**Data Science Project on Heart Disease**

Presented by: -

Donthi Venkata Aravind

* **TABLE OF CONTENT**

|  |  |
| --- | --- |
| **Sno** | **Content** |
| 1 | Table of Figures (or) Graphs |
| 2 | Abstract |
| 3 | Chapter’s |
| 4 | Complete Code  (Attached ipnyb file) |
| 5 | Conclusion |

* **TABLE OF FIGURES/GRAPHS**

|  |  |
| --- | --- |
| **Figure number** | **Page No** |
| Figure-1 | 15 |
| Figure-2 | 17 |
| Figure-31 | 18 |
| Figure-4 | 18 |
| Figure-5 | 19 |
| Figure-6 | 20 |
| Figure-7 | 21 |
| Figure-8 | 22 |
| Figure-9 | 23 |
| Figure-10 | 24 |
| Figure-11 | 25 |
| Figure-12 | 26 |
| Figure-13 | 34 |
| Figure-14 | 36 |
| Figure-15 | 37 |
| Figure-16 | 38 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Sno** | **Chapter** | **Topic covered** | **Page No** |
| 1 |  | Abstract | 4-4 |
| 2 | Chapter-1 | Datasets used in Analysis | 5-5 |
| 3 | Chapter-2 | Conditions applied on dataset for identifying disease | 6-6 |
| 4 | Chapter-3 | Creating and training a model | 7-7 |
| 5 | Chapter-4 | Importing Data Visualization libraries in python | 8-10 |
| 6 | Chapter-5 | Getting Familiar with the procedure | 11-31 |
| 7 | Chapter-6 | Test Cases | 32-38 |
| 8 |  | Conclusion | 39-39 |

**INDEX**

* **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. Nowadays, in healthcare industry, data analysis can save lives by improving the medical diagnosis. And with the huge development in software engineering, different data mining tools are available for researchers, and used to conduct studies and experiments.

Experiments with some python libraries can be used in prediction of a heart diseased person using trained model. A comparative analytical approach will be done for improving prediction accuracy in heart disease and to show its utility to predict disease at an early stage. The results of the study indicate that ensemble techniques, such as bagging and boosting, are effective in improving the prediction accuracy of weak classifiers, and exhibit satisfactory performance in identifying risk of heart disease.

The performance of the process will be further enhanced with a feature selection implementation, and the results significant improvement will be verified with prediction accuracy. + +

* **Chapter-1**

**Datasets used in this Analysis**

I have used open-source data-set which is available online it is not so huge data-set but it is very convenient for the beginners to analyze data and learn new things in this data set we have used some of the shortcuts. To get to know about those I am displaying them here.

age: age

sex: 1: male,

0: female

cp: chest pain type,

1: typical angina,

2: atypical angina,

3: non- anginal pain,

4: asymptomatic

trestbps: resting blood pressure

chol: serum cholestoral in mg/dl

fbs: fasting blood sugar > 120 mg/dl

restecg: resting electrocardiographic results (values 0,1,2)

thalach: maximum heart rate achieved

exang: exercise induced angina

oldpeak: oldpeak = ST depression induced by exercise relative to rest

slope: the slope of the peak exercise ST segment

ca: number of major vessels (0-3) colored by flourosopy

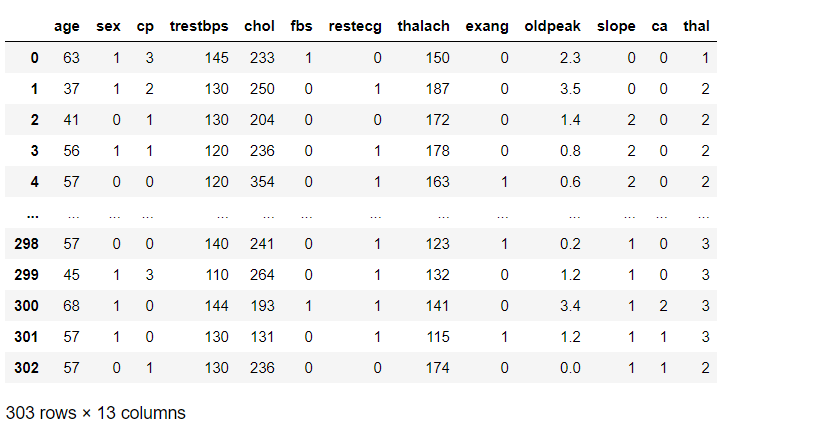
thal: thal: 3 = normal; 6 = fixed defect; 7 = reversable defect

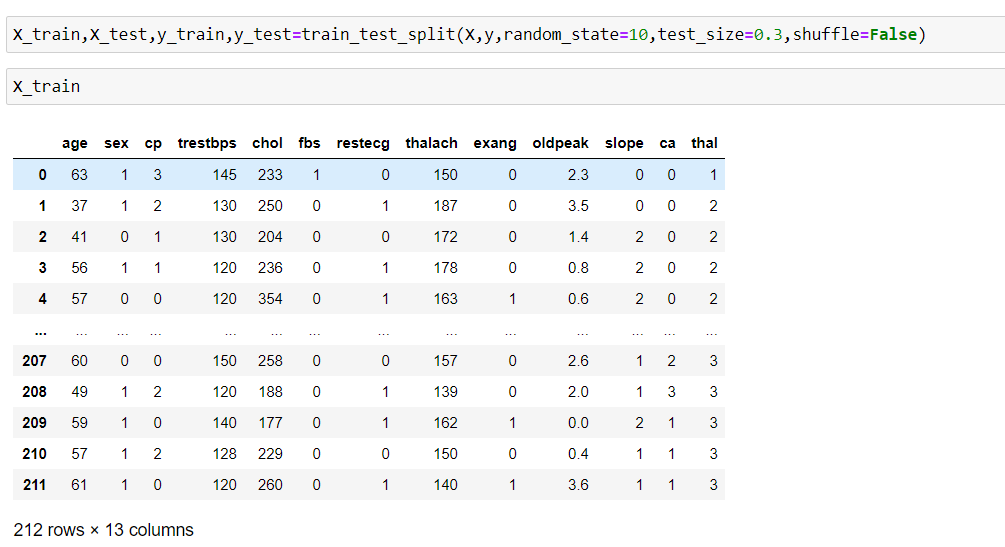
* **Chapter-2: - Conditions applied on dataset for identifying disease**

In this project we have used two important libraries and functions to determine whether the person will suffer from heart disease or not using :-

* Sklearn Model
* Decision Tree Classifier

Sklearn is the library used in this project to train and test the module and to create a module we have used “Decision Tree Classifier”.

**Example: -** (Before training the model)

**Example: -** (After training the model)

This is the major part in this project we need to analyze the data and we have to predict whether the person will suffer from Heart Disease or not

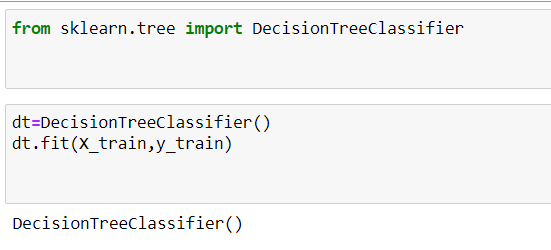
According to my analysis from past testing and on my observations “If our accuracy on test is in between 70%-92% our test will be successful.”

* **Chapter-3: - Creating and training a model**

To create a model, we have to import some functions from sklearn

* We use “Decision Tree Classifier” to create a model
* We train and test the data with help of respective functions used in the project

Here’s a shot how to create a module

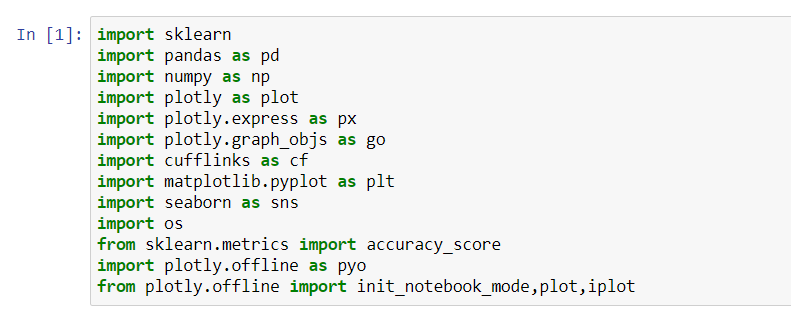


To train the model we have to use some more functions which will be discussed in later chapters

Decision Trees are a class of very powerful Machine Learning model cable of achieving high accuracy in many tasks while being highly interpretable. What makes decision trees special in the realm of ML models is really their clarity of information representation. The “knowledge” learned by a decision tree through training is directly formulated into a hierarchical structure. This structure holds and displays the knowledge in such a way that it can easily be understood, even by non-experts.

