**DATA SCIENCES**

(Predicting Amount of Purchase)

*Summer Internship Report Submitted in partial fulfillment of the requirement for undergraduate degree of*

**Bachelor of Technology**

In

**Computer Science Engineering**

By

## Siri.Anubolu

221710303055

*Under the Guidance of*

**Mr.**



Department Of Computer Science Engineering

GITAM School of Technology

GITAM (Deemed to be University)

Hyderabad-502329

June 2020

i

**DECLARATION**

I submit this industrial training work entitled **“PREDICTION OF HEART RATE DISEASES**” to GITAM (Deemed To Be University), Hyderabad in partial fulfillment of the requirements for the award of the degree of “**Bachelor of Technology**” in “**Computer Science Engineering**”. I declare that it was carried out independently by me under the guidance of **Mr.** , Asst. Professor, GITAM (Deemed To Be University), Hyderabad, India.

The results embodied in this report have not been submitted to any other University or Institute for the award of any degree or diploma.

Place: HYDERABAD Siri.Anubolu

Date: 12-07-2020 221710303055

 GITAM (DEEMED TO BE UNIVERSITY)

Hyderabad-502329, India

Dated:

**CERTIFICATE**

This is to certify that the Industrial Training Report entitled **“PREDICTION OF HEART RATE DISEASES”** is being submitted by SiriAnubolu (221710303055) in partial fulfillment of the requirement for the award of **Bachelor of Technology in Computer Science and Engineering** at GITAM (Deemed To Be University), Hyderabad during the academic year 2019-20

It is faithful record work carried out by her at the **Computer Science Engineering Department**, GITAM University Hyderabad Campus under my guidance and supervision.

### Mr.Phani Kumar

Assistant Professor Professor and HOD

Department of CSE Department of CSE

iv

**ACKNOWLEDGEMENT**

Apart from my effort, the success of this internship largely depends on the encouragement and guidance of many others. I take this opportunity to express my gratitude to the people who have helped me in the successful competition of this internship.

I would like to thank respected **Dr. N. Siva Prasad,** Pro Vice Chancellor, GITAM Hyderabad and **Dr. CH. Sanjay,** Principal, GITAM Hyderabad

I would like to thank respected **Dr.Pani Kumar,** Head of the Department of Electronics and Communication Engineering for giving me such a wonderful opportunity to expand my knowledge for my own branch and giving me guidelines to present a internship report. It helped me a lot to realize of what we study for.

I would like to thank the respected faculties **Mr.** who helped me to make this internship a successful accomplishment.

I would also like to thank my friends who helped me to make my work more organized and well-stacked till the end.

Siri Anubolu

221710303055

v

**ABSTRACT**

Machine learning algorithms are used to predict the values from the data set by splitting

the data set in to train and test and building Machine learning algorithms models of higher

accuracy to predict the values is the primary task to be performed on Cereals data set My

perception of understanding the given data set has been in the view of undertaking a data

requirement of the patient and doing a full body checkup is required.

To get a better understanding and work on a strategical approach for solution of the patient,

I have adapted the view point of looking at the health condition of the patient .

vi

### **TABLE OF CONTENTS**:

**CHAPTER 1: DATA SCIENCES**

* + 1. **WHAT IS DAT SCIENCE…………………………………………….9**
    2. **NEED OF DATA SCIENCE…………………………………………..9**
    3. **USES OF DATA SCIENCE…………………………………………...9**

**CHAPTER 2: MACHINE LEARNING**

* + 1. INTRODUCTION…………………………………………………….10

2.1.2 IMPORTANCE OF MACHINE LEARNING……………………………………………..10

2.1.3 USES OF MACHINE LEARNING… 12

2.1.4TYPES OF LEARNING ALGORITHMS… 13

* + - 1. Supervised Learning………………..………………………….13
      2. Unsupervised Learning… 14

2.1.4.3 Semi Supervised Learning… 15

* 1. RELATION BETWEEN DATA MINING,MACHINE LEARNING AND DEEP LEARNING…………………………………………………………15

### CHAPTER 3:PYTHON……………………………………………………………………………………………..16

* 1. INTRODUCTOIN TO PYTHON…………………………………………16
  2. HISTORY OF PYTHON… 17
  3. FEATURES OF PYTHON… 17
  4. HOW TO SETUP PYTHON… 17
     1. Installation(using python IDLE)… 18

3.4.2 Installation(using Anaconda)… 19

* 1. PYTHON VARIABLE TYPES… 20
     1. Python Numbers… 21
     2. Python Strings… 22
     3. Python Lists… 22
     4. Python Tuples… 23
     5. Python Dictionary… 23
  2. PYTHON FUNCTION… 24
     1. Defining a Function… 24
     2. Calling a Function………………………………………….24

3.7 PYTHON USING OOP’s CONCEPTS…………………………...25

3.7.1 Class…………………………………………………….25

3.7.2 init method in class………………………………...26

### CHAPTER 4: CASE STUDY. 27

* 1. PROBLEM STATEMENT…………………………………………………………………..27
  2. DATA SET………………………………………………………………27

4.3 OBJECTIVE OF THE CASE STUDY…………………………………..27

### CHAPTER 5: IMPORTING …………..………………………………………………………………………….29

* 1. IMPORT LIBRARIES…………………………………………………...29
  2. IMPORT Data Set………………………………………………………..30
  3. UNDESTANDING THE DATA…………………………………………31

CHAPTER 6: DATA PROCESSING……………………………………………………………………………………….35

6.1 DATA PROCESSING………………………………………………………………………………35

6.2 DATA SCALING…………………………………………………………………………………….35

6.3 HANDLING DATA…………………………………………………………………………………35

CHAPTER 7: MODEL BUILDING AND EVALUATIONS………………………………………………………….36

7.1 VALIDATION OF MODELS…………………………………………………………………………………..36

7.1.1 K-Neighbors Classifier……………………………………………………………………………36

7.1.2 Decision tree Classifier…………………………………………………………………………..38

7.1.3 Random Forest Classifier……………………………………………………………………….39

### CONCLUSION… 44

viii

**LIST OF FIGURES:**

Figure 1:The process flow…………………………………………….12

Figure 2:Unsupervised learning……………………………………….14

Figure 3:Semi Supervised learning…………………………………….15

Figure 4:Python download……………………………………………..18

Figure 5:Anaconda download………………………………………….20

Figure 6:Jupyter notebook……………………………………………20

Figure 7:Defining a class………………………………………………26

Figure 8:Importing libraries……………………………………………29

Figure 9:Scaling the function…………………………………………..29

Figure 10:Sklearn import algorithms…………………………………..29

Figure 11:Importing dataset……………………………………………30

Figure 12:Checking null values………………………………………..30

Figure 13:Full dataset checking………………………………………..31

Figure 14:Visualizationn of dataset…………………………………....31

Figure 15:Correlation matrix…………………………………………..32

Figure 16:Holograms of matrix………………………………………..33

Figure 17:Checking target classes……………………………………..33

Figure 18:Bar graph of target classes………………………………….34

Figure 19:Adding dummy values……………………………………...35

Figure 20:Scaling data…………………………………………………35

Figure 21:Test/Train data……………………………………………...35

Figure 22:K-nn splitting……………………………………………….36

Figure 23:K-nn algorithm……………………………………………...37

Figure 24:K-nn line graph……………………………………………..37

Figure 25:K-nn output…………………………………………………37

Figure 26:Decision tree classifier splitting…………………………….38

Figure 27: Decision tree classifier algorithm………………………….38

Figure 28: Decision tree classifier line graph………………………….38

Figure 29: Decision tree classifier output……………………………...39

Figure 30:Random forest classifier splitting…………………………..40

Figure 31: Random forest classifier algorithm………………………...40

Figure 32: Random forest classifier bar graph………………………...40

Figure 33: Random forest classifier output……………………………43

**CHAPTER 1**

**DATA SCIENCE**

* + 1. **WHAT IS DATA SCIENCE**

Data science is the study of data. It involves developing methods of

recording, storing, and analyzing data to effectively extract useful information. The goal of

data science is to gain insights and knowledge from any type of data — both structured and

unstructured.

* + 1. **NEED OF DATA SCIENCE**

Data is one of the important features of every organization because it helps business

leaders to makedecisions based on facts, statistical numbers and trends. Due to this growing

scope of data, data science cameinto picture which is a multidisciplinary field. It uses scientific

approaches, procedure, algorithms, andframework to extract the knowledge and insight from a

huge amount of data. The extracted data can be eitherstructured or unstructured. Data science is

a concept to bring together ideas, data examination, Machine Learning, and their related

strategies to comprehend and dissect genuine phenomena with data. Data science isan extension

of various data analysis fields such as data mining, statistics, predictive analysis and many more.

Data Science is a huge field that uses a lot of methods and concepts which belongs to other fields

likeinformation science, statistics, mathematics, and computer science. Some of the techniques

utilized in DataScience encompasses machine learning, visualization, pattern recognition,

probability model, data engineering, signal processing, etc.

