COMPARATIVE ANALYSIS OF MACHINE LEARNING ALGORITHMS FOR PREDICTING THE MORTALITY RATE OF HEPATITIS C VIRUS

ABSTRACT

Hepatitis C is an infectious disease that affects the liver. Some of the symptoms are fatigue, vomiting, fever, loss of appetite, nausea among others and its main mode of transmission is contact with the blood of host. At the initial stage of the infection, an individual is asymptotic with little symptoms. difficulties posed by early detection at an early stage and limited clinical diagnosis has forced medical practitioners to rely on comparing the results of the blood test of patients with another who has similar symptoms, after which, treatment would be administered. Liver biopsy have been used over the years for the detection and staging of liver fibrosis. The disadvantages of this method are the results are not fully reliable due to sampling errors, it causes pain to patients and invasive. To bypass these limitations, non-invasive diagnostic biomarkers and different imaging techniques have been deployed to monitor liver conditions of patients.

The dataset used in this research, collected from the UCI Machine Learning Expository on Hepatitis C, which is made up of 615 instances will be used in the prediction of the presence of HCV or not. The dataset contained missing values and the mean imputation by chained equations (mice) was used in filling the missing values. In the selection of important features, four different methods were considered, which are: the Pearson correlation coefficient, the Recursive Feature Elimination, select best k and Mutual Information Gained. The result of feature selection showed that 6 features which played significant role in the prediction are “AST”, “BIL”, “GGT”, “CHE”, “ALB” and “ALT”. The imbalance class was handled using SMOTE and upSample, hence two scenarios were treated for each of the five machine learning algorithms namely: Logistic Regression, Naïve Bayes, K Nearest Neighbor, Random Forest and XGBoost. The tools used are R and Python, and their performances compared using accuracy, precision, and area under the curve.

For the R tool, the Random Forest (SMOTE) is a better performer with an accuracy of 97.56%, precision of 0.9907 and an area under the curve of 0.96.

Also, for Python, the Random Forest (SMOTE) had better performance with an accuracy of 96%, precision of 0.96 and an area under the curve of 0.91.

Finally, comparing the result obtained by R and Python, we note that result of RF using R had a better accuracy value of 0.9756, precision value of 0.9907. Although, both had the same auc value of 0.96.

Keywords: Recursive Feature Elimination, Mutual Information Gained, Logistic Regression, Naïve Bayes, K Nearest Neighbor, Random Forest and XGBoost.

1. INTRODUCTION

Hepatitis C is an infectious disease that affects the liver. Some of the symptoms are fatigue, vomiting, fever, loss of appetite, nausea among others. the main challenge posed by this disease is early detection and non-availability of vaccination. According to a report by the World Health Organizations in 2023, an estimated number of 58 million individuals are infected with the disease with children and adolescent accounting for 3.2 million. New infections occur at an average rate of 1.5 million every year [1]. Factors responsible for the prevalence include the use of unsafe medical procedures and practices via the transfusion of contaminated blood and needles, the expansion of intravenous drugs, immigration from endemic regions of the world and unsafe sexual practices [2], [3] and [4].

At the initial stage of the infection, an individual is asymptotic with little symptoms. According to [5], one of the major features of the HCV is that a large percentage of those infected are unaware of their status as quite a lot of new cases are many new cases which are not diagnosed. This hampers efforts

in controlling the infections and treatment of those infected as accurate estimates are unavailable. The difficulty posed by this early detection and limited clinical diagnosis has forced medical practitioners to rely on comparing the results of the blood test of patients with another who has similar symptoms, after which, treatment would be administered. Few years later, liver biopsy was introduced for the detection and staging of liver fibrosis to aid decision making. The disadvantages inherent in this method are results obtained are not fully reliable due to sampling errors, it causes pain to patients and invasive. To overcome these challenges, non-invasive diagnostic biomarkers and different imaging techniques have been deployed to monitor liver conditions of patients [6].
Several Researchers such as [7] have shown that using machine learning classification techniques and feature selection, which are branches of Artificial Intelligence, alongside the clinical decision support system results in useful prediction of several types of diseases.
[8] considered the classification of patients infected with hepatitis C using Machine Learning Algorithms. Using the Egyptian dataset on hepatitis C, which was obtained from the UCI machine learning repository, made up of 1385 instances of records of infected patients, 29 features were used. They evaluated the performances of five machine learning algorithms namely: kNN, RF, NB, DT, Logistic Regression, AVG and XGBoost. They obtained a result that kNN performed better than other classifier with an accuracy score of with 94.40%. [6] predicted HCV using machine learning techniques, using Python and R. the dataset used is the HCV dataset from the UCI machine learning expository for Egyptian patients' which consists of a total of 1385 instances with 29 features. The models considered are Gaussian Naïve Bayes, k-Nearest Neighbour, MLP, Random Forest, Support vector Machine, Adaboost and Bagging. In their result, they observed that the R tool performed better than Python with an interval of 49.77% to 50.12% in terms of accuracy as compared to the result generated by python which have an interval of 24.15% to 24.8%. In addition, the result of feature selection showed that the following 12 features played a significant role in the prediction. They are RNAEF, RNAEOT, RNA12, Nausea, Platelets, Epigastric Pain, ALT1, ALT48, RBC, ALT24, WBC counts and RNA4. [9] assessed the risks and factors that influences the progression of HCV infectious disease. Using the HCV from the UCI machine learning expository which is made up of 615 instances and 13 features, which are ALB, ALP, AST, ALT, BIL, CHE, CHOL, CREA, GGT, PROT, Age, Sex and so on. They trained and evaluated the following algorithms Logistic Regression, Decision Tree, Naïve Bayes, Random Forest, Support Vector Machine, and k-Nearest Neighbour. Using the performance metric of accuracy and AUROC curve, they had that the Random Forest performed better than other algorithms considered with an accuracy of 97.29% and AUROC curve of 0.998.
The motivation for this work stems from the challenge that exists in the detection of the HCV which inhibits its treatment and to the knowledge of the researcher, extensive work is yet to be carried out that compares the performances of use of R tool and Python in the Prediction of HCV using this dataset.

2. BACKGROUND OF THE STUDY

The dataset used is the UCI Machine Learning Expository on Hepatitis C, which is made up of 615 instances and 12 features which are either blood serum markers or clinical variables. The blood serum markers are albumin (ALB), alkaline phosphate (ALP), aspartate amino transferase (AST), alanine amino transferase (ALT), bilirubin (BIL), choline esterase (CHE), cholesterol (CHOL), creatinine (CREA), γ – glutamyl – transferase (GGT), and prothrombin time (PROT), while clinical variables are age and sex [10].
The information in table 1 shows the number of instances present in each class label.