* **Chapter-4**

**Importing Data Visualization libraries in python**

****

* import pandas as pd

Pandas is a library that we all have to install in our machine locally during python installation or we have to use **pip install pandas.** Simply import the library the current namespace, but rather than using name ‘pandas’ it can be instructed as pd for our convenience.

* import numpy as np

NumPy is a general Python package that stands for ‘Numerical Python’. It is the core library for scientific computing, which contains a powerful n-dimensional array object. It is a general purpose array-processing package.

* import matplotlib.pyplot as plt

matplotlib.pyplot is a plotting library used for 2D graphics in python programming language. It can be used in python scripts, shell, web application servers and other graphical user interface toolkits.

One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

* import sklearn

sklearn is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction. Scikit-learn comes loaded with a lot of features such as **Supervised learning algorithms, Cross-validation, Unsupervised learning algorithms, Various toy datasets, Feature extraction**

* **import plotly as plot**

Plotly is known for developing and providing online analytics, statistics and graphing tools for individuals or companies. It also develops/provides scientific graphing libraries for Arduino, Julia, MATLAB, Perl, Python, R and REST.

* import plotly.express as px & import plotly.graph\_objs as go
* import plotly.offline as pyo
* from plotly.offline import init\_notebook\_mode,plot,iplot

Plotly Express is a new high-level Python visualization library: it’s a wrapper for Plotly.py that exposes a simple syntax for complex charts. It was specifically designed to have a terse, consistent and easy-to-learn API: with just a single import, you can make richly interactive plots in just a single function call, including faceting, maps, animations, and trendlines. It comes with on-board datasets, color scales and themes, and just like Plotly.py, Plotly Express is totally free: with its permissive open-source MIT license, you can use it however you like. Best of all, Plotly Express is fully compatible with the rest of Plotly ecosystem: use it in your Dash apps, export your figures to almost any file format using Orca, or edit them in a GUI with the Jupyter Lab Chart Editor!

* import cufflinks as cf

Cufflinks is another library that connects the Pandas data frame with Plotly enabling users to create visualizations directly from Pandas. The library binds the power of Plotly with the flexibility of Pandas for easy plotting.

* import seaborn as sns

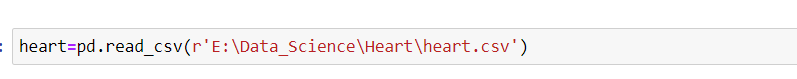
Seaborn aims to make visualization a central part of exploring and understanding data. Its dataset-oriented plotting functions operate on data frames and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots. The main idea of Seaborn is that it provides high-level commands to create a variety of plot types useful for statistical data exploration, and even some statistical model fitting.

* import os

Since the os module provides access to system features, the actual options that are available will depend on the underlying operating system. Many of the os options are available for all operating systems, but there are several which are available for UNIX (and most UNIX-based systems, including Linux and the Mac OS), but not for Windows, and some which are available only for certain flavors of UNIX. Since the available commands are system-dependent, when you create a Python program that uses os commands, you will probably want to include a check to find out which os module you’re dealing with, using the os.name command

All the procedure will be discussed in next chapter.

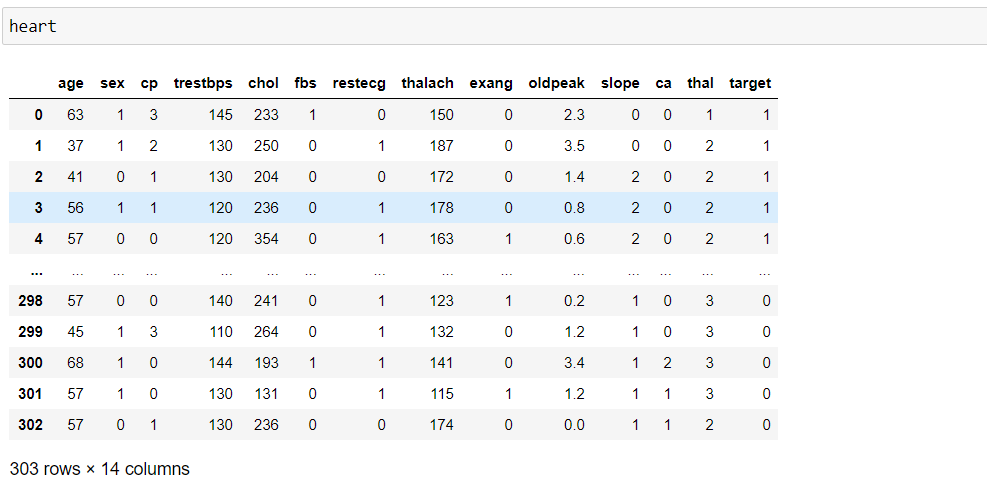
* **Chapter -5 Getting Familiar with the procedure**



We are declaring a variable as “heart” and into that heart variable we are trying to read the csv file which is taken from the open source

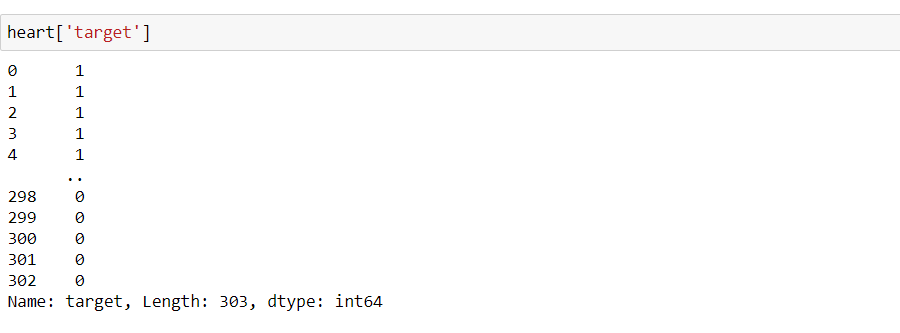
In my PC I have saved my file in Local Disk: E and in Data Science folder in Heart subfolder I have saved my CSV file

I have used ‘r’ at the beginning because in any programming languages ‘\’ will be considered as **Escape Characters** so as to avoid this we will use r to avoid the escape character.



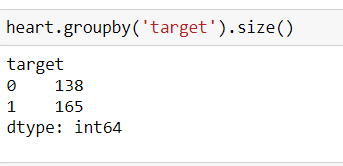
To check whether the data is imported into the variable or not

From this We can say that the data which is available in our CSV file is copied to heart variable.



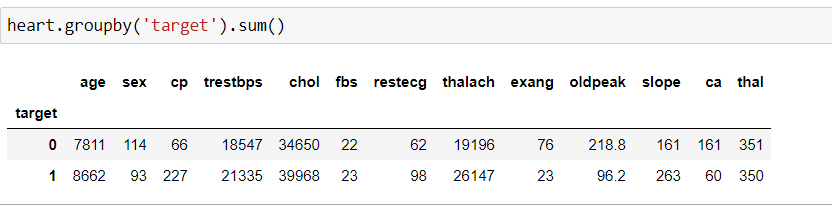
In this cell we have displayed that:

Already we have imported all the data from the CSV file into heart and in that dataset, we have a column called “target”. We are trying to display all the data which corresponds to target column.



We are taking same column name “target” in this code we are trying to group the content based on the value like 0’s-138 1’s-165

Size () is used to get those values it means that count of 0’s and 1’s

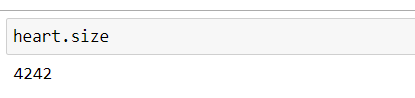


Same as previous this cell’s code looks similar, we group the target column and make then as a sum as we see in its output. Its main thing to add all the values based on target values



Shape-key word

This will help us to understand that how many rows and columns present in the heart variable.

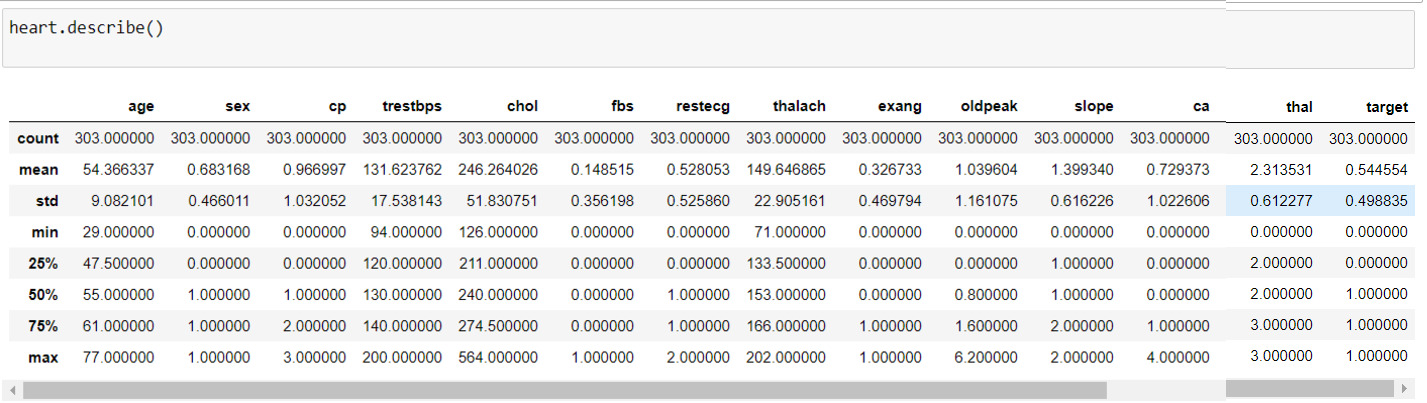


Size-key word

This might be something little-bit tricky in previous we used to determine the number of unique valued-values as a total count.

