**CHAPTER 1**

**INTRODUCTION**

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code.Python is a programming language that lets you work quickly and integrate systems more efficiently.

* 1. **FEATURES OF PYTHON**

**Python** is a dynamic, high level, free open source and interpreted programming language. It supports object-oriented programming as well as procedural oriented programming. In Python, we don’t need to declare the type of variable because it is a dynamic typed language.  
For example,x=10  
here x can be anything such as String, int etc.

There are many features in Python, some of which are discussed below –

**1.Easy**   
Python is high level programming language .Python is very easy to learn language as compared to other language like c, c#, java script, java etc. It is very easy to code in python language and anybody can learn python basic in few hours or days .It is also developer-friendly language.

**2.Free and Open Source:**

Python language is freely available at official website and you can download it from the given download link below click on the **Download Python** keyword .Download Python  
Since, it is open-source, this means that source code is also available to the public. So you can download it as, use it as well as share it.

**3.Object-Oriented Language:**  
One of the key features of python is Object-Oriented programming. Python supports object oriented language and concepts of classes, objects encapsulation etc.

**4.GUIProgrammingSupport:**  
Graphical Users interfaces can be made using a module such as PyQt5, PyQt4, wx Pythonor Tk in python.PyQt5 is the most popular option for creating graphical apps with Python.

**5. High-Level Language:**  
Python is a high-level language. When we write programs in python, we do not need to remember the system architecture, nor do we need to manage the memory.

**6.Extensible feature:**  
Python is a **Extensible** language. we can write our some python code into c or c++ language and also we can compile that code in c/c++ language.

**7. Python is Portable language:**  
Python language is also a portable language.for example, if we have python code for windows and if we want to run this code on other platform such as Linux, Unix and Mac then we do not need to change it, we can run this code on any platform.

**8. Python is Integrated language:**  
Python is also an Integrated language because we can easily integrated python with other language like c,c++ etc.

**9. Interpreted Language:**  
Python is an Interpreted Language. because python code is executed line by line at a time. like other language c, c++, java etc there is no need to compile python code this makes it easier to debug our code. The source code of python is converted into an immediate form called **bytecode**.

**10. Large Standard Library**  
Python has a large standard library which provides rich set of module and functions so you do not have to write your own code for every single thing. There are many libraries present in python for such as regular expressions, unit-testing, web browsers etc.

**11. Dynamically Typed Language:**  
Python is dynamically-typed language. That means the type (for example- int, double, long etc) for a variable is decided at run time not in advance. Because of this feature we don’t need to specify the type of variable.

**1.2 FEATURES OF MACHINE LEARNING**

# Introduction To Machine Learning using Python

Machine learning is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. Machine learning focuses on the development of Computer Programs that can change when exposed to new data. In this article, we’ll see basics of Machine Learning, and implementation of a simple machine learning algorithm using python.

**Setting up the environment**

Python community has developed many modules to help programmers implement machine learning. In this article, we will be using numpy , scipy and scikit-learn modules. We can install them using cmd command:

pip install numpy scipy scikit-learn

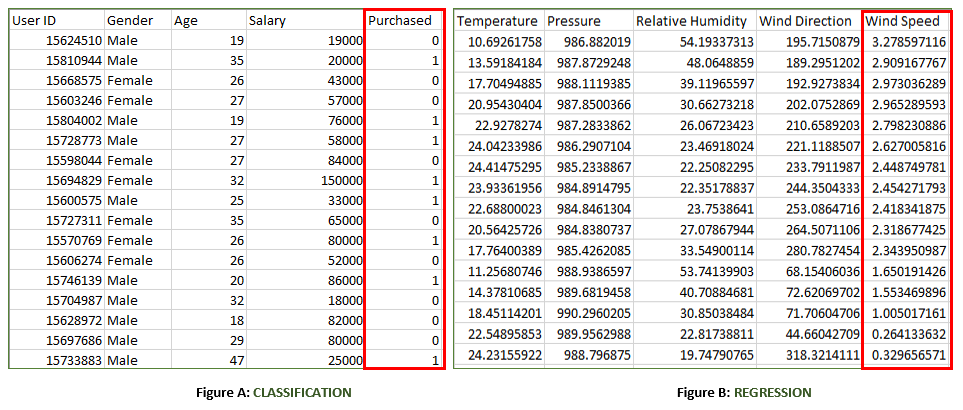
A better option would be downloading miniconda or anaconda packages for python, which come prebundled with these packages. Follow the instructions given [here](https://docs.continuum.io/) to use anaconda

.**1.2.1Machine Learning overview**

Machine learning involves computer to get trained using a given data set, and use this training to predict the properties of a given new data. For example, we can train computer by feeding it 1000 images of cats and 1000 more images which are not of a cat, and tell each time to computer whether a picture is cat or not. Then if we show the computer a new image, then from the above training, computer should be able to tell whether this new image is cat or not.  
Process of training and prediction involves use of specialized algorithms. We feed the training data to an algorithm, and the algorithm uses this training data to give predictions on a new test data. One such algorithm is [K-Nearest-Neighbor](https://www.geeksforgeeks.org/k-nearest-neighbours/) classification (KNN classification). It takes a test data, and finds k nearest data values to this data from test data set. Then it selects the neighbor of maximum frequency and gives its properties as

the prediction result.

**Types of Machine learning**

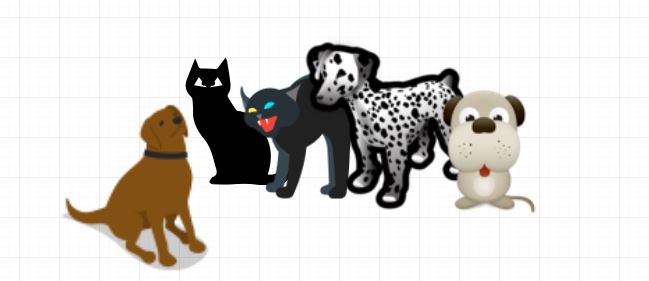
**a)Supervised Learning :**  
Supervised learning is when the model is getting trained on a labelled dataset. **Labelled** dataset is one which have both input and output parameters. In this type of learning both training and validation datasets are labelled as shown in the figures below.  


**Types of Supervised Learning:**

1. **Classification :**It is a Supervised Learning task where output is having defined labels(discrete value). For example in above Figure A, Output – Purchased has defined labels i.e. 0 or 1 ; 1 means the customer will purchase and 0 means that customer won’t purchase. The goal here is to predict discrete values belonging to a particular class and evaluate on the basis of accuracy.  
   It can be either binary or multi class classification. In **binary** classification, model predicts either 0 or 1 ; yes or no but in case of **multi class** classification, model predicts more than one class.  
   **Example:** Gmail classifies mails in more than one classes like social, promotions, updates, forum.
2. **Regression :**It is a Supervised Learning task where output is having continuous value.  
   Example in above Figure B, Output – Wind Speed is not having any discrete value but is continuous in the particular range. The goal here is to predict a value as much closer to actual output value as our model can and then evaluation is done by calculating error value. The smaller the error the greater the accuracy of our regression model.

**b)Unsupervised Learning**

Unsupervised learning is the training of machine using information that is neither classified nor labeled and allowing the algorithm to act on that information without guidance. Here the task of machine is to group unsorted information according to similarities, patterns and differences without any prior training of data.

Unlike supervised learning, no teacher is provided that means no training will be given to the machine. Therefore machine is restricted to find the hidden structure in unlabeled data by our-self.  
**For instance**, suppose it is given an image having both dogs and cats which have not seen ever.  


Thus the machine has no idea about the features of dogs and cat so we can’t categorize it in dogs and cats. But it can categorize them according to their similarities, patterns, and differences i.e., we can easily categorize the above picture into two parts. First first may contain all pics having **dogs** in it and second part may contain all pics having **cats** in it. Here you didn’t learn anything before, means no training data or examples.

Unsupervised learning classified into two categories of algorithms:

* **Clustering**: A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behavior.
* **Association**: An association rule learning problem is where you want to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y.
  + 1. **ADVANTAGES**

1) Easy to code

2) Contains rich libraries and are easy and direct to implement

3) Easily compatible on any os

4) Reduces the size of code

**1.3 OUTLINE OF PROJECT**

The main goal of the project is to find whether the person is having heart disease or not by training the machine with previous results taken from dataset using machine learning algorithms

**1.4 LITERATURE REVIEW**

As the project determines the heart disease or not based on previous data provided in dataset. we use supervised machine learning techniques to train data for future analysis of heart disease for a person .

**STUDY OF THE SYSTEM**

1. Numpy
2. Pandas
3. Matplotlib
4. Scikit –learn

**1 . 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.

**2.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.

**3.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](http://ipython.org/) shells, the [Jupyter](http://jupyter.org/) 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](https://matplotlib.org/tutorials/introductory/sample_plots.html) and [thumbnail gallery](https://matplotlib.org/gallery/index.html).

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.

**4.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. The library is built upon the SciPy (Scientific Python) that must be installed before you can use scikit-learn. This stack that includes:

* **NumPy**: Base n-dimensional array package
* **SciPy**: Fundamental library for scientific computing
* **Matplotlib**: Comprehensive 2D/3D plotting
* **IPython**: Enhanced interactive console
* **Sympy**: Symbolic mathematics
* **Pandas**: Data structures and analysis
* Extensions or modules for SciPy care conventionally named [SciKits](http://scikits.appspot.com/scikits). As such, the module provides learning algorithms and is named scikit-learn.

