WORLD JOURNAL OF PHARMACEUTICAL RESEARCH
Coden USA: WJPRAP Impact Factor 8.453
Volume 14, Issue 21, 1585-1599. Research Article ISSN 2277–7105

ENHANCED LIVER DISEASE PREDICTION USING MACHINE LEARNING ALGORITHMS: IMPLEMENTATION, EVALUATION, AND IMPROVEMENTS

M. Reagan[1]* and Gore Shriraj Laxman[2]

[1,2]Department of MCA, School of Engineering, Chanakya University, Bengaluru-562165.

Article Received on 11 October 2025,
Article Revised on 01 November 2025,
Article Published on 01 November 2025,
https://doi.org/10.5281/zenodo.17540844

***Corresponding Author**
M. Reagan
Department of MCA, School of Engineering, Chanakya University, Bengaluru-562165.

How to cite this Article: M. Reagan1* and Gore Shriraj Laxman2 (2025). Enhanced Liver Disease Prediction Using Machine Learning Algorithms: Implementation, Evaluation, And Improvements. World Journal of Pharmaceutical Research, 14(21), 1585–1599.

ABSTRACT

This study presents an enhanced machine learning framework for liver disease prediction using the Indian Liver Patient Dataset (ILPD). Prior research primarily relied on Logistic Regression, K-Nearest Neighbours, and Support Vector Machines, achieving moderate accuracy but lacking depth in evaluation. In this work, we re-implemented these baseline models and extended the scope by incorporating Decision Tree, Random Forest, Naive Bayes, Gradient Boosting, and XGBoost. Comprehensive preprocessing steps—including label encoding, outlier treatment, normalization, and stratified train-test splitting—were applied to improve model reliability. Performance was assessed using accuracy, sensitivity, specificity, and ROC-AUC to capture clinical trade-offs between detecting diseased and healthy individuals. Results show that Logistic Regression and SVM achieve high sensitivity but poor specificity, Naive Bayes excels in specificity but underperforms in sensitivity, while ensemble methods—particularly XGBoost—offer the best balance across all metrics. Overall, the expanded approach provides a more robust and clinically meaningful evaluation framework for liver disease prediction.

KEYWORDS: Liver Disease, Machine Learning, Classification, ROC-AUC, XGBoost.

I. INTRODUCTION

Liver disease is a major global health concern, contributing significantly to morbidity and mortality rates worldwide. The liver plays a central role in metabolism, detoxification, and biochemical regulation, and early diagnosis of liver dysfunction is crucial for effective treatment and improved patient outcomes. Traditional diagnostic methods rely on laboratory tests and physician expertise, which can be resource-intensive and may delay timely detection.

Machine learning (ML) techniques offer a promising alternative by enabling automated, data-driven prediction of disease outcomes. Prior research has applied algorithmssuch as Logistic Regression, K-Nearest Neighbours (KNN), and Support Vector Machine (SVM) to the Indian Liver Patient Dataset (ILPD), achieving moderate predictive performance. However, these studies were often constrained by limited model scope, insufficient preprocessing, and a lack of comprehensive evaluation metrics beyond accuracy.

This research addresses these gaps by re-implementing baseline models and incorporating more advanced classifiers such as Decision Tree, Random Forest, Gradient Boosting, Naive Bayes, and XGBoost. In addition, robust preprocessing steps—label encoding, missing value imputation, normalization, and outlier handling—were applied to improve model reliability. Performance evaluation was expanded to include accuracy, sensitivity, specificity, and ROC-AUC, ensuring a clinically meaningful comparison of models. By analyzing outcomes on datasets with and without outliers, this study provides deeper insight into how preprocessing choices influence model performance. The ultimate aim is to identify models that balance sensitivity and specificity, thereby offering practical utility for clinical decision support in liver disease diagnosis.

II. RELATED WORK

A growing body of research has applied machine learning to liver disease prediction, but many works differ in terms of dataset quality, preprocessing rigor, and model choice. Early approaches such as Thirunavukkarasu et al.[19] employed Logistic Regression, KNN, and SVM on the ILPD dataset, achieving modest accuracy but offering limited analysis of sensitivity and specificity. Mazaheri et al.[9] explored multiple algorithms for classification, underscoring challenges in balancing performance and interpretability, while Jin et al.[10] focused on tree-based methods for decision support in liver patient data.

Rahman et al.[13] conducted a comparative study of six supervised classifiers to predict chronic liver disease, aiming to reduce the high costs associated with medical diagnosis.