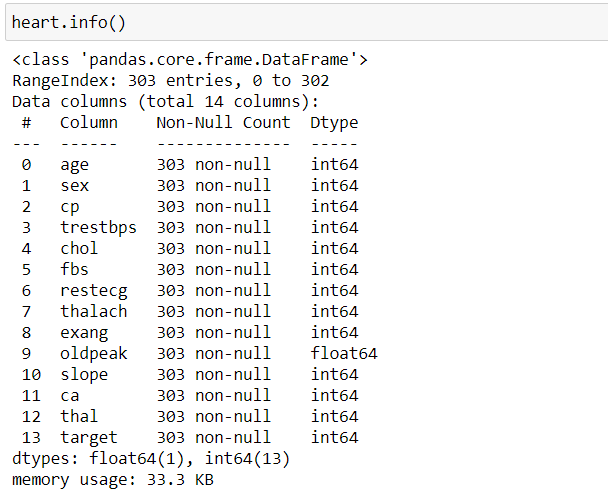
But as per this size-key word we can say that it gives “How many cells are used in the CSV file used to enter the data”

By this we can say that Python have huge set of libraries and just we have to why ad where we have use then we can do wonders in any branch of analytics used by python script.



Describe ():

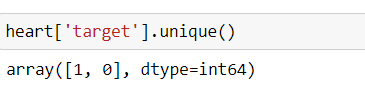
This very useful function to get an overview of the data which we have imported to the variable. It gives us all the data of total count, mean, standard deviation, minimum value in that column, Q1, Q2, Q3 of the data, at last we can have maximum of the column in that total info we can get the total of the respective column only.



Info ():

First of all we can have from which library we can have this info () based on the output we can say that it’s from pandas library. It gives us number of rows and columns are used to enter the data into the file.

Mainly it gives us title of the columns and Data-type used in that column.



Unique ():

It is used to check how many different values are present in the particular column. It returns as an array with the data-type of that column.

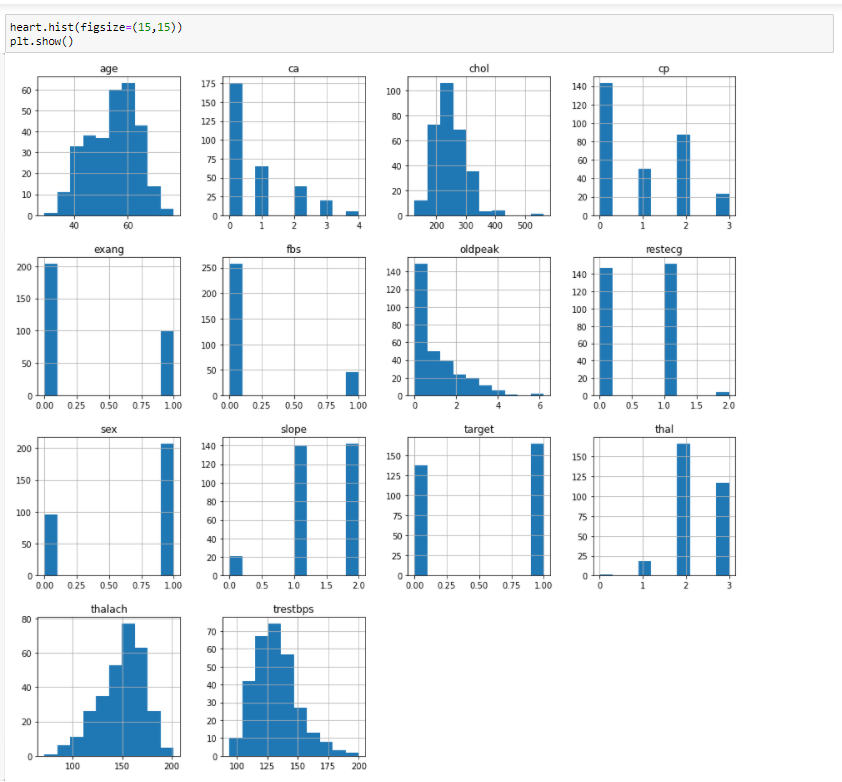


Figure-1

Hist ():

It is used to determine the written data which is in the numerical converted into the graphical format with respect to no of values entered. Best way to represent the data for quick picture about the data in the file.

All the columns in the y-axis and count on x-axis on this basis this is done automatically by the library in Python.

Fig-size: - It gives each graph size to be represented.

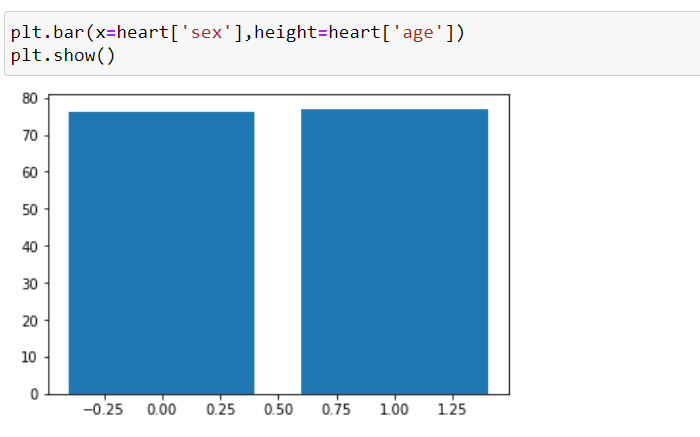


Figure-2

In this cell we have performed by using matplotlib library. We have decided to draw a bar graph with x-axis as Gender and y-axis as age which are from the data given. Plt show () is used to display in the format of graph.

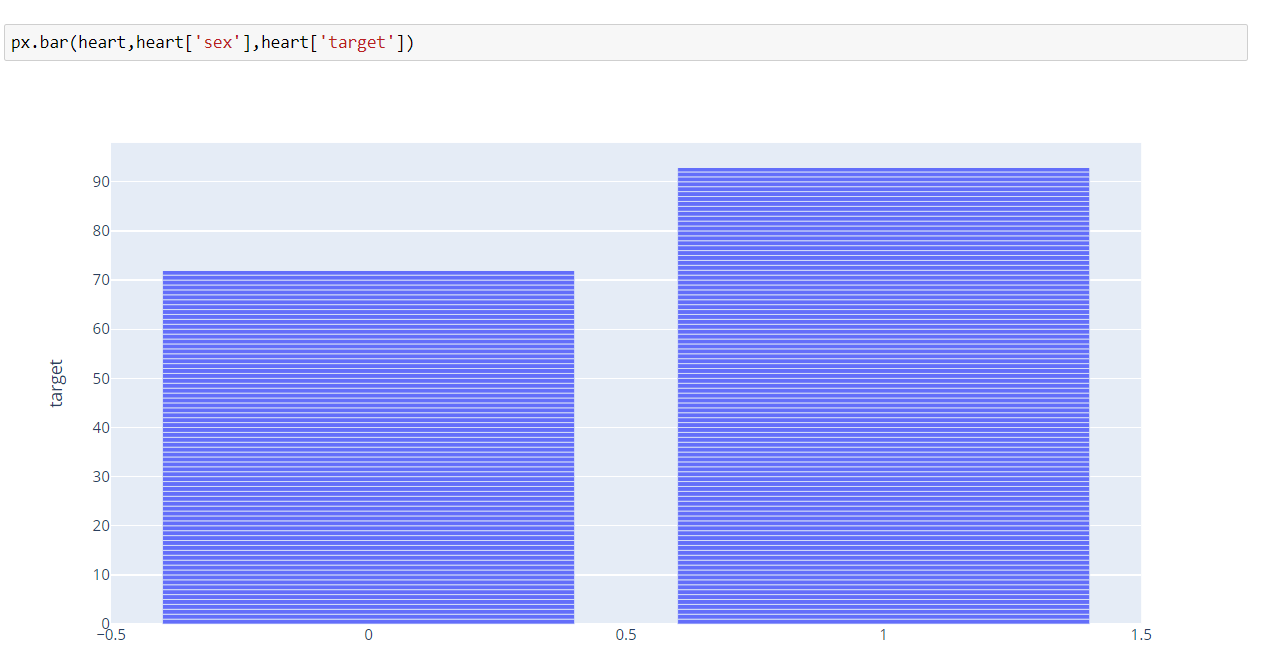


Figure-3

Px function is from the plotly.express library. It is drawn across gender and target from the data given in the heart variable. There are 3 arguments to be passed in this function

* 1->data\_variable
* 2&3-> what we want to display from the data.

