

Research Article

Comparative Analysis of Machine Learning and Deep Learning Models for Classification and Prediction of Liver Disease in Patients with Hepatitis C

Shalem Preetham Gandu¹ , M. Roshni Thanka² , E. Bijolin Edwin³ , Ebenezer Veemaraj^{2,*} ,
S. Stewart Kirubakaran³ 

¹ Data Analytics Engineering, Volgenau School of Engineering, George Mason University, Fairfax, VA 22030, United States

² Division of Data Science and Cyber Security, Karunya Institute of Technology and Sciences, Coimbatore, 641114, India

³ Division of Computer Science and Engineering, Karunya Institute of Technology and Sciences, Coimbatore, 641114, India

*Corresponding Author: Veemaraj Ebenezer, E-mail: ebenezer@karunya.edu

Article Info	Abstract
Article History	Hepatitis C, an infectious disease that claims thousands of lives annually, affects millions globally. Disease prediction and classification are crucial in medical research, as forecasting diseases helps identify patients at risk. This study used the Electronic Health Records dataset from the UCI Machine Learning Repository, which included 615 individuals, of which 540 were healthy, and 75 had hepatitis C. Machine learning (ML) algorithms proved particularly effective in developing prediction models for the presence of the Hepatitis C Virus (HCV), thereby improving performance and efficiency. After evaluating several categories of machine learning (ML) algorithms for classification, the study employed seven algorithms: an Artificial Neural Network (ANN) and an ensemble model combining these seven distinct algorithms. The classifiers used in model assessment included K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), Logistic Regression (LGR), XGBoost (XGB), and Gaussian Naive Bayes (GNB). Additionally, an ANN and an ensemble model were implemented. This study presents a comparative analysis of these algorithms, with the Random Forest (RF) classifier achieving the highest accuracy across all iterations. The experimental results demonstrate that the proposed ensemble framework achieved greater accuracy with XGB as the final estimator. Notably, the RF classifier reached an accuracy of 99.88%. In conclusion, the study demonstrates that using an ensemble model improves prediction accuracy for Hepatitis C, yielding more precise results than applying individual algorithms.
Received Jun 03, 2024	
Revised Feb 10, 2025	
Accepted Feb 22, 2025	
Keywords	
Classification	
Hepatitis C virus	
Liver fibrosis	
Cross-validation	
Ensemble methods	



Copyright: © 2025 Shalem Preetham Gandu, M. Roshni Thanka, E. Bijolin Edwin, Ebenezer Veemaraj, and S. Stewart Kirubakaran. This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY 4.0) license.

1. Introduction

Hepatitis C is a viral infection caused by the Hepatitis C Virus (HCV), which leads to liver inflammation. HCV infection can be either acute or chronic. Acute HCV infections are generally asymptomatic and rarely fatal. Approximately 30% of infected individuals can spontaneously clear the virus within six

months without treatment. Despite this, no effective vaccine currently exists for Hepatitis C.

According to the World Health Organization (WHO), over 58 million people suffer from chronic HCV globally, with approximately 1.5 million new infections occurring annually [1]. Genetically, HCV comprises 8 genotypes and 86 subtypes, which exhibit geographical variations in prevalence [2]. The disease burden is particularly higher in low- and middle-income countries in Asia and Africa compared to high-income regions, such as North America and Europe [3].

ML has emerged as a transformative approach in the medical field, especially for disease prediction and classification tasks[4]. ML models can identify patterns in electronic health records and other datasets more accurately and efficiently than traditional rule-based methods. Machine learning (ML) systems do not require constant human oversight and can adapt to various learning problems. However, achieving 100% accuracy remains difficult, and high-quality labeled data is often required.

This research utilizes data from the UCI Machine Learning Repository to evaluate several machine-learning algorithms for predicting HCV infection. Seven ML classifiers, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), Logistic Regression (LGR), XGBoost (XGB), and Gaussian Naive Bayes (GNB), are applied along with an Artificial Neural Network (ANN) and an ensemble stacking model. The study compares model performance across splitting ratios, cross-validation folds, and hyperparameters.

The primary goal of this study is to enhance prediction accuracy for Hepatitis C infection by identifying the best-performing classification model. This can enable timely diagnosis, personalized treatment, and better disease management, ultimately improving patient outcomes and reducing the burden on healthcare systems.

2. Literature Review

This section of the paper incorporates a variety of earlier studies that used the HCV dataset to predict and classify hepatitis C infection. Several studies have been conducted, but only six recent studies are considered in this paper. They are as follows:

The study by Kaunang [5] aimed to predict the Hepatitis C virus using various ML algorithms and to identify the optimum model for Hepatitis C disease classification. They presented a prediction model utilizing ML classifiers for the application dataset established in the UCI-ML repository based on KNN, SVM, RF, Neural Network (NN), GNB, and LGR. Based on the findings, each model performed well, but Logistic

Regression performed the best. A classification methodology based on ML algorithms such as KNN, GNB, NN, and RF was proposed by Syafaah, et al. [6]. In this study, the classification of the HCV virus was obtained using the UCI dataset. This study aimed to test and evaluate the accuracy of detecting HCV using classification ML algorithms. The results reveal that the accuracy of the NN is high, at 95.12%, when compared to the accuracies of the methods KNN, NB, and RF, which were 89.43%, 90.24%, and 94.31%, respectively.

This is an applied study by Orooji and Kermani [7] that uses the hepatitis C dataset from the UCI ML repository. Multi-layer perceptron, Bayesian network, and decision tree are three data mining methods that were implemented, and their effectiveness was assessed using a 10-fold cross-validation approach. Finally, the best method was used to design the user interface in MATLAB programming language. The results revealed that the oversampling strategy increased the performance metrics of the algorithms in predicting HCV, with random forest as the best method, achieving a 99.9% accuracy. Furthermore, the random forest exhibited the best performance metrics concerning sensitivity, accuracy, F-measure (99.9%), and 100% specificity. They claimed that the approach reported in this study outperformed all previously offered methodologies and could be used successfully to diagnose and assess the severity of HCV.

In the study by Ahammed, et al. [8], a machine learning-based methodology for classifying hepatitis C virus-infected patients' liver stages is presented. Egyptian patients with liver fibrosis disease cases were obtained from the UCI ML repository. A synthetic minority oversampling approach was employed to balance instances of several categories. Later, various feature selection approaches were employed to identify relevant characteristics of the hepatitis C virus in this dataset. Several classifiers were used to classify patients into balanced primary, features-chosen, and primary HCV cases. After reviewing the results, KNN surpassed all other classifiers with an accuracy of 94.40 %. The study by Nandipati, et al. [9] presented an HCV prediction model utilizing ML techniques for the application dataset established in the UCI-ML repository based on RF and KNN for programming with various numbers of features and attributes, Python and R. ML approaches were used to predict the clinical groups that combine the data to determine the most important features using RF classification. In binary class and multiclass labels, RF (54.56 %, Python) and KNN (51.06 %, R) demonstrated the highest precision. The comparison of all the evaluation metrics reveals that R is a superior tool in this scenario. On the other hand, the binary class performance score performs better than the multiclass label.

In order to create a model for predicting Egyptian children infected with the hepatitis C virus, [10] employed Random Forest, APRI, and FIB-4 scores and their AUCs. RF, APRI, and FIB-4 had high AUCs for predicting any form of fibrosis, advanced fibrosis, and distinguishing between moderate and advanced fibrosis, respectively. APRI had AUCs of 78 %, 81.6 %, and 77 %; FIB-4 had 74 %, 82.8 %, and 78 %; and RF had 90.3 %, 89.4 %, and 82.2 %. Machine learning is a valuable supplement to non-invasive techniques for staging and predicting juvenile liver fibrosis. Additionally, the acquired APRI and FIB-4 cutoff values performed well and were similar to some previously reported ones. There was some agreement between RF, APRI, and FIB-4 predictions for the prognosis and staging of fibrosis. Finally, it was suggested that computed scores or RF output cannot sometimes substitute a liver biopsy.

