

An Adaptive Balancing and Threshold-Tuning Framework with Hybrid Feature Selection for Liver Disease Classification

Ahood Aldalbahi

Shaqra University

Samia Dardouri

s.dardouri@su.edu.sa

École Supérieure des Communications de Tunis

Research Article

Keywords: liver disease detection, early detection, ILPD, LDPD, 5C-HFS, ABTT, hybrid feature selection, imbalanced data, threshold tuning, CatBoost

Posted Date: October 23rd, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11333364/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

An Adaptive Balancing and Threshold-Tuning Framework with Hybrid Feature Selection for Liver Disease Classification

Ahood Aldalbahi¹ and Samia Dardouri^{1,2*}

¹ Department of Computer Science, College of Computing and Information Technology, Shaqra University, Shaqra, Saudi Arabia;

² Innov'Com Laboratory, Sup'Com, University of Carthage, Tunis, Tunisia;

* Correspondence: s.dardouri@su.edu.sa (S.D.)

Abstract

Accurate and early detection of liver disease is essential for reducing missed cases, supporting timely clinical intervention, and improving patient outcomes. One of the most widely used benchmark datasets for developing liver disease prediction models is the Indian Liver Patient Dataset (ILPD), which presents several practical challenges, including class imbalance, overlapping class boundaries, redundant laboratory variables, and limited sample size. These challenges motivate the development of adaptive screening models rather than conventional classifier-only pipelines. This study proposes a machine learning-based framework that integrates Five-Criteria Hybrid Feature Selection (5C-HFS) with Adaptive Balancing and Threshold Tuning (ABTT). The 5C-HFS component selects clinically meaningful features based on predictive relevance, class separation, bootstrap stability, redundancy, and required tests, while ABTT identifies the most suitable balancing strategy and operating threshold. Several models were evaluated, including Logistic Regression, Random Forest, XGBoost, LightGBM, CatBoost, and ensemble variants. The proposed CatBoost-based framework achieved an accuracy of 79.5%, with a recall of 91.6%, specificity of 50.0%, F1-score of 86.4%, MCC of 0.467, ROC-AUC of 0.786, and PR-AUC of 0.881. Additional validation on the Liver Disease Patient Dataset (LDPD) supported the adaptive behavior of the proposed framework. These results demonstrate the framework's potential to support interpretable and recall-oriented liver disease screening in clinical decision-support settings.

Keywords: liver disease detection; early detection; ILPD; LDPD; 5C-HFS; ABTT; hybrid feature selection; imbalanced data; threshold tuning; CatBoost

1. Introduction

Liver disease is a serious and widespread global health problem, spanning conditions from mild liver dysfunction to cirrhosis and chronic liver failure [1–3]. It is particularly dangerous because it often develops silently. Many patients experience no symptoms until the disease has reached an advanced stage, at which point effective treatment options become more limited [2]. Early detection is therefore important because timely intervention can slow disease progression and improve patient outcomes [1].

However, early diagnosis remains challenging. Diagnosis typically depends on biochemical blood tests and clinical indicators, and interpretation may vary according to patient condition, disease stage, and available medical resources [1]. Machine learning can support this process by learning nonlinear relationships across laboratory biomarkers and identifying high-risk patients from routine clinical data [4,5].

Several classifiers have been explored for liver disease prediction, including Logistic Regression, Random Forest, Support Vector Machines, Decision Trees, K-Nearest Neighbors, Naive Bayes, and boosting-based models [5,6]. Prior studies generally indicate that clinical laboratory features can support automated classification, but performance varies considerably depending on the dataset, preprocessing choices, selected features, and imbalance-handling strategy [4,7]. More importantly, accuracy alone is insufficient for medical screening because the clinical cost of missing a sick patient differs from the cost of a false alarm [8,9].

Meaningful challenges remain. The Indian Liver Patient Dataset (ILPD), one of the most widely used benchmarks in this field, illustrates these issues clearly [4,7,10]. It contains many difficulties common to clinical tabular data, including class imbalance, missing values, skewed biochemical markers, outliers, correlated laboratory variables, and overlapping distributions between liver and non-liver cases [7]. These properties make ILPD useful for evaluation, but they also show that simple classification pipelines may become unreliable when preprocessing, feature selection, and imbalance handling are not carefully designed [4,8].

These issues go beyond imbalance alone. Liver disease data often contain unclear class boundaries, especially in early-stage cases where laboratory values can resemble those of healthy individuals. Many laboratory variables are also redundant or strongly correlated, meaning that using more tests does not necessarily improve prediction and may be impractical in resource-limited settings [7,11]. A fixed pipeline for balancing, feature selection, and thresholding is therefore not well suited to these realities [4,9].

Despite these advances, existing liver disease prediction pipelines still tend to treat feature selection, imbalance handling, threshold tuning, and explainability as separate steps rather than as one integrated early-detection framework.

This study proposes an adaptive machine learning framework for liver disease screening. The first component is 5C-HFS, a five-criteria hybrid feature selection strategy that identifies a compact but clinically meaningful feature subset by considering predictive relevance, class separation, bootstrap stability, redundancy, and required tests. The second component is ABTT, an adaptive balancing and threshold tuning strategy that evaluates multiple imbalance-handling approaches and selects the balancing-threshold configuration that best fits the dataset rather than defaulting to a fixed resampling method.

The goal is a high-recall screening framework that detects liver patients reliably while preserving acceptable specificity. This recall emphasis is deliberate because, in early screening, false negatives may allow disease progression without diagnosis [8]. At the same time, the proposed framework does not optimize recall blindly. It uses adaptive threshold tuning and balanced evaluation to reduce missed liver cases without allowing non-liver class performance to collapse [9,12].

The pipeline begins with exploratory data analysis to characterize missing values, skewness, outliers, correlations, and class overlap. The data are then preprocessed, clinically motivated features are engineered, 5C-HFS narrows the feature space to the most informative elements, and ABTT selects the most suitable balancing strategy and operating threshold. Five models—Logistic Regression, Random Forest, XGBoost, LightGBM, and CatBoost—are trained and compared, with CatBoost emerging as the final model based on its screening performance. The study also incorporates SHAP-based patient-level explainability and a simple interactive interface that returns risk probability, classification, risk level, and explanation for a given set of laboratory inputs. The remainder of the paper is organized as follows: Section 2 reviews the related literature, Section 3 presents the methodology, Section 4 reports the results, Section 5 discusses the findings, and Section 6 concludes the study.

2. Related Work

Machine learning has been applied to liver disease prediction for some time, with early work built largely around familiar classifiers trained on clinical and biochemical data. Logistic Regression, Support Vector Machines, Decision Trees, Random Forest, K-Nearest Neighbors, and Naive Bayes have appeared regularly in these studies, and the broad finding has been encouraging: routine laboratory results can support automated liver disease classification [4–6]. However, reported performance varies considerably across studies depending on the dataset, preprocessing design, and evaluation metrics, making direct comparison difficult [4,5].

Rahman et al. provide a clear example of early classifier benchmarking work [6]. Using ILPD, they evaluated six classifiers and found Logistic Regression performing best at 75% accuracy, followed by Random Forest at 74% and Naive Bayes at 53%. The study supported machine learning as a low-cost diagnostic aid, but it relied mainly on standard classifier comparison and basic correlation-based feature reduction, leaving several practical questions related to clinical screening less explored.

Subsequent work became more methodologically layered. Rehman et al. developed MaLLiDD from a systematic review of 58 published articles, assembling a pipeline that combined attribute selection, brute-force feature optimization, SMOTE upsampling, normalization, cross-validation, and a voting ensemble [4]. On the Cirrhosis dataset, the framework achieved 99.56% accuracy, but performance fell to 76.56% on ILPD and 76.11% on the Cirrhosis staging dataset. This inconsistency suggests that liver disease prediction is dataset-sensitive and that imbalance-handling strategies may not transfer equally across distributions.

Feature selection has become a significant thread in this literature. Correlation filtering, feature importance scores, PCA, statistical ranking, metaheuristic search, and model-based selection have all been explored [4,7,11,13–16]. However, these approaches often do not jointly consider whether selected features are stable across patient subsets, clinically obtainable in routine screening, and sufficiently distinct from one another to justify inclusion.

Class imbalance is a related but distinct problem. Liver disease datasets frequently contain uneven class distributions, and the dominant class may differ across datasets. Standard remedies such as random oversampling, SMOTE, and stratified sampling have been widely applied [4,7,9,17]. The difficulty is that these techniques are usually chosen once before training and then held fixed. In datasets with poorly defined class boundaries or near-duplicate synthetic samples, committing to a single strategy may introduce problems that become visible only later.

Ensemble and boosting approaches have also been widely investigated. Ganie et al., Md et al., and Noor et al. evaluated configurations involving Random Forest, Gradient Boosting, XGBoost, LightGBM, bagging, stacking, and voting-based methods, generally reporting strong predictive performance [5,7,13,18]. However, comparisons across studies are limited by different train-test splits, preprocessing decisions, imbalance treatments, and reported metrics. Accuracy alone reveals little about how a model would behave in screening, where a missed positive diagnosis represents a different clinical failure than a false alarm [8].

Explainability has become increasingly important because a clinical prediction system that cannot explain its outputs is difficult to trust. Recent liver disease studies have started to incorporate explainability methods to clarify which biomarkers influence predictions. For example, Almusallam and Khan used SHAP-optimized hybrid features for chronic liver disease classification, while Mamun et al. proposed an explainability-enhanced liver disease diagnosis technique [19,20]. Related interpretable liver AI studies have also used feature-impact analysis to connect model outputs with clinically meaningful biomarkers

[21,22]. Even so, explainability is still often added after model development rather than treated as a design requirement from the outset.

Overall, the literature shows progress but limited integration. Feature selection, imbalance handling, ensemble modeling, and explainability have each attracted attention, but mostly as separate problems addressed by separate methods [4,7,20]. The present study addresses this gap by combining 5C-HFS feature selection, ABTT, comparative multi-model evaluation, CatBoost-based final classification, SHAP-based explanations, and an interface prototype within one clinical workflow. Table 1 summarizes this methodological positioning.

Table 1. Methodological comparison of related liver disease prediction studies.