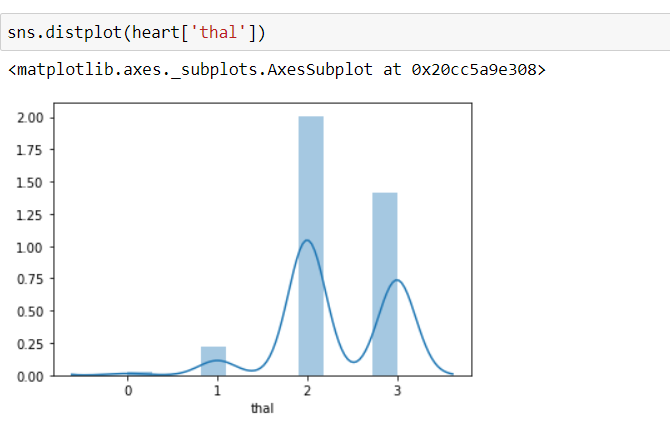


Figure-4

It is imported from the sea born library it is huge library and it contain “iris” data set also in it. From this now we take distplot to draw the graph of the thal. We can pass one or two arguments in this function.

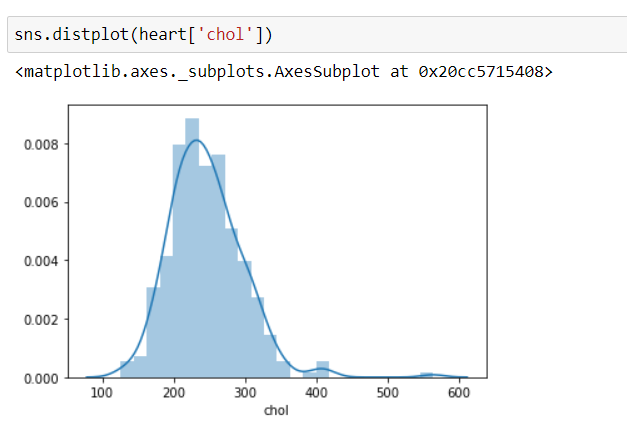
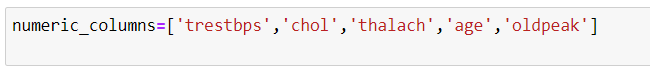


Figure-5

Same as above we have just changed the argument from thal to chol. Both are just the same.



We have declared another variable as numeric\_columns in that we have given an array of 5 column names and that will be used later in upcoming cells of code.

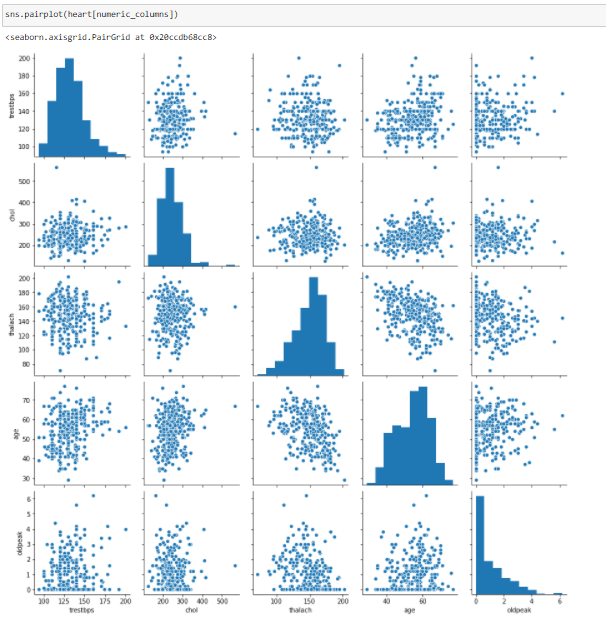
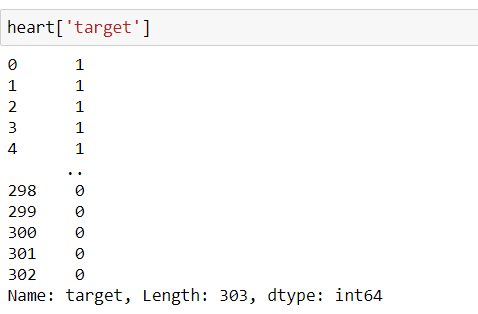


Figure-6

Sns pair plot of numeric columns:

It shows us the all the graphs of the columns present in that variable. Mainly it shows the graphs and by the other graphs we can analyze the relation between the columns.



Here we display the target values with respect to the column values.



We have stored heart[‘target’] data in the variable “y”.

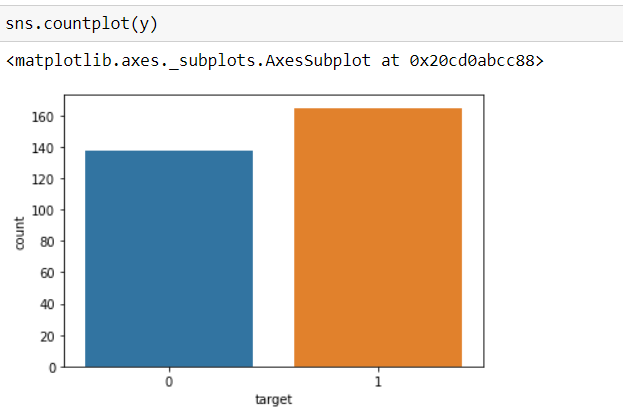
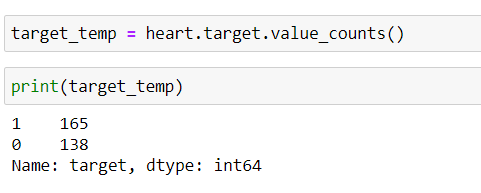


Figure-7

Count is used to count the how many numbers of 0’s and 1’s present in the variable ‘y’. Plot is used to display then in terms of graphs.



In previous slide we have plotted in terms of graphs now we are getting in terms of numerical and individual elements.

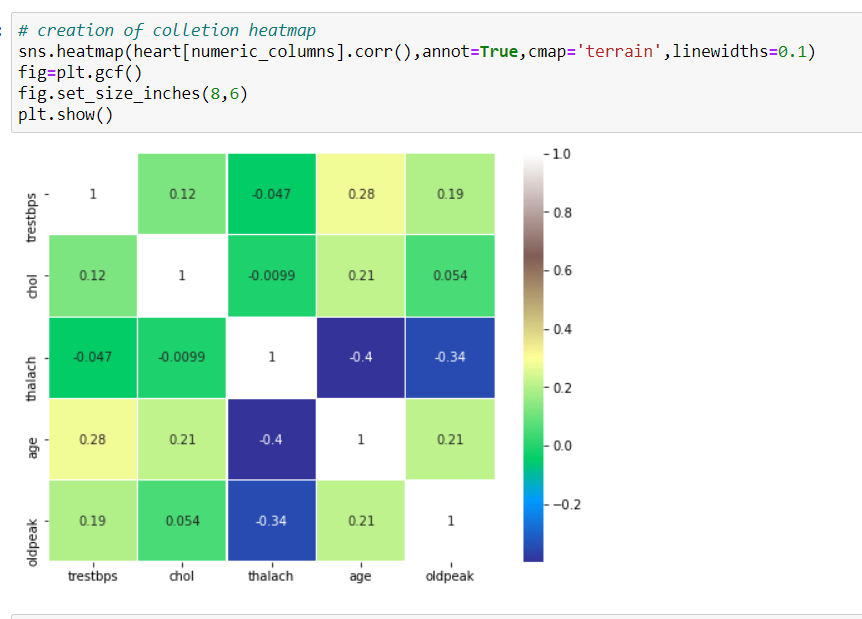


Figure-8

Creating a heat map using python we have choose the attributes whether it is terrain or anything to describe be the color of the hear map corr() is used to get the co-relation between the numeric columns.

