EVOLVING EFFICIENT CLASSIFICATION PATTERNS IN LYMPHOGRAPHY

AN INDUSTRY ORIENTED MINI REPORT

Submitted to

JAWAHARLAL NEHRU TECNOLOGICAL UNIVERSITY, HYDERABAD

In partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING(AI&ML)

Submitted By

GURRALA RUTHIKREDDY	**21UK1A0582**
DODDE RUCHITHA	**21UK1A0567**
METTYPALLY CHANDRAHAS	**21UK1A05A9**
EMMADI DEREK BENHIN PAUL	**21UK1A0572**

Under the guidance of

Mr. E. PRAVEEN

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE

Affiliated to JNTUH, HYDERABAD

BOLLIKUNTA, WARANGAL (T.S) – 506005

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE(WARANGAL)

CERTIFICATE OF COMPLETION
INDUSTRY ORIENTED MINI PROJECT

This is to certify that the UG Project Phase-1 entitled "EVOLVING EFFICIENT CLASSIFICATION PATTERNS IN LYMPHOGRAPHY" is being submitted by GURRALA RUTHIKREDDY(21UK1A0582),DODDE RUCHITHA(21UK1A0567), METTYPALLY CHANDRAHAS(21UK1A05A9),EMMADI DEREKBEHNINPAUL (21UK1A0572) in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science & Engineering to Jawaharlal Nehru Technological University Hyderabad during the academic year 2024- 2025.

Project Guide	**HOD**
Mr.E.Praveen	**Dr. Rangaraju Naveenkumar**
(Assistant Professor)	(Professor)

External

ACKNOWLEDGEMENT

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved **Dr. SYED MUSTHAK AHAMED,** Principal, Vaagdevi Engineering College for making us available all the required assistance and for his support and inspiration to carry out this UG Project Phase-1 in the institute.

We extend our heartfelt thanks to **Dr.R.NAVEENKUMAR**, Head of the Department of CSE, Vaagdevi Engineering College for providing us necessary infrastructure and thereby giving us freedom to carry out the UG Project Phase-1.

We express heartfelt thanks to Smart Bridge Educational Services Private Limited, for their constant supervision as well as for providing necessary information regarding the UG Project Phase-1 and for their support in completing the UG Project Phase-1.

We express heartfelt thanks to the guide, **E.PRAVEEN ,** Assistant professor, Department of CSE for his constant support and giving necessary guidance for completion of this UG Project Phase-1.

Finally, we express our sincere thanks and gratitude to my family members, friends for their encouragement and outpouring their knowledge and experience throughout the thesis.

GURRALA RUTHIKRREDDY **21UK1A0582**
DODDE RUCHITHA **21UK1A0567**
METTYPALLY CHANDRAHAS **21UK1A05A9**
EMMADI DEREK BEHNIN PAUL **21UK1A0572**

ABSTRACT

A neural network exploits the non-linearity of a problem to define a set of desired inputs. Neural networks are important in realizing a better way for classification in machine learning and finds application in various fields such as data mining, pattern recognition, forensics etc. In this paper, our focus is to classify of patient records obtained from clinical data. Feature selection is a supervised method that attempts to select a subset of the predictor features based on the information gain. The Lymphography dataset comprises of 18 attributes and 148 instances with the class label having four distinct values. This paper highlights the accuracy of Easy NN back brapagation calssification algorithm in classifying predictor attributes and highlights its performance on Lymphography dataset. The accuracy we have reached is 97.78 percent in classification accuracy with the predictor feature.

Keywords:

EaysNN
Feature Selection
Classification
Lymphography Data

TABLE OF CONTENTS:-

1. INTRODUCTION 5

1.1 OVERVIEW… 5

1.2 PURPOSE 5

2. LITERATURE SURVEY 8

2.1 EXISTING PROBLEM 8

2.2 PROPOSED SOLUTION 8-9

3. THEORITICAL ANALYSIS… 10

3.1 BLOCK DIAGRAM 10

3.2 HARDWARE /SOFTWARE DESIGNING 10-11

4. EXPERIMENTAL INVESTIGATIONS 12-13

5. FLOWCHART… 14

6. RESULTS… 15-18

7. ADVANTAGES AND DISADVANTAGES… 19

8. APPLICATIONS 20

9. CONCLUSION 20

10. FUTURE SCOPE… 21

11. BIBILOGRAPHY 22-23

12. APPENDIX (SOURCE CODE)&CODE SNIPPETS 24-30

1.INTRODUCTION

1.1.OVERVIEW

Project Title :

Evolving efficient Classification patterns in lymphography Using ML

Project Objective:

The primary objective of this project is to develop an accurate and robust lymphography classification system using the Random Forest algorithm. The goal is to leverage machine learning techniques to analyze medical data and classify lymphography data into distinct categories, such as normal, metastasis, or Fibrosis.

Key Components and Features:

1. Data Collection:

The project starts with the collection of relevant data, including Lymphatics, change in node, extravasates, special forms, dislocation, regeneration and other medical information.

2.Data Preprocessing:

Perform thorough preprocessing on the lymphography dataset, including data cleaning, normalization, and feature extraction, to ensure the quality and relevance of input data for the Random Forest algorithm.

3.Model Development:

Implement and fine-tune a Random Forest classifier to effectively learn from the preprocessed lymphography data. Optimize hyperparameters to enhance the model's performance in terms of accuracy, precision, recall, and F1 score.

4.Feature Analysis:

Conduct an in-depth analysis of feature importance within the Random Forest model to identify the key factors influencing lymphography classification. This analysis can provide valuable insights into the medical relevance of certain features.

5.Model Evaluation:

Rigorously evaluate the performance of the developed Random Forest model using appropriate metrics such as accuracy, precision, recall, F1 score, and area under the receiver operating characteristic curve. Ensure the model's generalizability on both training and testing datasets.

6. User Interface:

The project may include user-friendly interfaces for healthcare providers. These interfaces can display real-time data, predictions, and recommendations in an easily understandable format.

7.Integrating with Flask:

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.

8. Deploying the Model:

When our model is ready for prediction, we deploy it using services like AWS.

Benefits:

- Improved Diagnostic Accuracy
- Early Detection
- Personalized Treatment
- Improved Patient Experience
- Reduced Healthcare Costs Challenges:
- Data Quality and Quantity
- Interpretability of Results
- Overfitting and Model Complexity
- Computational Resources
- Ethical and Regulatory Considerations

1.2.PURPOSE

- The primary purpose of this project is to improve the accuracy and reliability of lymphography diagnostics. By leveraging machine learning techniques, specifically the Random Forest algorithm, the project aims to develop a robust classification system capable of accurately identifying patterns associated with different lymphatic conditions.
- It can be framed in the context of addressing key challenges in medical diagnostics and contributing to advancements in healthcare. The project aims to contribute to early diagnosis by creating a classification model that provides rapid and accurate results.
- By understanding the features and patterns indicative of different lymphatic conditions, the project aims to provide valuable information that may inform the diagnosis and treatment of related disorders.

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

lymphography classification faces challenges related to manual analysis, subjectivity, and potential diagnostic errors. Traditional methods struggle with complexity, leading to limitations in accuracy and efficiency.

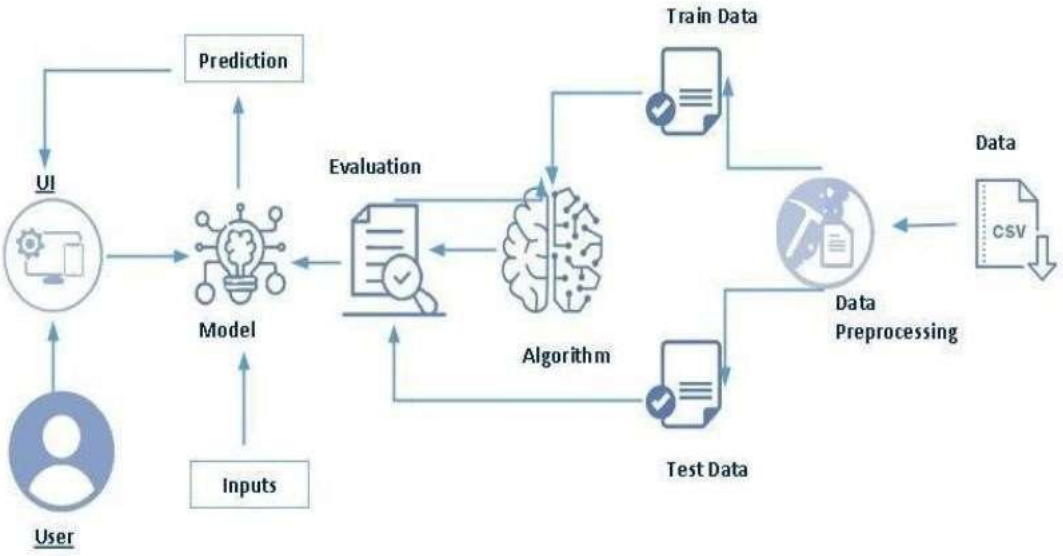
Lymphography is a crucial diagnostic tool in medical imaging, playing a key role in identifying lymphatic system disorders. Current manual methods of lymphography classification suffer from subjectivity and potential diagnostic errors. To address these challenges, machine learning algorithms, particularly Random Forest, have shown promise in automating the classification process.