Table 1: Distribution of Samples for each Class

Multi Class Label	**Sample Size**	**Binary Class Label**	**Sample Size**
0: Blood Donor	533	0: Blood Donor	540
0s: Suspect Blood Donor	7		
1: Hepatitis	24	1: Hepatitis	75
2: Fibrosis	21		
3: Cirrhosis	30		

The label in the original dataset is made 5 classes. In this paper our interest is in binary classification where the blood donor group and the suspect blood donor have been grouped together as the blood donor group while the remaining patients have been grouped as those having hepatitis. The number of samples in the Blood Donor group is over 700% above that of the Hepatitis group. This class imbalance has the potential of yielding biases during model training and in tackling this, the Synthetic Minority Oversampling Technique (SMOTE) and upSample were applied to the training set before training and fitting the model.

3. METHODOLOGY

3.1 Data Preprocessing

There are a total number of 31 missing entries and the breakdown of the counts according to the affected feature are shown in table 2 and visualized in figure 1.

Table 2: Frequency Distribution of Missing Entries

Frequency Distribution of Missing Entries					
ALB	ALP	ALT	CHOL	PROT	**TOTAL**
1	18	1	10	1	31

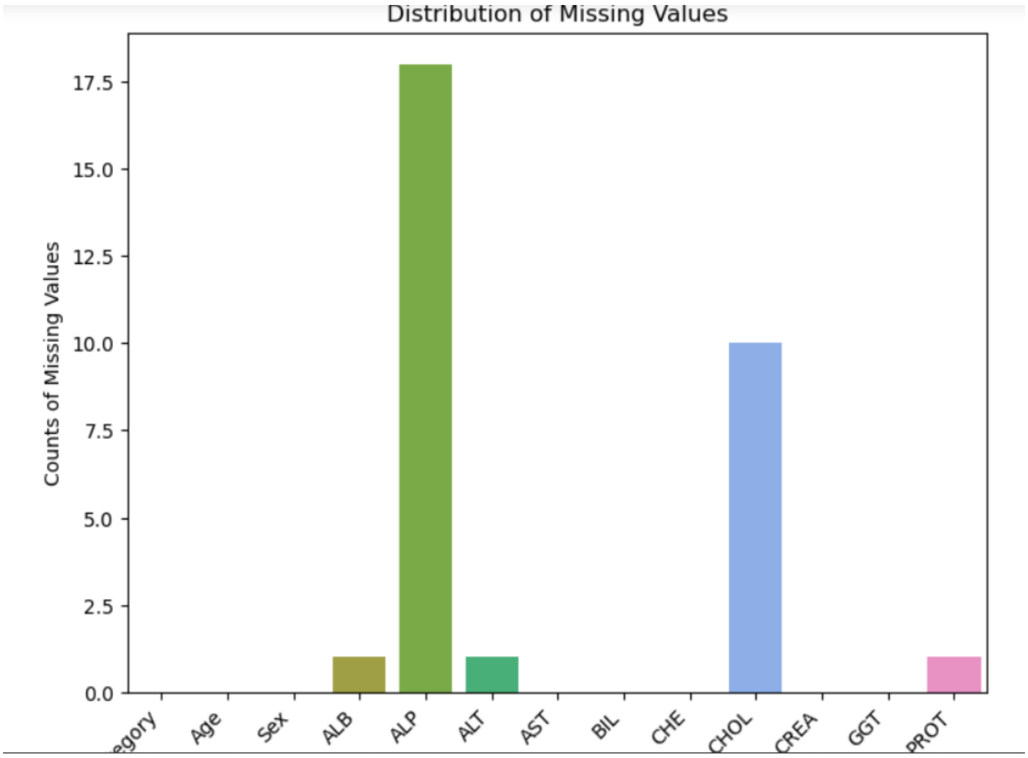

Figure 1: Distribution of Missing Entries

Following the suggestion put forward by [11] in which these missing entries can be ascribed to Missing at random (MAR), the multiple imputation by chained equations (MICE) will be used, where the number of iterations is 50. After which the mean values obtained is computed for the column and used in place for the missing values. This is to ensure robustness of the imputed value.

3.2 Data Analysis

As shown in the summary statistics in table 3, the minimum age of those who were sampled is 19 while the maximum age is 77. This indicates that those sampled were adults with the mean age of 47.4. The standard deviation of 10.1 shows that the data points of age are well dispersed.

Table 3: Summary Statistics of Numeric

Variables	Mean	SD	P0	P25	P50	P75	P100
Age	47.4	10.1	19	39	47	54	77
ALB	41.6	5.78	14.9	38.8	42	45.2	82.2
ALP	68.2	25.8	11.3	52.6	66.2	80.1	417.0
ALT	28.4	25.4	0.9	16.4	23	33.0	325.0
AST	34.8	33.1	10.6	21.6	25.9	32.9	324
BIL	11.4	19.7	0.8	5.3	7.3	11.2	254
CHE	8.20	2.21	1.42	6.94	8.26	9.59	16.4
CHOL	5.37	1.13	1.43	4.62	5.3	6.06	9.67
CREA	81.3	49.8	8	67	77	88	1079
GGT	39.5	54.7	4.5	15.7	23.3	40.2	651
PROT	72	5.42	44.8	69.3	72.2	75.4	90

Also, as shown by the pie chart in figure 2 where 0 denotes male and 1 denotes female, of the 615 patients, we observe that more male was sampled than female as 61% (377) are male while 39% (238) are female.

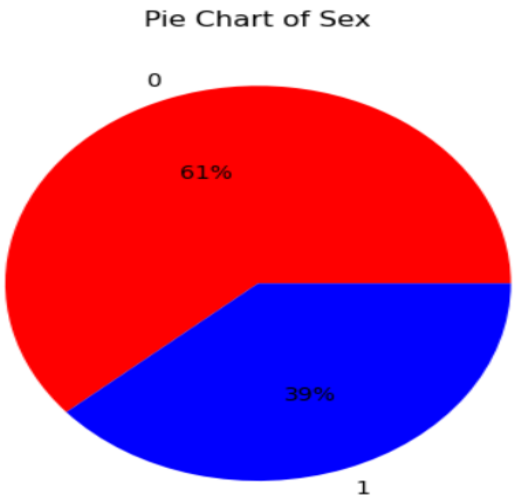

Figure 2: Pie Chart of Sex

3.3 Outlier Detection

Although some numeric features showed the presence of outliers were observed when visualized using boxplot as shown in figure 3, these were however left due to the sensitivity nature of the analysis and small sample size.

