Thesis Report

A Comparative Analysis of Machine Learning Models for Liver Disease Prediction

THESIS SUPERVISOR

Prof. Dr. Md. Ashikur Rahman Khan
Chairman
Department of ICE, NSTU

SUBMITTED BY

Raihan Uddin
Roll No: ASH1911023M
Session: 2018-2019
Year 04, Term 02
Department of ICE, NSTU

DEPARTMENT OF INFORMATION AND COMMUNICATION ENGINEERING

NOAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

13th May 2024

ACCEPTANCE

I hereby declare that I have checked this thesis and, in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Bachelor of Science degree in Information & Communication Engineering.

_____

Professor Dr. Md Ashikur Rahman Khan
Chairman,
Department of Information and Communication Engineering
Noakhali Science & Technology University

STUDENT'S DECLARATION

This thesis is submitted to the Department of Information and Communication Engineering of Noakhali Science & Technology University in partial fulfillment of the requirements for the award of the Bachelor of Science (B.Sc.) Engineering in Information and Communication Engineering (ICE). So, I hereby declare that this report has not been submitted elsewhere for the requirement of any kind of degree, diploma, or publication.

_____

Raihan Uddin
Student of Information and Communication Engineering,
Roll No.: ASH1911023M
Session: 2018-2019
Date: 13. 05. 2024

ACKNOWLEDGEMENT

I would like to thank my advisor, Professor Dr. Md Ashikur Rahman Khan, for his immense support and guidance throughout my bachelor's degree here at Noakhali Science & Technology University. Without his assistance, this thesis paper would be extremely difficult for me to complete. I am quite appreciative of all of his assistance. His pragmatism and advice helped me throughout my journey here. I am sure that the lessons learned will help me in my future endeavors.

My sincere thanks to all my classmates and members of the staff of the Information and Communication Engineering Department, NSTU, who helped me in many ways. And also, thanks to the authorities of those institutions who approved us to collect data. Last but not least, I would like to thank my parents for encouraging me and standing with me throughout my life.

ABSTRACT

Liver disease is a widespread global health concern that demands early detection for effective treatment and improved patient outcomes. Traditional diagnostic methods have their limitations, necessitating further exploration into the use of machine learning to enhance predictive accuracy, accessibility, and cost-effectiveness in liver disease diagnosis. This study provides an in-depth analysis of machine learning classifiers such as Support Vector Machine, K-Nearest Neighbors, Decision Tree, Multilayer Perceptron, and Random Forest for liver disease prediction and compares their performance. The primary objective of this research is to identify the most effective machine-learning classifier for precise and accessible liver disease prediction. This research has the potential to significantly impact patient care, as it can aid in the development of efficient diagnostic tools, reduce the global burden of liver disease, and improve patient outcomes. This study's robust methodology includes feature selection via the Pearson Correlation matrix, leveraging the 10-fold cross-validation technique, and a balanced dataset generated using the Synthetic Minority Over-sampling Technique (SMOTE) on our collected dataset from distinct hospitals across Bangladesh. Our approach, employing a Random Forest classifier, achieved an impressive 80.5 percent accuracy in liver disease prediction. Furthermore, our model demonstrated a precision of 0.88 and a recall of 0.85, indicating its robust performance in identifying positive and negative cases.

Keywords: Liver Disease, Pearson Correlation, Support Vector Machine, K-Nearest Neighbors, Decision Tree, Random Forest, Multilayer Perceptron.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Liver disease has emerged as a global threat, accounting for around two million deaths each year, or 4% of all global deaths (1 in 25) [1]. In particular, liver cancer alone kills 600,000 to 900,000 people each year. Cirrhosis and hepatocellular carcinoma complications are the leading causes of death, with acute hepatitis accounting for a smaller proportion. Notably, Asia accounts for 75% of all liver cancer cases, which are strongly associated with hepatitis B and hepatitis C infections. Addressing this crisis is vital because diagnosing liver disease early can prevent severe outcomes and reduce the significant healthcare expenses linked to advanced-stage liver diseases.

Early detection of liver illness is critical. Early identification enhances the prospect of more effective treatment and decreases the significant healthcare expenditures associated with advanced-stage liver illnesses. While traditional diagnostic procedures have proven useful, they frequently have accessibility, scalability, and cost-effectiveness restrictions. In response, machine learning has emerged as a powerful tool in solving the diagnostic issues provided by liver illnesses at the interface of healthcare and technology.

In recent years, machine learning has emerged as a powerful technique for solving the issues of liver disease diagnostics. Its ability to evaluate large datasets, identify subtle patterns, and make correct predictions gives hope in the field of hepatology. Machine learning models can improve early identification, risk assessment, and individualized

treatment techniques for liver illnesses.

This research focuses on using machine learning to diagnose liver disease. The major goal is to create predictive models that can properly determine whether a patient has liver disease, hence serving as a beneficial tool for early detection.

1.2 PROBLEM STATEMENT

Liver disease represents a significant health burden worldwide, encompassing various conditions with varying etiologies and clinical manifestations. Early detection and accurate prediction of liver disease are crucial for effective patient management and improved outcomes. However, existing predictive models in the field often fall short due to their reliance on generic, publicly available datasets that may not fully capture the complexity of Bangladesh's local clinical scenarios. Moreover, inadequate data preprocessing techniques and suboptimal feature selection methods further undermine these models' performance and clinical utility.

The primary challenge lies in developing predictive models that can effectively leverage local patient data from hospitals, ensuring authenticity, relevance, and representativeness of the underlying population. Furthermore, the data preprocessing phase poses significant hurdles, including addressing data quality issues, handling missing values, and identifying and removing noise and outliers. The resulting predictive models may suffer from inaccuracies and reduced clinical interpretability without robust preprocessing techniques.

Another critical aspect is featuring selection, where the challenge is to identify the most informative variables that contribute significantly to liver disease prediction while minimizing redundancy and overfitting. Traditional methods often overlook the intricate relationships among variables, leading to suboptimal model performance. A more systematic approach is needed to select features that enhance prediction accuracy and facilitate clinical understanding and decision-making.

Moreover, the choice of machine learning algorithms plays a pivotal role in developing effective predictive models. With many algorithms available, selecting the most

appropriate ones for liver disease classification requires careful consideration of model complexity, interpretability, and scalability. Each algorithm has its strengths and limitations, and the challenge lies in identifying the optimal combination of algorithms that can deliver accurate, clinically interpretable, and practicable predictive models.

Therefore, the overarching goal of this study is to address these challenges by proposing a comprehensive methodology for enhancing liver disease prediction models. By leveraging original local patient data from hospitals, employing rigorous data preprocessing techniques, refining feature selection methods, and evaluating a diverse set of machine learning algorithms, this research aims to develop predictive models that are accurate, reliable, clinically relevant, and actionable. Ultimately, the successful development and implementation of such models have the potential to revolutionize early diagnosis and patient treatment in hepatology, leading to improved healthcare outcomes and quality of life for individuals affected by liver disease.

1.3 OBJECTIVES

This research paper focuses on machine learning models and offers improved results that can assist doctors in making accurate diagnoses of liver disease. The following are the objectives of the research:

I. To generate our dataset by collecting original data from distinct hospitals across Bangladesh
II. To evaluate multiple classification algorithms
III. To perform a comparative analysis
IV. To enhance liver disease prediction

These objectives collectively direct our research toward creating an accurate and clinically applicable predictive model for liver disease detection

1.4 SCOPE OF THE STUDY

This thesis aims to identify a reliable machine-learning model for early liver disease detection, focusing on datasets collected from multiple hospitals across Bangladesh.

The study population encompasses patients from diverse demographic backgrounds and clinical profiles, comprehensively representing the Bangladeshi population's health status. Geographically, data collection spans various hospitals of Bangladesh, ensuring a broad geographical coverage and relevance to the local healthcare context. By employing a range of machine learning algorithms and evaluation techniques, the study seeks to identify the most accurate and reliable model for detecting liver diseases at an early stage. Through rigorous analysis and comparison of these models, the research aims to advance healthcare practices in Bangladesh by enhancing disease detection capabilities and informing clinical decision-making processes.

1.5 THESIS OUTLINE

Chapter 1 discusses the significance of the research, emphasizing early disease detection in liver diseases within the healthcare sector and thoroughly exploring the rationale behind the topic selection. Chapter 2 provides a comprehensive literature review, synthesizing previous work in early disease detection and machine learning applications in healthcare. It identifies key findings, methodologies, and gaps in current research, forming the theoretical framework for the methodology chapter. Chapter 3 details the research methodology, including data collection procedures, dataset characteristics, preprocessing techniques, the selection and implementation of machine learning algorithms for early liver disease prediction, and the evaluation metrics for performance assessment. Chapter 4 presents the study's findings, including the performance evaluation of different classification models for early liver disease detection, and discusses the implications of the results within the context of existing literature and theoretical frameworks. Chapter 5 concludes the thesis, summarizing the key findings and contributions and outlining potential future research directions.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

Numerous studies have investigated the imperative healthcare challenge of early disease prediction, exploring an array of machine-learning algorithms and methodologies. These studies have undertaken comprehensive comparative analyses to scrutinize the efficacy of diverse algorithms, feature selection techniques, and data preprocessing methodologies in enhancing predictive models for healthcare applications. Through rigorous examination and evaluation, researchers have endeavored to advance the state-of-the-art in disease prediction, striving towards more accurate and reliable models that can contribute significantly to early diagnosis and intervention strategies.

2.2 OBJECTIVES OF REVIEW

The objectives of the literature review chapter are multifaceted, encompassing a comprehensive examination of previous research within the field of early disease detection and machine learning applications in healthcare. Firstly, the literature review aims to identify and synthesize existing knowledge and findings related to liver disease prediction models, highlighting key methodologies, algorithms, and performance metrics utilized in previous studies. Secondly, it seeks to evaluate the strengths and limitations of different approaches employed in prior research, providing critical insights into the current state-of-the-art in the domain. Additionally, the literature review chapter aims to identify gaps and areas for further investigation, guiding the development of the research methodology and informing the direction of the study.

I. Analyze the techniques and methods employed in earlier research.
II. Determine the advantages and disadvantages of earlier strategies.

III. Draw attention to any gaps in the literature that need more research.
IV. Provide a theoretical foundation for the research.
V. Provide perspectives and possible directions for further investigation.

2.3 RELATED WORKS

The BUPA and ILPD datasets were analyzed by J. Singh et al. [2] to examine several machine learning methods, including Support Vector Machine, Naive Bayes, Decision Tree, Random Forest, Logistic Regression, and k-Nearest Neighbors. According to the results, Logistic Regression had a 72% accuracy rate on the ILPD dataset, and Random Forest had the greatest accuracy of 73% on the BUPA dataset.

Waikato Environment for Knowledge Analysis software was used by M. Singaravelu et al. [3] to preprocess data and apply machine learning methods. Various methods are examined for accuracy, and the Random Tree approach has the highest accuracy of 74.2%.

The Liver Disease Prediction approach was developed by D. Bhupathi et al. [4] and uses five algorithms: Classification and Regression Trees, Naïve Bayes, K-Nearest Neighbors, Linear Discriminant Analysis, and Support Vector Machine. The autoencoder network outperformed the K-NN method with 92.1% accuracy, suggesting its potential for liver disease prediction. The K-NN approach had the best accuracy of 91.7%.

Kuzhikalail et al. [5] used the Indian Liver Patient dataset from the UCI machine learning repository to assess several machine learning algorithms to predict liver disease. Different models of classification were examined. The genetic algorithm was used for feature selection. After removing outliers and feature selection, the Random Forest classifier has achieved 88% accuracy.

L. Anand et al. [6] suggested a model that improves accuracy over current techniques by categorizing patient data using extracted characteristics obtained using M-PSO and ANN. The performance of the proposed ANN technique is demonstrated in comparison

to existing classification systems, such as C5.0 and CHAID, using metrics for specificity, accuracy, and sensitivity.

Sayyidiyah Alfisahrin et al. [7] have proposed to use the 10 essential features of liver illness to determine whether or not a patient has a liver condition. In this case, employ the NB Tree, Naive Bayes, and Decision Tree algorithms. According to the outcome, the NB Tree algorithm provides the best accuracy. The NB Tree algorithm's performance will be used to determine the most crucial element for categorizing patients with liver illness to increase accuracy in the future.

Naive Bayes, Decision Trees, Random Forest, Logistic Regression, K-NN, SVM, Artificial Neural Networks, and C4.5 were utilized by S. Tokala et al. [8] In the prediction of liver illness. The random forest classifier has an accuracy of 88% among them.

Using the WEKA tool, Naik et al. [9] developed a model to analyze liver disease. The Naive Bayes, Decision Tree (DT), and J48 algorithms were used to develop their proposed model. The accuracy and execution time of the algorithms were assessed, and the outcomes were contrasted with those of the other possibilities. The results showed that in terms of accuracy, the J48 and DT algorithms performed better than the Naive Bayes method.

Using accuracy, precision, and recall as three main criteria for computation, Bahramirad et al. [10] deployed eleven data mining classification models to classify the AP and BUPA datasets. Some models examined in this study displayed better in the AP dataset than in the BUPA dataset.

A.K.M. Sazzadur et al. [11] aimed to lower the cost of diagnosis by assessing several machine-learning methods for predicting chronic liver disease. The following six algorithms were employed: Random Forest, K Nearest Neighbors, Decision Tree, Support Vector Machine, Naïve Bayes, and Logistic Regression. Among the algorithms tested, logistic regression came out at the top with an accuracy of 75%.

Saima et al. used supervised machine learning classification methods such as logistic regression, decision tree, random forest, AdaBoost, KNN, linear discriminant analysis, gradient boosting, and support vector machine. [12] suggested a model for liver disease detection. To improve prediction accuracy, the LASSO feature selection approach is used to find LD features that are highly correlated. The decision tree approach performs best when paired with LASSO feature selection, with f1-score values of 96%, 99%, 92%, and 94.295% for accuracy, precision, sensitivity, and f1-score, respectively.

Using a variety of machine learning techniques, such as SVM, Random Forest, Decision Trees, Artificial Intelligence, and Naïve Bayes, Auxilia [13] produced an accurate prediction for liver illness. R was used to conduct the study on the 583 occurrences and 11 characteristics of the Indian Liver Patient Records dataset. The results showed that the SVM algorithm had the greatest accuracy at 77%, followed by Random Forest at 77%, Decision Trees at 81%, Artificial Intelligence at 71%, and Naïve Bayes at 37%. It was found that the Decision Trees method had the lowest accuracy.

Atlaf Ifra et al. suggested a bagging meta-estimator and a feed-forward neural network-based voting ensemble with a mean decrease in impurity feature selection. [14] to identify disease datasets. The feature selection and voting approach helped the ensemble model perform better than individual base learners, increasing accuracy. The ensemble classifier's average Mean Squared Error, bias, and variance were determined to be 0.311, 0.217, and 0.094 for the Indian Liver Patient dataset and 0.233, 0.186, and 0.047 for the PIMA Indians diabetes dataset, respectively. The accuracy of the stacked ensemble using three base models on the ILPD dataset was 73.4.

B. Ramana et al. [15] assessed the accuracy, sensitivity, precision, specificity, and ROC area of many classification algorithms using health datasets. Bagging performs well on the ILPD dataset, while multilayer perceptron performs well in liver disorders.

According to Dr. S. Vijayarani et al. [16], the purpose of this research is to use classification algorithms to predict liver disorders. Support vector machines and Naïve Bayes algorithms were employed in this work. Based on performance metrics like execution time and classification accuracy, these algorithms are compared. Based on

the findings, this study concludes that the SVM classifier—which provides the highest classification accuracy—is the best classification method. In contrast, the Naïve Bayes classifier requires the least amount of execution time when compared to other implementations, where the SVM is a better classifier for predicting liver diseases.