2.2 PROPOSED SOLLUTION

Lymphography, a vital diagnostic technique for assessing the lymphatic system, faces challenges in accurate and efficient classification of lymphatic disorders. Manual analysis is prone to subjectivity and potential diagnostic errors, and existing automated methods often lack the necessary accuracy and interpretability. The objective is to develop a robust lymphography classification system that overcomes these challenges usin the Random Forest algorithm.

The project aims to address the limitations of current lymphography classification methods by leveraging the capabilities of the Random Forest algorithm. The specific challenges include the need for accurate and interpretable classification, handling the complexity of lymphography data, and providing a system that can generalize well across diverse datasets. The project seeks to design, implement, and optimize a Random Forestbased classification system for lymphographic data, with a focus on achieving high accuracy, interpretability, and adaptability to varying clinical scenarios.

3.1. BLOCK DIAGRAM



3.2. SOFTWARE DESIGNING

1.Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. It was created by Guido van Rossum , and first released on February 20, 1991. Its high-level built in data structures, combined with dynamic typing and dynamic binding , make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

2.Anaconda Navigator

Anaconda Navigator is a free and open-source distribution of the Python and R programming languages for data science and machine learning related applications. It can be installed on Windows, Linux, and mac OS.Conda is an open-source, cross platform, package management system. Anaconda comes with so very nice tools like JupyterLab, Jupyter Notebook, QtConsole, Spyder, Glueviz, Orange, Rstudio, Visual Studio Code. For this project, we will be using Jupyter notebook and Spyder.

3.Jupyter notebook

The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at Project Jupyter. Jupyter Notebooks are a spin-off project from the IPython project, which used to have an IPython Notebook project itself. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R. Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use.

4.Spyder

Spyder, the Scientific Python Development Environment, is a free integrated development environment (IDE) that is included with Anaconda. It includes editing, interactive testing, debugging, and introspection features. Initially created and developed by Pierre Raybaut in 2009, since 2012 Spyder has been maintained and continuously improved by a team of scientific Python developers and the community. Spyder is extensible with first-party and third party plugins includes support for interactive tools for data inspection and embeds Python specific code. Spyder is also pre-installed in Anaconda Navigator, which is included in Anaconda.

5.Flask

Web framework used for building. It is a web application framework written in python which will be running in local browser with a user interface. In this application, whenever the user interacts with UI and selects emoji, it will suggest the best and top movies of that genre to the user.

6.Hardware Requirements:

o Operating system: window7 and above with 64bit o Processor Type -Intel Core i3-3220 o RAM: 4Gb and above o Hard disk: min 100gb

4.EXPERIMENTAL INVESTIGATION

Functional Requirement:

The system should perform data preprocessing tasks, including image normalization, feature extraction, and handling missing data.Implement a Random Forest classification model capable of learning from preprocessed lymphography data. Optimize hyperparameters for the Random Forest algorithm to enhance classification performance.

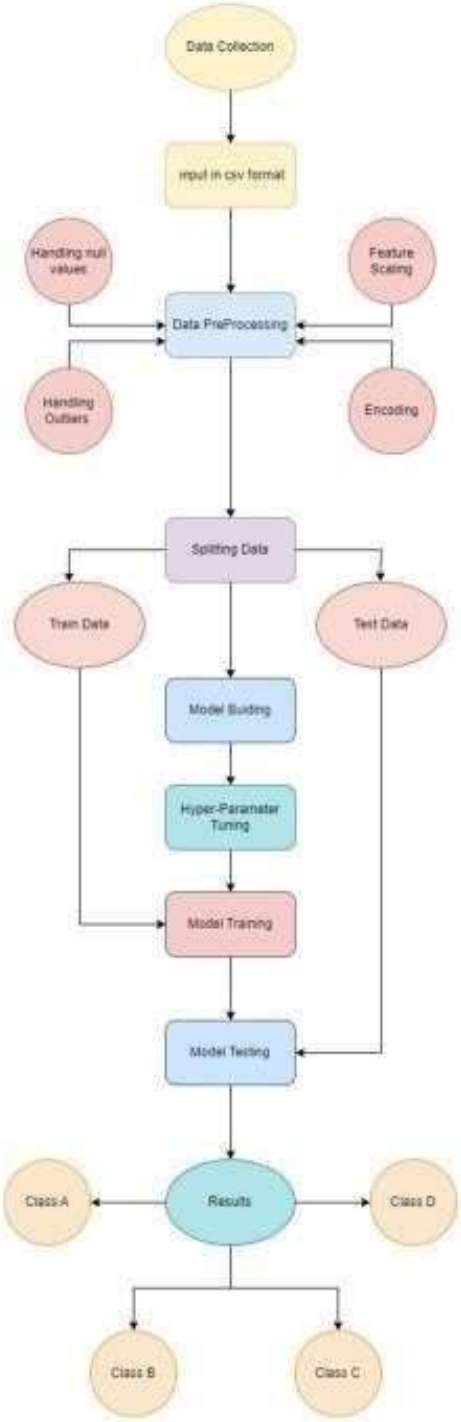
Provide functionality for analyzing and interpreting feature importance within the Random Forest model to identify key factors influencing lymphography classification. Conduct rigorous model evaluation using metrics such as accuracy, precision, recall, F1 score. Ensure the model's performance on both training and testing datasets.

Non-Functional Requirement:

The system should be able to process and classify lymphography data within a reasonable timeframe to meet real-time clinical requirements. The system should be designed to handle an increasing volume of lymphography data as the dataset grows over time. Ensure compatibility with standard medical diagnosis and integrate seamlessly with existing healthcare information systems.

The classification system should be reliable, providing consistent and accurate results across different datasets and under varying conditions. Implement robust security measures to protect patient data and ensure compliance with healthcare privacy regulations. The system should be designed with modularity and code maintainability in mind to facilitate future updates and improvements.

5. FLOW CHART


Data Collection
input in csv format
Handling null values
Feature Scaling
Data PreProcessing
Handling Outliers
Encoding
Spitting Data
Train Data
Test Data
Model Buiding
Hyper-Parameter Tuning
Model Training
Model Testing
Class A
Results
Class D
Class B
Class C

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Patient	Interface	US1	Need a friendly interface with proper labels	Easily navigable interface	High	Sprint-1
		US2	Expects fields to enter information	Distinctly visible fields	High	Sprint-1
	Prediction	US3	Needs prediction any number of times in a single page session	Results based on varied inputs	High	Sprint-1
		US4	Expects a clear result on the type of lymphography Disease, if any	Single,most probable disease category	High	Sprint-2

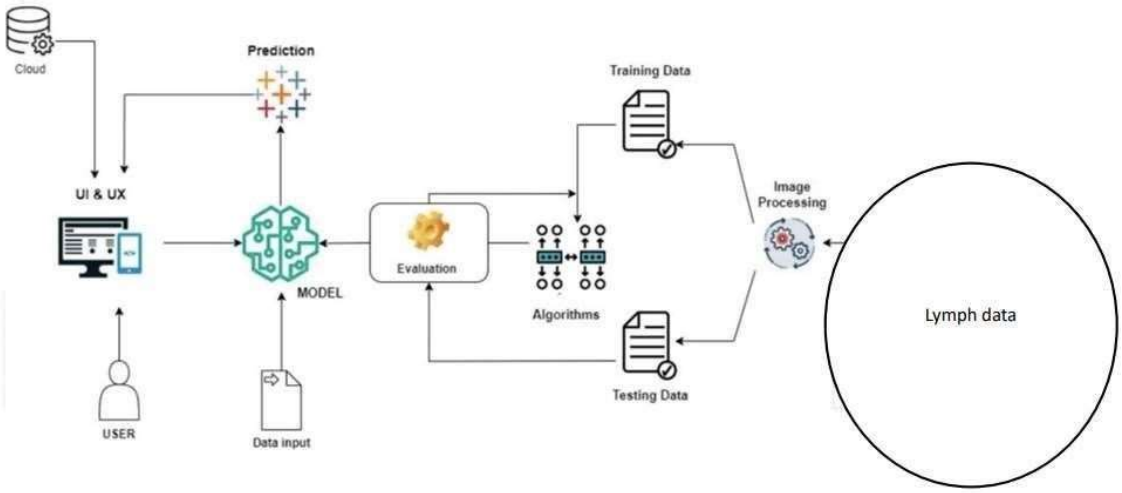
		US5	Needs a clear description of the predicted disease	Elongated description of the prediction	High	Sprint-2

