

Lymphography Classification Using ML

Date	16-Nov-23
Team ID	Team 591563
Project Name	Lymphography Classification Using ML

Project Description

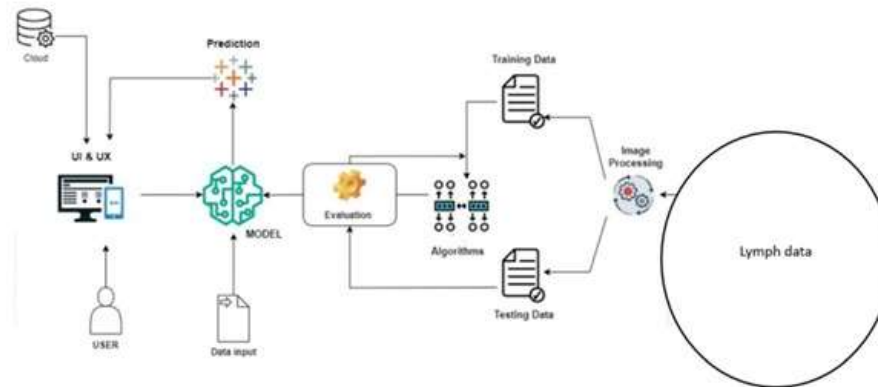
Lymph nodes play a pivotal role in our immune system, resembling small, oval-shaped organs scattered throughout the body. Their size can dynamically change due to various factors, ranging from minor issues to severe conditions like cancer. Monitoring the condition of lymph nodes is crucial, as it serves as a reliable indicator of the stage of cancer progression, ultimately influencing the choice of treatment.

Enter lymphography, a sophisticated medical imaging technique that involves injecting a contrast agent and capturing X-ray images. This process allows medical professionals to visualize and examine the intricate structures of the lymphatic system, including lymph nodes, ducts, tissues, capillaries, and vessels. The data obtained from lymphography becomes instrumental in determining whether the observed clinical details point towards a normal or abnormal state.

Notably, the current status of lymph nodes, as revealed by lymphography, holds significant diagnostic value. It can provide valuable clues about the potential presence of cancer, aiding in early detection and intervention. While the procedure of lymphography may pose certain challenges, the information derived from the images allows for precise and accurate assessments of the state of lymph nodes, ducts, and capillaries.

Effectively classifying and identifying relevant attributes from the lymphographic data simplifies the complex task of predicting diseases. This, in turn, facilitates the prompt implementation of preventive measures, underscoring the importance of this medical imaging technique in the broader context of healthcare.

Technical Architecture



Prerequisites:

To complete this project, you must require following software's, concepts and packages

- Anaconda navigator and pycharm:
 - o Refer the link below to download anaconda navigator
 - o Link : <https://youtu.be/1ra4zH2G4o0>
- Python packages:
 - o Open anaconda prompt as administrator
 - o Type “pip install numpy” and click enter.
 - o Type “pip install pandas” and click enter.
 - o Type “pip install scikit-learn” and click enter.
 - o Type ”pip install matplotlib” and click enter.
 - o Type ”pip install scipy” and click enter.
 - o Type ”pip install pickle-mixin” and click enter.
 - o Type ”pip install seaborn” and click enter.
 - o Type “pit0p install Flask” and click enter.
 - o Type “pip install more-intertools ” and click enter.

Prior Knowledge

You must have prior knowledge of the following topics to complete this project.

- **ML Concepts**

Supervised and Unsupervised Machine Learning:

<https://www.analyticsvidhya.com/blog/2020/04/supervised-learning-unsupervised-learning/>

Random Forest:

<https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/>

<https://www.javatpoint.com/machine-learning-random-forest-algorithm>

Evaluation Metrics:

<https://www.analyticsvidhya.com/blog/2021/07/metrics-to-evaluate-your-classification-model-to-take-the-right-decisions/>

Machine Learning Model:

<https://www.geeksforgeeks.org/metrics-for-machine-learning-model/>

Flask Basics:

<https://www.geeksforgeeks.org/flask-tutorial/>

https://www.youtube.com/watch?v=lj4I_CvBn

Deploying the Model:

<https://medium.com/shapeai/deploying-flask-application-with-ml-models-on-aws-ec2-instance-3b9a1cec5e13>

<https://youtu.be/ct1GbTvgVNM?si=MbLhGNVNGuCJ2COW>

Project Objectives:

- Know fundamental concepts and techniques used for machine learning.
- Gain a broad understanding about data.
- Have knowledge on pre-processing the data/transformation techniques
- Measuring the appropriate evaluation metrics for model performance.
- Hands-on experience with web framework such as flask.
- Hands-on experience with deploying an ML model in Cloud.

Project Flow:

- User interacts with the UI to enter the input.
- Entered input is analyzed by the model which is integrated.
- Once model analyses the input the prediction is showcased on the UI

To accomplish this, we have to complete all the activities listed below,

• Data collection

- Collect the dataset or create the dataset

• Visualizing and analyzing data

- Descriptive analysis
- Statistical Analysis

• Data pre-processing

- Checking for null values
- Handling outlier
- Encoding
- Feature Scaling
- Splitting data into train and test

• Model building

- Import the model building libraries
- Initializing the model
- Training and testing the model
- Evaluating performance of model
- Save the model

