International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 20s, 2025
https://theaspd.com/index.php

A Hybrid Multilayer Stack Ensemble Model for Early Prediction of Liver Disease

A. I. Anerao[1] , Dr. O. S. Jadhav[2]
[1]Research Scholar Department of Statistics Dr. BAMU, Chhatrapati Sambhajinagar,431004, (M.S) India. statistics.ain@bamu.ac.in
[2]Associate Professor Department of Statistics Dr. BAMU, Chhatrapati Sambhajinagar,431004, (M.S) India. osj.statistics@bamu.ac.in

Abstract: *One of the leading causes of death worldwide is liver disease. The number of people experiencing suffering is increasing consistently. Maintaining a healthy liver is crucial for supporting essential processes, including digestion and detoxification. Some of the most common liver issues that need medical treatment include fatty liver, cirrhosis, and hepatitis. Due to the mild symptoms, it is challenging to anticipate in the early phases. To tackle the problem, various heterogeneous data mining algorithms are used to analyse performance and identify the most appropriate model for diagnosing liver disease. In our research, we utilised a multi-layered stacked ensemble learning model to enhance the precision of liver disease prediction. Five different classification models, including Support Vector Machine, Decision Tree, XG Boost, Cat Boost, and Logistic Regression models, were employed in the base layer of our model. The meta-layer of the model consists of K-Nearest Neighbour, Logistic Regression, Support Vector Machine classification models, and we classified the observations based on the Voting Classifier that we deploy on the meta-layer model. Consequently, we discovered that this suggested framework achieves a 92.35% accuracy rate along with improved F1 Score, recall, and precision.*

Keywords: *Liver Disease, Classification Model, Ensemble Learning, Stacking, Hybrid Model*

1. INTRODUCTION

Liver disease represents a significant worldwide health concern, as India recorded 268,580 deaths from liver-related ailments in 2020, making up 3.17% of all fatalities. [1]. The liver, with a weight of around 1.36 kg, is the largest organ in the body and is essential for sustaining homeostasis. It handles essential functions like protein production, blood clotting, cholesterol and glucose processing, iron metabolism, and detoxifying substances. Although the liver is sturdy, it is susceptible to multiple stressors such as inconsistent eating habits, alcohol intake, contact with harmful vapours, and drug use, all of which together lead to a slow deterioration in its functionality. [2] [3]. Timely identification of liver diseases is essential to avert serious complications like organ failure or long-term conditions. Nonetheless, conventional diagnostic techniques are frequently invasive, expensive, and restricted in their capacity to detect the illness at an initial stage. Patients might show signs such as dark urine, pale stools, itching, and fluid buildup solely in the later stages, highlighting the importance of non-invasive and predictive diagnostic methods. [4].

Machine Learning (ML) presents itself as a forceful method to boost disease evaluation and forecasting capabilities for human hepatic conditions and other medical conditions. Machine Learning methods simplify the systematic evaluation of massive and varied clinical information databases that consist of patient background details as well as genealogical health records alongside medical records, electronic data, and outcomes from laboratory examinations and diagnostic images. Through these algorithms, researchers can recognise intricate patterns as well as associations that relate to liver disease. The acquired insights are then used to develop strong predictive models that aid both early detection and risk assessment. Several academic studies analyse the use of ML methods to forecast liver disease risks by processing clinical data from various sources. [5].

Ensemble learning is among the most effective ML methods for medical diagnosis. Ensemble classifiers embody the concept of combining decisions from different models to enhance performance and accuracy metrics. The ensemble method gives more precise results than regular models. The process for improving

performance with ensembles usually involves reducing the variance component of the prediction errors produced by the causal model. [6]. In this research, we employ a stacked ensemble classification model to forecast liver disease. We employ Gradient Boosting (GB), Extra Tree models (ET), XG Boost (XGB), Random Forests (RF), and Cat Boost (CB) as base models while using Support Vector Machine (SVM), K- Nearest Neighbour (KNN), and Logistic Regression (LR) as meta models. Following the application of the meta-model, we employ the Voting Classifier and classify new observations accordingly. After applying the suggested model, we discovered that it performs effectively with the data, achieving an accuracy of 92,35%.