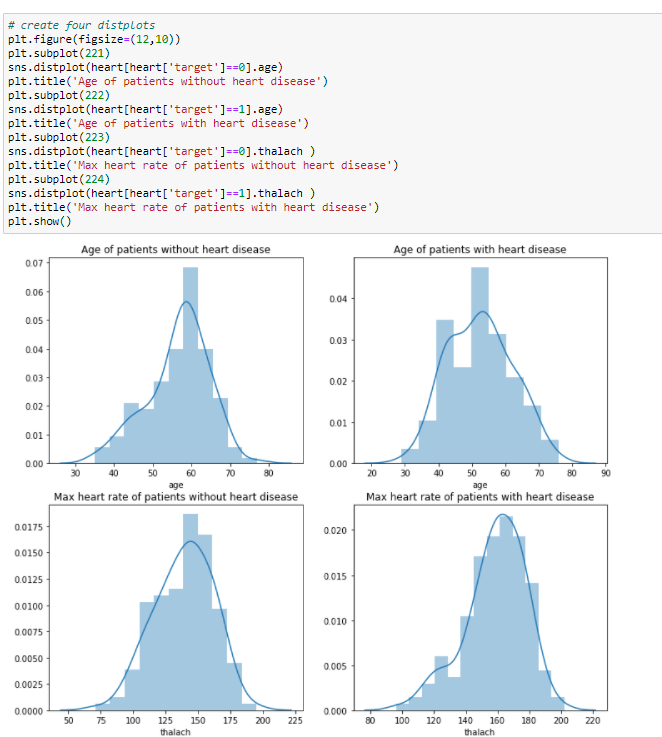


Figure-9

Subplot is used to get number of graphs can be fit into one row,

For example: - subplot (221) It means that there must be 2 rows and 2 columns and that present graph must be at the first position in the row.

Title() is used to Determine the name of the graph or what we will get by reading text of line for better understanding of the graph.

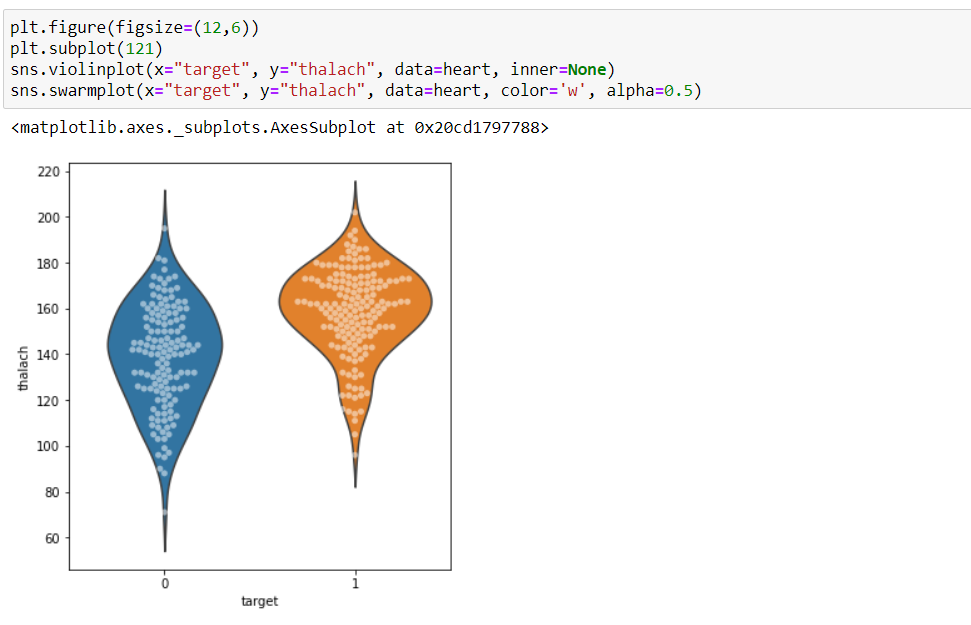


Figure-10

Fig size determines the size of the graph to be displayed in the output.

Violin plot is used to display the data in the shape of the violins. Swarmplot is used to display the data in terms of dots.

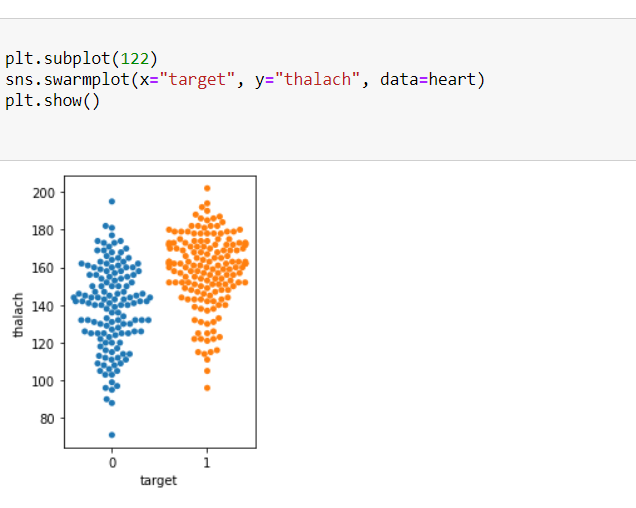


Figure-11

In previous image we have plotted the violin graph with the violin plot but here we have used only swarmplot in this so it looks like a dotted structure or simply a small irregular shape of honey comb.

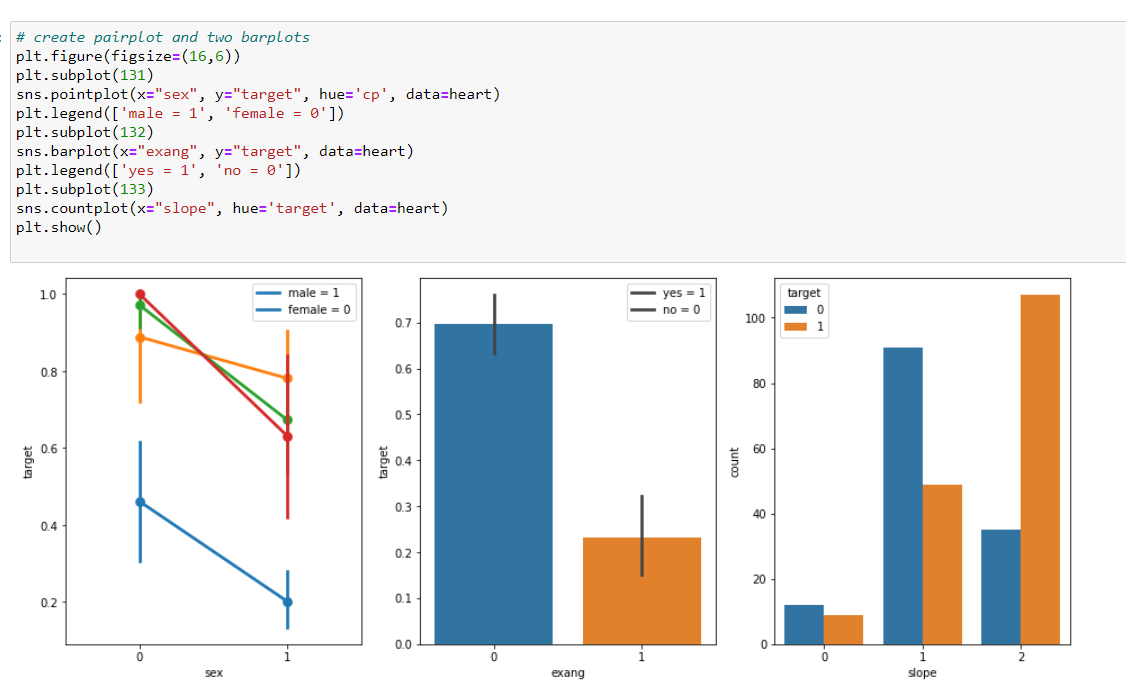
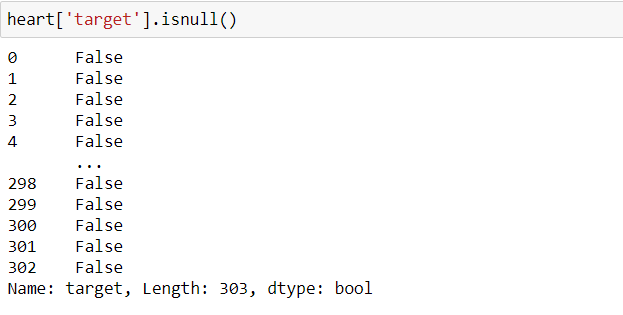


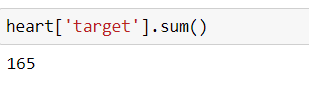
Figure-12

Here in this sub plot we have (131), that indicates 1 row 3 columns and it must be the 1st graph of that row. In this point plot we have seen that chest pain for the females will be more than men

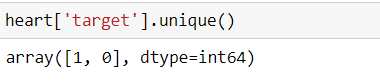
2nd graph is the bar plot with the coordinates (exang, target), 3rd graph has a legend we can determine with the legend () It may be very useful for better analysis of the user.



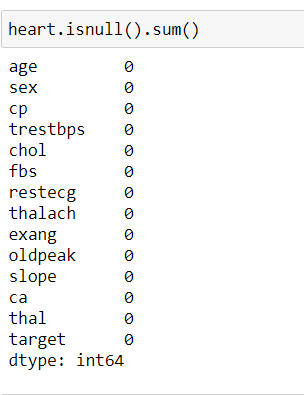
isNull () is used to determine the NULL value is present in the given column.