Their work highlighted the strengths and weaknesses of each algorithm, providing a foundation for more cost-effective predictive systems. Durai et al.[14] applied a multi-stage pipeline including min–max normalization, PSO-based feature selection, classification, and accuracy evaluation, reporting that J48 achieved the highest accuracy (95.04%), demonstrating the importance of feature selection in enhancing classification performance. Similarly, Azam et al.[15] proposed hybrid model designs and comparison analyses, concluding that KNN was highly effective in feature selection for liver disease prediction.

Hartatik et al.[16] evaluated KNN and Naive Bayes using patient histories and laboratory data, reinforcing the role of simple yet interpretable classifiers in supporting clinical diagnosis. More recent contributions include Rabbi et al.[1], who compared multiple ML algorithms for liver disorder detection, and Ghazal et al.[2], who developed an intelligent model for early prediction using ensemble methods. Wu et al.[3][9] investigated fatty liver prediction using ML, while Ghosh et al.[4] and Afreen et al.[5] explored boosting and ensemble strategies for classification. Suryaprakash Reddy et al.[7] and Abdalrada et al.[8] further applied diverse ML techniques, with logistic regression and boosting showing strong performance.

Additional works like Ambesange et al.[10], Idris and Bhoite[11], and Alqahtani and Ryu[12] examined hyperparameter tuning, broader applications, and biomarker-based diagnostics. Fathi et al.[17] specifically emphasized SVM for classification, while Gogi and Vijayalakshmi[18] demonstrated the prognostic use of ML algorithms. Tanwar and Rahman[20] provided a comprehensive review of current progress and future opportunities in ML for liver disease diagnosis.

Collectively, these studies highlight the evolution of liver disease prediction research—from basic classifiers with minimal preprocessing to advanced ensemble and hybrid methods with feature engineering. While accuracies vary across methods, ensemble approaches such as Random Forest, Gradient Boosting, and XGBoost consistently emerge as strong performers. Building on this foundation, our work integrates diverse classifiers, robust preprocessing (including outlier handling), and clinically meaningful evaluation metrics (sensitivity, specificity, ROC-AUC) to offer a more balanced and reliable framework for liver disease prediction.

III. DATASET AND PREPROCESSING

A. Dataset Description

The dataset used for this study is the Indian Liver Patient Dataset (ILPD), which was obtained from the UCI Machine Learning Repository. The dataset consists of 583 individual medical records of patients from Andhra Pradesh, India. Each record is comprised of 10 biological attributes and 1 target variable that indicates whether the patient is diagnosed with a liver disease.

Attribute	Description
Age	Patient age in years
Gender	Male or Female
Total_Bilirubin	Total bilirubin level (mg/dL)
Direct_Bilirubin	Direct bilirubin level (mg/dL)
Alkaline_Phosphotase	ALP enzyme level (U/L)
Alamine_Aminotransferase	ALT enzyme level (U/
Aspartate_Aminotransferase	AST enzyme level (U/L)
Total_Proteins	Total protein in blood (g/dL)
Albumin	Albumin level (g/dL)
Albumin_and_Globulin_Ratio	Ratio of albumin to globulin
Liver_Disease (Target)	1 = Disease, 2= No Disease

The dataset is inherently imbalanced, with a greater number of records labeled as having liver disease (1) compared to the healthy class (2). Therefore, accuracy alone is not a sufficient evaluation metric — metrics such as sensitivity, specificity, and ROC-AUC were considered to assess the models more effectively.

B. Data Cleaning and Feature Engineering

To ensure the quality of data before feeding it into machine learning models, a comprehensive data preprocessing pipeline was developed, as follows.

1) Target Label Encoding

The original dataset labelled the target variable as 1 for patients with liver disease and 2 for those without. To align with standard binary classification practices in machine learning where 1 typically represents the positive class and 0 the negative these labels were remapped accordingly: 1 indicating the presence of liver disease and 0 indicating its absence.

2) Categorical Encoding

The Gender attribute, which was originally categorical with values "Male" and "Female," was converted into a numeric binary format using label encoding—assigning 1 to Male and 0 to Female. This transformation ensures compatibility with machine learning algorithms like Logistic Regression and SVM, which require numerical input rather than text.

3) Handling Missing Values

The dataset was analysed for null and missing values. A few entries in Albumin_and_Globulin_Ratio were missing and were imputed using the mean value of the respective column, as it retains the central tendency without skewing distribution.

