Predicting Liver Disease Through ML Classification Methods

Narasimha Reddy K.V
Assistant professor
Computer Science Engineering
Narasaraopeta EngineeringCollege
Guntur, India
narasimhareddy@gmail.com

Manoj Menyim
Computer Science Engineering
Narasaraopeta EngineeringCollege
Guntur, India
menyimmanoj@gmail.com

Devudubabu Gollu
Computer Science Engineering
Narasaraopeta EngineeringCollege
Guntur, India
babugollu8742@gmail.com

Abdul Khayum Shaik
Computer Science Engineering
Narasaraopeta EngineeringCollege
Guntur, India
abdulkhayum@gmail.com

Abstract—**Machine Liver disease is a major global cause of mortality, affecting a significant portion of the population worldwide. Various factors such as obesity, undiagnosed hepatitis infections, and alcohol misuse can contribute to liver damage, leading to serious complications like abnormal nerve function, bleeding, kidney failure, jaundice, and liver encephalopathy. Early diagnosis is crucial for effective treatment, and modern technologies like sensors are being utilized for early detection of infections.The objective of this study is to assess the effectiveness of different Machine Learning (ML) algorithms in reducing the cost and complexity associated with diagnosing chronic liver disease. ML is a powerful tool used to uncover patterns in vast datasets, enabling automated decision-making processes. The dataset used in this research comprises information from liver patients, obtained from the UCI Repository, making it a supervised learning task.By leveraging historical patient data and employing various ML algorithms such as Logistic Regression, Decision Trees, Random Forests, K-Nearest Neighbors (KNN), Gradient Boosting, Extreme Gradient Boosting (XGBoost), and LightGBM, this study aims to predict future patient outcomes accurately. Feature selection techniques were applied to enhance algorithm performance, resulting in promising accuracy rates across the different models tested.**

KEYWORDS:LogisticRegression,DecisionTrees,RandomForests,KNN,GradientBoosting,XGBoost,LightGBM,Smote.

Introduction

The liver, which occupies the right portion of the abdominal cavity, is the largest gland in the human body and the second-largest organ after the skin. It performs a variety of vital activities. The liver, which weighs about 2% of an adult's body weight (1.4–1.8 kg for men and 1.2–1.4 kg for women), carries out more than 500 critical tasks that are necessary for our life. These include the production of serum proteins and lipids, the secretion of bile and glycogen, the detoxification of blood from endogenous and exogenous chemicals, and the storage of vital vitamins such as D, A, K, E, and B12.

Liver disease impairs the liver's capacity to function normally and is characterized by hepatic enlargement brought on by a variety of substances, germs, or genetic abnormalities. Between the ages of 40 and 60, liver problems are common, with a higher incidence in men. An estimated 10 lakh individuals are diagnosed with liver disease in India alone each year, which leads to over 1.4 lakh fatalities annually.

Healthcare has benefited from the use of machine learning (ML), a branch of artificial intelligence (AI), which helps with early diagnosis and effective therapy. In order to properly predict the existence of liver illness, machine learning algorithms can evaluate large, complicated datasets, such as the liver patient dataset [2] employed in this work. Since liver disease symptoms might be difficult to identify in the early stages, early detection of liver disorders is essential for increasing patient survival rates.

Liver problems comprise a range of ailments, each with unique symptoms and causes, including fatty liver disease, viral hepatitis, hereditary liver diseases, autoimmune liver diseases, and alcoholic liver disease. These conditions carry significant health concerns, such as liver failure or cancer, when they worsen to cirrhosis. Therefore, it is crucial to diagnose liver diseases as soon as possible and to treat them with tests like Liver Function Tests (LFTs). LFTs, which include tests for transaminases, albumin, and bilirubin, among others, assist medical professionals in evaluating the health of the liver and determining whether more testing is required for a precise diagnosis and course of therapy. The management of liver disease can be greatly impacted by early intervention and lifestyle modifications, which can also enhance patient outcomes.

This study evaluated many Machine Learning (ML) models, such as Gradient Boosting, XGBoosting, K-Nearest Neighbor (KNN), Decision Tree, Logistic Regression, and Random Forest, to predict liver illness in patients. The prediction accuracy increased as a consequence of the researchers' attention to and improvement over problems that earlier study had missed.