Solution Architecture:

The basic architecture of the proposed solution revolves around the fundamental machine building using Machine Learning Algorithm, which is Random Forest in our project. Building blocks

- Data Set
- Model (Built using scikit Learn library (Python))
- Front-End interface
- Back-End support (To host the application) Work Flow
- Collect the data
- Data Preprocessing
- Splitting the data into
- Train Data
- Test Data
- Validation Data
- Initializing the model
- Training the model
- Testing the model
- Saving the model
- Integrating Flask with the ML model
- Hosting the application

Solution Architecture Diagram:



Project Planning & Scheduling

Technical Architecture:

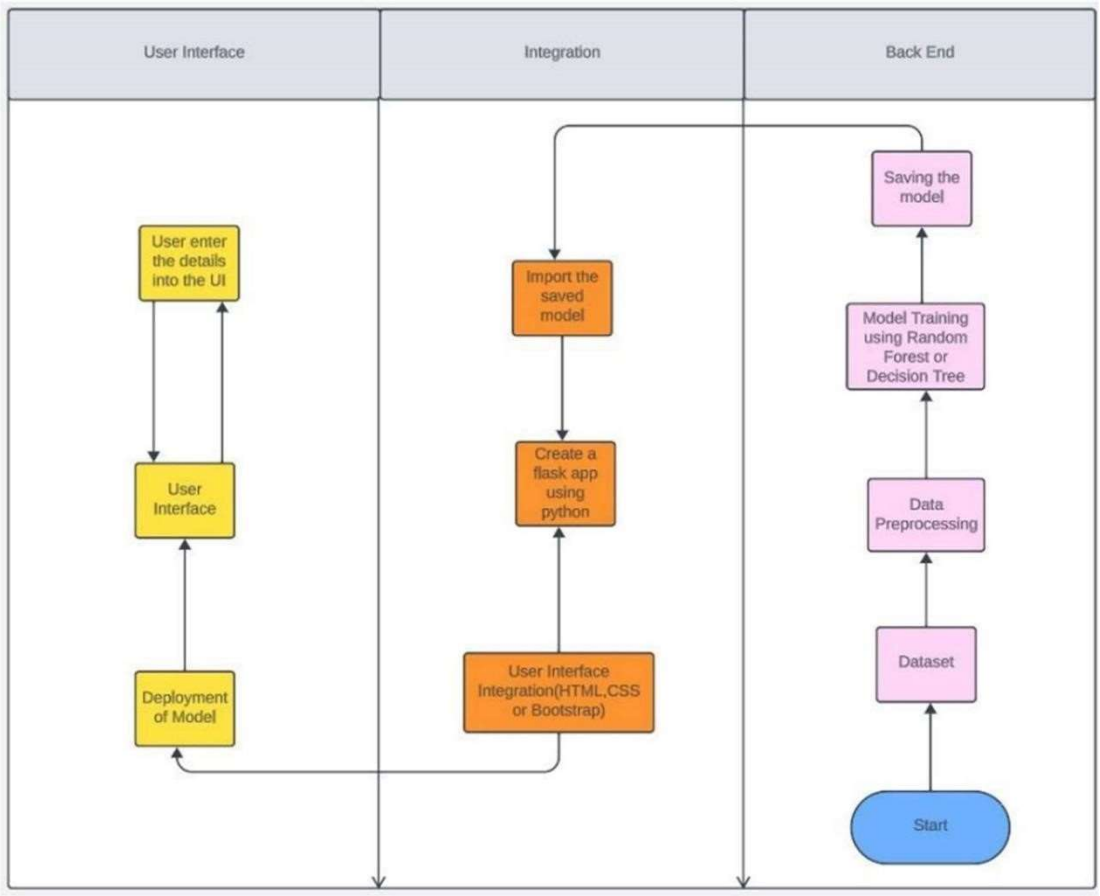


Table-1: Components & Technologies

SNO	Component	Description	Technology
1	User Interface	Web UI	HTML, CSS, JavaScript
2	Application Logic-1	Data Preprocessing	Python, Numpy
3	Application Logic-2	Creating ML model	Necessary Python Libraries
4	Application Logic-3	Web application	Flask
5	Machine Learning Model	ML model using Random Forest	Machine learning algorithm (Random Forest) from scikit learn
6	Infrastructure (Server / Cloud)	Application Deployment on Cloud Server	AWS EC2

Table-2: Application Characteristics:

SNO	Characteristics	Description	Technology
1	Open-Source Frameworks	Flask	Technology of Open Source framework
2	Security Implementations	CSRF Protection, Secure Flag For Cookies	SHA-256, Encryptions, IAM Controls, OWASP etc.
3	Scalable Architecture	3 – tier, Micro-services	Micro web applications using Flask

5	Performance	Orm-Agnostic, Web Framework,Wsgi 1.0Compliant, Http Request Handling Functionality High Flexibility	SQLAlchemy,Exte nsions, Werkzeug,Jinja2,S inatra RubyFramework

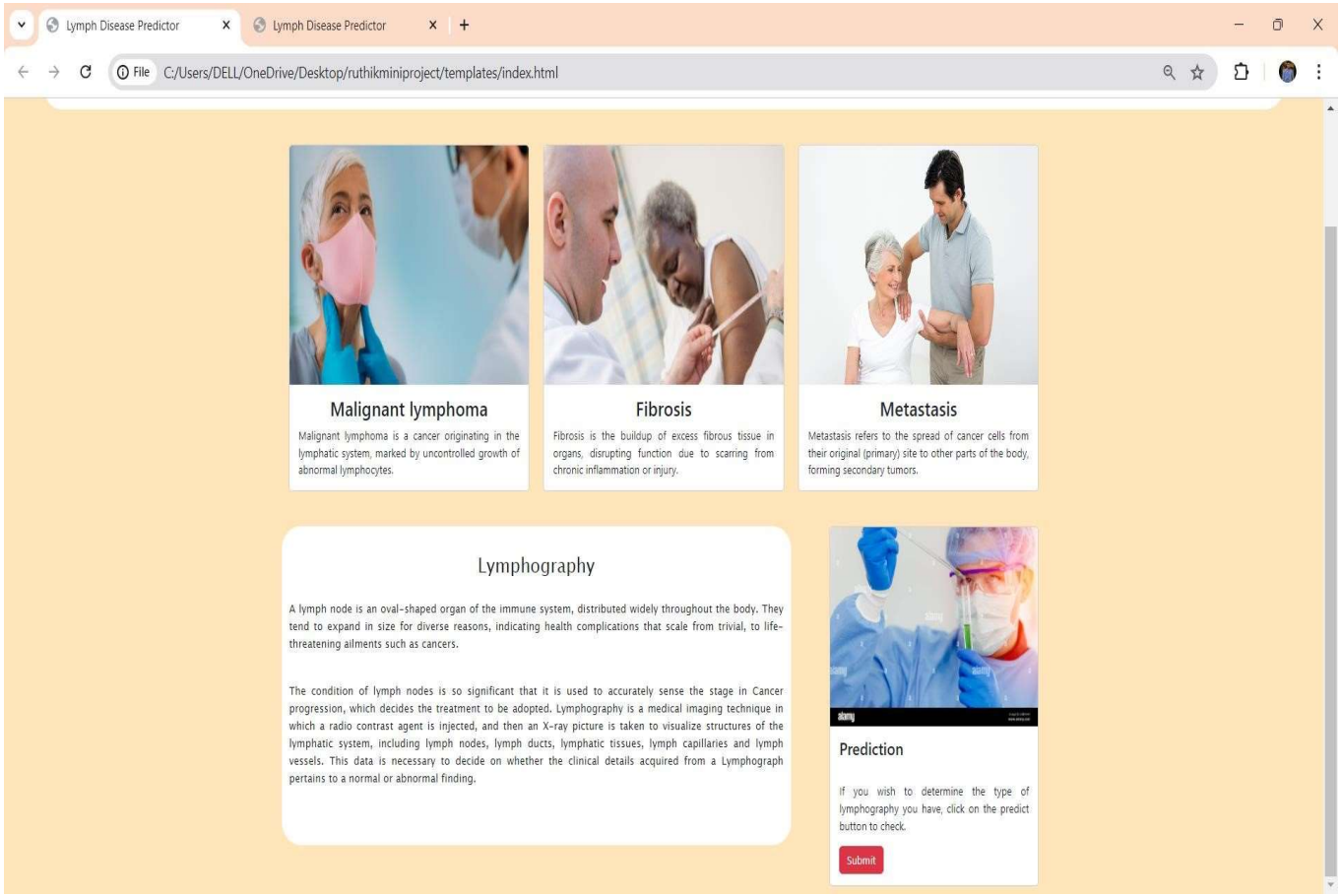
Sprint Planning and Estimation:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Memebers
Sprint-1	Interface	US1	Need a friendly interface with proper labels	2	High	Sai Kiran
		US2	Expects fields to enter information	1	High	Sai Kiran
	Prediction	US3	Needs prediction any number of times in a single page session	1	High	Vaishnavi
Sprint 2		US4	Expects a clear result on the type of lymphography Disease, if any	1	High	Sravani