Table 1. Summary of the related work on HCV Prediction

Author	Dataset	Methodology	Performance Metrics in %	Pros	Cons
Kaunang [5]	UCI ML Repository	KNN	Accuracy = 96.5	Provides insights into the most effective machine learning algorithm for HCV prediction	No discussion on data pre-processing or evaluation
		SVM	Accuracy = 96.7		
		RF	Accuracy = 97.3		
		NN	Accuracy = 97.1		
		NB	Accuracy = 96.0		
		LR	Accuracy = 97.9		
Syafaah, et al. [6]	UCI ML Repository	KNN	Accuracy = 89.43	Evaluation of feature selection techniques to enhance the accuracy of the models	Small dataset
		NB	Accuracy = 90.24		
		NN	Accuracy = 95.12		
		RF	Accuracy = 94.31		
Ahammed, et al. [8]	UCI ML Repository	KNN	Accuracy = 94.40	Utilizes multiple machine learning algorithms for the prediction of HCV infectious state	No comparison with other ML techniques
		RF	Accuracy = 92.48		
		SVM	Accuracy = 82.77		
Nandipati, et al. [9]	UCI ML Repository	KNN	Accuracy = 52.97	Provides insights into the class imbalance for HCV prediction	Limited discussion on dataset and evaluation metrics
		NB	Accuracy = 51.56		
		RF	Accuracy = 50.85		
Barakat, et al. [10]	Egyptian children with only HCV	RF	Accuracy = 87.5 AUC = 90.3 Sensitivity = 82.0	Use of machine learning techniques to predict hepatic fibrosis in children	Limited dataset

The reviewed papers focus on the application of ML algorithms in predicting and diagnosing hepatitis C virus (HCV) infections, which are also compared in Table 1. They highlight the high accuracy of these models in detecting HCV infections, which can enhance disease management and prevention. Imbalanced

datasets are a significant challenge in HCV prediction, and some studies focus on predicting HCV infection stages and identifying the infectious state of HCV patients. These studies aim to reduce the disease burden of HCV infections, improve patient outcomes, and advance the field of disease diagnosis and treatment. The use of ML shows promise in improving the diagnosis and treatment of HCV infections. Accurate prediction and identification of HCV infections can enable patients to receive timely and appropriate care, thereby reducing the disease burden on individuals and healthcare systems. The primary motivation for this research project was to address the critical need for accurate diagnosis and treatment of patients with lung disease. By leveraging machine learning techniques on a health dataset, the researchers aimed to improve the prediction and classification of lung diseases, enabling early detection, personalized treatment, and better patient outcomes. The research aimed to fill existing gaps in the diagnosis of lung disease and provide healthcare professionals with reliable tools for informed decision-making. Ultimately, the goal was to enhance the understanding of lung diseases, contribute to advancements in healthcare, and positively impact patient care and management strategies [11-14].

3. Methodology

The primary goal of this study is to compare the performance of various ML classification algorithms and a neural network in predicting HCV using different techniques. Both all features and selected features were evaluated to determine the accuracy of the classifiers. A range of popular ML classifiers and Neural Networks were employed to classify the selected characteristics into various categories. The classification report was used to compute metrics and identify the most effective model. Data pre-processing, splitting, and feature selection were all performed to prepare the dataset for model building. In addition to these phases, oversampling techniques such as SMOTE and ADASYN, normalization, and dimensionality reduction were employed to achieve optimal results, which were incorporated into the pre-processing phase and not presented separately. Figure 1 illustrates an architecture for predicting HCV disease. The study's main objective was to compare the results of ML classification algorithms and a neural network for HCV prediction based on the methods. All attributes and selected features were analyzed to compare the accuracy of the classifiers. Popular ML classifiers and Neural Networks were used to categorize the selected characteristics into different classes. The classification report was used to calculate metrics and identify the perfect model. Data pre-processing, splitting, and feature selection are performed to prepare the dataset for model

building. Apart from the aforementioned phases, several additional steps were necessary to achieve the desired results, including oversampling methods such as SMOTE and ADASYN, normalization, and dimensionality reduction. They are, however, included in pre-processing and are not specified separately. Figure 1 depicts an architecture for predicting HCV disease.

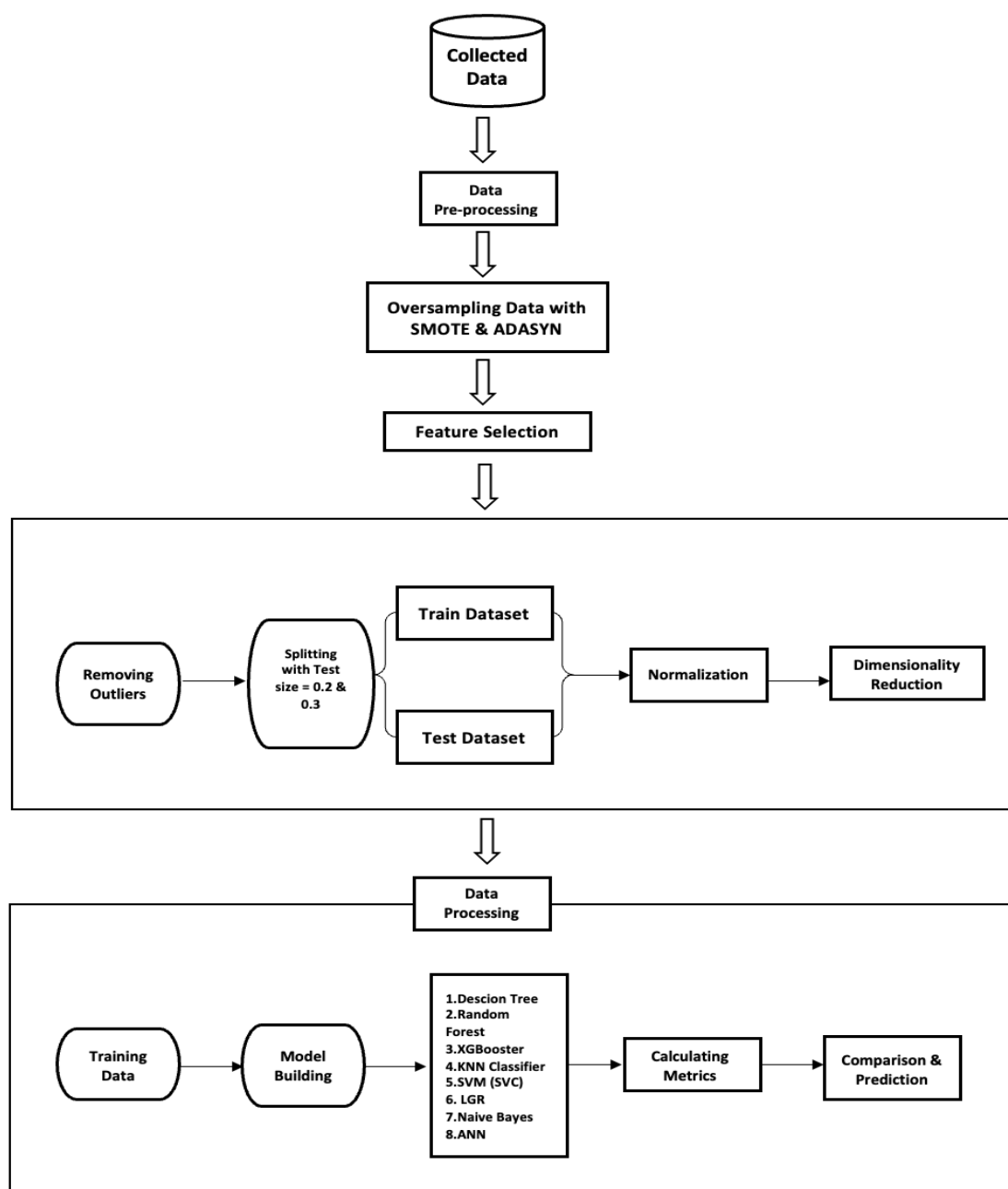


Figure 1. The architecture of HCV prediction

3.1. Data Description

A real-life laboratory dataset of hepatitis C was examined in this research. The targeted feature for classification is the category. This dataset was acquired from the UCI Machine Learning Repository. The

dataset contains 615 records (patients), each with 14 features comprising information about the patient. Twelve feature attributes are utilized as diagnostic inputs, with the 'Category' attribute selected as output. The dataset comprises demographic and laboratory data from blood donors and patients with hepatitis C. Table 2 summarizes the dataset used in this study.