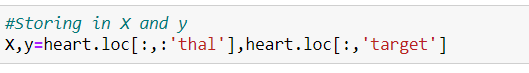
Sum () is used to do total summation of the column which is in int datatype



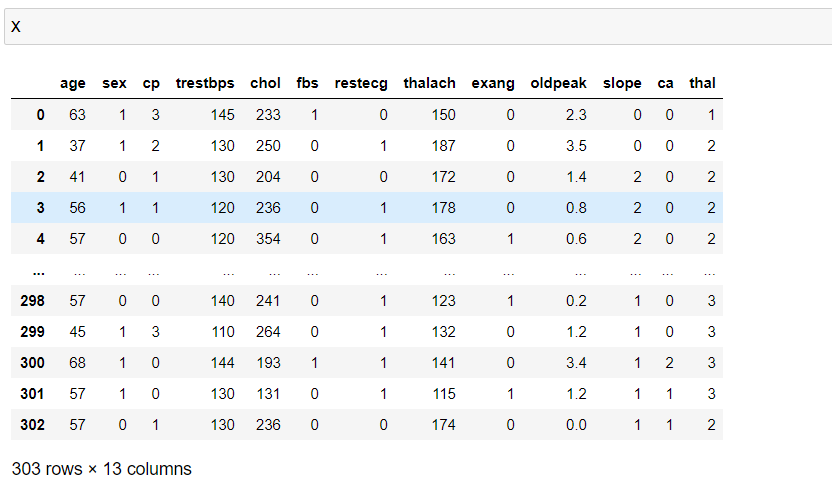
Unique () is used to tell us how many different values present in the column.

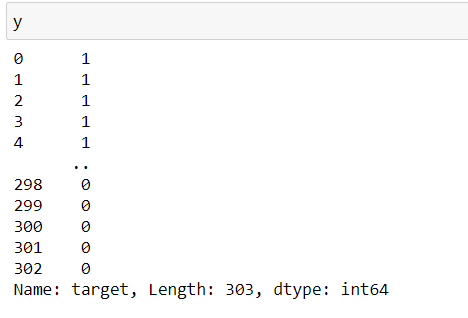


Is null and sum ()’s is used to determine if there are any NULL values in the column and it also determine the value if there is any NULL values in the column

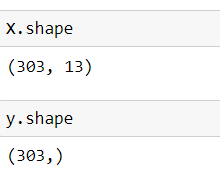


Here we are X as storing variable the values of columns up to thal column.

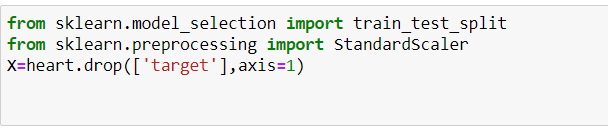




Y stores the vale after thal to target column.

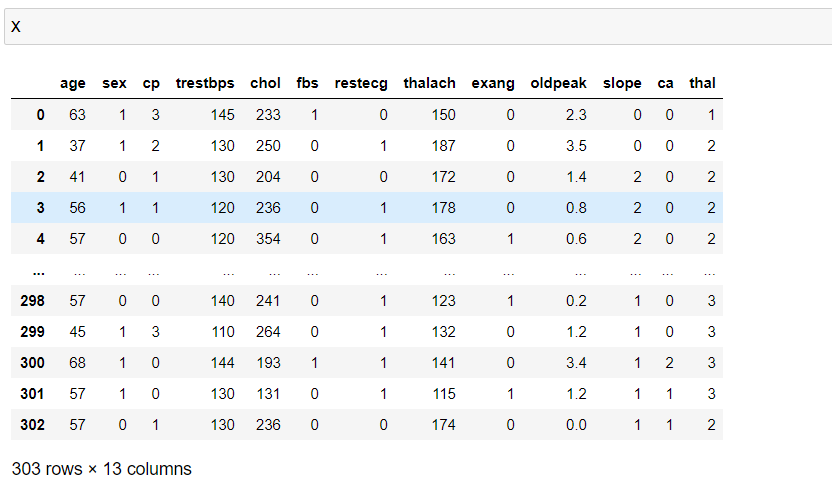


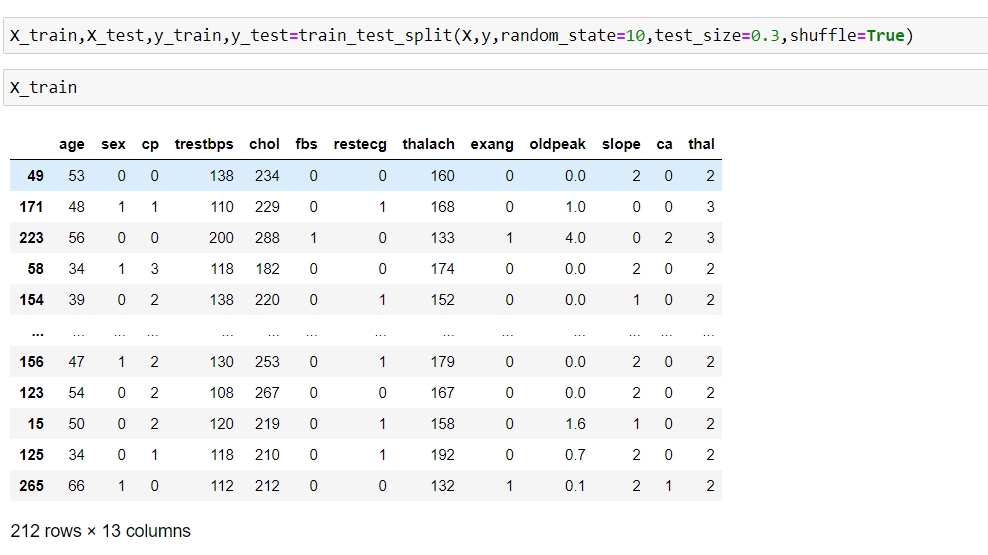
Gives us the number of rows and columns present in the variables x has 13 columns and y has only 1 column.



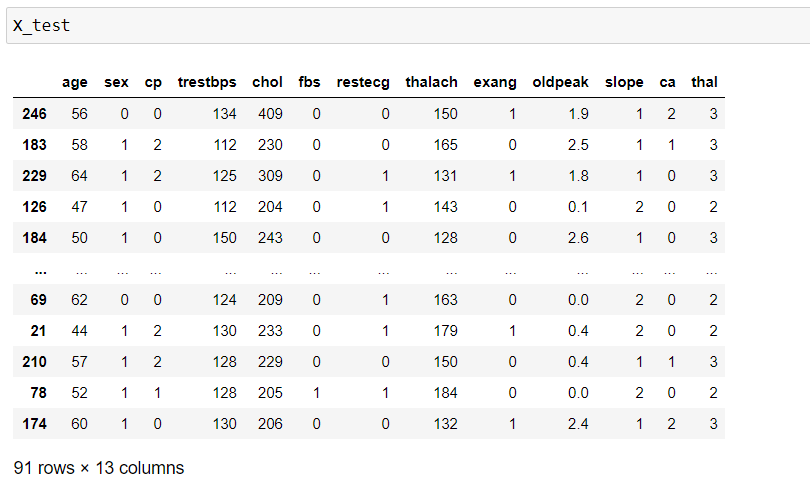
From this code we will import a machine learning library sklearn which is very useful for prediction of the data.

From this we will remove the target value from the data we have.

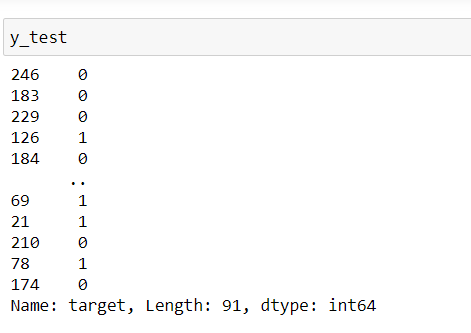




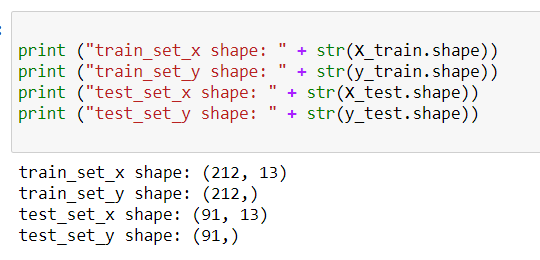
Here we are trying to train the model and we shuffling the to train the model, test size that we should take only 70% of the data.



Here we are displaying the testing model which we have tested in the above case.