• Application Building

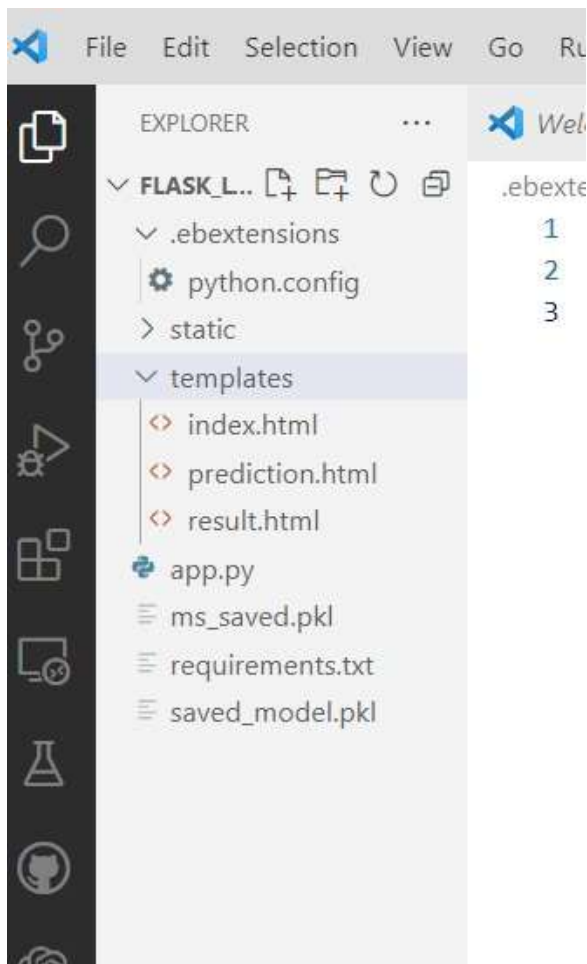
- Create an HTML file
- Add CSS
- Build python code

- **Deployment of model**

- Launching an EC2 Instance with Ubuntu AMI
- Installing Necessary Libraries in Ubuntu
- Uploading the required folders using Firezilla into Ubuntu
- Run app.py in Ubuntu
- Get the DNS Address

Project Structure:

Create the Flask folder as shown below;



- We are building a flask application which needs HTML pages stored in the templates folder and a python script app.py for scripting.

ms_saved.pkl is our saved Min-Max scalar.

- saved_model.pkl is our saved model. Further we will use this model for flask integration.

Milestone 1: Data Collection

The success of machine learning (ML) hinges significantly on the availability of relevant data, as it constitutes the essential foundation for training algorithms. This section provides the opportunity to acquire and download the necessary dataset for facilitating algorithmic training.

Activity 1: Download the dataset;

There are many popular open sources for collecting the data. Eg: kaggle.com, UCI repository, etc. In this project we have used lymphography dataset. This data is taken from UCI repository. We can either download the dataset or we can use the link to get the dataset in our project. Please refer the link given below incase if we want to download the dataset.

Link:

<https://archive.ics.uci.edu/ml/machinelearningdatabases/lymphography/lymphography.data>

Milestone 2: Visualizing and analyzing the data

As the dataset is downloaded or inherited. Let us read and understand the data properly with the help of some visualization techniques and some analysing techniques.

Note: There is n number of techniques for understanding the data. But here we have used some of it. In an additional way, you can use multiple techniques.

Activity 1: Importing the libraries

Import the necessary libraries as shown below:

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4 import seaborn as sns
```

```

1 from sklearn.preprocessing import StandardScaler
2 from sklearn.model_selection import train_test_split
3 from sklearn.ensemble import RandomForestClassifier
4 from sklearn.model_selection import GridSearchCV
5 from sklearn.metrics import accuracy_score, f1_score, confusion_matrix, classification_report
6 from sklearn.metrics import accuracy_score, f1_score, confusion_matrix, classification_report

```

Activity 2: Read the Dataset

Our dataset format might be in .csv, excel files, .txt, .json, etc. We can read the dataset with the help of pandas.

In pandas we have a function called `read_csv()` to read the dataset. As a parameter we have to give the directory of csv file.

But first the column names are changed according to our convenience as the column names in dataset are not understandable.

Note that the number of column names that you give should match with the number of columns in the dataset.

```

1 col_names = ['class', 'lymphatics', 'block of affere', 'bl. of lymph. c',
2              'bl. of lymph. s', 'by pass', 'extravasates', 'regeneration of',
3              'early uptake in', 'lym.nodes dimin', 'lym.nodes enlar',
4              'changes in lym.', 'defect in node', 'changes in node',
5              'changes in stru', 'special forms', 'dislocation of',
6              'exclusion of no', 'no. of nodes in']
7

```