Table 2. Dataset Description

Feature	Description	Type	Data Range or Unit
Category	The target attribute for classification	Nominal	O=Blood Donor, os=suspect Blood Donor, 1=Hepatitis, 2=Fibrosis, 3=Cirrhosis
Chronicle Age	Age of the Patient	Quantitative	Years
Gender	Gender or Sex	Nominal	m- Male, f- Female
ALB	Albumin level in blood	Quantitative	g/L
ALP	Alkaline phosphate level in blood	Quantitative	IU/L
AST	Aspartate aminotransferase level in blood	Quantitative	IU/L
BIL	Bilirubin level in blood	Quantitative	mg/dL
CHE	Choline esterase level in blood	Quantitative	IU/L
CHOL	Cholesterol level in blood	Quantitative	milligrams/dL
CREA	Serum creatinine level in blood	Numerical	mg/dL
GGT	Gamma-glutamyl-transferase level in blood	Numerical	IU/L
PROT	Total protein level in blood	Numerical	g/L

International units (IU), L stands for liter, mg for milligram, g for gram, and dL for deciliter are meanings and measuring units of clinical characteristics in the Lichtinghagen dataset [15, 16]. The authors mentioned some of these interpretations in the original study [16] and the article [17]. While the dataset of 615 patients used in this study may be considered relatively small, it is essential to understand that the success of an ML project is not solely dependent on the size of the dataset. Various techniques can be employed to overcome the potential limitations of a small dataset, including oversampling methods like SMOTE and ADASYN. However, it is crucial to carefully evaluate the quality and relevance of the synthetic data generated by these methods, as they may not accurately represent the true distribution of the underlying data.

Additionally, it is essential to perform appropriate pre-processing steps, such as feature selection and normalization, and assess the quality and representativeness of the original dataset. To assess the model's performance, it is essential to use appropriate evaluation metrics, such as cross-validation and holdout testing. Ensuring patient privacy is critical for this machine learning project involving health data. The data

type was carefully assessed, and necessary measures were taken to remove or anonymize any personally identifiable information (PII). For patients with hepatitis, it may be necessary to exclude any data that could be used to directly identify them, such as their identity number, name, or address.

To handle patient data privacy on a high level, the following measures were taken in the research work:

- **Data Minimization:** Only the patient data relevant to the research was obtained, thereby reducing the quantity of sensitive information retained and processed.
- **Secure Data Storage:** Patient data was stored in a highly secure environment, using encrypted storage systems and according to industry best practices to prevent unauthorized access.
- **Anonymization and de-identification:** Strict approaches were employed to anonymize and de-identify patient data by removing direct and indirect identifiers, ensuring that individuals could not be identified.
- **Controlled Data Access:** Strict access restrictions were installed, enabling only authorized individuals with a valid need to view patient data. Role-based access methods were employed to govern and monitor data access.

Categorical data is shown using bar charts and pie charts. As seen in the picture, both types of graphs have variations. The category column was shown as a pie chart in this study. Figure 2 illustrates that the dataset is unbalanced, as the values in each feature vary.

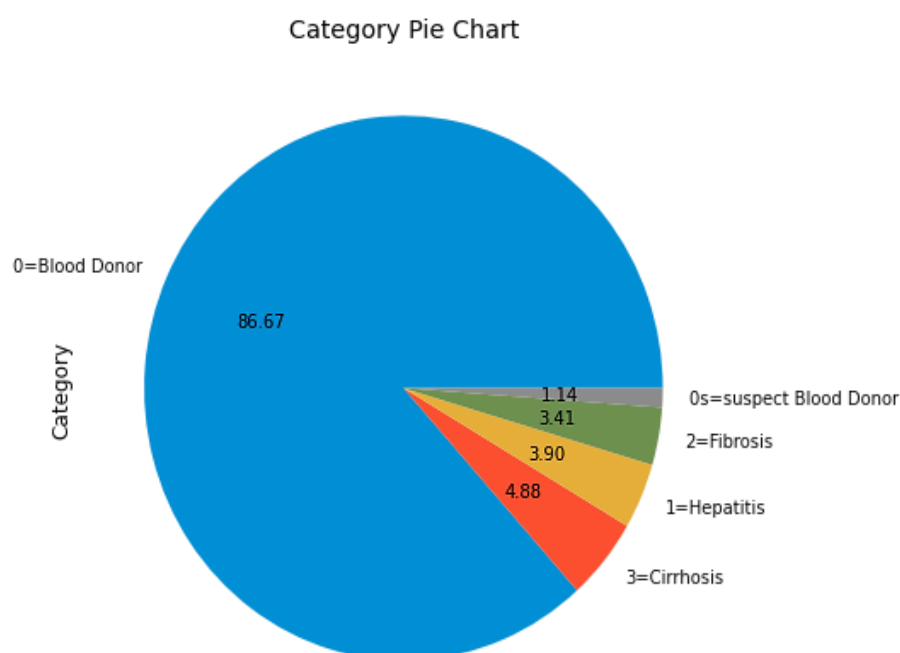


Figure 2. Imbalanced data in Category (Target)

3.2. Data Pre-processing

This phase will include lowering noisy values and replacing missing data for specified attributes. The experimental dataset is anticipated to have dealt with missing, inconsistent, and duplicate data. MinMaxScaler was imported from sklearn. Pre-processing to scale the data columns. With MinMaxScaler, each feature's minimum and maximum values were individually converted to 0 and 1, respectively, while all other values were changed to a decimal between 0 and 1.

3.2.1. SMOTE (Synthetic Minority Over-sampling Technique)

To address data imbalance, SMOTE was applied to generate synthetic minority class instances. This technique ensures that classifiers do not develop biases toward the majority class

3.2.2. ADASYN (Adaptive Synthetic Sampling Approach for Imbalanced Learning)

ADASYN was also implemented to enhance minority-class learning by generating synthetic examples. It adjusts the number of synthetic samples generated based on the learning difficulty of minority-class instances

3.3. Data Splitting

The various splitting methods utilized in this work are discussed in depth in this section.

3.3.1. K-fold cross-validation

Cross-validation (CV) is a prominent approach in machine learning. The primary objective of cross-validation is to create a consistent and trustworthy estimate of model performance, whereas statistical analysis findings will be generalized to an independent dataset [18]. In applied machine learning, it is typically used to evaluate the performance of an ML model on unseen data. For selecting model parameters, a technique is used that randomly splits the original sample into K equal-sized subsamples [19]. For each of the K subsamples, a single subsample is retained as validation data to test the model, while the remaining k-1 subsamples are used for model training.

The method is iterated based on the number of folds [20]. The value of k is commonly 5 or 10, although there is no hard and fast rule. If k is 10, the approach is referred to as the Stratified K-fold strategy. As this discrepancy lessens, the technique's bias decreases. Both 'k' values mentioned above are considered and utilized in this study.

3.3.2. Train-Test split

The train-test split randomly divides the dataset into train and test portions. The size of the dataset influences this strategy [21]. The data in the study was divided based on the test size. Initially, a test size of 0.2 was chosen, and the experiment was conducted. The experiment was then repeated using 0.3 as the test size. Ultimately, the results were compared and analyzed.

The choice of test size (0.2 and 0.3) in machine learning experiments is often based on empirical observations and best practices. These values are commonly used because they strike a balance between having enough data for training the model and sufficient data for evaluation. It is important to note that there is no hard rule for selecting a specific test size, which can vary depending on the problem and dataset. Experimentation and validation with different test sizes can provide insights into the optimal balance between training and evaluation data, ensuring reliable and robust model performance. In the case of using a test size of 0.2 or 0.3, the study's authors found that this ratio provided a good balance between training models effectively and having enough data to test the model's performance.

3.4. Feature Selection

The process of feature selection involves lowering the number of input variables. Its main objective is to choose a subset of the original features that will improve classification accuracy [22]. The effective use of feature selection methods for choosing features enhances inductive learning, either in terms of generalization capability, learning speed, or lowering the complexity [23]. The four fundamental phases in the feature selection process are subset creation, subset assessment, stopping criterion, and outcome validation [24]. This study uses 5-fold and 10-fold cross-validation on the training set to examine the selected feature.