Here, we review recent studies on liver production using data mining. Abdul Quadir Md et.al [7] Applied six ensemble learning algorithms to the Indian liver patient dataset (ILPD) utilising various preprocessing techniques, discovering that the suggested model employing extra trees and random Forest surpassed other approaches, achieving the highest accuracies of 91.82% and 86.06% respectively. Shahid Mohammad Ganie et. al. [5] Employed a three-ensemble method utilising nine classification algorithms assessed on a data set and discovered that the gradient boosting algorithm delivered the highest performance, achieving 98.80% accuracy and 98.50% for precision, recall, and F1 score individually. Abdullah Al Ahad et.al. [8] Used data preprocessing tools to increase the performance of machine learning models. Furthermore, they utilised an ensemble approach by merging different machine learning classifiers to increase the accuracy of the prediction for liver disease. Consequently, the ensemble model attained training and testing accuracies of 99.87% and 99.80%, respectively. Deepika Bhupathi et.al [9] forecasted liver disease employing five machine learning algorithms and evaluated their effectiveness. Consequently, they discovered that the K-K-nearest neighbour (KNN) model reached the highest accuracy of 91.7% among traditional algorithms, whereas the autoencoder network surpassed all techniques with an accuracy of 92.1%.

Rohini A. Bhusnurmath et.al [10] used a stacked ensemble model on the ILPD; they employed an RF as a meta learner, ANN as a base learner, achieving a classification accuracy of 98.23%, precision of 97.43%, and a recall rate of 100%. Tsehay Admassu Assegie1 et. al [11] used an RF model applied for recursive feature elimination during the preprocessing phase, and the SVM is trained using the optimum set of variables. The experimental findings indicate that the suggested support SVM model attained an accuracy of 78.3%. Mounita Ghosh et.al [12] compare various machine learning algorithms, including LR, RF, XG Boost, SVM, AdaBoost, KNN, and DT, which were evaluated for their effectiveness in predicting liver disease. The results show that the RF algorithm outperformed the others, achieving an accuracy of 83.70%. Ruhul Amin et.al. [13] developed an integrated methodology to extract features with the goal of liver disease assessment. The researchers conducted dimensionality reduction through PCA and also used FA along with LDA to process the ILPD dataset. The performance evaluation of various machine learning models, LR, RF, KNN, SVM, MLP and ensemble methods, occurred through 10-fold cross-validation of transformed features. Random Forest yielded the highest performance among models with 88.1% accuracy, 88.68% F1 Score, 85.33% precision and 92.3% recall. Kuzhippallil et al. [14] conducted research on different classification algorithms and feature selection strategies within machine learning for predicting liver disease. The authors used a genetic algorithm together with the XG Boost (XGB) model for selecting important features. The research examined multiple techniques, which included LR, KNN, DT, RF, Gradient Boosting (GB) and AdaBoost, XG Boost and Light GBM (LGBM) and stacking ensemble. Both LGBM and stacking models reached 86% accuracy when feature selection and outlier removal were applied during the prediction of liver disease. Fahad Mostafa et. al. [15] utilised feature significance ranking derived from the Gini index to validate the key predictors identified via Principal Component Analysis (PCA) and to evaluate the effectiveness of various binary classification algorithms. Their findings indicated that the RF model demonstrated the highest classification performance, achieving an accuracy of 98.14%.

2. METHODOLOGY

2.1 ***About Datasets*** In our study, we utilise datasets sourced from Kaggle.com. [16].The data comprises 11 attributes, with a total of 1700 instances available across all instances. 936 samples are affected by liver disease, while 764 samples are unaffected by it. The comprehensive explanation is provided below:

Table 1. Description Dataset

Sr. No.	Feature Name	Feature Type	Feature Description
1	Age	Numeric	Age of personnel, ranging between 20 and 80 years.
2	Gender	Categorical	Gender of the personnel: 0 represents Male, 1 represents Female.
3	BMI	Numeric	Body Mass Index, with values between 15 and 40.
4	Alcohol Consumption	Numeric	Weekly alcohol intake, measured in units from 0 to 20.
5	Smoking	Categorical	Smoking status: 0 for non-smokers, 1 for smokers.
6	Genetic Risk	Categorical	Hereditary risk factor: 0 = Low, 1 = Medium, 2 = High.
7	Physical Activity	Numeric	Time spent on physical activity per week, ranging from 0 to 10 hours.
8	Diabetes	Categorical	Presence of diabetes: 0 indicates No, 1 indicates Yes.
9	Hypertension	Categorical	Presence of hypertension: 0 for No, 1 for Yes.
10	Liver Function Test	Numeric	Liver function test value, ranging between 20 and 100.
11	Diagnosis	Categorical	Outcome variable: 0 = No liver disease, 1 = Liver disease present.

2.2 Data Preprocessing

2.2.1Data Normalization

Normalisation is a process where the features are transformed by "scaling down." In a characteristic, there is frequently a substantial discrepancy between the highest and lowest values, for example, 0.01 and 1000. Normalisation is done to reduce the magnitudes of values to significantly lower levels [17]. The most common methods of normalisation are given as:

Z -Z-Score Normalisation

$$x^* = \frac{x_i - mean}{Standard\ deviation}$$

Here, x^* It is a new feature and x It is an old feature.