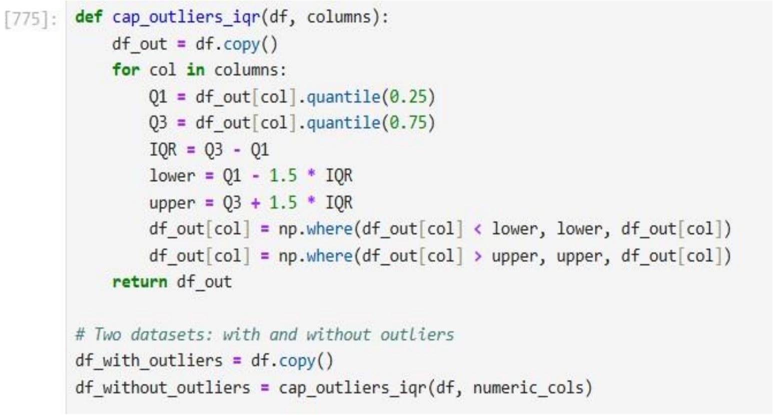
4) Outlier Detection and Handling

Boxplots were generated for all numerical attributes (Figure 1), which revealed significant high-value outliers in *Alkaline_Phosphotase*, *Alamine_Aminotransferase (ALT)*, and *Aspartate_Aminotransferase (AST)*, as well as right- skewed distributions in bilirubin variables. While some of these extreme values may represent true clinical cases, others likely stem from measurement or data entry errors. To minimize their undue influence on model training, we applied IQR-based capping (winsorization).

For each numeric feature, the first (Q1) and third (Q3) quartiles were computed, and the interquartile range (IQR = Q3 − Q1) was used to define lower and upper caps.

- **Lower bound** = Q1 − 1.5 × IQR
- **Upper bound** = Q3 + 1.5 × IQR

Values outside this range were replaced with the nearest bound. The procedure was implemented using the following function.



```
[775]: def cap_outliers_iqr(df, columns):
           df_out = df.copy()
           for col in columns:
               Q1 = df_out[col].quantile(0.25)
               Q3 = df_out[col].quantile(0.75)
               IQR = Q3 - Q1
               lower = Q1 - 1.5 * IQR
               upper = Q3 + 1.5 * IQR
               df_out[col] = np.where(df_out[col] < lower, lower, df_out[col])
               df_out[col] = np.where(df_out[col] > upper, upper, df_out[col])
           return df_out

       # Two datasets: with and without outliers
       df_with_outliers = df.copy()
       df_without_outliers = cap_outliers_iqr(df, numeric_cols)
```

This generated **two datasets**

- *With Outliers* (df_with_outliers) — the raw dataset preserved as-is.
- *Without Outliers* (df_without_outliers) — the IQR- capped version.

Both were used for training and testing, enabling direct comparison of how outliers affect model performance. Evaluation was performed separately on both versions (results_with and results_without) to highlight differences in accuracy, sensitivity, specificity, and ROC-AUC.

