PREDICTION OF HEART DISEASE USING MACHINE LEARNING

GROUP MEMBERS:

- VENKATESH.S(AP18110010656)→CSE-J
- VIJAY.B(AP18110010628)→CSE-J
- LOKESH TRINADH.K(AP18110010634)→CSE-J
- CHARAN SANDEEP.B(AP18110010642)→CSE-J
- BALA VAMSI.P(AP18110010617)→CSE-J

ABSTRACT:

Our project is all about the prediction of the heart disease using Machine Learning, where we used KNN as the algorithm to proceed our project. Due to age, smoking, high blood pressure (BP) and several other reasons many people are getting affected and facing this issue of heart disease. So, it's better to develop an idea to predict the person is having heart disease or not. So, this is our idea behind the selection of the project. The models based on the Supervised Learning algorithms such as KNN (K-Nearest Neighbour) Algorithm, Random Forest Algorithms, SVM, Decision Tree are very much popular and are used to sort out the result of the project.

Using this project we are trying to predict whether the person is having heart disease or not (0 or 1). Also we are going to predict the accuracy of the heart disease using the approached KNN Algorithm.

KEY WORDS:

K-Nearest Neighbour (KNN) , Decision Tree (DT) , Support Vector Machine (SVM) , Supervised Learning, Machine Learning

INTRODUCTION:

As we all know that heart is one of the most important part in our body and it pumps blood to each and every part of our body. It needs to function properly , as if it doesn't function properly the person will get affected and will die within few minutes.

Now a days, the people in very young age are also facing the problem of heart disease and the growth rate is been rapidly increasing day by day. As it is one of the most important part of the body , it is better to study and develop the solution to predict the heart disease using the knowledge which we have based on the input dataset (Gathered from websites).

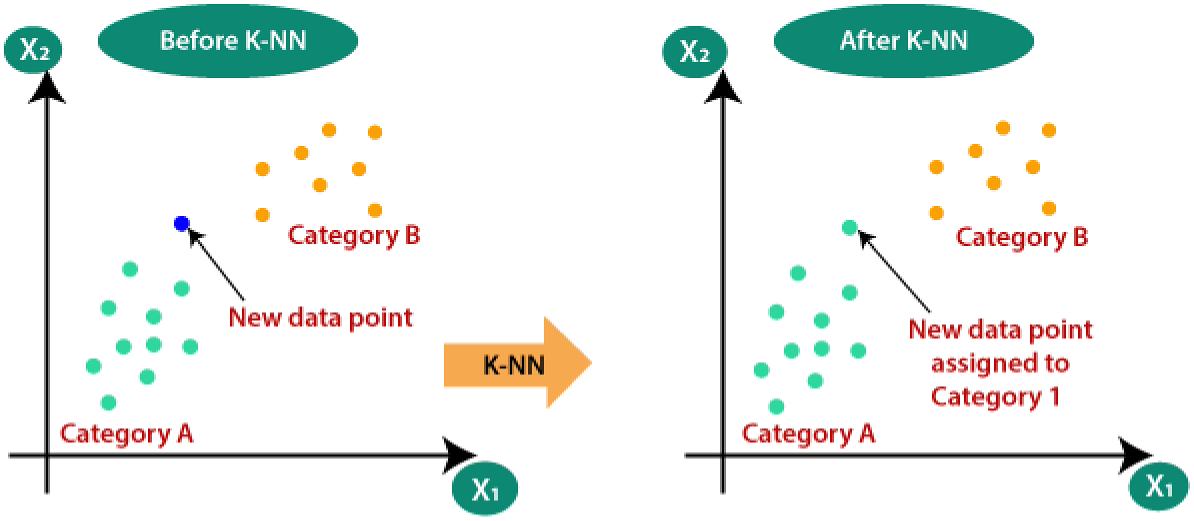
We also have various algorithms used in predicting the heart disease very precisely and accurately.

We used KNN as the algorithm in moving forward,

KNN ALGORITHM:

It is based on the supervised learning and it is one of the simple machine learning algorithms used. It mainly deals in solving the classification problems. It is also called as lazy learning algorithm as it do not learn anything from the training data but this training data is stored and at the time of classification, it performs the action of the dataset that it is stored.