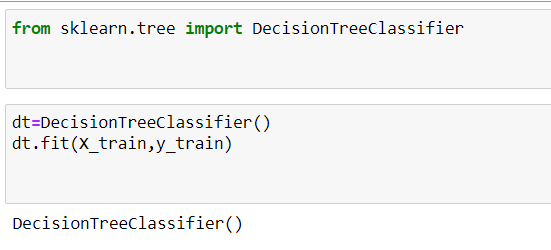
We are displaying the values stored in the y\_test.



Here we are printing number of rows and columns in the respective test and train modules.

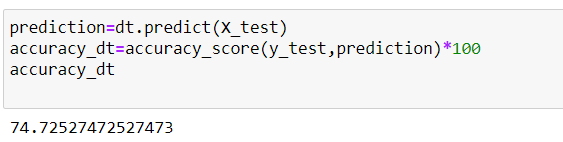


We are creating a model with the Decision tree Classifier and we have taken an statement in the form of array so that it may predict data.



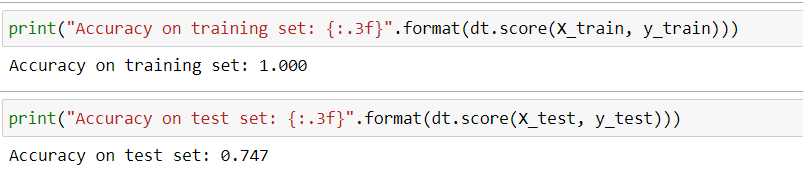
To create a model we have to import Decision tree Classifier from sklearn.tree library.

* **Chapter-6 Test Cases**

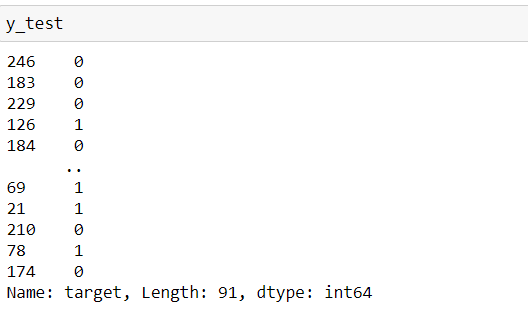


We have dt as Decision tree Classifier and that data we are predicting the data in the X\_test data.

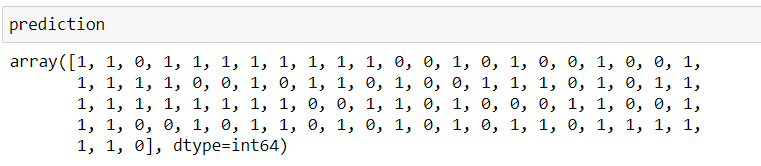
Accuracy score will predict us the data in correct manner. We re printing the accuracy in terms of percentage we have got 74%. It is very good while predicting the data.



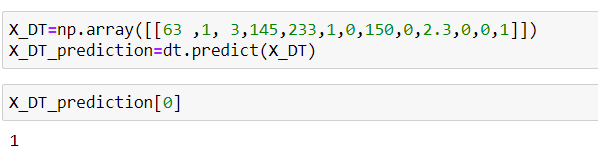
Here we are printing the score of x train, y train and x test and y test.



We are printing the y test data stored in that variable.

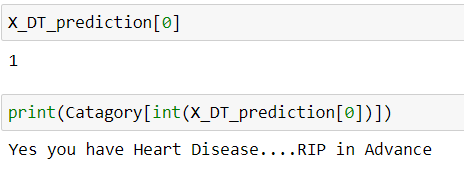


Prediction values of each in terms of array which are taken from the dt.predict of x\_test.

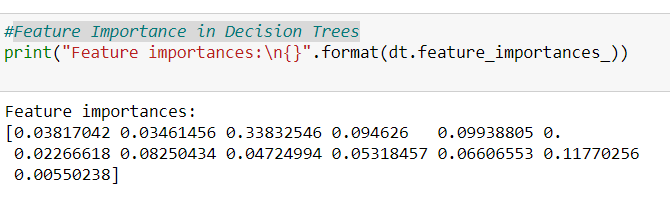


We are taking the data of the 1st row in np.array

In the prediction we are predicting the data which is stored int X\_DT variable.



We are trying to print the prediction value in the 1st row of code and in the catagory array we have stored and we are showing the array value in it will display in the content present in the catagory array.



Feature importance is calculated as the decrease in node impurity weighted by the probability of reaching that node. The node probability can be calculated by the number of samples that reach the node, divided by the total number of samples

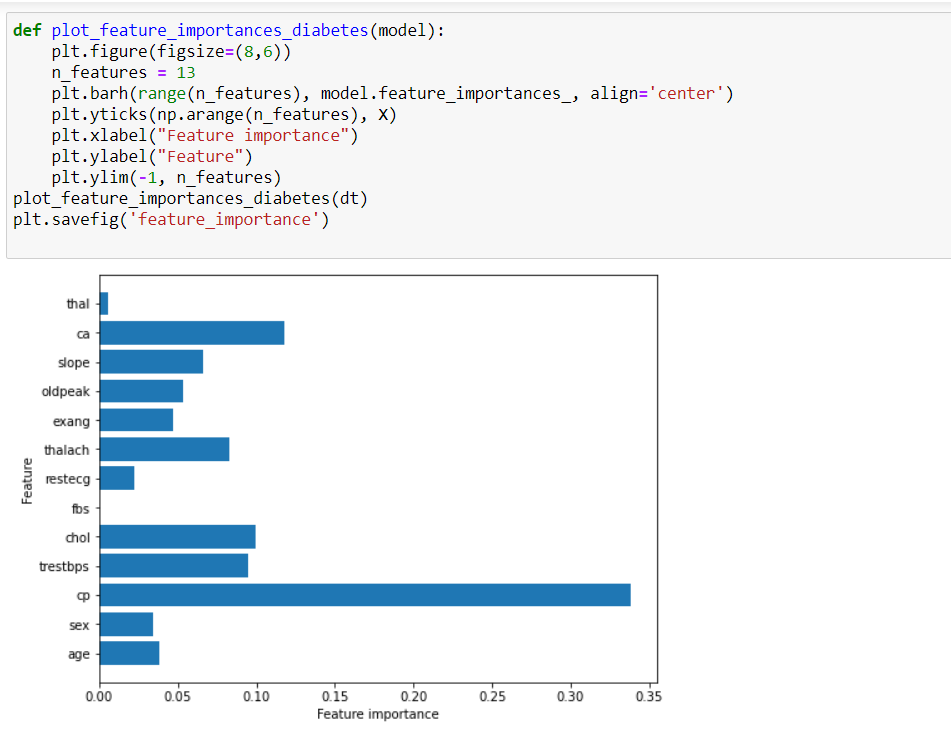
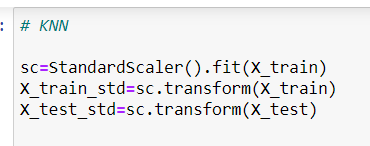
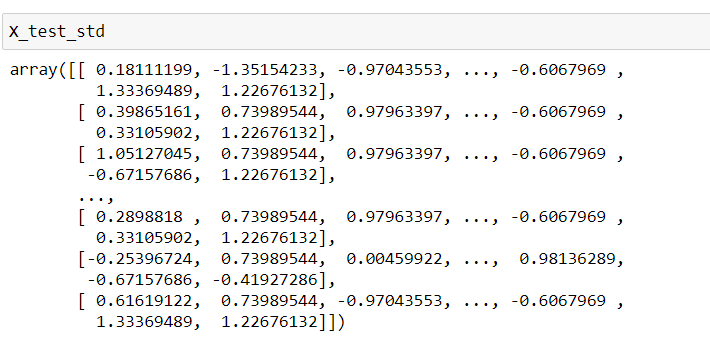


Figure-13

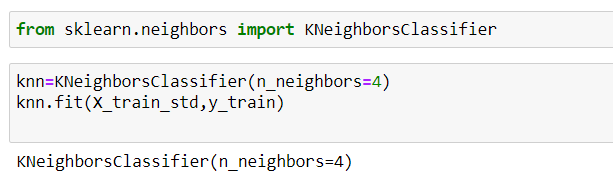
We are creating a function with the name plot\_feature\_importances\_diabetes, it will draw the graph Feature and Feature importance.



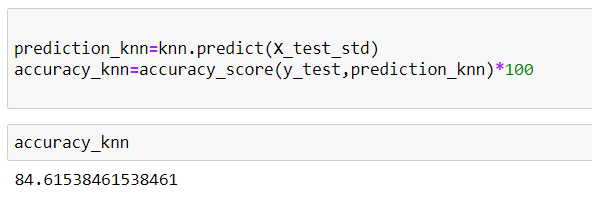
Here we are assigning the variables for different type of operations to transform to standard scale.