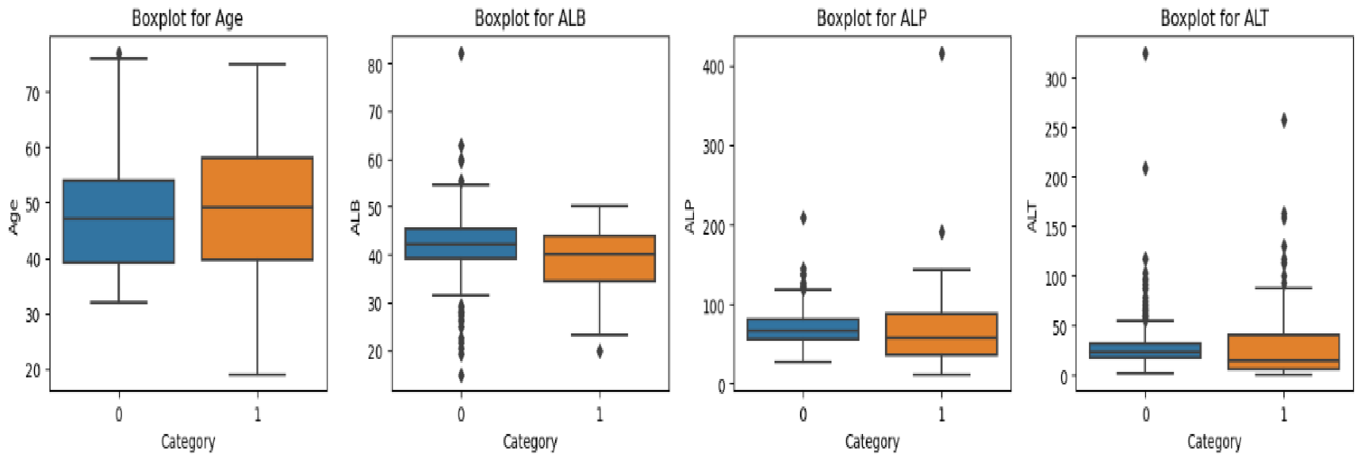

Figure 3a: Boxplot for Numeric Features

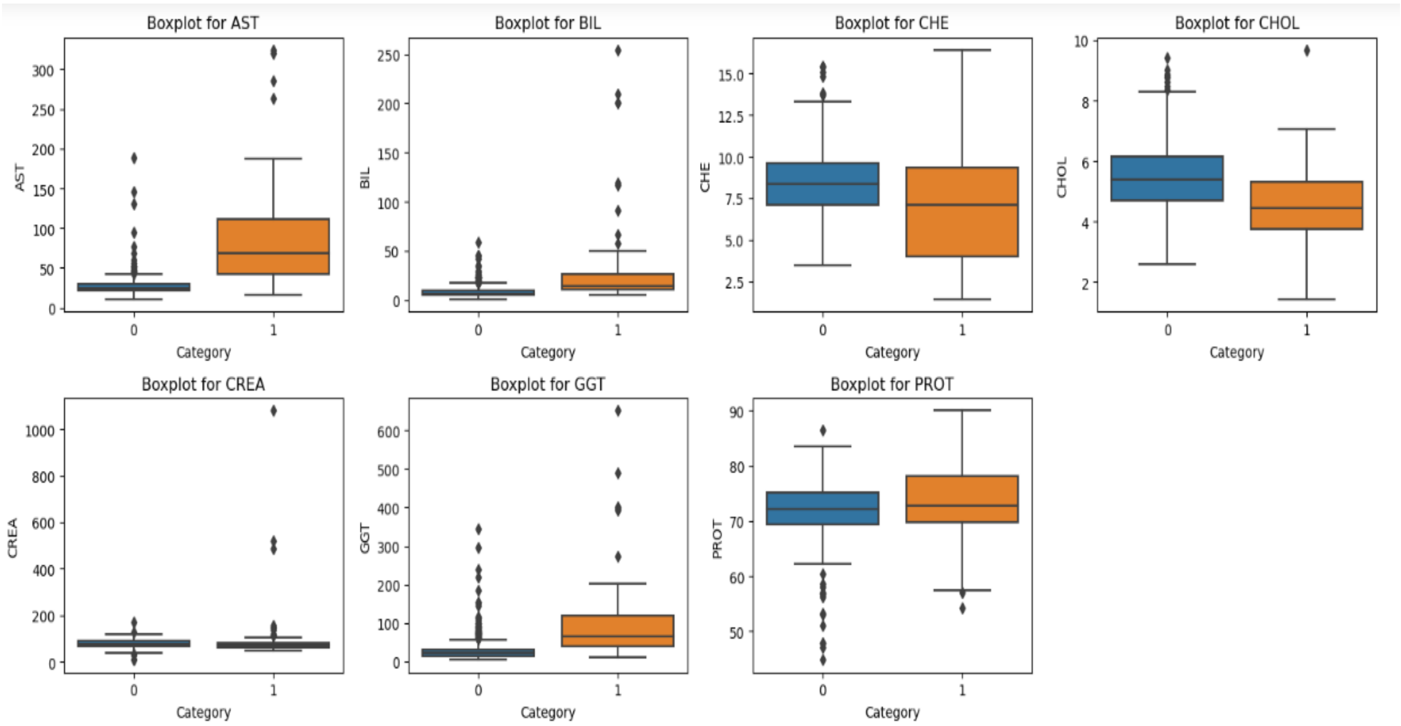

Figure 3b: Boxplot for Numeric Features

3.4 Feature Selection

The decision on the choice of features that were used in this analysis were done based on the following: Pearson correlation coefficient, the Recursive Feature Elimination, Selection of Best K and the Mutual Information Gained.

As suggested by [12], if r is the Pearson correlation coefficient, then if $|r| < 0.4$, the correlation between both variables is referred to as weak correlation, if $0.4 \leq |r| < 0.7$, then the correlation is moderate, while the correlation is strong if $0.7 \leq |r| < 1$.

Table 4: Correlation coefficients between features and label

Variables	Age	Sex	ALB	ALP	ALT	**AST**
Category	0.0378	0.07166	-0.2029	-0.2469	0.08375	**0.6217**
Variables	BIL	CHE	CHOL	CREA	**GGT**	PROT
Category	0.3985	-0.2308	-0.2634	0.1368	**0.4377**	0.2261