```
1 df.head()
```

	class	lymphatics	block of affere	bl. of lymph. c	bl. of lymph. s	by pass	extravasates	regeneration of	early uptake in	lym.nodes dimin	lym.nodes enlar	changes in lym.	defect in node	changes in node	changes in stru	special forms	dislocation of	exclusion of no	no. of nodes in
0	3	4	2	1	1	1	1	1	2	1	2	2	2	4	8	1	1	2	2
1	2	3	2	1	1	2	2	1	2	1	3	3	2	3	4	2	2	2	2
2	3	3	2	2	2	2	2	2	2	1	4	3	3	4	8	3	2	2	7
3	3	3	1	1	1	1	2	1	2	1	3	3	4	4	4	3	1	2	6
4	2	3	1	1	1	1	1	1	1	1	2	2	4	3	5	1	2	2	1

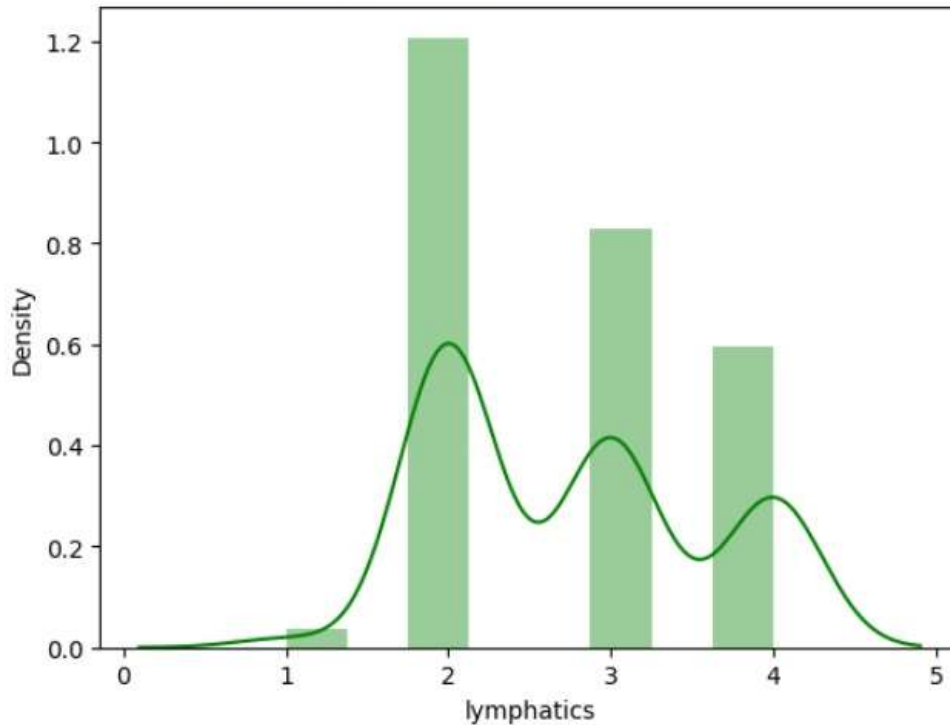
Here, Head prints the top 5 records in the dataset.

Activity 3: Univariate analysis

In simple words, univariate analysis is understanding the data with single feature. Here we have displayed two different graphs such as dist plot and count plot.

- Seaborn package provides a wonderful function dist plot. With the help of dist plot, we can find the distribution of the feature. To make multiple graphs in a single plot, we use subplot

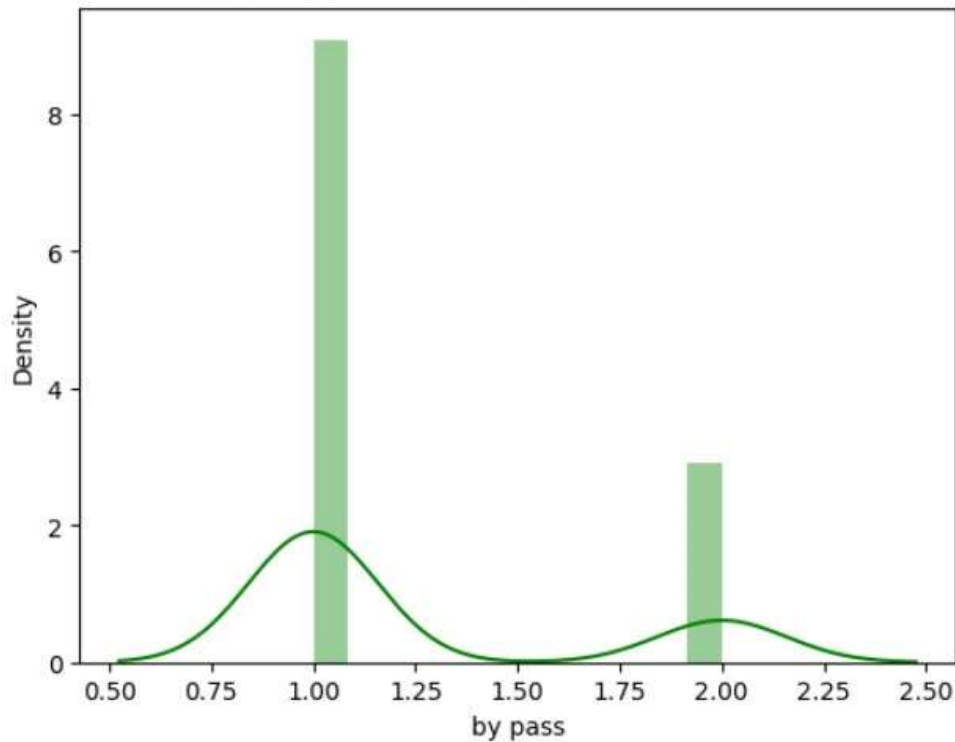
```
sns.distplot(df["lymphatics"],color='g')  
<Axes: xlabel='lymphatics', ylabel='Density'>
```



Activity 4: Bivariate analysis

Count plot is used here. As a 1st parameter we are passing x value and as a 2nd parameter we are passing hue value.