**CHAPTER 2**

**AIM AND SCOPE**

**2.1 AIM**

The main aim of the project is to find whether the person is having heart disease or not using machine learning techniques.

**2.2 INPUT AND OUTPUTS**

The following some are the projects inputs and outputs.

**Inputs:**

* Importing the all required packages like numpy, pandas, matplotlib, scikit – learn and required machine learning algorithms packages .
* Setting the dimensions of visualization graph.
* Downloading and importing the dataset and convert to data frame.

**Outputs:**

* preprocessing the importing data frame for imputing nulls with the related information.
* All are displaying cleaned outputs.
* After applying machine learning algorithms it will give good results and visualization plots.

**Scope:**

In future heart patients will be increased more and more. My main theme of the project is to improve the speed in process through machine learning that it will predict whether the patient is having heart disease or not through his report.This project is more helpful to cardiology hospitals in future in predicting heart diseases.

**CHAPTER 3**

**EXPERIMENTAL OR MATERIALS AND METHODS ALGORITHMS USED**

**3.1 ALGORITHMS USED**

**a)Logistic Regression** is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary).  Like all regression analyses, the logistic regression is a predictive analysis.  Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

b)**Random Forest** Random forests, also known as random decision forests, are a popular ensemble method that can be used to build [predictive models](https://www.datascience.com/resources/white-papers/executives-guide-to-predictive-data-modeling) for both classification and regression problems. Ensemble methods use multiple learning models to gain better predictive results — in the case of a random forest, the model creates an entire forest of random uncorrelated decision trees to arrive at the best possible answer.

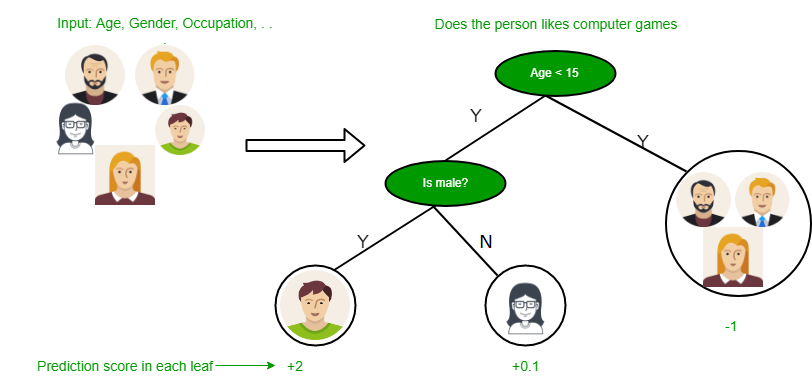
C**)Gaussian Naïve Bayes** Naïve Bayes classifiers are a collection of classification algorithms based on Bayes’ Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

d)**K-Nearest Neighbor Classfier** The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. KNN captures the idea of similarity (sometimes called distance, proximity, or closeness) with some mathematics we might have learned in our childhood— calculating the distance between points on a graph. There are other ways of calculating distance, and one way might be preferable depending on the problem we are solving. However, the straight-line distance (also called the Euclidean distance) is a popular and familiar choice.

e)**Decision Tree Classifier** Decision tree algorithm falls under the category of supervised learning. They can be used to solve both regression and classification problems.

Decision tree uses the tree representation to solve the problem in which each leaf node corresponds to a class label and attributes are represented on the internal node of the tree.

* We can represent any boolean function on discrete attributes using the decision tree.



***FIG 3.3 Decision tree***

.

**CHAPTER 4**

**REQUIREMENT SPECIFICATION**

**4.1 INTRODUCTION**

To be used efficiently, all computer software needs certain hardware components or other software resources to be present on a computer. These prerequisites are known as computer system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: Minimum and Recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. Industry analysts suggest that this trend plays a bigger part in driving upgrades to existing computer systems than technological advancements.

**4.2 SOFTWARE REQUIREMENTS**

The software requirements are:Software Requirements deal with definition of software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed.

SOFTWARE REQUIREMENTS FOR PRESENT PROJECT:

OPERATING SYSTEM: - Windows 7/ XP/8/10

LANGUAGE :- PYTHON,MACHINE LEARNING

**4.3 HARWARE REQUIREMENTS**

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in the case of operating systems. An HCL list tested, compatibility and sometimes incompatible hardware devices for a particular operating system or application. The following subsections discuss the various aspects of hardware requirements.

HARDWARE REQUIREMENTS FOR PRESENT PROJECT:

PROCESSOR: - Intel dual Core i3/i5/i7 RAM: - 1 GB HARD

DISK: - 80 GB

**CHAPTER 5**

**PROJECT DESCRIPTION**

The main aim of the project is to find whether the person is having heart disease or not using machine learning techniques. In future heart patients will be increased more and more. My main theme of the project is to improve the speed in process through machine learning that it will predict whether the patient is having heart disease or not through his report.This project is more helpful to cardiology hospitals in future in predicting heart diseases.

**5.1 FLOW CHART**

**Use Case Diagram:**



***FIG 5.1 Use case diagram***

**Sequence diagram:**

Sequence Diagrams Represent the objects participating the interaction horizontally and time vertically. A Use Case is a kind of behavioral classifier that represents a declaration of an offered behavior. Each use case specifies some behavior, possibly including variants that the subject can perform in collaboration with one or more actors. Use cases define the offered behavior of the subject without reference to its internal structure. These behaviors, involving interactions between the actor and the subject, may result in changes to the state of the subject and communications with its environment. A use case can include possible variations of its basic behavior, including exceptional behavior and error handling.



***FIG 5.2 Sequence diagram***

**Activity diagram:**



***FIG 5.3 Activity diagram***

**Collaboration Diagram:**

****

***FIG 5.4 Cpllaboration diagram***

**Class Diagram:**



***FIG 5.5 Class Diagram***

**5.2 ALGORITHMS**

**a) Logistic Regression** is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary).  Like all regression analyses, the logistic regression is a predictive analysis.  Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

b)**Random Forest** Random forests, also known as random decision forests, are a popular ensemble method that can be used to build [predictive models](https://www.datascience.com/resources/white-papers/executives-guide-to-predictive-data-modeling) for both classification and regression problems. Ensemble methods use multiple learning models to gain better predictive results — in the case of a random forest, the model creates an entire forest of random uncorrelated decision trees to arrive at the best possible answer.

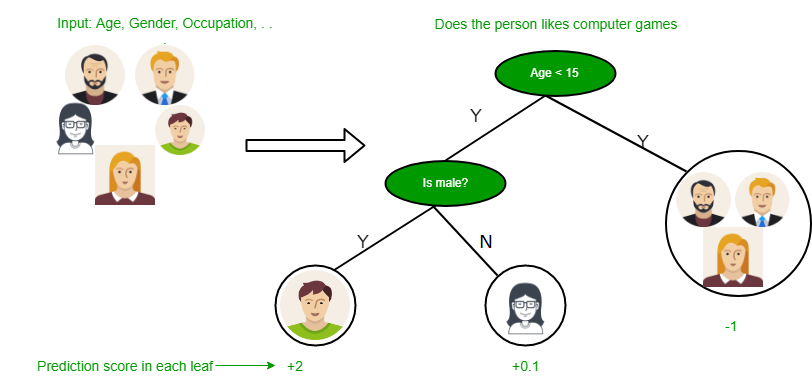
C**)Gaussian Naïve Bayes** Naïve Bayes classifiers are a collection of classification algorithms based on Bayes’ Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

d)**K-Nearest Neighbor Classfier** The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. KNN captures the idea of similarity (sometimes called distance, proximity, or closeness) with some mathematics we might have learned in our childhood— calculating the distance between points on a graph. There are other ways of calculating distance, and one way might be preferable depending on the problem we are solving. However, the straight-line distance (also called the Euclidean distance) is a popular and familiar choice.

e)**Decision Tree Classifier** Decision tree algorithm falls under the category of supervised learning. They can be used to solve both regression and classification problems.

Decision tree uses the tree representation to solve the problem in which each leaf node corresponds to a class label and attributes are represented on the internal node of the tree.

* We can represent any boolean function on discrete attributes using the decision tree.