From the work of V. Chawla et al. [17] it's evident that the Synthetic Minority Over-sampling Technique significantly enhances classifier accuracy for minority classes compared to traditional methods like plain under-sampling. SMOTE's effectiveness lies in its ability to induce focused learning and introduce a bias towards the minority class, resulting in broader decision regions encompassing nearby minority class points. This characteristic allows classifiers to learn from a more diverse set of minority class samples, leading to improved performance across various datasets with varying levels of class imbalance. Overall, the findings underscore SMOTE's value in addressing imbalanced class distributions and improving classifier performance in such scenarios.

A technique to identify and track coronary artery disease was proposed by Otoom et al. [18] Three algorithms—Bayes Net, Support Vector Machine, and Functional Trees—were employed along with two tests. WEKA is the tool utilized for the prediction method in this article. This article employed seven features that were chosen at random. Bayes Net provided an accuracy of 84.5%, SVM provided an accuracy of 85.1%, and FT accurately identified 84.5% of the features.

Pushpendra Kumar et al. [19] used SVM and KNN on ILPD and MPRLPD datasets using a four-fold cross-validation technique. SVM performed better than KNN in the MPRLPD dataset when it was balanced. SVM achieved 96.42% accuracy. KNN achieved 74.67% accuracy on a balanced ILPD dataset.

Anju Gulia et al. [20] worked on the UCI ILPD Dataset. He employed SVM, J-48, MLP, Random Forest, and Bayesian Network classifier. A different result was obtained with and without feature selection. Without feature selection, SVM performed better with 71.35% accuracy, and with feature selection, KNN performed better than SVM, with 71.87% accuracy.

Ruhul A et al. [21] combined PCA, FA, and LDA for feature extraction. He used both train test split and 10-fold cross-validation to compare the results of both approaches. With effective feature extraction methods, he achieved excellent results. On a balanced ILPD dataset with a train-test split, MLP showed 85.50% accuracy. On the other hand, on a balanced ILPD dataset with 10-fold cross-validation, Random Forest showed 87.80% accuracy.

Using the BUPA and ILPD datasets, J. Singh et al. [22] examined liver disease prediction using machine learning methods such as SVM, NB, DT, RF, ANN, LR, and k-NN. RF obtained an accuracy rate of 73% on the BUPA dataset, whereas LR demonstrated a 72% accuracy rate on the ILPD dataset.

A variety of classification models, including MLP, Logistic Regression, Random Forest, k-nearest neighbor, Gradient Tree Boosting, Naive Bayes, and Support Vector Machine Voting classifier, were examined by E. Dritsas et al. [23] on the ILPD dataset. Using SMOTE with 10-fold cross-validation, the voting classifier proved to be the best-performing model, with an accuracy, recall, and F-measure of 80.1%, 80.4% precision, and 88.4% AUC.

2.4 RESEARCH GAP

While studies have primarily relied on classification accuracy as the main evaluation metric for liver disease prediction models, there exists a notable gap in considering other performance measures such as precision, recall, and F1-score. A more comprehensive comparison of these evaluation metrics is essential to provide a holistic assessment of the predictive capabilities of different models. Additionally, despite the prevalence of imbalanced datasets in liver disease prediction research, there is a lack of exploration into advanced techniques for handling class imbalance, particularly the Synthetic Minority Oversampling Technique. Future research should focus on evaluating the effectiveness of various approaches for addressing class imbalance and their impact on model performance, thus bridging the gap between current methodologies and the evolving needs of the field.

2.5 CONTRIBUTIONS

In this study, a comprehensive dataset was meticulously curated by collecting patient data from various hospitals across Bangladesh, ensuring a diverse and representative sample. Extensive preprocessing was performed to enhance the dataset's quality and suitability for analysis. Subsequently, a thorough performance evaluation of various classification models was conducted, employing multiple performance metrics to provide a robust and detailed assessment of each model's predictive capabilities. This work not only contributes a valuable dataset to the field of liver disease prediction but also offers a rigorous comparative analysis of machine learning algorithms, facilitating the identification of the most effective approaches for early detection of liver disease.

2.6 SUMMARY

The literature review highlights the utilization of various machine learning algorithms in liver disease prediction research, showcasing promising accuracy rates. However, there is a need for a more comprehensive evaluation of performance metrics beyond classification accuracy, including precision, recall, and F1-score. Additionally, addressing the prevalence of imbalanced datasets in liver disease prediction is crucial, with synthetic oversampling methods like SMOTE offering potential solutions. Future research should focus on bridging these gaps to enhance the accuracy and reliability of liver disease prediction models.

CHAPTER 3

METHODOLOGY

3.1 INTRODUCTION

Our primary objective is to develop an effective predictive model for the early detection of liver disease. To do so, we must maintain a systematic process from the beginning to the end. Our methodology should be parallel to achieve the intended objective at the end. Our approach consists of many sequential steps necessary to obtain effective results. This chapter outlines the detailed methodology followed for early liver disease prediction. The workflow diagram of the whole process is shown in Figure 3.1.

The following sections will elaborate on important elements necessary for the progress of our research. Section 3.2 will cover data collection and the essential elements of data processing in detail. Section 3.3 will briefly describe the classification models we will use. Lastly, Section 3.4 will describe how our suggested technique is implemented, including the steps necessary to convert the results of our research into useful insights.

3.2 DATA COLLECTION

Our research starts with gathering and preprocessing the dataset to eliminate noise or unnecessary information. This section outlines the procedures we will take to ensure that the dataset is correctly handled for our next step. We will apply data cleaning techniques such as normalization, missing value imputation, and outlier detection to enhance data quality. Ensuring the integrity and accuracy of the dataset is crucial for effectively applying machine learning models in the subsequent phases of the study.

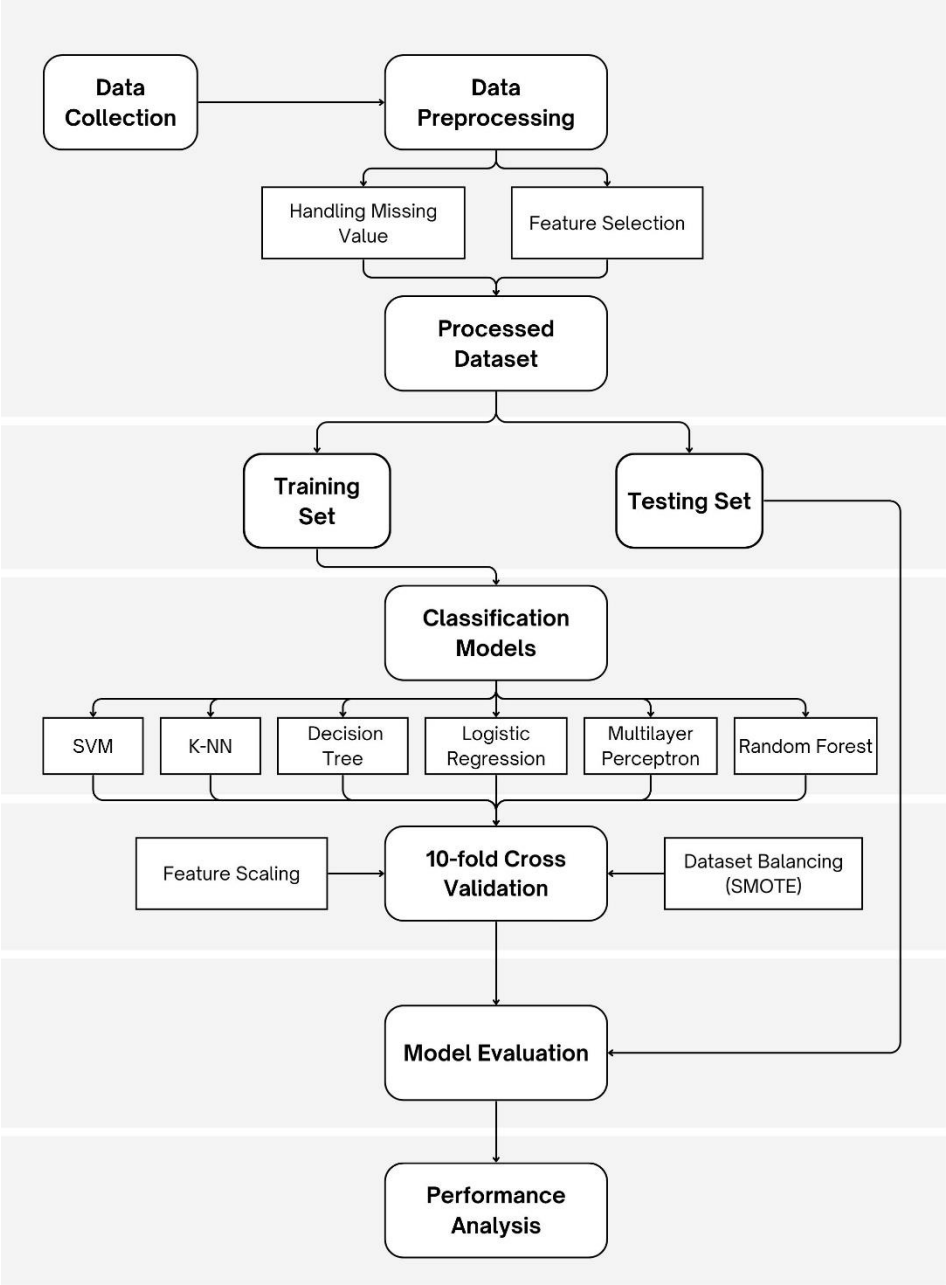

Figure 3.1: Workflow diagram

3.2.1 Features Description

The diagnosis of liver disease is done using various substances. These substances are required for early liver disease detection. Our dataset consists of two demographical data (Age, Gender) and seven substances produced by the liver, which are measured by blood tests. The features of our dataset are comprised of these nine variables and a target feature. All of the substances are briefly described below:

Direct Bilirubin & Total Bilirubin: A byproduct of the breakdown of old red blood cells is bilirubin. It is commonly assessed as direct and total bilirubin in the blood. Elevated levels of bilirubin suggest that the liver is having difficulty converting the substance into bile or that there could be a blockage in the bile ducts.

Albumin: The primary protein produced by the liver is Albumin. It carries out several vital biological processes. Our liver may not be operating at optimal capacity if we receive a low score on this test. This happens in conditions including cancer, cirrhosis, and starvation.

Alanine Aminotransferase: Liver cells contain an enzyme called Alanine Aminotransferase, which aids in converting proteins into energy. Alanine Aminotransferase levels rise in response to liver injury because it gets released into circulating blood. An alternative name is SGPT.

Aspartate Aminotransferase: Aspartate Aminotransferase aids in the body's breakdown of amino acids. Like Alanine Aminotransferase, it is often seen in blood at low concentrations. Increased AST values might indicate liver illness, injury to the liver, or harm to the muscles. An alternative name for this is SGOT

Total Proteins: Total protein measures the total amount of two types of proteins found in the liquid component of your blood: albumin and globulin. Liver malfunction may cause a decrease in the blood's total protein levels.

Alkaline Phosphatase: Alkaline Phosphatase is an essential enzyme involved in breaking down proteins in the liver and bone. High levels of Alkaline Phosphatase can indicate liver damage or illness, such as a blocked bile duct, or certain bone disorders. This enzyme plays a crucial role in various bodily processes, and its measurement is often used in diagnostic tests to assess liver function and bone health. Elevated levels may signal a range of conditions, making it a vital marker for medical professionals to monitor. Understanding the role and implications of Alkaline Phosphatase levels can aid in the early detection and management of liver and bone diseases.

An additional description of all features is given in Table 3.1.

Table 3.1: Dataset description

Attribute name	Description
Age	Age of the patient
Gender	Male/Female
TB (Total Bilirubin)	normal (less than 0.3), high (from 0.3 to 1.9 mg/dl)
DB (Direct Bilirubin)	low (less than zero), normal (0 to 0.3 mg/dl) and high (above 0.3 mg/dl)
Albumin	low (below 3.4g/dL) normal (3.4 to 5.4 g/dL) high (above 5.4 g/dL)
Sgpt (Alanine Aminotransferase)	For male, low (less than 10 U/L), normal (between 10 to 40 U/L) and high (above 40 U/L), For female, low (less than 7 U/L), normal (between 7 to 35 U/L) and high (above 35 U/L)
Sgot (Aspartate Aminotransferase)	For male, low (below 14 U/L), normal (between 14–20 U/L) and high (above 20 U/L), For female, low (below 10 U/L), normal (between 10–36 U/L) and high (above 36 U/L)
Total Proteins	low (below 6 g/dL) normal (6 to 8.3 g/dL) high (above 8.3 g/dL)
Alkaline Phosphatase	low (below 44 IU/L) normal (44 to 147 IU/L) high (above 147 IU/L)
Disease	Class 1 (Patient has the liver disease) and Class 0 (Patient has no liver disease)

3.2.2 Data Collection

The patient data were collected from various hospitals and diagnostic centers in Chattogram, Noakhali, and Sylhet, reflecting a comprehensive regional coverage. The dataset comprises information on 293 patients, 209 of whom were diagnosed with liver diseases, while the remaining 84 were not, providing a substantial basis for analysis. Specifically, 186 samples were obtained from the Hepatology Department and other wards of Chittagong Medical College Hospital, illustrating the significant contribution of this institution. Additionally, 42 samples were collected from Noakhali Sadar Hospital, and 65 samples were gathered from Sylhet MAG Osmani Medical College, ensuring a diverse and representative sample from different medical facilities.

3.2.3 Dataset Visualization

A detailed dataset visualization is illustrated in this section.

i. Disease Distribution

The collected data consists of 209 patients with liver disease and 84 patients without liver disease, as shown in Figure 3.2.

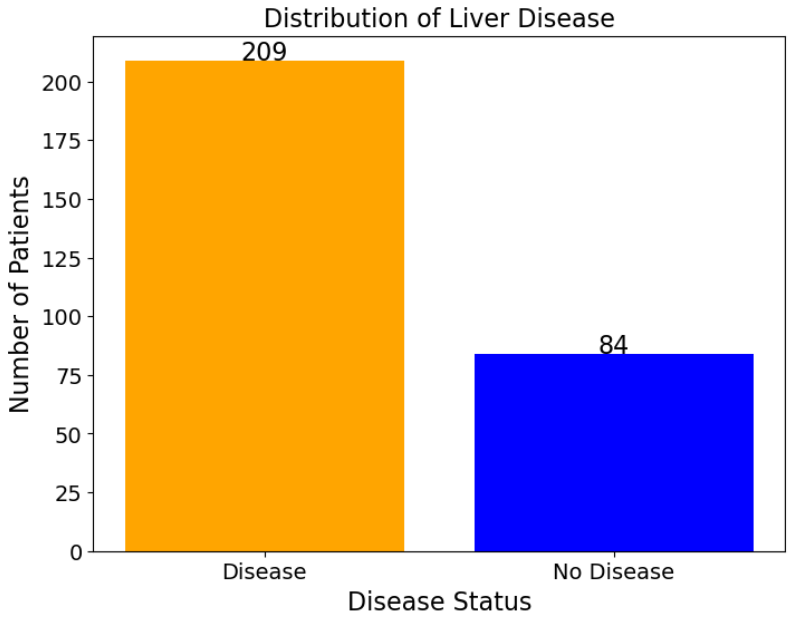

Figure 3.2: Distribution of liver disease

ii. Gender Distribution

Our dataset has information on 218 male and 75 female patients, as shown in Figure 3.3. It shows a large difference between gender classes.