```
sns.distplot(df["by pass"],color='g')
<Axes: xlabel='by pass', ylabel='Density'>
```



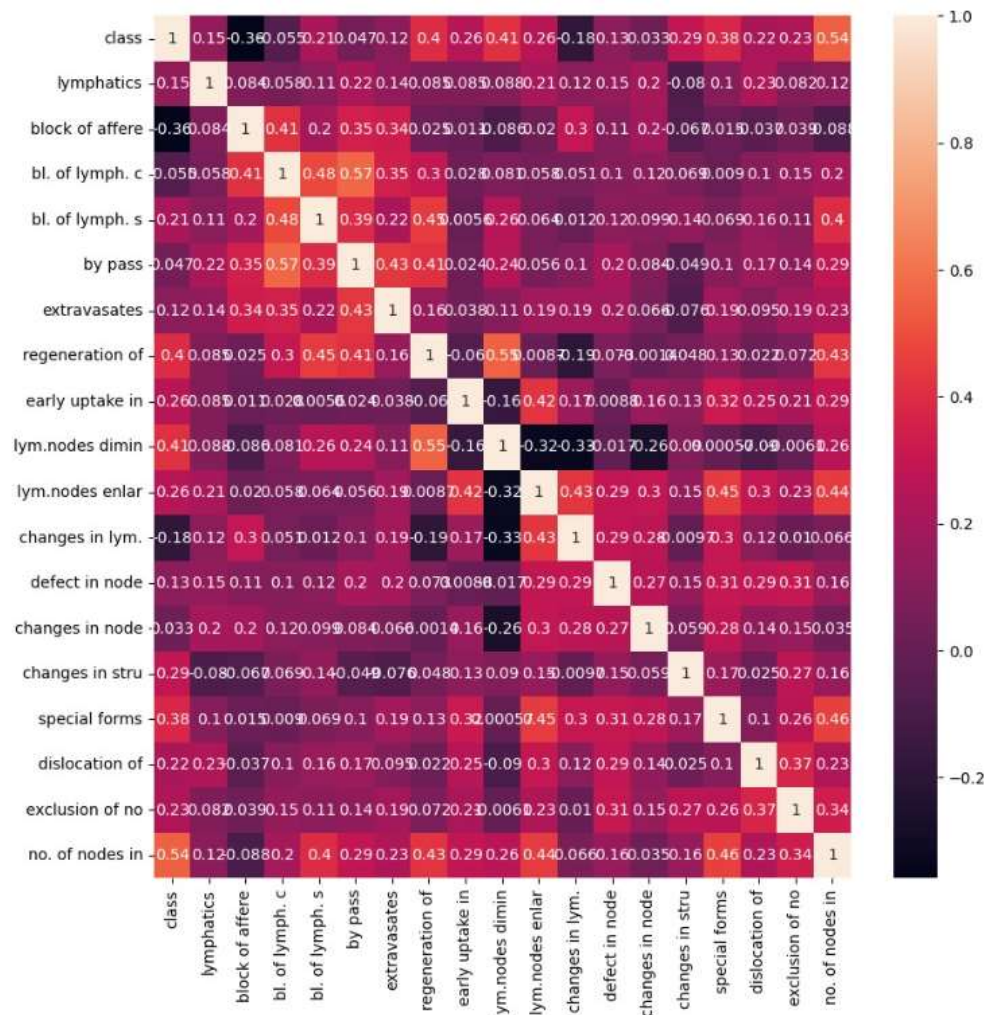
Activity 3: Descriptive analysis

Descriptive analysis is to study the basic features of data with the statistical process. Here pandas has a worthy function called describe. With this describe function we can understand the unique, top and frequent values of categorical features. And we can find mean, std, min, max and percentile values of continuous features.

```
1 df.describe()
```

	class	lymphatics	block of affere	bl. of lymph. c	bl. of lymph. s	by pass	extravasates	regeneration of	early uptake in	lym-nodes dimin	lym-nodes enlar	changes in lym.	defect in node	changes in node	changes in stru	special foras	dislocation of	exclusion of no	no. of nodes in
count	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000	148.000000
mean	2.452703	2.743243	1.654054	1.175676	1.047297	1.243243	1.506757	1.067568	1.702703	1.060811	2.472973	2.398649	2.966216	2.804054	5.216216	2.331081	1.662162	1.790541	2.601351
std	0.575396	0.817509	0.498757	0.381836	0.212995	0.430498	0.581652	0.251855	0.458621	0.313557	0.836627	0.566323	0.868395	0.761834	2.171368	0.777126	0.474579	0.408305	1.905023
min	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
25%	2.000000	2.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	2.000000	2.000000	2.000000	2.000000	4.000000	2.000000	1.000000	2.000000	1.000000
50%	2.000000	3.000000	2.000000	1.000000	1.000000	1.000000	2.000000	1.000000	2.000000	1.000000	2.000000	2.000000	3.000000	3.000000	5.000000	3.000000	2.000000	2.000000	2.000000
75%	3.000000	3.000000	2.000000	1.000000	1.000000	1.000000	2.000000	1.000000	2.000000	1.000000	3.000000	3.000000	4.000000	3.000000	8.000000	3.000000	2.000000	2.000000	3.000000
max	4.000000	4.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	3.000000	4.000000	3.000000	4.000000	4.000000	8.000000	3.000000	2.000000	2.000000	8.000000

Activity 4: Correlation



Milestone 3: Data Pre-processing

As we have understood how the data is let's pre-process the collected data. The download data set is not suitable for training the machine learning model as it might have so much randomness, so we need to clean the dataset properly in order to fetch good results. This activity includes the following steps.

- Handling missing values
- Feature Scaling
- Splitting dataset into training and test set

Note: These are the general steps of pre-processing the data before using it for machine learning. Depending on the condition of your dataset, you may or may not have to go through all these steps.

Activity 1: Checking for null values

- Let's find the shape of our dataset first, To find the shape of our data, df.shape method is used. To find the data type, df.inull() function is used.

```
1 df.isnull().sum()
```

```
class      0
lymphatics  0
block of affere  0
bl. of lymph. c  0
bl. of lymph. s  0
by pass    0
extravasates  0
regeneration of  0
early uptake in  0
lym.nodes dimin  0
lym.nodes enlar  0
changes in lym.  0
defect in node  0
changes in node  0
changes in stru  0
special forms  0
dislocation of  0
exclusion of no  0
no. of nodes in  0
dtype: int64
```

Activity 2: Feature Scaling

We have to normalize all the column values between 0 to 1 to get accurate results.