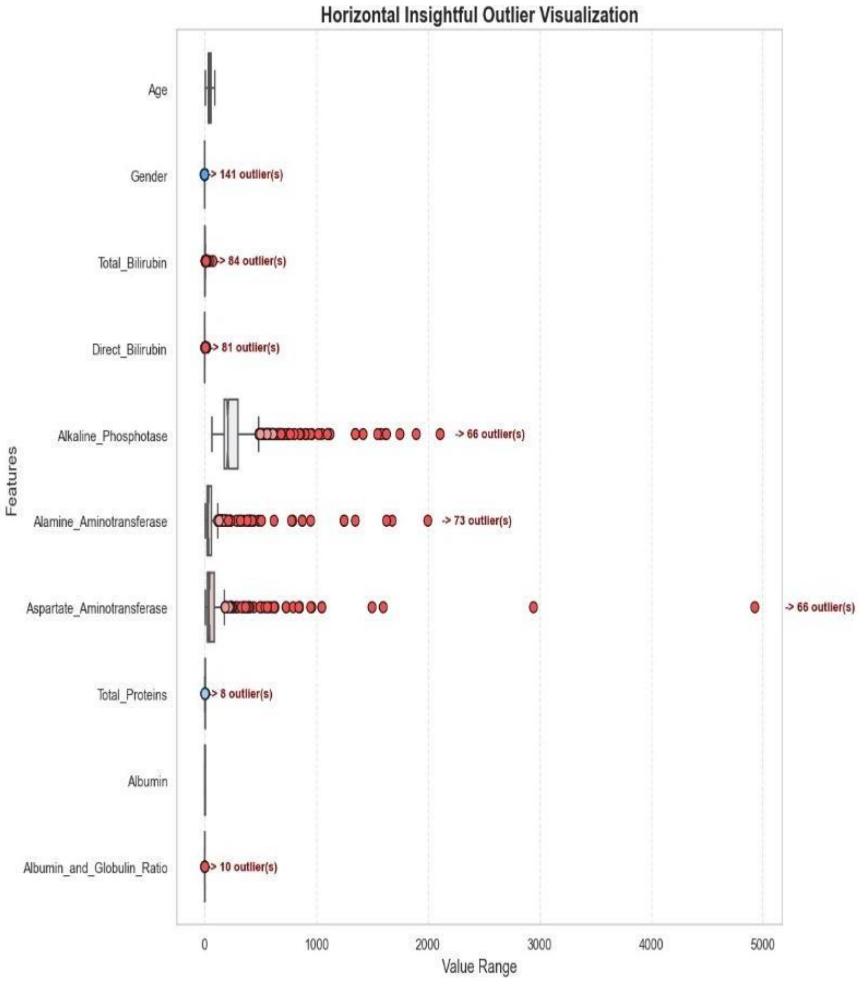


Figure 1: Boxplot for Outlier Detection across all numeric features.

5) Feature Scaling

As many features (e.g., bilirubin, enzyme levels) had skewed distributions, they were normalized using StandardScaler to ensure zero mean and unit variance. This scaling step is crucial for distance-based models (e.g., KNN) and gradient- based optimizers in SVM or Logistic Regression.

6) Correlation Analysis

To assess potential multicollinearity among the input features, a correlation matrix was generated. As anticipated, there was a strong positive correlation between Total Bilirubin and Direct Bilirubin, since the latter is a component of the former. Additionally, a significant linear relationship was observed between the liver enzymes AST (Aspartate Aminotransferase) and ALT (Alamine Aminotransferase), which often rise together in cases of liver dysfunction.

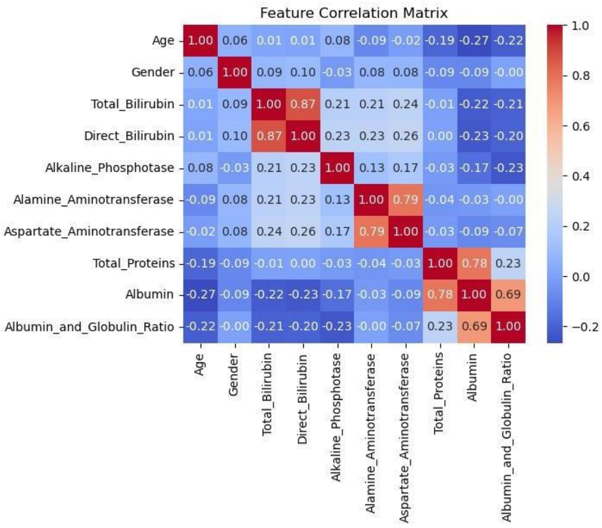


Figure 2: Heatmap illustrating the Pearson correlation between numerical attributes.

The heatmap helps in identifying redundant features and understanding inter-variable relationships, which is useful for feature selection and model interpretability.

C. Exploratory Data Visualization

To further understand class distribution and feature interaction, a pair plot was generated using the Seaborn library.