* + 1. **USES OF DATA SCIENCE**

Using data science, companies have become intelligent enough to push & sell products

as per customers purchasing power & interest. Here’s how they are ruling our hearts and minds:

1. Internet Search

### 2. Digital Advertisements (Targeted Advertising and re-targeting)

### 3. Recommender Systems

### 4. Image Recognition

5. Speech Recognition

6. Gaming

7. Price Comparison websites.

8. Airline Route planning

9. Fraud and risk detection

10. Delivery logistics

**CHAPTER 2**

**MACHINE LEARNING**

**2.1.1 INTRODUCTION:**

Machine Learning(ML) is the scientific study of algorithms and statistical models that computer systems use in order to perform a specific task effectively without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of Artificial Intelligence(AI).

**2.1.2 IMPORTANCE OF MACHINE LEARNING:**

Consider some of the instances where machine learning is applied: the self-driving Google car, cyber fraud detection, online recommendation engines—like friend suggestions on Facebook, Netflix showcasing the movies and shows you might like, and “more items to consider” and “get yourself a little something” on Amazon—are all examples of applied machine learning. All these examples echo the vital role machine learning has begun to take in today’s data-rich world.

Machines can aid in filtering useful pieces of information that help in major advancements, and we are already seeing how this technology is being implemented in a wide variety of industries.

With the constant evolution of the field, there has been a subsequent rise in the uses, demands, and importance of machine learning. Big data has become quite a buzzword in the last few years; that’s in part due to increased sophistication of machine learning, which helps analyze those big chunks of big data. Machine learning has also changed the way data extraction, and interpretation is done by involving automatic sets of generic methods that have replaced traditional statistical techniques.

The process flow depicted here represents how machine learning works

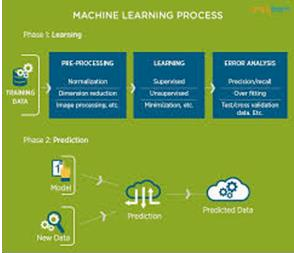


Figure 1 : The Process Flow

**2.1.3 USES OF MACHINE LEARNING:**

Earlier in this article, we mentioned some applications of machine learning. To understand the concept of machine learning better, let’s consider some more examples: web search results, real-time ads on web pages and mobile devices, email spam filtering, network intrusion detection, and pattern and image recognition. All these are by-products of applying machine learning to analyze huge volumes of data

Traditionally, data analysis was always being characterized by trial and error, an approach that becomes impossible when data sets are large and heterogeneous. Machine learning comes as the solution to all this chaos by proposing clever alternatives to analyzing huge volumes of data.

By developing fast and efficient algorithms and data-driven models for real-time processing of data, machine learning can produce accurate results and analysis.

**2.1.4 TYPES OF LEARNING ALGORITHMS:**

The types of machine learning algorithms differ in their approach, the type of data they input and output, and the type of task or problem that they are intended to solve.

**2.1.4.1 Supervised Learning :**

When an algorithm learns from example data and associated target responses that can consist of numeric values or string labels, such as classes or tags, in order to later predict the correct response when posed with new examples comes under the category of supervised learning.

Supervised machine learning algorithms uncover insights, patterns, and relationships from a labelled training dataset – that is, a dataset that already contains a known value for the target variable for each record. Because you provide the machine learning algorithm with the correct answers for a problem during training, it is able to “learn” how the rest of the features relate to the target, enabling you to uncover insights and make predictions about future outcomes based on historical data.

Examples of Supervised Machine Learning Techniques are Regression, in which the algorithm returns a numerical target for each example, such as how much revenue will be generated from a new marketing campaign.

Classification, in which the algorithm attempts to label each example by choosing between two or more different classes. Choosing between two classes is called binary classification, such as determining whether or not someone will default on a loan. Choosing between more than two classes is referred to as multiclass classification.

**2.1.4.2 Unsupervised Learning:**

When an algorithm learns from plain examples without any associated response, leaving to the algorithm to determine the data patterns on its own. This type of algorithm tends to restructure the data into something else, such as new features that may represent a class or a new series of uncorrelated values. They are quite useful in providing humans with insights into the meaning of data and new useful inputs to supervised machine learning algorithms.

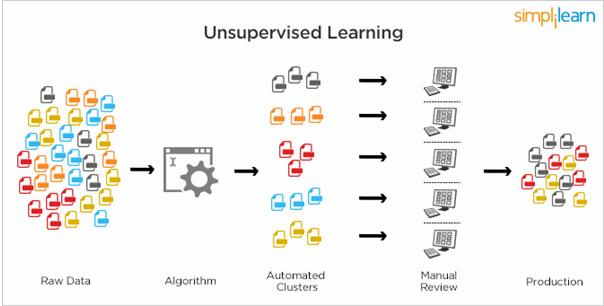


Figure 2 : Unsupervised Learning

Popular techniques where unsupervised learning is used also include self-organizing maps, nearest neighbor mapping, singular value decomposition, and k-means clustering. Basically, online recommendations, identification of data outliers, and segment text topics are all examples of unsupervised learning.

**2.1.4.3 Semi Supervised Learning:**

As the name suggests, semi-supervised learning is a bit of both supervised and unsupervised learning and uses both labeled and unlabeled data for training. In a typical scenario, the algorithm would use a small amount of labeled data with a large amount of unlabeled data.

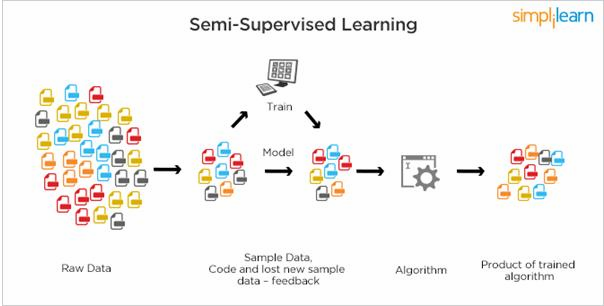


Figure 3 : Semi Supervised Learning

**2.1.5** **RELATION BETWEEN DATA MINING,MACHINE LEARNING AND DEEP LEARNING:**

Machine learning and data mining use the same algorithms and techniques as data mining, except the kinds of predictions vary. While data mining discovers previously unknown patterns and knowledge, machine learning reproduces known patterns and knowledge—and further automatically applies that information to data, decision-making, and actions.

Deep learning, on the other hand, uses advanced computing power and special types of neural networks and applies them to large amounts of data to learn, understand, and identify complicated patterns. Automatic language translation and medical diagnoses are examples of deep learning.

**CHAPTER 3**

**PYTHON**

Basic programming language used for machine learning is : PYTHON

**3.1 INTRODUCTION TO PYHTON:**

* + - Python is a high-level, interpreted, interactive and object-oriented scripting language.
    - Python is a general purpose programming language that is often applied in scripting roles
    - Python is Interpreted: Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is like PERL and PHP.
    - Python is Interactive: You can sit at a Python prompt and interact with the interpreter directly to write your programs.
    - Python is Object-Oriented: Python supports the Object-Oriented style or technique of programming that encapsulates code within objects.

**3.2 HISTORY OF PYTHON:**

* Python was developed by GUIDO VAN ROSSUM in early 1990’s
* Its latest version is 3.7 , it is generally called as python3

**3.3 FEATURES OF PYTHON:**

* + - Easy-to-learn: Python has few keywords, simple structure, and a clearly defined syntax, This allows the student to pick up the language quickly.
    - Easy-to-read: Python code is more clearly defined and visible to the eyes.
    - Easy-to-maintain: Python's source code is fairly easy-to-maintaining.
    - A broad standard library: Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
    - Portable: Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
    - Extendable: You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
    - Databases: Python provides interfaces to all major commercial databases.
    - GUI Programming: Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

**3.4 HOW TO SETUP PYTHON:**

* Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.
* The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python.

**3.4.1 Installation(using python IDLE):**

* Installing python is generally easy, and nowadays many Linux and Mac OS distributions include a recent python.
* Download python from www.python.org
* When the download is completed, double click the file and follow the instructions to install it.
* When python is installed, a program called IDLE is also installed along with it. It provides a graphical user interface to work with python.