Study	ML	Feature Selection	Imbalance Handling	Adaptive Balancing	Threshold Tuning	SHAP Interpretation	Interface
Rahman et al. [6]	✓	Limited	—	—	—	—	—
Rehman et al. [4]	✓	✓	✓	—	—	—	—
Ganie et al. [5]	✓	Feat. importance / Limited FS	✓	—	—	Limited	—
Md et al. [7]	✓	✓	✓	—	—	—	—
Noor et al. [13]	✓	✓	✓	—	—	SHAP / Limited	—
Mamun et al. [20]	✓	✓	—	—	—	SHAP / LIME	—
Proposed framework	✓	5C-HFS	ABTT	✓	✓	SHAP	✓

Symbols: ✓ = included; — = not reported or not central; Limited = partial feature handling or global/non-patient-level interpretability; SHAP / Limited = SHAP-based feature-impact visualization reported, but not integrated into a patient-level clinical interface.

3. Materials and Methods

This section describes the data preparation stages used before the proposed 5C-HFS + ABTT framework is applied. The focus is on the ILPD dataset, exploratory data analysis, preprocessing, and clinical feature engineering because these stages define the statistical and clinical quality of the input representation. The proposed framework itself is presented after the data preparation pipeline.

3.1. Dataset Description

The dataset used in this study is the Indian Liver Patient Dataset (ILPD), a widely used benchmark collected from clinical records in Andhra Pradesh, India and made available through the UCI Machine Learning Repository [10]. It contains 583 patient records and ten input attributes: age, gender, total bilirubin, direct bilirubin, alkaline phosphatase, alanine aminotransferase (ALT), aspartate aminotransferase (AST), total proteins, albumin, and albumin/globulin ratio. The target variable was encoded as a binary label, where liver disease cases were treated as the positive class and non-liver cases as the negative class. ILPD has been repeatedly used in liver disease prediction studies because it reflects several difficulties commonly observed in clinical tabular data, including class imbalance, missing laboratory values, skewed biochemical markers, correlated variables, and overlapping class boundaries [4,6,7,14].

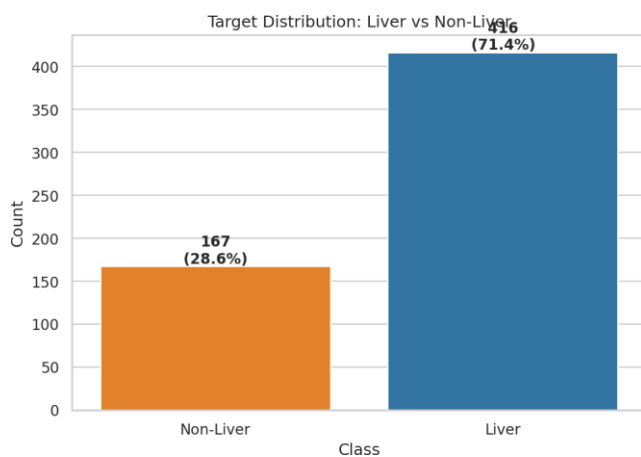
The dataset contains 416 liver disease cases and 167 non-liver cases, corresponding to 71.4% and 28.6% of the dataset, respectively. This gives an imbalance ratio of approximately 2.49:1. Because the intended application is early screening, this distribution is clinically important: a model may appear acceptable under accuracy while still failing to manage false negatives or minority-class behavior. Table 2 describes the variables used in this study.

Table 2. Description of the ILPD dataset attributes used in this study.

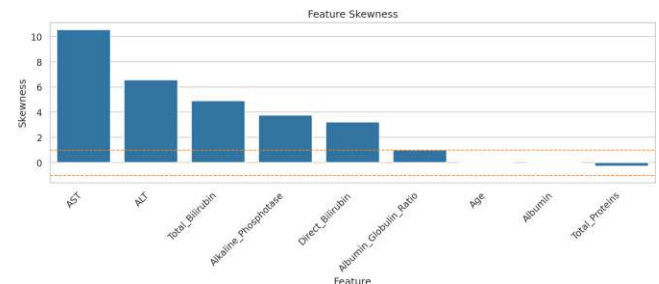
Attribute	Type	Description
Age	Numeric	Patient age in years.
Gender	Categorical	Patient gender; encoded numerically.
Total Bilirubin	Numeric	Total bilirubin concentration.
Direct Bilirubin	Numeric	Direct bilirubin concentration.
Alkaline Phosphatase	Numeric	Alkaline phosphatase enzyme level.
ALT	Numeric	Alanine aminotransferase marker.
AST	Numeric	Aspartate aminotransferase marker.
Total Proteins	Numeric	Total serum protein measurement.
Albumin	Numeric	Serum albumin measurement.
Albumin/Globulin Ratio	Numeric	Albumin-to-globulin ratio; four missing values.
Target	Binary	Liver disease vs. non-liver status.

3.2. Exploratory Data Analysis

Exploratory data analysis was performed before formal transformations to characterize class balance, missingness, distributional shape, outlier patterns, feature redundancy, class separation, and overlap structure. The class distribution and skewness findings are shown in Figure 1. Class imbalance was confirmed at 416 liver versus 167 non-liver cases, while missingness was limited to four records in albumin/globulin ratio. This supported imputation rather than record deletion because removing samples from a small clinical dataset could reduce statistical power.



(a) Target class distribution.



(b) Feature skewness analysis.

Figure 1. Class distribution and skewness analysis of the ILPD dataset. (A) Target class distribution showing imbalance between liver and non-liver cases. (B) Feature skewness analysis showing strong positive skewness in AST, ALT, total bilirubin, alkaline phosphatase, and direct bilirubin.

The most heavily skewed features were AST (10.55), ALT (6.55), total bilirubin (4.91), alkaline phosphatase (3.77), and direct bilirubin (3.21). These same biochemical markers also showed the highest outlier counts under the IQR rule, as summarized in Figure 2. Extreme biomarker values were not treated as simple data errors because abnormal enzyme and bilirubin readings may carry clinical meaning. Instead, the preprocessing stage used robust handling to reduce their excessive influence without aggressively deleting records.

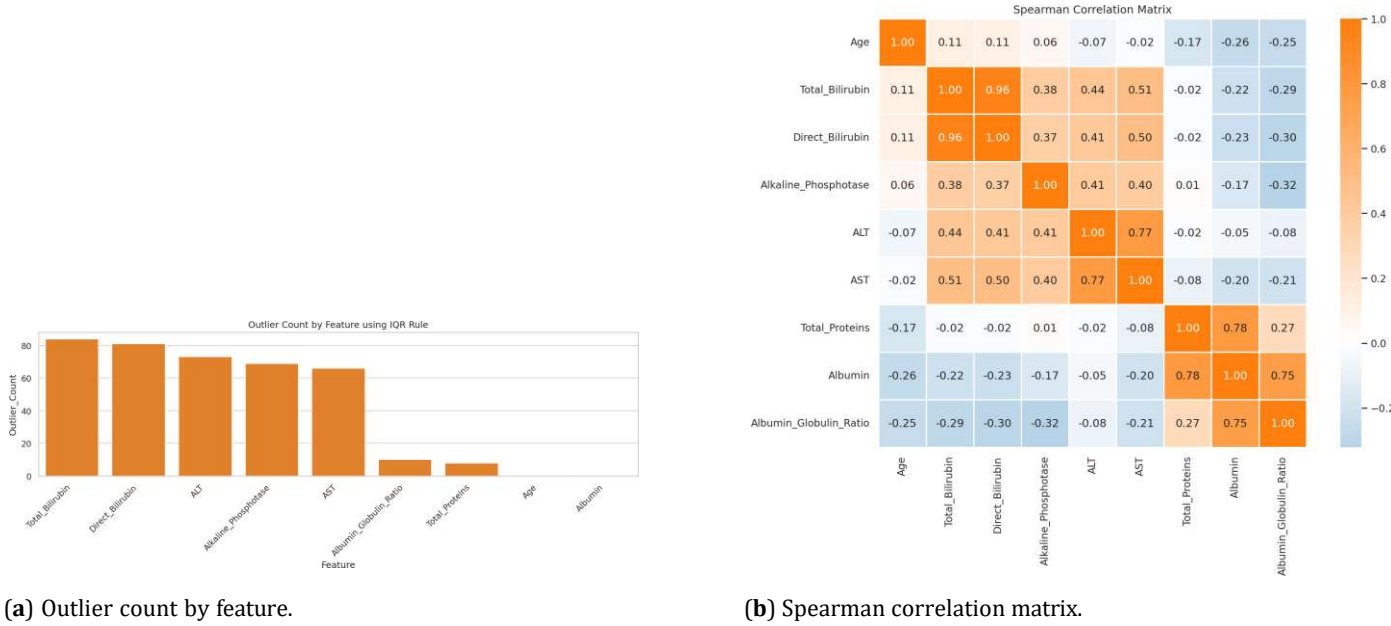


Figure 2. Outlier and redundancy analysis of the ILPD clinical variables. (A) Outlier count by feature using the IQR rule. (B) Spearman correlation matrix showing redundancy among bilirubin, enzyme, and protein-related variables.

Feature-target analysis indicated that AST, total bilirubin, direct bilirubin, ALT, and alkaline phosphatase had the strongest discrimination between liver and non-liver groups. However, the PCA projection in Figure 3 showed substantial overlap between the two classes, indicating that the dataset is not cleanly separable using a simple linear projection. Spearman correlation also confirmed high redundancy across several variable pairs, including total bilirubin and direct bilirubin (0.959), total proteins and albumin (0.779), ALT and AST (0.774), and albumin and albumin/globulin ratio (0.754). Together, these findings motivated robust preprocessing and feature selection designed to control redundancy explicitly.