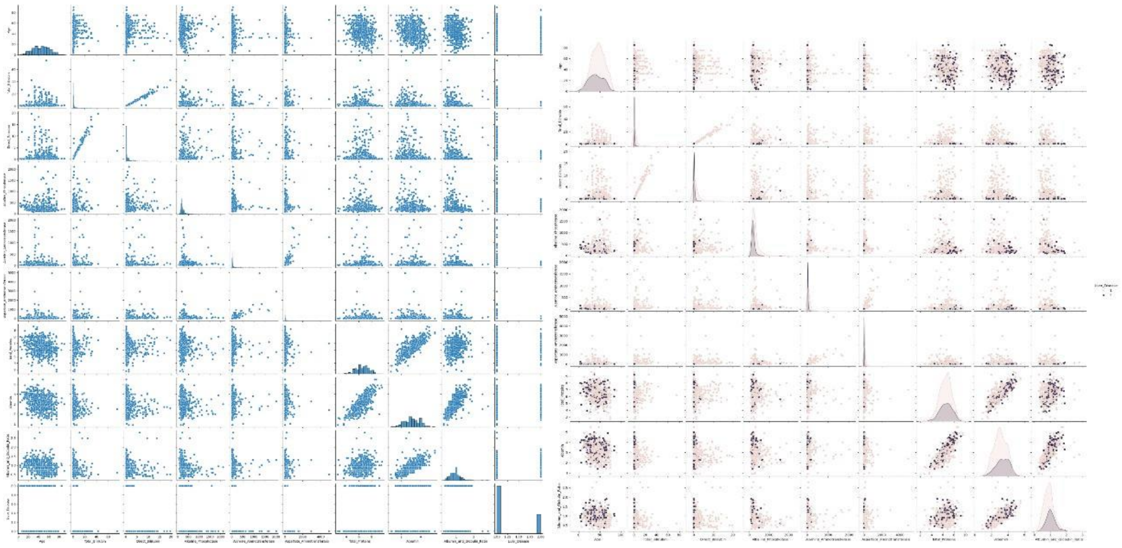


Figure 3.1 & 3.2: Pair plot showing feature relationships across liver disease and non-disease cases.

The pair plot shows that features like *Total Bilirubin*, *Direct Bilirubin*, and liver enzymes are generally higher in patients with liver disease, indicating some class separation. Strong correlations between *Total* and *Direct Bilirubin*, and among liver enzymes, suggest feature redundancy. The plot also confirms class imbalance, with more cases of liver disease. Several features are right-skewed, highlighting the need for normalization. Lastly, the lack of clear linear boundaries supports the use of ensemble or kernel-based models over simple linear classifiers.

IV. METHODOLOGY

In this study, we adopted a comprehensive machine learning approach for the prediction of liver disease, improving upon an earlier study by integrating more robust preprocessing, additional models, and modern evaluation strategies. The methodology is organized into two main components: the algorithms used and the processing pipeline.

A. Classification Models

1) Existing Models

The following models were originally implemented in the prior research and were re-evaluated in our enhanced study.

- **Logistic Regression**: A linear model used for binary classification that estimates the probability of class membership using the sigmoid function.
- **K-Nearest Neighbours (KNN)**: A non-parametric algorithm that classifies data points based on the majority class of the k nearest data points.
- **Support Vector Machine (SVM)**: A powerful classifier that aims to find the optimal separating hyperplane with the maximum margin between classes.

These models served as the baseline for comparative evaluation. While effective, they were limited in handling feature interactions and imbalanced datasets and lacked scalability for more complex patterns.

2) Enhanced Models (Implemented in This Study)

To address the limitations of the baseline models, more advanced classifiers were incorporated.

- **Decision Tree**: A simple yet interpretable model that recursively splits features. Susceptible to overfitting but serves as a foundation for ensemble methods.
- **Random Forest**: An ensemble of decision trees built on random subsets of data and features. Reduces overfitting, improves robustness, and provides feature importance.
- **XGBoost (Extreme Gradient Boosting)**: A scalable, regularized gradient boosting framework that builds trees sequentially, correcting prior errors. Known for high

predictive accuracy.

- **Gradient Boosting**: Similar to XGBoost but without advanced regularization. Effective in capturing non-linear feature interactions.
- **Naive Bayes**: A probabilistic model based on Bayes' theorem assuming independence among features. Efficient and interpretable but often less accurate when independence assumptions are violated.

These models were chosen due to their demonstrated success in biomedical classification problems where feature interactions and non-linearities are common.

B. Processing Pipeline

The complete machine learning pipeline used in this study followed these steps.

1. **Label Encoding and Missing Value Imputation** The Gender feature, originally in text form (Male/Female), was converted to numerical format using label encoding. Missing values, particularly in the Albumin and Globulin Ratio column, were filled using the mean of the respective column.
2. **Outlier Detection and Handling**

Boxplots revealed several extreme values, particularly in enzyme-related features (*ALP, ALT, AST*). To minimize their effect, we applied an **IQR- based capping procedure**, replacing values outside [Q1 − 1.5·IQR, Q3 + 1.5·IQR] with the respective boundary. Two datasets were thus created./

- **With Outliers** (original values)
- **Without Outliers** (IQR-capped values)

Both versions were used for training and evaluation, allowing us to assess the impact of outlier handling on model performance.

3. **Train-Test Splitting**

The dataset was split into training and testing sets using a 70:30 ratio. Stratified sampling was applied to maintain a balanced distribution of the target classes in both sets.

4. **Feature Normalization**

All numerical features were standardized using the StandardScaler technique. This step ensured that features with larger scales didn't dominate distance-based or gradient-based models like KNN or SVM.

5. Model Evaluation

Each model was evaluated using five key metrics to assess performance comprehensively.

