

# MINI PROJECT

## GRAPH REPRESENTATION OF DISEASES AND SYMPTOMS

Team Member 1:Poushali Chakrabarty  
Team Member 2:Jadhav Rani  
Team Member3:Ambati Venu Madhav  
Team Member 4:Shobha Rani

Department-Information Technology  
Semester-5

# Introduction

Human population is increasing day by day. But medical professionals are limited in number compared to the huge number of people in need of medical support everyday. It becomes very difficult for the limited number of doctors to give assistance to each and every individual of the huge population.

That's why automated medical diagnosis can be opted as a way out of this crisis.

With the advancement of science and technology automation has indeed played a big role in solving many practical problems, including healthcare.

# Disease Symptom Diagnosis

Diagnosis of disease is often challenging because many signs and symptoms are non specific.

A diagnostic procedure can be regarded as an attempt at classification of an individual's condition into separate and distinct categories that allow medical decisions about treatment and programs to be made.

# Types of Diagnosis

There are mainly three methods, used for disease diagnosis.

- Diagnostic criteria
- Pattern recognition
- Differential diagnosis

The method of Differential diagnosis is based on finding as many candidates or conditions as possible that can probably cause the symptoms, followed by a process of elimination or at least of rendering the entries more or less probable by further medical tests and other processings aiming to reach the point where only one candidate disease or condition remains as probable. The final result may remain a list of possible conditions ranked in order of probability or sincerity. Such a list is often generated by automated disease diagnosis.

# Computer Based Disease Diagnosis

Computer-aided diagnosis in medical field can be viewed as cutting edge intelligence systems in the interface of medicines and computer science. Such systems in medicine may use diagnostic rules to emulate the way a skilled human expert makes diagnosis. More advanced diagnostic systems have the capability to analyze clinical data and infer new knowledge. This new knowledge will in turn help in better disease diagnosis.

# Possible Approaches for Computer Based Diagnosis

- ❑ Graph Theory:

Example:- Using Knowledge graph, Semantic network, Linked data, Optimisation algorithm

- ❑ Machine Learning:

Example:- Regression model, Baseline model, Decision tree, Random forest

- ❑ Deep Learning:

Example:- Convolutional Neural Network, Recurrent Neural Network

- ❑ Artificial Intelligence:

Example:- Support Vector Machines, Discriminant analysis

- ❑ Big Data:

Example:- Apple watch, FitBit

# Objective of Project

Graphs are widely used as a natural framework that captures interaction between individual elements represented as nodes in a graph. In medical applications specifically nodes can represent individuals in large population, while the edges of the graphs increased association between subjects in an intuitive manner. This representation allows to incorporate the weight of imaging and non imaging information simultaneously.

# Organisation of Report

With the advancement of science and technology, it has become an integral part of medical studies. Disease diagnosis cannot be imagined without these technologies like X-Ray, CT Scan, MRI etc. Still these methods require human doctors to make correct medical decisions and these are not fully automated. Since medical professionals are few in number compared to the huge population, it is very difficult to provide medical help to each and every individual. Also opinions vary between doctors.

To tackle this situation, computerized disease detecting methods are the only solution. It helps doctors to arrive at the correct decision.

The diagnosis opinion in the sense it indicates either degree of abnormality or a continuous or kind of abnormality in a classification.

Automated decision support systems are rule-based systems that are automatically providing solutions to the repetitive management problems.

Differential diagnosis methods can be used to identify the presence of an entity where multiple alternatives are possible and also refers to include the candidate alternatives. This method needs a process of elimination or obtaining information that shrinks the probability of candidate conditions to be negligible.



# Literature Review

List of some previous work in this field are as follows:-

Iliad is an excellent diagnostic system which is used to find the relationships for finding the diseases. This system uses the Bayesian classification to compute the probability for possible detection.

OX plain is a medical decision support system, it generates the ranking for the list of diagnosis which is the most likely diseases yielding the lowest rank.

Hybrid algorithm is used to extract salient features from the huge biological dataset.

Machine Learning algorithm is used for the training set.

SOM is a toolbox, which is used to visualize the dataset and mapping the data from higher dimensional input space into lower dimensional input space.

LAMSTAR network was specially designed for applications to problems involving very large memory that relates to many different categories.