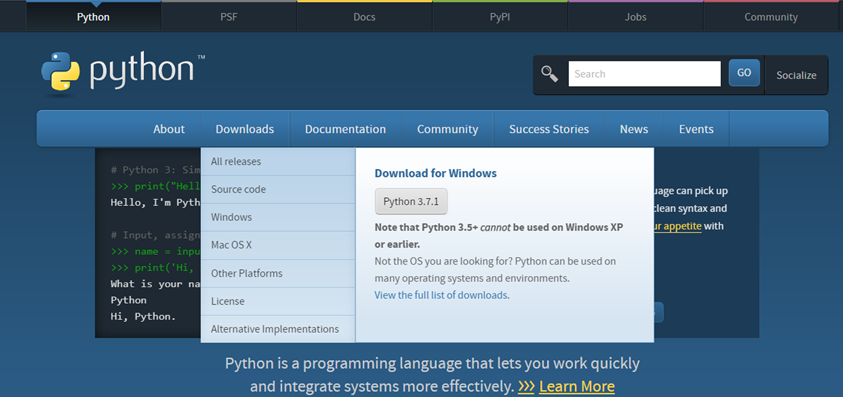


Figure 4 : Python download

**3.4.2 Installation(using Anaconda):**

* Python programs are also executed using Anaconda.
* Anaconda is a free open source distribution of python for large scale data processing, predictive analytics and scientific computing.
* Conda is a package manager quickly installs and manages packages.
* In WINDOWS:
* In windows
* Step 1: Open Anaconda.com/downloads in web browser.
* Step 2: Download python 3.4 version for (32-bitgraphic installer/64 -bit graphic installer)
* Step 3: select installation type( all users)
* Step 4: Select path(i.e. add anaconda to path & register anaconda as default python 3.4) next click install and next click finish
* Step 5: Open jupyter notebook ( it opens in default browser)

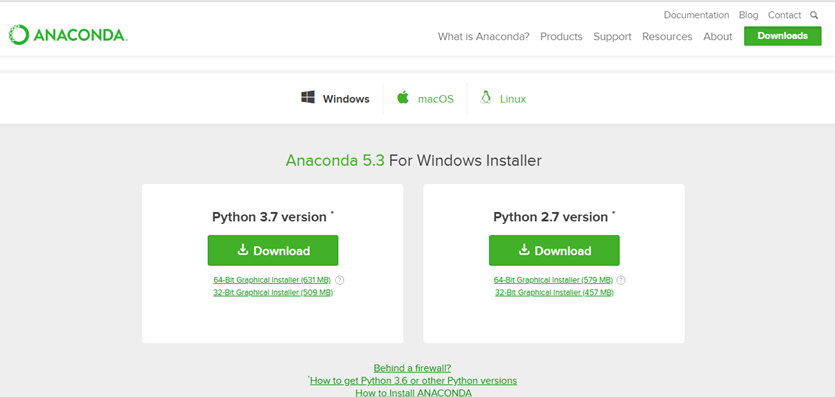


Figure 5 : Anaconda download

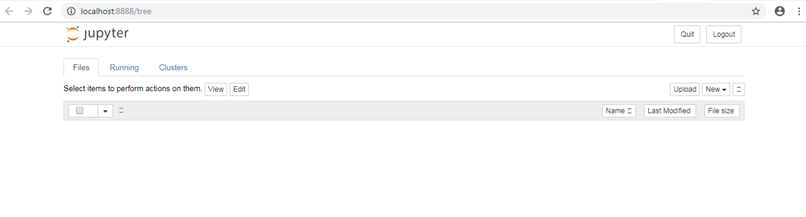
****

Figure 6 : Jupyter notebook

**3.5 PYTHON VARIABLE TYPES:**

* Variables are nothing but reserved memory locations to store values. This means that when you create a variable you reserve some space in memory.
* Variables are nothing but reserved memory locations to store values.
* Based on the data type of a variable, the interpreter allocates memory and decides what can be stored in the reserved memory.
* Python variables do not need explicit declaration to reserve memory space. The declaration happens automatically when you assign a value to a variable.
* Python has various standard data types that are used to define the operations possible on them and the storage method for each of them.
* Python has five standard data types –
  + Numbers
  + Strings
  + Lists
  + Tuples
  + Dictionary

**3.5.1 Python Numbers:**

* + - Number data types store numeric values. Number objects are created when you assign a value to them.
    - Python supports four different numerical types − int (signed integers) long (long integers, they can also be represented in octal and hexadecimal) float (floating point real values) complex (complex numbers).

**3.5.2 Python Strings:**

* Strings in Python are identified as a contiguous set of characters represented in the quotation marks.
* Python allows for either pairs of single or double quotes.
* Subsets of strings can be taken using the slice operator ([ ] and [:] ) with indexes starting at 0 in the beginning of the string and working their way from -1 at the end.
* The plus (+) sign is the string concatenation operator and the asterisk (\*) is the repetition operator.

**3.5.3 Python Lists:**

* + - Lists are the most versatile of Python's compound data types.
    - A list contains items separated by commas and enclosed within square brackets ([]).
    - To some extent, lists are similar to arrays in C. One difference between them is that all the items belonging to a list can be of different data type.
    - The values stored in a list can be accessed using the slice operator ([ ] and [:]) with indexes starting at 0 in the beginning of the list and working their way to end -1.
    - The plus (+) sign is the list concatenation operator, and the asterisk (\*) is the repetition operator.

**3.5.4 Python Tuples:**

* A tuple is another sequence data type that is similar to the list.
* A tuple consists of a number of values separated by commas. Unlike lists, however, tuples are enclosed within parentheses.
* The main differences between lists and tuples are: Lists are enclosed in brackets ( [ ] ) and their elements and size can be changed, while tuples are enclosed in parentheses ( ( ) ) and cannot be updated.
* Tuples can be thought of as read-only lists.
* For example − Tuples are fixed size in nature whereas lists are dynamic. In other words, a tuple is immutable whereas a list is mutable. You can't add elements to a tuple. Tuples have no append or extend method. You can't remove elements from a tuple. Tuples have no remove or pop method.
  + 1. **Python Dictionary:**
* Python's dictionaries are kind of hash table type. They work like associative arrays or hashes found in Perl and consist of key-value pairs. A dictionary key can be almost any Python type, but are usually numbers or strings. Values, on the other hand, can be any arbitrary Python object.
* Dictionaries are enclosed by curly braces ({ }) and values can be assigned and accessed using square braces ([]).
* You can use numbers to "index" into a list, meaning you can use numbers to find out what's in lists. You should know this about lists by now, but make sure you understand that you can only use numbers to get items out of a list.
* What a dict does is let you use anything, not just numbers. Yes, a dict associates one thing to another, no matter what it is.

**3.6 PYTHON FUNCTION:**

**3.6.1 Defining a Function:**

You can define functions to provide the required functionality. Here are simple rules to define a function in Python. Function blocks begin with the keyword def followed by the function name and parentheses (i.e.()).

Any input parameters or arguments should be placed within these parentheses. You can also define parameters inside these parentheses

The code block within every function starts with a colon (:) and is indented. The statement returns [expression] exits a function, optionally passing back an expression to the caller. A return statement with no arguments is the same as return None.

**3.6.2 Calling a Function:**

Defining a function only gives it a name, specifies the parameters that are to be included in the function and structures the blocks of code. Once the basic structure of a function is finalized, you can execute it by calling it from another function or directly from the Python prompt.

* 1. **PYTHON USING OOP’s CONCEPTS:**

**3.7.1 Class:**

* Class: A user-defined prototype for an object that defines a set of attributes that characterize any object of the class. The attributes are data members (class variables and instance variables) and methods, accessed via dot notation.
* Class variable: A variable that is shared by all instances of a class. Class variables are defined within a class but outside any of the class's methods. Class variables are not used as frequently as instance variables are.
* Data member: A class variable or instance variable that holds data associated with a class and its objects.
* Instance variable: A variable that is defined inside a method and belongs only to the current instance of a class.
* Defining a Class:
  + We define a class in a very similar way how we define a function.
  + Just like a function ,we use parentheses and a colon after the class name(i.e. ():) when we define a class. Similarly, the body of our class is indented like a functions body is.



Figure 7 : Defining a Class

**3.7.2 \_\_init\_\_ method in Class:**

* The init method — also called a constructor — is a special method that runs when an instance is created so we can perform any tasks to set up the instance.
* The init method has a special name that starts and ends with two underscores:\_\_init\_\_().

**CHAPTER 4**

**CASE STUDY**

**4.1 PROBLEM STATEMENT:**

To predict the heart rate diseases using three machine Learning algorithms:

1. K-Neighbors Classifier
2. Decision Tree Classifier
3. Random Forest Classifier

**4.2 DATA SET:**

The given data set consists of the following parameters:

0. age

1. sex

2. cp

3. trestbps

4. chol

5. fbs

6. restecg

7. thalach

8. exang

9. oldpeak

10. slope

11. ca

12. thal

13. target

**4.3 OBJECTIVE OF THE CASE STUDY:**

**Heart Diseases** describes a range of conditions that affect your heart. Diseases under the heart disease include blood vessel diseases, such as coronary artery disease, heart rhythm problems (arrhythmias) and heart defects you’re born with (congenital heart defects), among others.

The term “heart disease” is often used interchangeably with the term “cardiovascular disease”. Cardiovascular disease generally refers to conditions that involve narrowed or blocked blood vessels that can lead to a heart attack, chest pain (angina) or stroke. Other heart conditions, such as those that affect your heart’s muscle, valves or rhythm, also are considered forms of heart disease. Heart disease is one of the biggest causes of morbidity and mortality among the population of the world. Prediction of cardiovascular disease is regarded as one of the most important subjects in the section of clinical data analysis. The amount of data in the healthcare industry is huge. Data mining turns the large collection of raw healthcare data into information that can help to make informed decisions and predictions.

**CHAPTER 5**

## 5.1 Import libraries

Let's first import all the necessary libraries. I'll use numpy and pandas to start with. For visualization, I will use pyplot subpackage of matplotlib, use rcParams to add styling to the plots and rainbow for colors. For implementing Machine Learning models and processing of data, I will use the sklearn library.

