IUST
Iran University of Science and Technology

Iranian Journal of Electrical and Electronic Engineering

Journal Homepage: ijeee.iust.ac.ir

Enhancing Hepatitis C Diagnosis: The Impact of SMOTE, Optuna, and SHAP on Detection Methods

Mehzabeen S.M*,Gayathri R*, Pattunnarajam Paramasaivam*(C.A.) and Ramya A*

Abstract: Hepatitis C virus (HCV) detection is a critical aspect of early intervention and effective management of the disease. This paper presents a comprehensive study focused on enhancing the detection accuracy of HCV through the integration of advanced techniques - SMOTE, Optuna, and SHAP - alongside extensive exploratory data analysis (EDA). The study addresses class imbalance using Synthetic Minority Over-sampling Technique (SMOTE), optimizes model performance with Optuna for hyperparameter tuning, and provides model interpretability using SHAP (SHapley Additive exPlanations). EDA is leveraged to gain valuable insights into the dataset's characteristics, ensuring robust data preprocessing and feature engineering. The results show 97% improved HCV detection performance, highlighting the efficacy of the proposed methodology in medical diagnostics and aiding healthcare professionals in making informed clinical decisions.

Keywords: Hepatitis C virus, Synthetic Minority Over-sampling Technique, exploratory data analysis, SHapley Additive exPlanations, machine learning, classification algorithms, OPTUNA.

1 Introduction

THE hepatitis C virus causes hepatitis C, an RNA virus that mostly harms the liver (HCV). It is one of the main viral hepatitis strains and is regarded as a global public health issue. The main way that the virus is transmitted is through contact with an infected person's blood, most frequently when sharing needles when using drugs, getting contaminated medical care, or from infected mothers to their newborns while giving birth. According to estimates, this condition has a persistent impact on more than 150 million people globally. In data preprocessing data normalization has been done to replace the missing values of the dataset with mode values based on age attribute. Python and R are the two machine learning tools used to rescale the variables. A 1385-instance dataset with 29 attributes was used to test the classification model [1-2].

Iranian Journal of Electrical & Electronic Engineering, 2025.
Paper first received 20 Sep. 2024 and accepted 03 Apr. 2025.
* Department of Electronics and Communication Engineering, Sri Venkateswara College of Engineering, Sriperumbudur.
Email: mehzabeen@svce.ac.in, rgayathri@svce.ac.in, pattunarajamp@svce.ac.in, ramyaa@svce.ac.in
Corresponding Author: Pattunnarajam Paramasaivam.

Ruzicka *et.al* [3] investigated a thorough understanding of systemic symptoms in Japanese chronic HCV patients, and extended their research into various extrahepatic manifestations. Data analysis using R-based CARET and Python-based Scikit learn, and seven machine learning techniques and feature selection algorithms were discussed Kashif *et.al in [4].* Yang and Shami have been discussed about the merits of the ensemble model, adaboost and Bagging which serves as the more suitable classifiers among the five models [5]. The majority of HCV infections develop into chronic conditions, in which the virus stays in the body for a longer period of time and frequently causes permanent liver damage. Hepatitis C chronic infection can advance covertly for years without showing any signs. To detect Hepatitis C, several authors et al. [11],[13],[22] have suggested the first screening test as 'HCV antibody test' which is a blood test which tests the levels of antibodies present in the blood. In recent years, the use of artificial intelligence and machine learning in the medical field has shown tremendous potential, notably in the areas of disease diagnosis and prognostic modeling. These cutting-edge methods have the potential to have a big impact on healthcare by offering precise, quick, and affordable diagnostic options. In

order to improve the identification of hepatitis C disease, several authors *et. al*. [6-10] have suggested harnessing the potential of cutting-edge approaches such as Synthetic Minority Over-sampling Technique (SMOTE), Optuna [12], and SHAP (SHapley Additive exPlanations). Increasing the number of instances of a specific attribute in the dataset can be done mathematically using SMOTE. This imbalance can affect the machine learning model, leading to a lower accuracy and hindering the performance of the model. With using SMOTE we can ensure that there is a balanced dataset that is being used and hence improving the performance of the model. Building powerful machine learning models requires careful consideration of the hyperparameter tuning process. The choice of the best hyperparameters has a significant impact on the model's performance, yet manually investigating every combination is time- and resource-intensive. Enter Optuna, an automated framework for hyperparameter optimization that uses cutting-edge algorithms to find the optimal hyperparameter configuration. The hepatitis C detection model's accuracy and resilience by using Optuna has been optimized in our research. It's important to interpret machine learning models, especially in the medical industry where trust-building and clinical decision-making depend on openness. By assigning feature priority, the advanced model interpretability technique SHAP offers illuminating justifications for certain predictions. By examining the contribution of each feature to the model's output, healthcare professionals can gain a better understanding of the factors affecting the identification of hepatitis C disease. This allows them to obtain useful insights into the diagnostic process.

This study aims to enhance hepatitis C disease detection by employing SMOTE, Optuna, and SHAP techniques in the machine learning process. Section 2 reviews relevant literature on hepatitis C virus detection using machine learning models. Section 3 presents exploratory data analysis of the dataset, while Section 4 discusses the machine learning flow with the proposed methods. In Section 5, Results of performance of the classical model. Section 6 represents the conclusion of findings and potential implications for medical diagnostics.

2 LITERATURE SURVEY

Ahmed M. Elshewey *et al*. [14] have proposed the hyOPTGB Model for hepatitis C prediction. The hyOPTGB is an hyperparameter optimized Gradient Boosting Model in which 8 specific hyperparameters of Gradient Boosting are optimized. The dataset is preprocessed using Min- Max normalization followed by feature selection using Forward selection wrapped method. For the same dataset different machine learning models such as SVM, DT, DC, BC and RC were evaluated based on their accuracy, F-1 score, recall and precision and their performance is compared with hyOPTGB model, where the proposed hyOPTGB model outperforms with 95.3% accuracy.

Ali Mohd Ali *et al*. [15] implemented various machine learning models for comprehensive evaluation of their effect in predicting hepatitis C.Sequential Forward Selection (SFS) is used in the suggested framework to separate the most important attributes from the rest. Investigation into the effect of the synthetic minority oversampling technique (SMOTE) on accuracy led to the conclusion that the SMOTE had little to no impact on the models' accuracy. When machine learning models like LR, KNN, DT, NN, and RF were employed on the dataset, an average of 83% accuracy was attained. The predictions of the machine learning models are interpreted using the Shapley Additive Explanations (SHAP) approach.

Hashem *et al*. [16] have compared the performance between multiclass and binary class labels using an Egyptian patient’s dataset and highlights the impact of label categorization on model accuracy and predictive capability. Edeh *et al.* [17] have proposed an AI based ensemble model after examining various machine learning models for predicting hepatitis. The suggested approach could predict increasing fibrosis using clinical information and blood biomarkers. It has been discovered that individual models are capable of delivering accuracy of up to 94.67%. The next step was to develop the ensemble model, which consists of a Bayesian network, MLP, and QUEST decision trees. The Ensemble node combines three model nuggets (MLP, Bayesian Network, and QUEST) to produce predictions that are more accurate than any of the individual models. By merging predictions from various models, limitations in the MLP, Bayesian Network, and QUEST models were eliminated, leading to a higher overall accuracy. This combination of MLP, Bayesian Network, and QUEST models typically outperforms the top MLP, Bayesian Network, and QUEST models, if not better. The obtained accuracy was 94.10%.

Several authors *et al.* [18], [19] have suggested an artificial intelligence (AI) algorithm was presented to identify the stage of liver fibrosis in patients. The researchers looked at the medical records of 1240 people with chronic viral hepatitis C, and they used data from 689 patients who were divided into stages of liver fibrosis to build machine learning models. The established method for diagnosing the 3-4 stages of liver fibrosis in patients with chronic viral hepatitis C has an accuracy of 80.56% (95% CI: 69.53-88.94%), sensitivity of 66.67%, and specificity of 94.44% when compared to the "gold standard" of diagnosis (liver biopsy).

3 EXPLORATORY DATA ANALYSIS

The dataset used in this study includes demographic information, including age, as well as test results from Hepatitis C patients and blood donors. The data was sourced from the esteemed University of California, Irvine Machine Learning Repository [21].

Figure 1 depicts the percentage of blood donors who have had their health checked is displayed in a pie chart. Blood donors who have not been screened for any of the conditions make up the largest group, at 86.7%. Blood donors who have had cirrhosis testing make up the next-highest percentage, 4.9%. For blood donors who have undergone tests for fibrosis, hepatitis, and questionable blood donors, the remaining percentages apply. There are a total of 31 missing values in this dataset. Under analysis it is found that ALP and CHOL contribute the most to the missing values in the dataset.