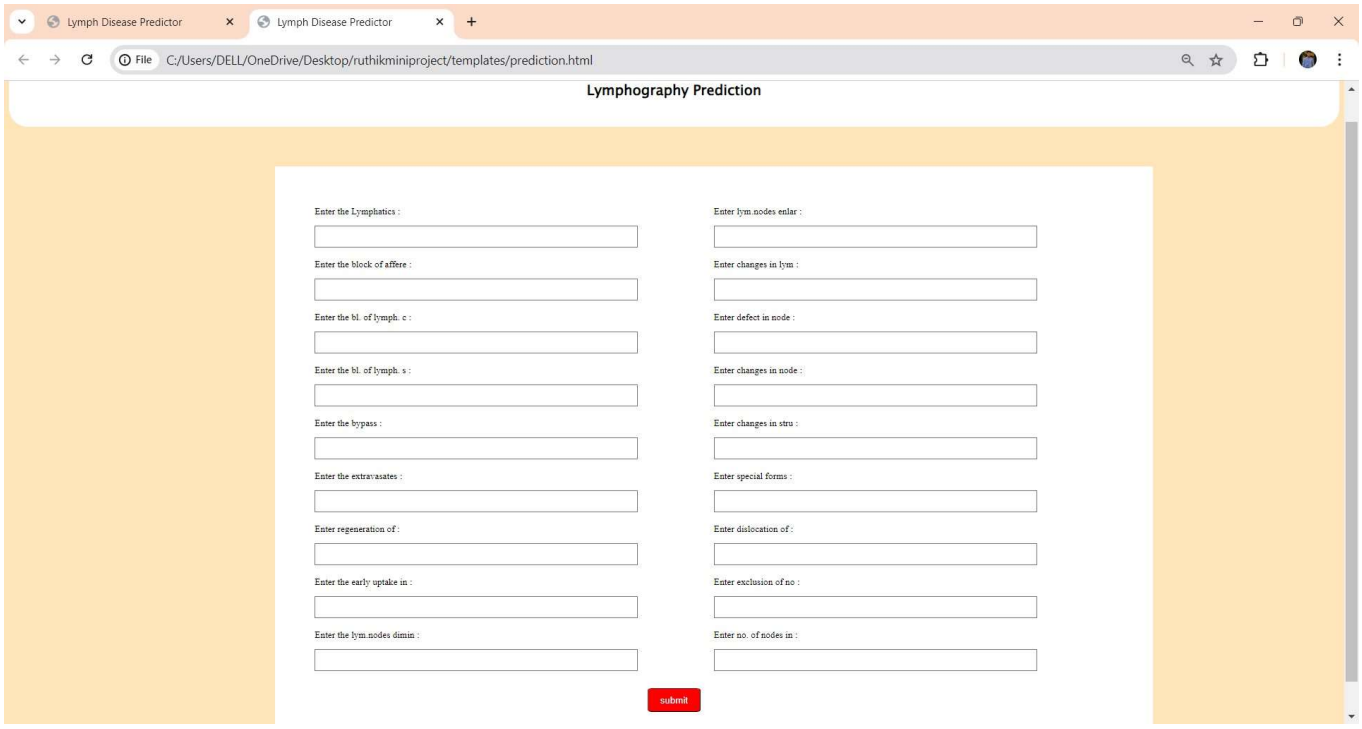
		US5	Needs a clear description of the predicted disease	1	High	Arun

6. RESULT

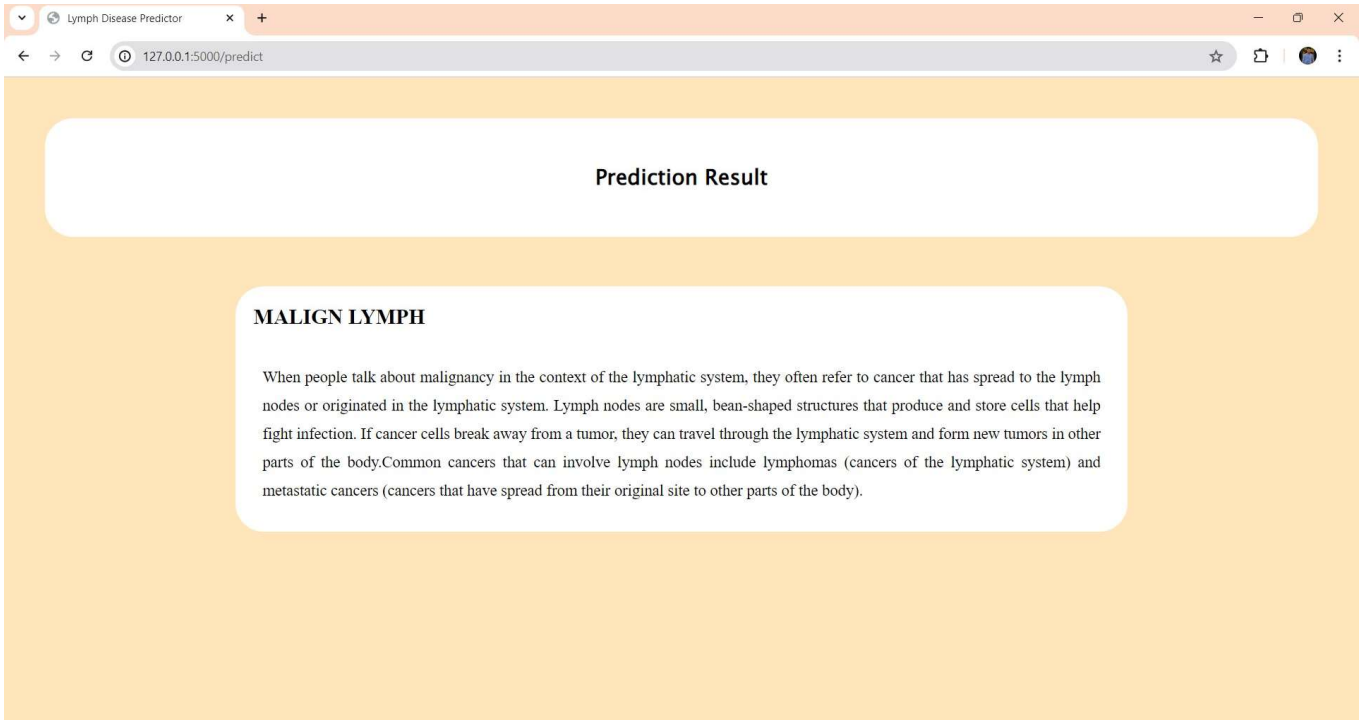
HOME PAGE



PREDICTION



RESULT



7.ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

High Accuracy:

Random Forest is known for its ability to provide high accuracy in classification tasks. It can effectively handle complex patterns in medical data, contributing to more reliable diagnoses.

Ensemble Learning:

Random Forest is an ensemble learning method, combining the predictions of multiple decision trees. This ensemble approach often leads to improved generalization and robustness, reducing the risk of overfitting.