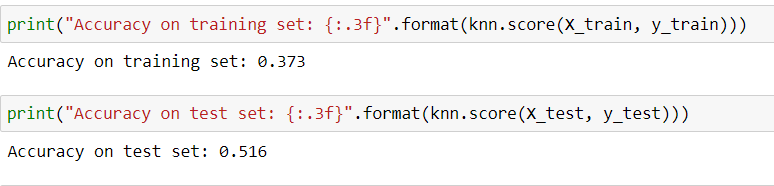
All the data which is in the X\_test have been normalized into a standard format.

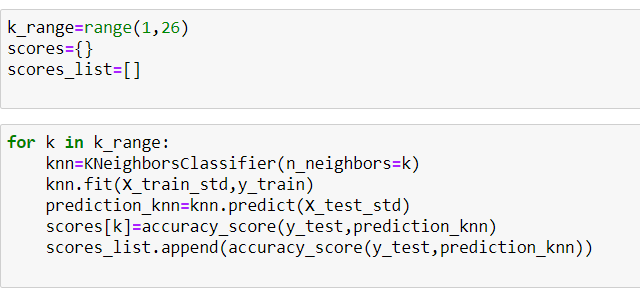


We have declared neighbor classifiers as 4 and we have done fit () to fit X\_train\_std and y\_train.



The prediction of knn score is damme high. Its result is very nice compared to the previous comparision.



Here we are printing the score of x train, y train and x test and y test of knn 

This code is taking the data and represented to form as a graph.

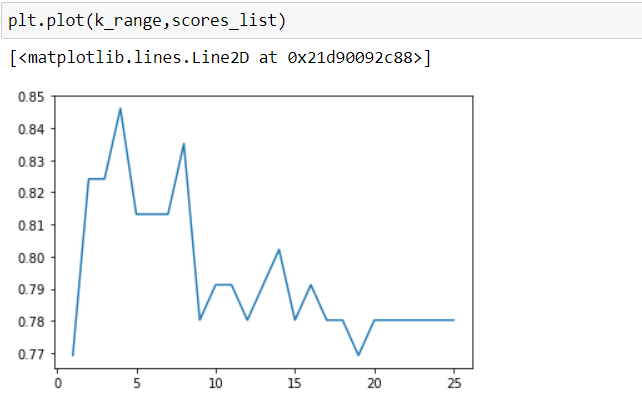


Figure-14

Data is representing in the form of graph.

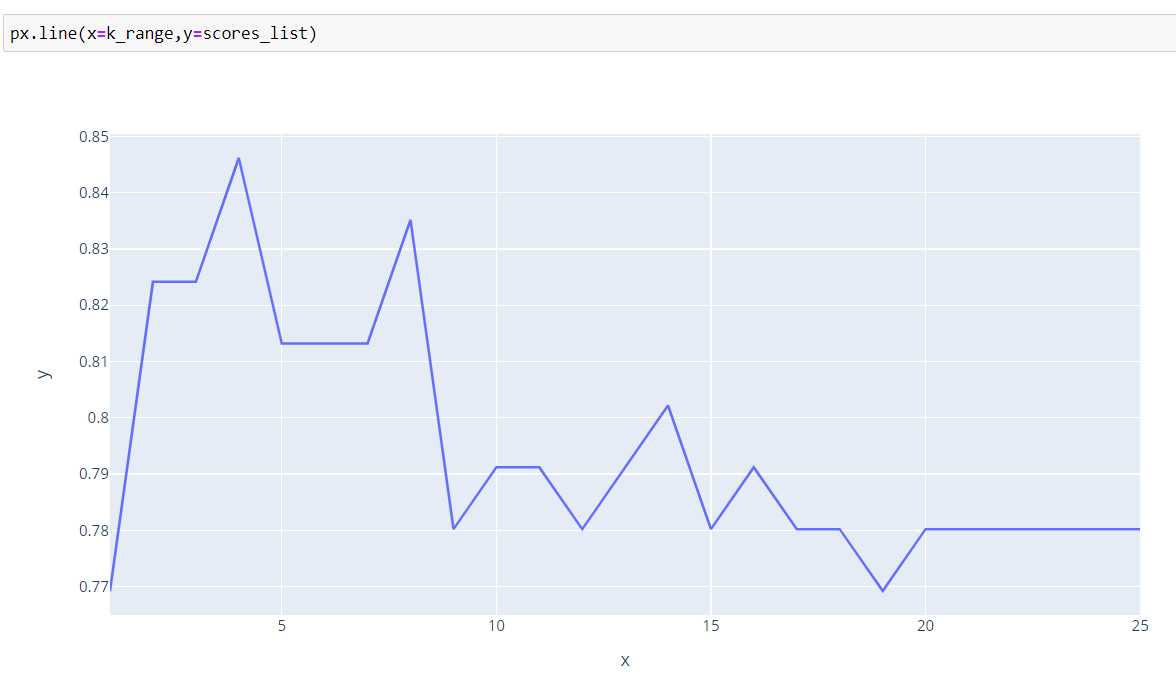
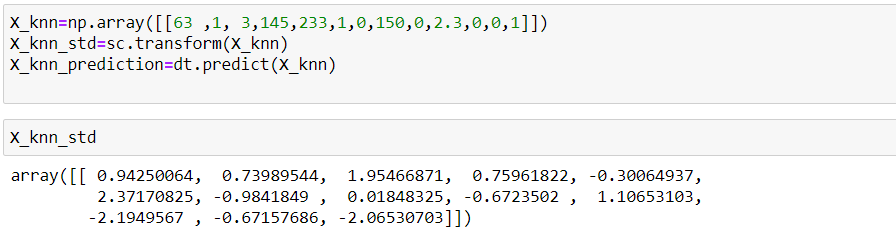
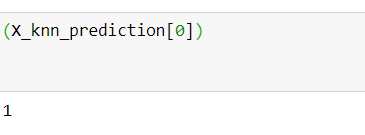


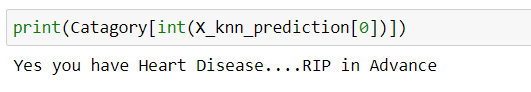
Figure-15

Graph is drawn between the k\_range and scores\_list.



Prediction values of each in terms of array which are taken from the dt.predict of x\_test from knn.





We are trying to print the prediction value in the 1st row of code and in the catagory array we have stored and we are showing the array value in it will display in the content present in the catagory array from knn.

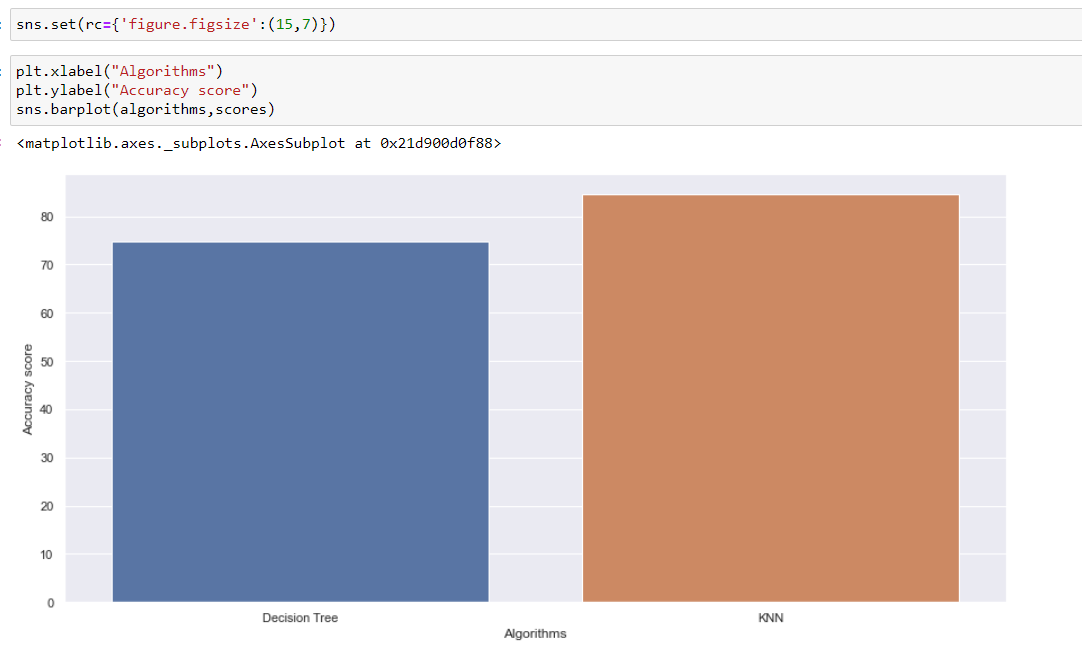


Figure-16

We have given figure size as 15 x 7 and we have drawn graph between the algorithms and accuracy. We have more accuracy in KNN when compares to Decision Tree.

* **Conclusion**

The project performed the different operations to get the maximum accuracy of disease identification from given data-set. The performance of the process has further enhanced with a feature selection implementation, and the results presented in the thesis and it is observed that there is significant improvement in prediction of the disease with high accuracy.