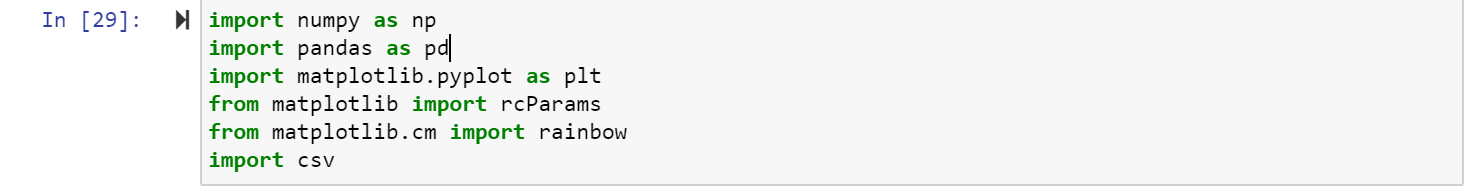


Figure8: importing libraries

For processing the data, I'll import a few libraries. To split the available dataset for testing and training, I'll use the train\_test\_split method. To scale the features, I am using StandardScaler.

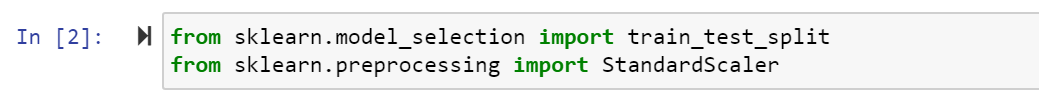


Figure9

Next, I'll import all the Machine Learning algorithm

1.K Neighbors Classifier

2.Decision Tree Classifier

3.Random Forest Classifier

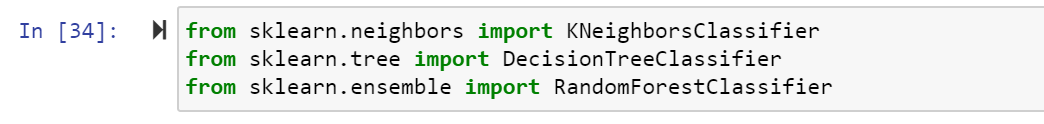


Figure 10

## 5.2 Import dataset

## Now that we have all the libraries we will need, I can import the dataset and take a look at it. The dataset is stored in the file df.csv. I'll use the pandas read\_csv method to read the dataset.

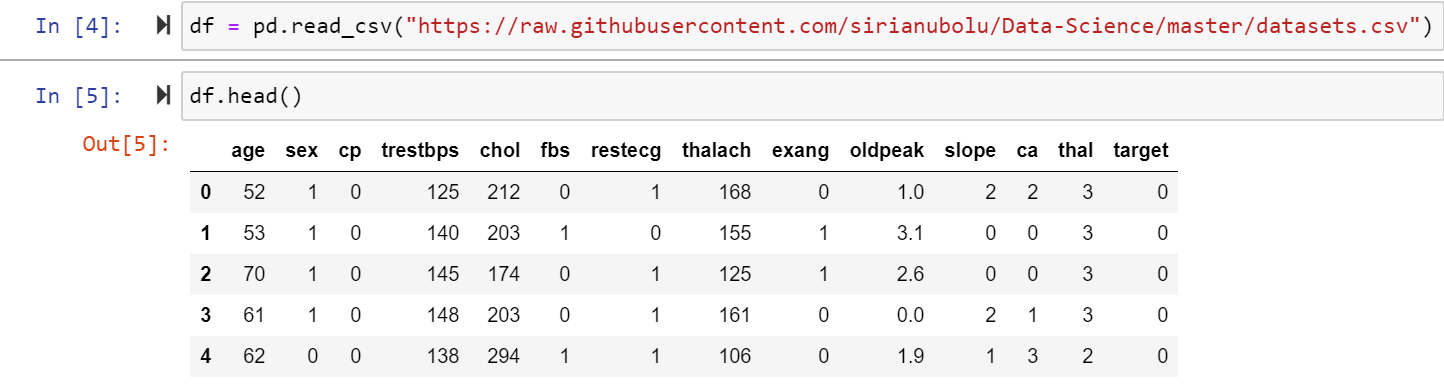


Figure11

Lets check if there are null values in the data set:

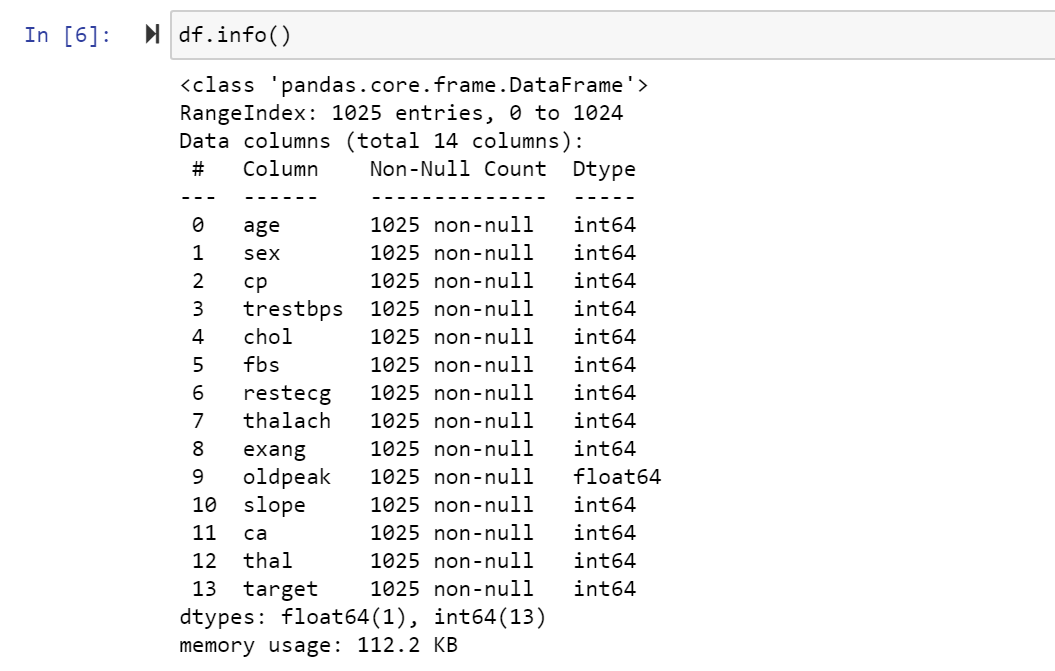


Figure12

The dataset has a total of 1025 rows and there are no missing values. There are a total of 13 features along with one target value which we wish to find.

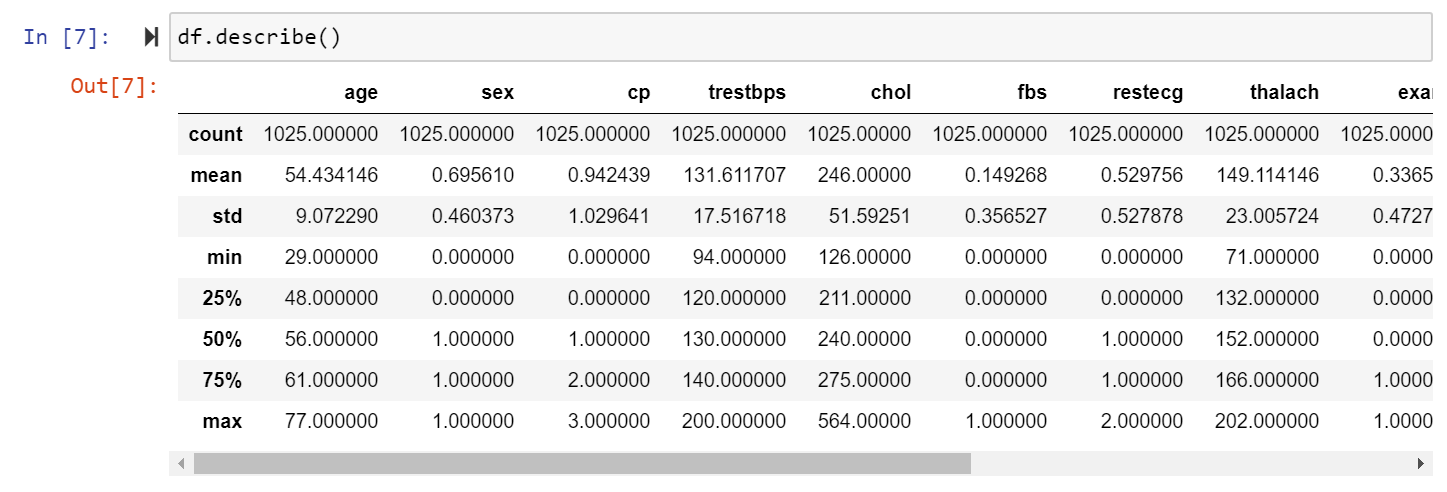
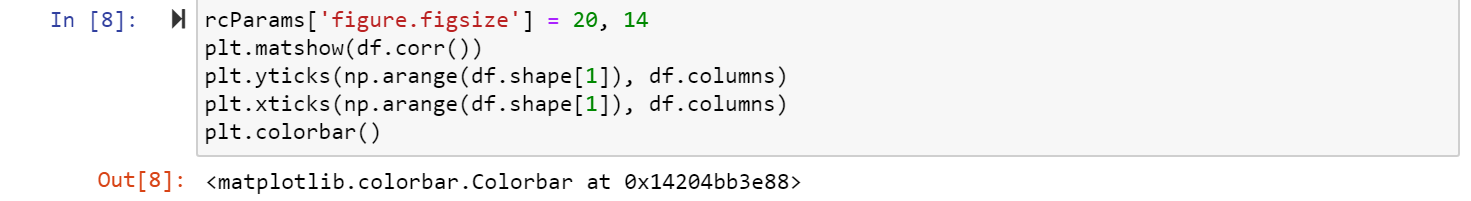