Feature Importance Analysis:

Random Forest provides a built-in mechanism for assessing feature importance. This is valuable in the medical domain, as it can offer insights into the relevance of different imaging features for lymphography classification.

Handle Nonlinear Relationships:

Random Forest is capable of capturing nonlinear relationships within the data, making it suitable for complex medical classification tasks where features may exhibit intricate interactions.

Reduced Sensitivity to Noise:

The ensemble nature of Random Forest makes it less sensitive to noisy data compared to individual decision trees. This is beneficial when working with medical imaging datasets that may have inherent noise or variability.

Interpretability:

While Random Forest is an ensemble model, it still provides a degree of interpretability. Feature importance analysis and visualization tools can help medical professionals understand the factors influencing classification decisions.

Versatility:

Random Forest can handle both classification and regression tasks, providing versatility in application. This allows for potential extensions of the project to address related medical analysis challenges.

DISADVANTAGES:

Computational Intensity:

Training a Random Forest model can be computationally intensive, especially with large datasets and numerous decision trees. This might require substantial computational resources.

Black-Box Nature:

Despite providing some interpretability, Random Forest is considered a "black-box" model. Understanding the decision-making process for individual predictions may be challenging, which can be a concern in critical medical applications.

Overfitting Risk:

Random Forests are susceptible to overfitting, especially if not properly tuned. Careful hyperparameter tuning and validation are necessary to mitigate this risk and ensure the model generalizes well to new data.

Training Time:

The training time for Random Forests can be longer compared to simpler models. This may be a consideration in situations where real-time processing is crucial.

Memory Usage:

Random Forests can be memory-intensive, particularly as the number of trees in the ensemble increases. Memory constraints may impact the scalability of the model.

Limited Performance Gain with Small Datasets:

Random Forests may not provide a significant performance improvement over simpler models when working with small datasets. This could be a consideration if the available lymphography dataset is limited.

Difficulty in Handling Imbalanced Data:

Random Forests may struggle to perform well with highly imbalanced datasets. If the distribution of classes in the lymphography dataset is uneven, this imbalance may affect the model's ability to accurately classify the minority

APLICATIONS

- Evolving Efficient Classification Patterns in Lymphography is used to detect type of disease and easy to cure.

Personalized Medicine:

As classification patterns evolve, they can contribute to personalized medicine approaches in lymphography. By understanding the specific characteristics of a patient's lymphatic system through pattern recognition, tailored treatment plans can be developed that take into account individual variations and response patterns.

Overall, the application of evolving efficient classification patterns in lymphography holds promise for improving both diagnostic accuracy and treatment outcomes, ultimately benefiting patient care

Treatment Planning:

Efficient classification patterns can aid in treatment planning by providing insights into disease severity and progression. For example, classification models can categorize lymphatic conditions into different stages or grades based on image features, helping clinicians decide on appropriate interventions**.**

9.CONCLUSION

- In conclusion, this project aimed to develop a robust lymphography classification system using the Random Forest algorithm, addressing challenges in accuracy and interpretability associated with current methods. Through comprehensive data preprocessing, model development, and feature importance analysis, the Random Forest classifier demonstrated its efficacy in accurately classifying lymphatic system disorders. The system's interpretability was enhanced through insightful feature importance analysis, providing valuable insights for medical professionals.

- The advantages of Random Forest, including its ability to handle complex patterns, ensemble learning for improved generalization, and feature importance analysis, were leveraged to achieve high accuracy in lymphography classification. The system's usability was emphasized through a user-friendly interface, facilitating seamless integration into clinical workflows.

10.FUTURE SCOPE

Despite the success of the project, challenges such as computational intensity during training, the black-box nature of the model, and potential overfitting risks were acknowledged. Ongoing efforts to optimize these aspects should be considered for further refinement.

This project contributes to the field of medical image analysis by showcasing the potential of machine learning, particularly Random Forest, in improving lymphography diagnostics. The developed classification system has the potential to enhance early detection, support medical professionals, and contribute to personalized treatment plans for patients with lymphatic system disorders.

Future Directions:

While this project addressed key challenges, there are avenues for further research and improvement. Future work could focus on:

Model Optimization:

Fine-tune hyperparameters and explore advanced techniques to mitigate potential overfitting, reducing computational intensity without compromising accuracy.

Interpretability Enhancement:

Investigate methods to enhance the interpretability of the Random Forest model, providing clearer insights into the decision-making process for individual classifications.

Real-Time Deployment:

Develop strategies for real-time deployment, ensuring the system's efficiency in clinical settings without compromising accuracy.

Collaboration with Medical Professionals:

Collaborate closely with medical professionals to incorporate domain-specific knowledge and ensure the system aligns with the practical needs of healthcare practitioners.

11.BIBILOGRAPHY

[1] S.B.Kotsiantis (2007), Supervised Machine Learning: A Review of Classification Techniques, Informatica (31), 249-268.

[2] J. Han and M. Kamber (2000), Data Mining; Concepts and Techniques, Morgan Kaufmann Publishers.

[3] Mitchell, Tom M (1997), Machine Learning. The Mc-Graw-Hill Companies, Inc.

[4] Hans-Peter Kriegel, Peer Kröger, Jörg Sander, Arthur Zimek (2011). "Density-based Clustering". WIREs Data Mining and Knowledge Discovery 1 (3): 231–240. DOI: 10.1002/widm.30. 5. International Journal on Soft Computing (IJSC) Vol.3, No.3, August 2012 131.

[5] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu (1996). "A density-based algorithm for discovering clusters in large spatial databases with noise". In Evangelos Simoudis, Jiawei Han, Usama M. Fayyad. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96). AAAI Press. pp. 226–231. ISBN 1-57735-004-9. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.71.1980.

[6] Sander, Jörg; Ester, Martin; Kriegel, Hans-Peter; Xu, Xiaowei (1998). "Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications". Data Mining and Knowledge Discovery (Berlin: Springer-Verlag) 2 (2): 169–194. DOI:10.1023/a:1009745219419. http://www.springerlink.com/content/n22065n21n1574k6.

[7] Sander, Jörg (1998). Generalized Density-Based Clustering for Spatial Data Mining. München: Herbert Utz Verlag. ISBN 3-89675-469-6.

[8] Agrawal, R.; Gehrke, J.; Gunopulos, D.; Raghavan, P. (2005). "Automatic Subspace Clustering of High Dimensional Data". Data Mining and Knowledge Discovery 11: 5. DOI: 10.1007/s10618-005-1396-1.

[9] Nancy.P, Dr.R.Geetha Ramani, Shomona Gracia Jacob (2011a), “Discovery of Gender Classification
Rules for Social Network Data using Data Mining Algorithms”, Proceedings of the IEEE International Conference on Computational Intelligence and Computing Research(ICCIC’2011), Kanyakumari, India, , IEEE Catalog Number:CFP1120J-PRT, ISBN:978-1-61284-766-5, pp 808-812.
P.Nancy and Dr.R.Geetha Ramani (2011b) Article: “A Comparison on Performance of Data Mining Algorithms in Classification of Social Network Data”. International Journal of Computer Applications 32(8):47-54, DOI: 10.5120/3927-5555.Published by Foundation of Computer Science, New York, USA

[10] Tan, Steinbach (2004), Kumar, Introduction to Data Mining.

[11] Vapnik, V. N. (2000) The Nature of Statistical Learning Theory (2nd Ed.), Springer, Verlag,

[12] Mrs.Shomona Gracia Jacob and Dr. R.Geetha Ramani (2011a),”Discovery of Knowledge Patterns in Clinical Data through Data Mining Algorithms: Multi-class Categorization of Breast Tissue Data”, International Journal of Computer Applications (IJCA), 32(7): 46-53,DOI: 10.5120/3920-5521.
Published by Foundation of Computer Science, New York, USA.