***Fig 5.6 Decision tree diagram***

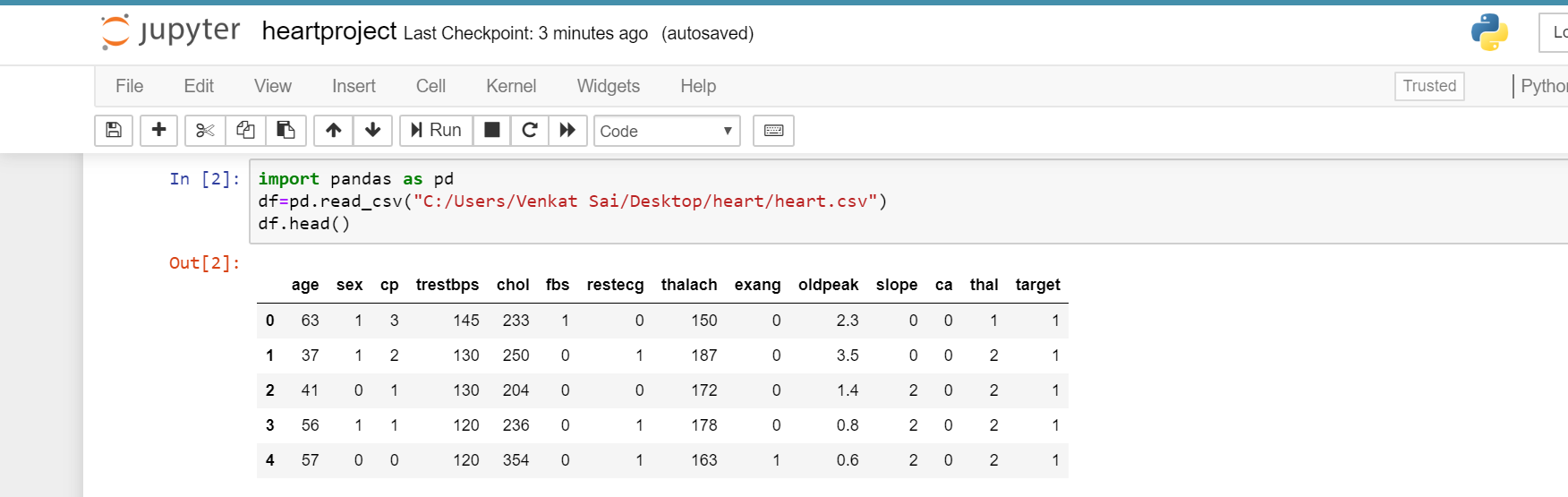
**CHAPTER 6**

**RESULTS AND DISCUSSION**

**6.1 RESULTS**

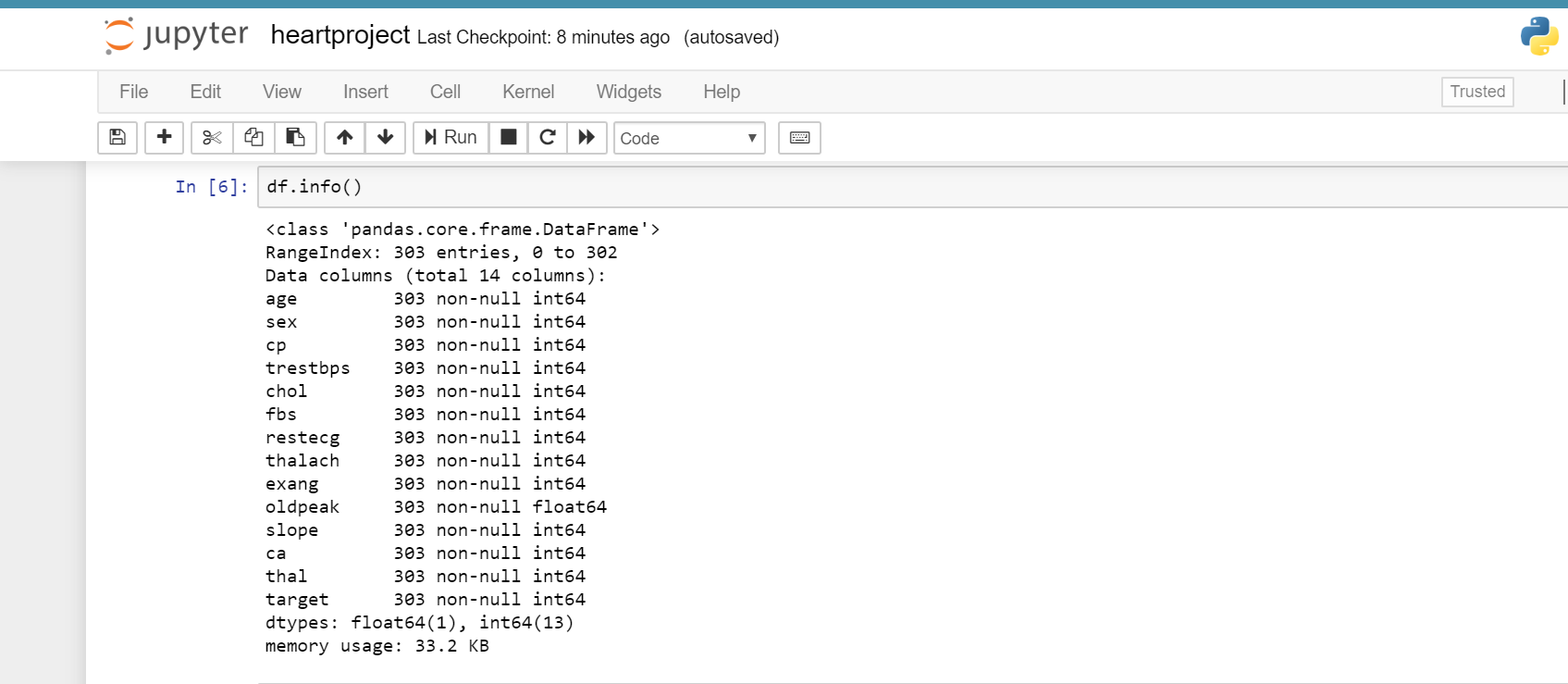
**6.1.1 SCREENSHOTS**

**Dataset Importing**



***Fig 6.1 Dataset Importing***

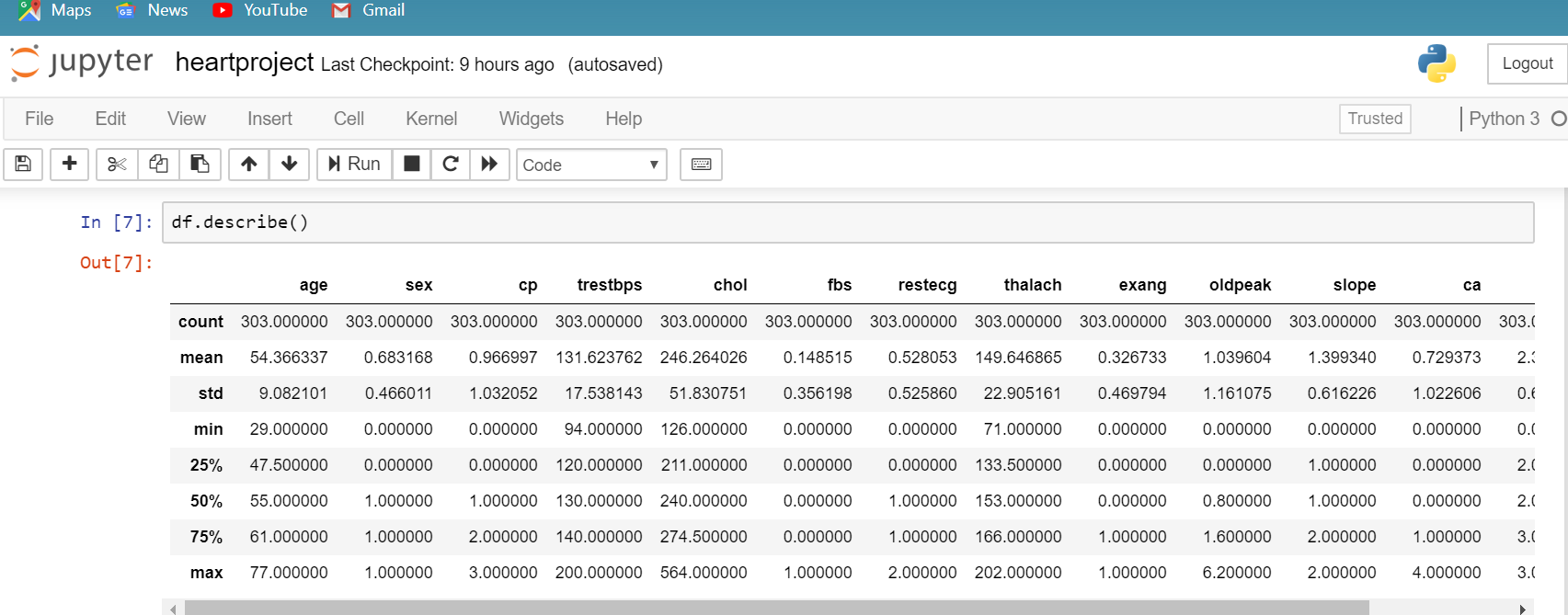
The above figure depicts the output genererated when we give run that shows dataset importing.