Networks have been successfully applied to make decision, diagnosis and recognition problems in various fields.

# Scope and Contribution

Nowadays Electronic Health Records(EHR) are the predominant way of documenting health care activities.

However the development towards No SQL database systems have highly influenced EHR.

Advantages in graph based databases are much more compared to relational databases, graph databases are much easier to handle,are faster especially at highly connected data and have higher level of connectivity than common relational databases.

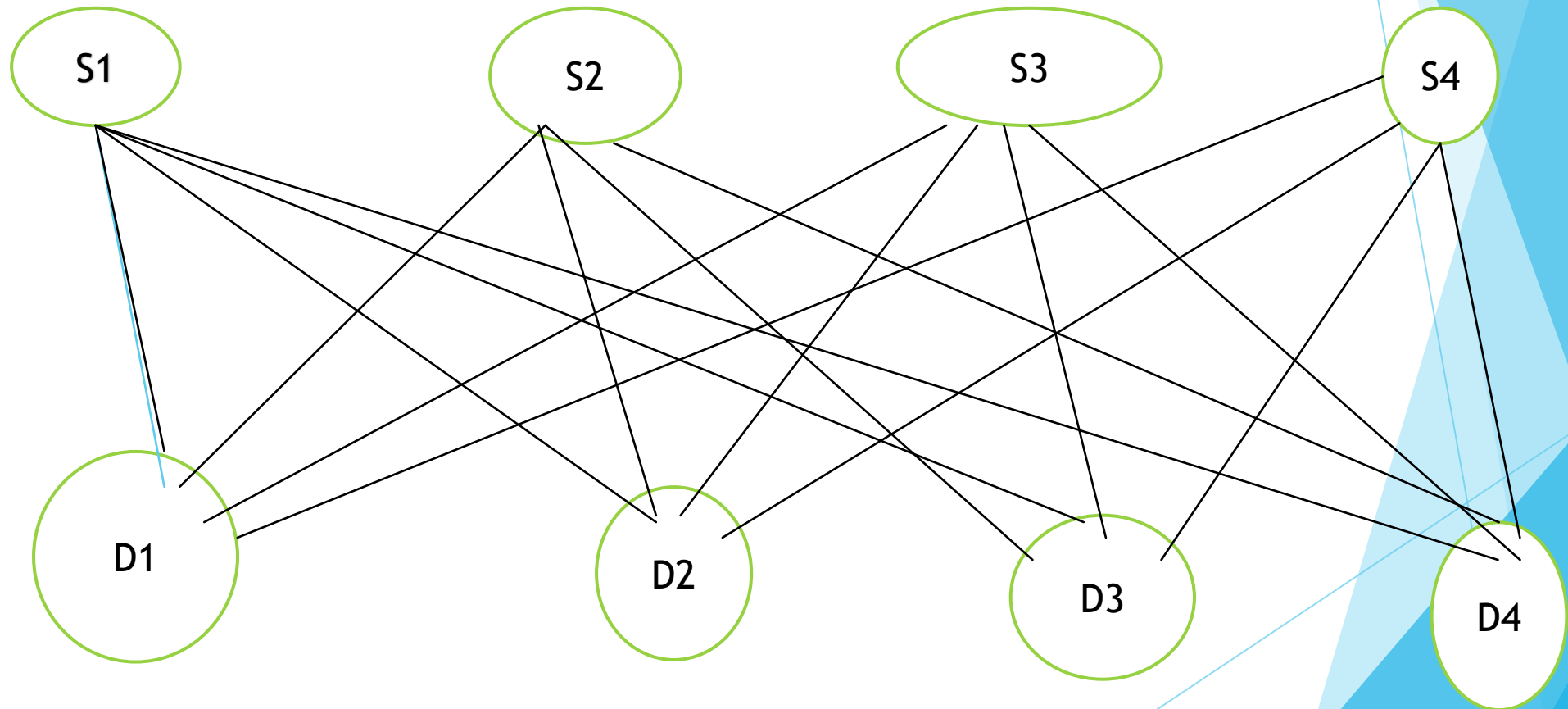
Considering the complexity in a human system, the greater the connectivity, more precise will be the diagnosis.That's why graph databases are preferable over relational databases in the context of medical decision making.

