy as np

import pandas as pd

import warnings

warnings.filterwarnings('ignore')

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import accuracy\_score heart\_data = pd.read\_csv('/content/dataset.csv')

heart\_data.head()

heart\_data.tail()

heart\_data.tail()

heart\_data.info()

heart\_data.isnull().sum()

heart\_data.describe()

heart\_data['target'].value\_counts()

X = heart\_data.drop(columns='target', axis=1)

Y = heart\_data['target']

print(X)

print(Y)

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size=0.2, stratify=Y, random\_state=2)

print(X.shape, X\_train.shape, X\_test.shape)

model = LogisticRegression()

model.fit(X\_train, Y\_train)

X\_train\_prediction = model.predict(X\_train)

training\_data\_accuracy = accuracy\_score(X\_train\_prediction, Y\_train) print('Accuracy on Training data : ', training\_data\_accuracy)

input\_data = (6,0,0,40,68,0,0,10,0,3.6,64,2,2)

input\_data\_as\_numpy\_array= np.asarray(input\_data)

input\_data\_reshaped = input\_data\_as\_numpy\_array.reshape(1,-1)

prediction = model.predict(input\_data\_reshaped)

print(prediction)

if (prediction[0]== 0):

print('The Person does not have a Heart Disease')

else:

print('The Person has Heart Disease')