**Info of data** 

***Fig 6.2 Info of data***

This output gives information of data n dataset

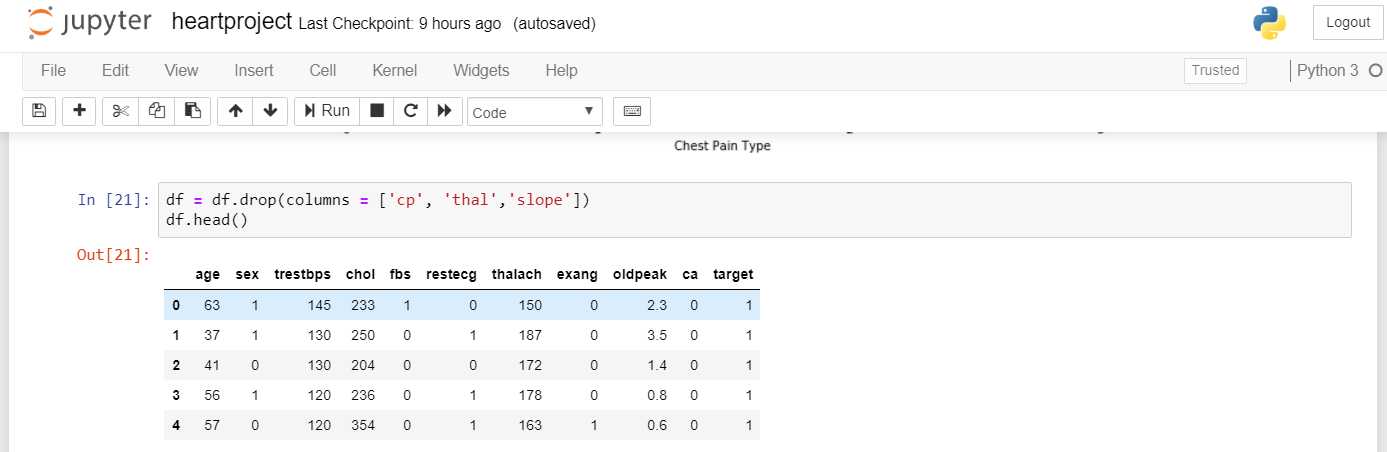
**Stats of data**

****

***Fig 6.3 Stats of data***

This output gives statistical information of the dataset.

**Dropping columns**



***Fig 6.4 Dropping of columns***

Dropping of some columns useless columns

**Graph of target**

****

***Fig 6.5 Graph of target***

This graph shows comparision of target values.