- **Accuracy**: overall correctness of predictions.
- **Confusion Matrix**: detailed breakdown of true/false predictions.
- **Sensitivity (Recall)**: how well the model detects actual disease cases.
- **Specificity**: how well the model identifies healthy individuals.
- **ROC-AUC**: a threshold-independent measure of overall classification quality.

V. RESULTS AND DISCUSSION

We evaluated the performance of each classifier on both datasets: the **original dataset (with outliers)** and the **IQR- capped dataset (without outliers)**. The results include accuracy, sensitivity, specificity, and ROC-AUC. These metrics allow us to evaluate not only overall correctness but also how well each model handles the clinical trade-off between detecting diseased patients (sensitivity) and avoiding false positives (specificity).

Table 1: Results With Outliers.

Model	Accuracy	Sensitivity	Specificity	AUC
Logistic Regression	0.726	0.912	0.26	0.733
Decision Tree	0.589	0.704	0.30	0.502
Random Forest	0.663	0.824	0.26	0.717
KNN	0.657	0.840	0.20	0.649
SVM	0.709	0.992	0.00	0.617
Naive Bayes	0.554	0.384	0.98	0.765
XGBoost	0.686	0.840	0.30	0.676

Table 2: Results Without Outliers (IQR-Capped).

Model	Accuracy	Sensitivity	Specificity	AUC
Logistic Regression	0.726	0.904	0.28	0.758
Decision Tree	0.600	0.712	0.32	0.516
Random Forest	0.674	0.824	0.30	0.713
KNN	0.714	0.848	0.38	0.739
SVM	0.709	0.992	0.00	0.653
Naive Bayes	0.651	0.560	0.88	0.782
XGBoost	0.686	0.864	0.24	0.690

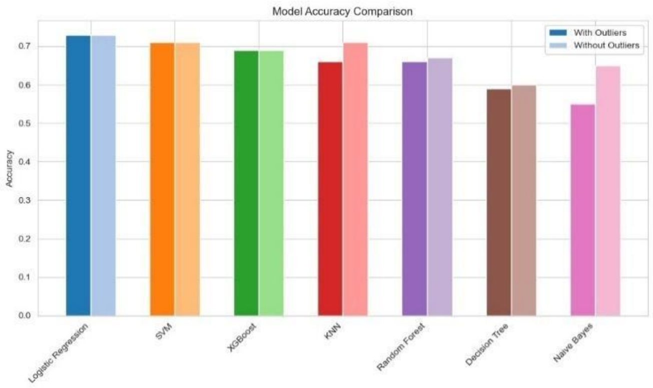


Figure 4: Accuracy comparison bar plot (With vs Without Outliers).

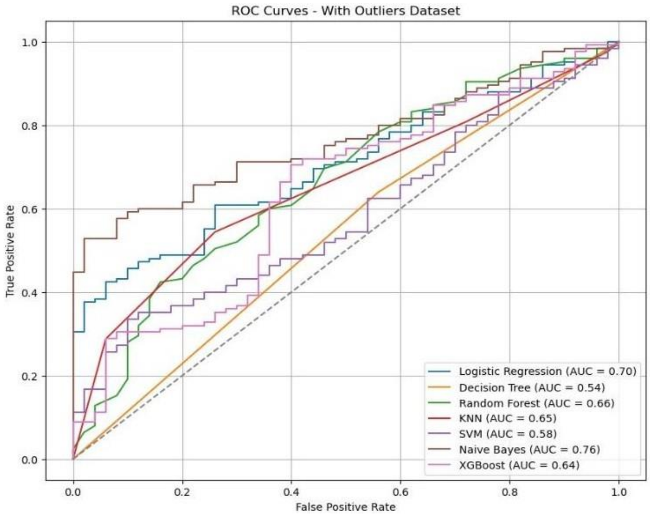


Figure 5: ROC curves of all models (With Outliers).

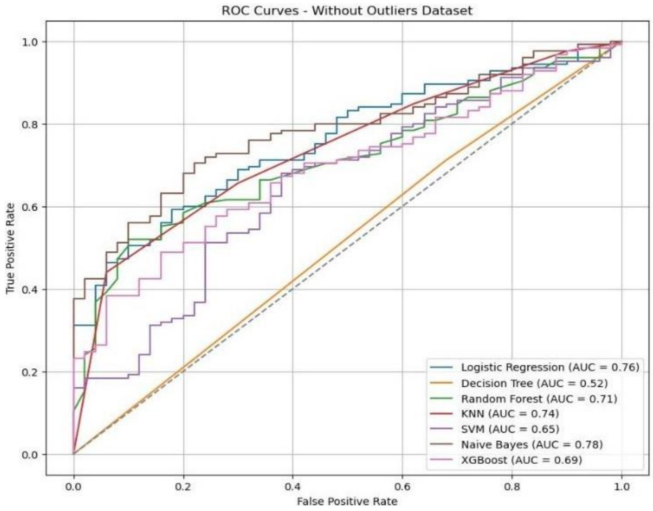


Figure 6: ROC curves of all models (Without Outliers).