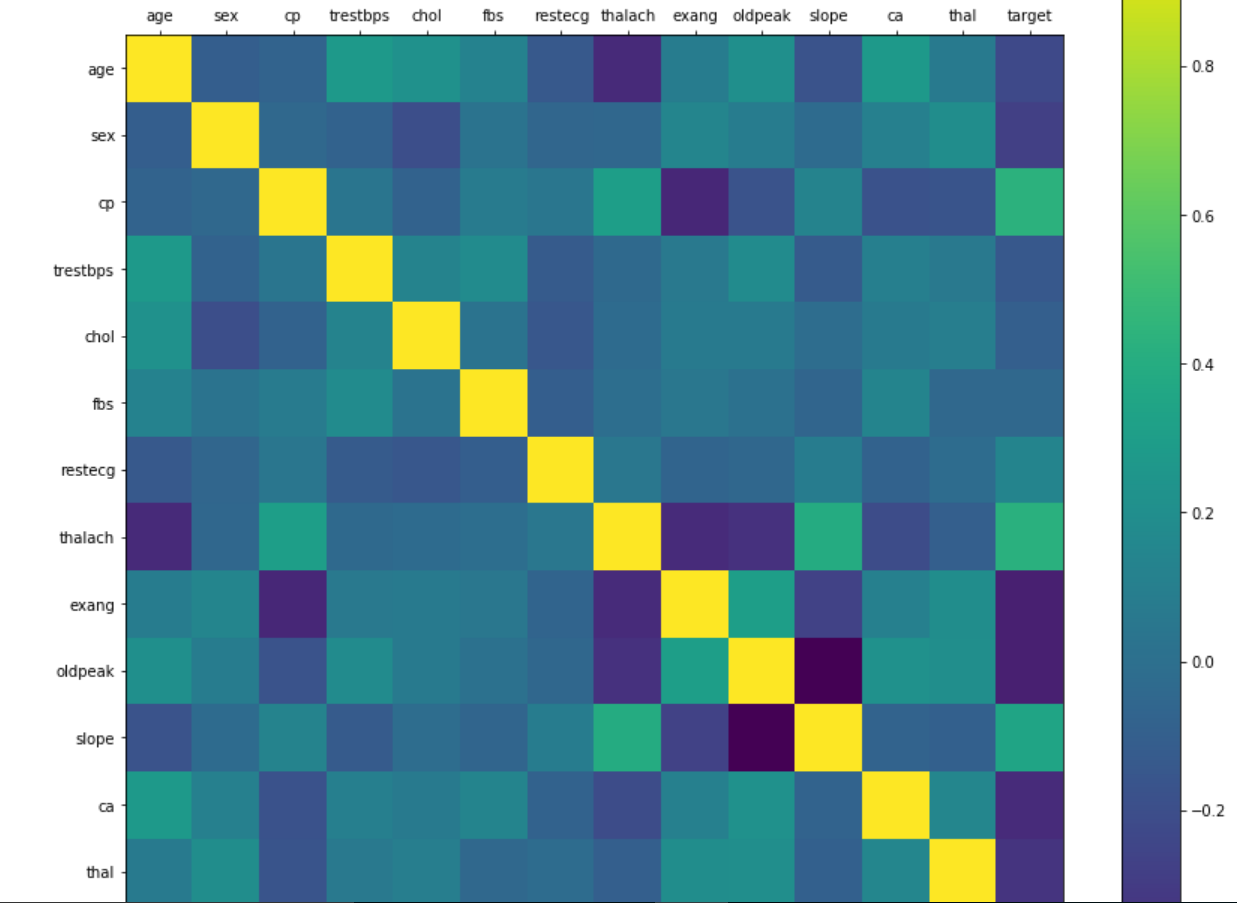
Figure13

**5.3 UNDESTANDING THE DATA**

Now, we can use visualizations to better understand our data and then look at any processing we might want to do.

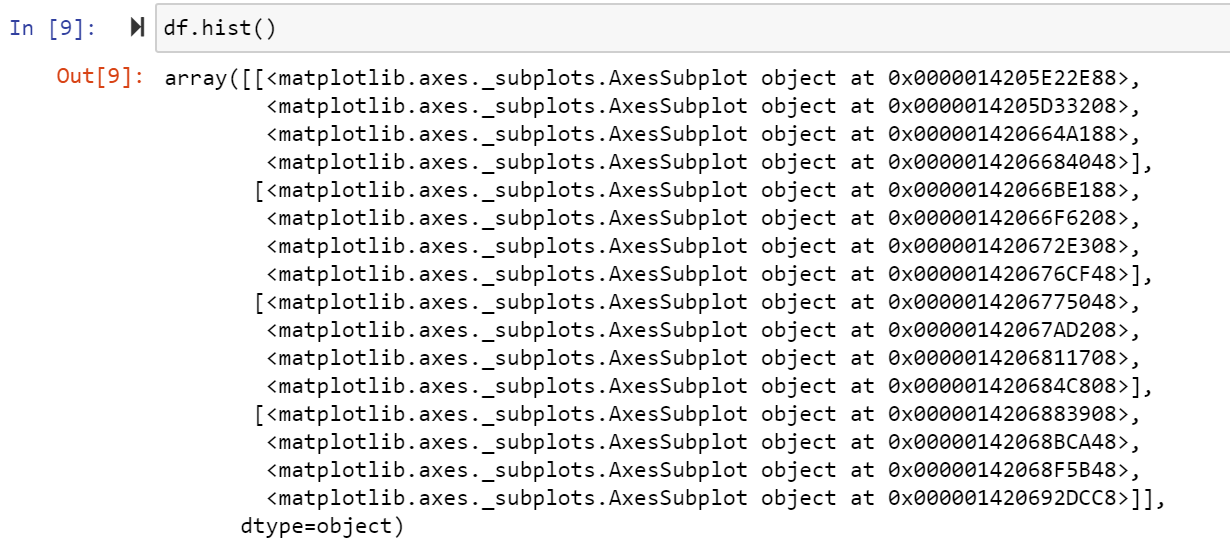


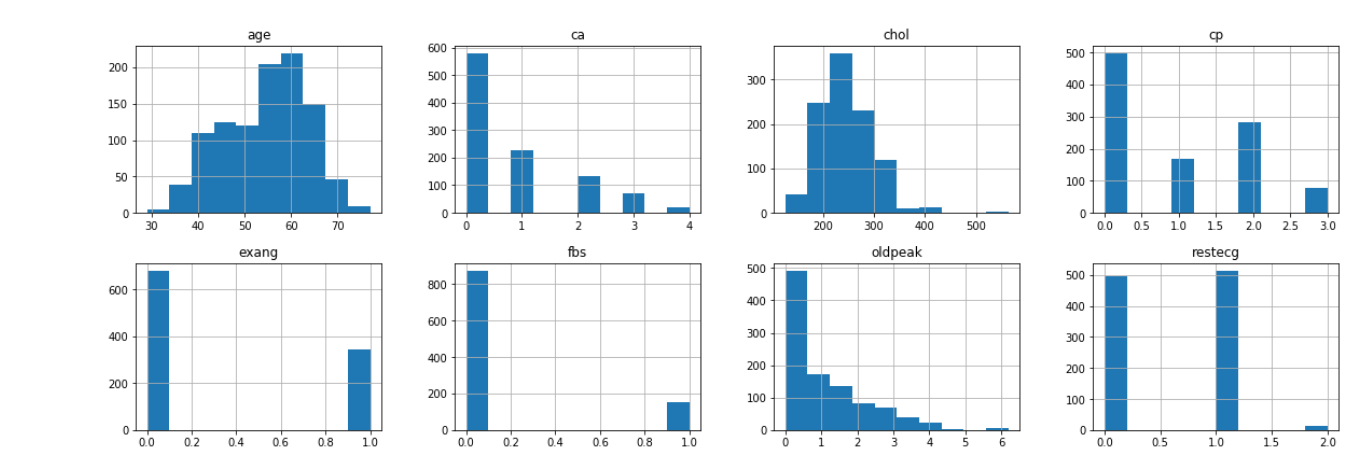
**Figure14**



**Figure15**

Taking a look at the correlation matrix above, we can say that a few features have negative correlation with the target value while some have positive.Now,histograms for each variable.