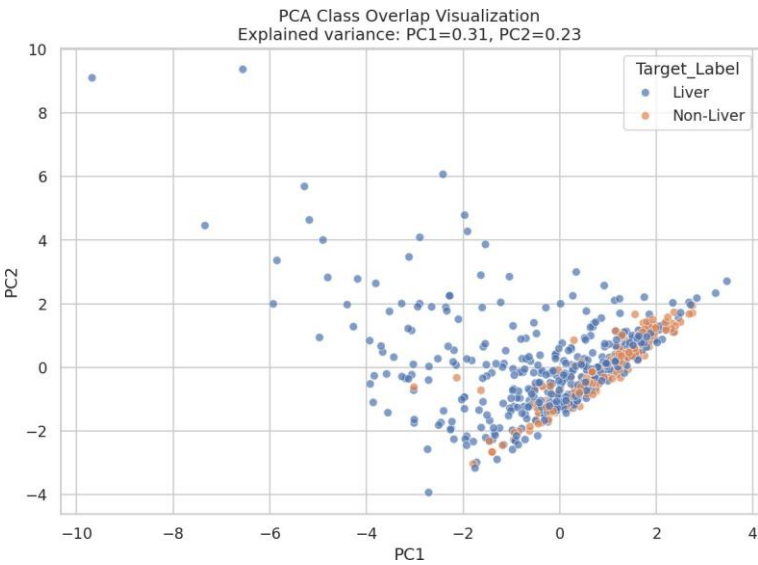


Figure 3. PCA-based class overlap visualization. The projection shows substantial overlap between liver and non-liver samples, supporting the need for nonlinear and adaptive modeling.

3.3. Data Preprocessing and Feature Engineering

The preprocessing stage prepared the ILPD records for reliable model development while minimizing data leakage. It included missing value treatment, gender encoding, outlier handling, feature scaling, and clinical feature engineering. These steps were guided by the exploratory findings and by prior evidence that preprocessing choices can substantially affect liver disease prediction performance, particularly when datasets contain skewed biomarkers, correlated variables, and class imbalance [4,7,23].

3.3.1. Missing Value Treatment and Encoding

Gender was encoded numerically, and the target label was converted into binary format. For the four missing albumin/globulin ratio values, median imputation was applied. Median imputation was selected because the affected feature and several related biochemical variables showed skewed distributions, making the median less sensitive to extreme readings than the mean. The imputation logic was applied within the training pipeline so that preprocessing parameters were learned from the training partition and then applied consistently to validation and test data, reducing the risk of data leakage [4,7,11].

3.3.2. Outlier Handling and Scaling

Rather than deleting records with extreme biomarker readings, percentile-based winsorization was applied with bounds learned strictly from the training set. This approach reduces the influence of extreme values while preserving clinically meaningful observations. After winsorization, numerical variables were standardized using z-score scaling. Scaling was required because enzyme markers, bilirubin values, protein measures, and age occupy different numerical ranges, and it also improves the behavior of scale-sensitive models such as Logistic Regression.

3.3.3. Clinical Feature Engineering

Clinical feature engineering was used to construct additional variables from routine liver function markers. The engineered features were added to the candidate feature space rather than forced into the final model. This allowed the later 5C-HFS stage to evaluate them alongside the original variables and retain only features that supported predictive relevance, class separation, redundancy control, and required-test efficiency. Similar liver disease studies support the value of engineered, projection-based, and feature-optimized representations, but they also show that unnecessary feature complexity should be controlled [11,23,24]. Table 3 summarizes the engineered features used in this study.

Table 3. Clinically motivated engineered features generated before 5C-HFS selection.

Engineered Feature	Formula / Construction	Purpose
De Ritis Ratio	AST / ALT	Captures the AST–ALT enzyme relationship.
Globulin Gap	Total Proteins – Albumin	Approximates non-albumin proteins.
Bilirubin Ratio	Direct Bilirubin / Total Bilirubin	Represents conjugated bilirubin proportion.
ALBI Proxy	$\log_{10}(\text{Total Bilirubin}) - 0.66 \times \text{Albumin}$	Approximates liver function severity.
Liver Stress Score	$(\text{AST} + \text{ALT}) / \text{Albumin}$	Relates enzyme elevation to albumin status.
Liver Enzyme Load	AST + ALT + Alkaline Phosphatase	Summarizes combined enzymatic burden.

3.4. Proposed Framework

The proposed framework is designed as an early-detection pipeline rather than a conventional classifier-selection workflow. As shown in Figure 4, the workflow begins with ILPD data preparation, continues through Five-Criteria Hybrid Feature Selection (5C-HFS), applies Adaptive Balancing and Threshold Tuning (ABTT), and then proceeds to final model development, evaluation, SHAP explainability, and interface demonstration. The main objective is to reduce false negatives while preserving acceptable specificity, controlling redundant features, and limiting the number of required clinical inputs.

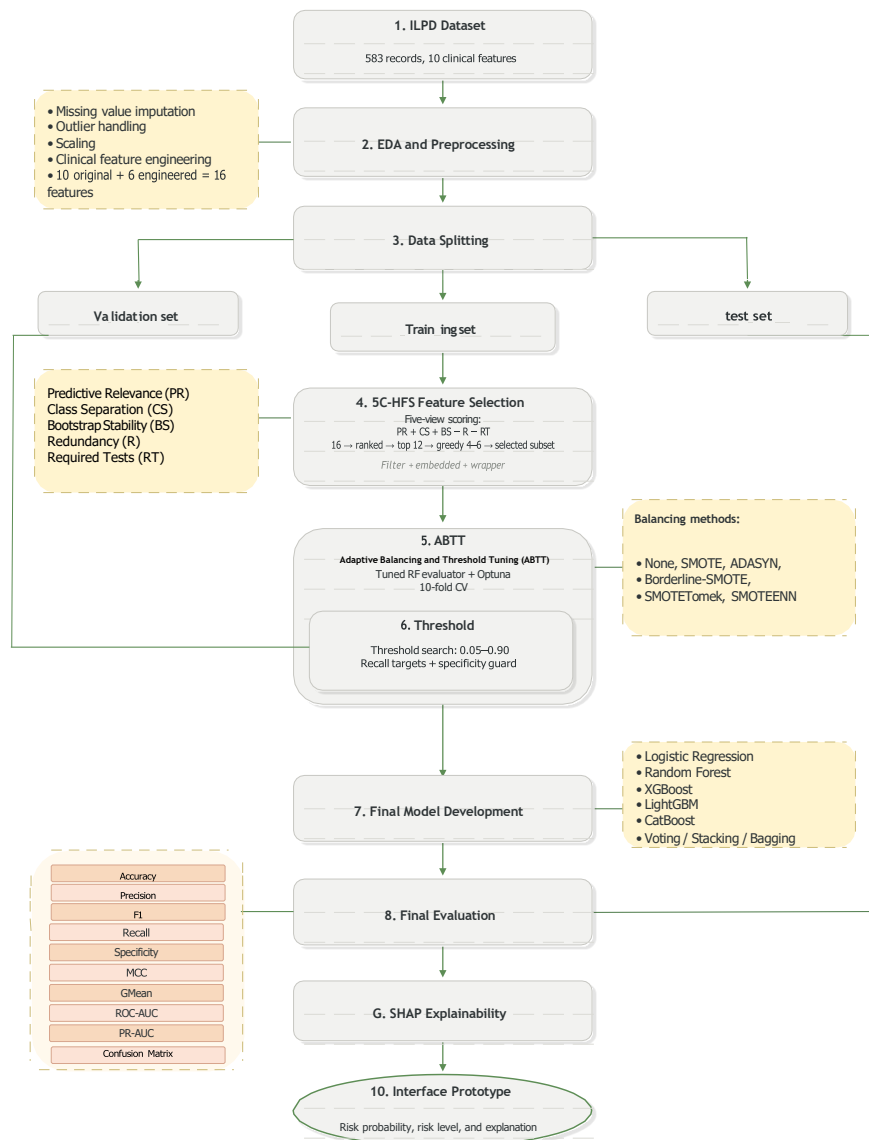


Figure 4. Overview of the proposed 5C-HFS + ABTT framework for early liver disease detection.

The framework is motivated by two properties observed in the exploratory analysis: class overlap and feature redundancy. The PCA projection showed that liver and non-liver cases are not cleanly separable, while the correlation analysis showed that several biochemical markers carry overlapping information. Therefore, a fixed feature selection method or a fixed resampling method is unlikely to be consistently optimal. The proposed design instead separates the problem into two adaptive layers: 5C-HFS selects a compact representation by balancing predictive relevance, class separation, bootstrap stability, redundancy, and required tests, while ABTT selects the most suitable balancing strategy

and decision threshold according to clinical utility. This design follows recent liver disease prediction studies that emphasize preprocessing, feature selection, imbalance handling, and ensemble learning, but extends them by integrating these components into a single clinically oriented pipeline [4,5,7,11].

3.4.1. Five-Criteria Hybrid Feature Selection

The proposed 5C-HFS component evaluates each candidate feature using five criteria: Predictive Relevance, Class Separation, Bootstrap Stability, Redundancy, and Required Tests. It is called hybrid because it combines three feature selection families. The filter-based part appears through class separation, redundancy, and mutual-information-based measures; the embedded part appears through model-derived feature importance; and the wrapper-based part appears later through greedy subset evaluation using cross-validation. This is important because previous liver disease studies show that feature selection and feature optimization can improve prediction, but many approaches rely mainly on ranking, projection, or a single model-based view without jointly considering stability, redundancy, and required tests [11,14–16,23,24].

The method started from 16 candidate features, consisting of the 10 original ILPD variables and six clinically engineered variables. A feature-level score was first computed for every candidate feature. The first three criteria increase the feature score because they represent predictive usefulness, distributional separation between classes, and stability across resampled training data. In contrast, Redundancy and Required Tests are subtracted because they represent informational repetition and additional original clinical tests. The final feature-level score is defined as follows:

$$\text{Score}(f) = 0.35 PR(f) + 0.25 CS(f) + 0.20 BS(f) - 0.12 R(f) - 0.03 RT(f), \quad (1)$$

where $PR(f)$ denotes Predictive Relevance, $CS(f)$ denotes Class Separation, $BS(f)$ denotes Bootstrap Stability, $R(f)$ denotes Redundancy, and $RT(f)$ denotes Required Tests. These weights give the highest emphasis to predictive usefulness and class-separation ability, while still rewarding stability and penalizing redundant features and features that require more original clinical measurements.