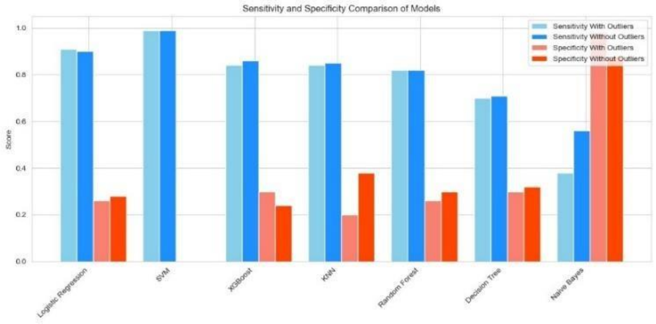


Figure 7: Sensitivity vs Specificity comparison chart.

The results show that Logistic Regression and SVM are very sensitive, consistently detecting nearly all diseased patients. However, they misclassify many healthy individuals, giving them low specificity. In contrast, Naive Bayes shows very high specificity, especially with outliers, but it fails to detect most diseased cases, leading to poor sensitivity. When outliers are capped, Naive Bayes becomes more balanced but still limited.

KNN benefits the most from outlier treatment: accuracy increases, and specificity nearly doubles while sensitivity remains strong. This is expected since KNN relies heavily on distance calculations, which are distorted by extreme values. Decision Trees perform modestly, offering interpretability but suffering from overfitting, and only show slight improvement after outlier handling.

Ensemble models—Random Forest and XGBoost—deliver the most stable and balanced outcomes. Random Forest provides consistent sensitivity with moderate specificity, while XGBoost offers the best overall trade-off. XGBoost maintains high sensitivity, decent specificity, and the strongest AUC scores, highlighting its ability to capture complex patterns and discriminate effectively between classes.

Overall, the effect of outlier treatment is most noticeable for distance-based and probabilistic models, while ensembles remain relatively robust. Clinically, model choice should depend on priorities: Logistic Regression or SVM when high sensitivity is vital, Naive Bayes when specificity is crucial, and XGBoost when a balanced, practical solution is required. Figures 4–7 visually reinforce these trade-offs, with bar plots showing accuracy shifts, ROC curves confirming ensemble superiority, and sensitivity–specificity charts highlighting clinical implications.

VI. CONCLUSION

This study showed how different machine learning models behave when predicting liver disease on datasets with and without outliers. Logistic Regression and SVM achieved very high sensitivity but at the expense of poor specificity, making them useful mainly for initial screening. Naive Bayes showed the opposite pattern, excelling in specificity but failing to capture most disease cases. KNN improved considerably once outliers were capped, highlighting the importance of preprocessing for distance-based models. Decision Trees offered interpretability but modest accuracy, while ensemble methods—especially XGBoost—consistently provided the best trade-off between sensitivity, specificity, and overall discrimination. Outlier handling improved certain models, but ensembles remained robust throughout. Clinically, XGBoost emerges as the most balanced choice for reliable early detection of liver disease.

VII. REFERENCE

1. Md. F. Rabbi, S. M. M. Hasan, A. I. Champa, Md. A. Zaman, and Md. K. Hasan, “Prediction of Liver Disorders using Machine Learning Algorithms: A Comparative Study,” *Proc. 2nd Int. Conf. Adv. Inf. Commun. Technol. (ICAICT)*, Nov. 2020. doi:10.1109/ICAICT51780.2020.9333528.
2. T. M. Ghazal, A. U. Rehman, M. Saleem, M. Ahmad, S. Ahmad, and F. Mehmood, “Intelligent Model to Predict Early Liver Disease using Machine Learning Technique,” *IEEE Xplore*, Feb. 2022. Available:https://ieeexplore.ieee.org/document/975 8929
3. C.-C. Wu et al., “Prediction of fatty liver disease using machine learning algorithms,” *Comput. Methods Programs Biomed.*, Mar. 2019; 170: 23–29,. doi:10.1016/j.cmpb.2018.12.032.
4. M. Ghosh et al., “A Comparative Analysis of Machine Learning Algorithms to Predict Liver Disease,” *Intell. Autom. Soft Comput.*, 2021; 30(3): 917–928, doi: 10.32604/iasc.2021.017989.
5. N. Afreen, R. Patel, M. Ahmed, and M. Sameer, “A Novel Machine Learning Approach Using Boosting Algorithm for Liver Disease Classification,” *ISCON, 2021*. doi: 10.1109/ISCON52037.2021.9702488.
6. X. Pei, Q. Deng, Z. Liu, X. Yan, and W. Sun, “Machine Learning Algorithms for Predicting Fatty Liver Disease,” *Ann. Nutr. Metab.*, 2021; 77(1): 38–45, doi:10.1159/000513654.
7. C. S. Reddy, L. S. Kiran, and A. Xavier, “Comparative Analysis of Liver Disease