2.2.2 Outlier Detection

Outliers are points that are very far from the rest data. It may be significantly greater or less in value when compared to the other data points in a dataset. There are numerous methods available for detecting outliers; however, we specifically utilise the boxplot method for outlier detection. Finding the interquartile range (IQR) is a first step in the box plot method. The difference between the Third quartile (Q3) and the first quartile (Q1) is known as IQR, and it is used to quantify statistical dispersion. To identify the outliers, we need to find the upper and lower boundaries. If any data point falls below of lower boundary or above of upper boundary, we consider that point as an outlier and remove it from the data. [18].

$$IQR = Q_3 - Q_1$$

To identify outliers, our decision boundary is:

$$\text{Lower limit (LB)} = Q_1 - 1.5 * IQR$$
$$\text{Upper limit (UB)} = Q_3 + 1.5 * IQR$$

2.2.3 K–Fold Cross Validation

The method of K-fold cross-validation involves splitting the dataset into k groups. The validation set is the remaining portion after the model has been trained on (k - 1) segments in each iteration. The number of times the process is repeated depends on the number of folds. Five-fold cross-validation was applied to the dataset in our investigation. [8].

2.3 Used a Classification model

2.3.1 Logistic Regression

Logistic Regression (LR) [19] Involves a dichotomous dependent variable, with at least one independent variable evaluated through probability calculations utilising the sigmoid function. It is also referred to as a logit model. The logistic regression model utilised is:

$$\pi(x) = \frac{e^{\beta_0+\beta_1 x}}{1+e^{\beta_0+\beta_1 x}}$$

A logit transformation is recognised as a transformation of π(x). This transformation is specified for π(x):

$$L(x) = \log\left[\frac{\pi(x)}{1-\pi(x)}\right] = \beta_0 + \beta_1 x$$

This $L(x)$ Called as log odds ratio or logit function. Here, β_0 & β_1 Are regression Coefficients.

2.3.2 Support Vector Machine

Support vector machine (SVM) [20] [21], represents the data set as points in an n-dimensional space, with n indicating the number of variables. The main target of SVM is to establish a hyperplane that separates datasets into various classes, ensuring the hyperplane maximises the gap between the different categories. To guarantee resilience, the hyperplane should be chosen to establish a wide margin and enhance the separation between the nearest data points of each category. SVM provides greater accuracy and can efficiently handle complex nonlinear data points while avoiding overfitting. According to the decision boundary, SVM is classified into two types: Linear SVM and Non-Linear SVM.

The equation of a hyperplane is:

$$w^T x + b = 0$$

Equation of linear SVM classifier:

$$\hat{y} = \{1 \ : \ w^T x + b \ \geq 0 \ 0 \ : w^T x + b < 0$$

Equation of Soft Margin SVM Classifier:

$$\frac{Minimize}{w,b}\frac{1}{2}\ w^T w + C\sum_{i=1}^{m} \quad \zeta_i$$

Subject to $y_i(w^T x + b\ ) \geq 1 - \ \zeta_i$ and $\zeta_i > 0, i = 1,2,3 - - - m.$

2.3.3 K-Nearest Neighbour

K-Nearest Neighbour (KNN) [22] is a classification technique that does not rely on parameters. In KNN, the output variable is divided into several categories. Before classifying a new observation, it is necessary to consider the value of k, which denotes the number of neighbours to account for. After selecting K, we calculate the Euclidean distance between the new observation and every other observation in the dataset. For every new data point, determine the K nearest neighbours based on the Euclidean distance after measuring it. Determine how many data points there are in each category among these K neighbours. The category with the most neighbours will receive new data points.

2.3.4 Random Forest

Random Forest (RF) [23] It is a significant alteration of bagging, which constructs a vast assortment of uncorrelated trees and subsequently averages them**.** Steps of the Random Forest model are given below:

For b = 1 to B

Step 1: Generate **Z** bootstrap samples, each of size **S**, from the original training dataset.

Step 2: For each bootstrap sample, build a decision tree C_b using the following recursive procedure at each terminal node until a stopping criterion is met:

I. Randomly select m features from the total p available features.
II. Identify the optimal splitting feature from the m selected features based on a chosen criterion.
III. Partition the current node into two child nodes based on the selected feature and its best split point

The decision on splitting nodes is determined by Gini impurity or Entropy, until the stopping criteria are reached.

$$\text{Gini (a)} = 1 - \sum_{b=1}^{B} \quad a_i^2$$

Step 3: $\hat{c}_b(x)$ Be the class prediction of both random Forest trees.

$$C_{rf}^{B}(x) = majority\ votes\ \{\hat{c}_b(x)\}_{b=1}^{B}$$

2.3.5 Extreme Gradient Boosting (XG Boost)

XG Boost (XGB) [24] An extensible decision tree ensemble model that uses gradient boosting for its design. XG Boost minimises a loss function to construct an additive extension of the objective function.