# Proposed Method

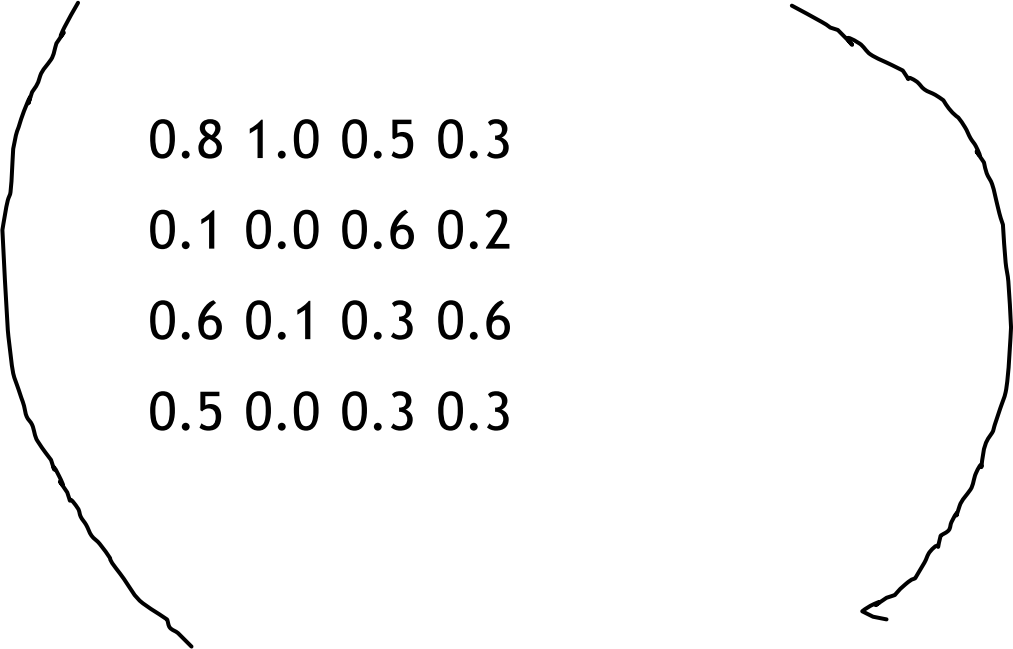
Our method uses graphical representation.

Human physiology is very complex. A particular symptom may occur in human body due to various diseases. So in our approach, we have assumed that each symptom in the dataset may take place due to every disease mentioned in the dataset. We have considered all symptom roots and every diseases its leaf nodes. We have assumed that branch exists between every disease and every symptom.

Diagram of the assumed graphical representation shown for 4 diseases and 4 symptoms