```
1 #feature scaling
2 from sklearn.preprocessing import MinMaxScaler
3 ms=MinMaxScaler()
4 x_scaled=pd.DataFrame(ms.fit_transform(x),columns=x.columns)
```

```
1 x_scaled
2
```

	lymphatics	block of afferents	bl. of lymph. c	bl. of lymph. s	by pass	extravasates	regeneration of	early uptake in	lym.nodes dimin	lym.nodes enlar	changes in lym.	defect in node	changes in node	changes in stru	special forms	dislocation of	exclusion of no	no. of nodes in
0	1.000000	1.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.333333	0.5	0.333333	1.000000	1.000000	0.0	0.0	1.0	0.142857
1	0.666667	1.0	0.0	0.0	1.0	1.0	0.0	1.0	0.0	0.666667	1.0	0.333333	0.666667	0.428571	0.5	1.0	1.0	0.142857
2	0.666667	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	1.000000	1.0	0.666667	1.000000	1.000000	1.0	1.0	1.0	0.857143
3	0.666667	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.666667	1.0	1.000000	1.000000	0.428571	1.0	0.0	1.0	0.714286
4	0.666667	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.333333	0.5	1.000000	0.666667	0.571429	0.0	1.0	1.0	0.000000
...

Activity 3: Splitting into Training and Testing Data

Now let's split the Dataset into train and test sets. First split the dataset into x and y and then split the data set Here x and y variables are created. On x variable, df is passed with dropping the target variable. And on y target variable is passed. For splitting training and testing data we are using train_test_split() function from sklearn. As parameters, we are passing x, y, test_size, random_state.

```
27] 1 #Splitting Data into Train and Test.
     2 from sklearn.model_selection import train_test_split
     3 x_train,x_test,y_train,y_test=train_test_split(x_scaled,y,test_size=0.2,random_state=0)

28] 1 x_train.shape,x_test.shape,y_train.shape,y_test.shape

    ((118, 18), (30, 18), (118,), (30,))
```

Milestone 4: Model Building

Now our data is cleaned and it's time to build the model. We can train our data . For this project we are applying Random Forest Classification.

```
[30] 1 from sklearn.ensemble import RandomForestClassifier
     2 rf=RandomForestClassifier()
```

Hyperparameter Tuning:

```
[31] 1 params={  
2     'max_depth':[9,10,11],  
3     'min_samples_leaf':[2,3],  
4     'n_estimators':[90,95,100,110],  
5     'max_features':[2,3,4,5]  
6 }
```

```
1 from sklearn.model_selection import GridSearchCV
```

```
[33] 1 grid_search=GridSearchCV(estimator=rf,  
2                               param_grid=params,  
3                               cv=2,  
4                               verbose=1,  
5                               scoring="accuracy")
```

```
[34] 1 grid_search.fit(x_train,y_train)
```

Fitting 2 folds for each of 96 candidates, totalling 192 fits

```
GridSearchCV  
estimator: RandomForestClassifier  
RandomForestClassifier
```

```
[35] 1 grid_search.best_score_
```

0.788135593220339

```
[36] 1 rf_best=grid_search.best_estimator_  
2 rf_best
```

```
RandomForestClassifier  
RandomForestClassifier(max_depth=10, max_features=2, min_samples_leaf=3)
```

Training the model with best parameters:

```
1 rf_classify=RandomForestClassifier(random_state=42,  
2                                   n_jobs=-1,  
3                                   max_depth=9,  
4                                   min_samples_split=2,  
5                                   max_features='sqrt',  
6                                   n_estimators=90,  
7                                   bootstrap=True)
```

```
[38] 1 rf_classify.fit(x_train,y_train)
```

Activity 2: Evaluating performance of the model and saving the model

We define the confusion matrix function description. The matrix shows the relation between the predicted and the actual values of output for the given input.

Confusion Matrix:

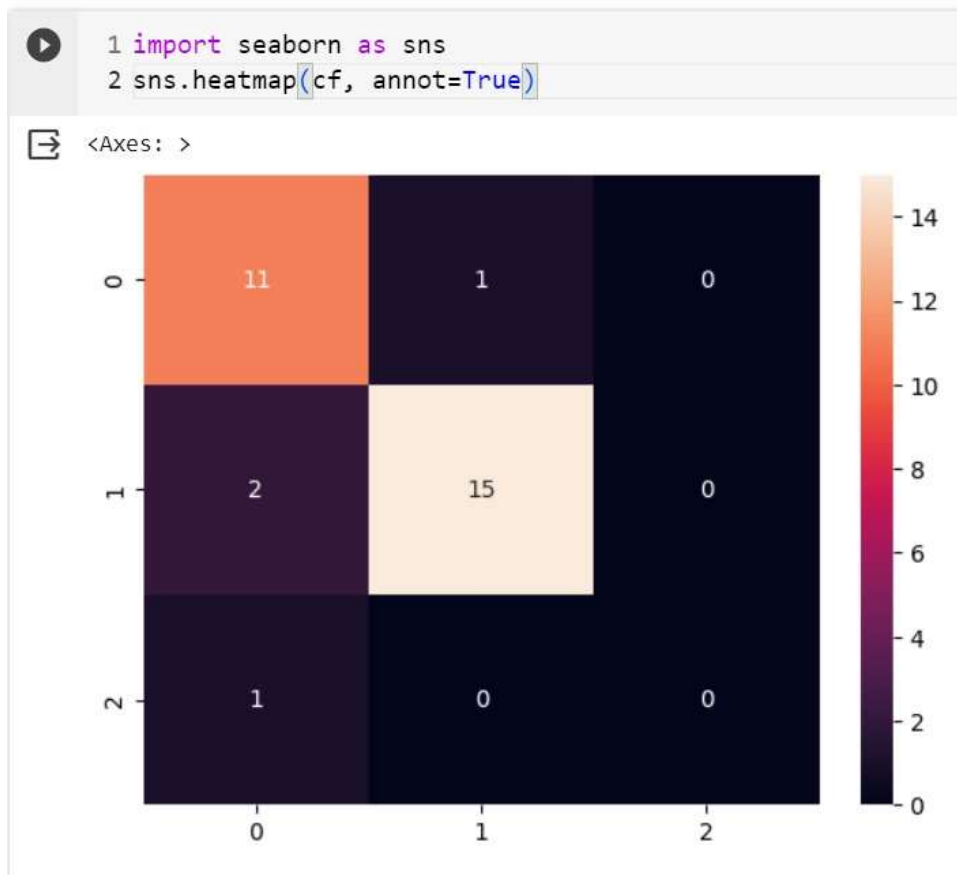
The function is called and executed the confusion matrix function block. The output is displayed as below:

```
[39] 1 from sklearn.metrics import accuracy_score
```

```
[40] 1 prediction=rf_classify.predict(x_test)
```

```
[41] 1 from sklearn.metrics import accuracy_score,f1_score,confusion_matrix,classification_report
```

```
1 confusion_matrix(y_test,prediction)
```



Model Accuracy:

Model accuracy is defined as the number of classifications a model correctly predicts divided by the total number of predictions made. It's a way of assessing the performance of a model, but certainly not the only way.

\

```
[43] 1 accuracy_score(y_test, prediction)
```

```
0.8666666666666667
```


Classification Report:

```
4] 1 print(classification_report(y_test,prediction))
```

	precision	recall	f1-score	support
2	0.79	0.92	0.85	12
3	0.94	0.88	0.91	17
4	0.00	0.00	0.00	1
accuracy			0.87	30
macro avg	0.57	0.60	0.59	30
weighted avg	0.85	0.87	0.85	30

Activity 3: Saving the Model

The pickle module keeps track of the objects it has already serialized, so that later references to the same object won't be serialized again, thus allowing for faster execution time. Allows saving model in very little time

The model here is saved as saved_model.pkl

```
1 import pickle
```

```
[46] 1 pickle.dump(rf_classify,open('saved_model.pkl','wb'))
```

```
[47] 1 from joblib import dump,load  
2 dump(sc,'std_scaler.bin',compress=True)
```

```
['std_scaler.bin']
```

```
[48] 1 import pickle  
2 pickle.dump(sc,open('sc_saved.pkl','wb'))
```

Milestone 5: Application Building

In this section, we will be building a web application that is integrated to the model we built. A UI is provided for the uses where he has to enter the values for predictions. The enter values are given to the saved model and prediction is showcased on the UI.

This section has the following tasks

- Building HTML Pages
- Building serverside script

Activity1: Building Html Pages:

For this project create HTML files namely

- index.html
- prediction.html
- result.html

and save them in templates folder

index.html:

```
> Users > pichi > OneDrive > Documents > Desktop > < > index (8).html > ...
1 |<!DOCTYPE html>
2 |<html lang="en">
3 |
4 |<head>
5 |   <meta charset="UTF-8">
6 |   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7 |   <title>Document</title>
8 |   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
9 |   <style>
10 |     body {
11 |       padding: 1%;
12 |       margin: 2%;
13 |       box-sizing: border-box;
14 |       background-color: rgba(255, 228, 181, 0.932);
15 |     }
16 |
17 |     #main-heading {
18 |       font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
19 |       text-align: center;
20 |       background-color: white;
21 |       color: black;
22 |       padding: 2%;
23 |       justify-content: center;
24 |       border-radius: 2rem;
25 |     }
26 |
27 |     #photos {
28 |       padding: 2%;
29 |       border-radius: 2rem;
30 |     }
31 |
32 |     #desc {
33 |       background-color: white;
34 |       text-align: justify;
35 |       padding: 1%;
36 |       font-size: medium;
37 |       font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
38 |       line-height: 1.5rem;
39 |       border-radius: 2rem;
40 |     }
41 |   </style>
42 | </html>
```

```
.Outer {
  display: flex;
  gap: 0.2%;
  align-items: center;
  justify-content: space-around;
  overflow-x: auto;
  margin-bottom: 20px;
}

h3 {
  text-align: center;
}

#pred {
  padding-left: 5%;
  height: 100%;
}

#pred .card {
  height: 100%;
  display: flex;
  flex-direction: column;
}

#pred img {
  height: auto;
  max-height: 100%;
  object-fit: cover;
}

#pred .card-body {
  flex-grow: 1;
}

p {
  text-align: justify;
}

.card {
  display: flex;
  flex-direction: column;
  height: 100%;
}
```

sers > pichi > OneDrive > Documents > Desktop > index (8).html > ...

```
.card img {
  height: 30vh;
  object-fit: cover;
}

.card:hover {
  transform: scale(1.09);
  transition: transform 0.3s ease;
}
</style>
</head>

<body>
  <div id="main-heading">
    <h2>Lymphography Classifier</h2>
  </div>

  <br>
  <br>

  <div class="container">
    <div class="row row-cols-1 row-cols-md-3 g-4 align-items-stretch">
      <div class="col">
        <div class="card">
          
          <div class="card-body">
            <h3>Class A</h3>
            <p class="card-text">Malignant lymphoma is a cancer originating in the lymphatic system, marked by
              uncontrolled growth of abnormal lymphocytes.</p>
          </div>
        </div>
      </div>
      <div class="col">
        <div class="card">
          
          <div class="card-body">
            <h3>Class B</h3>
            <p class="card-text">Fibrosis is the buildup of excess fibrous tissue in organs, disrupting
              function due to scarring from chronic inflammation or injury.</p>
          </div>
        </div>
      </div>
    </div>
  </div>
```

Jsers > pichii > OneDrive > Documents > Desktop > <> index(8).html > html > body > div.container > div.row > div#pred.col-sm-4 > div.card > img.card-img-top

```
<div class="col">
  <div class="card">
    
    <div class="card-body">
      <h3>Class C</h3>
      <p class="card-text">Metastasis refers to the spread of cancer cells from their original
        (primary) site to other parts of the body, forming secondary tumors.</p>
    </div>
  </div>
</div>
</div>
```