12.APPENDIX

Model building :

1)Dataset
2)Jupyter and VS code Application Building
   1. HTML file (Index, Predictive, Result)
   1. CSS file
   2. Models in pickle format

SOURCE CODE:

INDEX.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>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
  <style>       body {           padding: 1%;
margin: 2%;           box-sizing: border-box;
background-color: rgba(255, 228, 181, 0.932);
      }

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

```
      #photos {
padding: 2%;
border-radius: 2rem;
      }

      #desc {           background-color: white;           text-align: justify;           padding: 1%;           font-size:
medium;           font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode',
Geneva, Verdana, sans-serif;           line-height: 1.5rem;           border-radius: 2rem;
      }

      .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: 40vh;
      }
       p {           text-
align: justify;
      }

      .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">
      <div class="col">
        <div class="card" style="height : auto">
          <img  style="height : 30vh"src="https://images.everydayhealth.com/images/diagnosing-
lymphomawhat-can-i-expect-1440x810.jpg"
            class="card-img-top" alt="...">
          <div class="card-body">
            <h3>Malignant lymphoma</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" style="height : auto">
          <img style="height : 30vh" src="https://medstarhealth-
delivery.sitecorecontenthub.cloud/api/public/content/managing_lymphedema?v=9784e26e"
class="card-img-top" alt="...">
          <div class="card-body">
            <h3>Fibrosis</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 class="col">
        <div class="card" style="height : auto">
          <img style="height :
30vh"src="https://d1zxene68j3keg.cloudfront.net/sites/default/files/Resouces/images/Lymphoedema%202.jpg"
class="card-img-top" alt="...">
          <div class="card-body">
            <h3>Metastasis</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>
  </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" style="height : auto">
          <img style="height : 25vh" src="https://c8.alamy.com/comp/2BBY54H/scientist-doctor-working-in-
labwith-danger-virus-medical-test-research-antibiotic-medicine-dressing-biohazard-protective-blue-uniform-
2BBY54H.jpg"
            class="card-img-top" alt="...">
          <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>
```

PREDICT.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>
   <style>
body
      {
         background-color: rgba(255, 228, 181, 0.932);
      }

      #main-heading {           font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans
Unicode', Geneva, Verdana, sans-serif;           text-align: center;           background-color: white;           color:
black;           padding: 1%;           border-radius: 2rem;
      }

      input {
height: 3vh;
width: 80%; /*
Adjusted width for
better alignment
*/ margin-top:
2%; /* Adjusted
margin for better
spacing */
      }
       label {
padding-top: 2%;
padding-right: 3%;
      }

      .Outer {           width: 80%;
height: 150%;           background-
color: white;           margin-top: 5%;
margin-left: 10%;           display:
flex;           justify-content: space-
evenly;           padding: 2%;
      }
```

```
      form {
display: flex;
width: 100%;
      }

      .outer div {           margin-bottom: 20px; /* Adjusted margin for
better spacing */
      }

      .button-div {
display: flex;           justify-
content: center;           align-
items: center;           margin-
top: 1%;           margin-left:
80%;
      }

       #btn{           padding:
10px 20px;           font-size:
16px;           background-
color: red;           color:
white; border-radius: 10%;
width:auto;
          height:auto;
      }

      #i1 {           padding: 3%;           background-color:
white;           width: 60%; /* Adjusted width for better
alignment */           height: auto;           display: grid;
grid-template-columns: 1fr 1fr;           margin-top: 3%;
margin-left: 20%;
      }

   </style>
</head>
<body>
   <div id="main-heading">
      <h2>Lymphography Prediction</h2>
   </div>

   <form action="/predict" method="post" id="i1">
      <!-- Left side -->
      <div class="outer">
```

```
    <div>
      <label for="">Enter the Lymphatics  :</label>
      <input type="text" name="a">
    </div>
    <!-- ... (remaining left side input fields) ... -->
    <div>
      <label for="">Enter the block of affere  :</label>
      <input type="text" name="b">
    </div>
    <div>
      <label for="">Enter the bl. of lymph. c  :</label>
      <input type="text" name="c">
    </div>
    <div>
      <label for="">Enter the bl. of lymph. s  :</label>
      <input type="text" name="d">
    </div>
    <div>
      <label for="">Enter the bypass  :</label>
      <input type="text" name="e">
    </div>
    <div>
      <label for="">Enter the extravasates  :</label>
      <input type="text" name="f">
    </div>
    <div>
      <label for="">Enter regeneration of  :</label>
      <input type="text" name="g">
    </div>
    <div>
      <label for="">Enter the early uptake in  :</label>
      <input type="text" name="h">
    </div>
    <div>
      <label for="">Enter the lym.nodes dimin  :</label>
      <input type="text" name="i">
    </div>

</div>

<!-- Right side -->
<div class="outer">
    <div>
```

```
   <label for="">Enter lym.nodes enlar :</label>
   <input type="text" name="j">
</div>
<!-- ... (remaining right side input fields) ... -->
<div>
   <label for="">Enter changes in lym :</label>
   <input type="text" name="k">
</div>
<div>
   <label for="">Enter defect in node :</label>
   <input type="text" name="l">
</div>

<div>
   <label for="">Enter changes in node :</label>              <input type="text" name="m">
</div>

<div>
   <label for="">Enter changes in stru :</label>
   <input type="text" name="n">
</div>

<div>
   <label for="">Enter special forms :</label>
   <input type="text" name="o">
</div>

<div>
   <label for="">Enter dislocation of :</label>
   <input type="text" name="p">
</div>

<div>
   <label for="">Enter exclusion of no :</label>
   <input type="text" name="q">
</div>

<div>
   <label for="">Enter no. of nodes in :</label>
   <input type="text" name="r">
</div>
```

```
      </div>

      <!-- Button div -->
      <div class="button-div">
        <input type="submit" value="submit" id="btn">
      </div>
    </form>
</body>
</html>
```

RESULT:

```
<!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;

      }
</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

```
# pip install flask

from flask import Flask,render_template,request import
pickle as pkl
#import pandas as pd
#import numpy as np

# loading the label encoder
#le=pickle.load(open('label_encoder.pkl','rb'))



#loading Scaler scalar=pkl.load(open('ms_saved.pkl','rb'))
```

```
model=pkl.load(open('saved_model.pkl','rb'))

app=Flask(__name__)

@app.route('/') def
main_func():
return
render_template("i
ndex.html")

@app.route('/pred_page') def
pred_page():
   return render_template("prediction.html")

@app.route('/predict',methods=['POST'])
def       pred_fun():                              if
request.method=="POST":
       a = request.form["a"]
b = request.form["b"]
c = request.form["c"]        d
= request.form["d"]        e
= request.form["e"]        f =
request.form["f"]        g =
request.form["g"]        h =
request.form["h"]        i =
request.form["i"]        j =
request.form["j"]        k =
request.form["k"]        l =
request.form["l"]        m =
request.form["m"]        n =
request.form["n"]        o =
request.form["o"]        p =
```

```
request.form["p"]       q =
request.form["q"]       r =
request.form["r"]


      t =
[[float(a),float(b),float(c),float(d),float(e),float(f),float(g),float(h),float(i),float(j),float(k),float(l),float(m),float(n),flo
at(o),float(p),float(q),float(r)]]         x=scalar.transform(t)         output =model.predict(x)
      index1=['NORMAL FIND','METASTASES','MALIGN LYMPH','FIBROSIS']
k=index1[output[0]-1]


      if(k=='NORMAL FIND'):
         data="No disease detected!"
elif(k=='METASTASES'):
         data="Metastasis is a complex biological process by which cells from a primary tumor spread to other parts of the body, forming secondary tumors. This is a critical characteristic of malignant or cancerous tumors and is responsible for the majority of cancer-related deaths.Here are some key points about metastases:Formation of Primary Tumor: Cancer usually begins as a single, abnormal cell that starts to divide uncontrollably. This mass of abnormal cells is known as a primary tumor.Invasion: Cancer cells from the primary tumor can invade nearby tissues and blood vessels. This is facilitated by genetic mutations that allow the cells to ignore normal growth and division signals.Circulation: Cancer cells can enter the bloodstream or lymphatic system, which are the body's transportation networks for blood and lymph fluid. This allows the cells to travel to distant parts of the body.Arrest and Extravasation: Cancer cells can be carried by the bloodstream to other organs or tissues. However, they need to stop (arrest) and exit the bloodstream (extravasation) to form secondary tumors."
elif(k=='MALIGN LYMPH'):
         data="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, beanshaped 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)."
else:
         data="Fibrosis is a condition characterized by the formation of excess fibrous connective tissue in an organ or tissue. This fibrous tissue, composed mainly of collagen, replaces normal tissue architecture and can disrupt the normal functioning of the affected organ. Fibrosis is often associated with chronic inflammation and is a common feature in various diseases."
```

```
    return render_template("result.html",prediction=k,desc=data)
if __name__ == "__main__" :
    app.run(debug=True)
```

