HEART DISEASE PREDICTION AND IDENTIFICATION WITH DEEP LEARNING AND NEURAL NETWORKS

BY KOTESWAR ENAMADNI

PROPOSAL

Heart-related diseases are among the most widely recognized reasons for death around the world. Patients are frequently asymptomatic until a deadly occasion occurs, and in any event, when they are under perception, a prepared workforce is required to recognize a heart abnormality. Somewhat recently, there has been expanding proof of how deep learning can be utilized to recognize such oddities, because of the accessibility of Electrocardiograms (ECG) in computerized design. New advancements in innovation have permitted us to take advantage of such information to fabricate models ready to dissect the examples in the event of heartbeats, and spot abnormalities from them. In this work, master cardiologists across various clinics and nations can recognize 7 sorts of signs: Typical, AF, Tachycardia, Bradycardia, Arrhythmia, Other, or Boisterous. [1][2]

The determination of heart illness has turned into a troublesome clinical errand in the current clinical examination. This finding relies upon the point-by-point and exact examination of the patient's clinical test information on a singular's well-being history. The huge improvements in the field of profound learning look to make astute mechanized frameworks that help specialists anticipate and decide the illness with the web of things with Deep learning. Certainty adjustment is an especially important issue in a medical care setting like the one tended to in this composition: when a neural network model makes an expectation, it is vital that this result can be relied upon.[3]

The Deep learning algorithm will be applied in the prediction of heart disease with a different types of attributes with the dataset.

Objective:

The objective of heart disease identification with deep learning is to detect heart disease in the early stage itself with the available attributes. In this work, the dataset containing heart disease will be taken into consideration[1][6]. The pre-processing will be applied to the dataset, and the noisy and null value data will be removed. After the data will be analyzed and visualized for further processing. The Deep learning algorithm will be chosen to make the prediction.

The dataset will be divided into two parts. The first part of the dataset is 70% taken to provide training to the Deep learning algorithm and the remaining 30% of data is taken to the testing part.

Motivation :

Deep learning will be the Python-based application that contributes to finding out the heart disease's early stage. It will be helpful for humans to detect it early and to take the necessary treatments at the correct time.

Significance:

The project evaluation can be tested with the deep learning algorithm prediction results. Since the Deep learning algorithm will be used to predict the disease, the accuracy of the algorithm result will be helpful to evaluate the results. [4]The accuracy score of the algorithm in heart disease identification helps to evaluate the dataset.

The application will be developed with Google Colab Python Tool as the project can be directly executed in any type of computer system with an internet connection. There is no need for any specific software to be installed in the user system. The Colab Tool helps to develop and run the application directly inside the cloud server where the Python library files are installed. The deep learning algorithm libraries are built inside the Colab. It helps the project to use the deep learning algorithm in the finding of heart disease.

INCREMENT

Techniques Applied:

The task included the examination of the heart disease patient dataset with appropriate information handling. Then, at that point, various models were prepared and expectations are made with the Deep Learning model.[1][7] The machine learning library and deep learning libraries Sklearn and Keras are applied to the application.

Dataset:

Most of the columns in a dataset are noisy and contain lots of information. But with feature engineering, will get more good results. The first step is to import the libraries and load data. After that will take a basic understanding of data like its shape, sample, is there any NULL values present in the dataset. Understanding the data is an important step for prediction or any Deep learning project. It is good that there are no NULL values.

The dataset heart.csv is downloaded from the Kaggle website, it has 300 rows and 14 columns present.

Detailed Design of Features:

This dataset contains the fields needed for the analysis of the heart disease dataset. The exploratory examination is a cycle to investigate and comprehend the information and information related in a total profundity with the goal that it makes highlight designing and Deep Learning demonstrating steps smoothly and smoothed out for expectation. The exploratory examination assists with validating our presumptions or misleading.