Because XG Boost only uses decision trees as its classifiers, it controls the trees' complexity using a modified version of the loss function.

$$L_{xgb} = \sum_{i=1}^{N} L(y_i, F(x_i)) + \sum_{m=1}^{M} \Omega(h_m)$$

$$\Omega(h) = \gamma T + \frac{1}{2}\lambda \, ||\omega||$$

Where $L(y, F(x))$ It is a loss function; T is the number of leaves of the tree. ω Is output score of the leaves. This loss function can be incorporated into the split criterion of decision trees, resulting in a pre-pruning approach. Larger values of γ show to less complex trees. The value of γ indicates the minimum loss reduction gain needed to separate the internal node.

2.3.6 Gradient Boosting

Gradient boosting (GB) [24] Combines the capabilities of a poor learner with those of a good learner in an iterative manner to create a predictive model.

Let $f_0(x)$ Be the initial forecast for all observations, such that:

$$f_0(x) = argmin_\gamma \sum_{i=1}^{N} L(y_i, \gamma)$$

Where, $L(y_i, \gamma)$ It is a loss function.

The ultimate model after M iterations is the combination of the initial model and the adjustments from all the weak learners:

$$f_m(x) = f_0(x) + \sum_{i=1}^{M} \eta h_i(x)$$

Where, $h_i(x)$ Is the prediction of a weak learner in m steps.

2.3.7 Extremely Randomised Trees

Multiple decision trees, like Random Forest but with extra randomisation, are combined in the ensemble machine learning model, Extremely Randomised Trees (Extra Trees) [25]. Both employ numerous trees, but Extra Trees creates splits at random and doesn't look for ideal thresholds. For each tree, it usually uses the entire dataset rather than bootstrap sampling, but it keeps things random by choosing random split points. The majority vote of individual tree projections determines the Extra Tree model's final prediction.

$$\hat{y} = argmax_{y_k} \sum_{i=1}^{M} I(T_m(x) = y_k)$$

I It is an indicator function.

$T_m(x)$ Is the output of m^{th} Tree.

2.3.8 Cat Boost

Cat Boost and other gradient boosting techniques are designed to effectively handle category data while preventing overfitting. Although it uses decision trees as a foundational learner, its primary novelty is in the way it handles categorical information and enhances gradient, which increases accuracy and efficiency. Cat Boost minimises the following objective function:

$$L\big(y, f(x)\big) = \sum_{i=1}^{M} l\big(y_i, f(x_i)\big) + \Omega(h_m)$$

Where,

$l\big(y_i, f(x_i)\big)$ = Loss Function

$\Omega(h_m)$ = Regularisation term to prevent overfitting.

2.4 Proposed model:

The ensemble learning technique, stacking, uses a hierarchical structure to combine different base models, which improves their predictive capabilities. Base learning and meta-learning represent the two

main operational periods of this method. A group of individual models receives base training during this phase while using the original training dataset. After training the base models, the developed predictions are aggregated to generate a new dataset. The metacognitive learning phase runs on data obtained from the base models during training before using it to educate the meta-learner. The base learners' collective predictions must be aggregated by the meta-learner in order to reach the optimal performance accuracy closure. After completion of the training process, the meta-model performs predictions across the test set. Data preprocessing occurs first before the proposed model splits its datasets into training and testing components. The technique implements k-fold cross-validation to achieve robustness along with the reduction of model overfitting. The training data structure divides itself into two parts, with one segment used to develop base models and the second segment used for evaluating base model outputs that act as training data for the meta learner.

A new dataset is then created from the predictions of the base models, which serves as the input for the meta-learner. Based on the results of the base layer, a meta-model is developed. In this approach, we use three meta-models, and their predictions are aggregated using a voting classifier system. If the predictions of two or more meta-models are the same, the new observation is classified accordingly based on the majority vote. Here we use Gradient boosting, Extra Tree, Cat Boost, Random Forest, XG Boost, as base learners, and in the meta learner we use Logistic Regression, Support Vector Machine, K-Nearest Neighbour model.