Detection using Diverse Machine Learning Techniques," *ICICCS, 2022*. doi:10.1109/ICICCS53718.2022.9788208.

8. S. Abdalrada, O. H. Yahya, A. H. M. Alaidi, N. A. Hussein, H. T. Alrikabi, and T. A.-Q. Al-Quraishi, "A Predictive model for liver disease progression based on logistic regression algorithm," *Period. Eng. Nat Sci.*, Sep. 2019; 7(3): 1255. doi:10.21533/pen.v7i3.667.
9. S. Ambesange, R. Nadagoudar, R. Uppin, V. Patil, S. Patil, and S. Patil, "Liver Diseases Prediction using KNN with Hyper Parameter Tuning Techniques," *IEEE Xplore*, Oct. 2020. Available: https://ieeexplore.ieee.org/document/9297949
10. K. Idris and S. Bhoite, "Applications of Machine Learning for Prediction of Liver Disease," *Int. J. Comput. Appl. Technol. Res.*, 2019; 8(9): 394–396, doi: 10.7753/ijcatr0809.1012.
11. S. A. Alqahtani and S. Ryu, "Nonalcoholic fatty liver disease: use of diagnostic biomarkers and modalities in clinical practice," *Expert Rev. Gastroenterol. Hepatol.*, Aug. 2021; 21(10): 1065–1078, doi:10.1080/14737159.2021.1964958.
12. F. Rahman, M. Javed, Z. Tasnim, J. Roy, and S. A. Hossain, "A Comparative Study On Liver Disease Prediction Using Supervised Machine Learning Algorithms," *Int. J. Sci. Technol. Res.*, Nov. 2019; 8(11): 419–422.
13. V. Durai, S. Ramesh, and D. Kalthireddy, "Liver Disease Prediction using Machine Learning," *Int. J. Eng. Adv. Technol.*, Aug. 2019; 8(6): 2532–2534,. doi:10.35940/ijeat.F8365.088619.
14. M. Azam, F. Rahman, J. Iqbal, and S. Ahmed, "Prediction of Liver Diseases by Using Few Machine Learning Based Approaches," *Aust. J. Eng. Innov. Technol.*, Oct. 2020; 85–90. doi: 10.34104/ajeit.020.085090.
15. H. Hartatik, M. B. Tamam, and A. Setyanto, "Prediction for Diagnosing Liver Disease in Patients using KNN and Naïve Bayes Algorithms," *Proc. ICORIS 2020*. doi: 10.1109/icoris50180.2020.9320797.
16. M. Fathi, M. Nemati, S. M. Mohammadi, and R. Abbasi-Kesbi, "A Machine Learning Approach Based on SVM for Classification of Liver Diseases," *Biomed. Eng. Appl. Basis Commun.*, Jun. 2020; 32(3): 2050018, doi: 10.4015/S1016237220500180.
17. V. J. Gogi and M. N. Vijayalakshmi, "Prognosis of Liver Disease: Using Machine Learning Algorithms," *Proc. ICRIEECE 2018*. doi:10.1109/icrieece44171.2018.9008482.
18. K. Thirunavukkarasu, A. S. Singh, M. Irfan, and A. Chowdhury, "Prediction of Liver Disease using Classification Algorithms," *Proc. ICCCA, 2018*.

doi:10.1109/ccaa.2018.8777655.

19. N. Tanwar and K. F. Rahman, “Machine Learning in Liver Disease Diagnosis: Current Progress and Future Opportunities,” *IOP Conf. Ser.: Mater. Sci. Eng.*, Jan. 2021; 1022: p. 012029,. doi: 10.1088/1757-899X/1022/1/012029.