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
56	0	1	140	294	0	0	153	0	1.3	1	0	2	1
44	1	1	120	263	0	1	173	0	0	2	0	3	1
52	1	2	172	199	1	1	162	0	0.5	2	0	3	1
57	1	2	150	168	0	1	174	0	1.6	2	0	2	1
54	1	0	140	239	0	1	160	0	1.2	2	0	2	1
48	0	2	130	275	0	1	139	0	0.2	2	0	2	1
49	1	1	130	266	0	1	171	0	0.6	2	0	2	1
64	1	3	110	211	0	0	144	1	1.8	1	0	2	1
58	0	3	150	283	1	0	162	0	1	2	0	2	1
50	0	2	120	219	0	1	158	0	1.6	1	0	2	1
58	0	2	120	340	0	1	172	0	0	2	0	2	1
66	0	3	150	226	0	1	114	0	2.6	0	0	2	1
43	1	0	150	247	0	1	171	0	1.5	2	0	2	1
69	0	3	140	239	0	1	151	0	1.8	2	2	2	1
59	1	0	135	234	0	1	161	0	0.5	1	0	3	1
44	1	2	130	233	0	1	179	1	0.4	2	0	2	1
42	1	0	140	226	0	1	178	0	0	2	0	2	1
61	1	2	150	243	1	1	137	1	1	1	0	2	1

Analysis of Heart Disease Prediction:

It will begin from the principal segment and investigate every section and comprehend what influence it makes on the objective segment. At the necessary step, we will likewise perform preprocessing and include design undertakings. The point in acting top-to-bottom exploratory examination is to get ready and clean information for better Deep Learning demonstrating to accomplish elite execution and summed up models. So it should begin with breaking down and setting up the dataset for expectation.

Modules:

1) Dataset collection

2) Data cleaning

3) Data Analysis

4) Deep learning Modeling

5) Report

1) Dataset collection:

The information about the heart disease dataset with different types of attributes are collected from different type of patients.

2) Data Cleaning:

The large dataset contains more noisy and improper data which have to be pre-processed to produce a quality dataset for further pruning. The data is cleaned and processed with the initial stage of removing the null values.

3) Data Analysis

Exploratory analysis is a process to explore and understand the data and data relationship in complete depth so that it makes feature engineering and machine learning modeling steps smooth and streamlined for prediction. It helps to prove our assumptions true or false. In other words, it helps to perform hypothesis testing.

4) Deep learning Modeling

Deep learning modeling helps to find the best algorithm with the best hyperparameters to achieve maximum accuracy. The dataset is split into 2 variants. 70% of records are taken as training data and used to train the machine learning algorithm. The remaining 30% of the dataset is applied to testing which helps to predict the process.

5) Report:

The Data is visualized based on the output of the Deep Learning algorithm and the data is mapped with different types of graphs to analyze and visualize the exact data to the user for the prediction. Matplot libraries are implemented to map the results based on user requirements.