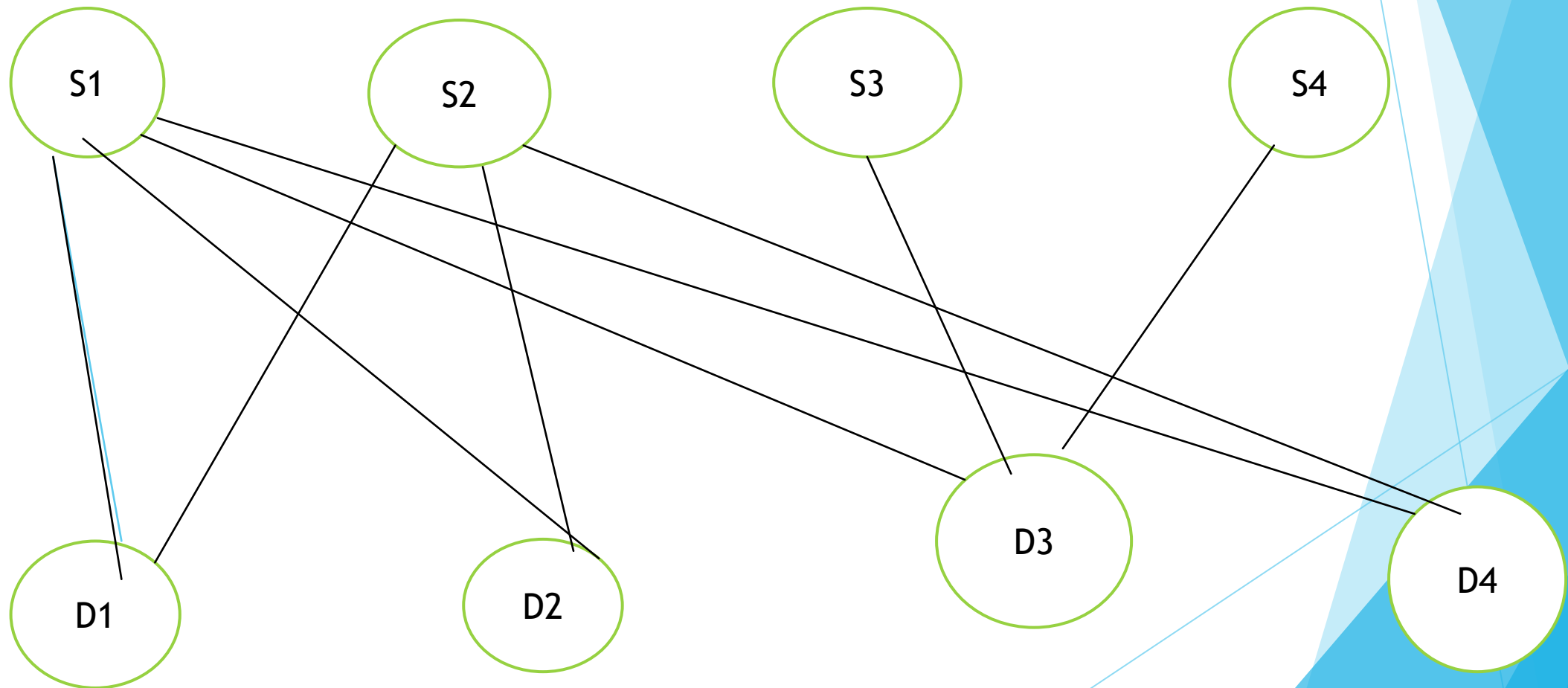


Now since we don't know which of these branches actually exists, we have generated random probabilities for each branch and stored in a matrix. The row represents symptoms and the columns represents diseases. Each element represents the probability of occurrence of a particular symptom due to a particular disease or the probability of existence of that particular branch.

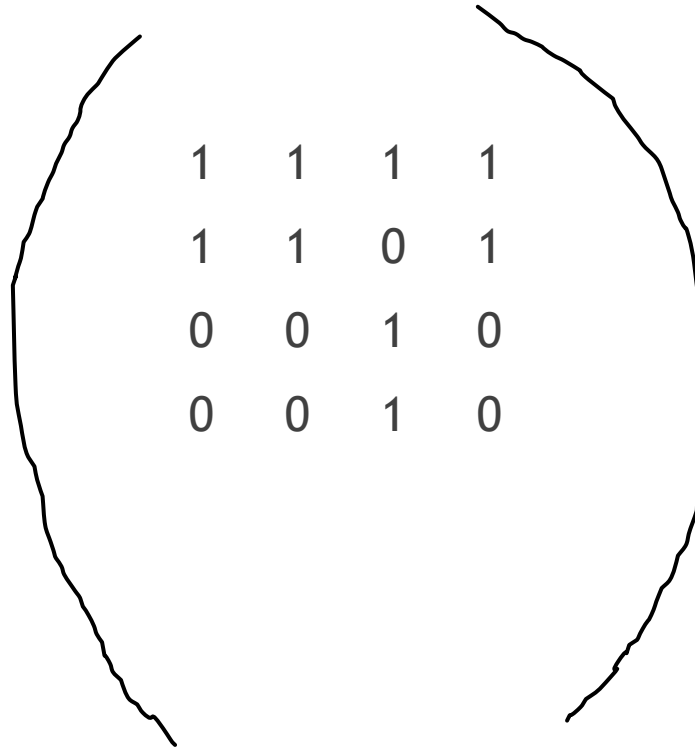


0.8	1.0	0.5	0.3
0.1	0.0	0.6	0.2
0.6	0.1	0.3	0.6
0.5	0.0	0.3	0.3

# Diagram showing Graphical Representation from the Dataset of Diseases and Symptoms



Now, following the graph drawn from the original dataset, which correctly lists the diseases and respective symptoms, we have created another matrix. It stores '1' if a particular Disease-Symptom exists and '0' if that branch doesn't exist. This matrix we have used as ground tool.