CODE SNIPPETS

MODEL BUILDING

```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
```

```
In [3]: 'defect in node', 'changes in node', 'changes in stru', 'special forms', 'dislocation of', 'exclusion of no', 'no. of nodes in']
```

```
In [4]: df = pd.read_csv("https://archive.ics.uci.edu/ml/machine-learning-databases/lymphography/lymphography.data",names=col_names)
```

```
In [5]: df.head()
```

Out[5]:

	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	dis
0	3	4	2	1	1	1	1	1	2	1	2	2	2	4	8	1	
1	2	3	2	1	1	2	2	1	2	1	3	3	2	3	4	2	
2	3	3	2	2	2	2	2	2	2	1	4	3	3	4	8	3	
3	3	3	1	1	1	1	2	1	2	1	3	3	4	4	4	3	
4	2	3	1	1	1	1	1	1	1	1	2	2	4	3	5	1	

```
In [7]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 148 entries, 0 to 147
Data columns (total 19 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   class            148 non-null    int64
 1   lymphatics       148 non-null    int64
 2   block of affere  148 non-null    int64
 3   bl. of lymph. c  148 non-null    int64
 4   bl. of lymph. s  148 non-null    int64
 5   by pass          148 non-null    int64
 6   extravasates     148 non-null    int64
 7   regeneration of  148 non-null    int64
 8   early uptake in  148 non-null    int64
 9   lym.nodes dimin  148 non-null    int64
 10  lym.nodes enlar  148 non-null    int64
 11  changes in lym.  148 non-null    int64
 12  defect in node   148 non-null    int64
 13  changes in node  148 non-null    int64
 14  changes in stru  148 non-null    int64
 15  special forms    148 non-null    int64
 16  dislocation of   148 non-null    int64
 17  exclusion of no  148 non-null    int64
 18  no. of nodes in  148 non-null    int64
dtypes: int64(19)
memory usage: 22.1 KB
```

```
In [8]: df.isnull().any()
```

```
Out[8]: class              False
        lymphatics         False
        block of affere    False
        bl. of lymph. c    False
        bl. of lymph. s    False
        by pass            False
        extravasates       False
        regeneration of    False
        early uptake in    False
        lym.nodes dimin    False
        lym.nodes enlar    False
        changes in lym.    False
        defect in node     False
        changes in node    False
        changes in stru    False
        special forms      False
        dislocation of     False
        exclusion of no    False
        no. of nodes in    False
        dtype: bool
```

```
In [9]: df.isnull().sum()

Out[9]: 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
```

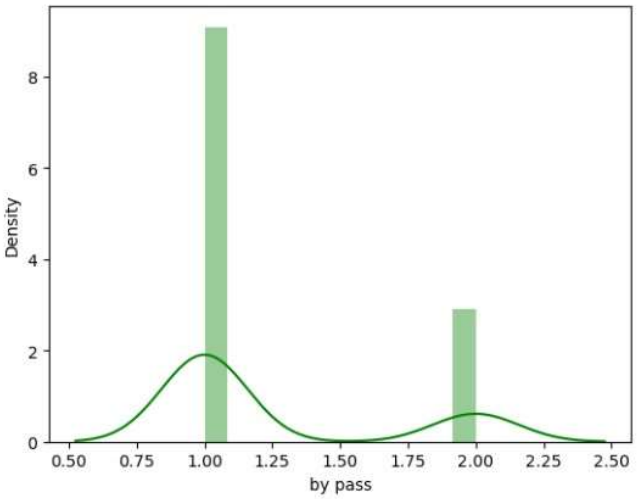
```
dtype: int64

In [10]: df.describe()

Out[10]:
```

	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.	
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	14
mean	2.452703	2.743243	1.554054	1.175676	1.047297	1.243243	1.506757	1.067568	1.702703	1.060811	2.472973	2.398649	
std	0.575396	0.817509	0.498757	0.381836	0.212995	0.430498	0.501652	0.251855	0.458621	0.313557	0.836627	0.568323	
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	
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	
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	
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	
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	

In [14]:
```
sns.distplot(df["by pass"],color='g')
```

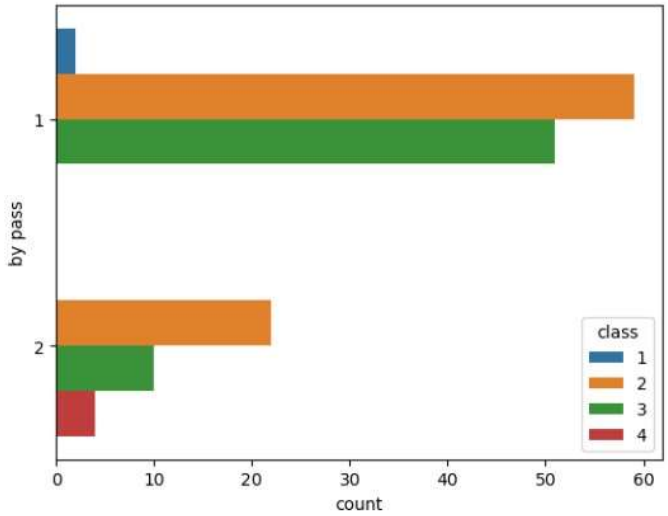
```
C:\Users\DELL\AppData\Local\Temp\ipykernel_3384\2194164255.py:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df["by pass"],color='g')
```

Out[14]: <Axes: xlabel='by pass', ylabel='Density'>



In [15]:
```
sns.countplot(data=df,y="by pass",hue="class")
```

Out[15]: <Axes: xlabel='count', ylabel='by pass'>

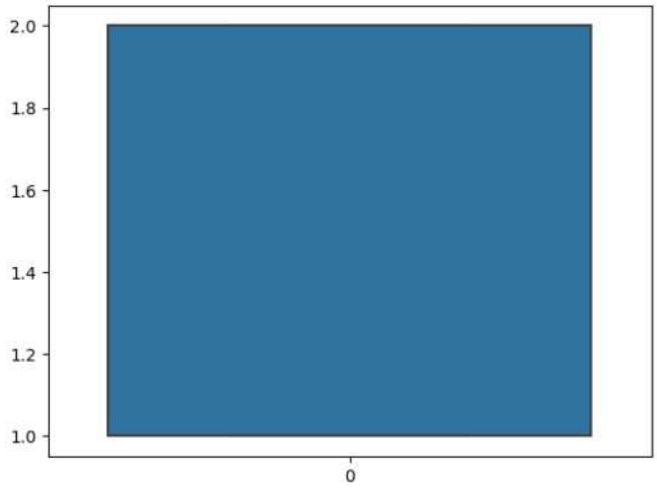


In [16]:
```
df.shape
```

Out[16]: (148, 19)

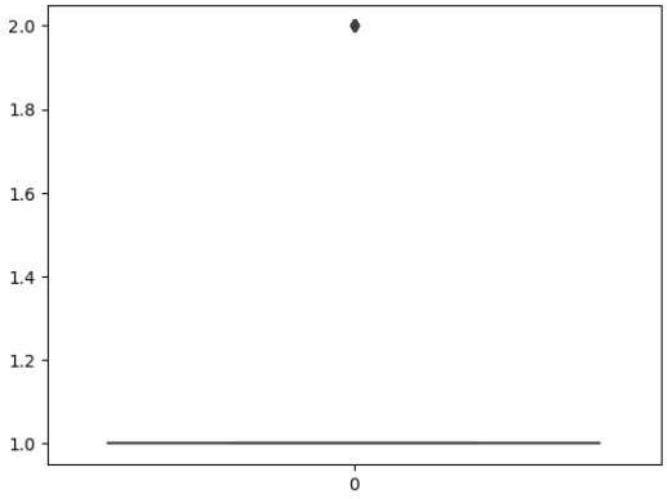
In [17]:
```
sns.boxplot(df["block of affere"])
```

Out[17]: <Axes: >



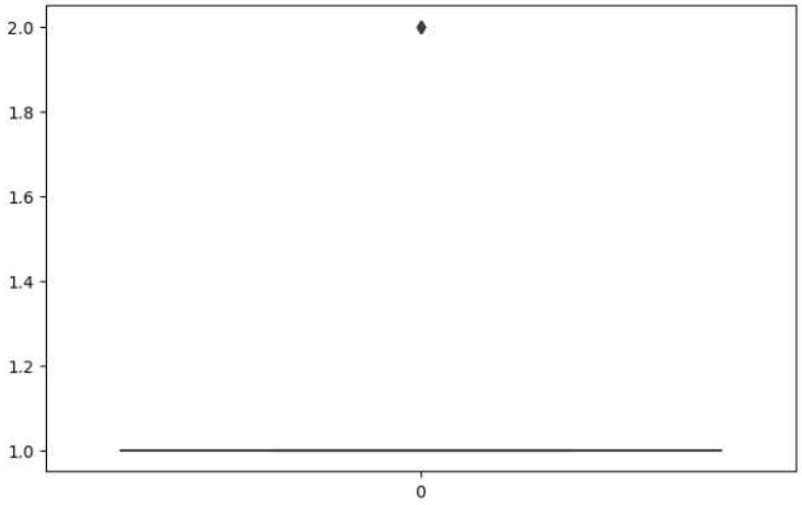
In [18]:
```
sns.boxplot(df["bl. of lymph. c"])
```