SYSTEM SPECIFICATION

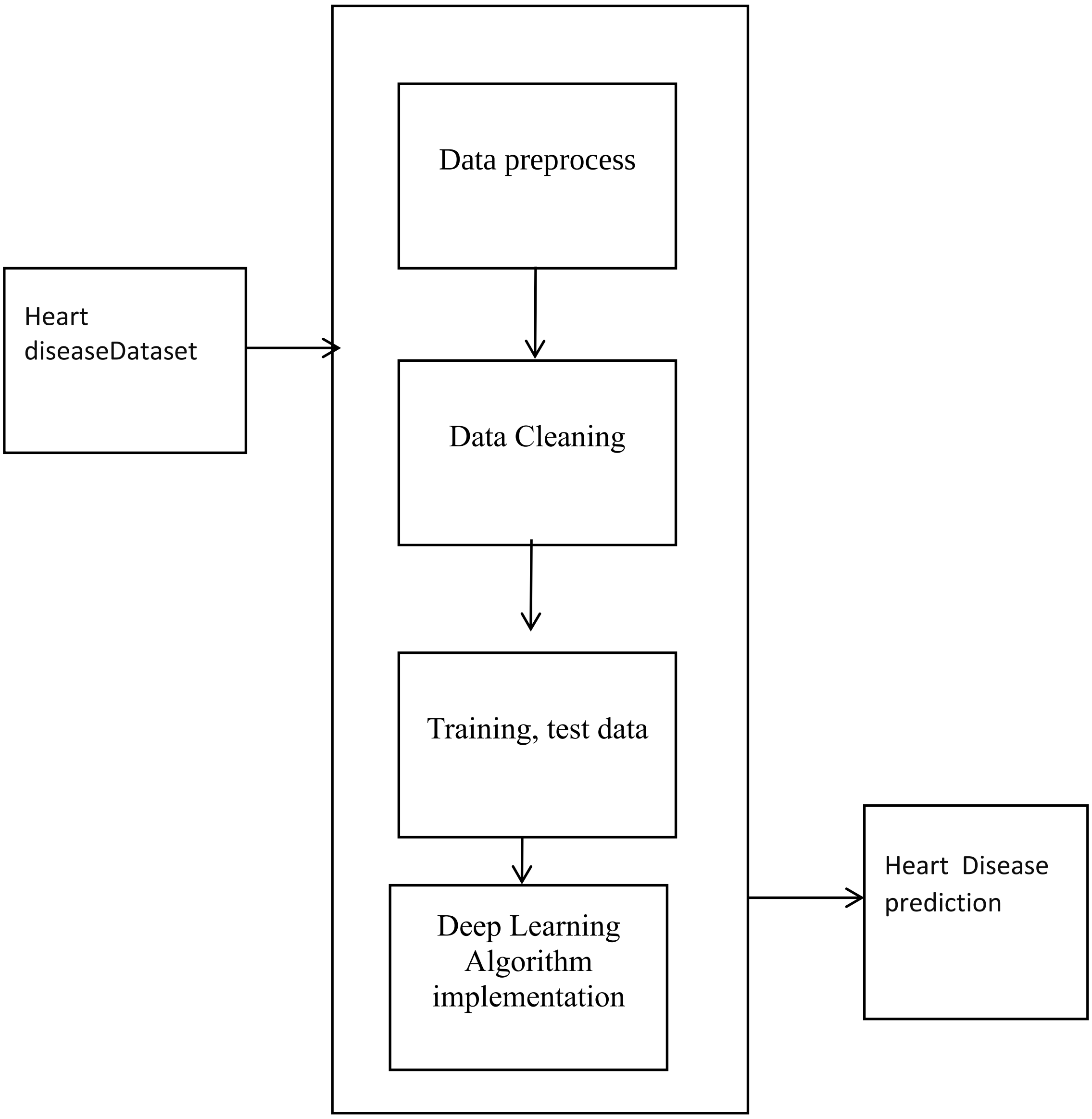
HARDWARE REQUIREMENTS

• Processor	Intel(R) Pentium(R) CPU G2010 @
• Clock Speed	2.80GHz
• RAM	2.00 GB
• Hard Disk	1 TB HDD
• Monitor	15.6 Inches
• Mouse	Logitech B100 Wired Optical Mouse
• Keyboard	Full-size island-style keyboard with number keypad
• Display Card	Super Video Graphics Adapter

SOFTWARE REQUIREMENTS

- ❖ Operating System: Windows 10
- ❖ Front-End Tool: Python in Google Colab

SYSTEM ARCHITECTURE DIAGRAM



ARCHITECTURE DIAGRAM

The heart disease dataset is given as input to the application and the pre-processing is applied, next the data cleaning is performed after the training and test data are splitted down and passed into the Deep Learning algorithm and the heart disease prediction is done.

Load Packages:

The first step is to import the necessary packages to the application:

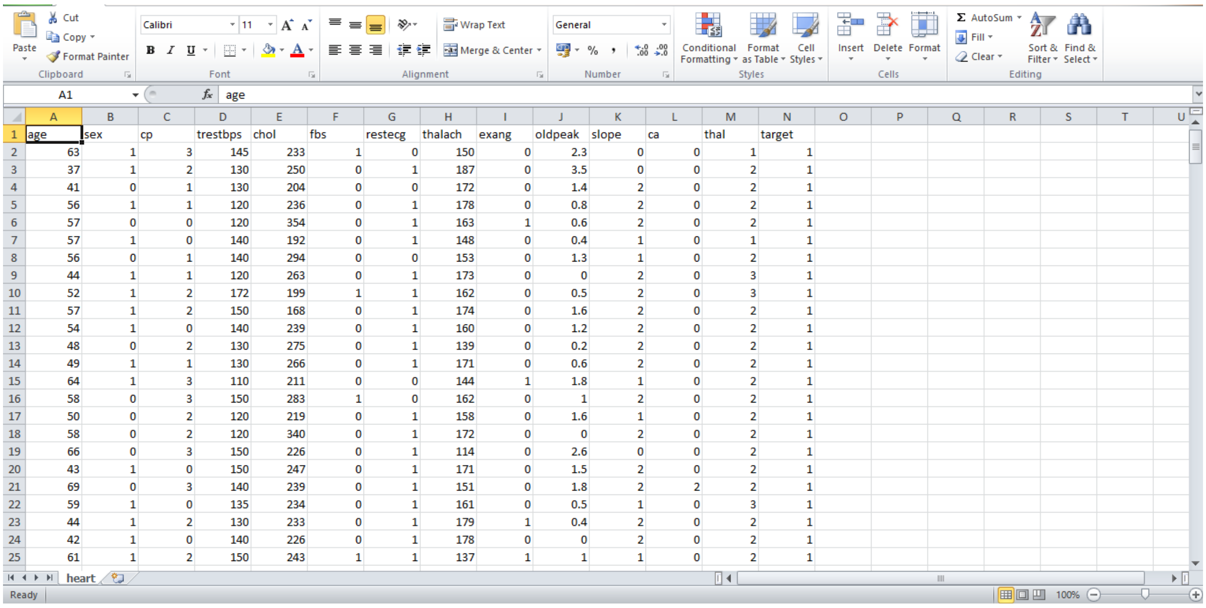
```
//---AI Library files
from keras.models import Sequential
from keras.layers import Dense


import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline

import os
print(os.listdir())

import warnings
warnings.filterwarnings('ignore')
```