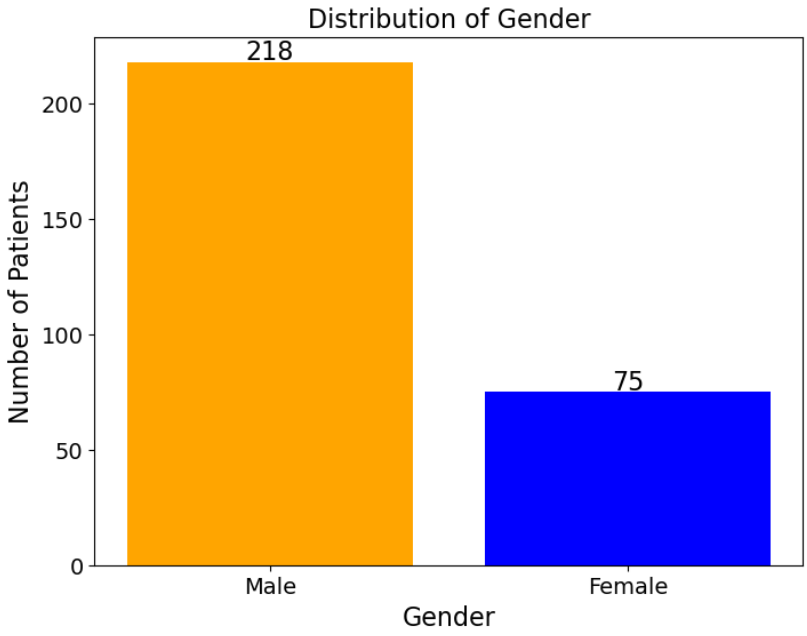

Figure 3.3: Gender Distribution of the dataset

iii. Disease Distribution in Male and Female

As shown in Figure 3.4, our dataset has 160 male patients and 49 female patients with liver disease. The number of female patients with liver disease is almost twice the number of female patients without liver disease. There is a large difference in number of male and female patients with liver disease.

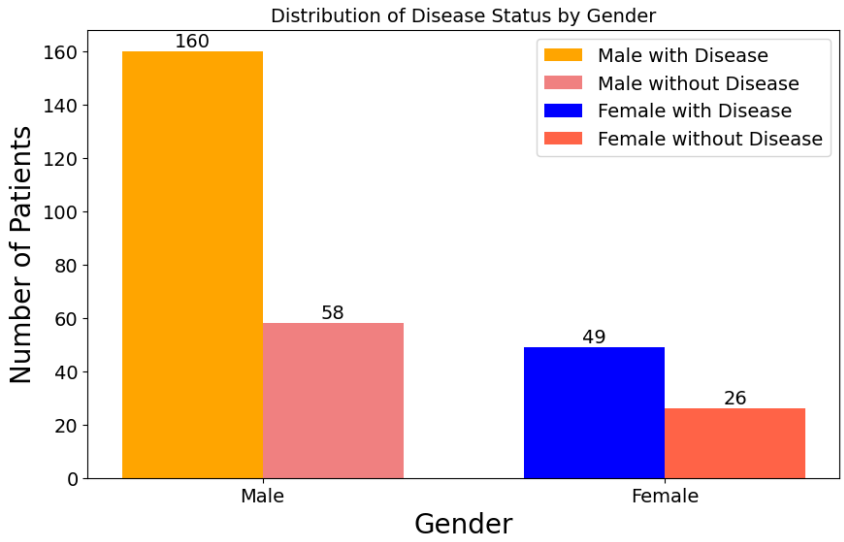

Figure 3.4: Distribution of disease status by gender

iv. Number of Samples in Age Ranges

Most of the patient's ages range from 21 to 80. It shows that 77% of the patients are between 21 to 60 years old.

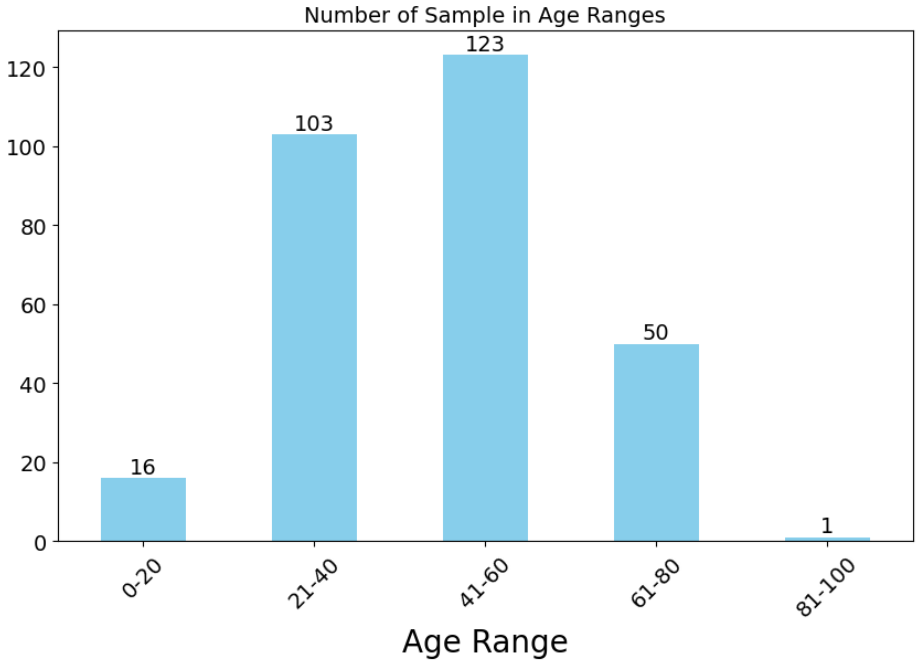

Figure 3.5: Number of patients in different age ranges

It is clearly shown that our dataset is not balanced. We need to take proper steps to overcome the effect of an imbalanced dataset.

3.3 DATA PREPROCESSING

Before working with the dataset, we must preprocess it to remove unnecessary data and apply the necessary techniques to improve its performance.

3.3.1 Handling Missing Values

A dataset with missing values can cause issues while working on this dataset with a classification model. Many classification models do not support missing values. Even if some classification model does support missing values, it generates incorrect results. So, handling missing values in the data preprocessing step is necessary. Our dataset may contain records with missing values of any variables, and we handled those missing values using the imputation method. In this method, we took each feature's mean and replaced each feature's missing value with its mean value.

3.3.2 Feature Scaling

Feature scaling is a common data preprocessing technique. This method prevents features with higher values from dominating features with lower values. It guarantees that each feature contributes equally by maintaining the values of the features on the same scale. Feature scaling must be used if the dataset includes features with varying ranges. If otherwise, the model's performance could be biased, making the learning process more challenging.

From the dataset description section (Section 3.2.1), we saw that our dataset contains features of varying ranges. For this reason, we needed to scale the features. We used Standardization for feature scaling. Standardization is less sensitive to outliers. The equation for standardization is shown below:

$$X_{stand} = \frac{X - mean(X)}{standard\ deviation(X)} \tag{3.1}$$

3.3.3 Dataset Balancing

In section 3.2.1, we saw that our dataset is not properly balanced. This dataset contains 209 patients with liver disease and 84 patients without liver disease. Performing classification over this imbalanced dataset may bias the classification model toward the positive class that defines samples with liver disease. Due to difficulties in the learning process, the model cannot predict the negative class that defines a patient with no liver disease.

To overcome this situation, we used the Synthetic Minority Oversampling Technique (SMOTE). It works by duplicating minority classes and making synthetic examples before fitting them into the classification model, and it doesn't provide any additional information to the dataset. This way, we balanced our dataset and avoided the model being biased toward one class.

3.4 CLASSIFICATION MODELS FOR LIVER DISEASE PREDICTION

In this work, we employ classification models to achieve early detection of liver disease. The task is defined as binary classification, as the primary objective is to correctly predict whether a patient has liver disease. We decided to explore several acknowledged classification models to accomplish this goal. Each of these models will be briefly discussed in the following sections.

3.4.1 Support Vector Machine

Support Vector Machine is a supervised machine learning method generally applied to classification tasks. It tries to find the best hyperplane in a dataset's efficient division of two different classes. In a two-class scenario, the goal of a model is to correctly classify data as class 1 or class 2. There might be several decision boundaries that separate these classes. This boundary is referred to as a straight line in the two-dimensional representation of the data points and a hyperplane in higher-dimensional spaces. The optimal hyperplane targeted by SVM maximizes the margin between the two classes and is distinguished by maximal separation from both, as shown in Figure 3.6.

3.4.2 K-Nearest Neighbors

In machine learning, K-Nearest Neighbor (KNN) is a supervised learning method that can handle problems including regression and classification. The KNN algorithm predicts responses for new data samples based on their similarity to existing or training data. The basic principle of its working is that data points with similar properties typically group together. The outcome of the k-nearest neighbor classification is a class membership. A majority vote among its neighbors classifies an item, assigning it to the class most popular among its k closest neighbors, as shown in Figure 3.7. Distance metrics, which enable the measurement of the degree of similarity between data points, are crucial to the algorithm's functioning. To be more precise, the algorithm uses metrics like the Euclidean distance approach to determine how close different data instances are. The algorithm finds the K nearest neighbors based on their proximity to the given observation when it encounters an observation that is missing from the dataset. The observation is then given a label or value based on the average value or dominant class among its closest neighbors.

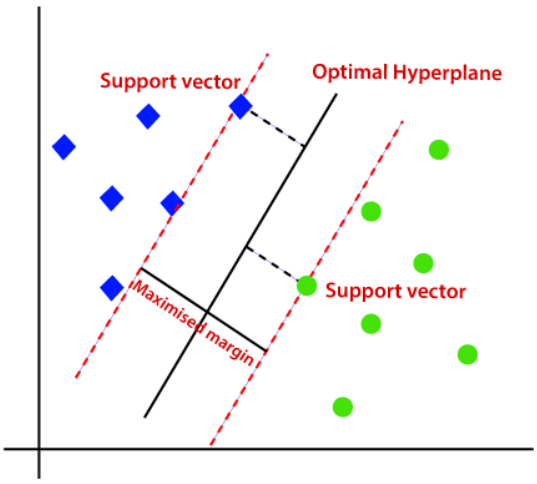

Figure 3.6: Working of SVM classifier

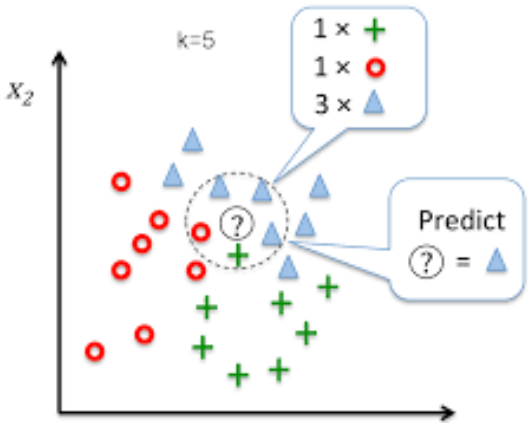

Figure 3.7: Working of KNN classifier

3.4.3 Decision Tree

Decision Tree algorithm belongs to a class of supervised learning algorithms. It is a tree structure that resembles a flowchart, with each leaf node representing the result, the branch representing a decision rule, and the inside node representing a feature (or attribute). The Decision Node and the Leaf Node are the two nodes that make up a

decision tree (shown in Figure 3.8). While leaf nodes represent the result of decisions and do not have any more branches, decision nodes are used to make any kind of decision and have numerous branches. Recursive partitioning is the method by which the tree is divided recursively. By learning simple decisions from past data (training data), a Decision Tree may be used to develop a training model that can be used to predict the class or value of the target variable. The method determines which attribute best fits to partition the data at each internal node based on factors like information gain or Gini impurity. This process is repeated for each subset, ensuring that the tree can handle both categorical and numerical data. Decision Trees are popular due to their simplicity, interpretability, and ability to handle large datasets efficiently. Until a stopping condition is satisfied, such as reaching a maximum depth or having a minimal number of instances in a leaf node, this splitting process keeps on.

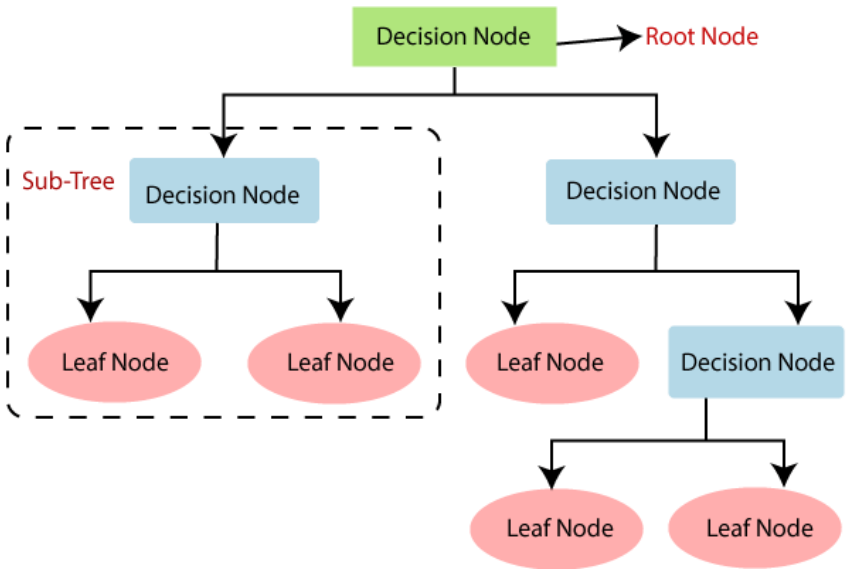

Figure 3.8: Tree creation by decision tree

3.4.4 Logistic Regression

Logistic Regression predicts a binary result given a collection of independent variables. Predictions and their probability are mapped using logistic regression with a logistic function known as the sigmoid function. An S-shaped curve that transforms any real number into a range between 0 and 1 is known as the sigmoid function. Moreover, the model predicts that the instance belongs to that class if the estimated probability

produced by the sigmoid function exceeds a predetermined threshold on the graph. The model anticipates that the instance does not belong in the class if the calculated probability is less than the predetermined threshold. For logistic regression, the sigmoid function is known as an activation function and is described as follows:

$$f(x) = \frac{1}{1 + e^{-x}}$$

where,

e = base of natural logarithms

x = numerical value to transform

The formula for logistic regression is expressed as follows:

$$p(X) = \frac{e^{b0+b1*X}}{1 + e^{b0+b1*X}}$$

where,

$p(X)$ is the probability of the event occurring

$b0$ = y-intercept

$b1$ = coefficient for X

X = input variable

Using maximum likelihood estimation, the coefficients b0 and b1 are obtained from the training data.

3.4.5 Multilayer Perceptron

A type of neural network with feedforward propagation is a multilayer perceptron, which comprises fully connected neurons with a nonlinear activation function. It is frequently used to separate data that is not linearly separable. It comprises an input layer, one or more hidden layers, and an output layer. The input layer gets the input data at the beginning of the operation. In this layer, every neuron corresponds to a feature of the incoming data. Weights are assigned to connections between neurons and are acquired

during training. The network's capacity to identify patterns in the data is greatly aided by these weights, which also establish the strength of the connections. One or more hidden layers exist after the input layer. These layers' neurons process the inputs by performing calculations. Each neuron's output is determined by adding a bias, applying a weighted sum of its inputs (from the preceding layer), and then passing the resultant sum through an activation function. The activation function is essential because it adds non-linearity to the model, which enables it to learn more complex patterns. ReLU (Rectified Linear Unit), sigmoid, and tanh are examples of common activation functions. Dropout regularization is often used in these layers to prevent overfitting by randomly deactivating neurons during training. The last layer produces the output. This layer often employs a sigmoid function for binary classification or a softmax function for multi-class classification tasks.