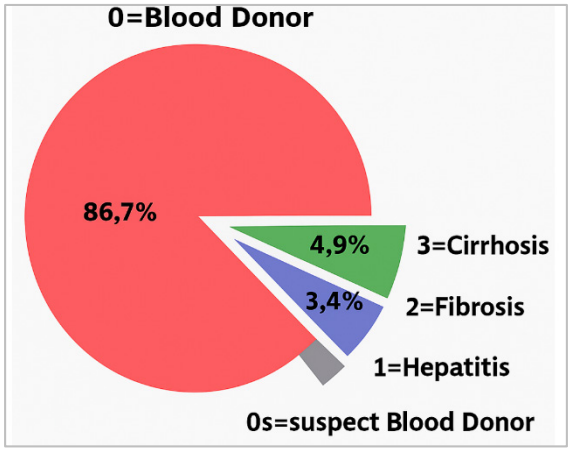

Fig 1. Proportion of Blood Donors and Hepatitis C-Related Conditions in the Dataset

3.1 Univariate Data Analysis

Univariate data analysis is a fundamental statistical approach used to examine and summarize individual variables in a dataset. It provides essential insights into the distribution, central tendency, and variability of each factor, facilitating a deeper understanding of their individual effects before proceeding with multivariate modeling. In the context of medical diagnostics, univariate analysis plays a crucial role in identifying patterns, outliers, and potential correlations within patient data. In this chapter the study focuses on the univariate analysis of critical biochemical and demographic factors associated with Hepatitis C diagnosis, including Age, Albumin Level, Alkaline Phosphatase Level, Alanine Transaminase Level, and Bilirubin Level. Each of these parameters provides significant clinical insights: **Age** influences disease progression and treatment outcomes. Albumin Level reflects liver function and protein synthesis capability. Alkaline Phosphatase (ALP) Level serves as a marker for liver or bile duct abnormalities. Alanine Transaminase (ALT) Level indicates liver cell damage and inflammation. Bilirubin Level assesses liver's ability to process waste, often signaling hepatic dysfunction.

A.Age

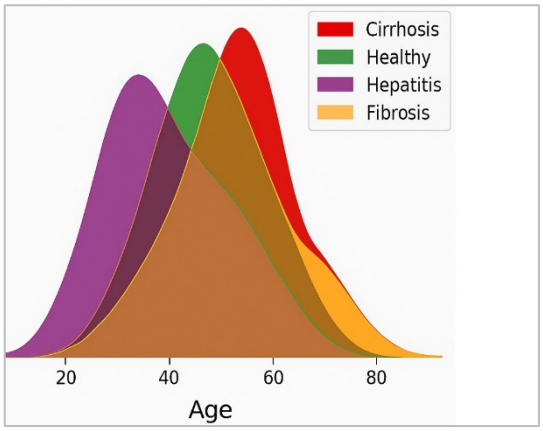

Fig 2. Prevalence of Hepatitis in Young Populations

Figure 2 illustrates the distribution of hepatitis rates among the young population by age and liver status. The overall age distribution demonstrates that the highest prevalence of hepatitis is in the population aged 20-30 years old. The liver is inflamed by hepatitis. When bodily tissues are harmed or diseased, inflammation with swelling occurs. The age distribution by liver status indicates that the bulk of people with hepatitis are healthy, but there is a larger population with fibrosis and cirrhosis. This is because hepatitis can damage the liver, and if the damage is severe enough, it can lead to fibrosis and cirrhosis. Fibrosis is a criterion where scar tissue builds up in the liver, and cirrhosis is a more sophisticated phase of liver damage where the liver becomes scarred and incapable to function properly.

B.Albumin Level

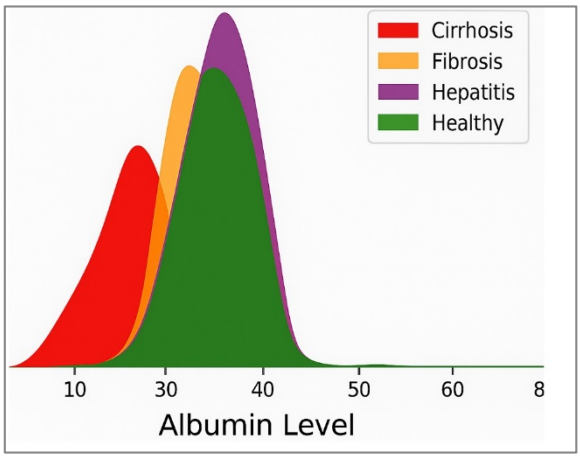

Fig 3. Albumin Level Variations in Liver Disease and Healthy Individuals

Figure 3 illustrates the distribution of albumin levels in populations with different liver status. The albumin level is a quantifier of the number of albumins in the blood. Albumin is a protein produced by the liver. The chart indicates that people with a healthy liver have higher rates of albumin than the population with liver damage. This is because the liver is responsible for generating albumin. When the liver is damaged, it can no longer generate as much albumin, which can lead to low albumin levels. The graph also indicates that the intensity of liver damage is correlated to the level of albumin deficiency. People with fibrosis have lower levels of albumin than people with healthy liver. People with cirrhosis have the lowest albumin levels of all. The above figure indicates that the average albumin level for population with healthy livers is 40 g/dL. The graph also illustrates that the average albumin level for people with fibrosis is 30 g/dL. The chart indicates that the average albumin level for the population with cirrhosis is 20 g/dL.

C.Alkaline Phosphatase Level

Figure 4 illustrates the distribution of alkaline phosphatase (ALP) levels in populations with different liver status. ALP is an enzyme fabricated by the liver. It helps break down fats and proteins. When the liver is damaged, it can no longer generate as much ALP, which can lead to low ALP levels. The graph also indicates that the severity of liver damage is correlated to the level of ALP deficiency. People with fibrosis have lower ALP levels than people with healthy livers. Under survey, people with liver cirrhosis has the lowest ALP values.

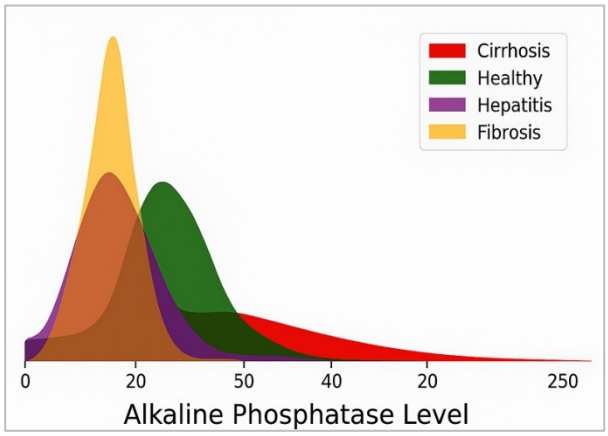

Fig 4. Density Distribution of Alkaline Phosphatase in Liver Disorders

D.Alanine Transaminase Level

Figure 5 illustrates the distribution of alkaline transaminase (ALT) levels by liver status. ALT is an enzyme generated in the liver. This enzyme aids the liver's process of converting food into energy. The quantity of ALT in the blood may increase in cases of liver injury. The above figure illustrates three different liver conditions: healthy, hepatitis, and cirrhosis. The healthy range for ALT levels is between 0 and 50 IU/L. ALT levels above 50 IU/L can signify a liver problem. The overall ALT distribution is higher in people with healthy livers, while the distribution by liver status is lower in people with hepatitis, fibrosis, and cirrhosis.

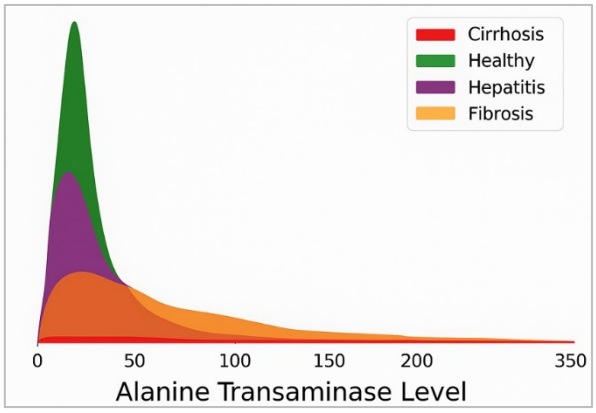

Fig 5. ALT Level Distribution in Healthy vs. Liver Disease Patient

E.Aspartate Aminotransferase Level

Figure 6 illustrates the distribution of aspartate aminotransferase (AST) levels by liver status. As soon as the liver is harmed, the enzyme AST is created there and secreted into the bloodstream. The graph illustrates three different liver conditions: healthy, hepatitis and cirrhosis. The healthy range for AST values is between 0 and 40 IU/L. AST levels above 40 IU/L can signify a liver problem. In people with hepatitis, most AST values are between 40 and 300 IU/L. In people with liver cirrhosis, most AST values are above 300 IU/L.