Next, the dataset would be connected from the Google Colab. Initially, the dataset is uploaded into the Google Colab folder. Then the Python file should connect to the path from the Google Colab folder.

II. Importing the dataset

In [2]:
```
# Load the dataset
dataset = pd.read_csv('heart.csv')
```

The information has an extremely straightforward design with elements. Each column is related to the heart disease attributes.



age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
56	0	1	140	294	0	0	153	0	1.3	1	0	2	1
44	1	1	120	263	0	1	173	0	0	2	0	3	1
52	1	2	172	199	1	1	162	0	0.5	2	0	3	1
57	1	2	150	168	0	1	174	0	1.6	2	0	2	1
54	1	0	140	239	0	1	160	0	1.2	2	0	2	1
48	0	2	130	275	0	1	139	0	0.2	2	0	2	1
49	1	1	130	266	0	1	171	0	0.6	2	0	2	1
64	1	3	110	211	0	0	144	1	1.8	1	0	2	1
58	0	3	150	283	1	0	162	0	1	2	0	2	1
50	0	2	120	219	0	1	158	0	1.6	1	0	2	1
58	0	2	120	340	0	1	172	0	0	2	0	2	1
66	0	3	150	226	0	1	114	0	2.6	0	0	2	1
43	1	0	150	247	0	1	171	0	1.5	2	0	2	1
69	0	3	140	239	0	1	151	0	1.8	2	2	2	1
59	1	0	135	234	0	1	161	0	0.5	1	0	3	1
44	1	2	130	233	0	1	179	1	0.4	2	0	2	1
42	1	0	140	226	0	1	178	0	0	2	0	2	1
61	1	2	150	243	1	1	137	1	1	1	0	2	1

The label has been set with different label description features of:

- age
- sex
- cp
- trestbps
- cholestral
- fbs
- restecg
- oldpeak
- slope
- ca
- thal
- target

```
[6] dataset.sample(5)
```

	ï»¿age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
185	44	1	0	112	290	0	0	153	0	0.0	2	1	2	0
249	69	1	2	140	254	0	0	146	0	2.0	1	3	3	0
24	40	1	3	140	199	0	1	178	1	1.4	2	0	3	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
207	60	0	0	150	258	0	0	157	0	2.6	1	2	3	0

The dictionary shows the records displayed with head values of the first 5 records from the dataset. The bar plot can be used to address the heart disease attributes.

```
[7] dataset.describe()
```

	ï»¿age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	s
count	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	1.39
std	9.082101	0.466011	1.032052	17.538143	51.830751	0.356198	0.525860	22.905161	0.469794	1.161075	0.61
min	29.000000	0.000000	0.000000	94.000000	126.000000	0.000000	0.000000	71.000000	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	1.00
50%	55.000000	1.000000	1.000000	130.000000	240.000000	0.000000	1.000000	153.000000	0.000000	0.800000	1.00
75%	61.000000	1.000000	2.000000	140.000000	274.500000	0.000000	1.000000	166.000000	1.000000	1.600000	2.00

The describe function describes the count, mean, max, and min statistical reports for the columns of age, sex, cp, chol, and other fields.

```
[8] dataset.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   ï»¿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)
```

The information about the data is displayed with the data type.

Attributes Exploration:

```
info = ["age","1: male, 0: female","chest pain type, 1: typical angina, 2: atypical angina, 3: non-anginal pain, 4: asymptomatic","resting blood pressure"," seru



for i in range(len(info)):
    print(dataset.columns[i]+":\t\t\t"+info[i])
```

```
ï»¿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
```

The heart disease dataset attributes are explored for further analysis.