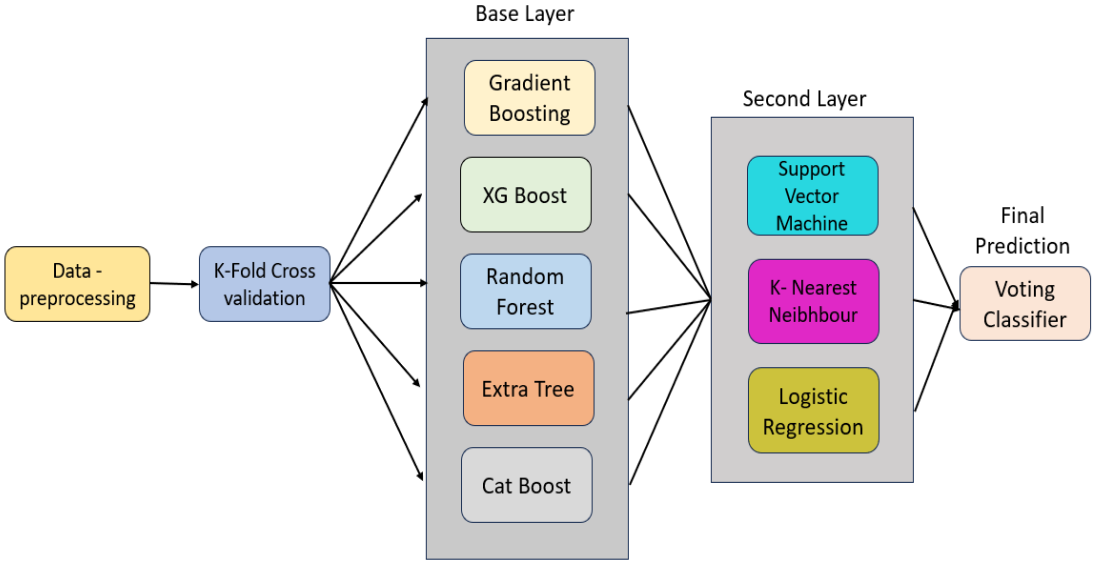

Figure 1. Proposed Model

2.5 Performance Evaluation Method

Confusion Matrix

The Confusion matrix is a tool for measuring performance, commonly used to evaluate classification problem performance. Assessing the classification model's performance involves comparing the predicted classes with the actual classes to determine its accuracy. The matrix consists of four crucial metrics that are determined by comparing actual results with predicted results. [26].

Table 2. Confusion Matrix

	Predicted Class	
Actual Class	**Yes**	**No**
Yes	TP	FN
No	FP	TN

The following key metrics are derived from the confusion matrix:

$$Accuracy\ Rate = \frac{TP + TN}{TP + TN + FN + FP}$$

Accuracy Rate measures the overall correctness of the model.

$$Precision = \frac{TP}{TP + FP}$$

Shows the proportion of true positive predictions out of all the predicted positive values.

$$Recall\ (Sensitivity) = \frac{TP}{TP + FN}$$

Alternatively known as the true positive rate, it illustrates the number of positive values that are accurately anticipated.

$$Specificity = \frac{TN}{TN + FP}$$

Show how well the model identifies the negatives it also known as the true negative rate.

$$F1\text{-}Score = 2 * \frac{Precision * Recall}{Predcision + Recall}$$

The F1 score penalises extreme values of precision and recall by taking the harmonic mean of both metrics. A perfect Precision and Recall are represented by an F1 score of 1, whereas a score of 0 indicates the model's prediction is entirely inaccurate. [27]

3. RESULT AND DISCUSSIONS

The entire dataset includes one dependent variable with ten independent variables at its disposal. First, exploratory data analysis was done, and then normalisation of the numeric variables and transformation of the categorical ones into factors were conducted. With regards to the outliers, inspection of the boxplot revealed that there were no outliers in the collected datasets. Finally, the prepared dataset was divided into the training set, containing 80 per cent of the data and the test set, containing the rest 20 per cent data. Subsequently, a variety of classifiers for classification tasks were used, such as Logistic Regression, SVM, KNN, Random Forest, Extra Trees, XG Boost, Gradient Boosting, and Cat Boost. The results obtained from these baseline models were compared with the results obtained from the proposed model. In the model that is being proposed in this paper, k-fold cross-validation was used to reduce the levels of overfitting. Cross validation was then performed, and other base algorithms used were Random Forest, Extra Trees, XG Boost, Gradient Boosting, and Cat Boost. As mentioned earlier, using the base learners, a meta-learner was trained using Logistic Regression, SVM and KNN. The last process was related to the reward-matrix & voting classifier to consolidate the outcome of the meta-learner. The method described in this paper produced better accuracy than all the baseline models developed in this study. The particulars are given below:

Table 3. Confusion Matrix of Result

Matrix	TP	TN	FP	FN
LR	119	165	34	22
SVM	124	165	29	22
KNN	108	147	45	40
RF	133	166	20	17
ET	127	170	26	17
GB	140	166	13	21
XGB	136	164	17	23
CB	142	166	11	21
Proposed	141	173	12	14

Based on the confusion matrix, we calculate the following metrics to check the performance of the classification model.