Predictive Relevance was computed using multiple relevance views rather than relying on a single model. These included mutual information, LightGBM feature importance, Random Forest feature importance, Logistic Regression coefficient magnitude, and MCC-based permutation importance. Class Separation, illustrated in Figure 5, was computed using the Kolmogorov-Smirnov statistic and absolute Cliff's Delta, which measure how differently a feature is distributed between liver and non-liver cases. Bootstrap Stability was estimated using 80 stratified bootstrap samples to identify features that remain informative across resampled training subsets rather than appearing important only by chance.

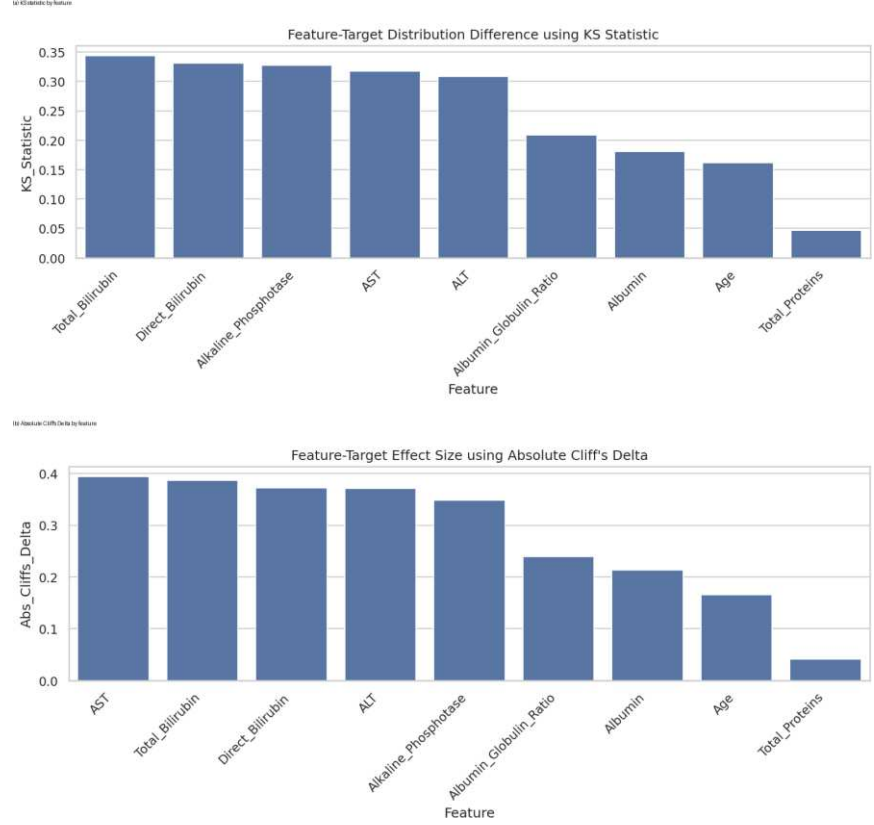


Figure 5. Class separation analysis using KS statistic and Cliff's Delta for candidate features. Higher values indicate stronger distributional differences between liver and non-liver cases, supporting the Class Separation component in 5C-HFS.

Redundancy was treated as a soft penalty rather than as a direct deletion rule. This is necessary because liver biomarkers may be naturally correlated. For example, AST and ALT or total bilirubin and direct bilirubin may reflect related physiological processes. Therefore, 5C-HFS measures redundancy from two perspectives: linear redundancy and nonlinear redundancy. Linear redundancy is based on Spearman correlation, while nonlinear redundancy is based on mutual information. These two components are normalized and combined into a single redundancy score:

$$R_{\text{linear}}(f) = \max_{g \in F, g \neq f} |\rho(f, g)|, \quad (2)$$

$$R_{\text{nonlinear}}(f) = \max_{g \in F, g \neq f} MI(f, g), \quad (3)$$

$$R(f) = 0.5 \tilde{R}_{\text{linear}}(f) + 0.5 \tilde{R}_{\text{nonlinear}}(f), \quad (4)$$

where $\rho(f, g)$ is the Spearman association between two features, $MI(f, g)$ is their mutual information, and $R(f)$ is the combined redundancy score. Because the linear and nonlinear redundancy components have different numerical scales, both components were normalized before combination, yielding $\tilde{R}_{\text{linear}}(f)$ and $\tilde{R}_{\text{nonlinear}}(f)$. A higher redundancy value indicates that a feature carries more repeated information relative to other candidate features. This does not automatically remove the feature, but it reduces its score unless the feature provides enough predictive relevance, class separation, or stability to justify inclusion.

The Required Tests criterion measures the number of original clinical measurements required to use a feature. This is especially important for engineered features because one engineered variable may require multiple original laboratory measurements. For example, Liver_Stress_Score requires AST, ALT, and Albumin, while Bilirubin_Ratio requires Direct_Bilirubin and Total_Bilirubin. The Required Tests criterion is computed as:

$$RT(f) = |\text{RequiredTests}(f)|, \quad (5)$$

and at the subset level as:

$$RT(S) = \sum_{f \in S} \text{RequiredTests}(f). \quad (6)$$

This formulation distinguishes between the number of selected features and the number of original laboratory tests required in practice, making the selected subset more realistic for clinical screening. Table 4 gives examples.

Table 4. Examples of required tests for engineered features.

Engineered Feature	Required Original Inputs	Required Tests
Liver_Stress_Score	AST, ALT, Albumin	3
Bilirubin_Ratio	Direct_Bilirubin, Total_Bilirubin	2
De_Ritis_Ratio	AST, ALT	2
Liver_Enzyme_Load	AST, ALT, Alkaline_Phosphatase	3
ALBI_Proxy	Total_Bilirubin, Albumin	2

After feature-level scoring, the features were ranked, and the top 12 features were passed to a greedy forward subset search. The method does not simply select the top-ranked five features because individually strong features may be redundant or may not perform well when combined. Candidate subsets within a range of four to six features were evaluated using 10-fold stratified cross-validation. Each subset was scored using a clinical utility function that combines Recall, MCC, GMean, and Specificity while penalizing excessive redundancy, larger subset size, and a higher number of required tests:

$$\begin{aligned} U(S) = & 0.35 \text{ Recall}(S) + 0.30 \text{ MCC}(S) + 0.20 \text{ GMean}(S) \\ & + 0.15 \text{ Specificity}(S) - 0.04 \text{ EMR}(S) - 0.04 \text{ EMaxR}(S) \\ & - 0.004|S| - 0.002 \text{ RT}(S). \end{aligned} \quad (7)$$

where EMR(S) and EMaxR(S) denote excess mean redundancy and excess maximum redundancy, respectively. This utility function favors subsets that improve sensitivity and balanced classification quality while avoiding unnecessarily large or redundant feature sets and features requiring more original tests. Additional protection was applied when recall or specificity fell below acceptable clinical limits, preventing the selection of a subset that appears numerically strong but is unsuitable for early screening.

3.4.2. Adaptive Balancing and Threshold Tuning

The proposed ABTT component is applied after feature selection to jointly select the balancing strategy and decision threshold. This joint design is important because balancing changes the training distribution and affects predicted probabilities, while the decision threshold determines how these probabilities are converted into class labels. Selecting a balancing method without threshold tuning may produce a sensitive model with too many false positives, while threshold tuning without testing balancing strategies may leave the classifier poorly adapted to the class structure. ABTT therefore combines balancing selection and threshold tuning under explicit screening constraints [4,9,17]. This adaptive

view is also consistent with recent work showing that patient heterogeneity and dynamic classifier selection can affect liver-related prediction tasks [25].

ABTT compared six balancing strategies, summarized in Table 5: None, SMOTE, ADASYN, Borderline-SMOTE, SMOTETomek, and SMOTEENN. An internally tuned Random Forest was used as the evaluation model for comparing balancing methods. For each balancing strategy, Optuna performed 25 trials to tune Random Forest hyperparameters, and performance was estimated using 10-fold stratified cross-validation. The Random Forest in this stage was used as a consistent internal evaluator rather than the final classifier.

Table 5. Balancing methods evaluated in ABTT.

Method	Purpose
None	Tests whether resampling is unnecessary or harmful.
SMOTE	Generates synthetic minority samples.
ADASYN	Focuses generation around difficult minority cases.
Borderline-SMOTE	Generates samples near class-boundary regions.
SMOTETomek	Combines SMOTE with Tomek-link cleaning.
SMOTEENN	Combines SMOTE with stronger ENN cleaning.

Within each cross-validation fold, ABTT trained the internal evaluator, obtained predicted probabilities, and searched for an operating threshold instead of using the default 0.50 cutoff. Thresholds were searched in the range from 0.05 to 0.90. The algorithm first attempted to satisfy high recall targets and gradually relaxed them if needed:

$$R_{\text{target}} \in \{0.90, 0.88, 0.85, 0.83, 0.80\}, \quad (8)$$

subject to:

$$\text{Recall}(t) \geq R_{\text{target}}, \quad \text{Specificity}(t) \geq 0.40, \quad (9)$$

where t denotes the decision threshold. If no threshold satisfies the recall-specificity constraint, the fallback rule selects the operating point with the highest MCC, followed by GMean and Recall. This makes threshold selection recall-protected while preventing specificity from collapsing below the minimum acceptable level.

After threshold tuning within each fold, a clinical score was computed as:

$$\text{ClinicalScore} = 0.35 \text{ Recall} + 0.30 \text{ MCC} + 0.20 \text{ GMean} + 0.15 \text{ Specificity}. \quad (10)$$

Additional penalties were applied if fold-level performance failed to satisfy the minimum clinical guards:

$$\text{ClinicalScore} = \text{ClinicalScore} - 0.10 \quad \text{if} \quad \text{Recall} < 0.80, \quad (11)$$

$$\text{ClinicalScore} = \text{ClinicalScore} - 0.10 \quad \text{if} \quad \text{Specificity} < 0.40. \quad (12)$$

For each balancing strategy, the mean and standard deviation of this clinical score were computed across the 10 folds. The final ABTT score was then calculated as:

$$\text{ABTT}(b) = \text{MeanClinicalScore}(b) - 0.05 \times \text{STDClinicalScore}(b), \quad (13)$$

where b denotes the balancing method. This formulation rewards high screening performance while penalizing instability, so a balancing method with strong but inconsistent performance is not automatically preferred.