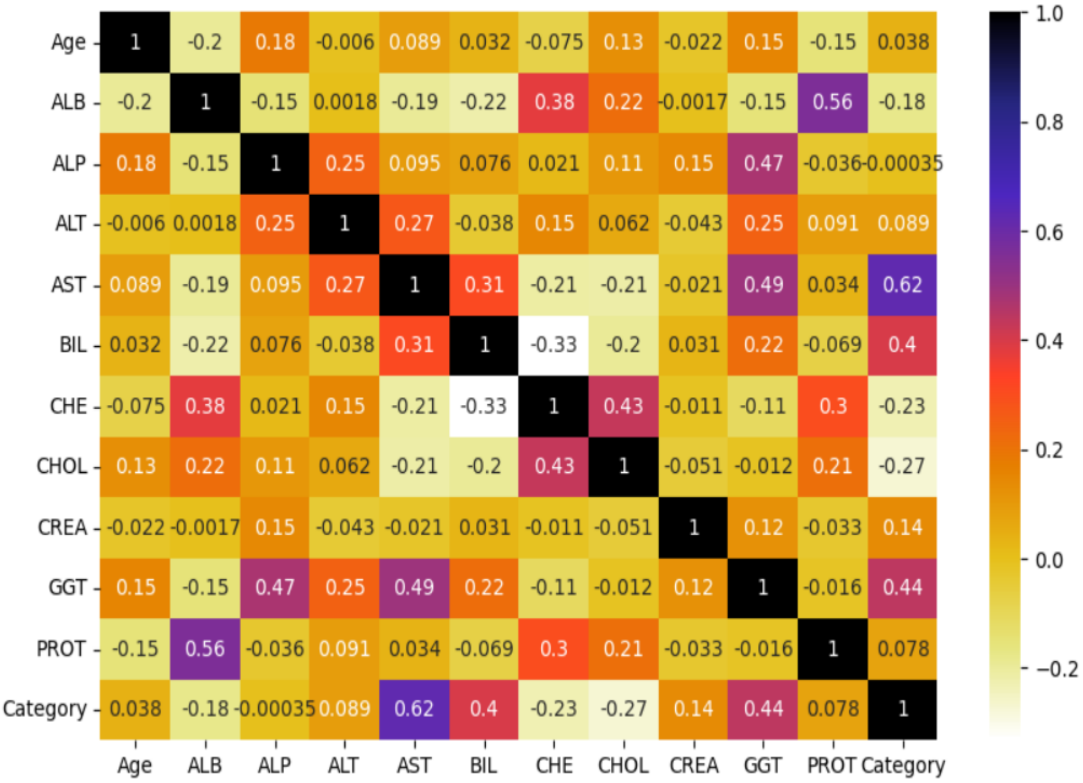

Figure 4: Scatter Plot

Thus, from figure 4 on heatmap of correlation matrix and table 4, “AST” and “BIL” have moderate relationship with the label, “BIL” has an approximately moderate correlation with the label, while the remaining features are very weakly correlated with the label.

In the Recursive Feature Elimination method, the Random Forest with cross validation of 10 was used. This method yielded “ALP”, “ALT”, “AST”, “BIL”, “CHE”, “GGT” and “PROT” as the top seven features that are linked with the label.

In the selecting K best method, the best 7 features that was suggested for use are “ALB”, “AST”, “BIL”, “CHE”, “CHOL”, “CREA”, and “GGT”.

Finally, in the Mutual Information Gained, the higher the value of a feature, the more the dependency of the feature on the label. In figure 5, the features are displayed in order of importance.

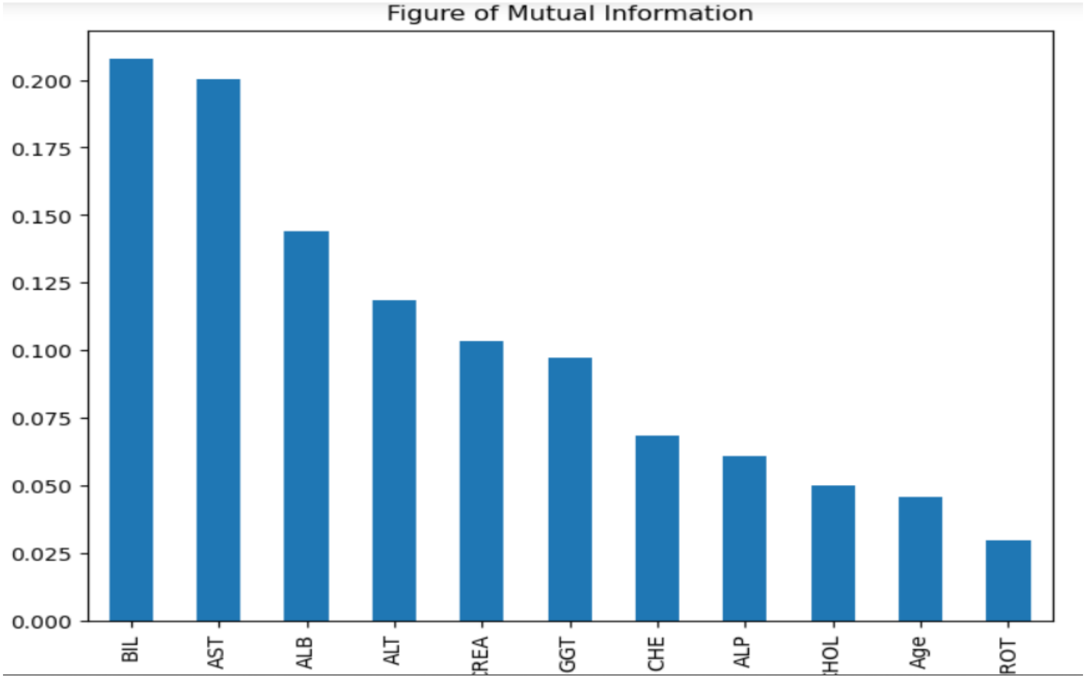

Figure 5: Chart for Mutual Information

Hence, table 5 shows the order of importance of features by the various methods and the chosen features.

Table 5: Summary of Feature Selected

Pearson Correlation Coefficient	**Recursive Feature Elimination**	**Select K Best**	**Mutual Information Gained**	**Chosen Features**
AST	ALP	ALB	BIL	AST
GGT	ALT	AST	AST	BIL
BIL	AST	BIL	ALB	GGT
-	BIL	CHE	ALT	CHE
-	CHE	CHOL	CREA	ALB
-	GGT	CREA	GGT	ALT
-	PROT	GGT	CHE	-
-	-	-	ALP	-
-	-	-	CHOL	-
-	-	-	PROT	-
-	-	-	AGE	-
-	-	-	SEX	-

3.5 Normalization and Scaling

To prevent the weights of the features whose magnitude are large from distorting our training algorithms thereby creating a bias in the system, the normalization function and scaling function in Python and R respectively were used. This reduced the weights of the numerical features to [0, 1] and [-1, -1] respectively.

3.6 Handling of Imbalanced Training Dataset

The normalized dataset was split into training and testing dataset, with 80% used for training while the remaining 20% for testing and model evaluation.

The table below shows the counts in each class in the training set.

Table 6: Counts of each class in the Training Set

	Classes		
Counts	0	1	Total
	438	54	492

The difference between the two classes is 384. With the binary class imbalanced, overfitting and data leakage are possibilities as machine learning algorithms exhibits extreme bias towards the majority classes, thereby treating all classes as majority class and have very low classification rates for minority classes. In dealing with this, the Synthetic Minority Oversampling Technique was used in which synthetic samples were generated for the minority class, this results in equality of both classes.

Also, the upSample was also introduced as this randomly selects samples from the majority class and equates this with the minority class.

The performance of these various means of handling imbalanced classes will be evaluated to determine the most optimum means.

We briefly explain the supervised machine learning algorithms used and their measures of their performance.