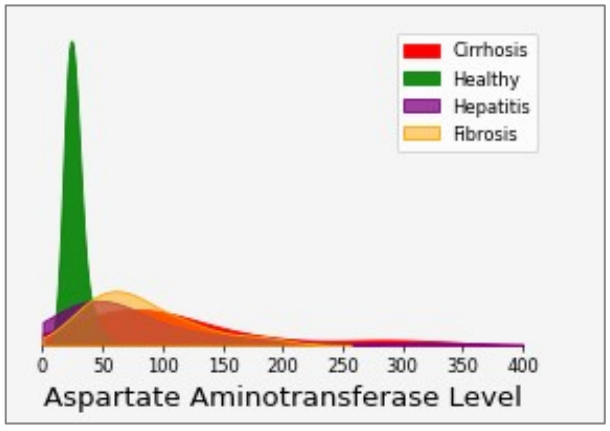

Fig 6. Impact of Liver Disorders on AST Levels

F.Bilirubin Level

Figure 7 illustrates the distribution of bilirubin levels by liver status. A consequence of the destruction of red blood cells is bilirubin. The level of bilirubin in the blood rises when the liver is not functioning properly because it cannot remove bilirubin from the blood. The

chart indicates that people with healthy liver have a bilirubin range of 0-1.5 mg/dL. As the liver disease progresses, the bilirubin level increases. People with hepatitis have a bilirubin range of 1.6-3 mg/dL, people with fibrosis have a bilirubin range of 3.1-5 mg/dL, and people with cirrhosis have a bilirubin range of >5 mg/dL.

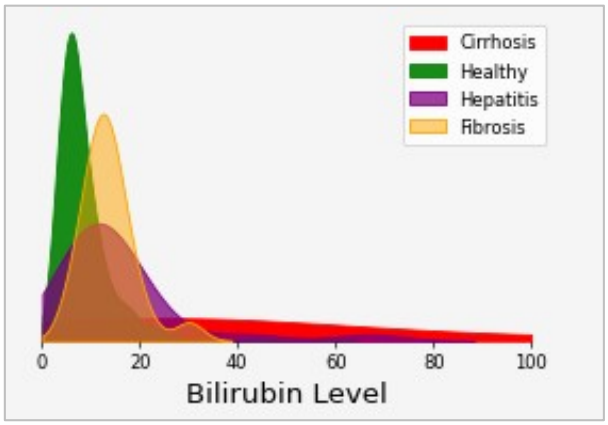

Fig 7. Bilirubin Trends in Cirrhosis, Hepatitis, Fibrosis, and Healthy Cases

G.Creatinine Level

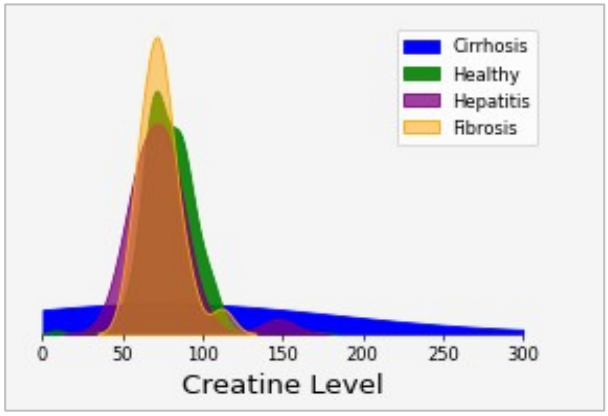

Fig 8. Creatine Level Variations in Healthy and Liver Disease Patients

Figure 8 shows the distribution of creatine levels in people with different liver statuses. The graph adopts a bar chart to show the distribution of creatine levels by liver status. The bars are colored to signify the different liver statuses: healthy (green), fibrosis (yellow), cirrhosis (red), and hepatitis (blue). The graph indicates that the average creatine level for people with healthy livers is 100 mg/dL. The graph also demonstrates that the creatine levels are distributed in a bimodal manner, with two separate peaks. This is probably because nutrition and genetic factors, which are two separate sources of creatine elevation, are involved. *Dietary variables contribute to the peak at the lower creatinine levels, whereas genetic factors contribute to the peak at the higher creatinine levels.*

H.Protein Level

Figure 9 shows a diagram of protein distribution by liver status. *The data shows that there is no clear correlation between protein levels and liver status.* The overall protein distribution is relatively uniform, with a slight increase in protein levels in people with hepatitis and cirrhosis. However, there is a wide range of protein levels in people with all liver statuses, and there are many people with healthy livers who have low protein levels.

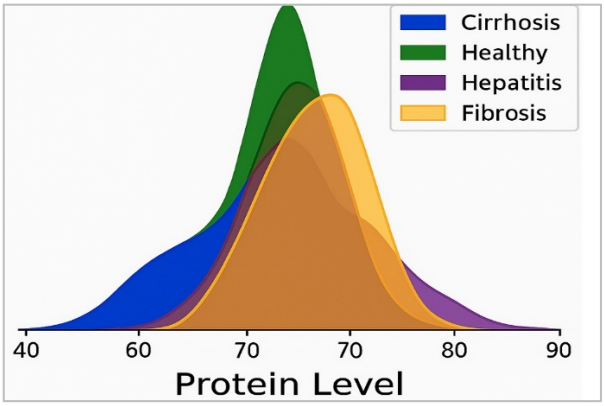

Fig 9. Protein Level Indicator

3.2 Bivariate Data Analysis

Bivariate analysis has been considered and compared with 2 attributes against each other [20]. In this analysis "ALT vs. AST" and "ALP vs. CHE" together have an impact on the dataset.

Alanine Transaminase (ALT) and Aspartate Transaminase (AST) are key liver enzymes used as biomarkers for liver function and damage. Bivariate analysis of ALT vs. AST helps in assessing liver disease severity, differentiating between various hepatic conditions, and predicting disease progression. While there was a comparison between each attribute in the dataset directly with the disease [20] and the relation it presented, in this bivariate analysis we are going to compare 2 attributes against each other. Bivariate analysis has been considered as useful way to identify relationships between variables that might not be immediately obvious. In this analysis "ALT vs. AST" and "ALP vs. CHE" together have an impact on the dataset.

ALT vs. AST

Figure 10 shows that people with healthy livers have relatively low levels of ALT and AST. However, people with hepatitis C have much higher levels of ALT and AST. This is because the hepatitis C virus damages liver cells, which releases ALT and AST into the bloodstream. But, predominantly a very high AST is seen to be common among people who have been affected by hepatitis C virus.

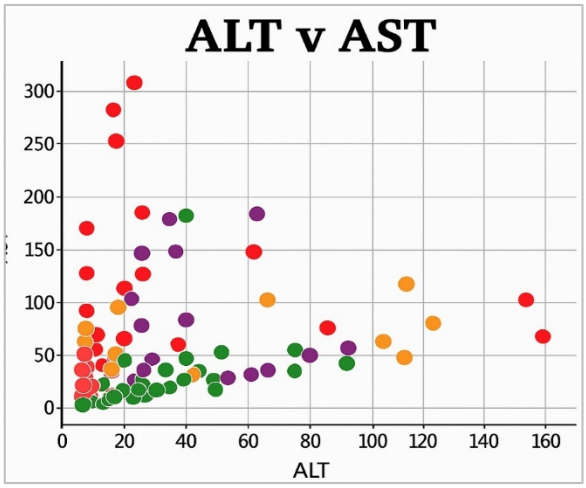

Fig 10. Comparative Analysis of ALT and AST Enzymes in Liver Health

ALP vs CHE

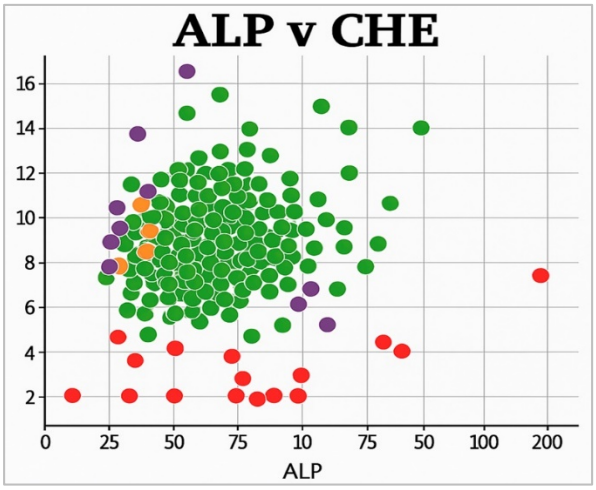

Fig 11. Exploring the Link Between ALP and CHE Levels

Figure 11 depicts the comparison between alkaline phosphatase (ALP) levels and cholinesterase (CHE) levels. It has been observed that low ALP levels and high CHE levels in a patient's test results may indicate a hepatitis C virus infection. In contrast, healthy individuals typically exhibit medium levels of both substances.