Table 4. Result of Models

Models	Accuracy	Precision	Recall	Specificity	F1-score
LR	83.53%	77.78%	84.40%	82.91%	80.95%
SVM	85.00%	81.05%	84.93%	85.05%	82.94%
KNN	75.00%	70.59%	72.97%	76.56%	71.76%
RF	87.94%	86.93%	88.67%	89.25%	87.79%

ET	87.35%	83.01%	88.19%	86.73%	85.52%
GB	90.00%	91.50%	86.96%	92.74%	89.17%
XGB	88.24%	88.89%	85.53%	90.61%	87.18%
CB	90.59%	92.81%	87.12%	93.79%	89.87%
Proposed	92.35%	92.16%	90.97%	93.51%	91.56%

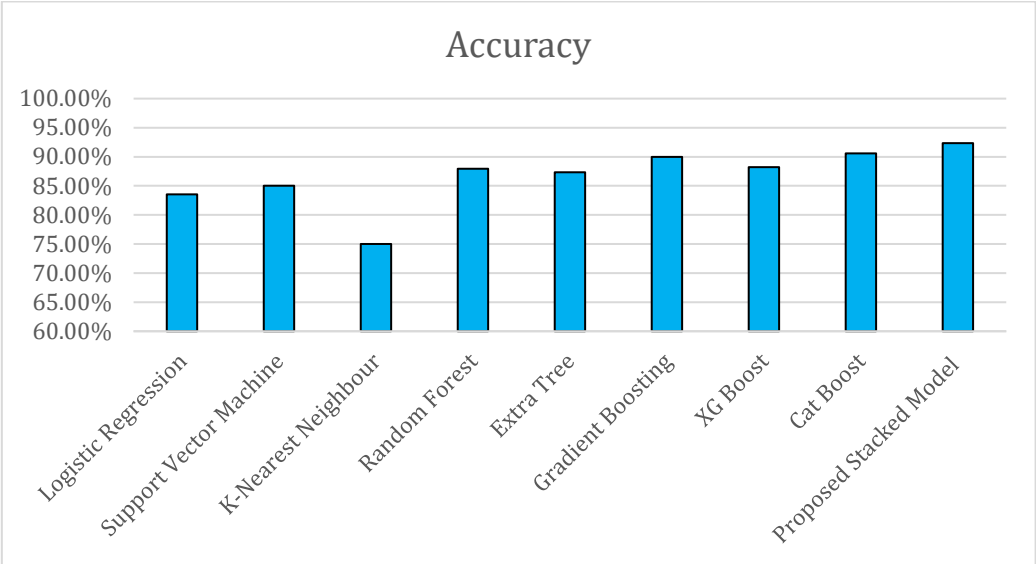

Figure 2. Comparison of Different Models' Accuracy

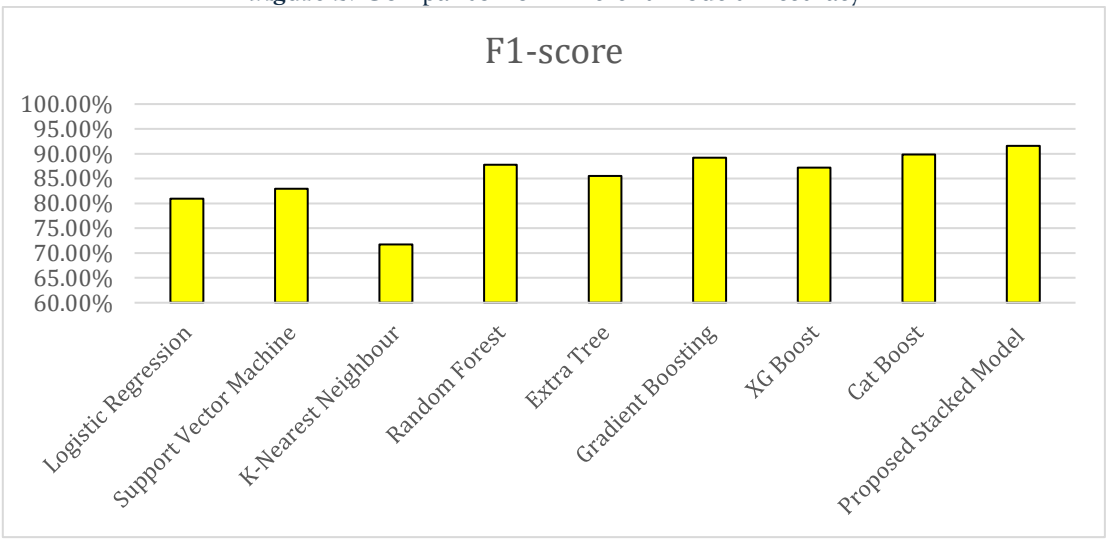

Figure 3. Comparison of Different Models F1-Score

The above table evaluates the efficacy of various machine learning models in predicting liver disease by leveraging metrics derived from the confusion matrix, including accuracy, which indicates precision, recall, specificity, and the F1 score. They evaluate the models' overall effectiveness and dependability concerning prediction accuracy and the proper balance of false positives and false negatives. Several of these models were additionally evaluated alongside an enhanced stacked model as proposed in the paper. The models delivered different results based on how their performance was evaluated. The accuracy rate for Logistic Regression reached 83.53% alongside precision at 77.78% and recall at 84.40% and specificity at 82.91% and F1-score of 80.95%. Support Vector Machine provided marginally higher prediction outcomes with performance metrics at 85.00% accuracy while precision scored 81.05%, recall achieved 84.93%, specificity reached 85.05% and the F1-score achieved 82.94%. The K-Nearest Neighbours algorithm displayed decreased performance metrics of 75.00% accuracy, together with 70.59% precision and 72.97% recall and 76.56% specificity and 71.76% F1-score. Among tree-based models, Random Forest classified patients with the best results, yielding 87.94% accuracy and 86.93% precision and recall of 88.67% and 89.25% specificity along with an F1-score of 87.79%. The Extra Trees

model achieved notable success with 87.35% accuracy alongside 83.01% precision, 88.19% recall, 86.73% specificity, as well as a well-balanced overall performance evaluation. The Gradient Boosting algorithm obtained 90.00% accuracy, as well as 91.50% precision and 86.96% recall and 92.74% specificity, which resulted in an F1-score of 89.17%. The best individual scores among the models belonged to Cat Boost, which achieved 90. The performance metrics for this particular test showed a Specificity of 93.79, together with a 59% accuracy rate, and precision of 92.81, and recall of 87.12, and an F1 score of 89.87.