3.7 Supervised Machine Learning

3.7.1 Logistic Regression

[13] suggested that the conditions for the use and features of a logistic regression are:
(1) The label must contain two distinct classes.
(2) Features and log-odds of successful outcomes have a linear connection.
(3) The features should be independent of one another.
(4) There is low to no high collinearity between the features.
The logistic regression hypothesis is given as:

$$f_\theta(x) = h(\theta^T x) \qquad [1]$$

Where h is a sigmoid function defined by:

$$h(z) = \frac{1}{1+ e^{-z}}, \qquad [2]$$

and it assumes values between 0 and 1.

3.7.2 Naïve Bayes

This classifier is based on the Bayes Theorem. It assumes that the features are conditionally independent on each other given the label class.
It is a probabilistic class that learns from the training dataset and carries out its prediction on the test data based on the highest probability.

3.7.3 K Nearest Neighbor

This classifier is based on the feature space's k nearest neighbours' class labels. This is used when there are complicated, non-linear relationships between the features and the label. Because it is non-parametric by nature, feature scaling or normalization is very necessary to ensure that no feature predominates.
This algorithm executes classification in three steps which are:

(1) initial stages it computes the K-value.
(2) for each test sample, it computes the distance between all the training data and sorts it;
(3) it provides the class name to the test sample data by using the majority voting approach.

The Euclidean distance used is computed using:

$$E_d = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2} \qquad [3]$$

Where a_i and b_i are the distance between two points.

3.7.4 Random Forest (RF)

The foundation of Random Forest is the ensemble of decision trees, namely the classification and regression trees. Each tree predictor in this ensemble is created using a random subset of characteristics to divide each node, and a random set of vectors sampled individually. The forest will have a smaller generalization error, the lesser the correlation between various growing trees. This generalization error is influenced by the strength of each individual tree; powerful trees have low error rates. Internally, it also calculates the significance of the features for the classification and regression tasks. Here, each tree casts a vote for a class, and the class with the most votes win classification for the new example. It is important to note that Random Forest excels in midst of large or unbalanced datasets. It is also resilient to over-fitting and has strong generalization performance that surpasses boosting methods [14].

3.7.5 eXtreme Gradient Boosting (XGBoost)

The gradient boosting algorithm was created to have extremely high predictive power. However, because the method had to generate one decision tree at a time to minimize the mistakes of all prior trees in the model, its acceptance was still quite restricted. Thus, even for such modest models, training required a significant investment of time. To mitigate against this, the eXtreme Gradient Boosting (XGBoost) was introduced. This method reduces search times, organizes data, and uses many cores to generate separate trees. In addition, the application of sparse awareness entails an automated handling of missing data values, the creation of block structures to facilitate the parallelization of tree building, and ongoing training to improve pre-fit models on new data [15].

3.8 Measures of Performance

The type of algorithm and expected outcome determines the type of performance metric to be used. Under supervised learning modelling, if the outcome is prediction, the expected measures are R^2, Root Mean Square Error (RMSE), F-ratio among others. On the other hand, if classification, the measure of performance includes confusion matrix, accuracy, recall also known as sensitivity, precision, F-measure also referred to as F1 score, and specificity.

Confusion Matrix

Each prediction made by any model can produce one of the following results: True Positive, True Negative, False Positive, or False Negative. This is tabulated in the confusion matrix in table 6.

Table 7: Confusion Matrix

	Prediction		
Actual		Positive	Negative
	Positive	True Positive	False Negative
	Negative	False Positive	True Negative

True Positive (TP): This occurs when a model predicts positive, and the label is truly positive. That is, correctly predicted Hepatitis C positive data.

True Negative (TN): This occurs when a model predicts negative, and the label is truly negative. That is, correctly predicted Hepatitis C negative data.

False Positive (FP): This occurs when a model predicts positive, and the label is truly negative. That is, wrongly predicted Hepatitis C positive data.

False Negative (FN): This occurs when a model predicts negative, and the label is truly positive. That is, wrongly predicted Hepatitis C negative data.

Accuracy

This is the proportion of correct predictions made by our model out of all the predictions.
This can be computed using the formula:

$$Accuracy = \frac{TP+TN}{TP + FP+FN+TN} \quad [4]$$

Precision

This is the proportion of positive predictions which were truly correct. This can be determined by the formula:

$$Precision = \frac{TP}{TP + FP} \quad [5]$$

Specificity

This is the proportion of true negatives that were accurately predicted. This is computed using:

$$Specificity = \frac{TN}{TN + FP} \quad [6]$$

Recall

Recall, also known as sensitivity, is the proportion of true positives that were truly predicted.

$$Recall = \frac{TP}{FN + TP} \quad [7]$$

[16] explained that accuracy works well if the classes considered are balance. However, in the presence of imbalanced classes in datasets and the objective of the model being classification, the appropriate metric for comparing various machine learning models is the area under the receiver operating characteristic (AUROC) curve alongside with accuracy.

4. DISCUSSIONS OF RESULTS

The results obtained will be discussed from different perspectives. First, we compare the results obtained by the two means used in handling the imbalanced to determine that optimum and we compare the results obtained by Python and R, using the performance metrics of accuracy. specificity and the area under receiver operating characteristic curve.

The result produced by Python and R software indicates that for the confusion matrix, the positive class is 0 for those who blood donors while the negative class is 1 for those who have hepatitis c virus.

Table 8: Summary of Performance Metrics for R

Algorithms	Scenarios on dataset	Accuracy	Specificity	Precision	AUC Value
Logistic Regression	SMOTE upSample	0.9268 0.9350	0.9333 0.8000	0.9901 0.9717	 0.88
Random Forest	**SMOTE** upSample	**0.9756** 0.9350	0.9333 0.9333	**0.9907** 0.9902	**0.96**
Naïve Bayes	SMOTE upSample	0.9187 0.9268	0.8000 0.8000	0.9712 0.9714	 0.87
K Nearest Neighbor	SMOTE upSample	0.9756 0.9431	0.8667 0.8000	0.9810 0.9720	0.91
XGBoost	SMOTE upSample	0.9675 0.9675	0.9333 0.9333	0.9907 0.9907	0.95