4 METHODOLOGY

In this study, we employ various machine learning techniques to analyze and classify Hepatitis C-related data. These methods range from traditional statistical models to advanced machine learning algorithms, each offering unique strengths in handling different data characteristics. The following classification techniques are explored:

Each of these models is evaluated based on classification performance metrics, ensuring a comprehensive comparison of their effectiveness in Hepatitis C diagnosis.

4.1 SMOTE

Synthetic Minority Over-sampling Technique is abbreviated as SMOTE. It is a machine learning data augmentation strategy for imbalanced datasets. When dealing with imbalanced datasets in which one class is severely under represented compared to others, SMOTE aids in class distribution balance by generating synthetic samples for the minority class. Due to a lack of sufficient samples, the machine learning model may not be able to fully learn from the minority class when SMOTE is not used on an imbalanced dataset. As the model may be skewed towards the dominant class, this could result in biased and erroneous predictions. An example from the minority group is chosen and its k nearest neighbors are determined (usually using Euclidean distance), and then creates new synthetic samples by randomly selecting a neighbor and adding a fraction of the difference between the two samples to the original sample shown in equation(1). This process helps expand the minority class, making it more balanced with the majority class.

$$New\ Sample = Samplei + Randam\_Fraction * (Samplei - Samplej) \quad (1)$$

Where:

- Sample_i is the original sample from the minority class.
- Sample j is one of its k nearest neighbors.
- Random_fraction ranges from 0 to 1 and it is random.

For our dataset we use SMOTE to balance the various classes namely Healthy, Hepatitis, Fibrosis, Cirrhosis

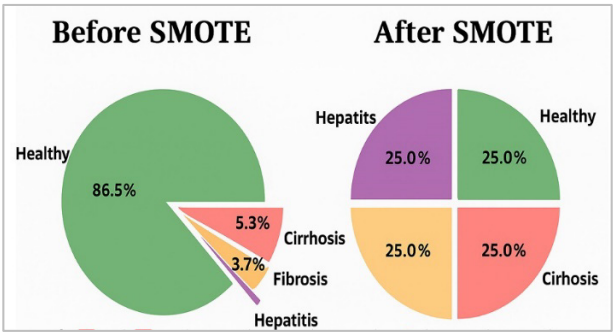

Fig 12. Data distribution with and without SMOTE

Figure 12 depict the distribution of patients with liver diseases before and after SMOTE. The original dataset was imbalanced, with 56.5% of the patients being healthy, 25% having hepatitis, 5.3% having cirrhosis, and 5.3% having fibrosis. SMOTE was used to balance the classes in the dataset by creating synthetic data points that were similar to the existing data points. This

resulted in a more evenly distributed dataset, with 25% of the patients in each class.

Table 1. Model Performance Analysis: Accuracy, F1-Score, and Precision Without SMOTE

Model	Accuracy	F1-Score	Precision
LR	0.903	0.628	0.615
SVM	0.946	0.609	0.736
KNN	0.849	0.605	0.611
Naive- Bayes	0.946	0.695	0.726
RF	0.946	0.653	0.702
MLP	0.956	0.613	0.619

Table 1 shows the accuracy and precision of various machine learning models in the absence of SMOTE. The MLP model is the most accurate, with a score of 0.956, followed by RF, LR, SVM, KNN, and Naive Bayes. At 0.619, the MLP model has the highest precision. While RF and LR are both good performers, they fall short of the MLP model.

Table 2. Model Performance Analysis: Accuracy, F1-Score, and Precision With SMOTE

Model	Accuracy	F1-Score	Precision
LR	0.881	0.614	0.607
SVM	0.946	0.609	0.736
KNN	0.924	0.614	0.669
Naive-Bayes	0.946	0.695	0.726
RF	0.946	0.630	0.684
MLP	0.956	0.713	0.747

Table 2 shows the accuracy, precision and F1-score, and of various machine learning models employing SMOTE (Synthetic Minority Over-sampling Technique). The MLP model has the highest accuracy (0.956), followed by RF, LR, SVM, K-Nearest Neighbors, Naive Bayes, Random Forest, and Multi-layer Perceptron. SMOTE improves the accuracy, f1-score, and precision of these models.

From our results, it is evident that all of the models' accuracy is slightly higher with SMOTE than without SMOTE. With and without SMOTE, the MLP model has the maximum accuracy. With SMOTE, RF and LR accuracy is slightly higher than without SMOTE. SVM, KNN, and Naive Bayes accuracy are not significantly different with and without SMOTE.

In general, total accuracy between with SMOTE and without SMOTE does not differ significantly. However, all models with SMOTE have a minor gain in accuracy. This demonstrates how SMOTE can be used to enhance machine learning models' accuracy on unbalanced datasets.

4.2 Hyperparameter Optimization: A Deep Dive with Optuna

4.2.1 OPTUNA

Optuna is a sophisticated hyperparameter optimization framework that employs state-of-the-art algorithms to effectively explore the hyperparameter space of machine learning and deep learning models. Hyperparameters are configuration parameters that can significantly influence and impact the performance of a model. By automating the process of tuning hyperparameters, Optuna can assist in identifying the optimal collection of hyperparameters to maximize the model's performance metric.

Optuna requires the definition of a search space for hyperparameters. Practitioners specify hyperparameters and their respective distributions, such as continuous parameters with uniform or log-uniform distributions, categorical parameters with predefined choices, and discrete parameters with integer values. This search space serves as the basis for Optuna's exploration of different hyperparameter configurations. Optuna utilizes a Bayesian optimization algorithm to select the hyperparameters for each trial. The Bayesian optimization algorithm uses a probabilistic model to represent the uncertainty about the hyperparameter space. This model is updated as new trials are run, and the algorithm uses this information to select the next set of hyperparameters. Until a stopping requirement, such as a maximum number of trials or period of time, is satisfied, the optimization process is carried out. At the end of the optimization process, Optuna returns the optimal set of hyperparameters.

The Optuna has been used to optimize the hyperparameter. It aids in finding the best hyperparameter which results in optimal performance of the machine learning model. For the prediction of hepatitis C disease, OPTUNA objective function has been created that uses a voting classifier to ensemble six different machine learning models: logistic regression, KNN, SVM, Random Forest, Naive Bayes, and MLP.

The objective function first defines the hyper parameters for each of the seven models.

4.2.1.1 Logistic Regression:

A statistical model is used to determine the likelihood of a binary result, like whether or not a client will click on an advertisement. A sigmoid function is used in the output layer of a linear regression model. The sigmoid function is a nonlinear function that maps real numbers to the interval [0, 1].

This allows the logistic regression model to predict probabilities.

The mathematical equation (2) for the logistic regression model is as follows [23] as,

$$p(y = 1 \mid x) = \frac{1}{(1+e^{(-wx)})} \quad (2)$$

Where:

- p(y = 1 | x) is the probability that the outcome is 1 given the input x
- w is the vector of weights
- x is the vector of features
- exp () is the exponential function

The below are the hyperparameters used in logistic regression:

- ***lr_penalty:*** *Type of regularization penalty ('l1', 'l2', or 'elasticnet') applied to logistic regression.*
- ***lr_solver1 and lr_solver2****: Solvers used for optimization in logistic regression, depending on 'lr_penalty' ('liblinear', 'saga', 'newton-cg', 'lbfgs', or 'sag').*
- ***lr_l1_ratio:*** *Mixing parameter for elasticnet penalty (0 for L2, 1 for L1), used when 'lr_penalty' is 'elasticnet'.*
- ***lr_tol:*** *Tolerance for stopping criteria during optimization (values between 1e-5 and 1e-2).*
- ***lr_C:*** *Inverse of regularization strength (C) for logistic regression (values between 0.0 and 1.0).*

The Logistic Regression achieved an accuracy of 88.17 % with F-1 score and precision being 0.61 and 0.60 respectively.

4.2.1.2 K-Nearest Neighbors (KNN):

Both classification and regression tasks can be performed using this non- parametric machine learning model. The algorithm finds the k training examples that are most like a new data point, and then predicts the class or value of the new data point using the classes or values of the k nearest neighbors. The similarity of two data points can be evaluated using a distance measure. Common distance units include the Minkowski distance, the Manhattan distance, and the Euclidean distance.