Out[18]: <Axes: >



In [19]:
```
plt.figure(figsize=(8,5))
sns.boxplot(df["bl. of lymph. s"])
```

Out[19]: <Axes: >



In [25]:
```
x=df.iloc[:,1:]
```

In [24]:
```
x.head()
```

Out[24]:

	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	dislocatio c
0	4	2	1	1	1	1	1	2	1	2	2	2	4	8	1	
1	3	2	1	1	2	2	1	2	1	3	3	2	3	4	2	
2	3	2	2	2	2	2	2	2	1	4	3	3	4	8	3	
3	3	1	1	1	1	2	1	2	1	3	3	4	4	4	3	
4	3	1	1	1	1	1	1	1	1	2	2	4	3	5	1	

In [27]:
```
y=df["class"]
```

In [28]:
```
y
```

Out[28]:
```
0      3
1      2
2      3
3      3
4      2
      ..
143    3
144    2
145    3
146    2
147    2
Name: class, Length: 148, dtype: int64
```

```
Name: class, Length: 148, dtype: int64
```

In [29]:
```
x.shape
```

Out[29]:
```
(148, 18)
```

In [30]:
```
y.shape
```

Out[30]:
```
(148,)
```

In [31]:
```
y.nunique()
```

Out[31]:
```
4
```

In [32]:
```
y.unique()
```

Out[32]:
```
array([3, 2, 4, 1], dtype=int64)
```

In [33]:
```
x.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 148 entries, 0 to 147
Data columns (total 18 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   lymphatics       148 non-null    int64
 1   block of affere  148 non-null    int64
 2   bl. of lymph. c  148 non-null    int64
 3   bl. of lymph. s  148 non-null    int64
 4   by pass          148 non-null    int64
 5   extravasates     148 non-null    int64
 6   regeneration of  148 non-null    int64
 7   early uptake in  148 non-null    int64
 8   lym.nodes dimin  148 non-null    int64
 9   lym.nodes enlar  148 non-null    int64
 10  changes in lym.  148 non-null    int64
 11  defect in node   148 non-null    int64
 12  changes in node  148 non-null    int64
 13  changes in stru  148 non-null    int64
 14  special forms    148 non-null    int64
 15  dislocation of   148 non-null    int64
 16  exclusion of no  148 non-null    int64
 17  no. of nodes in  148 non-null    int64
dtypes: int64(18)
memory usage: 20.9 KB
```

In [35]:
```
from sklearn.preprocessing import StandardScaler
```

In [37]:
```
sc=StandardScaler()
```

In [38]:
```
#feature scaling
from sklearn.preprocessing import MinMaxScaler
ms=MinMaxScaler()
x_scaled=pd.DataFrame(ms.fit_transform(x),columns=x.columns)
```

In [39]:
```
x_scaled
```

Out[39]:

	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	dislo
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	
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	
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	
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	
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	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
143	0.666667	1.0	0.0	0.0	1.0	1.0	0.0	1.0	0.0	0.333333	0.5	1.000000	0.666667	0.571429	0.5	
144	0.333333	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.000000	0.000000	0.285714	0.0	
145	0.333333	1.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.666667	1.0	0.666667	0.666667	1.000000	1.0	
146	0.333333	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.333333	0.5	1.000000	0.333333	0.142857	0.0	
147	0.333333	1.0	1.0	0.0	1.0	1.0	0.0	1.0	0.0	0.666667	1.0	1.000000	0.666667	0.428571	1.0	

148 rows × 18 columns

In [40]:
```
#Splitting Data into Train and Test.
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x_scaled,y,test_size=0.2,random_state=0)
```

In [41]:
```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

Out[41]:
```
((118, 18), (30, 18), (118,), (30,))
```

In [42]:
```
x_train.head()
```

Out[42]:

	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	dislo
33	0.333333	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.5	0.333333	0.666667	0.285714	0.0	
78	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.000000	0.000000	0.000000	0.0	
18	0.666667	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.333333	0.5	0.333333	0.666667	0.142857	0.0	
127	0.333333	1.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.333333	0.5	0.666667	0.666667	0.428571	0.5	
63	1.000000	1.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.333333	1.0	0.333333	0.666667	0.142857	0.5	

In [43]:
```
from sklearn.ensemble import RandomForestClassifier
rf=RandomForestClassifier()
```

In [44]:
```
params={
    'max_depth':[9,10,11],
    'min_samples_leaf':[2,3],
    'n_estimators':[90,95,100,110],
    'max_features':[2,3,4,5]
}
```

In [45]:
```
from sklearn.model_selection import GridSearchCV
```

In [46]:
```
grid_search=GridSearchCV(estimator=rf,
                         param_grid=params,
                         cv=2,
                         verbose=1,
                         scoring="accuracy")
```

In [47]:
```
grid_search.fit(x_train,y_train)
```

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

Out[47]:

GridSearchCV
estimator: RandomForestClassifier
RandomForestClassifier

In [48]:
```
grid_search.best_score_
```

Out[48]:
```
0.7966101694915255
```

In [49]:
```
rf_classify=RandomForestClassifier(random_state=42,
                                   n_jobs=-1,
                                   max_depth=9,
                                   min_samples_split=2,
                                   max_features='sqrt',
                                   n_estimators=90,
                                   bootstrap=True)
```

In [50]:
```
rf_classify.fit(x_train,y_train)
```

Out[50]:
```
RandomForestClassifier(max_depth=9, n_estimators=90, n_jobs=-1, random_state=42)
```

In [51]:
```
from sklearn.metrics import accuracy_score
```

In [52]:
```
prediction=rf_classify.predict(x_test)
```

In [53]:
```
from sklearn.metrics import accuracy_score,f1_score,confusion_matrix,classification_report
```

In [54]:
```
confusion_matrix(y_test,prediction)
```

Out[54]:
```
array([[11,  1,  0],
       [ 2, 15,  0],
       [ 1,  0,  0]], dtype=int64)
```

In [63]:
```
import pickle

pickle.dump(rf_classify,open('saved_mode2.pkl','wb'))
pickle.dump(ms,open('ms_saved.pkl','wb'))
```

In [64]:
```
x_train.head()
```

Out[64]:

	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	dislo
33	0.333333	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.5	0.333333	0.666667	0.285714	0.0	
78	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.000000	0.000000	0.000000	0.0	
18	0.666667	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.333333	0.5	0.333333	0.666667	0.142857	0.0	
127	0.333333	1.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.333333	0.5	0.666667	0.666667	0.428571	0.5	
63	1.000000	1.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.333333	1.0	0.333333	0.666667	0.142857	0.5	

In [65]:
```
df.head()
```

Out[65]:

	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	dis
0	3	4	2	1	1	1	1	1	2	1	2	2	2	4	8	1	
1	2	3	2	1	1	2	2	1	2	1	3	3	2	3	4	2	
2	3	3	2	2	2	2	2	2	2	1	4	3	3	4	8	3	
3	3	3	1	1	1	1	2	1	2	1	3	3	4	4	4	3	
4	2	3	1	1	1	1	1	1	1	1	2	2	4	3	5	1	

In [66]:
```
y_test1=rf_classify.predict(ms.transform([[4,2,1,1,1,1,1,2,1,2,2,2,4,8,1,1,2,2]]))
```

```
C:\Users\DELL\anaconda3\Lib\site-packages\sklearn\base.py:439: UserWarning: X does not have valid feature names, but MinMaxScal
er was fitted with feature names
  warnings.warn(
C:\Users\DELL\anaconda3\Lib\site-packages\sklearn\base.py:439: UserWarning: X does not have valid feature names, but RandomFore
stClassifier was fitted with feature names
  warnings.warn(
```

In [67]:
```
y_test1[0]
```

Out[67]:
```
3
```