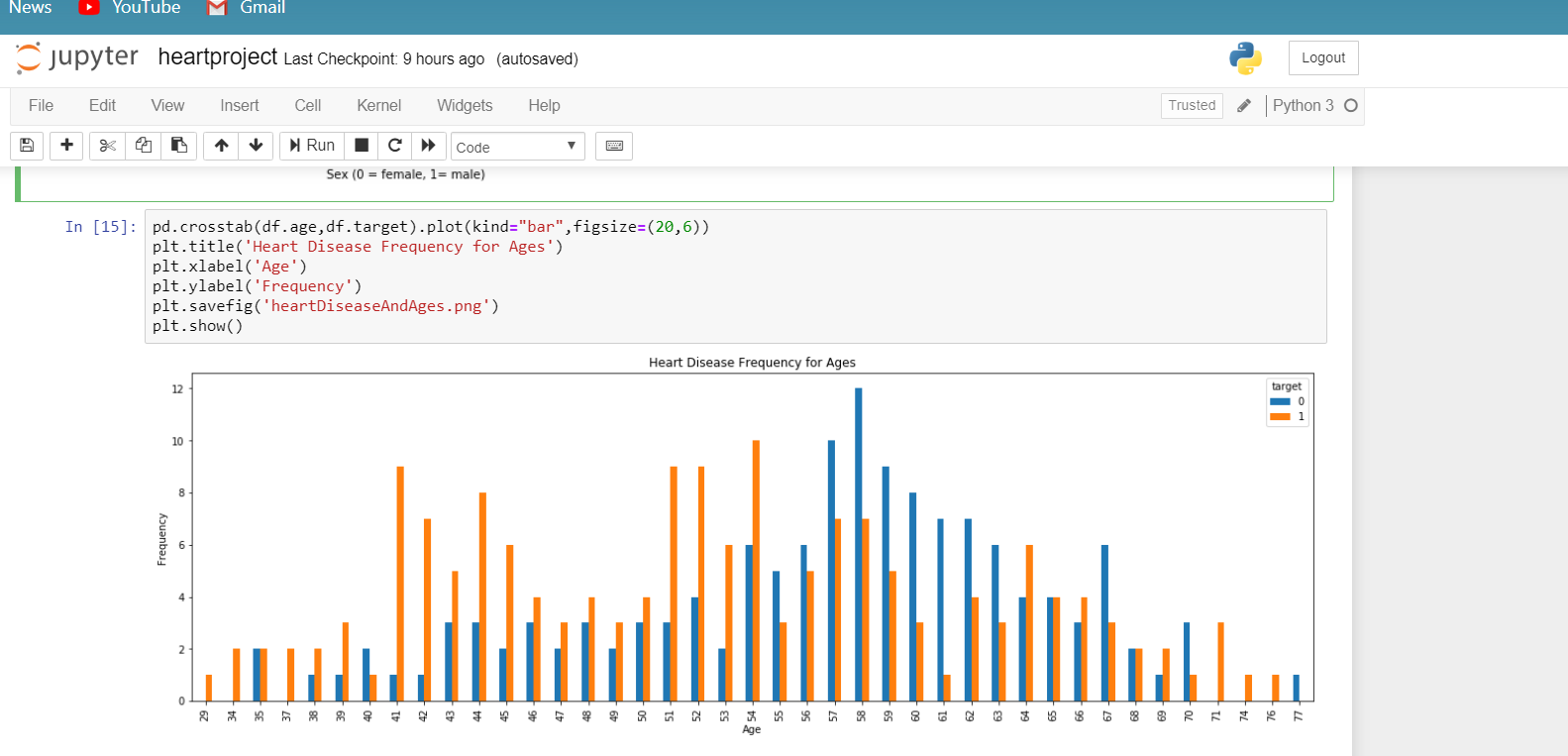
**Graph of Sex(male,female)**



***Fig 6.6 Graph of Sex(male,female)***

This graph shows how many and female is having heart diseasae

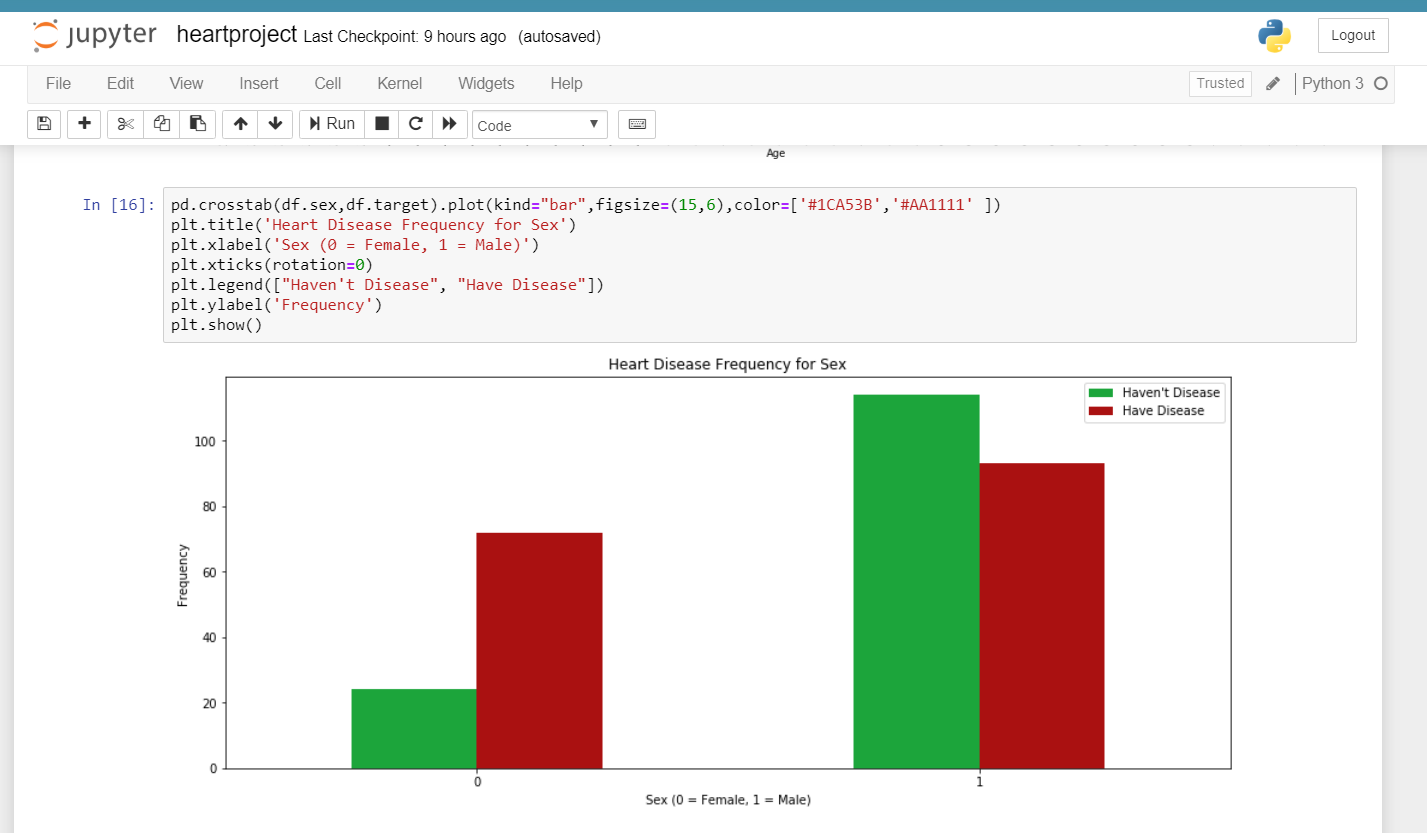
**Graph of age with target**



***Fig 6.7Graph of age with target***

This graph describes no of people getting heart diseses w.r.t age

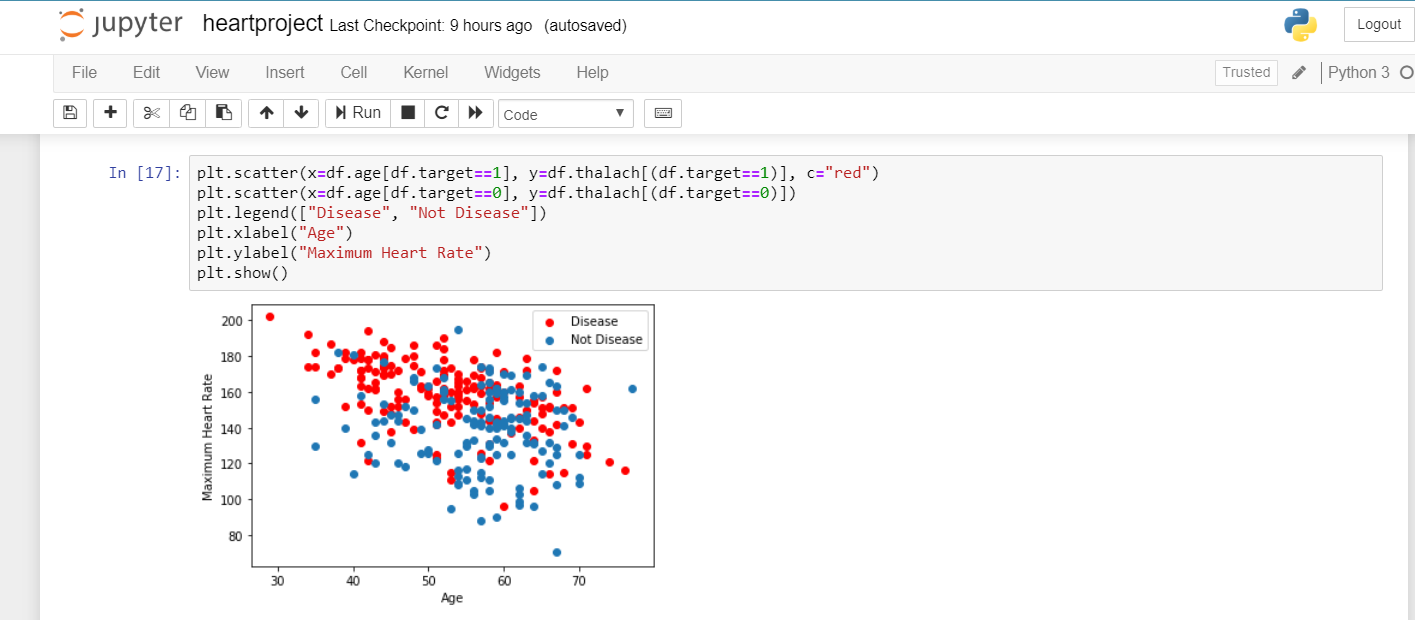
**Graph of sex with target**



***Fig 6.8 Graph of sex with target***

This graph describes no of people getting heart diseases w.r.t sex

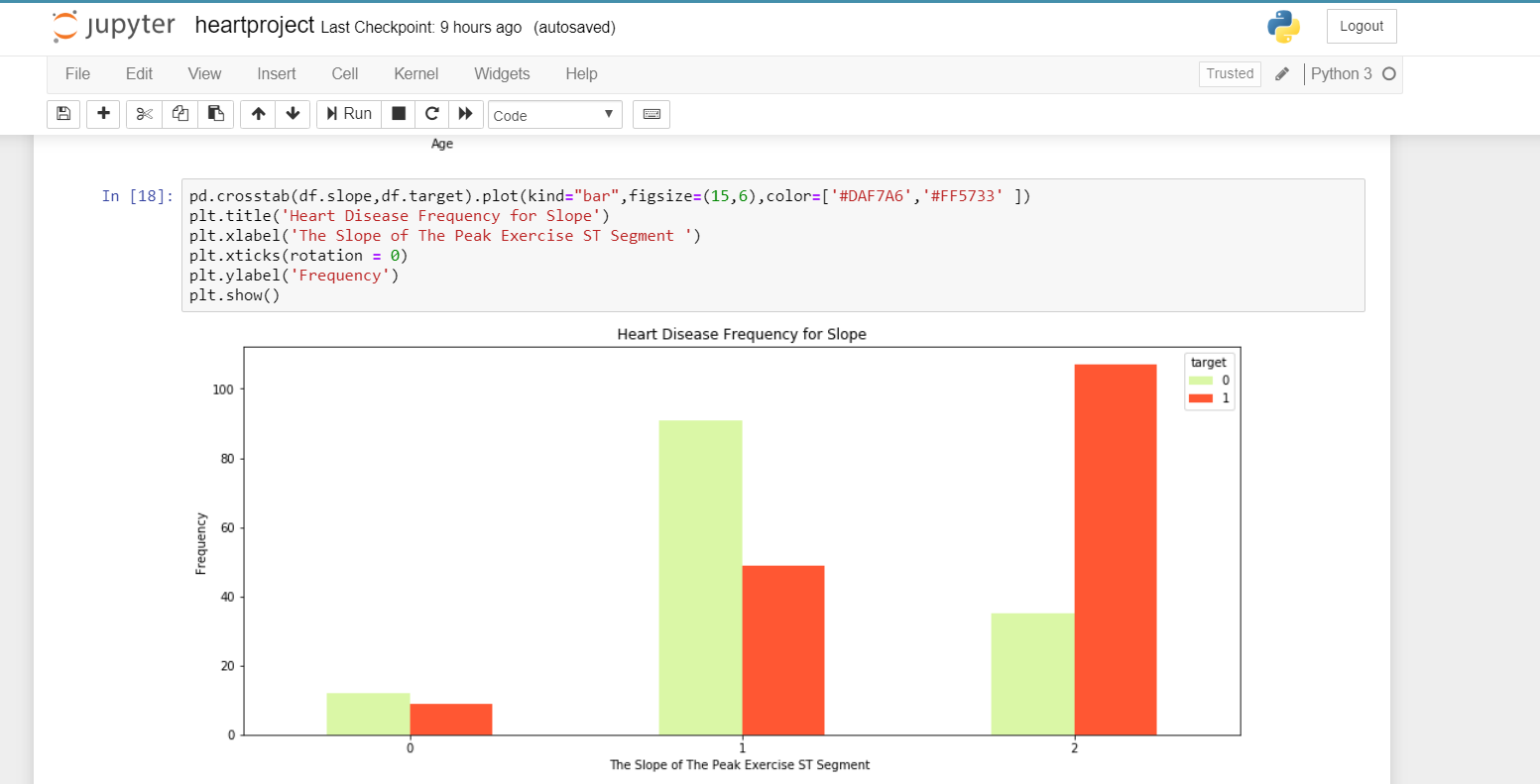