The distance metric employed in KNN most frequently is the Euclidean distance. The equation (3) is followed and defined as follows :

$$d(x, y) = \sqrt{\sum(x_i - y_i)^2} \quad (3)$$

Where:

- d(x, y) is the distance between the two data points x and y
- x_i and y_i are the i-th features of the two data points x and y

The Manhattan distance is defined as follows:

$$d(x, y) = \sum(\mid x_i - y_i \mid) \quad (4)$$

The Minkowski distance is defined as follows:

$$d(x, y) = (\sum(\mid x_i - y_i \mid^{p})^{(\frac{1}{p})}) \quad (5)$$

Where:

- p is a parameter that controls the weight of the distance between two data points

The hyperparameters we utilized in K-nearest neighbors are as follows:

- ***knn_neighbors:*** *An integer hyperparameter ranging from 2 to 100 that represents the number of neighbors utilized in K-Nearest Neighbors.*
- ***knn_weights:*** *A categorical hyperparameter with the options 'uniform' and 'distance' that determines how neighboring points are weighted for predictions ('uniform' for equal weight,' 'distance' for closer neighbors having more effect).*
- ***knn_p:*** *Categorical hyperparameter having values 1 and 2, corresponding to the power parameter for the Minkowski distance metric used in KNN ('1' for Manhattan distance, '2' for Euclidean distance).*

The K-Nearest neighbors (KNN) achieved an accuracy of 92.47 % with F-1 score and precision being 0.61 and 0.66 respectively.

4.2.1.3 Support Vector Machines (SVMs):

These models are an instance of supervised machine learning that may be utilized for both regression and classification. A line or plane known as a hyperplane divides data into two regions, with all the data points in one region belonging to one class and all the data points in the other region belonging to the other. The best hyperplane between two classes of data is chosen by SVMs to function. The SVM approach identifies the hyperplane with the largest margin between the two classes. The margin measures the separation between each class's closest data points and the hyperplane.

The mathematical equation (6) of SVM is defined as follows [28] as,

$$\min_{w,b} \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{m} \varepsilon_i \quad (6)$$

$\frac{1}{2} \| w \|^2$ represents the squared norm of the weight vector w. Minimizing this term helps maximize the margin (i.e., the separation between the two classes).

$\sum_{i=1}^{m} \varepsilon_i$ Slack variable penalty represents the sum of the slack variables, which measure the amount by which each data point violates the margin.

$'C'$ the hyperparameter C controls the trade off between maximizing the margin and minimizing classification errors.

The hyperparameters utilized in support vector machines are as follows:

- ***svm_C:*** *A uniform hyperparameter with values ranging from 0.0 to 1.0 that represents the regularization parameter (C) for SVM. Smaller C values indicate more regularization.*
- ***svm_kernel:*** *Categorical hyperparameter having values 'poly' and 'rbf' that determine the type of kernel used by SVM (polynomial kernel is known as 'poly' and radial basis function kernel is known as 'rbf'). If 'svm_kernel' is 'poly,' an integer hyperparameter with values ranging from 1 to 10, denoting the degree of the polynomial kernel. The number 3 is chosen as the default for 'rbf'.*
- ***svm_tol:*** *A uniform hyperparameter with values ranging from 1e-5 to 1e-2 that represents the tolerance for halting criteria during optimization.*

The Support vector machines (SVMs) achieved an accuracy of 94.62 % with F-1 score and precision being 0.60 and 0.73 respectively.

4.2.1.4 Random Forest:

This model combines various decision trees to produce predictions. The predictions from the several decision trees are then combined to provide a final forecast. Each decision tree is trained on a different random subset of the training data. The training data is first generated into a bootstrap sample by the random forest approach. A random sample taken from the training data and replaced is known as a bootstrap sample. A data point can thus appear multiple times in the bootstrap sample. Once created, the bootstrap sample is used to train a decision tree. The decision tree is constructed using a greedy strategy, which repeatedly divides the training data into ever-tinier chunks.

The splitting criterion used by the decision tree algorithm is the Gini impurity criterion. The Gini impurity criteria quantify the impurity of a node in a decision tree. The impurity of a node is a measure of how well the data points in the node are categorized. A node with low impurity has well-classified data points. When the decision tree approach reaches a stopping point, such as a minimum number of samples or a maximum tree depth, it will stop splitting the training data. After each decision tree has been trained, its projections are added together to get the final prediction. The projections of the different trees are often merged using a voting mechanism. The ultimate prediction in a voting scheme is the class anticipated by the majority of the decision trees.

The mathematical equation (7) for random forest is as follows [23] as

$$P(\frac{y}{x}) = \sum_{i=1}^{n} w_i \cdot p_i (\frac{y}{x}) \quad (7)$$

Where:

- $P(\frac{y}{x})$ is the predicted probability of class y for the input x
- w_i is the weight of the i[th] decision tree
- $p_i(\frac{y}{x})$ is the predicted probability of class y for the input x from the i[th] decision tree

The mathematical equation (8) for the Gini impurity Criterion is as follows [24] as,

$$Gini = \sum_{y} p(y) \cdot (1 - p(y)) \quad (8)$$

Where:

- P(y) is the probability of class y.

The following are the hyperparameters we utilized in Random Forest:

- ***rf_estimators:*** *The number of decision trees in the Random Forest (numbers between 1 and 500).*
- ***rf_criterion:*** *A criterion for measuring split quality (also known as 'entropy' or 'gini').*
- ***rf_max_depth:*** *The maximum depth of decision trees (values between 1 and 100).*
- ***rf_min_samples_split:*** *The quantity of samples required to divide an internal node. (values between 2 and 50).*
- ***rf_min_samples_leaf:*** *The minimum amount of samples that must be present at a leaf node (1 to 25).*

The Random Forest achieved an *accuracy of 94.62 % with F-1 score and precision being 0.63 and 0.68*

respectively.

4.2.1.5 Naive Bayes:

The Bayes theorem is used to produce predictions with Naive Bayes. The Bayes theorem is a mathematical formula that estimates the probability of one event happening given the probability of another. The chance of a data point belonging to a class is calculated in Naive Bayes as the product of the probabilities of the data point's features. The classifier is named Naive Bayes because the probabilities of the features are considered to be independent of one another.

The following are the hyperparameters we utilized in Naive Bayes:

nb_smoothing: Uniform hyperparameter with values between 1e-10 and 1e-6. It represents the smoothing parameter (variance smoothing) used to avoid zero probabilities and improve the robustness of the Gaussian Naive Bayes model.

Then the Gaussian Naive Bayes model (nb) is created using the Gaussian NB class with the selected nb_smoothing hyperparameter, which enables the model to handle continuous data and make predictions based on the assumption of normal distribution for each feature. The Naive Bayes achieved an accuracy of 94.62 % with F-1 score and precision being 0.69 and 0.72 respectively.

4.2.1.6 Multilayer Perceptron:

A type of artificial neural network (ANN) is the MLP. It is made up of multiple layers of perceptrons. Perceptrons are basic units that can compute linear functions. An MLP's various layers of perceptrons enable the network to learn more complex functions. MLPs are frequently employed in classification and regression tasks. They are also utilized for image classification [25],[26], natural language processing, and speech recognition, among other things.

An MLP's input layer is the top layer. Data enters the network at the input layer. The following layer is the concealed layer. The network learns to represent data in the hidden layer. The number of hidden layers in an MLP might vary. Predictions are made in the output layer.

Each layer's perceptrons are linked together. The connections between the perceptrons are weighted. Weights are taught during the training procedure. The training phase involves modifying the weights in the network so that it can generate correct predictions.

A backpropagation algorithm is commonly used in the training of an MLP. The backpropagation algorithm is an iterative technique that modifies the weights in the network to reduce the error between anticipated and actual values.

The following are the hyperparameters utilized in Multilayer perceptron:

- ***mlp_hidden_layers:*** *Categorical hyperparameter with choices [1, 2, 3], representing the total number of hidden layers present in the MLP model.*
- ***mlp_hidden_units:*** *Integer hyperparameter with values between 16 and 128, representing each hidden layer's total number of neurons.*
- ***Mlp_activation:*** *Categorical hyperparameter with choices ['relu', 'tanh', 'logistic'], determines the hidden layer's activation function.*
- ***mlp_alpha:****Uniform hyperparameter with values between 1e-6 and 1e-3, representing the L2 regularization parameter for weight decay to prevent overfitting.*

The Multilayer perceptron (MLP) achieved an accuracy of 95.69 % with F-1 score and precision being 0.71 and 0.74 respectively.

4.2.1.7 Ensemble Model and Voting Classifier:

An ensemble model is a grouping of numerous machine learning models (classifiers or regressors) that collaborate to deliver a more accurate and reliable prediction. A Voting Classifier is a machine learning ensemble model that combines the predictions of numerous base classifiers (or models) to provide a final prediction. It works on the majority voting concept, where each base classifier's prediction is treated as a "vote" for a specific class, and the class with the most votes becomes the final predicted class.

In our proposed framework a Voting Classifier class is used to build an ensemble model called 'vc', which combines predictions from six base models: Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Random Forest, Naive Bayes, and Multi-Layer Perceptron. Each base model is given a weight between 0.0 and 1.0 (lr_w, knn_w, svm_w, rf_w, nb_w, and mlp_w) to determine its influence on the final prediction. The ensemble model is then fitted on the balanced training data (X_bal, y_bal), and the accuracy (acc) is calculated using predictions made on the validation set (X_val). The goal of this ensemble strategy shown in Figure 13 is used to enhance the prediction.