It does not make any assumptions about the dataset and it is generally used in classification tasks, with the available datasets it has. This algorithm helps in finding the k-nearest data points (k represents the integer value) in the training set to the data point for which a target value is not available and assigning the average value of the found data points to it.



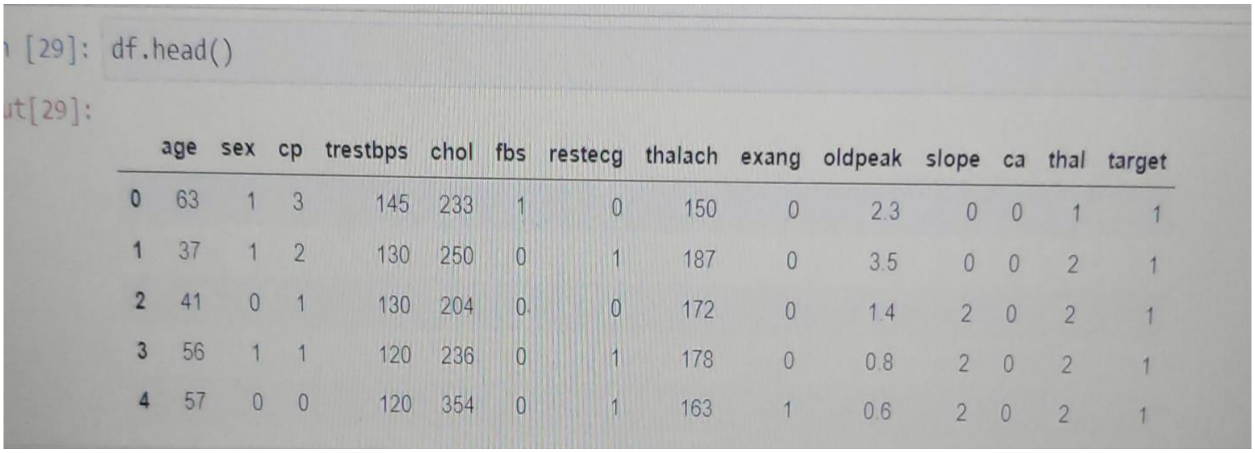
In case we have 2 categories of Category A and Category B, where we have a set of data point inputs, 10 data points of Category A and 8 data points of Category B. These can be separated using the classifier, but in future if we have a unseen data point given and asks us to classify in one of the two groups, then we can use this KNN approach to classify the given dataset into the respective category by just selecting the value of K.

By using Euclidean Formula, we can find the nearest neighbors by taking K=5 from the given new data point, then we have 3 neighbors from Category A and 2 from Category B. So, we can classify the given input data point into Category 1 as shown in the above figure.

PROPOSED WORK:

As our main aim to predict the accuracy and show the graph of number of persons having heart disease and number of persons not having heart disease. So, we proceeded with KNN algorithm in predicting. By clicking df.head(), it displays all the values of the input datasets we have choosen.