**Scatterplot of age and target**



***Fig 6.9 Scatterplot of age and target***

This scatterplot describes no of people getting heart diseases w.r.t age

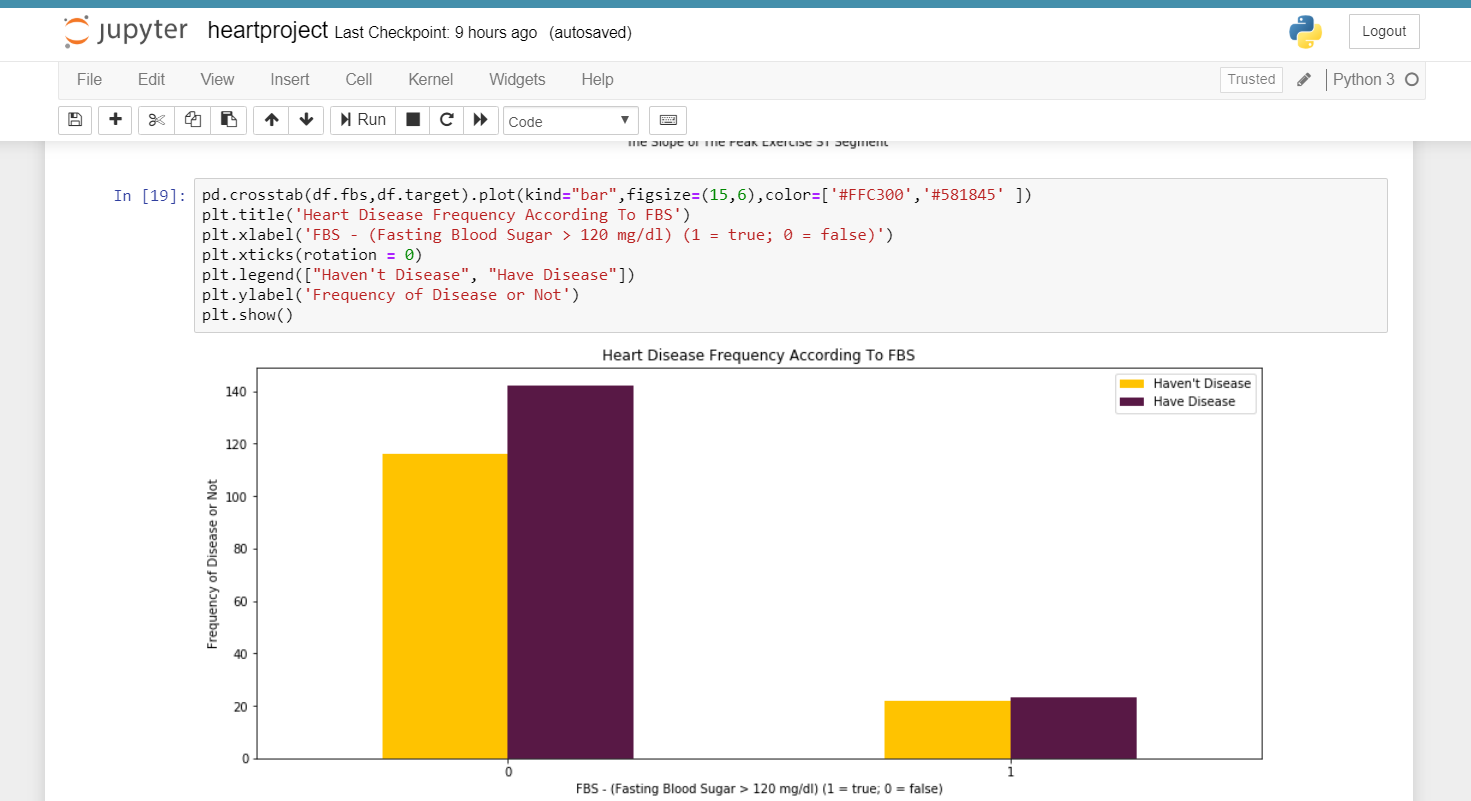
**Graph of slope and target**



***Fig 6.1 Graph of slope and target***

This graph describes no of people getting heart disease w.r.t slope

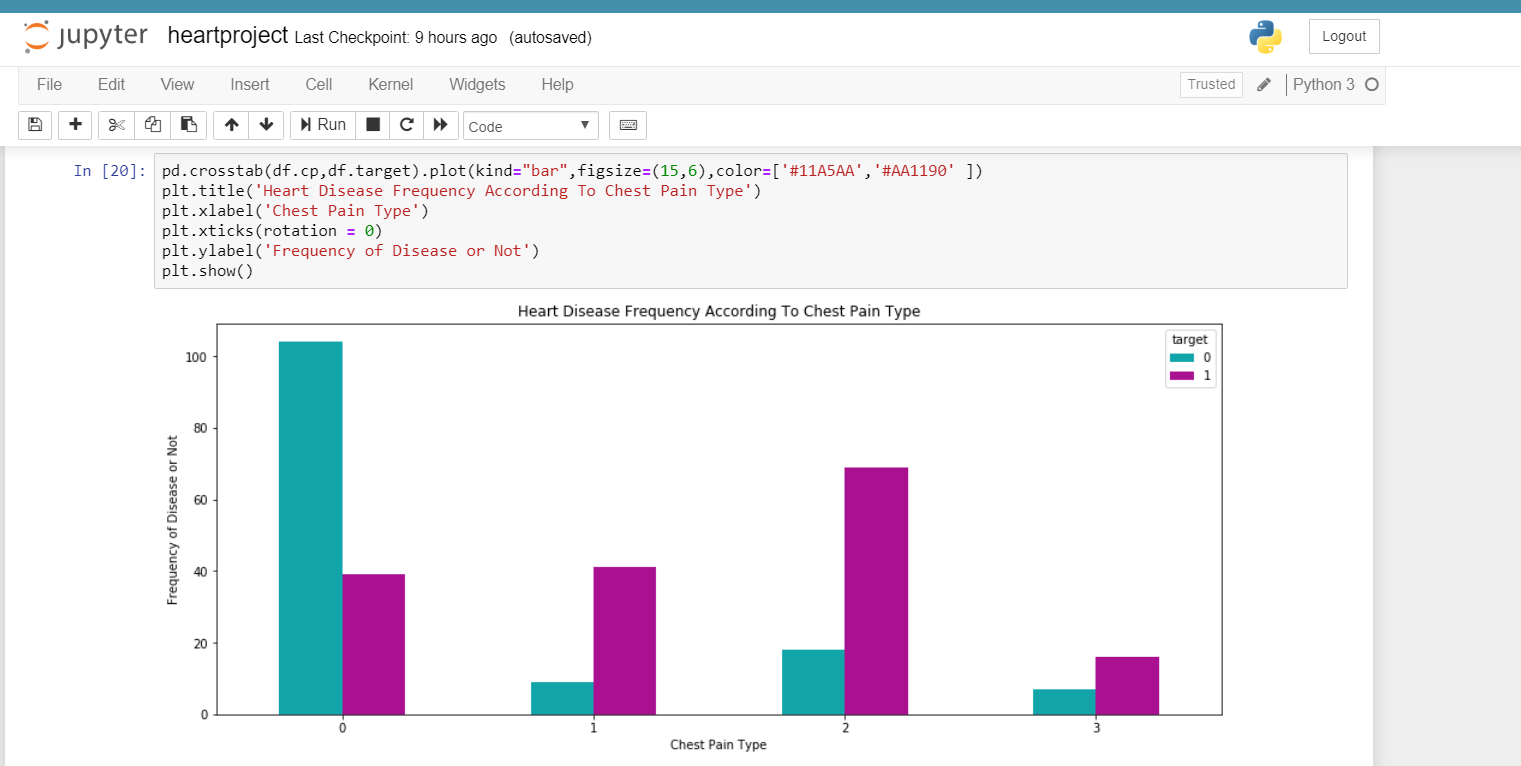
**Graph of fbs and target**



***Fig 6.11 Graph of fbs and target***

This graph describes no of people getting heart diseses w.r.t fbs

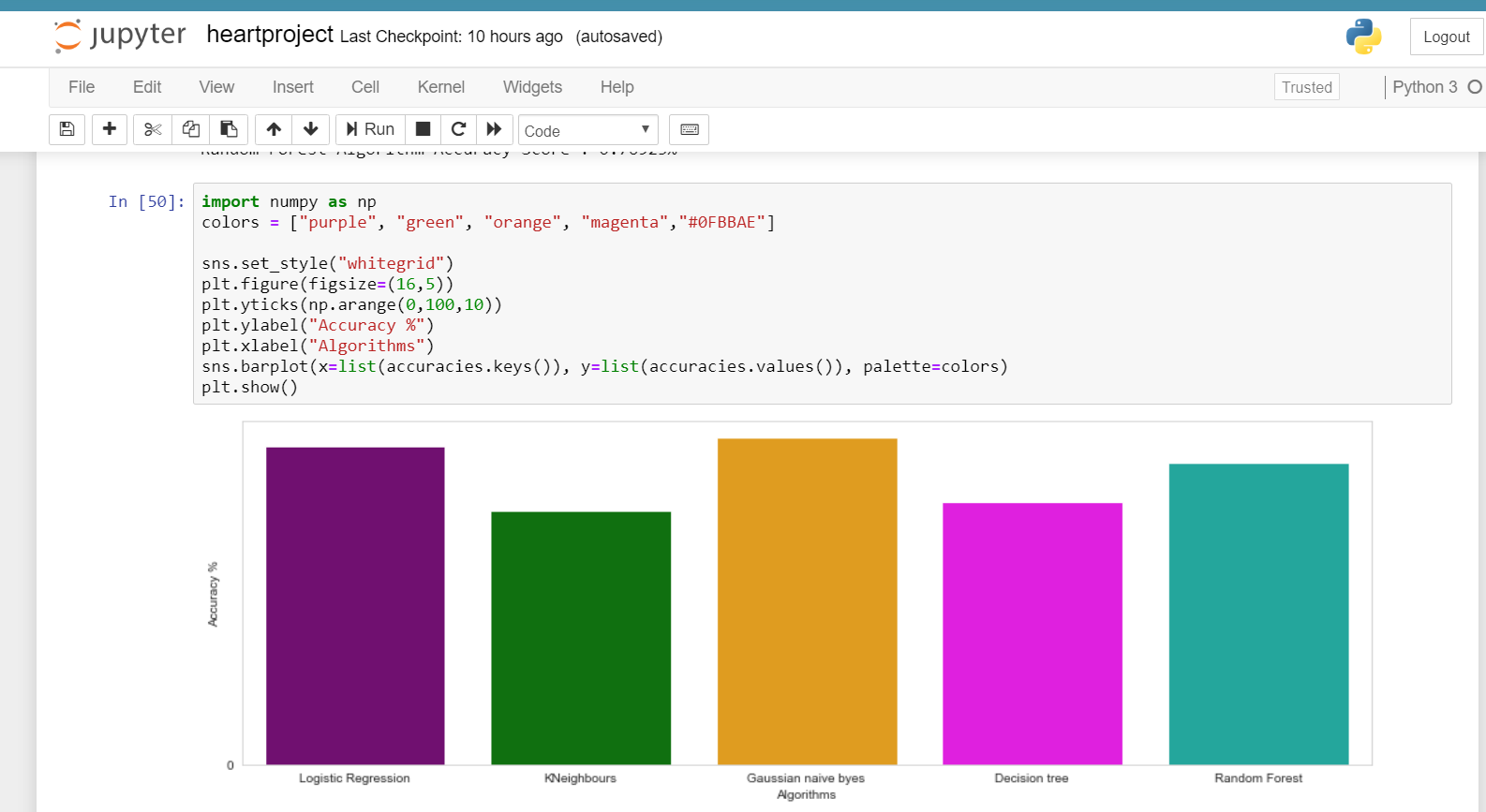
**Graph of cp and target**



***Fig 6.12 Graph of cp and target***

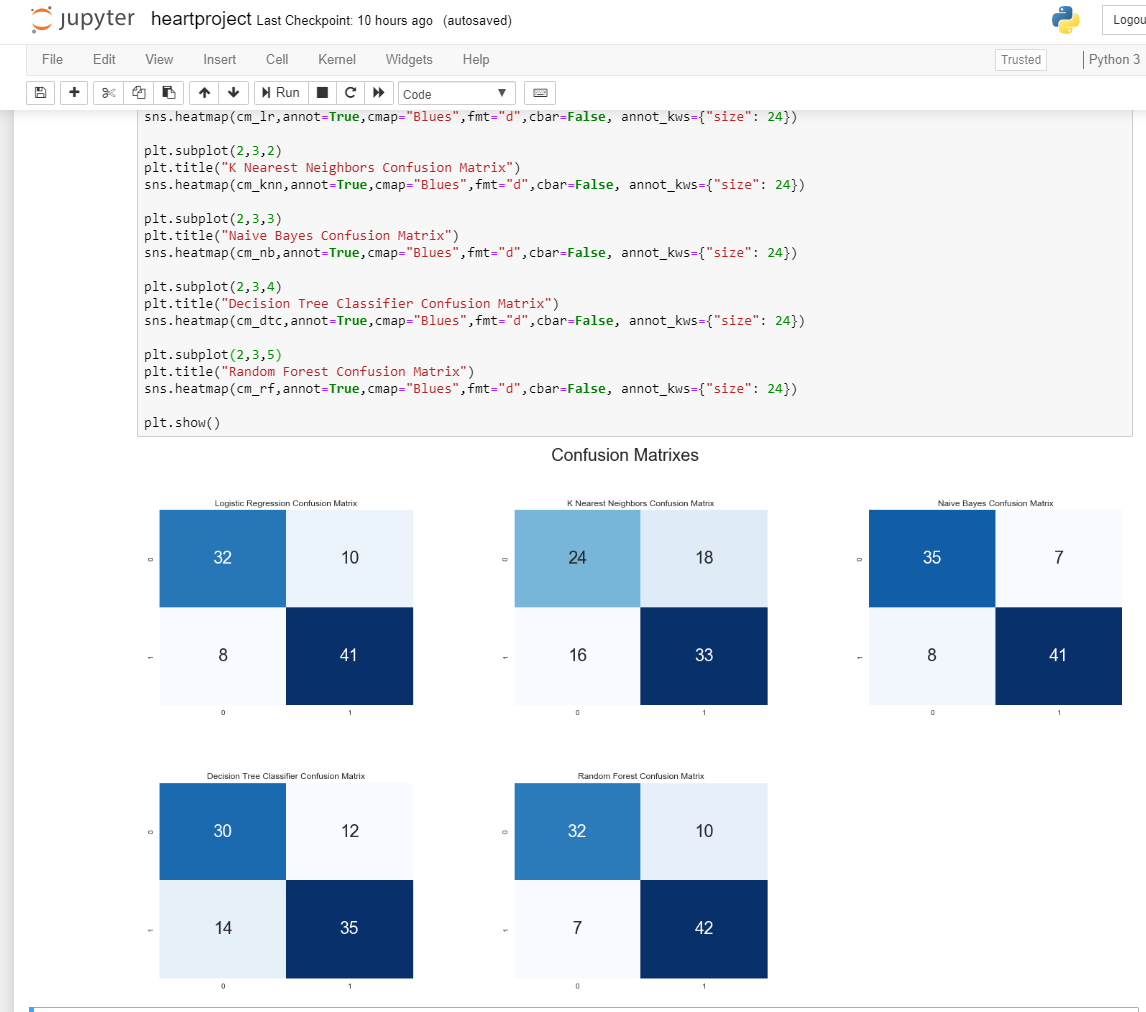
This graph describes no of people getting heart diseses w.r.t cp