1	1	1	1
1	1	0	1
0	0	1	0
0	0	1	0

Then, we have assumed one threshold probability value for checking existence of a particular branch of our assumed model. If any element value of the assumed probability matrix is greater than or equal to threshold value, then we consider that the branch exists, otherwise the branch does not exist. Next, we have compared the results with the groundtool matrix and found probability of success in detection ( $P_s$ ) and probability of failure in detection ( $P_f$ )

Our target is to maximize  $z = (P_s + 1 - P_f)$  to get the best possible results. To achieve this we have followed one iterative algorithm, where the threshold is incremented by 0.2 again and again and the  $z$  value calculated each time. After few iteration  $P_s, P_f, z$  will attain saturation, i.e. incrementing threshold will no longer affect  $z, P_s$  or  $P_f$ . The threshold value for which for the first time saturation is attained is taken as the best possible threshold.

This threshold obtained by using the above algorithm from the training dataset can be used in other testing datasets to detect diseases with high accuracy.



# Code Section

project.py - C:\Users\Poushali\Desktop\DEV\work\Lib\site-packages\project.py (3.7.3)

File Edit Format Run Options Window Help

```
import pandas as pd
import random
import numpy as np
import matplotlib.pyplot as plt

def fun3(m,th,y):
    ar=[]
    for i in range(5):
        arl=[]
        for j in range(25):
            if m[i][j]>th:
                arl.append(1)
            else:
                arl.append(0)
        ar.append(arl)
    ps=[]
    pf=[]
    for i in range(5):
        for j in range(25):
            if y[i][j]==ar[i][j]:
                ps.append(1)
                pf.append(0)
            else:
                ps.append(0)
                pf.append(1)
    z2=sum(ps)/125
    z3=sum(pf)/125
    z=((sum(ps))/125)+1-((sum(pf))/125)
    z4=[]
    z4.append(z)
    z4.append(z2)
    z4.append(z3)

    return z4

def fun(x,n):
    for i in range(len(n)):
        x.append(n[i])
    return x

def fun1(s):
    k=[]
    for i in range(s):
```

```
        k.append((random.randint(0,10))/10)
    return k

d=pd.read_csv('dataset1.csv')
p=d.values.tolist()
np.array(p).reshape(5,9)
n=[]
for i in range(9):
    for j in range(5):
        if p[i][j] not in n:
            n.append(p[i][j])

s=len(n)

x1=[]
x2=[]
x3=[]
x4=[]
x5=[]
k1=[]
k2=[]
k3=[]
k4=[]
k5=[]
x1.append('Fever')
x2.append('Cough')
x3.append('Headache')
x4.append('Sweating')
x5.append('Chest pain')
x1=fun(x1,n)
k1=fun1(s)
x2=fun(x2,n)
k2=fun1(s)
x3=fun(x3,n)
k3=fun1(s)
x4=fun(x4,n)
k4=fun1(s)
x5=fun(x5,n)
k5=fun1(s)
m=[]
```

```

k5=[]
x1.append('Fever')
x2.append('Cough')
x3.append('Headache')
x4.append('Sweating')
x5.append('Chest pain')
x1=fun(x1,n)
k1=fun1(s)
x2=fun(x2,n)
k2=fun1(s)
x3=fun(x3,n)
k3=fun1(s)
x4=fun(x4,n)
k4=fun1(s)
x5=fun(x5,n)
k5=fun1(s)
m=[]
m.append(k1)
m.append(k2)
m.append(k3)
m.append(k4)
m.append(k5)
y=[]
for i in range(5):
    y1=[]
    for j in range(25):
        y1.append(0)
    y.append(y1)
np.array(p)
p=np.array(p).reshape(5,9)
for j in range(9):

    for i in range(5):
        for k in range(25):
            if n[k] in p[i][j]:
                y[i][k]=1

zs=[]
zf=[]
zm=[]
zt=[]
z=0