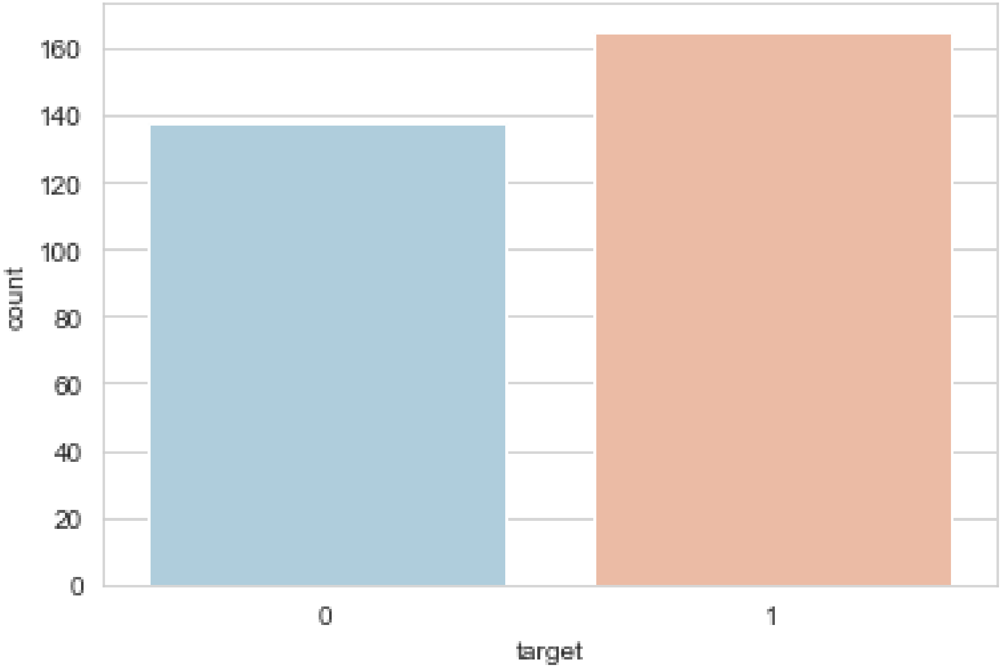
Input Dataset values:



```
[29]: df.head()
ut[29]:
```

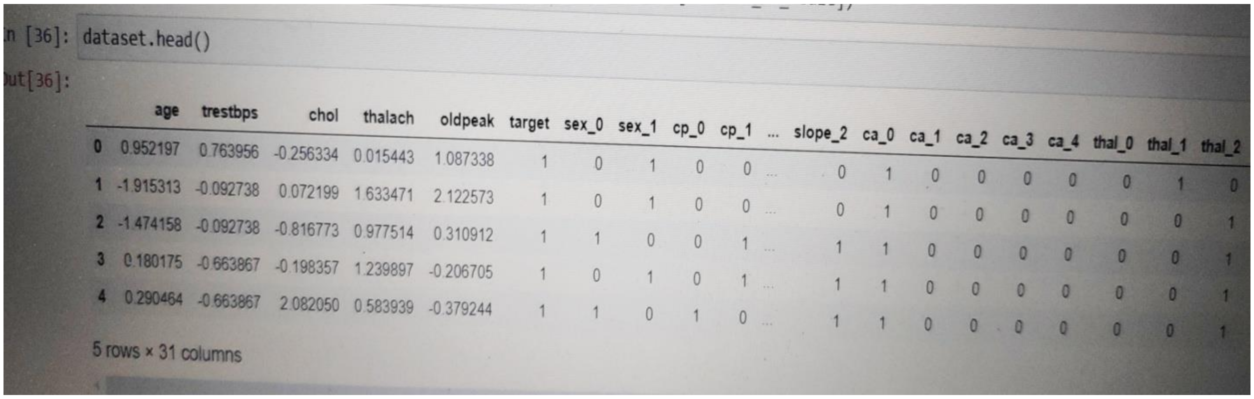
	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

Then after importing the packages and necessary libraries, df.describe() used to display the information related to count, mean and many other attributes. Then finally we have the number of persons having heart disease and the ones who are not having heart disease, and they are plotted in a graph using matplotlib library.



This is also used to check whether the data is imbalanced or not.

Later we need to perform the standard scaling to have all the attributes of similar data type. From the sklearn we have imported standard scaler. We also created dummy variables in this case and finally by performing dataset.head() we will have the updated table with new dummy variables created.