```
<br>
<br>
```

```
<div class="container">
  <div class="row">
    <div class="col-sm-8" id="desc">
      <br>
      <h3>Lymphography</h3>
      <br>
      <p>A lymph node is an oval-shaped organ of the immune system, distributed widely throughout the body.
        They tend to expand in size for diverse reasons, indicating health complications that scale from
        trivial, to life-threatening ailments such as cancers.</p><br>
      <p>The condition of lymph nodes is so significant that it is used to accurately sense the stage in Cancer
        progression, which decides the treatment to be adopted. Lymphography is a medical imaging technique
        in which a radio contrast agent is injected, and then an X-ray picture is taken to visualize
        structures of the lymphatic system, including lymph nodes, lymph ducts, lymphatic tissues, lymph
        capillaries and lymph vessels. This data is necessary to decide on whether the clinical details
        acquired from a Lymphograph pertains to a normal or abnormal finding.</p><br>
    </div>
    <div class="col-sm-4" id="pred">
      <div class="card">
        
        <div class="card-body">
          <h4>Prediction </h4>
          <br>
          <p class="card-text">If you wish to determine the type of lymphography you have, click on the
            predict button to check.</p>
          <form action="/pred_page">
```

```
            <form action="/pred_page">
              <input type="submit" class="btn btn-danger">
            </form>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</body>

</html>
```

Prediction.html:

```
<div class="col">
  <div class="card">
    
    <div class="card-body">
      <h3>Class C</h3>
      <p class="card-text">Metastasis refers to the spread of cancer cells from their original
        (primary) site to other parts of the body, forming secondary tumors.</p>
    </div>
  </div>
</div>
</div>
<br>
<br>
<div class="container">
  <div class="row">
    <div class="col-sm-8" id="desc">
      <br>
      <h3>Lymphography</h3>
      <br>
      <p>A lymph node is an oval-shaped organ of the immune system, distributed widely throughout the body.
        They tend to expand in size for diverse reasons, indicating health complications that scale from
        trivial, to life-threatening ailments such as cancers.</p><br>
      <p>The condition of lymph nodes is so significant that it is used to accurately sense the stage in Cancer
        progression, which decides the treatment to be adopted. Lymphography is a medical imaging technique
        in which a radio contrast agent is injected, and then an X-ray picture is taken to visualize
        structures of the lymphatic system, including lymph nodes, lymph ducts, lymphatic tissues, lymph
        capillaries and lymph vessels. This data is necessary to decide on whether the clinical details
        acquired from a Lymphograph pertains to a normal or abnormal finding.</p><br>
    </div>
    <div class="col-sm-4" id="pred">
      <div class="card">
        
        <div class="card-body">
          <h4>Prediction </h4>
          <br>
          <p class="card-text">If you wish to determine the type of lymphography you have, click on the
            predict button to check.</p>
          <form action="/pred_page">
            <input type="submit" class="btn btn-danger">
          </form>
        </div>
      </div>
    </div>
  </div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

result.html:

ates > <> result.html > ...

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Lymph Disease Predictor</title>
</head>
<style>
  body {
    padding: 1%;
    margin: 2%;
    box-sizing: border-box;
    background-color: rgba(255, 228, 181, 0.932); /* Light grayish background */
  }

  #main-heading {
    font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
    text-align: center;
    background-color: white; /* Blue background for the heading */
    color: black; /* White text for contrast */
    padding: 2%;
    justify-content: center;
    border-radius: 2rem;
  }
}

.div1 {
  background-color: white;
  width : 70%;
  height : auto;
  align-items: center;
  border-radius: 2rem;
  display: block;
  margin-left: auto;
  margin-right: auto;
}

p {
  text-align: justify;
  line-height: 2rem;
  padding-top: 0%;
  padding-left: 3%;
  padding-right: 3%;
  padding-bottom: 3%;
  font-size: large;
}
```

```
p {
  text-align: justify;
  line-height: 2rem;
  padding-top: 0%;
  padding-left: 3%;
  padding-right: 3%;
  padding-bottom: 3%;
  font-size: large;
}

</style>
<body>
  <div id="main-heading">
    <h2>Prediction Result</h2>
  </div>
  <br><br>
  <div class="div1">
    <h2 style="padding: 2%;">{{prediction}}</h2>
    <p>
      {{desc}}</p>
  </div>
</body>
</html>
```

app.py:

```
app.py > pred_fun
1  # pip install flask
2
3  from flask import Flask,render_template,request
4  import pickle as pickle
5
6  #loading Scaler
7  scalar=pickle.load(open('ms_saved.pkl','rb'))
8
9  model=pickle.load(open('saved_model.pkl','rb'))
10
11 app=Flask(__name__)
12
13 @app.route('/')
14 def main_func():
15     return render_template("index.html")
16
17 @app.route('/pred_page')
18 def pred_page():
19     return render_template("prediction.html")
20
21 @app.route('/predict',methods=['POST'])
22 def pred_fun():
23     if request.method=="POST":
24         a = request.form["a"]
25         b = request.form["b"]
26         c = request.form["c"]
27         d = request.form["d"]
28         e = request.form["e"]
29         f = request.form["f"]
30         g = request.form["g"]
31         h = request.form["h"]
32         i = request.form["i"]
33         j = request.form["j"]
34         k = request.form["k"]
35         l = request.form["l"]
36         m = request.form["m"]
37         n = request.form["n"]
38         o = request.form["o"]
39         p = request.form["p"]
40         q = request.form["q"]
41         r = request.form["r"]
42
43
44     t = [[float(a),float(b),float(c),float(d),float(e),float(f),float(g),float(h),
45             float(i),float(j),float(k),float(l),float(m),float(n),float(o),float(p),float(q),float(r)]]
46     x=scalar.transform(t)
47     output =model.predict(x)
48     index1=['NORMAL FIND','METASTASES','MALIGN LYMPH','FIBROSIS']
49     k=index1[output[0]-1]
50
51     if(k=='NORMAL FIND'):
52         data="No disease detected!"
53     elif(k=='METASTASES'):
54         data="Metastasis is a complex biological process by which cells from a primary tumor spread to other parts of the
55     elif(k=='MALIGN LYMPH'):
56         data="When people talk about malignancy in the context of the lymphatic system, they often refer to cancer that h
57     else:
58         data="Fibrosis is a condition characterized by the formation of excess fibrous connective tissue in an organ or t
59
60     return render_template("result.html",prediction=k,desc=data)
61
62 if __name__ == "__main__" :
63     app.run(debug=True)
```


Milestone 6: Run the Application:

- Open anaconda prompt from the start menu
- Navigate to the folder where your python script is.
- Now type “python app.py” command
- Navigate to the localhost where you can view your web page.

Click on the predict button from the top right corner, enter the inputs, click on the submit button and see the result/prediction on the web.

.....

Home Page:

Lymphography Classifier



Malignant lymphoma

Malignant lymphoma is a cancer originating in the lymphatic system, marked by uncontrolled growth of abnormal lymphocytes.



Fibrosis

Fibrosis is the buildup of excess fibrous tissue in organs, disrupting function due to scarring from chronic inflammation or injury.



Metastasis

Metastasis refers to the spread of cancer cells from their original (primary) site to other parts of the body, forming secondary tumors.

Lymphography

A lymph node is an oval-shaped organ of the immune system, distributed widely throughout the body. They tend to expand in size for diverse reasons, indicating health complications that scale from trivial, to life-threatening ailments such as cancers.

The condition of lymph nodes is so significant that it is used to accurately sense the stage in Cancer progression, which decides the treatment to be adopted. Lymphography is a medical imaging technique in which a radio contrast agent is injected, and then an X-ray picture is taken to visualize structures of the lymphatic system, including lymph nodes, lymph ducts, lymphatic tissues, lymph capillaries and lymph vessels. This data is necessary to decide on whether the clinical details acquired from a Lymphograph pertains to a normal or abnormal finding.



Prediction

If you wish to determine the type of lymphography you have, click on the predict button to check.

[Submit](#)

Once we click on the submit button, it will redirect us to the predict page.

Lymphography Prediction

Enter the Lymphatics :

Enter the block of affere :

Enter the bl. of lymph. c :

Enter the bl. of lymph. s :

Enter the bypass :

Enter the extravasates :

Enter regeneration of :

Enter lym.nodes enlar :

Enter changes in lym :

Enter defect in node :

Enter changes in node :

Enter changes in stru :

Enter special forms :

Enter dislocation of :

Enter the block of affere :

Enter the bl. of lymph. c :

Enter the bl. of lymph. s :

Enter the bypass :

Enter the extravasates :

Enter regeneration of :

Enter the early uptake in :

Enter the lym.nodes dimin :

Enter changes in lym :

Enter defect in node :

Enter changes in node :

Enter changes in stru :

Enter special forms :

Enter dislocation of :

Enter exclusion of ao :

Enter no. of nodes in :

submit

User entered their details:

Lymphography Prediction

Enter the Lymphatics :

4

Enter the block of affere :

2

Enter the bl. of lymph. c :

1

Enter the bl. of lymph. s :

1

Enter the bypass :

1

Enter the extravasates :

1

Enter regeneration of :

1

Enter lym.nodes enlar :

2

Enter changes in lym :

2

Enter defect in node :

2

Enter changes in node :

4

Enter changes in stru :

8

Enter special forms :

1

Enter dislocation of :

1

Enter the block of affere :

Enter the bl. of lymph. c :

Enter the bl. of lymph. s :

Enter the bypass :

Enter the extravasates :

Enter regeneration of :

Enter the early uptake in :

Enter the lym.nodes dimin :

Enter changes in lym :

Enter defect in node :

Enter changes in node :

Enter changes in stru :

Enter special forms :

Enter dislocation of :

Enter exclusion of ao :

Enter no. of nodes in :

Enter the block of affere :	Enter changes in lym :
<input type="text" value="2"/>	<input type="text" value="2"/>
Enter the bl. of lymph. c :	Enter defect in node :
<input type="text" value="1"/>	<input type="text" value="2"/>
Enter the bl. of lymph. s :	Enter changes in node :
<input type="text" value="1"/>	<input type="text" value="4"/>
Enter the bypass :	Enter changes in stru :
<input type="text" value="1"/>	<input type="text" value="8"/>
Enter the extravasates :	Enter special forms :
<input type="text" value="1"/>	<input type="text" value="1"/>
Enter regeneration of :	Enter dislocation of :
<input type="text" value="1"/>	<input type="text" value="1"/>
Enter the early uptake in :	Enter exclusion of no :
<input type="text" value="2"/>	<input type="text" value="2"/>
Enter the lym.nodes dimin :	Enter no. of nodes in :
<input type="text" value="1"/>	<input type="text" value="4"/>

After submitting the values, it will redirect the user to the result page and the Prediction is as Follows;

Prediction Result

MALIGN LYMPH

When people talk about malignancy in the context of the lymphatic system, they often refer to cancer that has spread to the lymph nodes or originated in the lymphatic system. Lymph nodes are small, bean-shaped structures that produce and store cells that help fight infection. If cancer cells break away from a tumor, they can travel through the lymphatic system and form new tumors in other parts of the body. Common cancers that can involve lymph nodes include lymphomas (cancers of the lymphatic system) and metastatic cancers (cancers that have spread from their original site to other parts of the body).