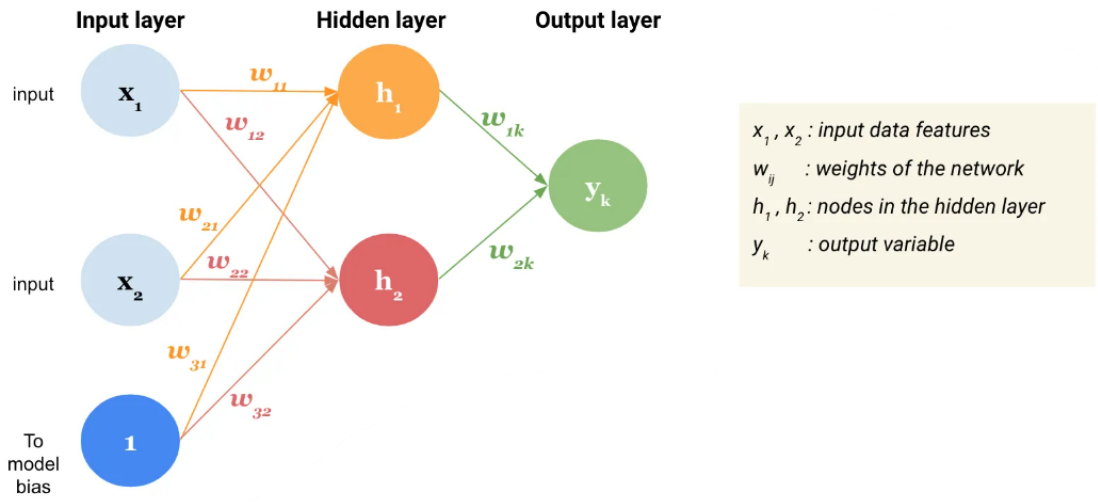

Figure 3.9: Layers of Multilayer Perceptron

3.4.6 Random Forest

Random Forest is a member of the supervised learning method. It may be applied to ML problems involving both classification and regression. The basis for it is the idea of ensemble learning, which merges several classifiers to solve a complicated problem and enhance the model's functionality. Several decision trees are grown using Random Forest and combined to provide a more accurate prediction. The Random Forest model's rationale is based on the idea that several uncorrelated models, independent decision

trees, perform significantly better when combined than when used alone. Each tree provides a 'vote' when employing a Random Forest for classification. The classification that receives the most 'votes' is chosen by the forest. When used for regression, the Random Forest algorithm selects the mean of all tree outputs. An overview of the process is shown in Figure 3.10. The important thing to note in this case is that the decision trees that constitute the Random Forest model have little to no association with one another. The majority of the decision trees in the group will be accurate, even while some may make mistakes, which will move the final result in the proper direction. This method also helps to mitigate the risk of overfitting, making Random Forests robust and reliable. Furthermore, feature importance can be evaluated using this model, offering insights into which variables most significantly impact the prediction.

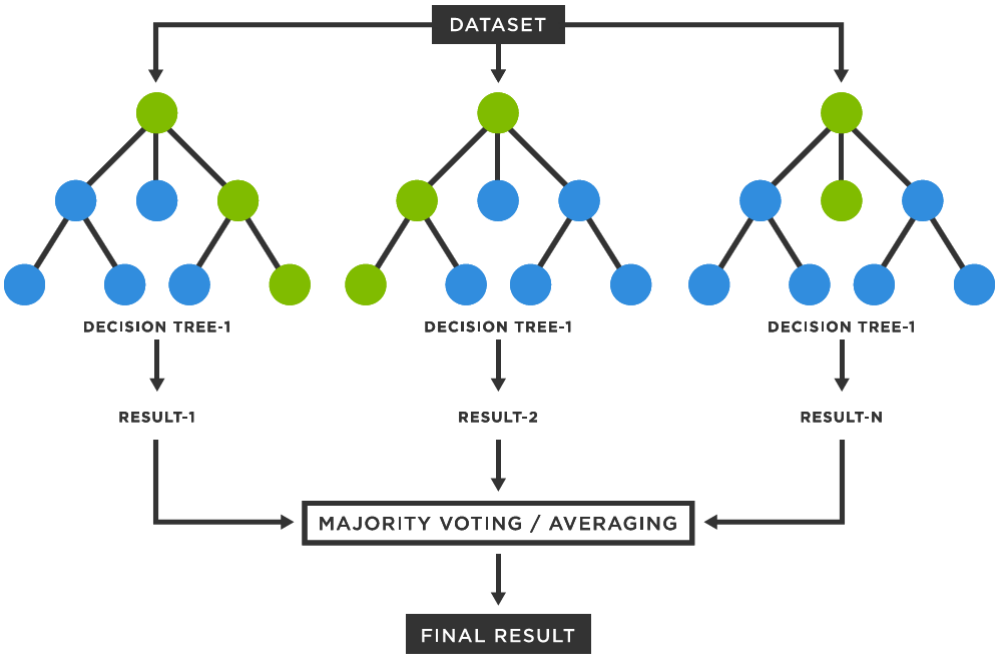

Figure 3.10: Working of Random Forest classifier

3.5 IMPLEMENTATION

We started the implementation phase after preprocessing the dataset and selecting classification models. This section describes attaining a reliable model with improved predictive capability for early liver disease detection.

3.5.1 Feature Selection

Feature selection is a very important step that comes right before the training. Not all features necessarily add value to the model; indeed, multiple features could convey redundant information, which creates a bias in model outcomes.
As such, we implemented a feature selection methodology to remove non-essential features from the dataset. We used the Pearson correlation coefficient and removed features with a correlation coefficient of at least 0.85 with other features.

3.5.2 Training and Testing Set

Our whole dataset is used for training and testing using 10-fold cross-validation techniques. That means all samples are used as training and testing samples, but not simultaneously.

3.5.3 10-Fold Cross-Validation

In this study, 10-fold cross-validation was employed to evaluate the performance of the developed predictive model for detecting liver disease at an early stage. Cross-validation is a popularly accepted method for valuing performance for machine learning and statistical models, mainly used when the dataset size is small, or the risk of overfitting is high.

The procedure of 10-fold cross-validation splits the dataset into ten approximately equally sized subsets or folds. Model training is then done on nine of these folds, with the remaining fold used for validation. After splitting, the training and testing set is scaled separately, and then oversampling is performed on the training set. This training set and testing set is then used in classification models. This is repeated ten times such that each fold is the validation set exactly once. The performance metrics from each iteration are averaged to give a general assessment of predictive capability. 10-fold cross-validation provides a robust evaluation method for machine learning models, helping to mitigate issues such as overfitting and ensuring reliable performance estimation across different subsets of the data. The 10-Fold validation process is illustrated in Figure 3.11.

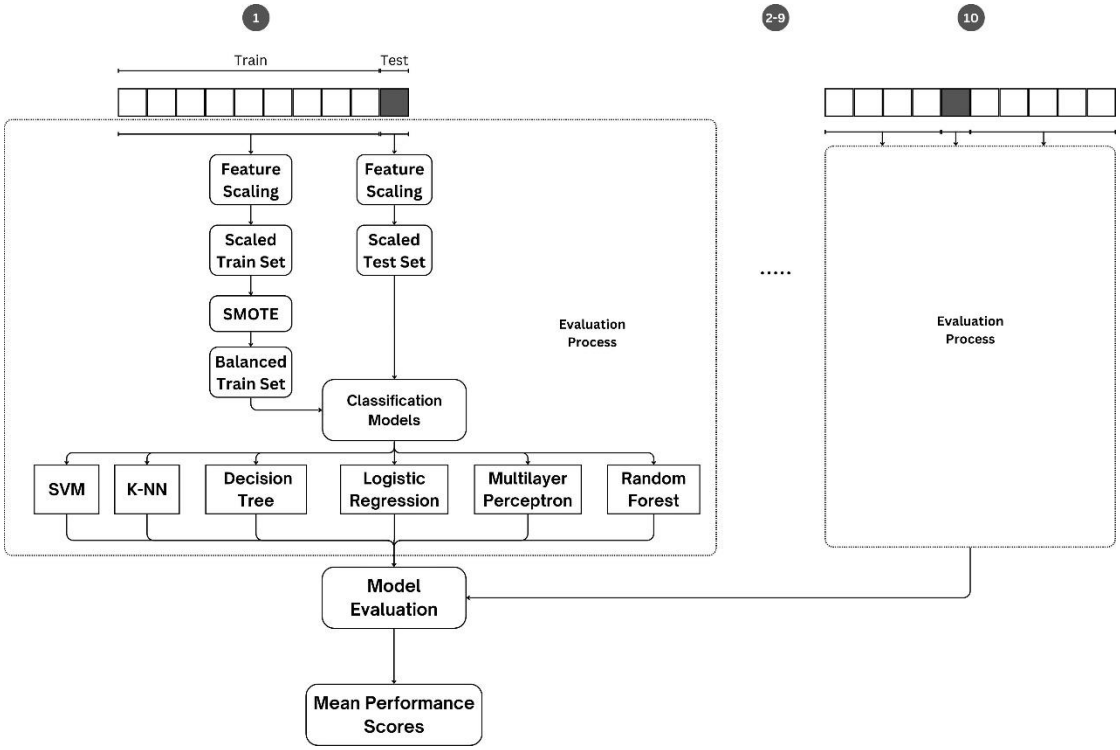

Figure 3.11: 10-Fold Cross-Validation

3.6 MODEL EVALUATION

This section defines the metrics used to evaluate the performance of the developed model for predicting early liver disease. Accurate model performance evaluation is important to define its effectiveness in clinical applications and decision-making. Therefore, a set of evaluation metrics was used to better understand the different aspects of predictive capacity.

3.6.1 Confusion Matrix

A confusion matrix is a performance measurement matrix for a classification problem. It summarizes the model's predictive accuracy concisely by tabulating the actual and predicted classes for a dataset. The confusion matrix is typically organized into a grid format, with rows representing the actual class labels and columns representing the predicted class labels. A confusion matrix is shown in Figure 3.11. Confusion matrix introduces some properties:

i. **True Positives (TP):** Instances where the model correctly predicts the positive class.

ii. **False Positives (FP):** Instances where the model incorrectly predicts the positive class.

iii. **True Negatives (TN):** Instances where the model correctly predicts the negative class.

iv. **False Negatives (FN):** Instances where the model incorrectly predicts the negative class

		Predicted	
		Negative **(N)** -	Positive **(P)** +
Actual	Negative -	True Negative **(TN)**	**False Positive (FP) Type I Error**
	Positive +	**False Negative (FN) Type II Error**	True Positive **(TP)**

Figure 3.12: Confusion Matrix

v. Positive Predictive Value / Precision

The ratio of true positives to all positive predictions is known as Positive Predictive Value or Precision.

$$PPV = \frac{TP}{PP} \tag{3.6}$$

$$pp = Predicted\ Positive$$

vi. False Omission Rate

The ratio of false negatives to all negative predictions is known as False Omission Rate.

$$FOR = \frac{FN}{PN} \tag{3.7}$$

$$PN = Predicted\ Negative$$

vii. False Discovery Rate

The false positive to all positive predictions ratio is known as the False Discovery Rate.

$$FDR = \frac{FP}{PP} = 1 - PPV \tag{3.8}$$

viii. Negative Predictive Value

The ratio of true negatives to all negative predictions is known as Negative Predictive Value.

$$NPV = \frac{TN}{PN} = 1 - FOR \tag{3.9}$$

3.6.2 Accuracy, Recall, F1-Score, and Area Under the Receiver Operating Characteristic Curve

i. Accuracy

It is the result of dividing the total number of accurate predictions produced by the model by the total number of predictions made by the model, including all wrong projections.

$$Accuracy = \frac{number\ of\ \ correct\ prediction}{Total\ Number\ of\ predictions} \tag{3.10}$$

ii. Recall or True Positive Rate

The ratio of predicted positive cases, or true positives, to the number of actual positive samples is known as recall.

$$Recall = \frac{number\ of\ true\ positives}{Total\ number\ of\ positive\ samples} \tag{3.11}$$

iii. F1-Score

The F1 score offers a fair assessment of a model's effectiveness in binary classification tasks by considering false positives and false negatives. It is calculated as the harmonic mean of precision and recall. Because it provides a single parameter to evaluate a model's capacity to achieve high recall and high precision simultaneously, it is

especially helpful in situations when there is an imbalance in the costs associated with false positives and false negatives.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.12}$$

iv. Area Under the Receiver Operating Characteristic Curve

The AUC-ROC curve, depicting the area under the receiver operating characteristic curve, visually represents how well a binary classification model performs across various classification thresholds. Widely utilized in machine learning, it assesses the model's ability to distinguish between two classes, typically the positive and negative classes.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 INTRODUCTION

This section demonstrates the results of using various classifiers (K Nearest Neighbors, Support Vector Machine, Decision Tree, Logistic Regression, Random Forest, and Multilayer Perceptron) on our datasets acquired from hospitals across Bangladesh. Through an extensive analysis of classification measures, we explore each classifier's performance in this study. This includes AUC-ROC (area under the receiver operating characteristic curve) and other metrics like recall, accuracy, precision, and f1-score. To determine the most effective classifier for the precise prediction of liver disease, we will compare our findings with previous studies on the subject. Additionally, we will conduct a detailed examination of each model's computational efficiency and scalability. This will provide a comprehensive understanding of the trade-offs between accuracy and performance. Finally, we will discuss potential reasons for discrepancies between our results and those of earlier research.

The subsequent sections will discuss the detailed results of applying various classifiers to predict liver disease. After discussing the feature selection in Section 4.2, we will focus on the performance of a specific classifier, including Support Vector Machine in Section 4.3, K-Nearest Neighbors in Section 4.4, Decision Tree in Section 4.5, Logistic Regression in Section 4.6, Multilayer Perceptron in Section 4.7, and Random Forest in Section 4.8. Following analyzing individual classifiers, we will test their performance on separate datasets in Section 4.9 and conduct a comparative analysis to determine the most effective model in Section 4.10. Additionally, we will explore sensitivity analysis to evaluate the robustness of our findings in Section 4.11. Finally, we will discuss the

results in the context of previous related work in Section 4.12, providing insights into the broader implications of our research findings and potential avenues for future investigation.

4.2 FEATURE SELECTION

We employed the feature selection method using the Pearson Correlation coefficient. We have chosen a threshold of 0.85 in this method. The model will only keep one feature if two are highly correlated; if their correlation exceeds that threshold value, that is 0.85. That means those two features provide almost the same information. So, we will drop one feature and take the other. This way, redundancy is removed from the input, and only the feature set is reduced, reducing the computational cost. The correlation matrix we have generated using the Pearson correlation coefficient for all the features is shown in Figure 4.1.

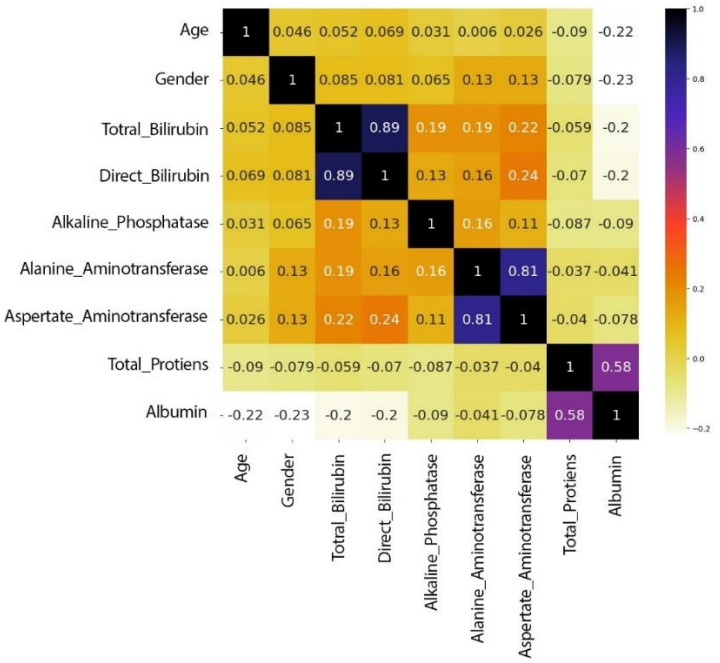

Figure 4.1: Correlation matrix of features

As the correlation matrix shows, direct bilirubin and total bilirubin are highly correlated and exceed our threshold. Therefore, we removed the Direct Bilirubin features from the dataset.