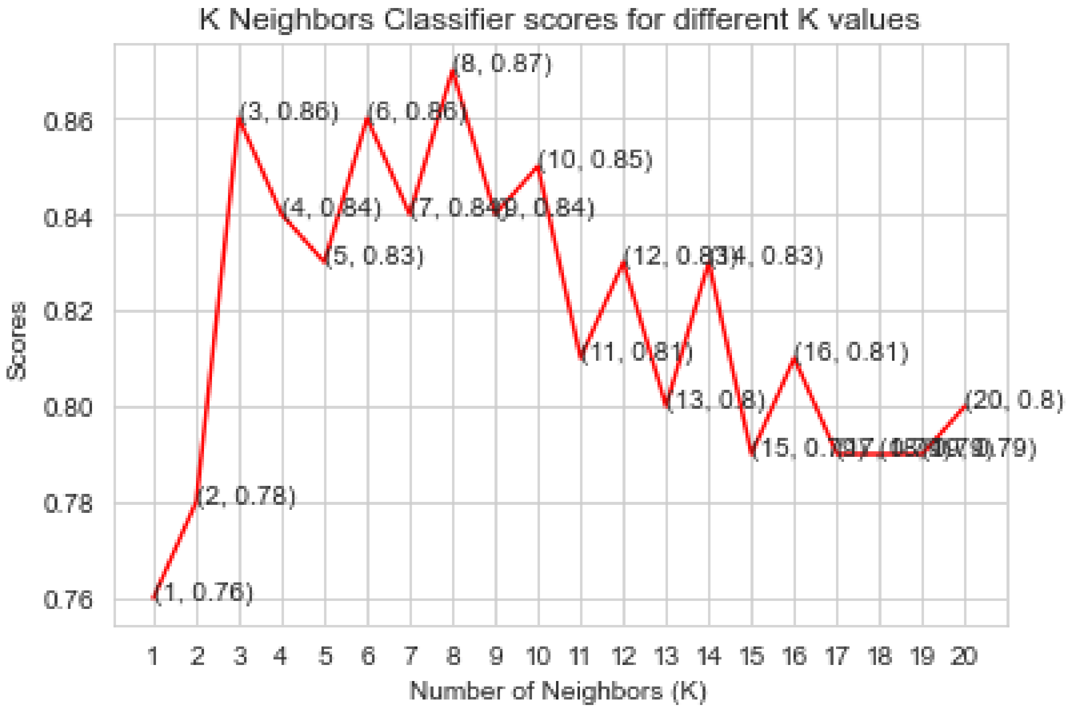


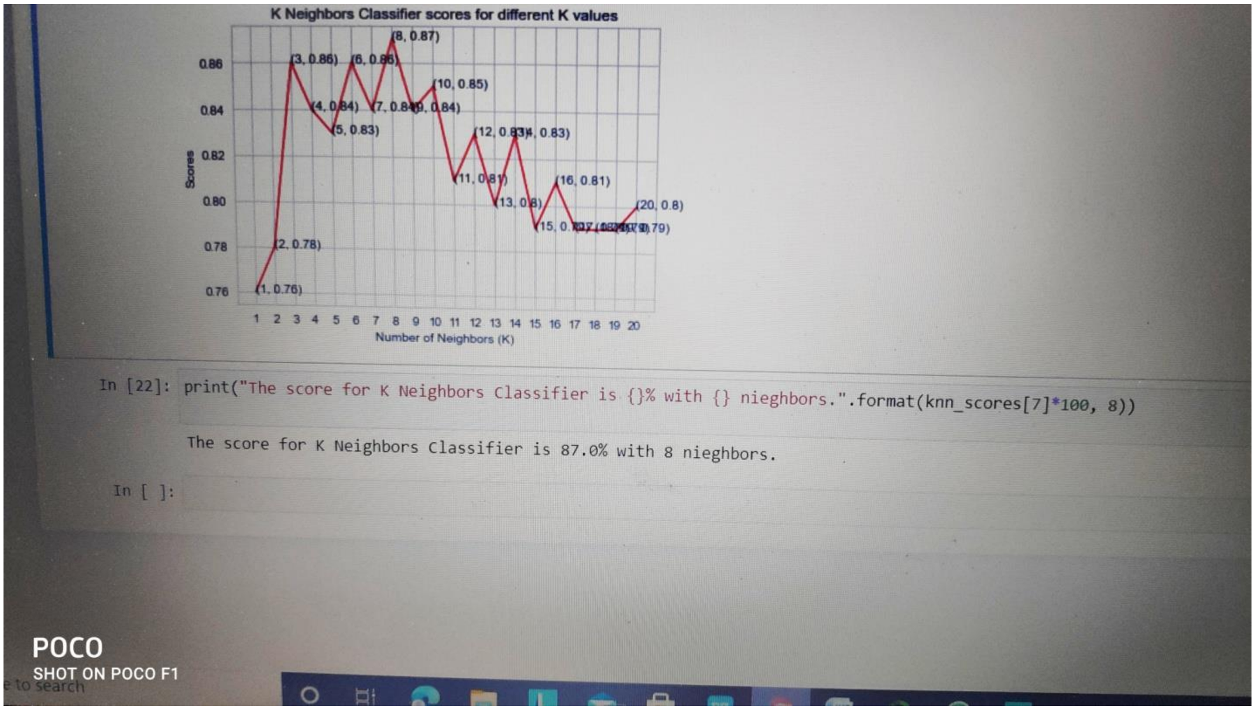
```
n [36]: dataset.head()
ut[36]:
```

	age	trestbps	chol	thalach	oldpeak	target	sex_0	sex_1	cp_0	cp_1	...	slope_2	ca_0	ca_1	ca_2	ca_3	ca_4	thal_0	thal_1	thal_2
0	0.952197	0.763956	-0.256334	0.015443	1.087338	1	0	1	0	0	...	0	1	0	0	0	0	0	1	0
1	-1.915313	-0.092738	0.072199	1.633471	2.122573	1	0	1	0	0	...	0	1	0	0	0	0	0	0	1
2	-1.474158	-0.092738	-0.816773	0.977514	0.310912	1	1	0	0	1	...	1	1	0	0	0	0	0	0	1
3	0.180175	-0.663867	-0.198357	1.239897	-0.206705	1	0	1	0	1	...	1	1	0	0	0	0	0	0	1
4	0.290464	-0.663867	2.082050	0.583939	-0.379244	1	1	0	1	0	...	1	1	0	0	0	0	0	0	1

5 rows × 31 columns

Then finally we came across the main part of the project, where we predicted the accuracy using KNN algorithm for having the neighbors ranging from 1 to 21. Then finally we calculated accuracy for each point and plot a graph for each neighbor what is the accuracy. For k=8, we have the high accuracy of nearly 87 percent accuracy.





```