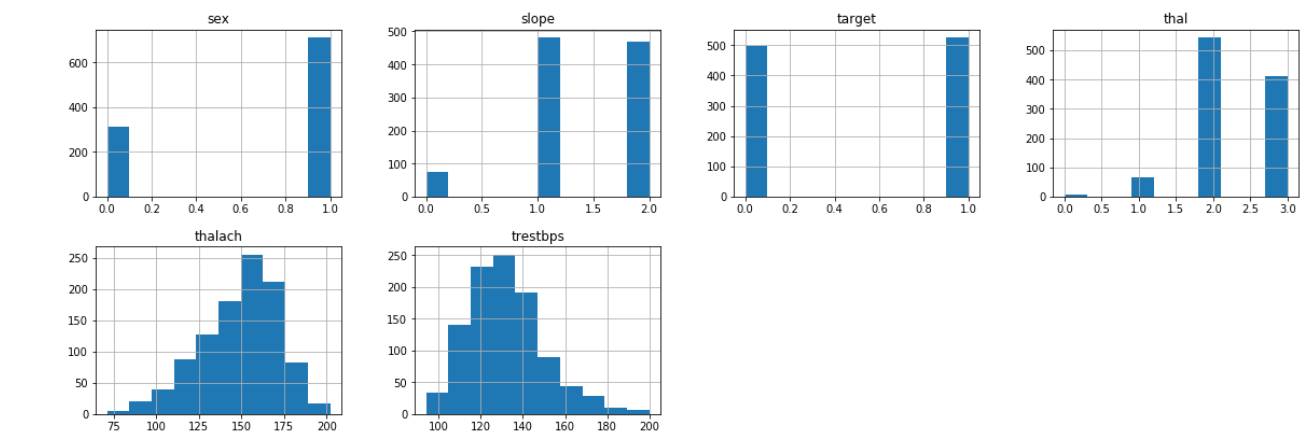


Figure 16

Taking a look at the histograms above, I can see that each feature has a different range of

distribution. Thus, using scaling before our predictions should be of great use. Also, the

categorical features do stand out.

It's always a good practice to work with a dataset where the target classes are of approximately

equal size. Thus, let's check for the same.

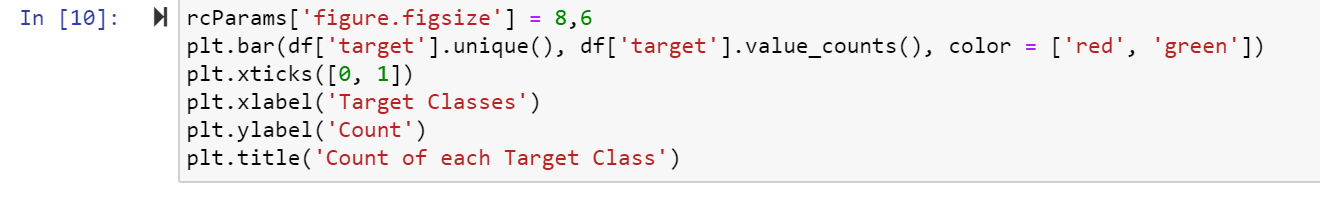


Figure 17

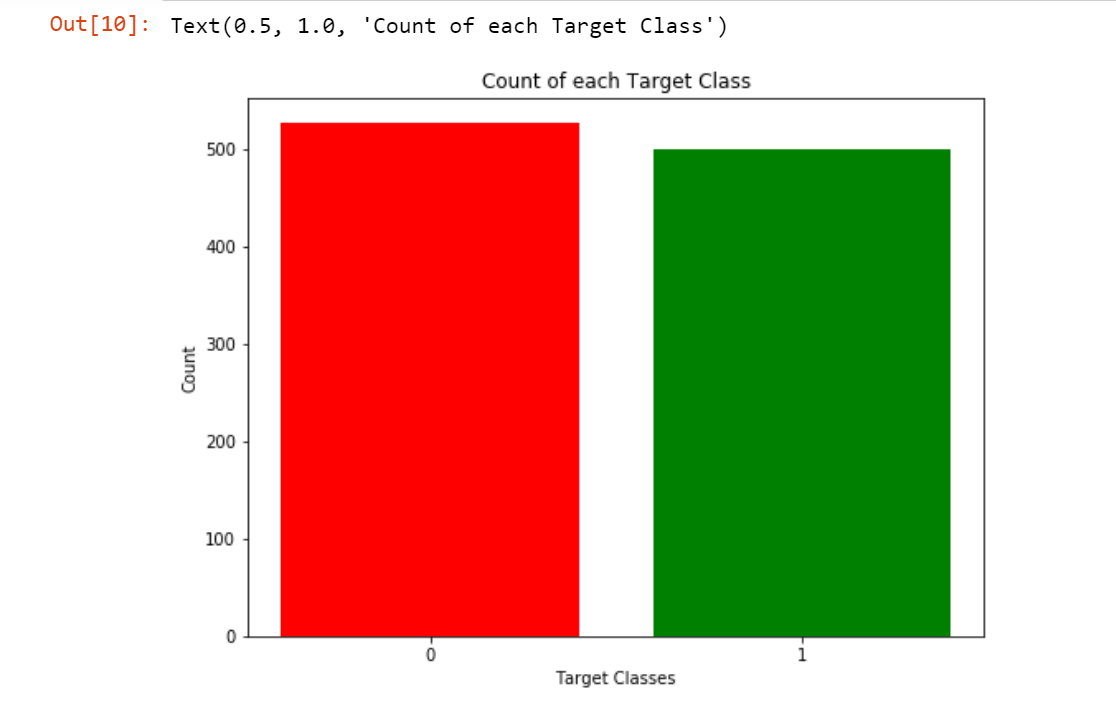


Figure 18

The two classes are not exactly 50% each but the ratio is good enough to continue without

dropping/increasing our data.

**CHAPTER 6**

**DATA PROCESSING**

**6.1 DATA PROCESSING**

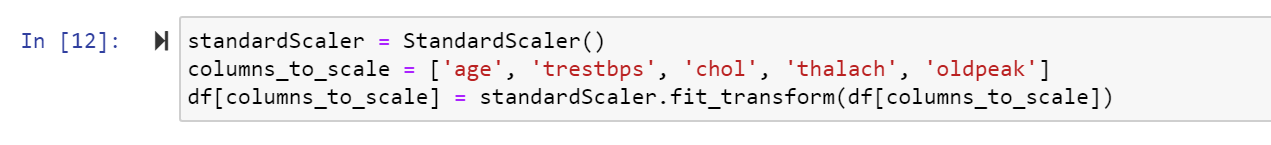
After exploring the dataset, I observed that I need to convert some categorical variables into dummy variables and scale all the values before training the Machine Learning models. First, I'll use the get\_dummies method to create dummy columns for categorical variables.



**Figure19**

**6.2 DATA SCALING**

Now, I will use the StandardScaler from sklearn to scale my dataset.



**Figure20**

**6.3 HANDLING DATA**

I'll now import train\_test\_split to split our dataset into training and testing datasets. Then, I'll import all Machine Learning models I'll be using to train and test the data.

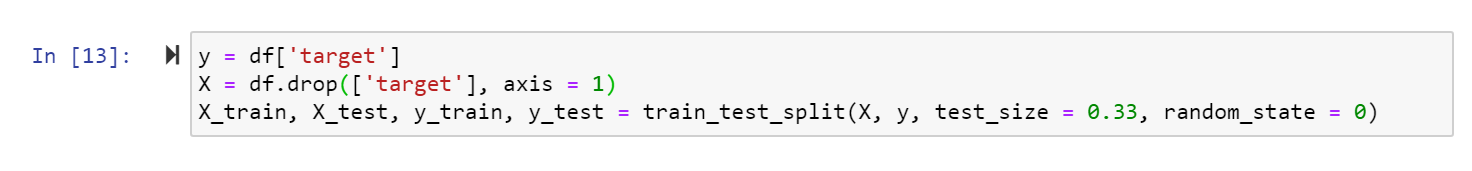


Figure21

**CHAPTER 7**

**MODEL BUILDING AND EVALUATION**

**7.1 VALIDATION OF MODELS**

7.1.1 K-Neighbours Classifiers

The ​k ​-nearest neighbors algorithm (​k ​-NN) is a non-parametric method proposed by Thomas

Cover used for classification and regression.​[1]​ In both cases, the input ​consists of the ​k closest

training examples in the feature space. The output depends on whether ​k ​ -NN is used for

classification or regression. In ​k-NN classification ​ , the output is a class membership. An object

is classified by a plurality vote of its neighbors, with the object being assigned to the class most

common among its ​k nearest neighbors (​k ​ is a positive integer, typically small). If ​k ​ = 1, then

the object is simply assigned to the class of that single nearest neighbor. The classification score

varies based on different values of neighbors that we choose. Thus, I'll plot a score graph for

different values of K (neighbors) and check when do I achieve the best score.

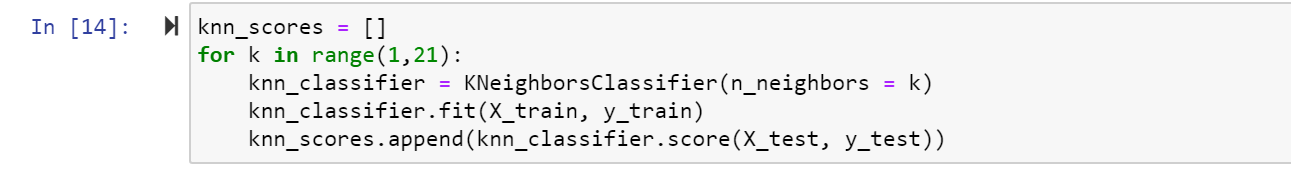


Figure22

I have the scores for different neighbor values in the array knn\_scores. I'll now plot it and see

for which value of K did I get the best scores.