4.3 TRAINING AND TESTING

As we are using 10-fold cross-validation, all of the samples were used for training and testing. Following section described the performance of each classification models after training and testing.

4.3.1 Effect of Feature Scaling

Feature Scaling is done when the dataset is split in the train and test set while in the 10-fold cross-validation process. As we used Standardization for feature scaling, it made the mean of each feature ‘0’ and standard deviation to ‘1’.

	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alanine_Aminotransferase	Aspartate_Aminotransferase	Total_Protiens	Albumin
count	278.0	278.0	278.0	278.0	278.0	278.0	278.0	278.0	278.0
mean	43.5	0.7	2.4	1.2	290.4	51.9	54.7	6.5	3.4
std	15.7	0.4	4.2	2.3	267.5	72.0	76.2	1.1	0.8
min	4.0	0.0	0.0	0.1	68.0	8.0	7.0	2.6	0.8
25%	33.0	0.2	0.8	0.3	175.0	24.0	21.0	5.9	3.1
50%	42.0	1.0	0.9	0.4	209.0	32.0	35.0	6.5	3.4
75%	54.8	1.0	1.9	1.1	299.5	54.8	55.0	7.2	4.0
max	82.0	1.0	30.3	16.2	2410.5	945.0	1020.0	9.6	5.3

Figure 4.2: Before Scaling

	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alanine_Aminotransferase	Aspartate_Aminotransferase	Total_Protiens	Albumin
count	278.0	278.0	278.0	278.0	278.0	278.0	278.0	278.0	278.0
mean	0.0	0.0	0.0	-0.0	-0.0	-0.0	-0.0	0.0	0.0
std	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
min	-2.5	-1.7	-0.6	-0.5	-0.8	-0.6	-0.6	-3.6	-3.3
25%	-0.7	-1.1	-0.4	-0.4	-0.4	-0.4	-0.4	-0.5	-0.4
50%	-0.1	0.6	-0.3	-0.3	-0.3	-0.3	-0.3	0.0	-0.0
75%	0.7	0.6	-0.1	-0.0	0.0	0.0	0.0	0.7	0.7
max	2.5	0.6	6.6	6.6	7.9	12.4	12.7	2.9	2.4

Figure 4.3: After Scaling

As the Figure 4.2 and 4.3 shows, after scaling mean of all the features became ‘0’ and standard deviation became ‘1’. This process ensures that all features contribute equally to the model, preventing features with larger ranges from dominating those with smaller ranges, thus improving the model's performance and convergence during training.

Standardization does not change the distribution. It rescales the feature, as shown in Figure 4.4.

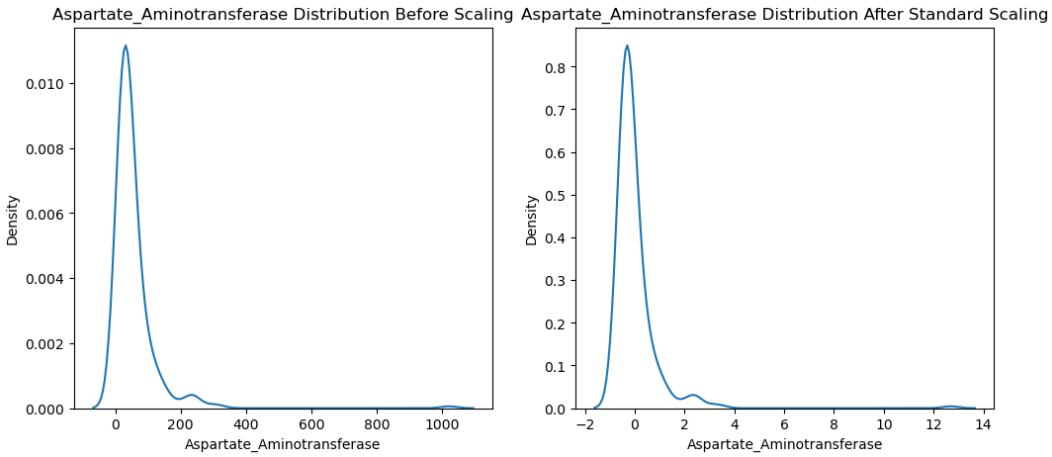

Figure 4.4: Verifying distribution before and after scaling

4.3.2 Support Vector Machine

We used 10-fold cross-validation to assess the Support Vector Machine's ability to predict liver disease. Using this approach, the dataset is randomly divided into ten equal-sized subsets. Then, nine of the subsets are used for training, and one is chosen to be used as the testing subset. This process is repeated ten times, with each subset serving as the testing data exactly once. The SVM model obtained a mean accuracy of 70.09% with a True Positive Rate of 67% and a True Negative Rate of 78.21%. We identified 17 False Positives and 66 False Negatives from the resulting confusion matrix, as shown in Figure 4.5. The true positive rate and true negative rate are calculated using the equation 3.2 and 3.4.

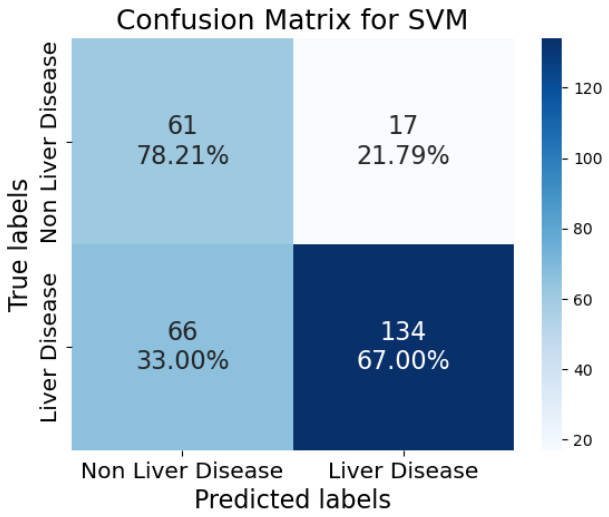

Figure 4.5: Confusion matrix of Support Vector Machine Classifier

$$\boldsymbol{True\ Positive\ Rate} = \frac{TP}{TP + FN} \times 100 = \frac{134}{134 + 66} \times 100 = 67\%$$

$$\boldsymbol{True\ Negative\ Rate} = \frac{TN}{TN + FP} \times 100 = \frac{61}{61 + 17} \times 100 = 78.21\%$$

Table 4.1: Accuracy Result of Support Vector Machine classifier

Correctly Classified Instances	195	70%
Incorrectly Classified Instances	83	30%
Total	278	100%

We have a higher Predicted Positive Value of 88.74%, which indicates a strong ability to accurately identify positive instances, with only 11.26% of positive predictions turning out to be false. However, the relatively high False Omission Rate of 51.97% is worth noting, indicating a significant proportion of false negatives. This may indicate that the classifier is having difficulty correctly identifying the negatives. The Predicted Positive Value, False Discovery Rate, Negative Predictive Value and False Omission Rates are calculated below:

$$\boldsymbol{PPV}(\%) = \frac{TP}{PP} \times 100 = \frac{134}{151} \times 100 = \ 88.74\%$$

$$\boldsymbol{FDR}(\%) = 1 - PPV = 11.26\%$$

$$\boldsymbol{NPV}(\%) = \frac{TN}{PN} \times 100 = \frac{61}{127} \times 100 = 48.03\%$$

$$\boldsymbol{FOR} = 1 - NPV = 51.97\%$$

$PP = Predicted\ Positive$

$PN = Predicted\ Negative$

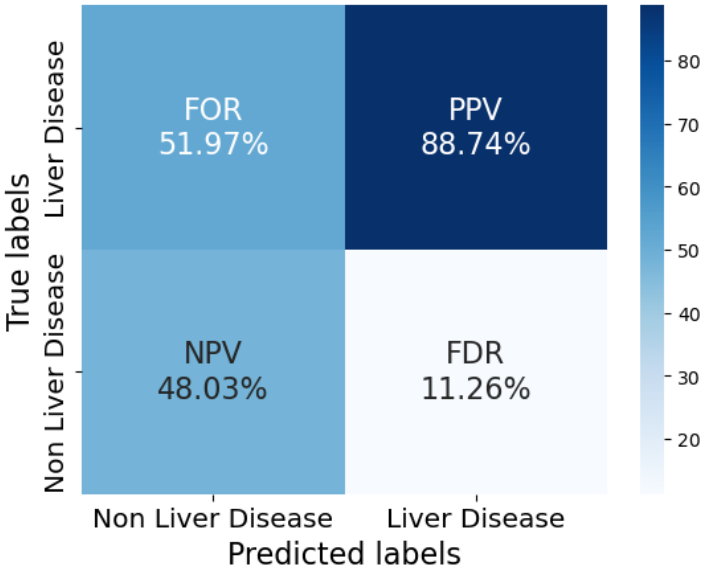

Figure 4.6: PPV/FDR/FOR/NPV by Support Vector Machine

Table 4.2: Confusion matrix for Support Vector Machine

	Original Number	True Prediction	False Prediction
Liver Disease	200	134	66
Non-Liver Disease	78	61	17
Total	278	195	83

The following curve shown in the figure is for the Receiver Operating Characteristic curve for measuring performance.

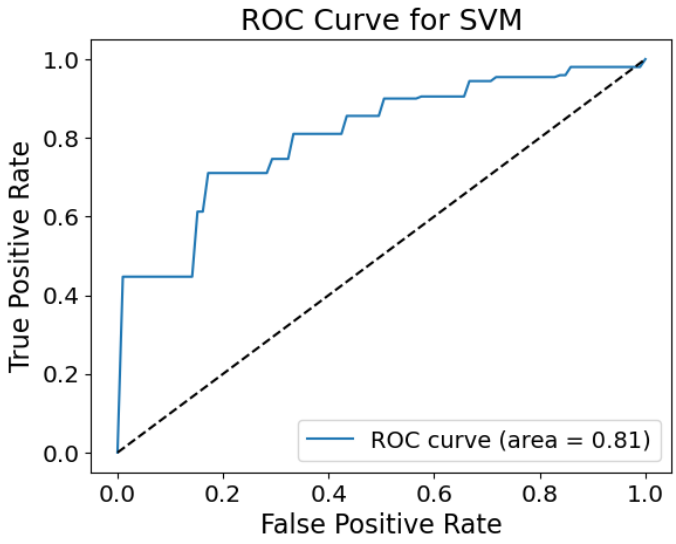

Figure 4.7: Receiver Operating Characteristic curve of the Support Vector Machine

The classifier's Receiver Operating Characteristic curve exhibits an area under the curve score of 0.81, which means the SVM is performing reasonably well.

4.3.3 K-Nearest Neighbors

We used 10-fold cross-validation to assess the K-Nearest Neighbors algorithm's predictive ability in diagnosing liver disease. With a True Positive Rate of 63% and a True Negative Rate of 79.49%, the model achieved an average accuracy of 67.6%. As shown in Figure 4.8, the confusion matrix analysis revealed 16 False Positives and 74 False Negatives.

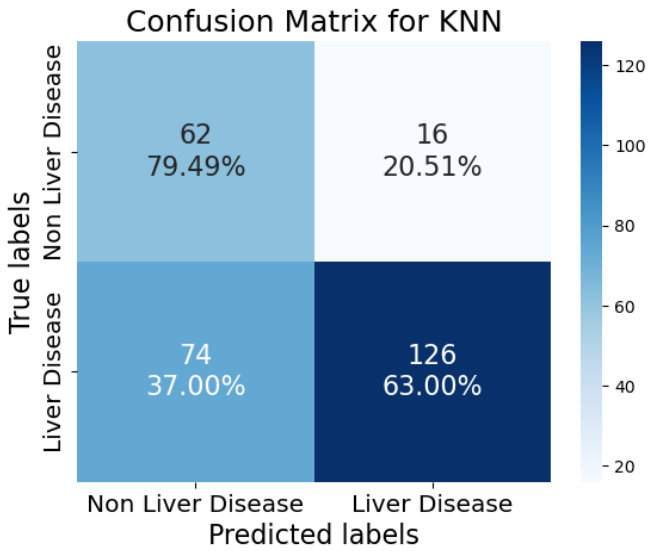

Figure 4.8: Confusion matrix of K-Nearest Neighbors Classifier

Table 4.3: Accuracy Result of K-Nearest Neighbors classifier

Correctly Classified Instances	188	67.6%
Incorrectly Classified Instances	90	30.4%
Total	278	100%

With a noteworthy Positive Predictive Value of 88.73%, we demonstrated excellent performance in properly predicting positive values. However, the 54.41% False Omission Rate indicates difficulties detecting negative values precisely.

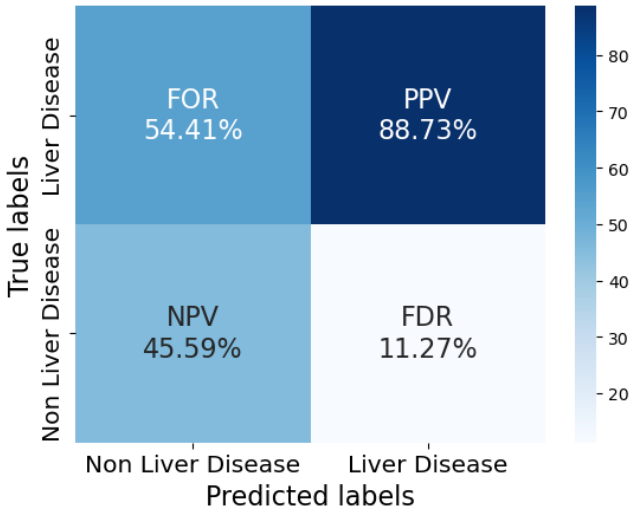

Figure 4.9: PPV/FDR/FOR/NPV by K-Nearest Neighbors

Table 4.4: Confusion matrix for K-Nearest Neighbors

	Original Number	**True Prediction**	**False Prediction**
Liver Disease	200	126	74
Non-Liver Disease	78	62	16
Total	278	188	90

The predicted and actual values are displayed in the above table. In this case, the predicted value is shown in the top row, while the actual values are in the left column.

The following curve shown in the figure is for the ROC curve for measuring performance.

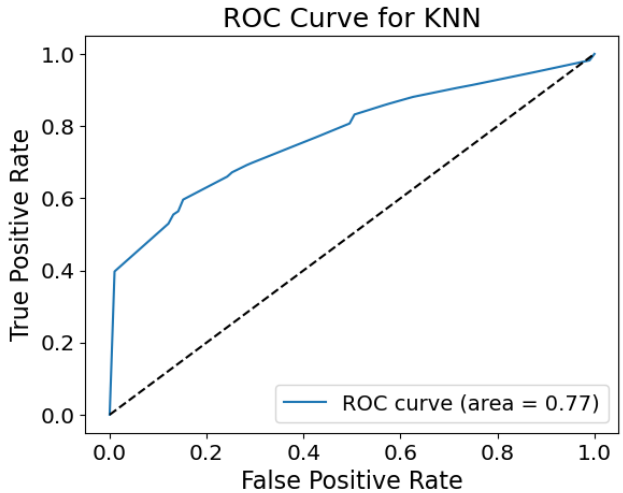

Figure 4.10: Receiver Operating Characteristic curve of K-Nearest Neighbors

The classifier's Receiver Operating Characteristic curve exhibits an area under the curve score of 0.77, which is less than the score we achieved from SVM.