```

File Edit Format View Options Window Help

```

for j in range(9):

    for i in range(5):
        for k in range(25):
            if n[k] in p[i][j]:
                y[i][k]=1

zs=[]
zf=[]
zm=[]
zt=[]
z=0
z1=0
thh=[]
th=0.1
thh.append(th)
zt=fun3(m,th,y)
z=zt[0]
zs.append(zt[1])
zf.append(zt[2])
t2=(z-z1)

z1=z
th=th+0.2
thh.append(th)
zt=fun3(m,th,y)

z=zt[0]
zs.append(zt[1])
zf.append(zt[2])
t2=z-z1
th=th+0.2
while t2>=0.001 and th<=1:

    zt=[]
    z1=z

    thh.append(th)
    zt=fun3(m,th,y)
    z=zt[0]

    zs.append(zt[1])

```

```

z1=z
th=th+0.2
thh.append(th)
zt=fun3(m,th,y)

z=zt[0]
zs.append(zt[1])
zf.append(zt[2])
t2=z-z1
th=th+0.2
while t2>=0.001 and th<=1:

    zt=[]
    z1=z

    thh.append(th)
    zt=fun3(m,th,y)
    z=zt[0]

    zs.append(zt[1])

    zf.append(zt[2])
    t2=(z-z1)/10
    th=th+0.2

print("Probability of Successful detection ",zs)
print("Probability of Failure in Detection ",zf)

plt.plot(thh,zs,label='Plot showing probability of success in detection with increasing threshold')
plt.xlabel("Threshold->")

plt.plot(thh,zf,label='Plot showing probability of failure in detection with increasing threshold')
plt.xlabel("Threshold ")
plt.ylabel("Probability of success/failure in; detection")
plt.legend()
plt.show()
thre=thh[len(thh)-1]
print(thre)
# 'thre' is the required threshold. If any branch probability value is greater than or equal to 'thre' then it exists, else it doesn't exist

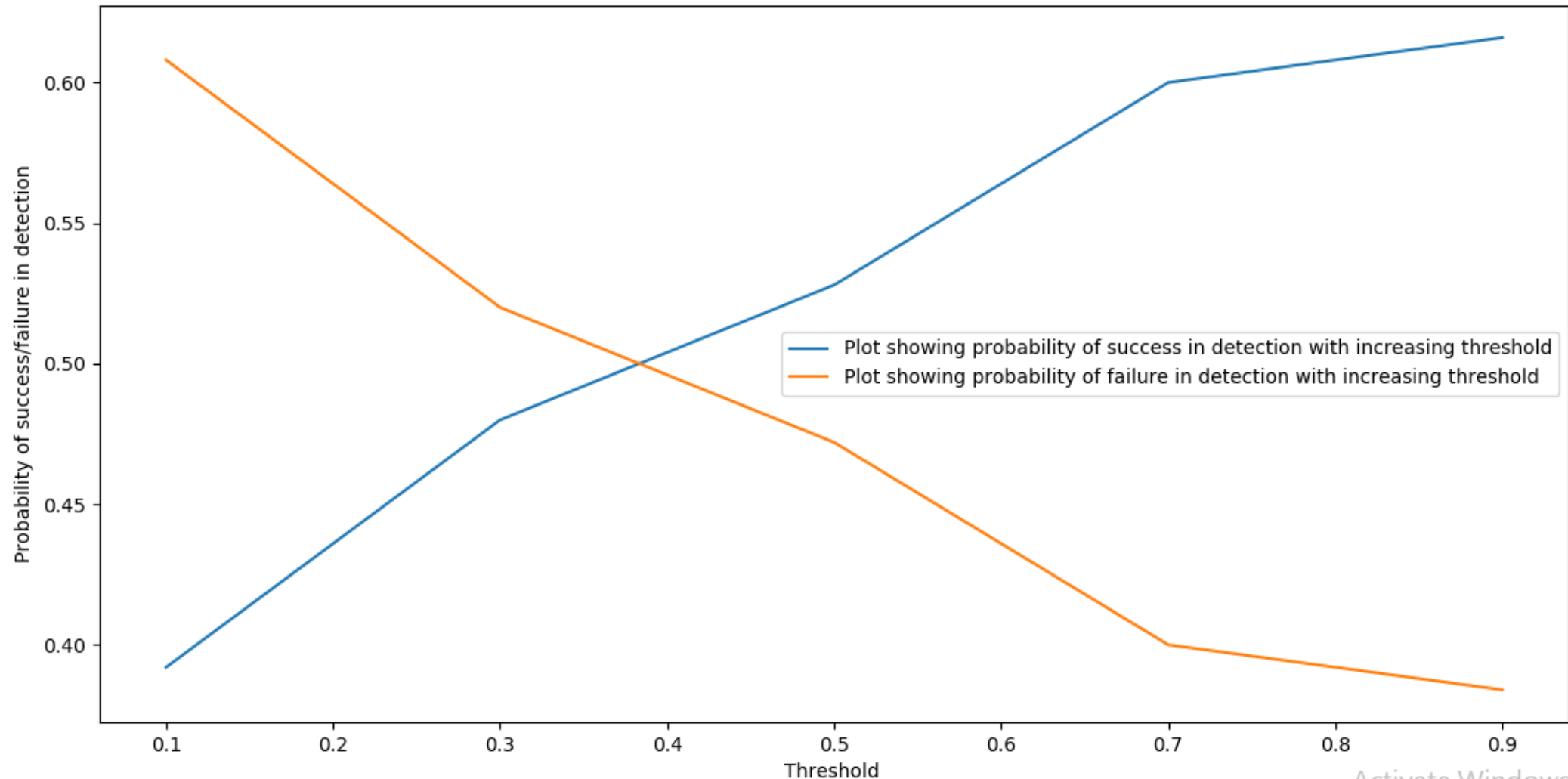
```