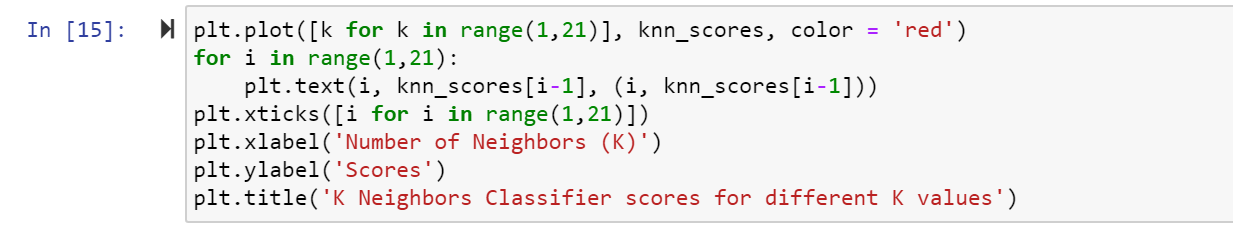


Figure 23

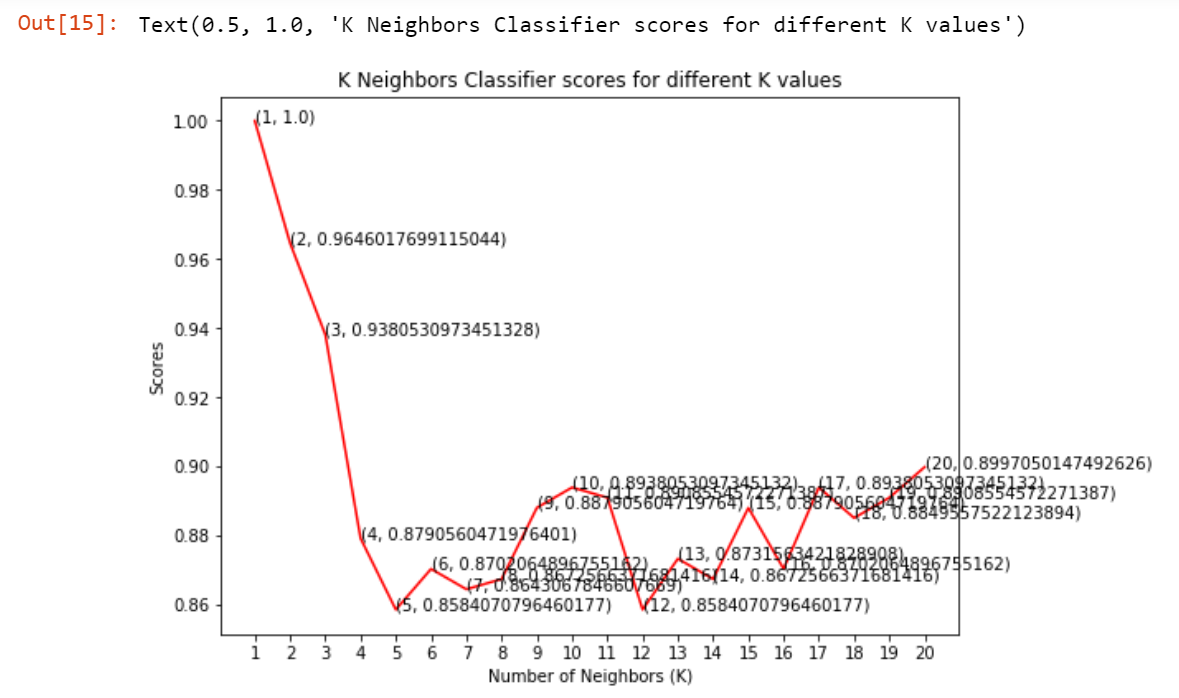


Figure 24

From the plot above, it is clear that the maximum score achieved was 1.0 for the 7 neighbors.

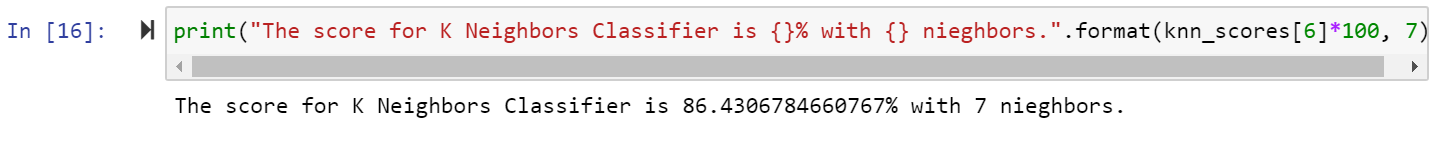


Figure25

7.1.2 DECISION TREE CLASSIFIER

A decision tree is a decision support tool that uses a tree-like graph or model of decisions and

their possible consequences, including chance event outcomes, resource costs, and utility. It is

one way to display an algorithm that only contains conditional control statements.

A decision tree is a flowchart-like structure in which each internal node represents a “test” on an

attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of

the test, and each leaf node represents a class label (decision taken after computing all attributes).

The paths from root to leaf represent classification rules. The Decision Tree Classifier to model

the problem at hand. I'll vary between a set of max\_features and see which returns the best

accuracy.

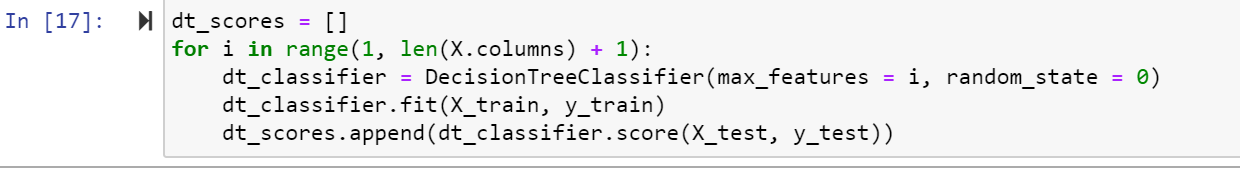


Figure26

I selected the maximum number of features from 1 to 30 for split. Now, let's see the scores for

each of those cases.

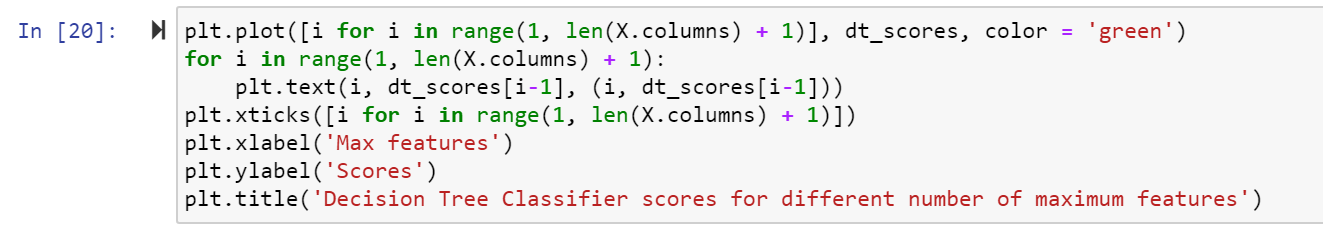


Figure27

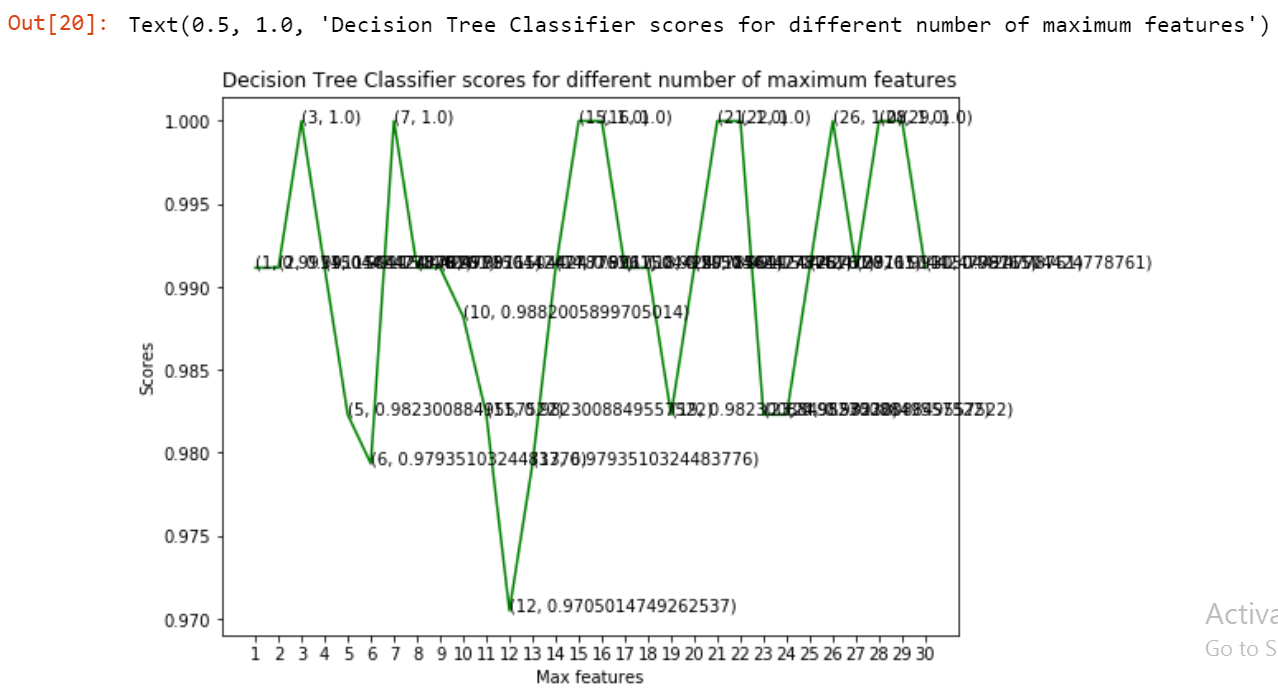


Figure28

The model achieved the best accuracy at three values of maximum features 3,7,26.