4.3.4 Decision Tree

We assessed the predictive capability of the Decision Tree algorithm for liver disease diagnosis using 10-fold cross-validation. This involved partitioning the dataset randomly into ten subsets, utilizing nine for training and one for testing in each iteration. We obtained a robust evaluation of the model's performance by averaging the results over these iterations. The Decision Tree yielded an accuracy of 71.2%, with a True Positive Rate of 74.50% and a True Negative Rate of 62.82%. Notably, our analysis identified 29 False Positives and 51 False Negatives, as evidenced in the confusion matrix shown in Figure 4.11.

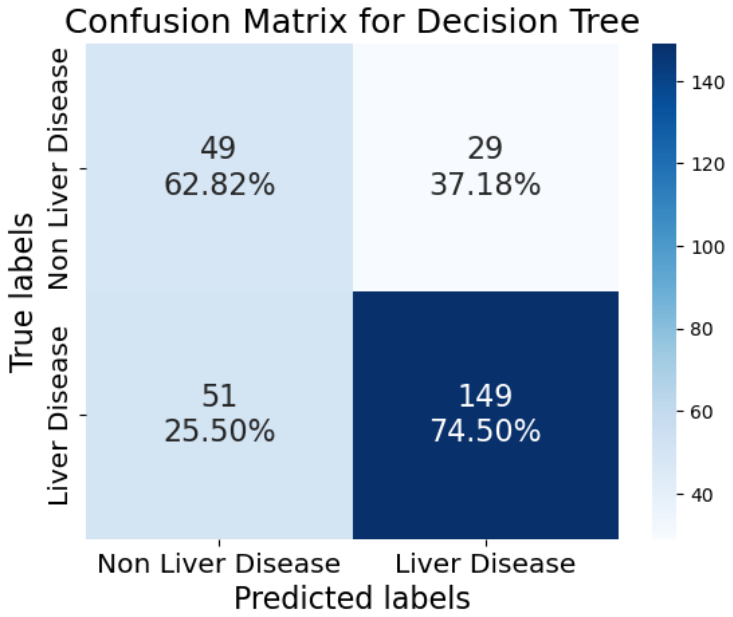

Figure 4.11: Confusion matrix of Decision Tree Classifier

Table 4.5: Accuracy Result of Decision Tree classifier

Correctly Classified Instances	198	71.2%
Incorrectly Classified Instances	80	28.8%
Total	278	100%

Using the Decision Tree classifier, we calculated a Positive Predictive Value of 83.71% and a False Omission Rate of 51%, as shown in Figure 4.12. This score indicates that the classifier outperforms K-Nearest Neighbors and Support Vector Machine classifiers regarding negative value prediction.

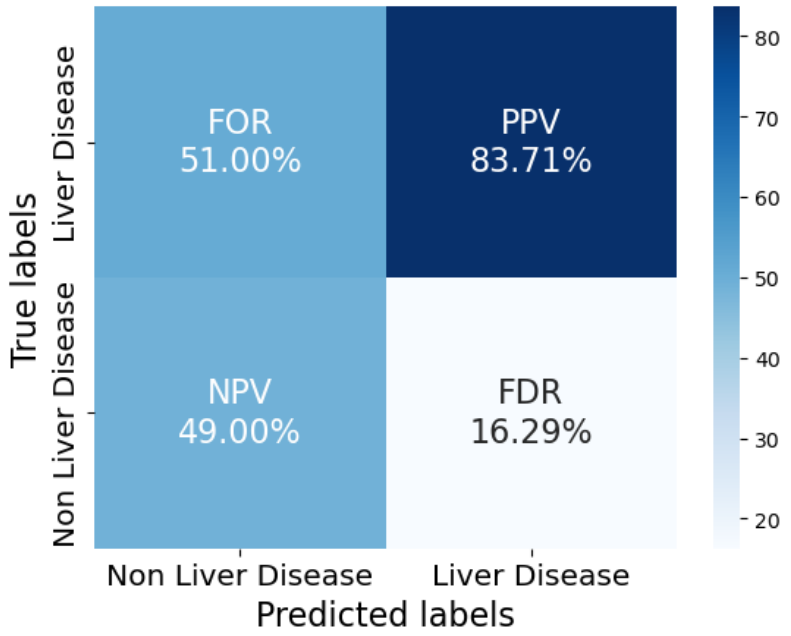

Figure 4.12: PPV/FDR/FOR/NPV by Decision Tree

Table 4.6: Confusion Matrix for Decision Tree

	Original Number	True Prediction	False Prediction
Liver Disease	200	149	51
Non-Liver Disease	78	49	29
Total	278	198	80

The following curve in the figure is for the Receiver Operating Characteristic curve for measuring performance.

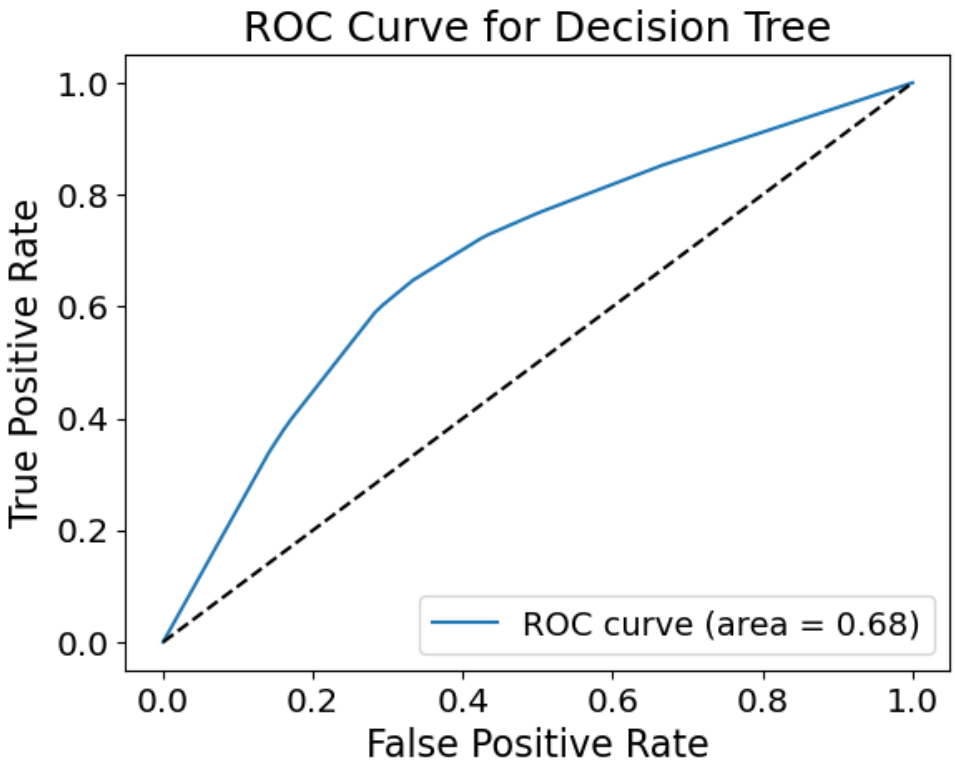

Figure 4.13: Receiver Operating Characteristic curve of Decision Tree

The Receiver Operating Characteristic curve for the classifier exhibits an area under the curve score of 0.68. This suggests that even while the classifier shows some degree of discriminative capacity, it might not be able to distinguish between the two classes, as well as SVM or KNN.

4.3.5 Logistic Regression

In our study, we employed Logistic Regression, applying the 10-fold cross-validation method for reliable assessment. The Logistic Regression model achieved an accuracy of 69.42%. Additional analysis demonstrated its ability to accurately detect both positive and negative instances, which yielded a True Positive Rate of 66.50% and a True Negative Rate of 76.92%. However, our evaluation identified 67 False Negatives and 18 False Positives, indicating instances where the model could improve. These conclusions are drawn from the confusion matrix shown in Figure 4.14.

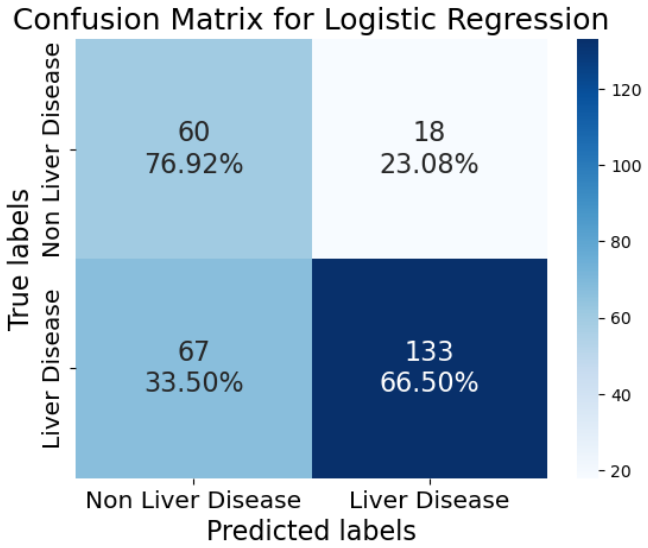

Figure 4.14: Confusion matrix of Logistic Regression classifier

Table 4.7: Accuracy Result of Logistic Regression classifier

Correctly Classified Instances	193	69.42%
Incorrectly Classified Instances	85	30.58%
Total	278	100%

With the Logistic Regression classifier, we have achieved a Positive Predictive Value of 88.08% and a False Omission Rate of 52.76%. These percentages show that the classifier performs almost as well as the SVM classifier.

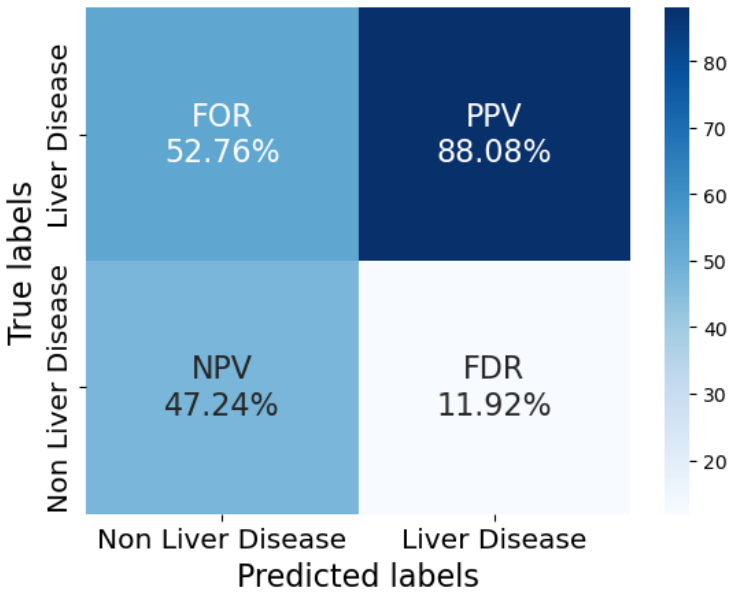

Figure 4.15: PPV/FDR/FOR/NPV by Logistic Regression

Table 4.8: Confusion Matrix for Logistic Regression

	Original Number	True Prediction	False Prediction
Liver Disease	200	133	67
Non-Liver Disease	78	60	18
Total	278	193	85

The predicted and actual values are displayed in the above table. In this case, the predicted value is shown in the top row, while the actual values are in the left column.

The following curve shown in the figure is for the Receiver Operating Characteristic curve for measuring performance.

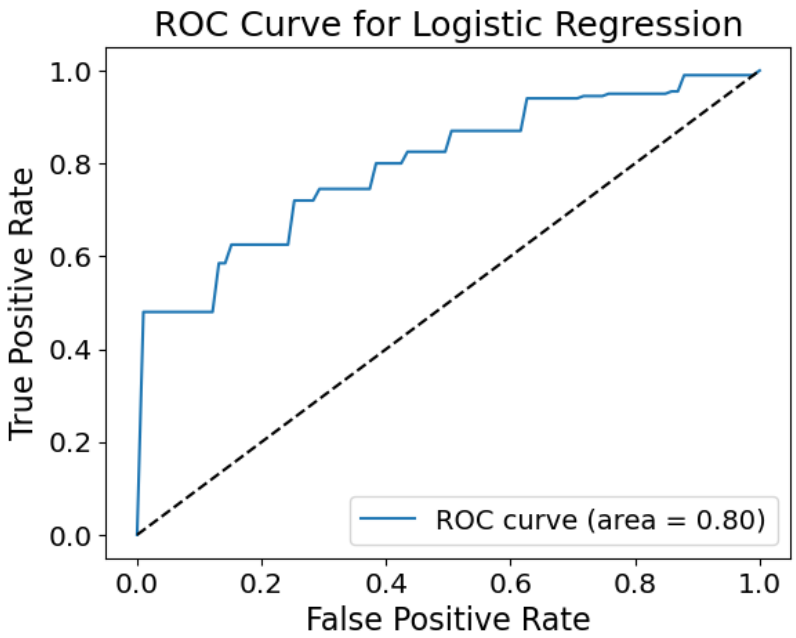

Figure 4.16: Receiver Operating Characteristic curve of Logistic Regression

The Receiver Operating Characteristic curve for this classifier returns an AUC score of 0.80, which further indicates that the classifier has relatively good discriminative ability; the higher the Area Under Curve score, the better the model can distinguish between positive and negative instances.

4.3.6 Multilayer Perceptron

Evaluating the Multilayer Perceptron algorithm's performance for liver disease involved applying a 10-fold cross-validation methodology. A comprehensive evaluation of the predicted accuracy was obtained by combining the results from all of the fold iterations. Along with a 77% True Positive Rate and a 57.69% True Negative Rate, the Multilayer Perceptron showed an accuracy of 71.58%. Additionally, the confusion matrix shown in Figure 4.17 indicates that the study produced 33 False Positives and 46 False Negatives.

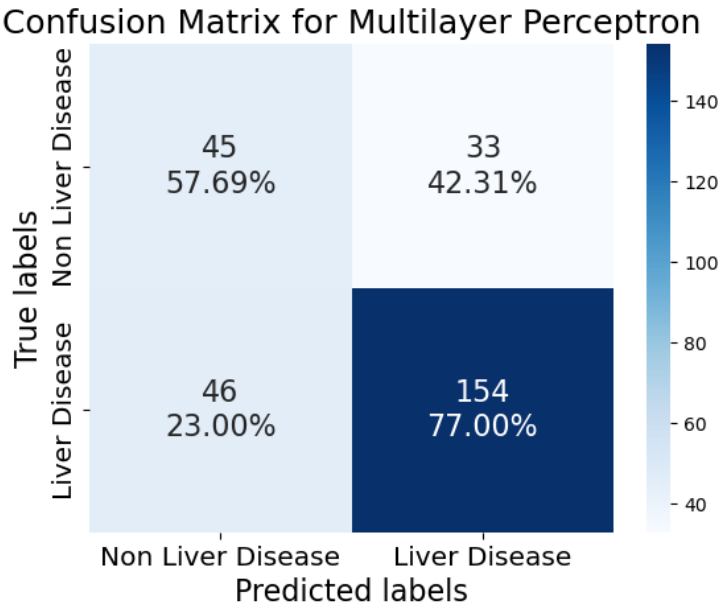

Figure 4.17: Confusion matrix of Multilayer Perceptron

Table 4.9: Accuracy Result of Multilayer Perceptron classifier

Correctly Classified Instances	199	71.58%
Incorrectly Classified Instances	79	28.42%
Total	278	100%

We have a lower False Omission Rate (50.55%) and a greater Positive Predictive Value (82.35%) with MLP as shown in Figure 4.18. This indicates that, on the negative values for this dataset, the Multilayer Perceptron performed much better than the other classifiers considered so far.