The suggested stacked model proved to be the most effective, integrating Random Forest, Extra Trees, XG Boost, Gradient Boosting, and Cat Boost as its base learners. Logistic Regression, SVM, and KNN, as meta classifiers, used a voting classifier for the ultimate prediction. The stacked model demonstrated the following results: an accuracy of 92.35%, a precision of 92.16%, a recall rate of 90.97%, a specificity of 93.51%, and an F1-score of 91.56%, which supports the argument for employing multiple algorithms to predict the selected prognostic factors accurately.

4. CONCLUSION

The results distinctly indicate that the created stacked ensemble model is superior for predicting liver disease compared to all the separate models proposed. Although it is evident that models like Cat Boost and Gradient Boosting showed strong performance, the stacked model excelled with an accuracy of 92.35%, in addition to exhibiting balanced metrics in every category. The power of the stacked model lies in its scalability to leverage the advantages of various algorithms. The model provides highly dependable and uniform predictions, initially by integrating strong base models like Gradient Boosting and Cat Boost, and subsequently by employing meta learners like Logistic Regression and SVM. The voting classifier enhances reliable decision-making by minimising the mistakes from all the various models. This study demonstrates that stacking is among the most effective methods applicable in essential areas like healthcare. In particular, the remarkable effectiveness of the stacked model demonstrates that this method can deliver prompt and precise diagnosis of liver disease.

Conflicts of Interest

The author declares that he has no personal or financial interests that would influence the conclusions of this article.

Funding Statement

This study is not funded by any organisation.

Data Availability

Data is available on the Kaggle.com website

5.REFERENCES

[1.] "Liver Disease in India," *World Life Expectancy*. [Online]. Available: https://www.worldlifeexpectancy.com/india-liver-disease

[2.] D. Sindhuja and R. J. Priyadarshani, "A Survey on Classification Techniques in Data Mining for Analysing Liver Disease Disorder," *International Journal of Computer Science and Mobile Computing*, vol. 5, no. 5, pp. 483-488, 2016.

[3.] M. Cassagnou, A. Boruchowicz, F. Guillemot, Y. Gheyssens, L. Devisme, A. Cortot, and J. F. Colombel, "Hepatic steatosis revealing celiac disease: a case complicated by transitory liver failure," *American Journal of Gastroenterology*, pp. 1291-1292, 1996.

[4.] H. Jin, S. Kim, and J. Kim, "Decision Factors on Effective Liver Patient Data Prediction," *International Journal of Bio-Science and Bio-Technology*, vol. 6, pp. 167-168, 2014.

[5.] S. M. Ganie, P. K. D. Pramanik, and Z. Zhao, "Improved liver disease prediction from clinical data through an evaluation of ensemble learning approaches," *BMC Medical Informatics and Decision Making*, 2024.