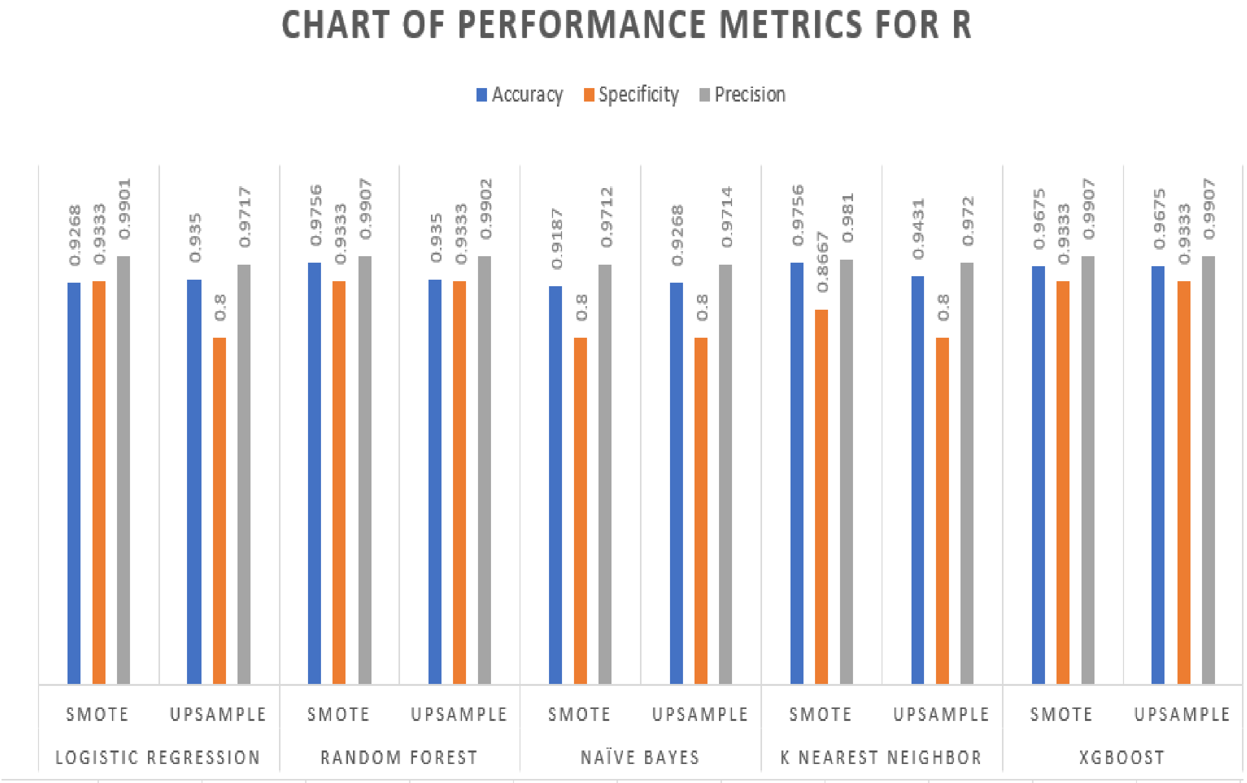

Figure 6: Chart of Performance Metrics for R

From the chart in figure 6 and table 8, it can be noted that:

(1) Accuracy: The accuracy score obtained by RF and KNN using SMOTE were higher than their counterpart using upSample with a marginal difference of 0.0406 and 0.0324, respectively. The reverse is the case for LR and NB where the accuracies for upSample was slightly higher than their SMOTE counterpart. However, for XGBoost, the accuracy value remained the same.

(2) Specificity: In the ability to correctly predicted the negative cases of those who were hepatitis c patient, the SMOTE performed better than upSample in LR and KNN, while it remained the same for RF, NB and XGBoost.

(3) Precision: In the ability to correctly predict the positive cases of those who are blood donors, the value obtained by the SMOTE cases of LR, RF an KNN were better than those obtained by upSample. The upSample had a better performance over SMOTE in NB while it remained unchanged in XGBoost.

Thus, using the R software, the result produced by SMOTE in which synthetic samples were generated for the minority class performed better than upSample, where the samples in the majority class was reduced.

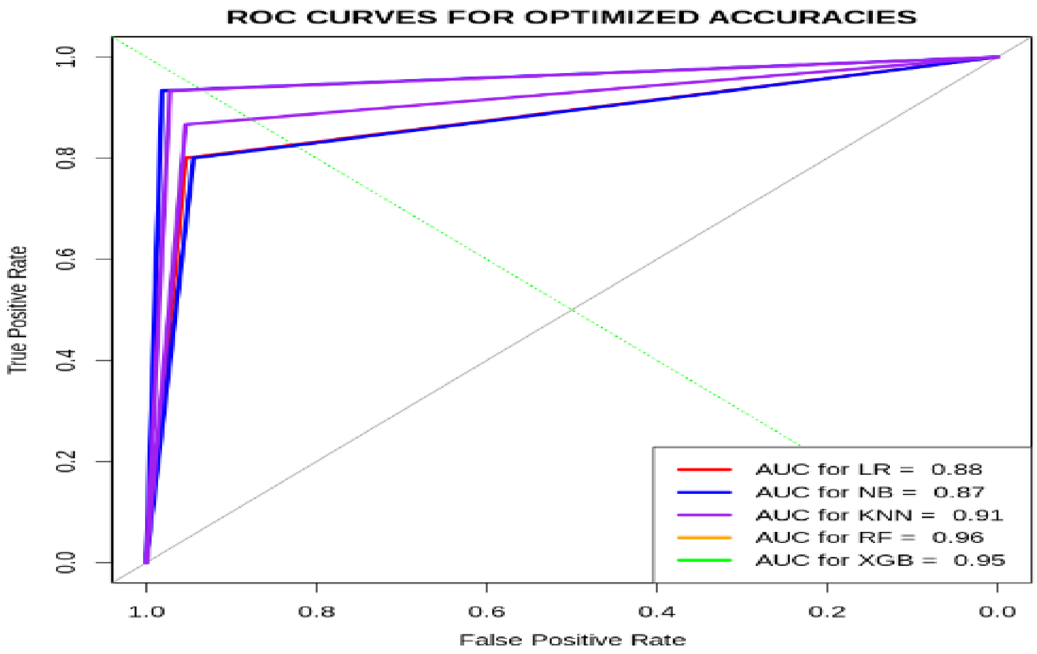

Figure 7: ROC Curve for Accuracies in R

Overall, considering the ROC curve in figure 7, Random Forest (SMOTE) performed better with an AUC value of 0.96, an accuracy value of 0.9756, although this accuracy value is also shared with KNN (SMOTE).