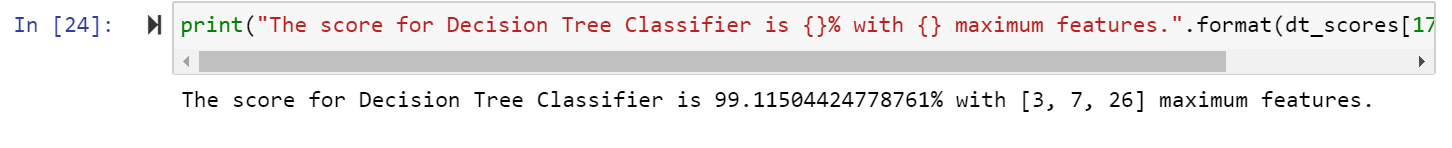


Figure29

7.1.3 RANDOM FOREST CLASSIFIER

Random forests or random decision forests are an ensemble learning method for classification,

regression and other tasks that operate by constructing a multitude of decision trees at training

time and outputting the class that is the mode of the classes (classification) or mean prediction

(regression) of the individual trees. Random decision forests correct for decision trees' habit of

overfitting to their training set. The first algorithm for random decision forests was created by

Tin Kam Ho using the random subspace method, which, in Ho's formulation, is a way to

implement the "stochastic discrimination" approach to classification proposed by Eugene

Kleinberg. We use the ensemble method, Random Forest Classifier, to create the model and vary

the number of estimators to see their effect.

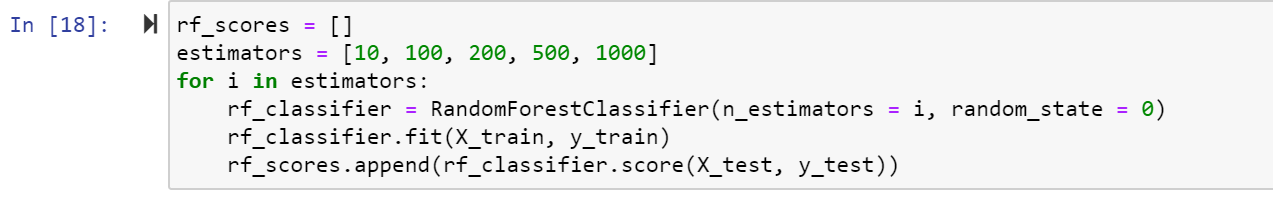
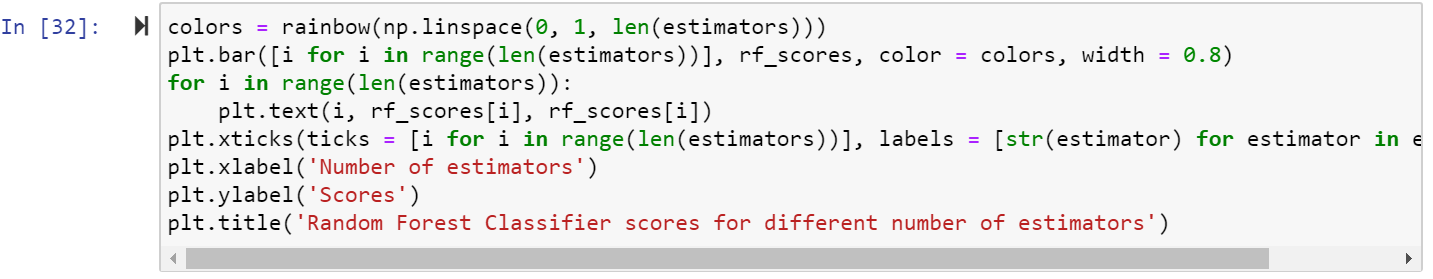


Figure30



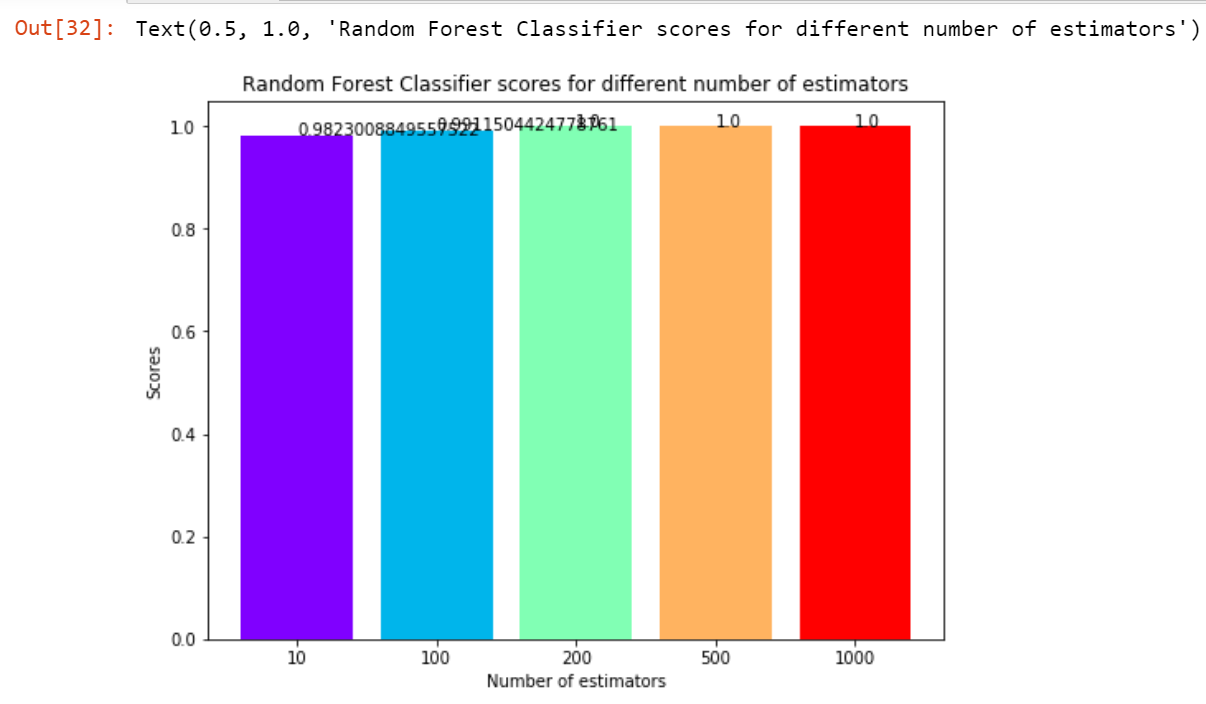


Figure32

The maximum score is achieved when the total estimators are 100,500 or 1000.

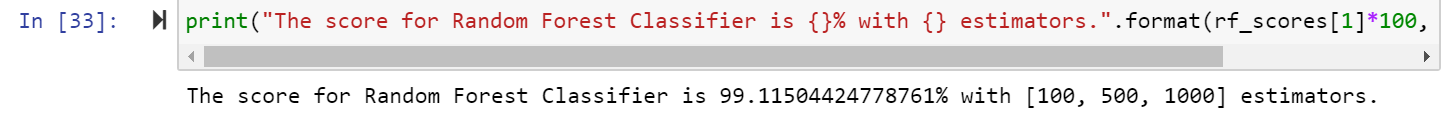


Figure33

**CONCLUSION**

In this project, I used Machine Learning to predict whether a person is suffering from a heart

disease. After importing the data, I analysed it using plots. Then, I did generated dummy

variables for categorical features and scaled other features. I then applied three Machine

Learning algorithms, K Neighbors Classifier,Decision Tree Classifier and Random Forest

Classifier. I varied parameters across each model to improve their scores. In the end, Decision

Tree Classifier and Random Forest Classifier achieved the highest score of

99.11504424778761% with 3 estimators.

**SIRI.ANUBOLU**

**PIN NUMBER: 221710303055**

**DATA SCIENCES**