In [22]: print("The score for K Neighbors Classifier is {}% with {} nieghbors.".format(knn_scores[7]*100, 8))

         The score for K Neighbors Classifier is 87.0% with 8 nieghbors.

In [ ]:
```

PROPOSED ALGORITHM:

→Firstly we need to import Libraries:

1. Numpy : We require numpy to work with arrays.
2. Pandas: We import Pandas library to include csv datafiles and Jframes.
3. Matplotlib : Used to create the graph and show the accuracy of the model using the particular algorithm.
4. Warnings: This library is used to ignore all the warnings that the working notebook or IDE will show.
5. train_test_split: We use this to split the dataset into training data and testing data

6. **Standard Scaler: To scale all the features, so that the Machine Learning model better adapts to the dataset.**

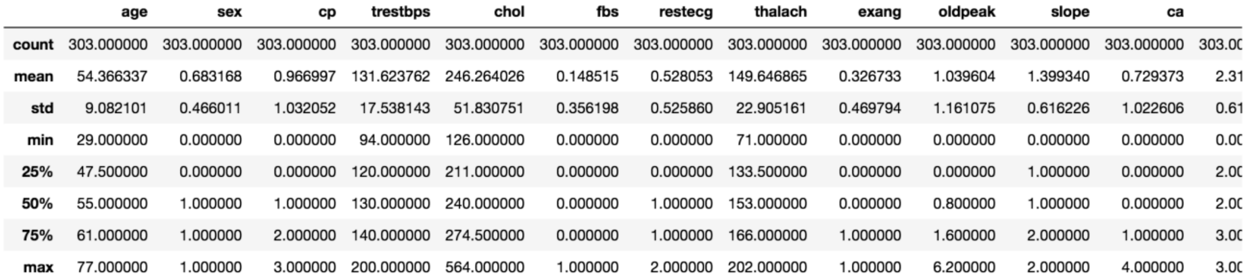
→Import dataset

After downloading the dataset , I saved it in my PC with the name `dataset.csv`. Next, `read_csv()` is used to read the dataset and then saved the dataset into the notebook where I run the code.