Table 9: Summary of Performance Metrics for Python

Algorithms	Scenarios on dataset	Accuracy	Specificity	Precision	AUC Value
Logistic Regression	SMOTE	0.94	0.85	0.94	0.93
	upSample	0.94	0.77	0.94	
Random Forest	**SMOTE**	**0.96**	0.92	**0.96**	**0.91**
	upSample	0.92	0.92	0.92	
Naïve Bayes	SMOTE	0.92	0.77	0.91	
	upSample	0.92	0.77	0.91	0.85
K Nearest Neighbor	SMOTE	0.94	0.92	0.94	0.93
	upSample	0.94	0.77	0.94	
XGBoost	SMOTE	0.95	0.92	0.95	0.94
	upSample	0.93	0.92	0.93	

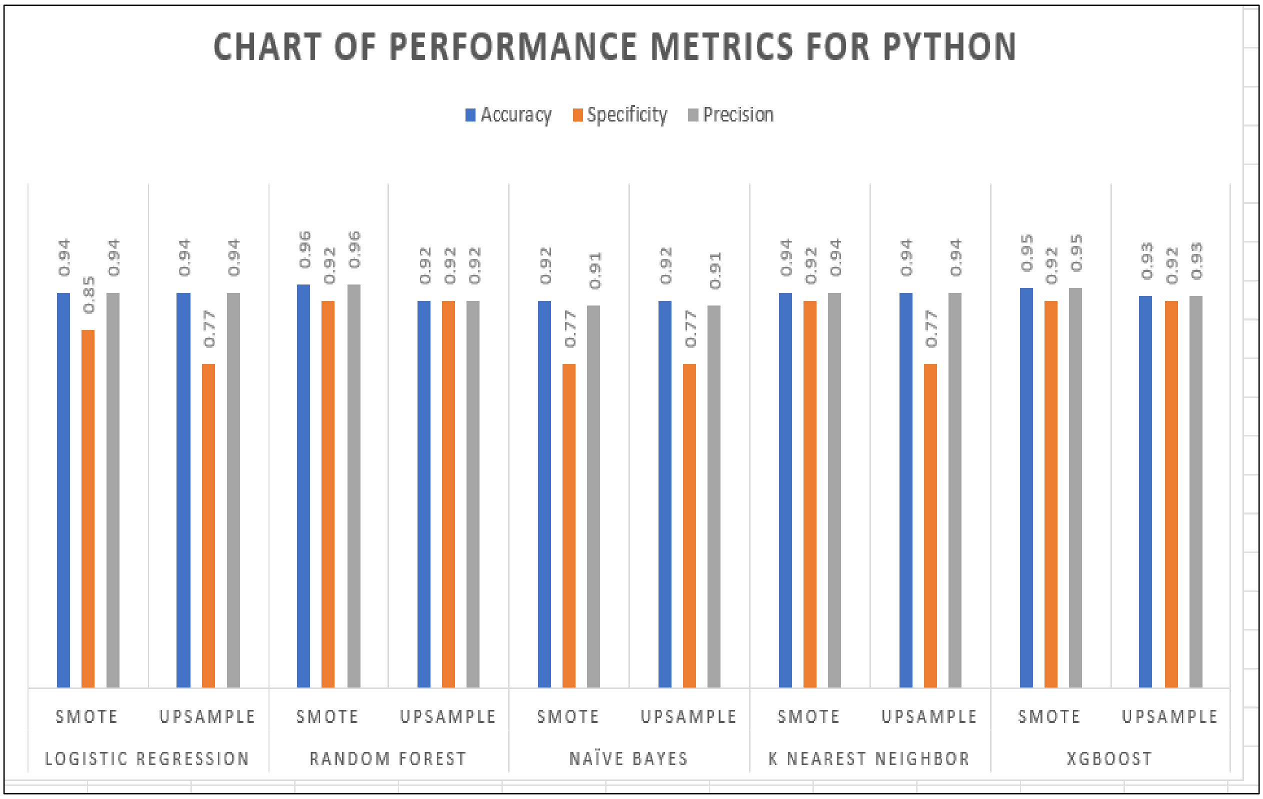

Figure 8: Chart of Performance Metrics for Python

From the chart in figure 8 and table 9, it can be noted that:

(1) Accuracy: In ascertaining the overall correctness of prediction made by the algorithms considered, the Random Forest (SMOTE) and XGBoost (SMOTE) had values of 94% and 95% respectively which are marginally higher than the results obtained using upSample. But for other algorithms, their respective values are the same.

(2) Specificity: In correctly predicting the negative cases of patients with hepatitis c, the LR (SMOTE) and KNN (SMOTE) had higher values of 0.85 and 0.92, respectively than those obtained using upSample, whose values are 0.77 and 0.77 respectively. However, for other algorithm, no difference was noticeable.

(3) Precision: The RF (SMOTE) and XGBoost (SMOTE) have higher predicting power in the classification of those who are blood donors with a value of 0.96 and 0.95 respectively,

which is marginally higher than the result obtained using upSample which are 0.92 and 0.93, respectively. However, for other algorithms, both methods had similar results.

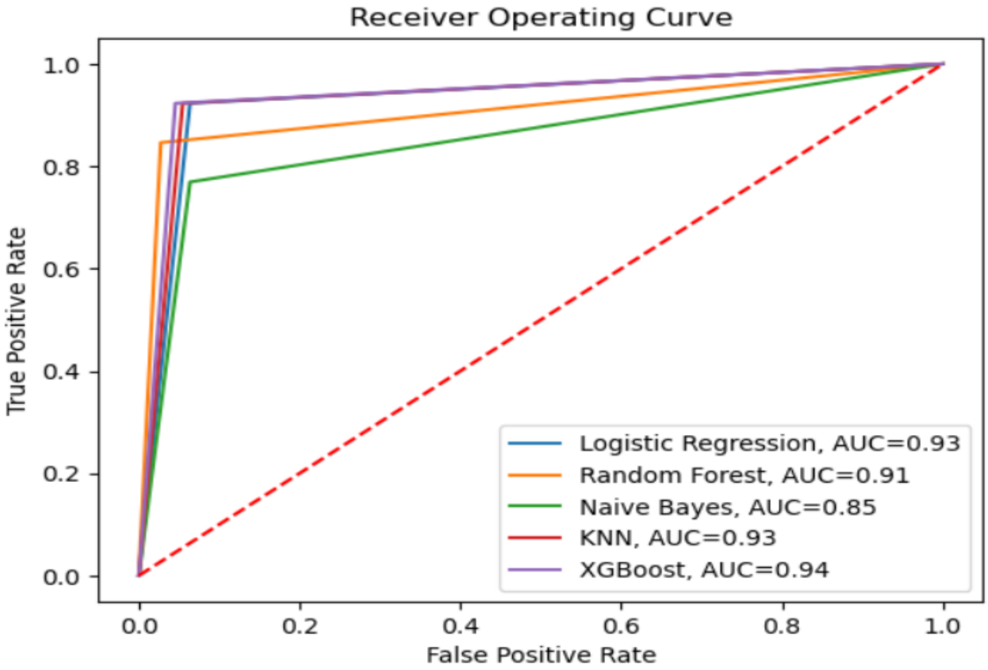

Figure 9: ROC Curve for Accuracies in Python

For Python, considering the ROC curve in figure 9, Random Forest (SMOTE) performed better with an AUC value of 0.95, an accuracy value of 0.96, and a precision of 0.91, which are higher than any other values obtained for the algorithms studied.

CONCLUSION

In conclusion, using Pearson correlation coefficient, the Recursive feature Elimination, Selecting Best K and the Mutual Information Gained, the set of features which played significant role in the prediction of Hepatitis C virus from the dataset are “AST”, “BIL”, “GGT”, “CHE”, “ALB” and “ALT”.