3.5. Data Oversampling

Data imbalance is an uneven distribution of classes or labels in a given dataset. The most straightforward solution to this problem is the resampling approach, which involves adding records to the minority class and removing them from the majority class [25]. In this scenario, researchers implemented conventional oversampling techniques, such as SMOTE and ADASYN, to add records to the minority classes, specifically 0, 0s, 1, 2, and 3 in the category. Oversampling strategies improve fairness while lowering prediction accuracy [26]. After applying the oversampling techniques, it is noticeable that all features are balanced within the category, as clearly illustrated in Figure 3.

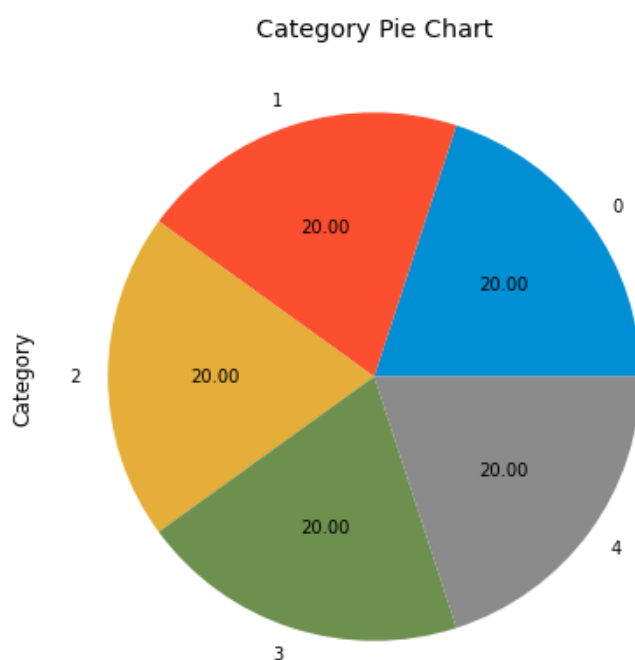


Figure 3. Data after oversampling the category (Target)

It is common for real-life application datasets to be biased or imbalanced. This means that the dataset may have an unequal representation of certain classes or attributes, which can affect the performance of models.

Several techniques can be employed to address biased or imbalanced datasets, which are also discussed in Section 3.4. and 3.5.

1. **Resampling Techniques:** Resampling approaches might be utilized to balance the distribution of classes in a dataset. Two common methods are:
 - a. **Under-sampling:** randomly removing instances from the majority class to balance it with the minority class.
 - b. **Oversampling:** creating synthetic instances for the minority class using methods like SMOTE or ADASYN.
2. **Ensemble Techniques:** Ensemble techniques involve combining two or more models to increase the system's overall performance. This can be useful when the dataset is not balanced or biased.
3. **Feature Selection:** Feature selection approaches might be utilized to discover the most important characteristics for the classification task. This can help to mitigate the influence of irrelevant or biased dataset features.

Dealing with biased or imbalanced datasets in real life requires a careful approach. To address bias, we thoroughly analyze the dataset to identify potential biases based on demographics or data collection

methods. We then strive to collect additional data from underrepresented groups to ensure better representation. In cases of class imbalance, we employ techniques such as oversampling or undersampling the required class to balance the dataset. We also experiment with different algorithms and evaluation metrics that effectively handle imbalanced data. Regularly monitoring and re-evaluating the dataset and models help us mitigate bias and improve the system's overall performance.

3.6. Classification Process

This study applied seven machine learning techniques and an artificial neural network to predict HCV. An ensemble model comprising seven machine learning algorithms was also built for more accurate outcomes. Default parameters were used when running the code. It means that the values assigned to the parameters are the pre-set or default values provided by the software or package. Default parameters in machine learning offer simplicity, efficiency, and generalization. They strike a balanced choice suitable for diverse scenarios, ensuring efficient model convergence, preventing overfitting, and saving time and resources by eliminating the need for extensive hyperparameter tuning. Additionally, they serve as a valuable baseline for initial model performance assessment before customization.

This study primarily focuses on the various methods used on the data before it is trained to the machine learning models. Specifically, the study compares and evaluates the effectiveness of oversampling, test size, and cross-validation techniques on small and imbalanced datasets. To showcase the effectiveness of these techniques, seven popular ML algorithms and an ANN are utilized in the work. The performance of these algorithms is evaluated after applying the different methods to the data. This comparison is expected to provide valuable insights into the strengths and limitations of each technique, helping researchers and practitioners choose the most suitable one for their specific application. Many recent studies have employed deep learning techniques. As mentioned, the study delves deeper into the implementation and comparison of the proposed methods across all the algorithms employed. It focuses on seven popular ML algorithms and ANN to showcase the effectiveness of the proposed methods. However, the same techniques can also be applied to other machine-learning algorithms. The methods proposed in the study can be applied to any ML or Deep learning algorithm as they are applied to the dataset before training the model.

3.7. Performance Evaluation

Evaluating a model is crucial in developing an effective machine-learning model. 'Accuracy' should be the most frequently used classification evaluation metric. One could think the model is adequate when

the accuracy rate is 99 percent. However, this is not always the case, and it can sometimes be deceptive. To evaluate the models, the study considers four key classification metrics [27]: Precision, Recall, Matthews Correlation Coefficient (MCC), and Accuracy. These are frequently used skewed assessment measures and should not be utilized without a thorough understanding of the biases and the associated identification of chance or base-case values of the statistic [28].

The outcomes of the Seven Machine Learning Algorithms and one Neural Network will be examined using the aforementioned criteria to determine which model is the best. The choice of evaluation metrics depends on the problem being addressed and the characteristics of the dataset. In this study, the authors have chosen the aforementioned metrics for evaluation. These metrics are commonly used in binary classification problems and provide insight into different aspects of model performance.

Recall is the proportion of true positives among the actual positives, whereas precision measures the proportion of true positives among predicted positives. The MCC is a correlation coefficient that measures the quality of binary or multiclass classifications. It considers true positives, false positives, and false negatives, providing a measure of correlation that reflects the agreement between the predicted and actual values. Accuracy measures the proportion of correct predictions overall. Using these evaluation metrics can help assess the performance of the machine learning models in identifying instances of the hepatitis C virus. However, it is essential to note that relying solely on accuracy may not provide a comprehensive picture of model performance, particularly in cases where the dataset is imbalanced. In this study, the authors found high accuracy before oversampling the dataset. However, this may be misleading as the imbalanced dataset may have led to biased predictions. By oversampling the dataset using techniques such as SMOTE or ADASYN, the authors have attempted to balance the dataset and enhance the model's performance.

4. Result and Discussion

In this section, the study describes the experiments conducted to evaluate the effectiveness of the ML framework in diagnosing HCV illness. As previously mentioned, the HCV dataset from the UCI ML Repository was utilized to build the framework. An Apple M1 machine with macOS BigSur operating system and 8GB of primary RAM was used for the experiments. Python programming was used to experiment. The first half of this chapter focuses on assessing the impact of various oversampling methods. While conducting a series of tests, the second section focuses on the impact of various data-splitting approaches on classification performance.

4.1. Comparison of Performance with Data Splitting Methods

The effectiveness of an ML model heavily relies on the quality of the data and the methodology used to process it [29]. Therefore, understanding how different data-splitting methods impact the performance of an ML model is crucial in improving modeling accuracy. To evaluate the effect of various data partitioning strategies on the performance of an ML model, this study compared different approaches using real HCV data. The approaches included random splitting with ratios of 80:20 and 70:30 for the train and test sets, as well as $k = 5$ and $k = 10$. Oversampling methods (SMOTE and ADASYN) were used in each iteration to obtain optimal results. Both oversampling techniques were used for each type of data splitting to ensure data completeness.