After all balancing strategies were scored, ABTT applied a clinical eligibility rule. It first searched for strategies satisfying $CV_Recall \geq 0.85$ and $CV_Specificity \geq 0.40$. If no method satisfied this condition, the recall requirement was relaxed to 0.83 and then 0.80. If no strategy met these relaxed constraints, the best available method according to the internal objective was used. When multiple methods were eligible, they were ranked by ABTT score, CV_MCC, CV_GMean, CV_Recall, and CV_Specificity. Validation metrics were then recorded before moving to the final model-comparison stage.

3.4.3. Model Development, SHAP Explainability, and Interface Prototype

After 5C-HFS and ABTT, the selected feature representation and balancing configuration were used for final model development. Several model families were evaluated because they represent different learning assumptions suitable for clinical tabular data. Logistic Regression was included as an interpretable linear baseline. Random Forest was included as a bagging-based tree ensemble that can model nonlinear relationships while reducing variance. XGBoost, LightGBM, and CatBoost were included as gradient-boosting models known to perform strongly on structured tabular data. In addition, voting, stacking, and bagging variants were evaluated to determine whether ensemble aggregation could improve classification behavior beyond individual models [5,7,13,18,26].

After the final model was selected, SHAP was used to interpret model outputs. Global SHAP analysis identified which selected features contributed most strongly to predictions, while local SHAP explanations showed how individual patient feature values shifted predictions toward liver or non-liver classification. This is important because a clinical screening model should not only return a prediction but also provide an interpretable explanation [13,19,20].

A lightweight interface prototype was also developed to demonstrate how the proposed framework could be connected to an applied screening workflow. The interface accepts the required original clinical inputs, applies the same preprocessing and feature engineering logic, generates selected features, and returns predicted probability, final class, risk level, and SHAP-based explanation. The interface is not presented as a clinically validated product; rather, it serves as a proof of concept showing that the proposed framework can be translated into an interpretable screening tool after appropriate clinical review and external validation.

4. Results

This section reports the experimental setup, evaluation metrics, feature selection results, ABTT balancing and threshold tuning results, final model performance, baseline comparison, ablation analysis, comparison with prior ILPD studies, LDPD validation, and SHAP explainability. The results are interpreted from a screening perspective: recall and false-negative reduction are prioritized, while specificity and balanced performance are monitored to avoid a one-sided classifier [4,5,13].

4.1. Experimental Setup

The experiments were implemented in Python using Google Colab. The main libraries included scikit-learn, imbalanced-learn, XGBoost, LightGBM, CatBoost, Optuna, and SHAP. A fixed random seed of 42 was used to improve reproducibility. The ILPD dataset was split using stratified sampling into training, validation, and sealed test subsets. The training set was used for preprocessing, clinical feature engineering, 5C-HFS feature selection, ABTT balancing comparison, and model fitting; the validation set supported threshold tuning; and the sealed test set was reserved for final evaluation.

The final framework used five 5C-HFS-selected features and seven required original clinical tests. ABTT selected SMOTE as the final balancing strategy for the ILPD experiment, and the final model used CatBoost with a tuned operating threshold of 0.30. Table 6 summarizes the experimental setup.

Table 6. Experimental setup summary.

Setting	Value
Environment	Google Colab / Python
Random seed	42
Data split	326 training / 140 validation / 117 sealed test samples
Feature selection	5C-HFS
Selected features	Liver_Stress_Score; Alkaline_Phosphatase; ALT; Age; Bilirubin_Ratio
Required original tests	ALT; AST; Age; Albumin; Alkaline_Phosphatase; Direct_Bilirubin; Total_Bilirubin
Balancing methods	None; SMOTE; ADASYN; Borderline-SMOTE; SMOTETomek; SMOTEENN
Selected balancing	SMOTE
Models	Logistic Regression; Random Forest; XGBoost; LightGBM; CatBoost
Ensemble variants	Hard Voting; Stacking; Bagging
Final model	CatBoost + SMOTE
Final threshold	0.30
Explainability	SHAP

4.2. Evaluation Metrics

The evaluation used multiple metrics because accuracy alone can be misleading for imbalanced medical screening data. Recall was prioritized to reduce missed liver disease cases, while specificity, F1-score, GMean, MCC, ROC-AUC, and PR-AUC were used to assess class balance and ranking quality. These metrics were computed from the confusion matrix components: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Table 7 summarizes the metrics used in the experiments.

Table 7. Evaluation metrics used in the experiments.

Metric	Formula	Purpose
Accuracy	$(TP + TN)/(TP + TN + FP + FN)$	Overall proportion of correct predictions.
Precision	$TP/(TP + FP)$	Reliability of positive predictions.
Recall	$TP/(TP + FN)$	Sensitivity to liver disease cases.
Specificity	$TN/(TN + FP)$	Ability to identify non-liver cases.
F1-score	$2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$	Balance between precision and recall.
GMean	$\sqrt{\text{Recall} \times \text{Specificity}}$	Balance between sensitivity and specificity.
MCC	$(TP \times TN - FP \times FN)/D$	Balanced correlation between predicted and true labels.
ROC-AUC	Area under ROC curve	Ranking quality across false-positive rates.
PR-AUC	Area under precision-recall curve	Positive-class ranking quality under imbalance.
For MCC, $D = \sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}$.		

4.3. 5C-HFS Feature Selection Results

The 5C-HFS stage selected a compact subset of five features: Liver_Stress_Score, Alkaline_Phosphatase, ALT, Age, and Bilirubin_Ratio. These features required seven original clinical tests: ALT, AST, Age, Albumin, Alkaline_Phosphatase, Direct_Bilirubin, and Total_Bilirubin. Under the screening constraint, the selected subset achieved cross-validated recall of 0.885, specificity of 0.503, MCC of 0.463, and GMean of 0.627. This result supports the goal of reducing input complexity while preserving predictive value for early detection. Table 8 presents the selected features and main scoring components.

Table 8. Features selected by 5C-HFS and main scoring components.

Selected feature	5C-HFS score	PR	CS	BS	RT
Liver_Stress_Score	0.611	0.845	0.999	1.000	3
ALT	0.522	0.754	0.825	0.700	1
Age	0.458	0.580	0.303	0.950	1
Alkaline_Phosphatase	0.382	0.355	0.708	0.675	1
Bilirubin_Ratio	0.222	0.275	0.693	0.350	2

The final subset was selected at the fifth greedy step, where the mean redundancy was 0.293 and the maximum redundancy was 0.832. Adding a sixth feature did not improve clinical utility sufficiently to justify additional complexity. This supports the role of 5C-HFS as a compact subset-selection layer rather than a simple top-ranked-feature selector.

4.4. ABTT Balancing and Threshold Tuning Results

As summarized in Table 9, ABTT compared six imbalance-handling strategies using cross-validation and a clinical objective based on recall, specificity, MCC, GMean, and stability. Although the no-resampling setting produced a slightly higher ABTT score, SMOTE was selected by the

the 5C-HFS-selected representation. This is consistent with the screening objective, where reducing false negatives is prioritized while specificity remains constrained.

Table 9. ABTT comparison across balancing methods.

Method	ABTT score	Recall	Specificity	MCC	GMean
NONE	0.631	84.5%	59.4%	0.476	0.695
SMOTE	0.619	91.5%	50.4%	0.508	0.638
ADASYN	0.588	78.8%	62.0%	0.443	0.656
Borderline-SMOTE	0.553	73.0%	63.3%	0.368	0.646
SMOTETomek	0.614	82.4%	59.6%	0.458	0.675
SMOTEENN	0.559	72.5%	69.9%	0.442	0.676

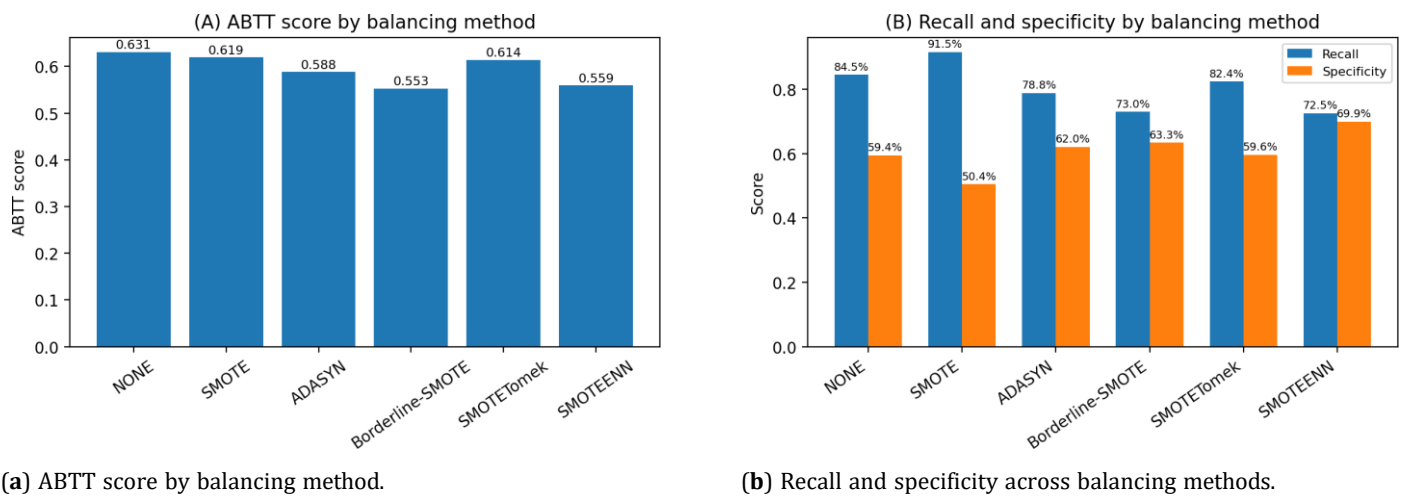


Figure 6. ABTT balancing behavior. (A) ABTT score by balancing method. (B) Recall and specificity across balancing methods.