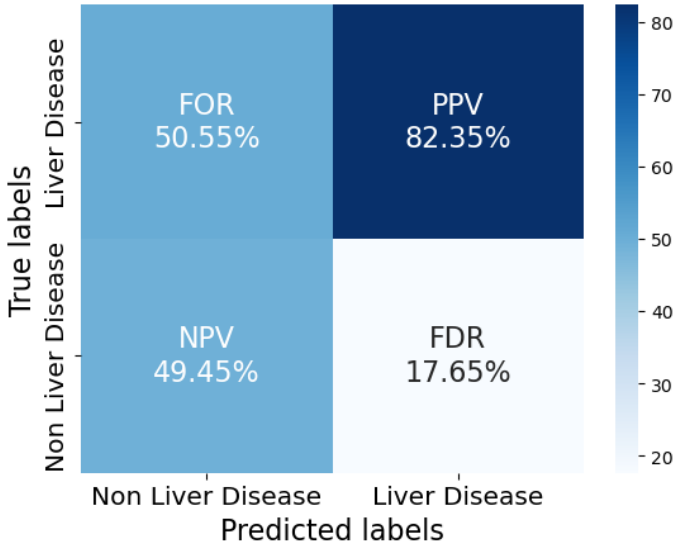

Figure 4.18: PPV/FDR/FOR/NPV by Multilayer Perceptron

Table 4.10: Confusion matrix for Multilayer Perceptron

	Original Number	True Prediction	False Prediction
Liver Disease	200	154	46
Non-Liver Disease	78	45	33
Total	278	199	79

The following curve shown in the figure is for the Receiver Operating Characteristic curve for measuring performance.

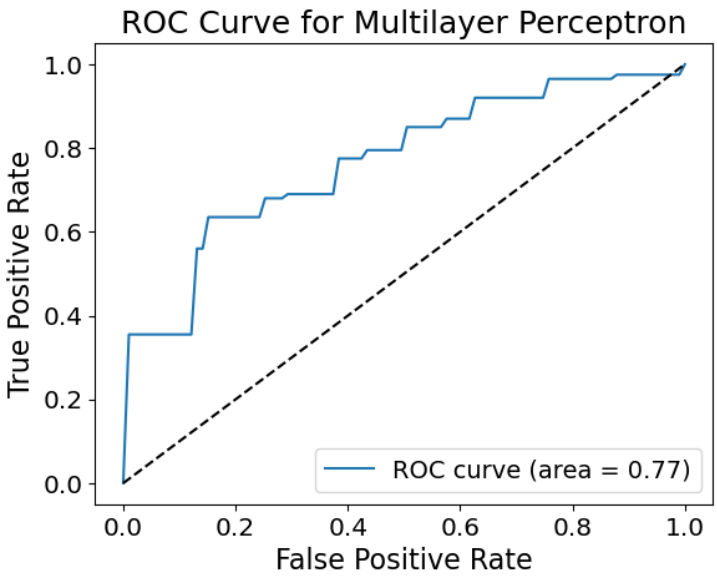

Figure 4.19: Receiver Operating Characteristic curve of Multilayer Perceptron

The classifier's Receiver Operating Characteristic curve yields an AUC value of 0.77, indicating its excellent discriminative capacity. This demonstrates that the deterministic power of both MLP and KNN is the same for positive and negative values.

4.3.7 Random Forest

Using 10-fold cross-validation, we achieved 80.5% accuracy with the classifier, the highest accuracy we have with all six classifiers used in this work. Here, the True Positive and True Negative rates are notably higher at 85% and 69.23%, respectively. The confusion matrix (shown in Figure 4.20) generated shows that there are 24 False Positives and 30 False Negatives.

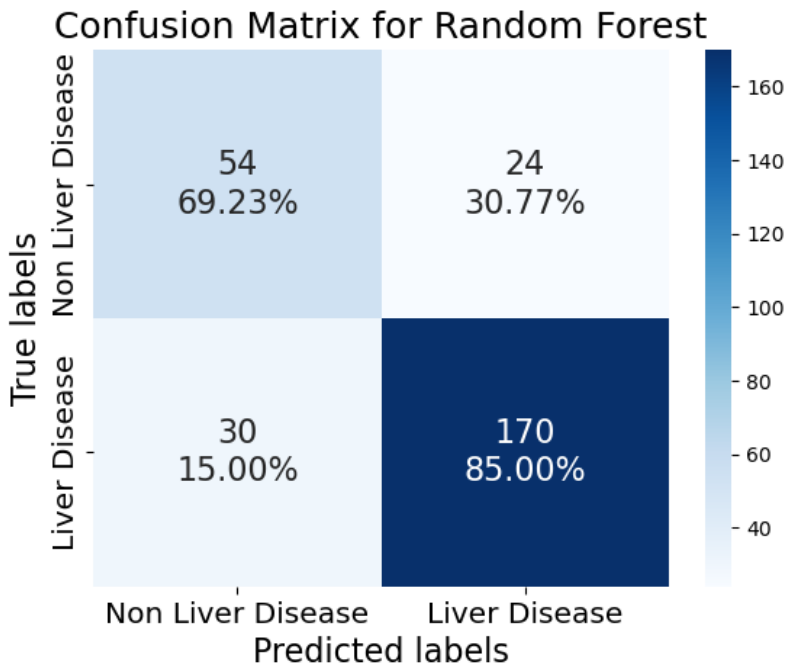

Figure 4.20: Confusion matrix of Random Forest

Table 4.11: Accuracy Result of Random Forest

Correctly Classified Instances	224	80.57%
Incorrectly Classified Instances	54	19.42%
Total	278	100%

We have the lowest False Omission Rate (35.71%) and greater Positive Predictive Value (87.63%) with Multilayer Perceptron, as shown in Figure 4.21. This indicates that the Random Forest classifier performs much better on the negative and positive values for this dataset than the other classifiers considered so far.

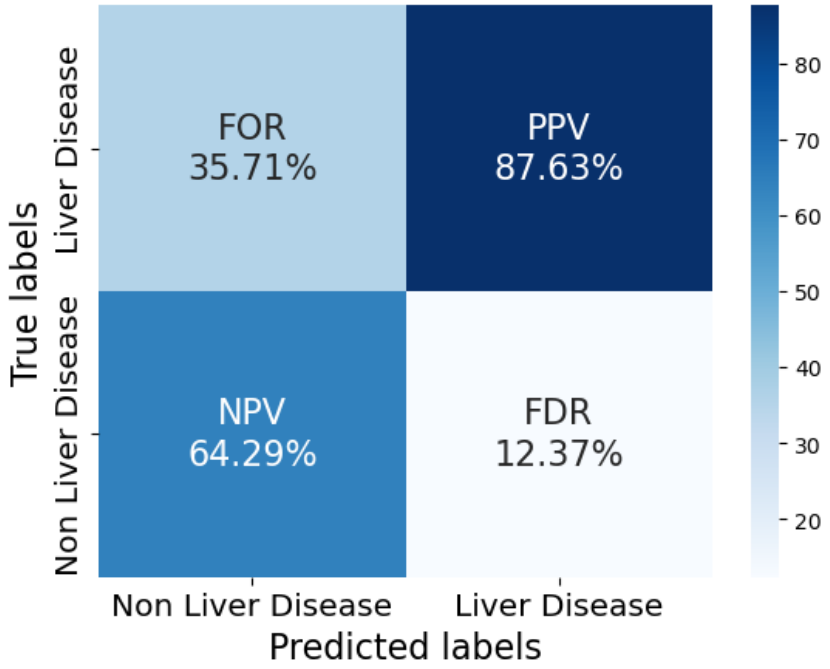

Figure 4.21: PPV/FDR/FOR/NPV by Random Forest

Table 4.12: Confusion matrix for Random Forest

	Original Number	True Prediction	False Prediction
Liver Disease	200	170	30
Non-Liver Disease	78	54	24
Total	278	224	54

The predicted and actual values are displayed in the above table. In this case, the predicted value is shown in the top row, while the actual values are in the left column.

The following curve shown in the figure is for the ROC curve for measuring performance.

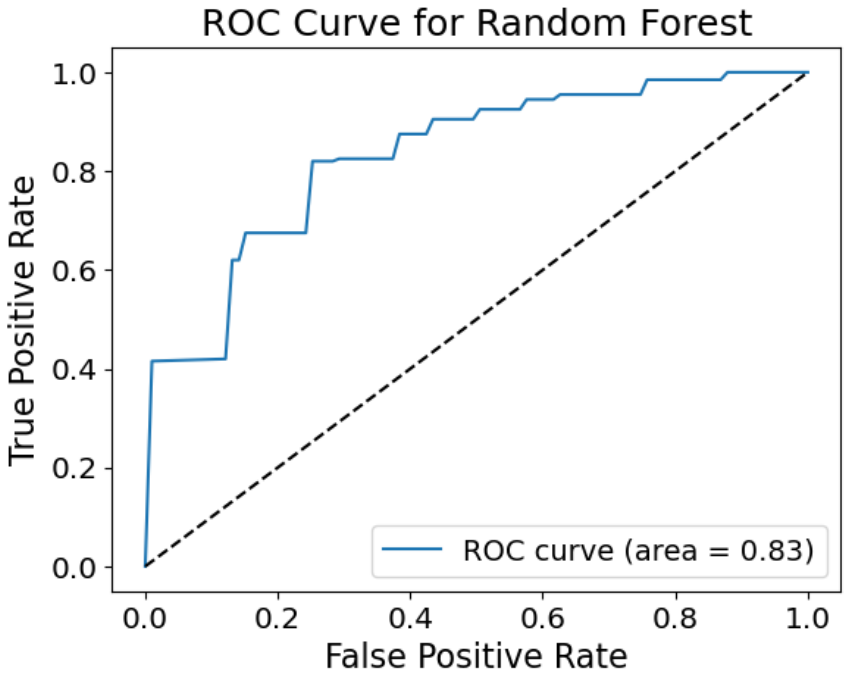

Figure 4.22: ROC curve for measuring the performance of Random Forest

Out of the six classifiers, Random Forest has the highest AUC score, 0.83. This indicates that it performs far better than the other classifiers mentioned earlier in identifying both positive and negative values. The superior performance of the Random Forest classifier can be attributed to its ensemble nature, which combines the predictions of multiple decision trees to improve accuracy and reduce overfitting. Furthermore, this high AUC score demonstrates its robustness and reliability in handling the variability within the dataset. Consequently, Random Forest stands out as the most effective model among the tested classifiers, making it a valuable tool for our predictive tasks.

4.4 TESTING WITH SEPARATE DATA

We have done a thorough performance analysis of the various classification models. Now, we will test the outcome using the classifiers on a set of test data that was previously unseen to the classifiers. To begin with, we would assign fifteen samples for the test set. Next, we would predict this test set using the trained classification model. After the prediction process, we would store the test inputs, actual outputs, and predicted outputs for each classifier in an Excel file.

SL	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alanine_Aminotransferase	Aspartate_Aminotransferase	Total_Protiens	Albumin	Actual Output
1	48	1	1.6	0.9	591	78	110	6.8	2.3	1
2	34	1	2.2	1.4	512	42	25	6.6	3.1	1
3	54	0	1.6	0.9	212	39	17	8.1	3.8	0
4	61	1	1.1	0.5	198	28	30	6.1	2.3	1
5	50	1	0.7	0.1	201	23	19	6.9	3.8	0
6	56	1	0.8	0.3	268	13	21	6.5	3.2	1
7	40	0	1.1	0.4	155	17	35	7.3	3.5	1
8	40	0	1.1	0.3	287	31	28	7.6	3.6	1
9	42	0	1	0.3	165	22	30	6.5	3.4	0
10	31	1	0.8	0.3	270	16	55	5.4	2.2	1
11	50	1	1.8	0.7	345	39	56	7.6	3.5	1
12	70	0	0.1	0.3	122	15	31	7.1	3.3	0
13	21	1	18.7	9.2	403	393	499	5.5	4.2	1
14	42	1	0.8	0.3	162	26	21	6.2	3.2	0
15	41	1	0.1	0.3	134	34	15	7.3	3.1	0

Figure 4.23: Data samples for prediction

SL	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alanine_Aminotransferase	Aspartate_Aminotransferase	Total_Protiens	Albumin	Actual Output	SVM	KNN	Decision Tree	Logistic Regression	MLP	Random Forest
1	48	1	1.6	0.9	591	78	110	6.8	2.3	1	1	1	1	1	1	1
2	34	1	2.2	1.4	512	42	25	6.6	3.1	1	1	1	1	1	1	1
3	54	0	1.6	0.9	212	39	17	8.1	3.8	0	0	1	1	0	1	1
4	61	1	1.1	0.5	198	28	30	6.1	2.3	1	1	1	1	1	1	1
5	50	1	0.7	0.1	201	23	19	6.9	3.8	0	0	1	1	0	1	0
6	56	1	0.8	0.3	268	13	21	6.5	3.2	1	1	1	0	1	1	1
7	40	0	1.1	0.4	155	17	35	7.3	3.5	1	0	1	0	0	1	0
8	40	0	1.1	0.3	287	31	28	7.6	3.6	1	0	1	1	0	1	1
9	42	0	1	0.3	165	22	30	6.5	3.4	0	0	1	0	0	0	0
10	31	1	0.8	0.3	270	16	55	5.4	2.2	1	1	1	0	1	1	1
11	50	1	1.8	0.7	345	39	56	7.6	3.5	1	1	1	1	1	1	1
12	70	0	0.1	0.3	122	15	31	7.1	3.3	0	0	0	0	0	0	0
13	21	1	18.7	9.2	403	393	499	5.5	4.2	1	1	1	1	1	1	1
14	42	1	0.8	0.3	162	26	21	6.2	3.2	0	0	0	1	0	0	0
15	41	1	0.1	0.3	134	34	15	7.3	3.1	0	0	1	0	0	1	0

Figure 4.24: Prediction of new samples

Incorrectly Identified

From the Figure 4.24, we can see that SVM, Logistic Regression and Random Forest classifier have predicted thirteen out of fifteen test samples correctly. So, these classifier performed better in case of this test set of fifteen samples.

4.5 COMPARATIVE ANALYSIS OF CLASSIFICATION MODELS

In this section, we aim to do a comparative analysis of the classifiers used to predict liver diseases. By doing this, we want to pinpoint the most reliable model for our

problem. To this effect, we will evaluate each classifier using the different measures of classification.

Table 4.13: Performance metrics for different classification algorithms

Classifier	FP Rate	Precision	F1-Score	Recall	ROC Area	Accuracy	Observations
SVM	0.23	0.89	0.76	0.67	0.81	70.09%	1. The Decision Tree algorithm exhibited a higher true positive rate; however, its elevated false positive rate contributed to a lower AUC score. 2. The Random Forest Classifier demonstrated superior performance across nearly all metrics compared to other classifiers.
K-NN	0.25	0.88	0.73	0.63	0.78	67.58%	
Decision Tree	0.45	0.84	0.79	0.74	0.68	71.16%	
Logistic Regression	0.26	0.88	0.75	0.66	0.80	69.40%	
Multilayer Perceptron	0.39	0.82	0.79	0.77	0.77	71.60%	
Random Forest	0.30	0.88	0.86	0.85	0.83	80.50%	

The SVM and Logistic Regression classifiers are relatively balanced, with decent TPRs and low FPRs, indicating good precision-recall trade-offs. The K-NN classifier is reasonably similar but shows a bit worse Recall. Interestingly, the Decision Tree classifier stands out with a much higher Recall, suggesting that it performs well in correctly identifying the positive cases while keeping the false negatives minimal. On the other hand, its FPR is relatively higher, meaning it has a lower precision and ROC area value. That means while the Decision Tree classifier does very well regarding Recall, its performance across thresholds might not be as strong as some other models. On the other hand, models like Random Forest and Multilayer Perceptron show balanced precision and recall, reflected in their higher ROC Area values.