Exploratory Data Analysis (EDA), data pre-processing, outlier removal, application of the Synthetic Minority Over-sampling Technique (SMOTE), and use of both basic and sophisticated ML algorithms were the procedures involved in predicting liver illness from the dataset.583 entries from the Indian Liver Patient Dataset (ILPD)[2], obtained from the UCI Machine Learning Repository, made up the dataset

used in this study. Of these records, 167 were to non-liver patients, while 416 were from liver patients. The northeastern part of Andhra Pradesh, India, is where the data was gathered. The term "Dataset" refers to the class label that is used to group things together (e.g., liver patient or not).

RELATED WORK

Rajeswari and Reena [4] investigated the use of data mining algorithms in the analysis of liver problems. To extract useful insights and patterns about liver illnesses, the study probably applied a variety of data mining techniques, feature selection, and data preparation approaches. The performance comparison of several algorithms in correctly detecting liver problems may have been covered in the study, offering insightful information for medical diagnosis and therapy.

A unique machine learning strategy using boosting algorithms was presented for the categorization of liver illness by N. Afreen, R. Patel, M. Ahmed, and M. Sameer [5]. The study probably investigated how well boosting algorithms classified cases of liver illness, offering insights into sophisticated machine learning methods for medical diagnostics.

Barnaghi, Sahzabi, and Bakar [7] . Their study explored the subtleties and effectiveness of several classification strategies, providing important information about how they compare. They made significant contributions to the discipline by analyzing a variety of techniques and illuminating the advantages and disadvantages of each strategy. This comparative study serves as a fundamental resource for comprehending classification schemes in various settings.

Prof. MS Prasad Babu and BR Sarath Kumar [10] introduced a unique method of automatic liver status diagnostics utilizing Bayesian classification techniques in their paper published in Ramana . Their work demonstrates the potential of machine learning techniques in healthcare and represents a major breakthrough in medical diagnostics. By using a novel approach, they add to the increasing corpus of research on the use of computational tools in medical diagnostics, especially in liver-related health evaluations. This work emphasizes how crucial Bayesian categorization is to enabling automated and precise health assessments.

Afzal, Masroor, and Beg [11] investigated the use of ultrasound scoring criteria in the evaluation of chronic liver disease. Their research explores the viability and efficacy of liver health assessment utilizing ultrasonography-based scoring systems. The authors provide insightful analysis of these factors, which helps to advance methods for diagnosing long-term liver diseases. This study constitutes a major endeavor to improve the efficacy and precision of liver disease diagnosis using imaging methods.

Machine learning algorithms were used to predict fatty liver disease in a 2019 study by Wu et al., demonstrating developments in medical diagnostics. The study highlights the use of computational techniques in utilizing data for precise disease prognosis, especially in evaluations of liver health. This study highlights the potential of machine learning to improve healthcare outcomes by offering insightful information about using it for predictive modeling in medical contexts [13].

Using imbalanced datasets, Arbain and Balakrishnan's [14] study investigated a number of data mining methods for the prediction of liver illness. They evaluated the efficacy of several approaches, offering suggestions for enhancing the precision of disease prediction in healthcare analytics. The study advances the field of medical data analysis and is published in the International Journal of Data Science and Advanced Analytics.

PROPOSED METHODS

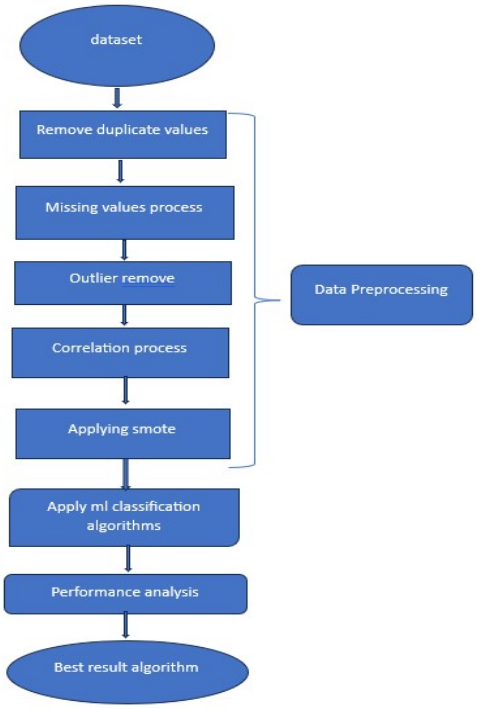

Fig 1.Steps in the method.