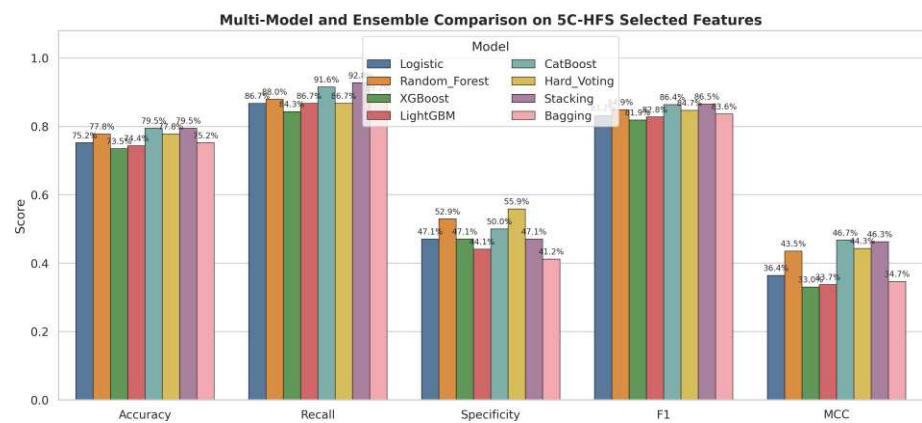
As shown in Figure 6, the selected SMOTE configuration achieved cross-validated recall of 91.5%, specificity of 50.4%, MCC of 0.508, and GMean of 0.638. After balancing selection, the validation stage selected an operating threshold satisfying the screening constraint, and the final CatBoost model was evaluated on the sealed test set using a threshold of 0.30.

4.5. Final Model Performance

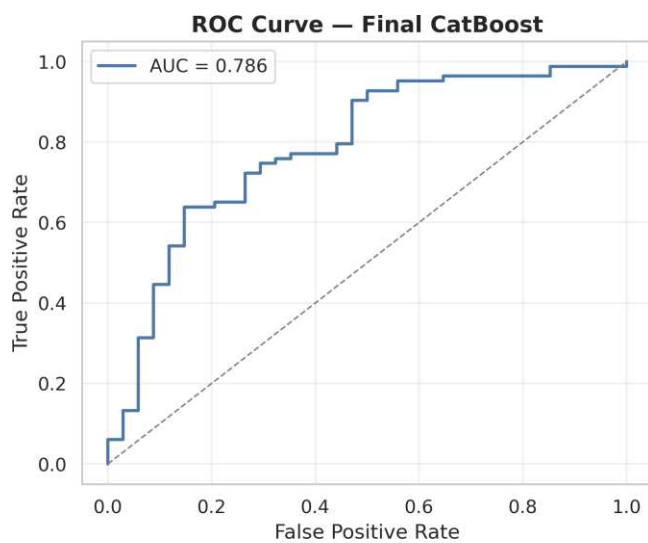
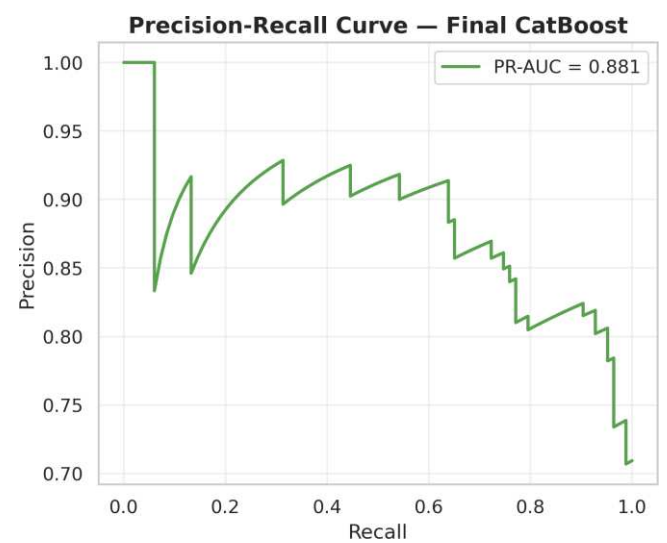
The final model evaluation, summarized in Table 10 and Figure 7, compared five individual learners and three ensemble variants using the 5C-HFS-selected features and SMOTE balancing. CatBoost achieved the best overall clinically relevant trade-off among the individual models, with 79.5% accuracy, 91.6% recall, 50.0% specificity, 86.4% F1-score, 0.467 MCC, 0.786 ROC-AUC, and 0.881 PR-AUC. Stacking achieved slightly higher recall at 92.8%, but CatBoost was retained as the final model because it provided a stronger balance of recall, specificity, MCC, model simplicity, and deployability in the interface prototype.

Table 10. Final individual and ensemble model comparison on the sealed test set.

Model	Acc.	Recall	Spec.	Prec.	F1	MCC	ROC-AUC	PR-AUC
Logistic	75.2%	86.7%	47.1%	80.0%	83.2%	0.364	0.817	0.928
Random Forest	77.8%	88.0%	52.9%	82.0%	84.9%	0.435	0.838	0.928
XGBoost	73.5%	84.3%	47.1%	79.5%	81.9%	0.330	0.792	0.888
LightGBM	74.4%	86.7%	44.1%	79.1%	82.8%	0.337	0.792	0.894
CatBoost	79.5%	91.6%	50.0%	81.7%	86.4%	0.467	0.786	0.881
Hard Voting	77.8%	86.7%	55.9%	82.8%	84.7%	0.443	0.803	0.885
Stacking	79.5%	92.8%	47.1%	81.1%	86.5%	0.463	0.815	0.913
Bagging	75.2%	89.2%	41.2%	78.7%	83.6%	0.347	0.821	0.919

**Figure 7.** Multi-model and ensemble comparison using the selected 5C-HFS features.

The diagnostic performance of the final CatBoost model is further illustrated by the receiver operating characteristic (ROC) curve and the precision–recall curve, as shown in Figure 8.

**(a)** ROC curve.**(b)** Precision–recall curve.**Figure 8.** Diagnostic curves for the final CatBoost model. **(A)** ROC curve. **(B)** Precision–recall curve.

4.6. Baseline vs. Proposed Framework

As summarized in Table 11, the proposed framework was compared with a baseline CatBoost model trained using all engineered features, no balancing, and the default threshold of 0.50. The proposed pipeline improved accuracy from 73.5% to 79.5%, recall from 90.4% to 91.6%, specificity from 32.4% to 50.0%, F1-score from 82.9% to 86.4%, and MCC from 0.280 to 0.467. The confusion matrices show that the proposed framework reduced false negatives from 8 to 7 and reduced false positives from 23 to 17, which is important for clinical screening. Figure 9 compares the baseline CatBoost model with the proposed framework in terms of evaluation metrics and confusion matrices, highlighting improvements in overall classification performance and the reduction of false-positive and false-negative predictions.

Table 11. Baseline CatBoost versus the proposed framework.

Setting	Features	Bal.	Thr.	Acc.	Recall	Spec.	F1	MCC	Confusion matrix
Baseline	All engineered	NONE	0.50	73.5%	90.4%	32.4%	82.9%	0.280	TN=11, FP=23, FN=8, TP=75
Proposed	5C-HFS selected	SMOTE	0.30	79.5%	91.6%	50.0%	86.4%	0.467	TN=17, FP=17, FN=7, TP=76

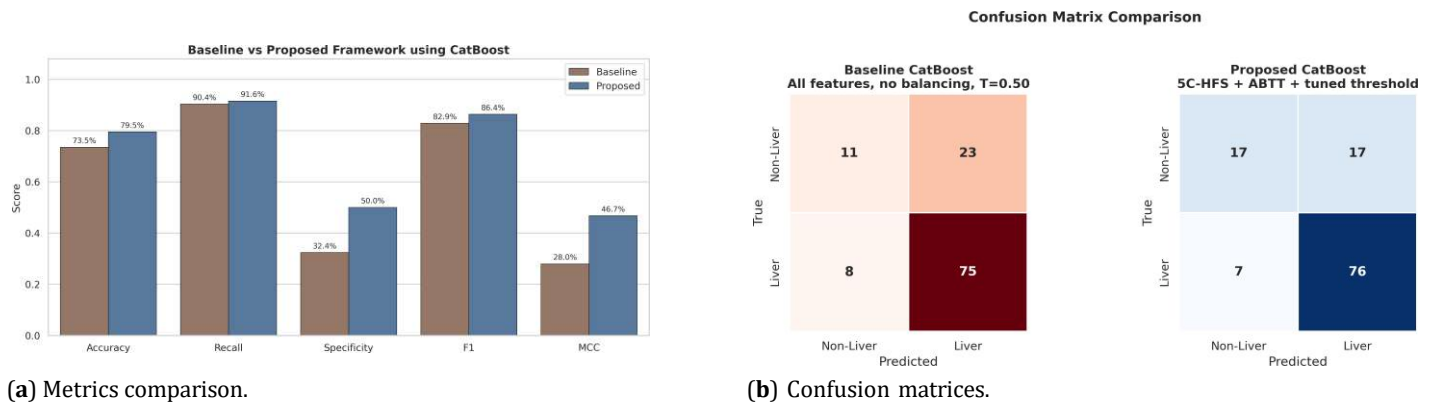


Figure 9. Comparison between the proposed framework and the baseline. (A) Metric comparison. (B) Confusion matrix comparison between baseline and proposed CatBoost models.

4.7. Ablation Study

Table 12 summarizes the ablation study, which isolated the effect of the main framework components. Feature selection alone improved specificity and MCC compared with the baseline. Threshold tuning further improved balanced behavior. The full proposed framework achieved the best overall result, with the highest recall, F1-score, and MCC among the tested configurations. The setting using 5C-HFS and ABTT-selected balancing with a fixed 0.50 threshold reduced recall, confirming that balancing and threshold tuning should be selected jointly rather than applied independently.

Table 12. Ablation study results.