Table 4.14: Disease and non-disease prediction performance of all classifiers

Classifier	Disease (Actual)	Disease (Predicted)	Accuracy (%)	Non-Disease (Actual)	Non-Disease (Predicted)	Accuracy (%)
SVM	200	134	67.0%	78	61	78.2%
K-NN		126	63.0%		62	79.5%
Decision Tree		149	74.5%		49	62.8%
Logistic Regression		133	66.5%		60	76.9%
Multilayer Perceptron		154	77.0%		45	57.7%
Random Forest		170	85.0%		54	69.2%

The table above compares the accuracy of predicting liver disease and non-disease patients. It shows that random forest obtained the highest accuracy in predicting Disease, and K-NN obtained the highest accuracy in predicting patients with no liver disease.

The following bar chart in Figure 4.25 shows the overall accuracy of every classifier using 10-fold cross-validation.

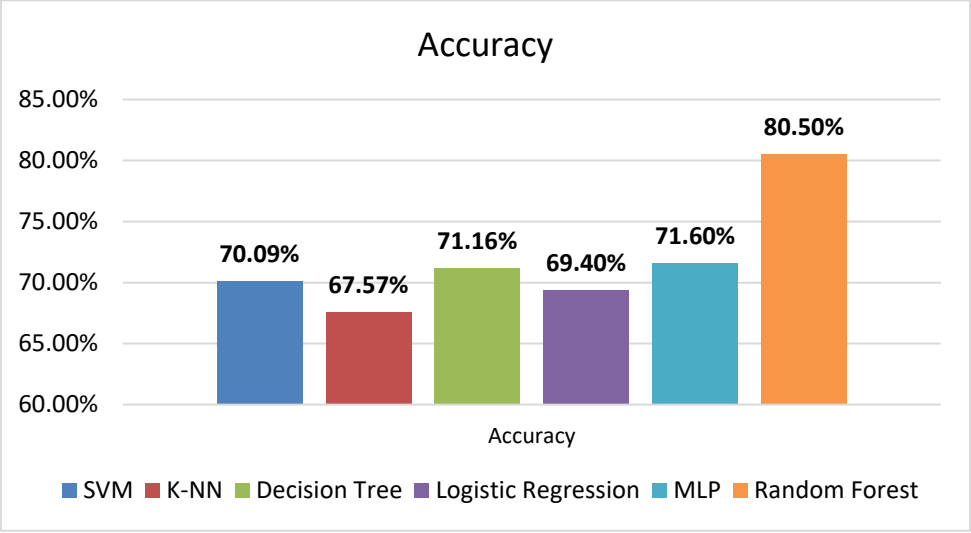

Figure 4.25: An accuracy comparison chart of different classifiers

The accuracy bar chart shows that we achieved the highest accuracy of 80.5% using the

Random Forest classifier.

4.6 SENSITIVITY ANALYSIS

It is possible that not every feature needs to be defined for classification. Which qualities are comparatively more essential than others can be determined by sensitivity analysis. It contributes to reducing feature size, which minimizes computing costs without overall degrading performance. After selecting the top-performing model, we used a Random Forest classifier to train the model on the dataset. The scikit-learn classifier attribute known as "feature_importances_" is then used to determine which feature is more significant.

The feature importance is displayed in a bar chart, as seen below:

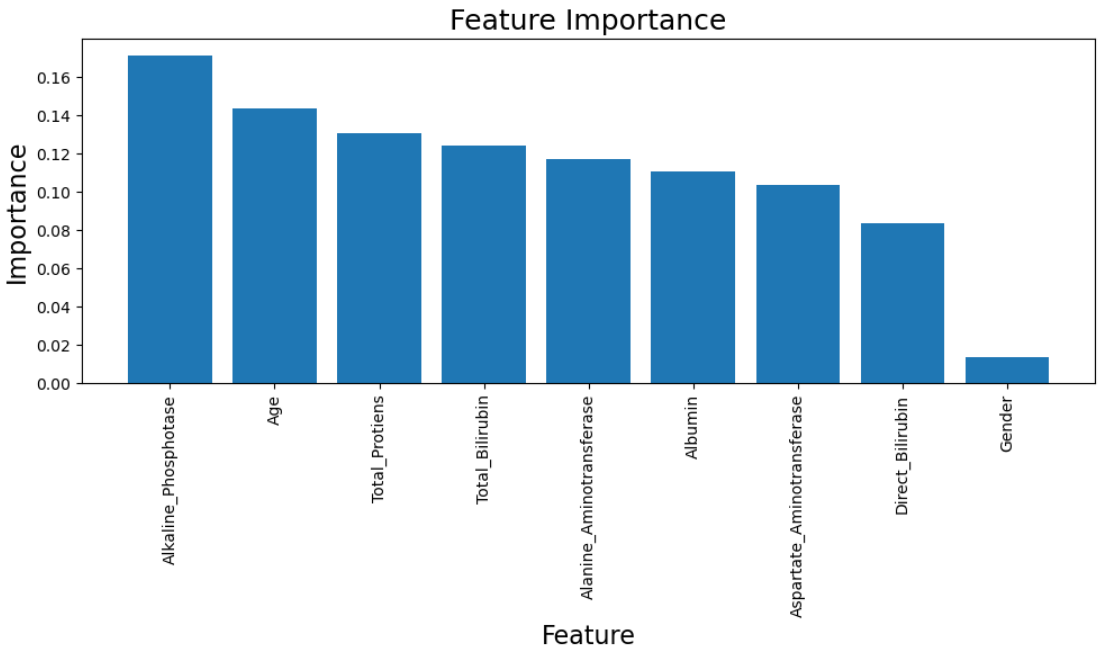

Figure 4.26: Sensitivity analysis chart

The above chart shows that the patient's gender has the lowest importance level relative to other features, and Alkaline Phosphatase has the highest importance. That means the output is more sensitive to the change in the value of Alkaline Phosphatase.

4.7 RESULT DISCUSSION WITH PREVIOUS RELATED WORK

We have included our proposed model alongside previous works on this topic in Table 4.15. Regarding accuracy, our suggested model performed reasonably well. It is essential to note that most of the previous studies have utilized publicly available datasets, such as the Indian Liver Patient Dataset and the BUPA liver disease dataset. Our dataset differs from those utilized in previous studies, and we have fewer samples than two of the datasets employed in prior research. With the application of appropriate data preprocessing techniques and access to larger datasets, we anticipate further improvements in our results.

Table 4.15: Comparison with previous results

Authors	Year	Dataset	Procedure	Classifier	Accuracy
Ramana et al. [15]	2019	ILPD	10-fold cross-validation	Bagging	69.30%
Gulia et al. [24]	2014	ILPD	WEKA	Random Forest	71.87%
J. Singh et al. [2]	2022	BUPA	WEKA	Random Forest	73.00%
Altaf et al. [14]	2022	ILPD	Train-Test	Voting Ensemble	73.56%
Singh et al. [25]	2020	ILPD	10-fold cross-validation	Logistic Regression	74.36%
Geetha et al. [26]	2021	ILPD	Train-Test	SVM	75.04%
Choudhary et al. [27]	2021	ILPD	5-fold cross-validation	Logistic Regression	79.50%
Dritsas et al. [23]	2023	ILPD	10-fold cross-validation	Voting Classifier	80.10%
Proposed Model	**2024**	**Collected**	**10-fold cross-validation**	**Random Forest**	**80.50%**
Kuzhippallil et al. [28]	2020	ILPD	Train-Test	Random Forest	88.00%
R. Amin et al. [21]	2023	ILPD	10-fold cross-validation	Random Forest	88.10%

CHAPTER 5

CONCLUSIONS & FUTURE WORK

Our research provided insights into the liver of disease patients. It evaluated different classification models on early liver disease detection, namely SVM, K-NN, Decision Tree, Logistic Regression, Multilayer Perceptron, and Random Forest. Those classification models are compared using different performance metrics to enhance the outcome of this work. As no single classification model performs optimally across all datasets, our analysis focused on comparing these models across different performance metrics to identify the most reliable approach for liver disease detection. Evaluating each classification shows that SVM achieved 71.90%, K-NN achieved 69%, Decision Tree achieved 71.30%, Logistic Regression 71.70%, Multilayer Perceptron achieved 74.10%, and Random Forest achieved 80.50% accuracy. This work concludes that the Random Forest classifier is the most reliable and best classification model for liver disease prediction. It can be used for early liver disease prediction in healthcare.

This research aims to find the best-performing classification model for liver disease prediction. We can find a best-suited model using different performance metrics. However, our study is constrained by the limited size of the dataset, which may have impacted the performance outcomes. In the future, new classification models can be analyzed on a dataset with more samples. We can also optimize the classification models to improve the results further.

REFERENCES

[1] D. Harshad, A. Sumeet K., A. Juan Pablo, N. Yvonne Ayerki, P. Elisa, and K. Patrick S., "Global Burden of Liver Disease: 2023 Update," *J Hepatol*, vol. 79, no. 2, Mar. 2023, doi: 10.1016/j.jhep.2023.03.017.

[2] Jaswinder Singh and Kirti Kangra, "Prediction and Analysis of Liver Disorder Disease using Machine Learning Algorithms," *International Journal of Computer Science & Communication*, vol. 13, no. 2, pp. 26–32, 2022.

[3] K. R. Makkena and K. Natarajan, "Classification Algorithms for Liver Epidemic Identification," *EAI Endorsed Trans Pervasive Health Technol*, vol. 9, pp. 1–13, May 2023, doi: 10.4108/eetpht.9.4379.

[4] D. Bhupathi, C. N.-L. Tan, S. S. Tirumula, and S. K. Ray, "Liver disease detection using machine learning techniques."

[5] Maria Alex Kuzhippallil, Joseph Carolyn, and A Kannan, "Comparative Analysis of Machine Learning Techniques for Indian Liver Disease Patients," *International Conference on Advanced Computing & Communication Systems*, 2020.

[6] L. Anand and V. Neelanarayanan, "Liver disease classification using deep learning algorithm," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 12, pp. 5105–5111, Oct. 2019, doi: 10.35940/ijitee.L2747.1081219.

[7] S. N. N. Alfisahrin and T. Mantoro, "Data mining techniques for optimization of liver disease classification," in *Proceedings - 2013 International Conference on Advanced Computer Science Applications and Technologies, ACSAT 2013*, IEEE Computer Society, 2013, pp. 379–384. doi: 10.1109/ACSAT.2013.81.

[8] S. Tokala *et al.*, "Liver Disease Prediction and Classification using Machine Learning Techniques," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 14, no. 2, 2023, [Online]. Available: www.ijacsa.thesai.org

[9] A. Naik and L. Samant, "Correlation Review of Classification Algorithm Using Data Mining Tool: WEKA, Rapidminer, Tanagra, Orange and Knime," in *Procedia Computer Science*, Elsevier B.V., 2016, pp. 662–668. doi: 10.1016/j.procs.2016.05.251.

[10] Sina Bahramirad, Aida Mustapha, and Maryam Eshraghi, "Classification of Liver Disease Diagnosis: A Comparative Study," 2013.

[11] A.K.M Sazzadur Rahman, F. M. Javed Mehedi Shamrat, Zarrin Tasnim, Joy Roy, and Syed Akhter Hossain, "A COMPARATIVE STUDY ON LIVER DISEASE PREDICTION USING SUPPORT VECTOR MACHINE ALGORITHM," *International Research Journal of Modernization in Engineering Technology and Science*, Jan. 2023, doi: 10.56726/irjmets33175.

[12] S. Afrin *et al.*, "Supervised machine learning based liver disease prediction approach with LASSO feature selection," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 6, pp. 3369–3376, Dec. 2021, doi: 10.11591/eei.v10i6.3242.

[13] Alice Auxilia, "Accuracy Prediction using Machine Learning Techniques for Indian

Patient Liver Disease," 2018.

[14] I. Altaf, M. A. Butt, and M. Zaman, "HARD VOTING META CLASSIFIER FOR DISEASE DIAGNOSIS USING MEAN DECREASE IN IMPURITY FOR TREE MODELS," *Review of Computer Engineering Research*, vol. 9, no. 2, pp. 71–82, May 2022, doi: 10.18488/76.v9i2.3037.

[15] B. V. Ramana and R. S. Kumar Boddu, "Performance comparison of classification algorithms on medical datasets," in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC 2019*, Institute of Electrical and Electronics Engineers Inc., Mar. 2019, pp. 140–145. doi: 10.1109/CCWC.2019.8666497.

[16] V. Mohan, "Liver Disease Prediction using SVM and Naïve Bayes Algorithms," 2015. [Online]. Available: https://www.researchgate.net/publication/339551659

[17] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," 2002.

[18] A. F. Otoom, E. E. Abdallah, Y. Kilani, A. Kefaye, and M. Ashour, "Effective diagnosis and monitoring of heart disease," *International Journal of Software Engineering and its Applications*, vol. 9, no. 1, pp. 143–156, 2015, doi: 10.14257/ijseia.2015.9.1.12.

[19] Pushpendra Kumar and Ramjeevan Singh Thakur, "Early Detection of the Liver Disorder from Imbalance Liver Function Test Datasets," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 4, 2019, [Online]. Available: www.ijitee.org

[20] A. Gulia, R. Vohra, and P. Rani, "Liver Patient Classification Using Intelligent Techniques," *International Journal of Computer Science and Information Technologies*, vol. 5, no. 4, 2014, [Online]. Available: http://archive.ics.uci.edu/ml/

[21] R. Amin, R. Yasmin, S. Ruhi, M. H. Rahman, and M. S. Reza, "Prediction of chronic liver disease patients using integrated projection based statistical feature extraction with machine learning algorithms," *Inform Med Unlocked*, vol. 36, Jan. 2023, doi: 10.1016/j.imu.2022.101155.

[22] Jaswinder Singh and Kirti Kangra, "Prediction and Analysis of Liver Disorder Disease using Machine Learning Algorithms," *International Journal of Computer Science & Communication*, vol. 13, no. 2, pp. 26–32, 2022, Accessed: May 09, 2024. [Online]. Available: https://csjournals.com/IJCSC/PDF13-2/5.%20Jas.pdf

[23] E. Dritsas and M. Trigka, "Supervised Machine Learning Models for Liver Disease Risk Prediction," *Computers*, vol. 12, no. 1, p. 19, Jan. 2023, doi: 10.3390/computers12010019.

[24] A. Gulia, R. Vohra, and P. Rani, "Liver Patient Classification Using Intelligent Techniques." [Online]. Available: http://archive.ics.uci.edu/ml/

[25] J. Singh, S. Bagga, and R. Kaur, "Software-based Prediction of Liver Disease with Feature Selection and Classification Techniques," in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 1970–1980. doi: 10.1016/j.procs.2020.03.226.

[26] C. Geetha and A. R. Arunachalam, "Evaluation based Approaches for Liver Disease

Prediction using Machine Learning Algorithms," in *2021 International Conference on Computer Communication and Informatics, ICCCI 2021*, Institute of Electrical and Electronics Engineers Inc., Jan. 2021. doi: 10.1109/ICCCI50826.2021.9402463.

[27] R. Choudhary, T. Gopalakrishnan, D. Ruby, A. Gayathri, V. S. Murthy, and R. Shekhar, "An Efficient Model for Predicting Liver Disease Using Machine Learning," in *Data Analytics in Bioinformatics: A Machine Learning Perspective*, wiley, 2021, pp. 443–457. doi: 10.1002/9781119785620.ch18.

[28] Alex Kuzhippallil, Carolyn Joseph, and Kannan A, "Comparative Analysis of Machine Learning Techniques for Indian Liver Disease Patients," *International Conference on Advanced Computing & Communication Systems (ICACCS)*, 2020.