**Comparision of algorithms**



***Fig 6.13 Comparision of algorithms***

Here I am comparing accuracies of all algorithms

**Comparision of confusion matrices**

***Fig 6.14* *Comparision of confusion matrices***

Here I am comparing confusion matrices of all algorithms

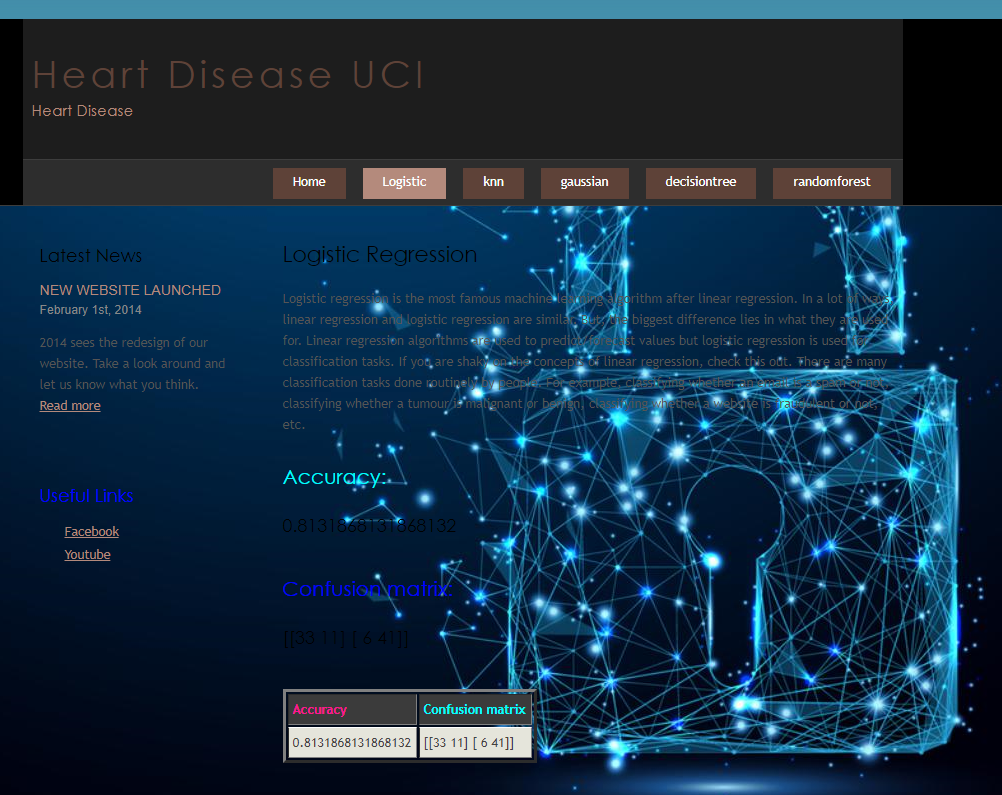
**Django output of Home**



***Fig 6.15 Django output of Home***

It contains details of types of algorithms used in project

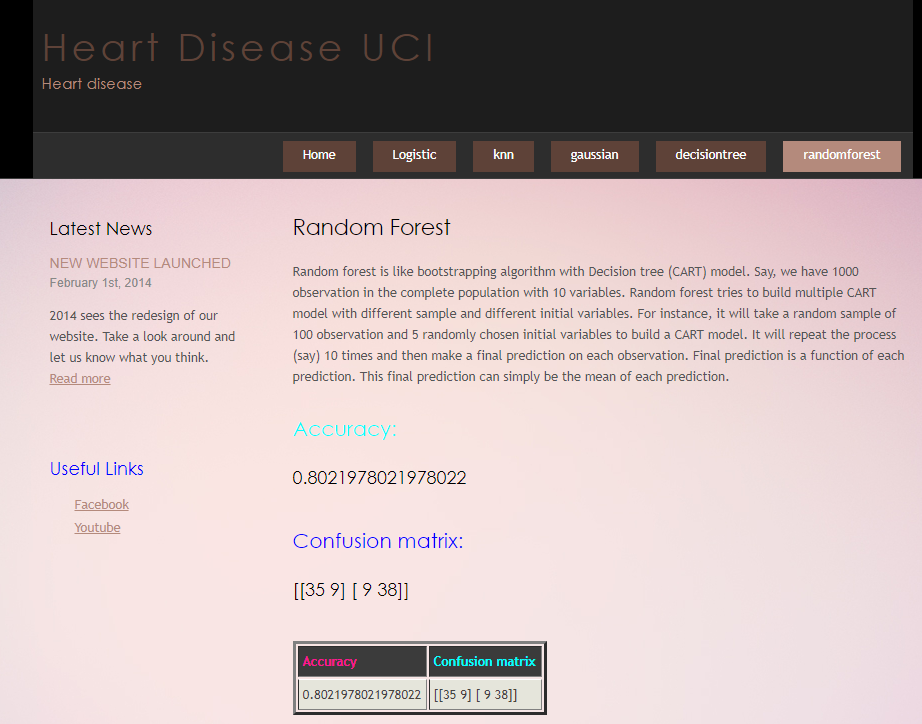
**Logistic Regression**



***Fig 6.16 Logistic Regression***

In this diagram we will display Logistic Regression confusion matrix and accuracy