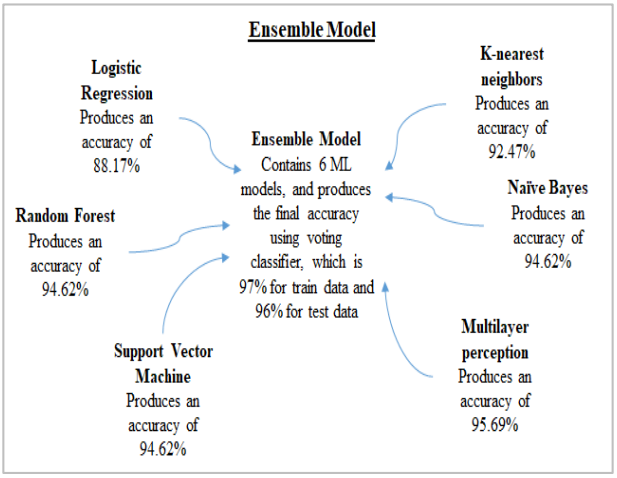

Fig 13. Ensemble Problem Model

Model accuracy on the validation set from the best trial is found to be 97%, as well as the best hyperparameters discovered during the investigation.

Table 3 gives a detailed list of hyperparameter names and its corresponding optimal values.

Finally, ensemble model using the best hyperparameters has been obtained from a hyperparameter optimization study. The best optimized hyperparameters are stored in the "study. best_params" attribute, which are used to initialize the ensemble model. The resulting model is stored in the variable model for further use and evaluation of the test data

4.3 SHAP

Model interpretability is critical for understanding and trusting prediction models in the field of machine learning research. SHAP (SHapley Additive exPlanations) is a powerful and extensively used framework for evaluating the predictions of complex machine learning models. In this research, we explore the use of SHAP, concentrating on its permutation-based approach to computing SHAP values. The use of SHAP in our study entails developing a SHAP explanation to help comprehend the model's predictions. First, we created a SHAP explanation using the shap. Explainer functionality. The explanation is put up to interpret our model's predictions on a certain dataset. To provide explicit feature attribution, we label the features appropriately by supplying the relevant feature names. We compute SHAP values using a permutation-based technique.

The SHAP explanation gives information about the computation's progress and timing throughout the procedure [27].

By using SHAP the importance of the attributes in the dataset were considered.

we were able to visually depict the importance of the attributes in the dataset.

Figure 14 depicts the importance of each attribute. The average SHAP value for each characteristic is shown in the bar graph. The SHAP value of a feature indicates how much it contributes to the model's prediction.

Figure 14 is separated into two sections: positive and negative SHAP values. A feature is said to favorably contribute to the model's prediction if the SHAP value is positive. Negative SHAP values suggest that the feature has a detrimental effect on the prediction made by the model. The magnitude of the SHAP values is also used to sort the bar graph. The properties with the highest SHAP values are the most crucial to the model's predictions.

The aspartate aminotransferase "AST" is the most essential feature for the model's predictions. This feature is an enzyme that is mostly located in the liver but is also present in muscles and other organs. When AST-containing cells are destroyed, the AST is released into the bloodstream.

Table 3. Best Hyperparameter Configurations for Enhanced Prediction

SI.NO	HYPERPARAMETER	VALUES
1	lr_penalty	'l2'
2	lr_solver2	'lbfgs'
3	lr_tol	0.003814159012031626
4	lr_C	0.42350470965469134
5	knn_neighbors	83
6	knn_weights	'distance' knn_p
7	svm_C	0.9731171875077148
8	svm_kernel	'poly'
9	svm_degree	5
10	svm_tol	0.002292510698981856
11	Rf_estimators	70
12	rf_criterion	'gini'
13	rf_max_depth	38
14	rf_min_samples_split	29
15	rf_min_samples_leaf	14
16	nb_smoothing	2.596301857486734e-07 mlp_hidden_layers
17	mlp_hidden_units	113
18	mlp_activation	'tanh'
19	mlp_alpha	0.004011044419757664
20	et_n_estimators	38
21	et_criterion	'gini'
22	et_max_depth	29
23	et_min_samples_split	30
24	et_min_samples_leaf	21
25	lr_w	0.029872771851786117
26	knn_w	0.2574479288604744
27	svm_w	0.5286677776732325
28	rf_w	0.6069934816118663
29	nb_w	0.3189569725246839
30	mlp_w	0.8092536312976131
31	et_w	0.40729870664578616

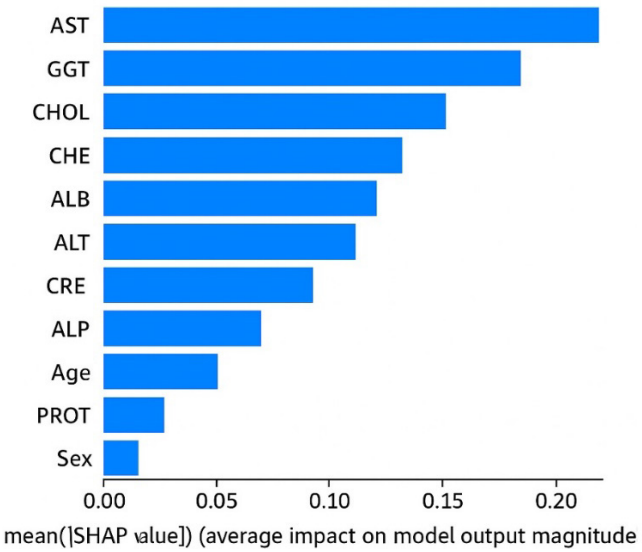

Fig 14. SHAP Value Analysis: Key Predictors in the Model

Gamma-glutamyl Transferase (GGT) test is the component with the second-highest SHAP score. This test in the feature establishes the blood level of GGT. GGT is an enzyme that is present throughout the body, but it is most common in the liver. GGT may seep into the bloodstream if the liver is injured.This is also visualized using Force Plot

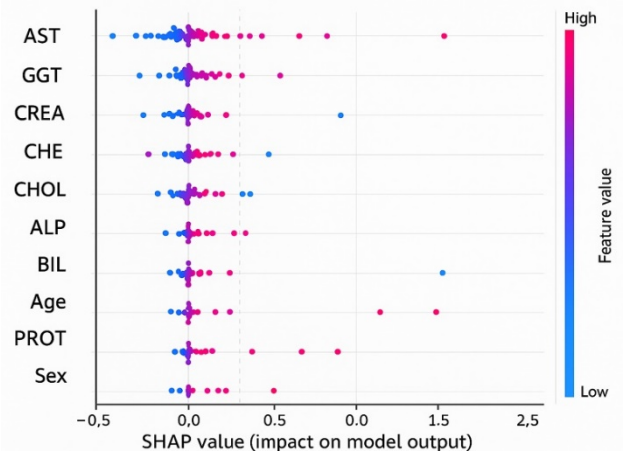

Fig 15. SHAP Summary Plot: Feature Impact on Model Output

Figure 15 depicts the SHAP force plot for a model that predicts whether the patient will be diagnosed with liver illness or not. The dataset contains the following features: age, AST, ALT, GGT, ALB, BIL, and PROT. The arrows in the graphic represent each feature's contribution to the model's prediction. The size of the arrows represents the magnitude of the contribution. The contribution is indicated by the direction of the arrows. For example, the arrow for the feature "AST" points to the right, indicating that a high AST score is connected with an increased likelihood of being diagnosed with liver illness. It also gives us the information that AST, ALT, and GGT are the most relevant features for the model's predictions. These characteristics are all connected to liver function and are all associated with an increased risk of being diagnosed with liver disease.

The other properties in the image are also significant for the model's predictions, but not as much as the AST, ALT, and GGT.

Figure 16 indicates that AST levels rise before any other indications of liver disease arise. This suggests that AST levels could be used to to fine-tune the ensemble model, consisting of SVM, KNN, RF, Naive Bayes, LR, and MLP classifiers. When models were observed individually, the accuracy of MLP tops the other models with an accuracy of 95.69%. And the optimized ensemble model achieved an impressive 97% accuracy on the validation data and maintained its robustness, yielding 96% accuracy on the test data. Moreover, the implementation of SHAP features provided valuable insights into the model's predictions and increased its interpretability. The framework provides importance of each feature using Bar Graph and Force Plot; it also demonstrates the effectiveness of combining data preprocessing techniques, hyperparameter tuning, and interpretability tools to build powerful and reliable classification models.