Table 3. Precision vs. Recall vs. MCC vs. Accuracy at $k=5,10$ and 80:20 split (SMOTE)

Model	Data Split - 80:20 (SMOTE)							
	k=5				k=10			
	Precision	Recall	MCC	Accuracy	Precision	Recall	MCC	Accuracy
KNN	82	88	89	90.24	80	83	87	90.81
SVC	81	83	85	88.56	84	79	86	90.06
DT	86	86	88	90.62	88	91	86	90.99
RF	96	93	96	96.81	97	97	95	97
LGR	75	80	77	86.11	75	84	82	86.3
XGB	96	88	93	96.06	94	96	95	96.4
GNB	76	71	80	84.61	81	72	82	84.43

Table 4. Precision vs. Recall vs. MCC vs. Accuracy at $k=5,10$ and 70:30 split (SMOTE)

Model	Data Split - 70:30 (SMOTE)							
	k=5				k=10			
	Precision	Recall	MCC	Accuracy	Precision	Recall	MCC	Accuracy
KNN	86	92	90	91.5	89	94	92	94.5
SVC	80	81	87	89.62	81	80	86	91
DT	87	86	89	91	88	91	90	94.38
RF	98	98	98	98	97	96	97	98.25
LGR	72	80	79	84.88	73	73	80	86.11
XGB	97	97	97	97.5	97	94	95	96.5
GNB	76	75	83	83.25	79	57	80	85.11

Table 5. Precision vs. Recall vs. MCC vs. Accuracy at k=5,10 and 80:20 split (ADASYN)

Model	Data Split - 80:20 (ADASYN)							
	k=5				k=10			
	Precision	Recall	MCC	Accuracy	Precision	Recall	MCC	Accuracy
KNN	80	94	87	90.62	82	90	88	89.87
SVC	79	75	81	90.99	86	72	86	90.99
DT	89	86	85	90.81	83	90	90	90.99
RF	94	95	94	95.3	97	94	97	96.06
LGR	67	78	78	85.18	76	74	81	82.93
XGB	91	92	92	94.93	92	91	96	95.67
GNB	76	69	77	87.24	83	72	85	81.24

Table 6. Precision vs. Recall vs. MCC vs. Accuracy at k=5,10 and 70:30 split (ADASYN)

Model	Data Split - 70:30 (ADASYN)							
	k=5				k=10			
	Precision	Recall	MCC	Accuracy	Precision	Recall	MCC	Accuracy
KNN	89	91	92	93	88	91	93	92
SVC	80	80	85	90.62	80	81	84	91.5
DT	90	90	89	92	90	89	90	91.5
RF	96	98	97	97.5	97	97	98	97.25
LGR	71	76	78	86.75	76	74	77	88.12
XGB	94	97	96	96.5	95	94	95	96.88
GNB	74	57	76	87.75	73	56	78	87.38

The evaluation metrics of the ML algorithms are presented in Tables 3, 4, 5, and 6. It can be seen from the visualization line graphs that the results are influenced by the splitting strategies, k value, and oversampling methodology used in this study. However, it does not imply giving 99 percent to train and the rest to test. Remember that dividing the model into uneven proportions will result in overfitting or underfitting. The accuracy of ML algorithms before cross-validation is shown in Figure 4.

The value of 'k' affects the models' performance metrics since one-fold is used for evaluation, and k-1 folds are used for training. After analyzing the results of all features and data splitting strategies, the Random Forest model emerged as the best, with the highest accuracy, Recall, MCC, and Precision in this study. Random Forest is preferred over other models because it can look for the best feature from a random subset of characteristics instead of the most important one when splitting a node. Tables 4 and 6 demonstrate

that the 70:30 data-splitting strategy improved classifier performance in four performance metrics, indicating that oversampling with SMOTE and ADASYN is more effective. Furthermore, the RF and XGB classes outperformed the DT and KNN classifiers in the HCV dataset. The results of the artificial neural network are presented in Figures 5 to 8 and are compared in the following paragraph.



Figure 4. Accuracy of ML algorithms before cross-validation

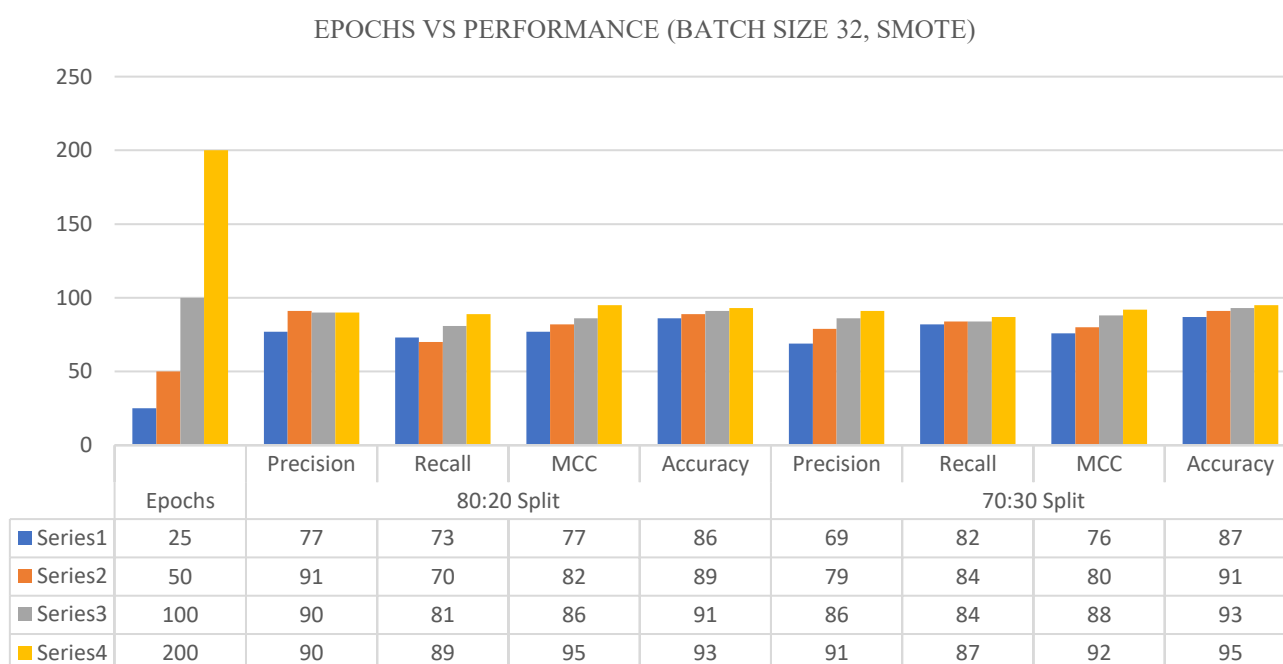


Figure 5. Visualization of Precision vs. Recall vs. MCC vs. Accuracy of ANN with Batch Size=32 (SMOTE)

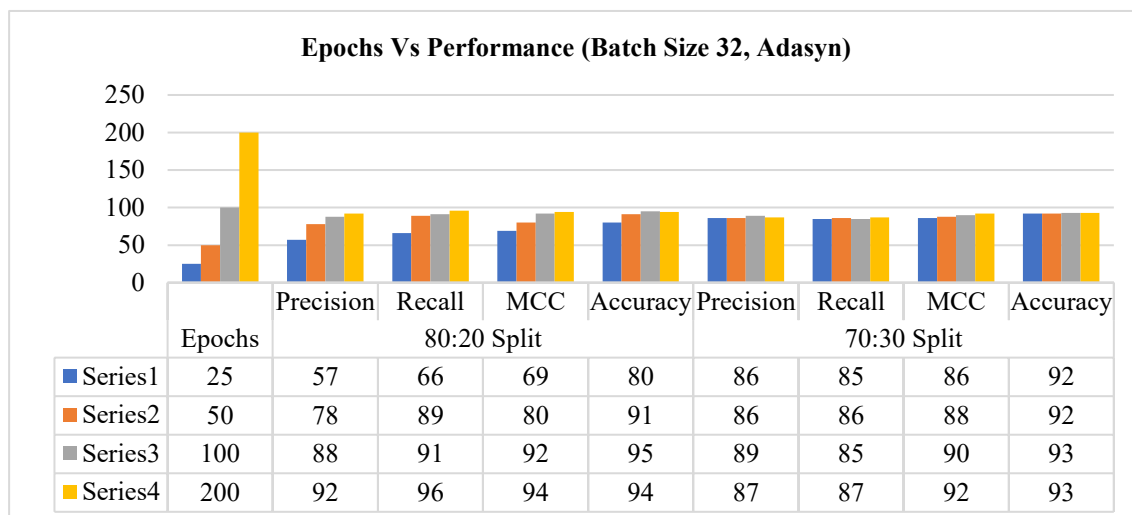


Figure 6. Visualization of Precision vs. Recall vs. MCC vs. Accuracy of ANN with Batch Size=32 (ADASYN)