**Random Forest**



***Fig 6.17 Random Forest***

In this diagram we will display Random Forest confusion matrix and accuracy

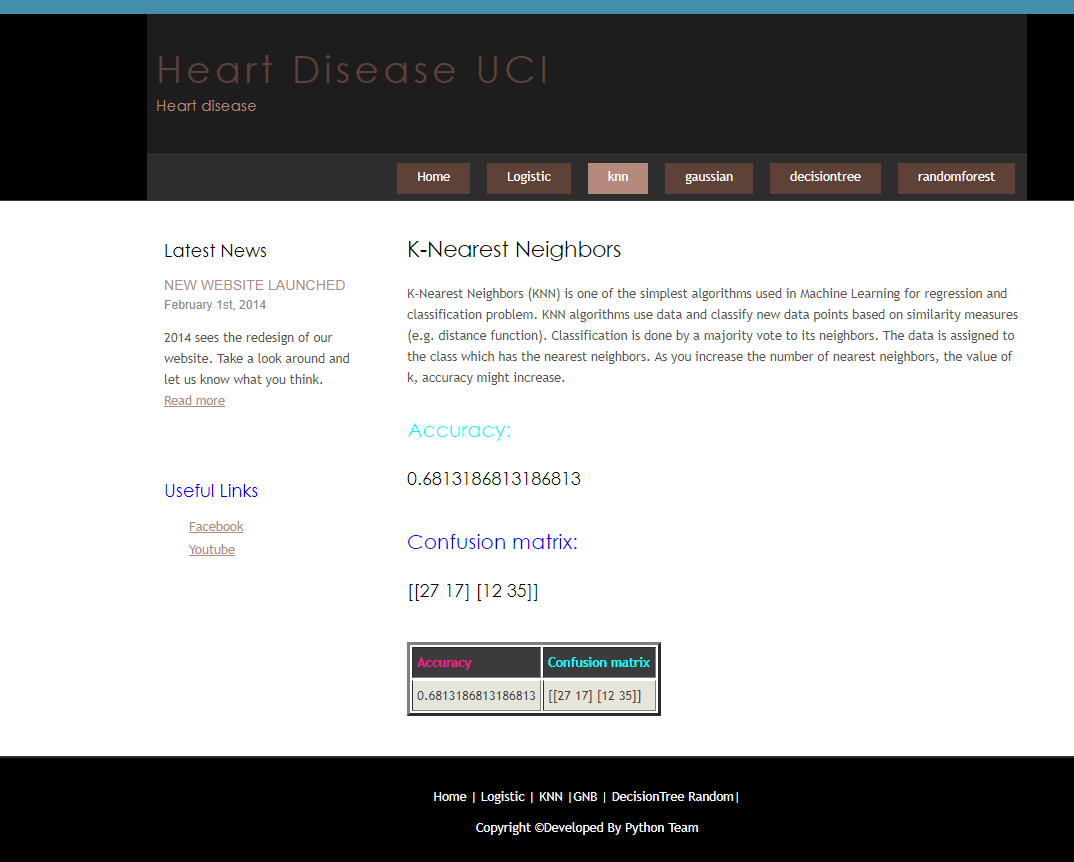
**Gaussian naïve Bayes**



***Fig 6.18 Gaussian naïve Bayes***

In this diagram we will display Gaussian Naïve Bayes confusion matrix and accuracy

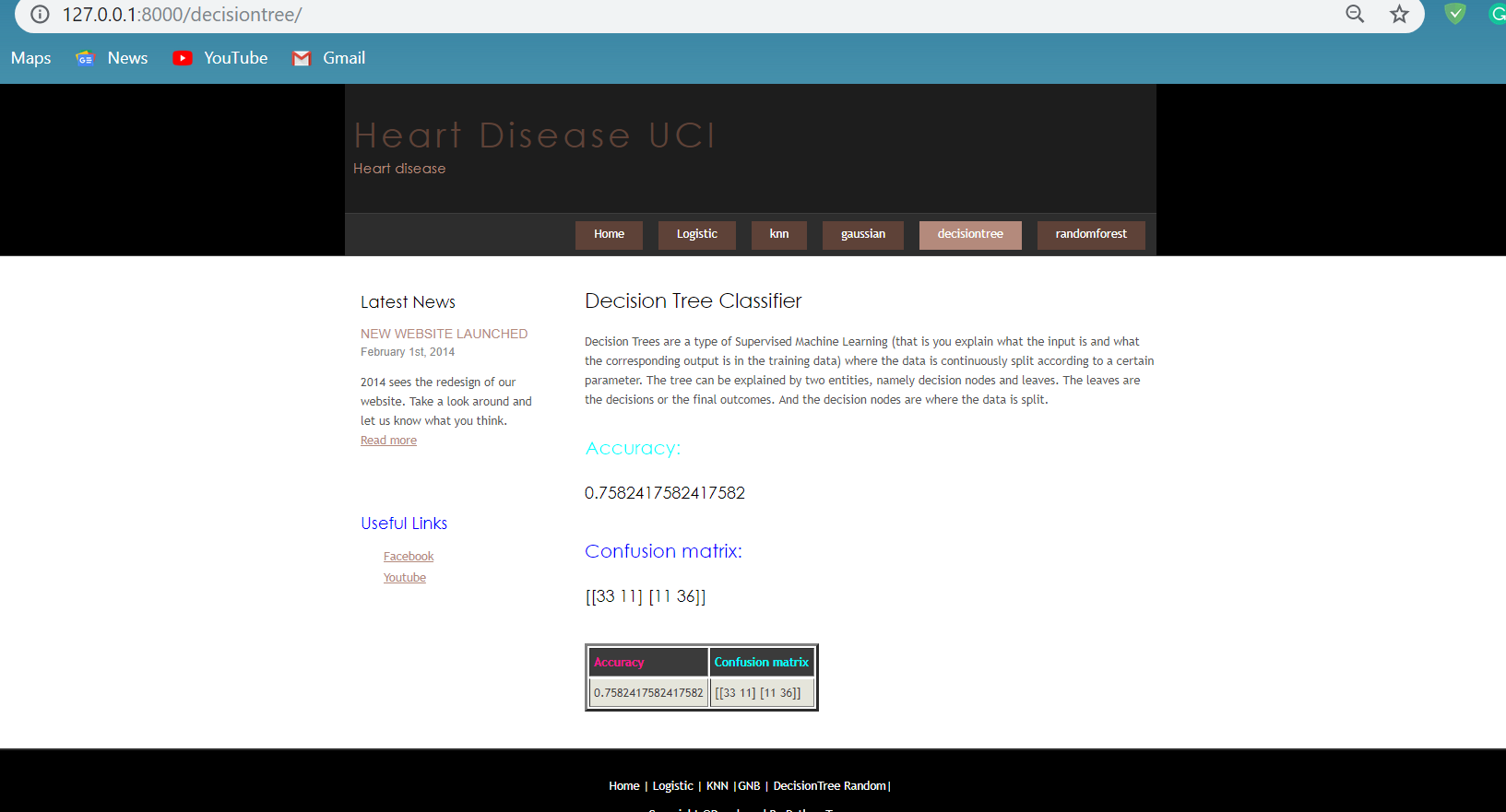
**K-Nearst Neighbors Classifier**



***Fig 6.19 K-Nearst Neighbors Classifier***

In this diagram we will display K-Nearst Neighbors Classifier confusion matrix and accuracy

**Decision Tree Classifier**



***Fig 6.20 Decision Tree Classifier***

In this diagram we will display Decision Tree Classifier and confusion matrix and accuracy

**CHAPTER 7**

**CONCLUSION AND FUTURE WORK**

**7.1 CONCLUSION**

Hence this project is useful in hospitals for finding heart diseases for people who is suffering with symptoms .this is useful in predicting a heart disease for a person

the performance of the Logistic Regression are compared the accuracy obtained between dataset and patients real time dataset to predict heart diseases.

**7.2 FUTURE WORK**

This project is useful in predicting cardiologist attacks in various hospitals in future Because of more people are suffering from heart attacks without knowing their disease and they are dieing unexpectedly .This machine learning framework is useful for predicting heart diseases for somany people through reports.

**REFERENCES**

Creators:   
1. Hungarian Institute of Cardiology. Budapest: Andras Janosi, M.D.   
2. University Hospital, Zurich, Switzerland: William Steinbrunn, M.D.   
3. University Hospital, Basel, Switzerland: Matthias Pfisterer, M.D.   
4. V.A. Medical Center, Long Beach and Cleveland Clinic Foundation: Robert Detrano, M.D., Ph.D.

**SOURCE CODE**

import pandas as pd

df=pd.read\_csv("C:/Users/Venkat Sai/Desktop/heart/heart.csv")

print(df)

df.head()

df = df.drop(columns = ['cp', 'thal', 'slope'])

df.head()

df['chol'].value\_counts

X=df.iloc[:,:-1]

X.head()

y=df.iloc[:,-1]

y.head()

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

X\_train,X\_test,y\_train,y\_test=train\_test\_split(X,y,test\_size=0.3)

len(X\_train)

len(y\_train)

len(X\_test)

len(y\_test)

accuracies={}

from sklearn.linear\_model import LogisticRegression

lr=LogisticRegression()

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

pred\_lr=lr.predict(X\_test)

pred\_lr

from sklearn.metrics import accuracy\_score,confusion\_matrix,classification\_report

cm\_lr=confusion\_matrix(y\_test,pred\_lr)

cm\_lr

print(classification\_report(y\_test,pred\_lr))

acc=accuracy\_score(y\_test,pred\_lr)

accuracies['Logistic Regression'] = acc

print("Logistic Regression Algorithm Accuracy Score : {:.5f}%".format(acc))