# Simulation Results

On running the code we are getting the following results.

```
===  
== RESTART: C:\Users\Poushali\Desktop\DEV\work\Lib\site-packages\project.py ==  
Probability of Successful detection  [0.376, 0.448, 0.568, 0.6, 0.656]  
Probability of Failure in Detection [0.624, 0.552, 0.432, 0.4, 0.344]  
Best Threshold 0.8999999999999999  
>>> |
```

# Plot Showing Probability of Success/Failure with increasing Threshold



From the output, we can see that as the threshold value increases, the probability of successful detection of a disease ( $P_s$ ) increases. It goes on increasing with the increasing threshold till it attains saturation. Similarly, probability of failure in detecting a disease ( $P_f$ ) decreases with decreasing threshold, till it attains saturation.

The threshold for which  $P_s, P_f$  attains saturation for the first time is the optimized threshold which can be further used in any other dataset to detect diseases with accuracy.

# Conclusion and Future Scope

The usage of various modern clinical facilities to diagnose the disease by the medical practitioners are the demands of the days. Graph convolution followed by deep learning determines the most probable disease associated with the given symptom with high accuracy.

Training the models with larger datasets having more variety will increase its accuracy in prediction. Sometimes diseases with similar symptoms creates confusing in predicting the disease correctly. Such problems can be solved through experience. More the training with diverse datasets the better will be its accuracy in predicting diseases.

# Some References

- ▶ Ali, L., Rahman, A., Khan, A., Zhou, M., Javeed, A., & Khan, J. A. (2019).  
▶ An automated diagnostic system for heart disease prediction based on 2  
▶ statistical model and optimally configured deep neural network. IEEE  
▶ Access, 7, 34938{34945.
- ▶ Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., & Vandergheynst, P.  
▶ (2017). Geometric deep learning: going beyond euclidean data. IEEE  
▶ Signal Processing Magazine, 34 (4), 18{42.
- ▶ De Fauw, J., Ledsam, J. R., Romera-Paredes, B., Nikolov, S., Tomasev, N.,  
▶ Blackwell, S., . . . others (2018). Clinically applicable deep learning for  
▶ diagnosis and referral in retinal disease. Nature Medicine, 24 (9), 1342{  
▶ 1350.
- ▶ Geras, K. J., Wolfson, S., Shen, Y., Wu, N., Kim, S., Kim, E., . . . Cho, K.  
▶ (2017). High-resolution breast cancer screening with multi-view deep  
▶ convolutional neural networks. arXiv preprint arXiv:1703.07047 .
- ▶ Hao, Y., Usama, M., Yang, J., Hossain, M. S., & Ghoneim, A. (2019). Recurrent  
▶ convolutional neural network based multimodal disease risk prediction.  
▶ Future Generation Computer Systems, 92, 76{83.



**THANK YOU**