To check the information about the dataset we can choose df.info()

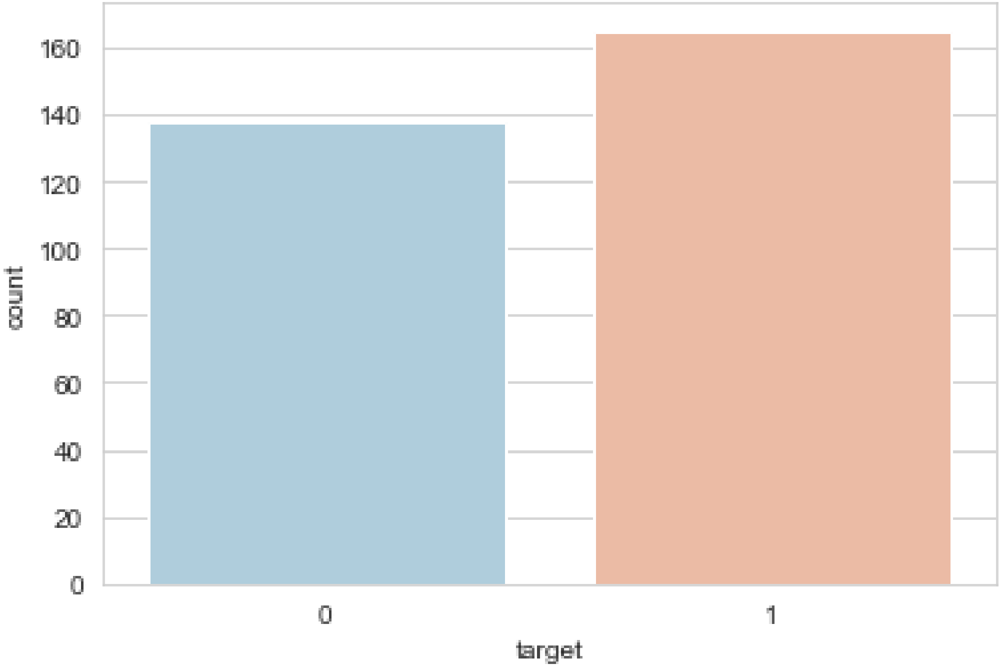
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       303 non-null    int64
 1   sex       303 non-null    int64
 2   cp        303 non-null    int64
 3   trestbps  303 non-null    int64
 4   chol      303 non-null    int64
 5   fbs       303 non-null    int64
 6   restecg   303 non-null    int64
 7   thalach   303 non-null    int64
 8   exang     303 non-null    int64
 9   oldpeak   303 non-null    float64
 10  slope     303 non-null    int64
 11  ca        303 non-null    int64
 12  thal      303 non-null    int64
 13  target    303 non-null    int64
dtypes: float64(1), int64(13)
memory usage: 33.3 KB
```

- ➔ **Using df.describe() we can have a total of 13 features and 1 target variable**



	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	
count	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.00
mean	54.366337	0.683168	0.966997	131.623762	246.264026	0.148515	0.528053	149.646865	0.326733	1.039604	1.399340	0.729373	2.31
std	9.082101	0.466011	1.032052	17.538143	51.830751	0.356198	0.525860	22.905161	0.469794	1.161075	0.616226	1.022606	0.61
min	29.000000	0.000000	0.000000	94.000000	126.000000	0.000000	0.000000	71.000000	0.000000	0.000000	0.000000	0.000000	0.00
25%	47.500000	0.000000	0.000000	120.000000	211.000000	0.000000	0.000000	133.500000	0.000000	0.000000	1.000000	0.000000	2.00
50%	55.000000	1.000000	1.000000	130.000000	240.000000	0.000000	1.000000	153.000000	0.000000	0.800000	1.000000	0.000000	2.00
75%	61.000000	1.000000	2.000000	140.000000	274.500000	0.000000	1.000000	166.000000	1.000000	1.600000	2.000000	1.000000	3.00