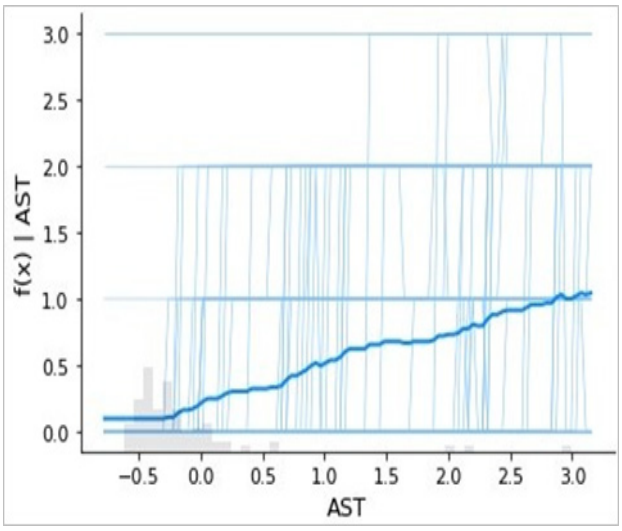

Fig 16. SHAP Dependence Plot for AST

Figure 16 indicates that AST levels rise before any other indications of liver disease arise evaluate people for liver illness before they show any symptoms. When AST levels are confirmed to be elevated, additional tests may be performed to confirm the diagnosis of liver disease.

The graph also indicates that AST levels can change over time in persons with liver disease. This means that a single AST test may not be enough to diagnose liver disease. However, if AST values are consistently excessive, this is a clear indication of liver disease.

Figure 15 indicates that AST levels rise before any other indications of liver disease arise. This suggests that AST levels could be used to fine-tune the ensemble model, consisting of SVM, KNN, RF, Naive Bayes, LR, and MLP classifiers. When models were observed individually, the accuracy of MLP tops the other models with an accuracy of 95.69%. And the optimized ensemble model achieved an impressive 97% accuracy on the validation data and maintained its robustness, yielding 96% accuracy on the test data. Moreover, the implementation of SHAP features provided valuable insights into the model's predictions and increased its

interpretability. The framework provides importance of each feature using Bar Graph and Force Plot; it also demonstrates the effectiveness of combining data preprocessing techniques, hyperparameter tuning, and interpretability tools to build powerful and reliable classification models.

Table 4. Benchmarking Machine Learning Models for Hepatitis C Prediction with Accuracy (%)

Name of Model	Accuracy (%)
Logistic Regression	88.17
Random Forest	94.62
K- nearest neighbors	92.47
Naive Bayes	94.63
Support Vector Machines	94.62
Multilayer perceptron	95.69

The **Aspartate Aminotransferase (AST) levels** tend to rise before other clinical indicators of liver disease become apparent. This suggests that AST can serve as an early biomarker for liver dysfunction, enabling the evaluation of individuals **before** they exhibit noticeable symptoms. When **elevated AST levels** are detected, further diagnostic tests are conducted to confirm the presence and severity of liver diseases. This early detection mechanism is crucial for timely intervention, potentially preventing disease progression and improving patient outcomes.

5 Results and Discussion

The proposed framework presents a comprehensive approach to enhance the performance of a classification model. It begins by employing SMOTE to address data imbalance, creating a balanced dataset.

According to the study, SMOTE boosts model accuracy marginally, with the MLP model having the highest accuracy. SMOTE improves RF and LR accuracy slightly, but SVM, KNN, and Naive Bayes accuracy remains insignificant. Subsequently, hyperparameter optimization using Optuna is applied to fine-tune the ensemble model, consisting of SVM, KNN, RF, Naive Bayes, LR, and MLP classifiers. When models were observed individually, the accuracy of MLP tops the other models with an accuracy of 95.69%. And the optimized ensemble model achieved an impressive 97% accuracy on the validation data and maintained its robustness, yielding 96% accuracy on the test data. Moreover, the implementation of SHAP features provided valuable insights into the model's predictions and increased its interpretability. The framework provides importance of each feature using Bar Graph and Force Plot; it also demonstrates the effectiveness of combining data preprocessing techniques, hyperparameter tuning, and interpretability tools to build powerful and reliable classification models. Table 4 presents the Model name with their corresponding individual accuracy score. Table 5 summarizes the techniques and its outcome.

Table 5. Impact of SMOTE, OPTUNA, and SHAP on Model Performance

Techniques	Uses And Results
SMOTE	Balanced the dataset, so that the output produced is unbiased.
OPTUNA	Tuned and optimized the hyperparameters of the machine learning model.
SHAP	Increased

The libraries and parameters have been used to address feature selection methodologies. The effectiveness of the classifier was assessed using 10-fold cross-validation, accuracy, precision, recall, and overall average scores for datasets with multiple and binary labels. This investigation compared the effectiveness of tools and classifiers on the HCV dataset's multi- and binary-class labels. A random forest analysis of the multiclass dataset revealed that KNN (26.44%) and the multiclass dataset (28.36%) had the highest accuracy. The accuracy of NN's binary class labels was the greatest (53.12%), although KNN's performance in recall and precision was better. SVM, RF, KNN, and NB (51.31%) were the next most accurate models after the R multiclass dataset. KNN (53.66%) and boosting (54.23%) both demonstrated great accuracy. Precision and recall displayed varied results, despite the accuracy of multiclass and binary class labels performing similarly in both cases.

Experimental results demonstrate an HCV detection accuracy of 97%, highlighting the potential of this approach in medical diagnostics. By addressing real-world challenges such as class imbalance, computational feasibility, and clinical interpretability, our proposed methodology empowers healthcare professionals with reliable decision-support and ultimately improves patient outcomes and facilitates early disease intervention.

The findings of this study demonstrate the effectiveness of integrating SMOTE, Optuna, and SHAP in improving HCV detection accuracy. However, real-world implementation in healthcare presents several challenges that must be addressed to ensure practical applicability.

6 Conclusion

Our proposed method offered a complete framework that successfully improved the performance of a classification model. Hence, the outstanding results obtained by using SMOTE to handle data imbalance and Optuna for hyperparameter optimization in an ensemble

model consist of several classifiers. The MLP classifier achieved the greatest individual accuracy of 95.69%. The optimized ensemble model, on the other hand, beat all individual models, achieving an outstanding 97% accuracy on the validation data and displaying robustness with 96% accuracy on the test data. Furthermore, the inclusion of SHAP features improved the model's interpretability, providing significant insights into its predictions. Visualizations such as the Bar Graph and Force Plot, which highlighted feature relevance, added to the framework's efficacy. To potentially attain even greater accuracy, advanced data preprocessing approaches, a more diverse range of classifiers, and additional hyperparameter optimization strategies must be incorporated in future research. Additionally, expanding the application of our model to different domains or larger datasets can offer valuable insights on its scalability and generalizability. Lastly, our work creates new avenues for advancements in the field of machine learning and offers critical insights into the creation of reliable and strong categorization models.

Conflict of Interest

The authors declare no conflict of interest.

Acknowledgment

The authors wish to thank SVCE college Sriperumbudur for great support to do this project in the ECE department research center.

References

[1] C. M. Rice, J. T. Stapleton, and P. Simmonds, "Expanded classification of hepatitis C virus into 7 genotypes and 67 subtypes: updated criteria and genotype assignment web resource," Hepatology, vol. 59, no. 1, pp. 318-327, Jan. 2014.

[2] S. M. Borgia, C. Hedskog, B. Parhy, R. H. Hyland, L. M. Stamm, D. M. Brainard, M. G. Subramanian, J. G. McHutchison, H. Mo, E. Svarovskaia, and S. D. Shafran, "Identification of a novel hepatitis C virus genotype from Punjab, India: expanding classification of hepatitis C virus into 8 genotypes," The Journal of Infectious Diseases, vol. 218, no. 11, pp. 1722-1729, Nov. 2018.

[3] D. J. Ruzicka, J. Tetsuka, G. Fujimoto, and T. Kanto, "Comorbidities and co-medications in populations with and without chronic hepatitis C virus infection in Japan between 2015 and 2016,"BMC Infectious Diseases, vol. 18, no. 237, pp. 1-10, May 2018.

[4] A. A. Kashif, B. Bakhtawar, A. Akhtar, S. Akhtar, N. Aziz, and M. S. Javeid, "Treatment response prediction in hepatitis C patients using machine learning techniques," International Journal of Technology, Innovation and Management (IJTIM), vol. 1, no. 2, pp. 79-89, 2021.

[5] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," Neurocomputing, vol. 415, pp. 295-316, Dec. 2020

[6] J. Cai, J. Luo, S. Wang, and S. Yang, "Feature selection in machine learning: A new perspective," Neurocomputing, vol. 300, pp. 70-79, May 2018.

[7] N. Tran, J. G. Schneider, I. Weber, and A. K. Qin, "Hyper-parameter optimization in classification: To-do or not-to-do," Pattern Recognition, vol. 103, p. 107245, Jan. 2020.