Below are the values:

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 cholesterol 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 fluoroscopy
thal:	thal: 3 = normal; 6 = fixed defect; 7 = reversible defect

Analyzing:

The target variables are analyzed

Analysing the 'target' variable

```
[11] dataset["target"].describe()

count    303.000000
mean       0.544554
std        0.498835
min        0.000000
25%        0.000000
50%        1.000000
75%        1.000000
max        1.000000
Name: target, dtype: float64
```

Correlation of data:

The correlation of data with respect to the attributes is linked and displayed.

```
[13] print(dataset.corr()["target"].abs().sort_values(ascending=False))

target      1.000000
exang       0.436757
cp          0.433798
oldpeak     0.430696
thalach     0.421741
ca          0.391724
slope       0.345877
thal        0.344029
sex         0.280937
ï»¿age     0.225439
trestbps    0.144931
restecg     0.137230
chol        0.085239
fbs         0.028046
Name: target, dtype: float64
```

Exploratory Data Analysis (EDA)

First, analysing the target variable:

```
[ ]  y = dataset["target"]

     sns.countplot(y)


     target_temp = dataset.target.value_counts()

     print(target_temp)

     1    165
     0    138
     Name: target, dtype: int64
```

The target variable is assigned to the y variable.

Implementation with Algorithms:

The testing and training variables are split and passed into the algorithm for heart disease prediction.

Keras model of Deep Learning:

Keras is a brain network Application Programming Connection point for Python that is firmly coordinated with Tensor Stream, which is utilized to construct AI models. Keras models offer a straightforward, easy-to-understand method for characterizing a brain organization, which will then, at that point, be fabricated.

Keras is a strong and simple to-involve free open-source Python library for creating and assessing profound learning models. It is important for the Tensor Flow library and permits you to characterize and prepare brain network models in only a couple of lines of code.

Keras Deep Learning working process:

1. Dataset load
2. Keras Model Definition
3. Keras Model Compilation
4. Keras Model Fit and evaluation.
5. Predictions

```
[43] from sklearn.model_selection import train_test_split

     predictors = dataset.drop("target",axis=1)
     target = dataset["target"]

     X_train,X_test,Y_train,Y_test = train_test_split(predictors,target,test_size=0.20,random_state=0)

[44] X_train.shape

     (242, 13)

[45] X_test.shape

     (61, 13)

[46] Y_train.shape
```

The testing size is 20% and the training size is 80% of the dataset.

Evaluation with Neural Network of Deep Learning with Keras Model:

The possibility of ANNs depends on the conviction that the working of the human cerebrum by making the right associations can be imitated involving silicon and wires as living neurons and dendrites.

The human cerebrum is made out of 80 billion nerve cells called neurons. They are associated with other thousand cells by Axons. Upgrades from outside climate or contributions from tangible organs are acknowledged by dendrites. These data sources make electric motivations, which rapidly travel through the brain organization. A neuron can then send the message to another neuron to deal with the issue or doesn't send it forward.

```
[ ]  from keras.models import Sequential
     from keras.layers import Dense
```

```
[ ]  # https://stats.stackexchange.com/a/136542 helped a lot in avoiding overfitting

     model = Sequential()
     model.add(Dense(11,activation='relu',input_dim=13))
     model.add(Dense(1,activation='sigmoid'))

     model.compile(loss='binary_crossentropy',optimizer='adam',metrics=['accuracy'])
```

Preliminary Results:

The accuracy of the neural network is given below:

```
[76]  score_nn = round(accuracy_score(Y_pred_nn,Y_test)*100,2)

      print("The accuracy score achieved using Neural Network is: "+str(score_nn)+" %")

      #Note: Accuracy of 85% can be achieved on the test set, by setting epochs=2000, and number of nodes = 11.

      The accuracy score achieved using Neural Network is: 83.61 %
```

The final results are compared with different type of algorithm accuracy levels.

```
es = [score_lr,score_nb,score_svm,score_knn,score_dt,score_rf,score_xgb,score_nn]
ithms = ["Logistic Regression","Naive Bayes","Support Vector Machine","K-Nearest Neighbors","Decision Tree","Random Forest","XGBoost","Neural Network"]

i in range(len(algorithms)):
print("The accuracy score achieved using "+algorithms[i]+" is: "+str(scores[i])+" %")

The accuracy score achieved using Logistic Regression is: 85.25 %
The accuracy score achieved using Naive Bayes is: 85.25 %
The accuracy score achieved using Support Vector Machine is: 81.97 %
The accuracy score achieved using K-Nearest Neighbors is: 67.21 %
The accuracy score achieved using Decision Tree is: 81.97 %
The accuracy score achieved using Random Forest is: 90.16 %
The accuracy score achieved using XGBoost is: 78.69 %
The accuracy score achieved using Neural Network is: 83.61 %
```