[6.] G. A. Shanbhag, K. A. Prabhu, N. V. S. Reddy, and B. A. Rao, "Prediction of Lung Cancer using Ensemble Classifiers," *Journal of Physics: Conference Series*, 2022.

[7.] A. Q. Md, S. Kulkarni, C. J. Joshua, T. Vaichole, and S. Mohan, "Enhanced Preprocessing Approach Using Ensemble Machine Learning Algorithms for Detecting Liver Disease," *Biomedicines*, 2023.

[8.] A. A. Ahad, B. Das, M. R. Khan, N. Saha, and A. Zahid, "Multiclass liver disease prediction with adaptive data preprocessing and ensemble modelling," *Results in Engineering*, 2024.

[9.] D. Bhupathi, T. C. Nya-Ling, T. S. Sremath, and R. S. Kumar, "Liver disease detection using machine learning techniques," 2022.

[10.] R. A. Bhusnurmath and S. Betageri, "Deep Ensemble Stacked Technique for the Classification of Liver Disease using Artificial Neural Networks at the Base Level and Random Forest at the Meta Level," *International Journal of Computer Applications*, vol. 186, pp. 42-48, 2024.

[11.] T. A. Assegie, R. Subhashni, N. K. Kumar, J. P. Manivannan, P. Duraisamy, and M. F. Engidaye. [Note: This reference appears incomplete in the original list]
[12.] M. Ghosh, Md. Mohsin Sarker Raihan, M. Raihan, Laboni Akter, A. Kumar Bairagi, S. S. Alshamrani, and M. Masud, "A Comparative Analysis of Machine Learning Algorithms to Predict Liver Disease," *Intelligent Automation & Soft Computing*, vol. 30, 2021.
[13.] R. Amin, R. Yasmin, S. Ruhi, M. H. Rahman, and M. S. Reza, "Prediction of chronic liver disease patients using integrated projection-based statistical feature extraction with machine learning algorithms," *Informatics in Medicine Unlocked*, 2023.
[14.] M. A. Kuzhippallil, C. Joseph, and A. Kannan, "Comparative Analysis of Machine Learning Techniques for Indian Liver Disease Patients," in *6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Coimbatore, India, 2020.
[15.] M. F., H. E., W. M., and K. H., "Statistical Machine Learning Approaches to Liver Disease Prediction," *Livers MDPI*, pp. 294-312, 2021.
[16.] "Predict Liver Disease 1700 Records Dataset," *Kaggle*. [Online]. Available: https://www.kaggle.com/datasets/rabieelkharoua/predict-liver-disease-1700-records-dataset/data
[17.] S. B. Kotsiantis, D. Kanellopoulos, and P. E. Pintelas, "Data Preprocessing for Supervised Learning," *International Journal of Computer Science*, vol. 1, pp. 111-117, 2006.
[18.] P. Patil, "Outlier Detection and Removal using the IQR Method," *Medium*, September 23, 2023. [Online]. Available: https://medium.com/@pp1222001/outlier-detection-and-removal-using-the-iqr-method
[19.] D. Hosmer and S. Lemeshow, *Applied Logistic Regression*, New York: Wiley, 2013.
[20.] C. Cortes and V. Vapnik, "Support Vector Networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995.
[21.] "Support Vector Machine (SVM) Algorithm," *GeeksforGeeks*, July 4, 2024. [Online]. Available: https://www.geeksforgeeks.org/support-vector-machine-algorithm
[22.] M. Suyal and P. Goyal, "A Review on Analysis of K-Nearest Neighbour Classification Machine Learning Algorithms based on Supervised Learning," *International Journal of Engineering Trends and Technology*, vol. 70, no. 7, pp. 43-48, 2022.
[23.] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, New York, USA: Springer Series in Statistics, 2001.
[24.] C. Bentejac, A. Csorgo, and G. Martínez-Munoz, "A Comparative Analysis of XGBoost," 2019, pp. 1-20.
[25.] S. Baladram, "Extra Trees, Explained: A Visual Guide with Code Examples," *Medium*, 2024. [Online]. Available: https://towardsdatascience.com/what-when-how-extratrees-classifier. [Accessed 2024].
[26.] V. Singh and D. Jain, "A Hybrid Parallel Classification Model for the Diagnosis of Chronic Kidney Disease," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 8, pp. 14-28, 2021.
[27.] S. A. Hicks, I. Strümke, V. Thambawita, M. Hammou, M. A. Riegler, P. Halvorsen, and S. Parasa, "On evaluation metrics for medical applications of artificial intelligence," *Springer Nature*, 2022.
[28.] A. A. Ahad, B. Das, M. R. Khan, N. Saha, and A. Zahid, "Multiclass liver disease prediction with adaptive data preprocessing and ensemble modelling," *Results in Engineering*, 2024.