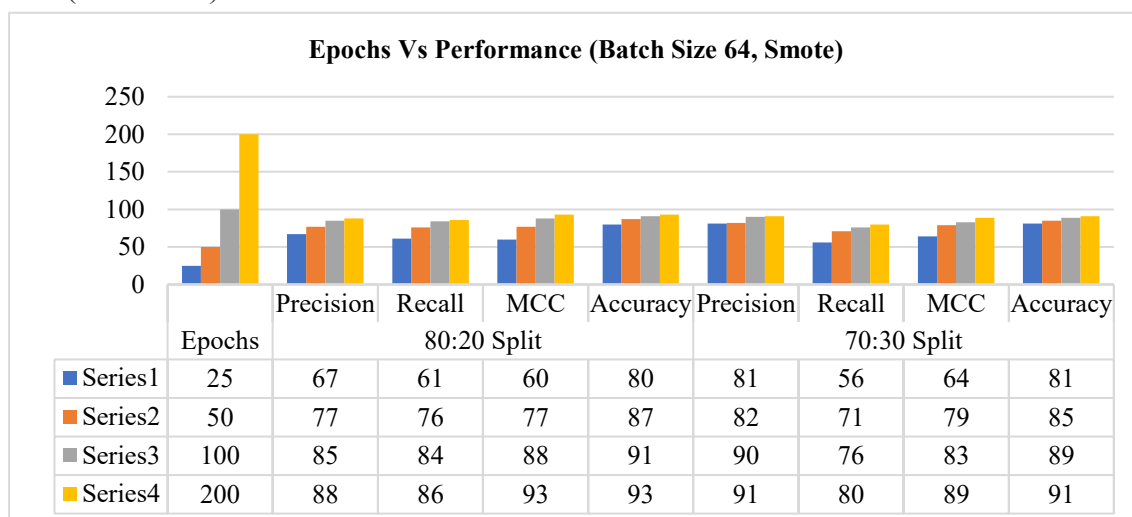


Figure 7. Visualization of Precision vs. Recall vs. MCC vs. Accuracy of ANN with Batch Size=64 (SMOTE)

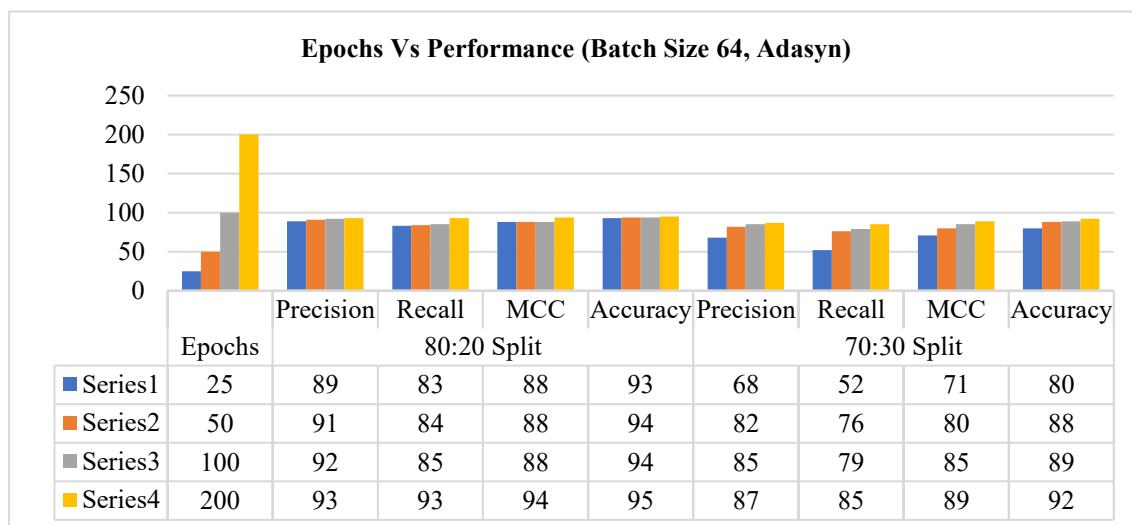


Figure 8. Visualization of Precision vs. Recall vs. MCC vs. Accuracy of ANN with Batch Size=64 (ADASYN)

According to Figure 6, an artificial neural network trained with the ADASYN oversampling technique on 80:20 split data with a batch size of 32 and 200 epochs achieved outstanding performance with 94 percent MCC, 92 percent precision, 96 percent recall, and 94 percent accuracy. On the other hand, the ADASYN oversampling technique performed better in the second case, as shown in Figure 7, with 94 percent MCC, 93 percent precision, 93 percent recall, and 95 percent accuracy in 80:20 split data with a batch size of 64 and 200 epochs. In summary, the artificial neural network trained on a reduced dataset performed better on the HCV dataset but took longer to develop compared to the model trained on the original dataset.

4.2. Significance of Basic and Advanced Ensemble Methods

Ensemble approaches are considered the most advanced solutions for many machine learning problems. These techniques involve training multiple models and combining their predictions to achieve a more accurate result. This study introduces the concept of ensemble learning, examines conventional, cutting-edge, and novel ensemble approaches, and explores the current issues and developments in the field [30]. There are several techniques, and a selection of them are listed below.

4.2.1. Basic Ensemble Techniques

4.2.1.1. Averaging

The averaging technique involves generating multiple predictions for each data point, and the overall prediction is obtained by averaging the results of all the models. The average accuracy of this approach is provided in Table 7.

Table 7. Average - Accuracy at k=5 & 10 at 80:30 & 70:30 split with SMOTE & ADASYN

Ensemble Method	With Smote				With ADASYN			
	80-20 split		70-30 split		80-20 split		70-30 split	
	k=5	k=10	k=5	k=10	k=5	k=10	k=5	k=10
Average	90.43	90.85	90.82	92.26	90.72	89.67	92.01	92.09

4.2.1.2. Max Voting

For classification problems, the max voting approach is typically applied. This method makes predictions for each data point using various models. Each model's predictions are regarded as a "vote." Most models' projections serve as the basis for the conclusion. The voting method results with all seven models as estimators are shown in Table 8.

Table 8. Voting - Accuracy at k=5 & 10 at 80:30 & 70:30 split with SMOTE & ADASYN

Ensemble Method	With Smote				With ADASYN			
	80-20 split		70-30 split		80-20 split		70-30 split	
	k=5	k=10	k=5	k=10	k=5	k=10	k=5	k=10
Voting	98.87	99.06	97.37	97.75	98.88	98.49	98.37	99

4.2.2. Advanced Ensemble Techniques

Although there are numerous advanced ensemble approaches, this paper primarily focuses on stacking.

4.2.2.1. Stacking

Stacking is an advanced technique in ensemble learning where a new model is built using predictions from other models, as shown in Figure 9. This method involves meta-learning, which minimizes generalization errors by reducing generalizer bias and creating an asymptotically optimal learning system. The stacked ensemble model makes predictions on the test set by combining the predictions of the base models. The training data is divided into K-folds, and a base model is fitted to K-1 parts of the data for each fold to predict the K-th component. The base model's performance is determined by fitting it to the training dataset. The basic classifiers use the training set to train the layers and build a new feature matrix for the meta-classifier layer. The meta-classifier layer picks and predicts the final estimate for classification.