From the result obtained, when class imbalance is present, this should be handled using the Synthetic Minority Oversampling Technique (SMOTE) where synthetic samples were generated for the minority class. This is supported by the result obtained for both R and Python, where, for instance, the RF for R software yielded an accuracy of 0.9756, precision of 0.9907 and auc of 0.96, which are the best result obtained. Also, for Python, the RF for SMOTE performed better than other algorithms with an accuracy, precision and auc value of 0.96 each and 0.91 respectively.

Finally, comparing the result obtained by R and Python, we note that result of RF using R had a better accuracy value of 0.9756, precision value of 0.9907. Although, both had the same auc value of 0.96. This result agrees with the work of [6], in which R tool performed better than Python, who used a different dataset.

Comparing our result with those of other researchers who used the same dataset. The results are summarized in the table below.

Table 10: Comparison of Results with other Researchers

Author	Algorithms	Accuracy	Precision	AUC-Score
[17]	Rule-Based Decision Trees	75.3%	-	-
[18]	Random Forest and Logistic Regression	96.19%	-	-
[10]	Random Forest	97.29%	-	0.998
[19]	AST/ALT ratio	95.4%	-	-
[20]	Fusion Model	95.45%	-	-
Vincent and Ulgen	**Random Forest (SMOTE)**	**97.56%**	**0.9907**	**0.96**

From table 10, our side-by-side comparison showed that the Random Forest yielded the highest accuracy score of 97.56%. This is closely followed by the work of [10], with an accuracy of 97.29%, who also used SMOTE in handling imbalanced class.

In addition to the above, apart from [10], other researchers failed to report on the auc-score. Using this, the Random Forest had a score of 0.96, which is slightly lower than that obtained by [10], who obtained a score of 0.998.

Finally, the precision value of 0.9907 shows that the correct proportion of those who are blood donors.

5. LIMITATIONS AND FUTURE WORK

Our study is limited to five machine learning algorithms. In the future, it is important to consider the performance of few deep learning algorithms for better results. Also, combination of ensemble methods will be helpful will a more efficient and accurate classification and prediction as these methods will possess the advantages of each of the models.

Another limitation is the unavailability of larger dataset in which gene is a factor. It has been reported that hepatitis C virus is geographical based and as such, the inclusion of such feature will go a long way in the treatment of the disease.

REFERENCE

[1] World Health Organization (WHO), “Hepatitis C,” 2023, https://www.who.int/news-room/fact-sheets/detail/hepatitis-c. View at: Google Scholar

[2] DALGARD, O., JEANSSON S., SKAUG, K., RAKNERUD, N., and BELL, H., 2003. Hepatitis C in the General Adult Population of Oslo: Prevalence and Clinical Spectrum. Taylor and Francis, p 11

[3] HARRIS, R. J., MARTIN, N. K., RAND, E., MANDAL, S., MUTIMER, D., VICKERMAN, P., RAMSAY, M. E., ANGELIS, D., HICKMAN, E. and HARRIS, H. E., 2016. New Treatments for Hepatitis C Virus (HCV): Scope for Preventing Liver Disease and HCV Transmission in England. Journal of Viral Hepatitis. 23(8), 631 – 643. https://doi.org/10.1111/jvh.12529

[4] MUHAMMAD, L.J., et al., 2021. Supervised Machine Learning Models for Prediction of COVID-19 Infection using Epidemiology Dataset. SN Computer Science, 2 (1), 11.

[5] DENNISTON, M.M., et al., 2012. Awareness of infection, knowledge of hepatitis C, and medical follow-up among individuals testing positive for hepatitis C: National Health and Nutrition Examination Survey 2001-2008. Hepatology (Baltimore, Md.), 55 (6), 1652-1661.

[6] SATISH CR NANDIPATI, CHEW XINYING and KHAW KHAI WAH, 2020. Hepatitis C Virus (HCV) Prediction by Machine Learning Techniques. Applications of Modelling and Simulation, 4, 89-100

[7] Reddy N. Satish Chandra, S. N. Song, Z. M. Lim and C. Xin Ying, Classification, and feature selection approaches by machine learning techniques: Heart disease prediction, International Journal of Innovative Computing, 9(1), 2019, 39-46

[8] K. AHAMMED, MD. SHAHRIARE S., MD. I. KHAN and MD. WHAIDUZZAMAN, 2020. 2020 IEEE Region 10 Symposium (TENSYMP), 5-7 June 2020, Dhaka, Bangladesh

[9] Reza Safdari, Amir Deghatipour, Marsa Gholamzadeh, and Keivan Maghooli, 2022. Applying Data Mining Techniques to classify Patients with suspected hepatitis C virus infection. Intelligent Medicine 2 (2022), 193 – 198.

[10] Source of data: UCI Machine Learning Repository: HCV data Data Set

[11] Heymans MW, Twisk JWR. Handling missing data in clinical research. J Clin Epidemiol. 2022 Nov; 151:185-188. doi: 10.1016/j.jclinepi.2022.08.016. Epub 2022 Sep 21. PMID: 36150546.

[12] Bruce Ratner,2009. The correlation coefficient: Its values range between +1/−1, or do they? Journal of Targeting, Measurement and Analysis for Marketing, (17), pp 139-142.

[13] HASAN, S., et al., 2018. Comparison of Machine Learning Approaches for Prediction of Advanced Liver Fibrosis in Chronic Hepatitis C Patients. IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB), 15 (3), 861-868.

[14] Breiman, L., 2001. Random Forests. *Machine Learning*. 45 10.1023/A:1010933404324

[15] R. SANTHANAM, N. UZIR, S. RAMAN and S. BANERJEE, 2017. Experimenting XGBoost Algorithm for Prediction and Classification of Different Datasets. International Journal of Control Theory and Applications ISSN: 0974–5572 © International Science Press Volume 9 • Number 40 • 2016

[16] Tom Fawcett, An introduction to ROC analysis, Pattern Recognition Letters, Volume 27, Issue 8,2006, Pages 861-874, ISSN 0167-8655, https://doi.org/10.1016/j.patrec.2005.10.010.

[17] G. Hoffmann, A. Bietenbeck, R. Lichtinghagen, and F. Klawonn, ``Using machine learning techniques to generate laboratory diagnostic pathways – a case study,'' *J. Lab. Precis. Med.*, vol. 3, p. 58, Jun. 2018.

[18] T.-H.-S. Li, H.-J. Chiu, and P.-H. Kuo, ``Hepatitis c virus detection model by using random forest, logistic-regression, and ABC algorithm,'' *IEEE Access*, vol. 10, pp. 91045_91058, 2022.

[19] 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.

[20] S. S. Chawathe, ``Diagnostic classification using hepatitis c tests,'' in *Proc. IEEE Int. IoT, Electron. Mechatronics Conf. (IEMTRONICS)*, Sep. 2020, pp. 1_7