Experiment	Bal.	Thr.	Acc.	Recall	Spec.	F1	MCC
A1 Baseline	NONE	0.50	73.5%	90.4%	32.4%	82.9%	0.280
A2 5C-HFS only	NONE	0.50	75.2%	89.2%	41.2%	83.6%	0.347
A3 5C-HFS + threshold tuning	NONE	0.53	76.9%	89.2%	47.1%	84.6%	0.401
A4 5C-HFS + ABTT, T=0.50	SMOTE	0.50	71.8%	78.3%	55.9%	79.8%	0.334
A5 Full proposed	SMOTE	0.30	79.5%	91.6%	50.0%	86.4%	0.467

4.8. Comparison with Previous ILPD Studies

To contextualize the final performance, Table 13 compares the proposed framework with selected ILPD-based studies whose reported accuracy was lower than the final ac-

curacy achieved in this study [4,27,28]. This comparison is used only as a contextual benchmark because previous studies may differ in preprocessing, validation strategy, model selection, and evaluation protocol.

Table 13. Accuracy comparison between the proposed framework and selected ILPD studies.

Study	Year	Dataset	Best reported method	Reported accuracy
Gupta et al. [27]	2022	ILPD	Random Forest / AD Boosting Tree /	63.00%
			LightGBM	
Altaf et al. [28]	2022	ILPD	Voting Classifier	73.56%
Rehman et al. [4]	2024	ILPD	MaLLiDD framework	76.56%
Proposed framework	2026	ILPD	5C-HFS + ABTT + CatBoost	79.50%

The proposed framework achieved 79.5% accuracy, exceeding the selected ILPD-based studies while maintaining a recall of 91.6%. This should not be interpreted as a universal superiority claim because the compared studies may use different experimental protocols; rather, it indicates that the proposed pipeline is competitive within ILPD literature while also emphasizing false-negative reduction.

4.9. LDPD Validation and Generalization

To examine whether the proposed adaptive design can generalize beyond ILPD, the framework was evaluated on LDPD, as summarized in Table 14. The framework was also applied to the Liver Disease Patient Dataset (LDPD), a larger Kaggle dataset containing approximately 30K training records [29]. Because the Kaggle test portion is unlabeled, the reported results are based on internal stratified evaluation on the cleaned labeled portion of the dataset. This experiment was used as a generalization check rather than as a replacement for external clinical validation.

Table 14. LDPD validation and generalization summary.

Item	Value
Dataset	LDPD (Kaggle Liver Disease Patient Dataset)
Original labeled rows	30,691
Cleaned labeled rows	19,364
Unlabeled test rows	2,110
Selected features	De_Ritis_Ratio; Liver_Stress_Score; ALT; Albumin_Globulin_Ratio; Total_Bilirubin
Required original tests	ALT; AST; Albumin; Albumin_Globulin_Ratio; Total_Bilirubin
ABTT-selected balancing	NONE
Operating threshold	0.53
Accuracy	99.20%
Recall	99.49%
Specificity	98.47%
F1-score	99.44%
MCC	0.980
ROC-AUC / PR-AUC	0.997 / 0.998
Confusion matrix	TN=1094, FP=17, FN=14, TP=2748

Table 15. ABTT balancing comparison on LDPD.

Method	ABTT score	Recall	Specificity	MCC	Mean threshold
NONE	0.989	99.49%	98.68%	0.982	0.524
SMOTE	0.988	99.53%	98.53%	0.981	0.411
ADASYN	0.988	99.44%	98.56%	0.980	0.384
Borderline-SMOTE	0.988	99.43%	98.65%	0.981	0.457
SMOTETomek	0.989	99.28%	99.10%	0.981	0.616
SMOTEENN	0.986	99.25%	98.62%	0.977	0.446

The LDPD experiment supports the adaptive logic of ABTT, with the balancing comparison reported in Table 15. Unlike the ILPD experiment, where SMOTE was selected, ABTT selected no resampling for LDPD, with an operating threshold of 0.53. This difference is important because it shows that the balancing strategy is not fixed in advance; instead, ABTT adapts to the dataset structure and may select either resampling or no resampling depending on which option satisfies the clinical screening objective most consistently.

4.10. SHAP Explainability

SHAP was used to interpret the final CatBoost model. Table 16 reports the global SHAP feature importance, while Figure 10 presents both the mean absolute SHAP importance and the beeswarm analysis. Global SHAP ranking identified Liver_Stress_Score as the most influential feature, followed by Age, Bilirubin_Ratio, Alkaline_Phosphatase, and ALT. This ordering is clinically plausible because the final model relies on liver enzyme stress, bilirubin behavior, patient age, and enzyme-related indicators. The SHAP beeswarm plot also shows that feature effects vary across patients, supporting patient-level explanation rather than relying only on global feature importance.

Table 16. SHAP feature importance for the final CatBoost model.

Feature	Mean(SHAP value)
Liver_Stress_Score	0.102
Age	0.091
Bilirubin_Ratio	0.074
Alkaline_Phosphatase	0.072
ALT	0.071

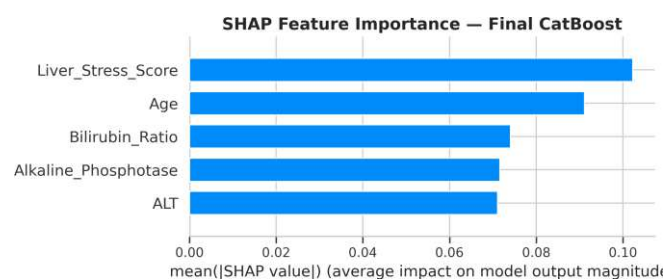
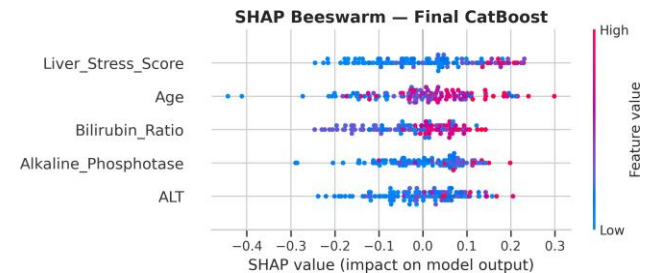
**(a)** SHAP feature importance.**(b)** SHAP beeswarm.

Figure 10. SHAP explainability for the final CatBoost model. **(A)** Mean absolute SHAP feature importance. **(B)** SHAP beeswarm plot showing feature effects across test samples.

5. Discussion

Overall, the results show that the proposed framework improved screening performance compared with a baseline model trained on all engineered features. The main gain was not only higher accuracy, but also a more clinically balanced decision profile: recall remained high, specificity increased substantially, and MCC improved from 0.280 to 0.467.

This suggests that 5C-HFS and ABTT helped reduce unnecessary feature complexity while improving the balance between detecting liver cases and identifying non-liver cases.

The ablation study provides an important explanation of the framework behavior. Feature selection improved the baseline by reducing the input representation to a more focused subset, while threshold tuning improved the classifier operating point. However, applying the ABTT-selected balancing configuration with a fixed threshold reduced recall, showing that resampling and threshold tuning should not be treated as independent steps. This supports the proposed joint design: balancing changes the training distribution, and the tuned threshold determines how predicted probabilities are converted into screening decisions.

CatBoost was selected as the final model even though Stacking achieved slightly higher recall. This choice was justified because CatBoost provided a stronger overall screening trade-off, including better specificity than Stacking, the highest MCC among the tested models, and a simpler deployment pathway for the interface prototype. In screening applications, the best model is not necessarily the model with the highest recall alone; it must also preserve acceptable specificity and stable balanced performance.

The LDPD generalization experiment further supports the adaptive premise of the framework. In ILPD, ABTT selected SMOTE, whereas in LDPD it selected no resampling. This contrast is valuable because it demonstrates that the framework does not assume one imbalance-handling method is universally optimal. Instead, ABTT evaluates the available strategies and selects the method and operating threshold that best fit the dataset under clinical constraints.

The selected features are clinically interpretable. Liver_Stress_Score combines enzyme elevation and albumin status, while Bilirubin_Ratio captures bilirubin composition. Retaining ALT, alkaline phosphatase, and age suggests that the final model preserved both biochemical and demographic information. SHAP results support this interpretation by identifying the same selected features as the major drivers of final CatBoost predictions.

Despite these improvements, limitations remain. The final specificity of 50.0% indicates that the model still produces false positives, which may be acceptable for the early-detection objective but is not sufficient for stand-alone diagnosis. ILPD is also relatively small and contains class overlap, missing values, skewed biomarkers, and correlated variables. The LDPD experiment supports generalization at the dataset level, but it does not replace prospective clinical validation. Therefore, the current framework should be interpreted as an early-detection decision-support prototype rather than a clinically approved diagnostic system.

6. Conclusions and Future Work

This study proposed an adaptive machine learning framework for liver disease prediction. The framework integrates clinical feature engineering, 5C-HFS feature selection, ABTT adaptive balancing and threshold tuning, multi-model comparison, final CatBoost classification, SHAP explainability, and a lightweight interface prototype. On ILPD, 5C-HFS selected five features requiring seven original clinical tests, while ABTT selected SMOTE as the most suitable balancing method under the clinical eligibility rule. The final CatBoost model achieved 79.5% accuracy, 91.6% recall, 50.0% specificity, 86.4% F1-score, 0.467 MCC, 0.786 ROC-AUC, and 0.881 PR-AUC on the sealed test set.

Compared with baseline CatBoost, the proposed framework improved accuracy, specificity, F1-score, and MCC while preserving high recall. The ablation study confirmed that feature selection, adaptive balancing, and threshold tuning contributed differently to the final classification behavior, and that the full framework produced the strongest overall balance. SHAP analysis showed that final predictions were mainly driven by

Liver_Stress_Score, Age, Bilirubin_Ratio, Alkaline_Phosphatase, and ALT, supporting the clinical interpretability of the selected representation.

Future work should validate the proposed framework using larger multi-center clinical datasets, prospective patient cohorts, and more diverse liver disease subtypes, including richer clinical and omics data sources where available [3]. The 5C-HFS strategy should also be tested on other medical datasets to evaluate whether the five-criteria selection logic generalizes beyond liver disease. Similarly, ABTT should be applied to other imbalanced medical datasets because its adaptive design can compare multiple balancing strategies and automatically select the most suitable configuration rather than assuming that one resampling method is always best.