The above fig 1 shows the data pretreatment procedures are shown in the flowchart and include handling the dataset, finding and eliminating duplicate values, filling in missing values, and spotting outliers. It also has a step where variables are correlated and the Synthetic Minority Over-sampling Technique (SMOTE) is used to address class imbalance. After that, different algorithms are used to process the data, and a performance comparison is used to determine which algorithm performs best for predicting liver disease on unbalanced datasets.

PROPOSED SYSTEM

Our model is proposed based on certain criteria as follow.

A. Data Collection:-

Data Gathering , In order to choose important records for analysis and to use different data mining techniques to produce or constructively gain knowledge, data selection is crucial.

B. Data Exploration:-

Before moving on to more in-depth research, data exploration functions as the first stage of data analysis, with the main objective being to summarize and comprehend the dataset. It entails looking through data to find trends, abnormalities, and correlations between different variables. A range of visual aids, including correlation matrices, box plots, scatter plots, and histograms, are frequently employed to examine data and find possible patterns.

C. Data Preprocessing:-

1.Eliminating redundant values:

Eliminating redundant values from the dataset is a crucial measure to improve the efficiency and quality of the data. This procedure entails locating and removing duplicate records or entries that could distort analysis findings or cause errors. Data integrity is enhanced by removing duplicates, guaranteeing that analysis and ensuing decision-making processes are founded on correct and trustworthy information.

2.Missing values:

Imputation of missing values is the process of addingmissing values to a dataset to close the gaps. This method was applied to four missing variables pertaining to the albumin and globulin ratio in the Indian liver disease patient dataset. The median values for these variables were used to fill in the missing values. Through this procedure, the dataset is kept complete and analytically useful, preserving data correctness and integrity for statistical modeling and decision-making.

3. Outlier detection and eradication:-

Finding and addressing data points that substantially differ from the rest of the dataset as a result of measurement mistakes or anomalous behavior is known as outlier detection and eradication. Univariate outliers, which are anomalies in a single feature, and multivariate outliers, which take into account the interactions among several characteristics or attributes in the dataset, are the two basic categories into which outliers can be divided. Multivariate outliers, like those found in the features of the ILPD dataset, consider the correlations and distributions across several dimensions of the data in addition to extreme values within a single variable. Univariate outliers are detected based on these characteristics. In order to ensure that aberrant data points do not unnecessarily affect statistical analyses or machine learning algorithms, the process of outlier detection and elimination attempts to improve the quality and reliability of the dataset for analysis and modeling purposes. Maintaining the integrity and correctness of the data analysis outputs depends on accurately identifying and dealing with outliers.

4. Resampling:-

A technique called Synthetic Minority Over-sampling Technique (SMOTE) is used to re-sample in order to correct for the imbalance in the linear dataset, which shows a large excess of liver ailment cases over non-liver illness cases. In order to create a balanced dataset, SMOTE creates synthetic samples for the minority class, or people without liver ailment, and places them strategically among existing cases. By preventing bias against the majority class in machine learning models, this balanced dataset enhances overall model accuracy and reliability in predicting cases of both liver illness and non-liver illness.

D. Machine learning Classification Techniques:-

Machine learning algorithms are trained utilizing a variety of methods during the categorization process. This involves classifying and categorizing data according to particular traits and patterns using various machine learning techniques.

1.Logistic Regression Algorithm :-

Based on input factors, the binary classification process known as logistic regression is used to forecast the likelihood that a given class label will occur. It models the link between independent factors and the probability of the outcome variable by fitting the data to a logit function. The resultant logistic function, which aids in class label determination based on a predetermined threshold, is commonly depicted by an S-shaped curve called the sigmoid function.