[8] A. Nugroho and H. Suhartanto, "Hyper-parameter tuning based on random search for dense net optimization," in 7th IEEE International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), Semarang, Indonesia, 2020, pp. 96-99.

[9] Z. Cai, Y. Long, and L. Shao, "Classification complexity assessment for hyperparameter optimization," Pattern Recognition Letters, vol. 125, pp. 396-403, Nov. 2019.

[10] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next generation hyperparameter optimization framework," in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 2019, pp. 2623-2631.

[11] M. Yağanoğlu, "Hepatitis C virus data analysis and prediction using machine learning", Data & Knowledge Engineering, vol. 142, p. 102087, Feb. 2022.

[12] S. T. I. Tonmoy and S. M. Zaman, "OOG-Optuna optimized GAN sampling technique for tabular imbalanced malware data," IEEE International Conference on Big Data (Big Data), pp. 6534-6539, November 2022.

[13] M. Y. Shams, E. M. El-kenawy, A. Ibrahim, and A. M. Elshewey, "A hybrid dipper throated optimization algorithm and particle swarm optimization (DTPSO) model for hepatocellular carcinoma (HCC) prediction," Biomedical Signal Processing and Control, vol. 85, p. 104908, 2023.

[14] M. Yağanoğlu, "Hepatitis C virus data analysis and prediction using machine learning," Data & Knowledge Engineering, vol. 142, p. 102087, Feb. 2022.

[15] A. M. Ali, M. R. Hassan, F. Aburub, M. Alauthman, A. Aldweesh, A. Al-Qerem, I. Jebreen, and A.

Nabot, "Explainable machine learning approach for hepatitis C diagnosis using SFS feature selection," Applied Computing and Informatics, 2023, in press.

[16] Hashem, G. Esmat, W. Elakel and H. Shahira, Comparison of machine learning approaches for prediction of advanced liver fibrosis in chronic hepatitis C patients, IEEE/ACM Trans Computational Biology and Bioinformatics, vol.15, no.3, pp.861-868,2018.

[17] M. O. Edeh, S. Dalal, I. Ben Dhaou, C. C. Agubosim, C. C. Umoke, N. E. Richard-Nnabu, and N. Dahiya, "Artificial intelligence-based ensemble learning model for prediction of hepatitis C disease," Frontiers in Public Health, vol. 10, p. 892371, 2022.

[18] V. Tsvetkov, I. Tokin, and D. Lioznov, "Machine learning model for diagnosing the stage of liver fibrosis in patients with chronic viral hepatitis C," IEEE Transactions on Biomedical Engineering, February 2021(online).

[19] E. Dritsas and M. Trigka, "Supervised machine learning models for liver disease risk prediction," Computers, vol. 12, no. 1, p. 19, Jan. 2023

[20] M. Alauthman, A. Aldweesh, A. Al-Qerem, F. Aburub, Y. Al-Smadi, A. M. Abaker, O. R. Alzubi, and B. Alzubi, "Tabular data generation to improve classification of liver disease diagnosis," Applied Sciences, vol. 13, no. 2678, 2023.

[21] UCI Machine Learning Repository, "HCV data," UCI Machine Learning Repository, https://archive.ics.uci.edu/ml/datasets/HCV (online).

[22] M. M. Asha, G. N. Balaji, S. Mythili, A. Karthikeyan, and N. Thillaiarasu, "An efficient brain tumor detection algorithm based on segmentation for MRI system," International Conference on Recent Advancements in Information Technology, Science and Engineering (ICRAITSE - 17), Thoothukudi, India, Dec. 2017, pp. 1-8.

[23] Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. , "*Applied Logistic Regression"* (3rd ed.). John Wiley & Sons, 2013.

[24] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning with Applications in R,* Springer, 2017.

[25] Mercy Theresa M., A. Jesudoss., P. Pattunnarajam., Sudha Rajesh., Jaanaa Rubavathy., A. Raja, "CAD Based Automatic Detection of Tuberculosis in Chest Radiograph using Hybrid Method," Inderscience International Journal of Engineering Systems Modelling and Simulation, Vol.14, No.4, pp.179-185, 2022. doi: 10.1504/IJESMS.2022.10044924.

[26] S.M. Mehzabeen, R Gayathri, "Heuristically Improvised rice disease classification framework based on adaptive segmentation with the fusion of LSTM layer into Multi-Scale Residual Attention Network", Biomedical Signal Processing and Control, Elseveir, England SCI LTD, vol.99, pp. 106875,2024.

[27] G. Naveen Balaji, and D. Rajesh, Python Based Reverse Timing Algorithm for Human Brain Activity Using Color Psychology. International Journal of Indian Psychology, vol. 4, no. 3, pp. 79-86, 2017.

[28] Cristianini, N., & Shawe-Taylor, J., "*An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods,"* Cambridge University Press,2000.

Biographies

S.M. Mehzabeen graduated with a Bachelor of Engineering (B.E.) degree from Alagappa Chettiar College of Engineering and Technology, affiliated with Madurai Kamaraj University. 2013 she obtained her Master of Engineering (M.E.) degree from Velammal Engineering College, under Anna University. She achieved a University rank during her Master's degree. In 2019, she completed an ISRO-funded project titled "Design and Implementation of High-Performance Hyperspectral Target Detection System Using FPGA" as a co-principal investigator. She is currently pursuing a doctoral degree in Information and Communication Engineering on a part-time basis. Her research interests encompass Image Processing, Computer Vision, Embedded Systems, and IoT Design.

R.Gayathri completed her B.E. degree in Electronics and Communication Engineering from the University of Madras in 1999, her M.Tech. degree, and her PhD degree in Information and Communication Engineering from Anna University in 2001 and 2014, respectively. She is working as a professor at Sri Venkateswara College of Engineering. She is a recognized supervisor at Anna University. Her research includes computer vision, pattern recognition, VLSI signal processing, remote

sensing, and machine learning for video analytics, with a focus on human tracking, multi-resolution video processing, biologically inspired spatial-temporal filtering, hyperspectral image processing, natural language processing, etc. She has received and successfully completed the ISRO-funded project titled "Design and Implementation of a High-Performance Hyperspectral Target Detection System Using FPGA" during 2019. She has published in a number of SCI- and Scopus-indexed international publications. She is the recipient of the Global Teacher Award 2021 from the AKS Education Awards. She is the recipient of the International Best Researcher Award in the field of "Artificial Intelligence and Machine Learning" (IIRA 2022), awarded by ISSN, the World Research Council, and the Times of Research. She has been serving as an active reviewer in the Measurement journal, Elsevier, the Imaging Science Journal, Taylor & Francis Ltd., England, the Journal of Intelligent and Fuzzy Systems, IOS Press, Netherlands, the British Journal of Mathematics and Computer Science, the Journal of Computer Sciences, the American Journal of Applied Science, the Inder science Journal, the Journal of Sensors and IEEE conferences, Springer, the British Journal of Mathematics and Computer Science, the Journal of Computer Sciences, etc. She has been honored as an academic editor of "The Asian Journal of Research in Computer Science" and as an academic editor of "The DECENT Journals of Brisbane, Queensland, Australia. She has mentored many undergraduate and postgraduate research students in computer vision and data science projects.

PATTUNNARAJAM PARAMASIVAM received the B.E. degree in Electronics and Communication Engineering from Madras University, Chennai, Tamil Nadu, India, in 1997, her M.Tech. degree in VLSI design from Bharath University, Chennai, in 2006, and her Ph.D. degree in VLSI design from Anna University, Chennai, in 2020. She is working as an Associate Professor with the Sri Venkateswara College of Engineering, Sriperumbudur, India.

She has more than 24 years of teaching and research experience in VLSI design. She has supervised various bachelor's and master's projects. She has authored and coauthored many technical offerings, including articles in international refereed journals and international/national conferences. She has published in a number of SCI- and Scopus-indexed international publications Her research interests include low power VLSI circuits, Testing of VLSI circuits, and Digital Electronics, Nano Electronics. Dr. Pattunnarajam has become an Active Member of ISTE, India (LM'13), IETE, India (LM'13), IAENG (M'21), and SDIWC (M'19). She is a reviewer in reputed peer-review journals, include Interscience and Springer (JAIHSC).

Ramya Anandanatarajan received the Bachelor's degree in Electronics and Communication Engineering from Pondicherry University, Puducherry, India, in 2013, the Master's Degree in Applied Electronics from Anna University, Chennai, India, in 2016, and the Ph.D. degree from the National Institute of Technology, Tiruchirappalli, India, in 2022. She is currently an Assistant Professor at Sri Venkateswara College of Engineering, Sriperumbudur, India. Her current research interests include signal conditioning, artificial intelligence, and data acquisition.