Conclusion and Future Enhancement:

The heart disease with a different types of algorithms is compared and the accuracy levels were identified and compared. The Deep learning algorithm provides better accuracy levels for the prediction. In the future, the large dataset can be used for the prediction which will help to increase the efficiency of the algorithm in predictions.

Work completed:

• Description

The heart disease dataset containing the information of the heart disease patient details report has been analyzed with Deep learning algorithms and the results are compared with the graphical reports.

References/Bibliography:

[1]. Samiul based shuvo 1, shams nafisa ali 1, (student member, ieee), soham irtiza swapnil1, mabrook s. al-rakhami 2, (senior member, ieee), and abdu gumaei “CardioXNet: A Novel Lightweight Deep Learning Framework for Cardiovascular Disease Classification Using Heart Sound Recordings”. February 24, 2021, date of publication March 2, 2021, date of current version March 9, 2021.IEEE.

[2]. Eun s. lee 1, (member, ieee), byeong g. choi 2, (member, ieee), myung y. kim1, (Member, IEEE), AND SEUNG H. HAN 3, (Member, IEEE)” An Imbalanced-DataProcessingAlgorithmforthe Prediction of Heart Attack in Stroke Patients”. January 14, 2021, date of publication February 8, 2021, date of current version February 16, 2021, IEEE.

[3]. Hao ren2,1∗ aslan b. wong1∗, wanmin lian3∗, weibin cheng2, ying zhang1, jianwei he1. “Interpretable Pneumonia Detection by Combining Deep Learning and Explainable Models with Multisource Data” 10.1109/ACCESS.2021.3090215, IEEE Access.

[4] G. Liang and L. Zheng, “A transfer learning method with deep residual network for pediatric pneumonia diagnosis,” Computer methods and programs in biomedicine, vol. 187, p. 104964, 2020

[5] D. Margaritis, “Learning bayesian network model structure from data,” 2003.

[6] X.-W. Chen, G. Anantha, and X. Lin, “Improving bayesian network structure learning with mutual information-based node ordering in the k2 algorithm,” IEEE Transactions on Knowledge and Data Engineering, vol. 20, no. 5, pp. 628–640, 2008.

[7] S. Watanabe, “A widely applicable bayesian information criterion,” Journal of Machine Learning Research, vol. 14, no. Mar, pp. 867–897, 2013.

[8] J. Yao, J. E. Burns, D. Forsberg, A. Seitel, A. Rasoulian, P. Abolmaesumi, K.Hammernik,M.Urschler,B.Ibragimov,R.Korezetal.,“Amulti-center milestonestudyofclinicalvertebralctsegmentation,”ComputerizedMedical Imaging and Graphics, vol. 49, pp. 16–28, 2016.

[9] H. R. Roth, L. Lu, A. Seff, K. M. Cherry, J. Hoffman, S. Wang, J. Liu, E. Turkbey, and R. M. Summers, “A new 2.5 d representation for lymph node detection using random sets of deep convolutional neural network observations,” in International conference on medical image computing and computer-assisted intervention. Springer, 2014, pp. 520–527.

[10] H. R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E. B. Turkbey, and R.M.Summers,“Deeporgan:Multi-level deep convolutionaln etworksfor automatedpancreassegmentation,”inInternationalconferenceonmedical image computing and computer-assiste dintervention. Springer,2015,pp. 556–564.

[11] L. Yao, E. Poblenz, D. Dagunts, B. Covington, D. Bernard, and K.Lyman,“Learningtodiagnosefromscratchbyexploitingdependencies among labels,” CoRR, vol. abs/1710.10501, 2017. [Online]. Available: http://arxiv.org/abs/1710.10501

[12] , “Learning to diagnose from scratch by exploiting dependencies among labels,” arXiv preprint arXiv:1710.10501, 2017.

[13] https://colab.research.google.com/

[14] https://www.tutorialspoint.com/google_colab/what_is_google_colab.htm

[15] https://www.codingforentrepreneurs.com/courses/python-google-colab-sheets-drive/