Future research should also optimize the computational time of ABTT so that balancing comparison and threshold tuning can run faster on larger datasets and in practical clinical workflows. This includes reducing redundant search steps, improving the efficiency of cross-validation, and exploring faster tuning strategies while preserving the clinical eligibility logic. In addition, future work should further develop the interface prototype into a more professional, robust, and hospital-ready decision-support tool, following the growing direction of interpretable and clinically deployable AI frameworks [21,22]. This would require usability testing, model calibration, external validation integration, and workflow testing across hospitals, laboratories, and patient populations.

Declarations

Data Availability Statement: The ILPD dataset is publicly available from the UCI Machine Learning Repository at <https://archive.ics.uci.edu/dataset/225/ilpd+indian+liver+patient+dataset>. The LDPD dataset is publicly available from Kaggle at <https://www.kaggle.com/datasets/abhi8923shriv/liver-disease-patient-dataset>.

Acknowledgments: We would like to thank the Deanship of Scientific Research at Shaqra University for supporting this work.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

Funding Declaration: The authors received no external funding for this research.

Ethics and Consent to Participate declarations: not applicable.

Clinical trial number: not applicable.

References

- Spann, A.; Yasodhara, A.; Kang, J.; Watt, K.; Wang, B.; Goldenberg, A.; Bhat, M. Applying Machine Learning in Liver Disease and Transplantation: A Comprehensive Review. *Hepatology* **2020**, *71*, 1093–1105. <https://doi.org/10.1002/hep.31103>.
- Farahi, R.; Derakhshanfard, N. A Comprehensive Review of the Methods of Diagnosing and Predicting Liver Diseases Using Smart Methods. *Discover Artificial Intelligence* **2025**, *5*, 230. <https://doi.org/10.1007/s44163-025-00483-7>.
- Ghosh, S.; Zhao, X.; Alim, M.; Brudno, M.; Bhat, M. Artificial Intelligence Applied to Omics Data in Liver Disease: Towards a Personalised Approach for Diagnosis, Prognosis and Treatment. *Gut* **2025**, *74*, 295–311. <https://doi.org/10.1136/gutjnl-2023-331740>.
- Rehman, A.U.; Butt, W.H.; Ali, T.M.; Javaid, S.; Almufareh, M.F.; Humayun, M.; Rahman, H.; Mir, A.; Shaheen, M. A Machine Learning-Based Framework for Accurate and Early Diagnosis of Liver Diseases: A Comprehensive Study on Feature Selection, Data Imbalance, and Algorithmic Performance. *International Journal of Intelligent Systems* **2024**, *2024*, 6111312. <https://doi.org/10.1155/2024/6111312>.
- Ganie, S.M.; Dutta Pramanik, P.K.; Zhao, Z. Improved Liver Disease Prediction from Clinical Data through an Evaluation of Ensemble Learning Approaches. *BMC Medical Informatics and Decision Making* **2024**, *24*, 160. <https://doi.org/10.1186/s12911-024-02550-y>.
- Rahman, A.K.M.S.; Shamrat, F.M.J.M.; Tasnim, Z.; Roy, J.; Hossain, S.A. A Comparative Study on Liver Disease Prediction Using

- Supervised Machine Learning Algorithms. *International Journal of Scientific and Technology Research* **2019**, *8*. 589
7. Md, A.Q.; Kulkarni, S.; Joshua, C.J.; Vaichole, T.; Mohan, S.; Iwendi, C. Enhanced Preprocessing Approach Using Ensemble Machine Learning Algorithms for Detecting Liver Disease. *Biomedicines* **2023**, *11*, 581. <https://doi.org/10.3390/biomedicines11020581>. 590
8. Straw, I.; Wu, H. Investigating for Bias in Healthcare Algorithms: A Sex-Stratified Analysis of Supervised Machine Learning Models in Liver Disease Prediction. *BMJ Health and Care Informatics* **2022**, *29*, e100457. <https://doi.org/10.1136/bmjhci-2021-100457>. 593
9. Araf, I.; Idri, A.; Chairi, I. Cost-Sensitive Learning for Imbalanced Medical Data: A Review. *Artificial Intelligence Review* **2024**, *57*, 80. <https://doi.org/10.1007/s10462-023-10652-8>. 596
10. Ramana, B.; Venkateswarlu, N. ILPD (Indian Liver Patient Dataset). UCI Machine Learning Repository, 2022. <https://doi.org/10.24432/C5D02C>. 598
10. Amin, R.; Yasmin, R.; Ruhi, S.; Rahman, M.H.; Reza, M.S. Prediction of Chronic Liver Disease Patients Using Integrated Projection Based Statistical Feature Extraction with Machine Learning Algorithms. *Informatics in Medicine Unlocked* **2023**, *36*, 101155. <https://doi.org/10.1016/j.imu.2022.101155>. 600
11. Raschka, S. Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning. arXiv preprint, 2020. <https://doi.org/10.48550/arXiv.1811.12808>. 602
12. Noor, S.; AlQahtani, S.A.; Khan, S. XGBoost-Liver: An Intelligent Integrated Features Approach for Classifying Liver Diseases Using Ensemble XGBoost Training Model. *Computers, Materials and Continua* **2025**, *83*, 1435–1450. <https://doi.org/10.32604/cmc.2025.061700>. 603
13. Ganji, A.; Usha, D.; Rajakumar, P.S. Enhanced Early Diagnosis of Liver Diseases Using Feature Selection and Machine Learning Techniques on the Indian Liver Patient Dataset. *Scalable Computing: Practice and Experience* **2025**, *26*, 1104–1115. <https://doi.org/10.12694/scpe.v26i3.4254>. 604
14. Anand, L.; Neelananarayanan, V. Feature Selection for Liver Disease Using Particle Swarm Optimization Algorithm. *International Journal of Recent Technology and Engineering* **2019**, *8*, 6434–6439. <https://doi.org/10.35940/ijrte.C3770.098319>. 605
15. Khan, S. Advanced Feature Selection Techniques in Medical Imaging—A Systematic Literature Review. *Computers, Materials and Continua* **2025**, *85*, 2347–2401. <https://doi.org/10.32604/cmc.2025.066932>. 606
16. Mustapha, M.; Abdulganiyu, O.H.; Abubakar, I.N.; Majikumna, K.U.; Suleiman, G.; Ech-chariy, M.; Olivier, M.M. A Focal Loss and Sequential Analytics Approach for Liver Disease Classification and Detection. *Healthcare Analytics* **2025**, *8*, 100424. <https://doi.org/10.1016/j.health.2025.100424>. 607
17. Chowdhury, S.H.; Mamun, M.; Shaikat, M.T.A.; Hussain, M.I.; Iqbal, M.S.; Hossain, M.M. An Ensemble Approach for Artificial Neural Network-Based Liver Disease Identification from Optimal Features through Hybrid Modeling Integrated with Advanced Explainable AI. *Medinformatics* **2025**. <https://doi.org/10.47852/bonviewMEDIN52024744>. 608
18. Almusallam, N.; Khan, S. Chronic Liver Disease Classification Using Deep Learning with SHAP-Optimized Hybrid Features. *iScience* **2025**, *28*, 113972. <https://doi.org/10.1016/j.isci.2025.113972>. 609
19. Mamun, M.; Chowdhury, S.H.; Hossain, M.M.; Khatun, M.R.; Iqbal, S. Explainability Enhanced Liver Disease Diagnosis Technique Using Tree Selection and Stacking Ensemble-Based Random Forest Model. *Informatics and Health* **2025**, *2*, 17–40. <https://doi.org/10.1016/j.infoh.2025.01.001>. 610
20. Long, Z.; Zhang, L. Detection of Hepatocellular Carcinoma Using Optimized miRNA Combinations and Interpretable Machine Learning Models. *IEEE Access* **2025**, *13*, 66078–66093. <https://doi.org/10.1109/ACCESS.2025.3559105>. 611
21. Alqaralleh, B.A.Y.; Alksasbeh, M.Z.; Kulakli, A.; Zreikat, A.I. An Intelligent Healthcare Framework for Hepatocellular Carcinoma Diagnosis Based on Aggregated Learners from Biomedical Data Utilising Explainable Artificial Intelligence. *Scientific Reports* **2026**, *16*, 9357. <https://doi.org/10.1038/s41598-026-39871-z>. 612
22. Noor, S.; AlQahtani, S.A.; Khan, S. Chronic Liver Disease Detection Using Ranking and Projection-Based Feature Optimization with Deep Learning. *AIMS Bioengineering* **2025**, *12*, 50–68. <https://doi.org/10.3934/bioeng.2025003>. 613
23. Joloudari, J.H.; Saadatfar, H.; Dehzangi, A.; Shamshirband, S. Computer-Aided Decision-Making for Predicting Liver Disease Using PSO-Based Optimized SVM with Feature Selection. *Informatics in Medicine Unlocked* **2019**, *17*, 100255. <https://doi.org/10.1016/j.imu.2019.100255>. 614
24. Shi, Z.; Huang, L.; Wang, H. Predictive Modeling of Pediatric Drug-Induced Liver Injury: Dynamic Classifier Selection with Clustering Analysis. *Digital Health* **2025**, *11*, 20552076251330078. <https://doi.org/10.1177/20552076251330078>. 615
25. El Atifi, W.; El Rhazouani, O.; Khan, F.M.; Sekkat, H. Optimizing Ensemble Machine Learning Models for Accurate Liver Disease Prediction in Healthcare. *PLOS ONE* **2025**, *20*, e0330899. <https://doi.org/10.1371/journal.pone.0330899>. 616
26. Gupta, K.; Jiwani, N.; Afreen, N.; D, D. Liver Disease Prediction Using Machine Learning Classification Techniques. In Proceedings of the 2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT), Indore, India, 2022; pp. 221–226. <https://doi.org/10.1109/CSNT54456.2022.9787574>. 617
27. Altaf, I.; Butt, M.A.; Zaman, M. Hard Voting Meta Classifier for Disease Diagnosis Using Mean Decrease in Impurity for Tree Models. *Journal of Social Sciences Advancement* **2022**, *9*, 71–82. <https://doi.org/10.18488/76.v9i2.3037>. 618
28. Shrivastava, A. Liver Disease Patient Dataset 30K Train Data. Kaggle, 2020. Accessed: 2026-05-06. 619