max	77.000000	1.000000	3.000000	200.000000	564.000000	1.000000	2.000000	202.000000	1.000000	6.200000	2.000000	4.000000	3.00

- ➔ **import seaborn as sns**
  Importing seaborn package and using it to plot the values against count and target.



- ➔ **We need to create dummy variables and then we need to do standard scaling to convert all the variables into single data types.**

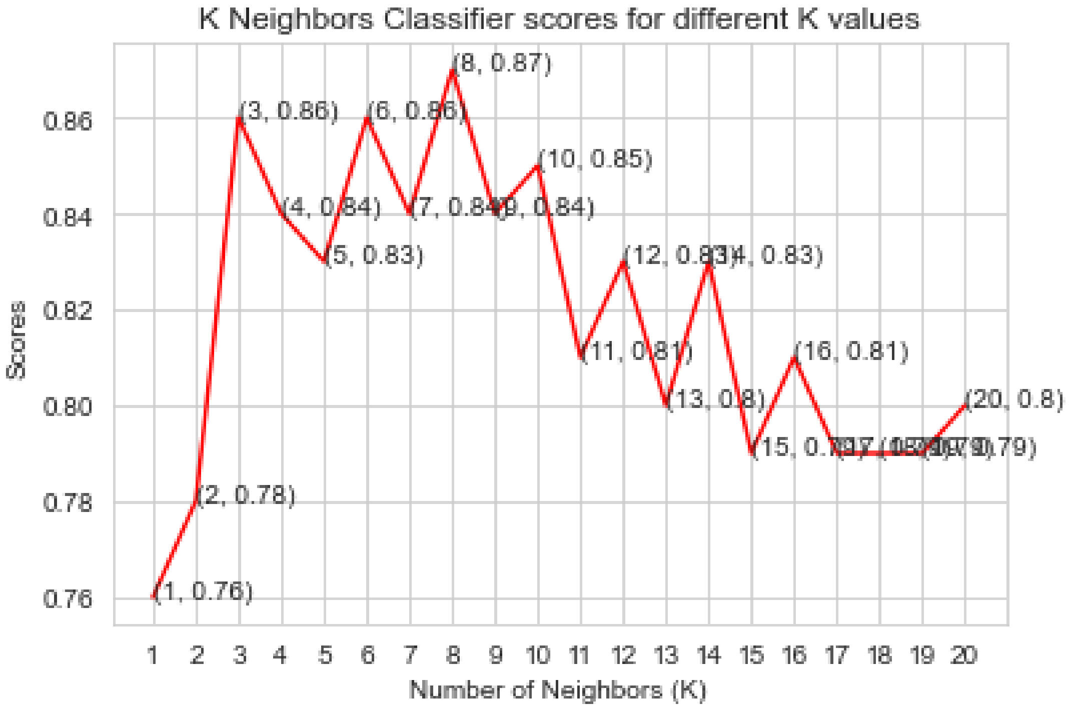
- ➔ **get_dummies() method is used from pandas.**

- ➔ **Then we can perform our project by taking KNN algorithm by considering dataset divided into 67% of training data and 33% of testing data.**

- ➔ **Then using KNN algorithm, having the neighbors varying from 1 to 21, and then calculating the test score at each case.**

- ➔ **Then finally, we can plot the graph for the test scores and number of neighbors.**

- ➔ **We can finally predict that when we have number of neighbors as 8, we find out the accuracy to be 87%.**



- ➔

EXPERIMENTAL ANALYSIS:

LIBRARIES USED:

- ➔Numpy
- ➔Pandas
- ➔Matplotlib
- ➔Warnings

SOFTWARE USED:

- ➔Anaconda Navigator Platform is used. Jupiter Notebook is the IDE used to launch and run the program.

PROGRAMMING LANGUAGE USED:

- ➔PYTHON

CONCLUSION:

This project helped me to calculate the heart disease using the dataset with having the data trained and tested by considering KNN Algorithm.

We got the accuracy of 87% using KNN Classifier algorithm.

We got the accuracy at the number of neighbors =8. We also helped in calculating the number of persons who are suffering from heart disease and who are not suffering from heart disease.

____