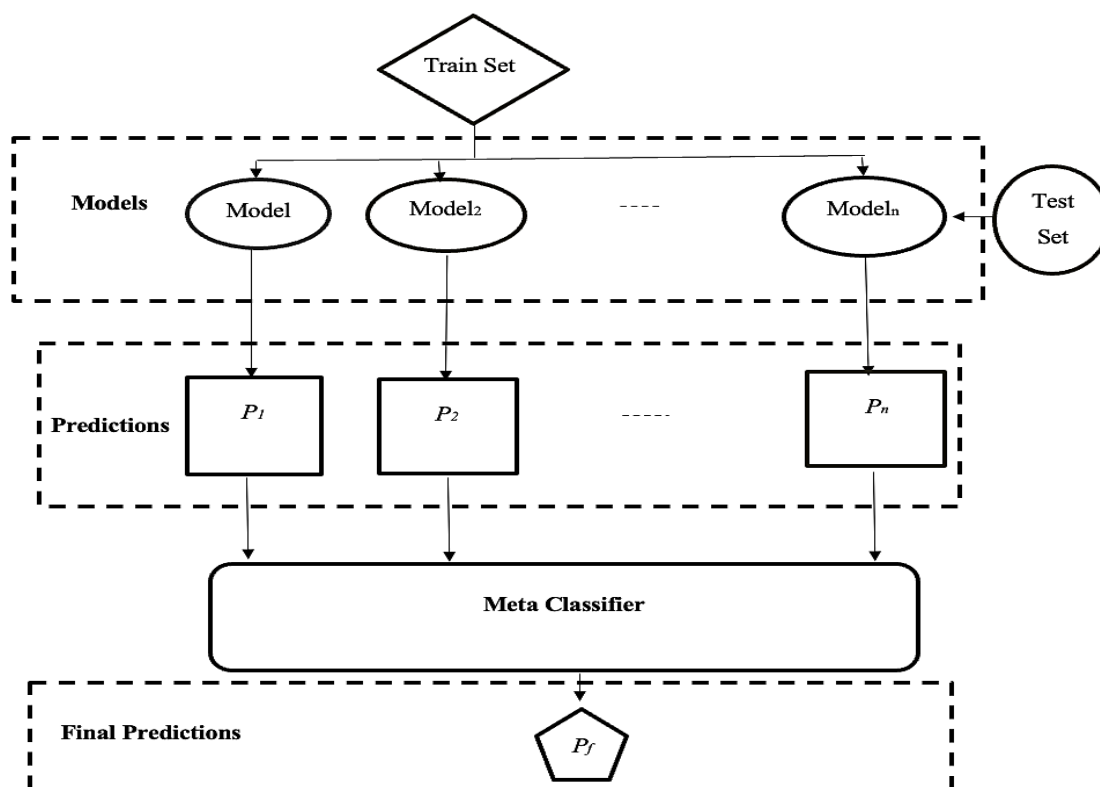
**Figure 9.** The architecture of stacking

Table 9. Stacking - Accuracy at k=5 & 10 at 80:30 & 70:30 split with SMOTE & ADASYN

Final Estimator	Smote				ADASYN			
	80-20 split		70-30 split		80-20 split		70-30 split	
	k=5	k=10	k=5	k=10	k=5	k=10	k=5	k=10
KNN	98.87	99.06	99.88	99.12	99.25	98.87	99.12	99.62
SVC	99.44	99.25	98.75	99.5	99.44	99.44	99.38	99.12
DT	99.25	99.81	99.88	99.62	99.62	99.62	99.88	99.5
RF	99.62	99.44	99.38	99.75	99.99	99.44	99.75	99.5
LGR	99.62	99.06	99.62	99.62	99.44	99.62	99.38	99.25
XGB	99.81	99.44	99.75	99.88	99.5	99.81	99.62	99.75
GNB	99.44	99.06	99.12	99.62	99.44	99.44	99.5	99.5

Table 9 shows that the new model created by stacking and using XGB as the final estimator yields the best accuracy for practically every splitting approach and k value. The performance of models using the final estimators KNN, DT, LGR, and RF is comparable across the columns. Although it is less accurate compared to its rivals, a model with GNB as the final estimator still produces good accuracy. This study aims to enhance the accuracy of ML classification algorithms for HCV diagnosis by presenting an ML framework, an ANN-based model, and an ensemble model. This project aims to develop strategies to improve the accuracy of ML classification algorithms for diagnosing hepatitis C using real-time HCV data. To do this, the dataset was first balanced by oversampling it using SMOTE and ADASYN. The performance of classifiers was then evaluated in relation to the impact of the data-splitting strategy. Moreover, an ensemble model was created, which outperforms an individual model in terms of accuracy, and its precision is affected by the final estimator.

The following summarizes the results of the experiments described in this work:

- The 10-fold cross-validation with the k-folds approach and the 70:30 splitting of the HCV information set were the best options for ML modeling when applying classification algorithms that utilized various splitting methods. It can be seen that when the RF classifier is used with 10-fold cross-validation and k-folds, along with a 70:30 splitting, the best results are obtained.
- The artificial neural network with a smaller dataset performed best in comparison to the HCV and HCV datasets. Although the task required to construct the model took longer, it was faster than one using the original dataset.
- Regarding ANN, the ADASYN upsampling algorithm with batch sizes of 32, 64, and 200 epochs, in an 80:20 split of the HCV dataset, was the best choice.

- Finally, it can be demonstrated that choosing XGB as the final estimator for Ensemble techniques yields the highest accuracy for almost every splitting strategy and k value.

Table 10. Comparison of performance between Literature Survey and Present work

Model	Highest Accuracy in Literature Survey	Accuracy of this Work
KNN	96.5	99.88
SVC	96.7	99.5
DT	-	99.88
RF	99.9	99.91
LGR	97.9	99.62
XGB	-	99.88
GNB	96	99.62

In this comparative analysis, our work demonstrates exceptional accuracy across multiple machine learning models, surpassing the highest reported accuracy in literature. Notably, our K-Nearest Neighbors (KNN) and Decision Tree (DT) models achieve an impressive 99.88% accuracy, while Support Vector Classifier (SVC) and Logistic Regression (LGR) models exhibit accuracy rates of 99.5% and 99.62%, respectively. Our Random Forest (RF) model also outperforms the literature's best-reported accuracy, achieving an outstanding 99.91%. These results demonstrate the significant advancements in our approach and the robustness of our models in the task presented in Table 10.

5. Conclusions and Future Work

In order to classify and predict HCV-infected individuals based on the enrolled features, a machine learning framework based on KNN, SVM, DT, RF, LGR, XGB, and GNB, an ANN, and an ensemble model is provided in this research. Data upsampling is first performed to balance the dataset. The dataset was also examined by implementing all the attributes or characteristics of the employed models to modify the enrolled data without feature selection. Then, an ANN and an ensemble model were implemented. The findings show that using a perfect final estimator significantly improves the precision or exactness of the ensemble model. The classification accuracy scores for KNN, SVM, Random Forest, Neural Network, Naive Bayes, and Logistic Regression are 96.5%, 96.7%, 97.3%, 97.1%, 96.0%, and 97.9%, respectively. This study demonstrates the effectiveness of machine learning in diagnosing Hepatitis C. By using SMOTE and ADASYN for data balancing and leveraging ensemble techniques, and we achieved state-of-the-art classification performance. In the future, a systematic investigation will be conducted on a more comprehensive

and balanced dataset with a substantial number of patients, distinct attributes, and robust deep learning algorithms, and these algorithms will be combined with SHAP to yield a more accurate model.

Funding: The authors declare that no funding was received for this research.

Declaration of Competing Interest: The authors declare they have no known competing interests.

Research Involving Human/Animal Participants: This article does not involve any studies conducted by authors on animals or human participants.

References

- [1] WHO. Hepatitis C [Online] Available: <https://www.who.int/news-room/fact-sheets/detail/hepatitis-c>
- [2] Wendy Spearman, Geoffrey M Dusheiko, Margaret Hellard, and M. Sonderup, "Hepatitis C," *Lancet*, vol. 394, no. 10207, pp. 1451-1466, 2019, doi: 10.1016/S0140-6736(19)32320-7.
- [3] H. Rumgay *et al.*, "Global, regional and national burden of primary liver cancer by subtype," *European Journal of Cancer*, vol. 161, pp. 108-118, 2022, doi: <http://dx.doi.org/10.1016/j.ejca.2021.11.023>.
- [4] D. A. Mahmood and S. A. Aminfar, "Efficient Machine Learning and Deep Learning Techniques for Detection of Breast Cancer Tumor," *BioMed Target Journal*, vol. 2, no. 1, pp. 1 - 13, 06/30 2024, doi: <https://doi.org/10.59786/bmtj.211>.
- [5] F. J. Kaunang, "A comparative study on hepatitis C predictions using machine learning algorithms," *8ISC Proceedings: Technology*, pp. 33-42, 2022.
- [6] L. Syafaah, Z. Zulfatman, and M. Lestandy, "Comparison of machine learning classification methods in hepatitis C virus," *JOIN (Jurnal Online Informatika)*, vol. 6, no. 1, pp. 73-78, 2021, doi: <http://dx.doi.org/10.15575/join.v6i1.719>.
- [7] A. Orooji and F. Kermani, "Machine learning based methods for handling imbalanced data in hepatitis diagnosis," *Frontiers in Health Informatics*, vol. 10, no. 1, p. 57, 2021, doi: <http://dx.doi.org/10.30699/fhi.v10i1.259>.
- [8] K. Ahammed, M. S. Satu, M. I. Khan, and M. Whaiduzzaman, "Predicting infectious state of hepatitis c virus affected patient's applying machine learning methods," in *2020 IEEE Region 10 Symposium (TENSYP)*, 2020: IEEE, pp. 1371-1374, doi: <http://dx.doi.org/10.1109/TENSYP50017.2020.9230464>.
- [9] S. C. Nandipati, C. XinYing, and K. K. Wah, "Hepatitis C virus (HCV) prediction by machine learning techniques," *Applications of modelling and simulation*, vol. 4, pp. 89-100, 2020.
- [10] N. H. Barakat, S. H. Barakat, and N. Ahmed, "Prediction and staging of hepatic fibrosis in children with hepatitis C virus: A machine learning approach," *Healthcare Informatics Research*, vol. 25, no. 3, pp. 173-181, 2019, doi: <http://dx.doi.org/10.4258/hir.2019.25.3.173>.
- [11] G. J. Dore, "Monitoring hepatitis C elimination among people who inject drugs: A broader approach is required," *International Journal of Drug Policy*, vol. 137, p. 104712, 2025, doi: <http://dx.doi.org/10.1016/j.drugpo.2025.104712>.
- [12] N. Gupta *et al.*, "Best practices for hepatitis C linkage to care in pregnant and postpartum women: perspectives from the Treatment In Pregnancy for Hepatitis C Community of Practice," *American journal of obstetrics and gynecology*, vol. 231, no. 4, pp. 377-385, 2024, doi: <http://dx.doi.org/10.1016/j.ajog.2024.06.028>.
- [13] I. A. Naqid *et al.*, "Hepatitis B and C virus infection among couples undergoing premarital screening in Iraqi Kurdistan," *IJID regions*, vol. 14, p. 100492, 2025, doi: <http://dx.doi.org/10.1016/j.ijregi.2024.100492>.
- [14] N. Posuwan *et al.*, "Hepatitis B prevalence in an endemic area of hepatitis C virus: A population-based study implicated in hepatitis elimination in Thailand," *Journal of Virus Eradication*, vol. 10, no. 4, p. 100577, 2024, doi: <http://dx.doi.org/10.1016/j.jve.2024.100577>.
- [15] G. Hoffmann, A. Bietenbeck, R. Lichtinghagen, and F. Klawonn, "Using machine learning techniques to generate

- laboratory diagnostic pathways—a case study," *Journal of Laboratory and Precision Medicine*, vol. 3, no. 6, 2018, doi: <http://dx.doi.org/10.21037/jlpm.2018.06.01>.
- [16] R. Lichtinghagen, D. Pietsch, H. Bantel, M. P. Manns, K. Brand, and M. J. Bahr, "The Enhanced Liver Fibrosis (ELF) score: normal values, influence factors and proposed cut-off values," *Journal of hepatology*, vol. 59, no. 2, pp. 236-242, 2013, doi: <http://dx.doi.org/10.1016/j.jhep.2013.03.016>.
- [17] D. Chicco and G. Jurman, "An ensemble learning approach for enhanced classification of patients with hepatitis and cirrhosis," *IEEE Access*, vol. 9, pp. 24485-24498, 2021, doi: <http://dx.doi.org/10.1109/ACCESS.2021.3057196>.
- [18] G. Valente, A. L. Castellanos, L. Hausfeld, F. De Martino, and E. Formisano, "Cross-validation and permutations in MVPA: Validity of permutation strategies and power of cross-validation schemes," *NeuroImage*, vol. 238, p. 118145, 2021, doi: <http://dx.doi.org/10.1016/j.neuroimage.2021.118145>.
- [19] J. Wiecezorek, C. Guerin, and T. McMahon, "K-fold cross-validation for complex sample surveys," *Stat*, vol. 11, no. 1, p. e454, 2022, doi: <http://dx.doi.org/10.1002/sta4.454>.
- [20] M. Rafał, "Cross validation methods: Analysis based on diagnostics of thyroid cancer metastasis," *ICT Express*, vol. 8, no. 2, pp. 183-188, 2022, doi: <http://dx.doi.org/10.1016/j.ict.2021.05.001>.
- [21] Q. H. Nguyen et al., "Influence of data splitting on performance of machine learning models in prediction of shear strength of soil," *Mathematical Problems in Engineering*, vol. 2021, no. 1, p. 4832864, 2021, doi: <http://dx.doi.org/10.1155/2021/4832864>.
- [22] M. Mandal, P. K. Singh, M. F. Ijaz, J. Shafi, and R. Sarkar, "A tri-stage wrapper-filter feature selection framework for disease classification," *Sensors*, vol. 21, no. 16, p. 5571, 2021, doi: <http://dx.doi.org/10.3390/s21165571>.
- [23] M. Abdulla and M. T. Khasawneh, "Integration of aggressive bound tightening and Mixed Integer Programming for Cost-sensitive feature selection in medical diagnosis," *Expert Systems with Applications*, vol. 187, p. 115902, 2022, doi: <http://dx.doi.org/10.1016/j.eswa.2021.115902>.
- [24] M. Ajdani and H. Ghaffary, "Improving network intrusion detection by identifying effective features based on probabilistic dependency trees and evolutionary algorithm," *Cluster Computing*, vol. 25, no. 5, pp. 3299-3311, 2022, doi: <http://dx.doi.org/10.1007/s10586-022-03564-9>.
- [25] R. Mohammed, J. Rawashdeh, and M. Abdullah, "Machine learning with oversampling and undersampling techniques: overview study and experimental results," in *2020 11th international conference on information and communication systems (ICICS)*, 2020: IEEE, pp. 243-248, doi: <http://dx.doi.org/10.1109/ICICS49469.2020.239556>.
- [26] S. Rančić, S. Radovanović, and B. Delibašić, "Investigating oversampling techniques for fair machine learning models," in *International Conference on Decision Support System Technology*, 2021: Springer, pp. 110-123, doi: http://dx.doi.org/10.1007/978-3-030-73976-8_9.
- [27] V. P. Torane, P. Patil, S. Wanjare, and G. Nataraj, "Prevalence of HIV, syphilis, hepatitis B, and hepatitis C in patients attending STI/RTI clinic of a tertiary care teaching hospital," *Journal of the National Medical Association*, vol. 116, no. 3, pp. 211-218, 2024, doi: <http://dx.doi.org/10.1016/j.jnma.2024.01.001>.
- [28] F. Correia et al., "Clinical outcomes after hepatitis C treatment in patients with advanced chronic liver disease," *Journal of Hepatology*, vol. 78, p. S1189, 2023, doi: [http://dx.doi.org/10.1016/S0168-8278\(23\)03342-1](http://dx.doi.org/10.1016/S0168-8278(23)03342-1).
- [29] T. Minami et al., "Machine learning for individualized prediction of hepatocellular carcinoma development after the eradication of hepatitis C virus with antivirals," *Journal of hepatology*, vol. 79, no. 4, pp. 1006-1014, 2023, doi: <http://dx.doi.org/10.1016/j.jhep.2023.05.042>.
- [30] K. Zachou et al., "Treatment responses and outcomes in patients with autoimmune hepatitis and concomitant features of non-alcoholic fatty liver disease," *JHEP Reports*, vol. 5, no. 8, p. 100778, 2023, doi: <http://dx.doi.org/10.1016/j.jhepr.2023.100778>.