2.Naive Bayes algorithm :-

The Naive Bayes method is regarded as an eager learning algorithm since it can swiftly identify novel instances without the need for extra training on test data. It does this by using Bayes' theorem to statistical classification. In terms of classification performance, it is similar to decision trees and neural networks. Many Naive Bayes classifiers, including GaussianNB and MultinomialNB, are included in the sklearn library. The two fundamental tenets of Naïve Bayes are that each predictor contributes equally to the result and that predictors are independent of one another. Its classification formula, which is based on the Bayes theorem, uses the prior probability, likelihood, and evidence components to determine the conditional probability of a class given predictor values.

3.K-Nearest Neighbor (KNN) algorithm:-

The 'K' stands for the number of close neighbors that are taken into consideration for classification in the K-Nearest Neighbor (KNN) algorithm, which classifies data by evaluating similarity. To find closest neighbors, it uses

distance metrics such as Manhattan, Euclidean, Chebyshev, or Hamming. In contrast to other algorithms, KNN requires data normalization for best results but does not require training data. KNN is considered a non-parametric, instance-based, lazy learning algorithm since it doesn't assume anything about the functional form of the problem, uses training instances for predictions, and doesn't rely on a pre-trained model.

4.Decision Tree Algorithm **:-**

A non-parametric supervised classifier, the Decision Tree approach builds a tree-like structure with nodes and branches. Every node represents a choice made in response to an attribute value, which might result in several branching and child nodes as the final product. The optimum features for splitting nodes in a decision tree are determined by using metrics like entropy and the Gini index during the creation process.
The Gini index, which goes from 0 to 1, calculates the likelihood that a sample will be incorrectly classified. Lower values indicate better attribute selections for node splitting. The formula Another criterion for building decision trees is entropy, which evaluates the homogeneity or heterogeneity

5.Random Forest algorithm:-

The Random Forest algorithm is an ensemble learning method that enhances accuracy by combining several separate decision trees into an ensemble. It uses bootstrap samples from the dataset to train are combined to create the final prediction, which yields reliable results for both regression and classification applications.

Random Forest reduces the danger of overfitting prevalent in individual decision trees by building numerous trees using distinct subsets of data and applying the bootstrap sampling approach. When it comes to classification tasks, it chooses the mode of predicted classes to identify the class, but in regression tasks, it computes the average of projected values over all trees. With this method, generalization and predictive power are enhanced.

6.Gradient Boosting algorithm:-

The Gradient Boosting algorithm is a sequential learning method that builds a sequence of weak learners, usually decision trees, gradually in order to minimize a differentiable loss function. By fitting successive models to the residuals, or the disparities between predicted and actual values, it focuses on minimizing the mistakes of earlier models. The algorithm adds new base models one at a time while maintaining the models that have already been added, iteratively improving model predictions by minimizing the loss function through the use of a gradient descent technique. By ensuring that every new model corrects the faults of the ensemble, this stage-by-stage additive approach improves overall forecast accuracy.

RESULTS:-

In order to assess the model's predictive accuracy for the diagnosis of liver disease, we compare its performance indicators and use the ILPD dataset to investigate the model's efficacy. Finding the most dependable method or strategy for precise forecasts is made easier with the help of this comparative analysis.

A. Dataset:-

We compare the model's performance using the ILPD dataset [2] in order to assess the model's predictive capacity for liver disease.The ILPD dataset includes 583 entries about liver health in total, 416 of which are associated with patients who have been diagnosed with liver diseases and 167 of which are associated with patients who do not have liver conditions. The northeastern part of Andhra Pradesh, India, was the location of the data gathering. In this dataset, the class label "Dataset" is used to differentiate between those classified as liver patients and those classified as non-liver patients.The characteristics/attributes found in this dataset are:-

```
Data columns (total 11 columns):
 #   Column
---  ------
 0   Age
 1   Gender
 2   Total_Bilirubin
 3   Direct_Bilirubin
 4   Alkaline_Phosphotase
 5   Alamine_Aminotransferase
 6   Aspartate_Aminotransferase
 7   Total_Protiens
 8   Albumin
 9   Albumin_and_Globulin_Ratio
 10  Dataset
```

B. Methods of Preprocessing:-

1.Removing duplicate values:-

Eliminating duplicate values from the dataset is vital to enhance data quality and efficiency. This process involves identifying and removing redundant records to prevent distortions in analysis findings. By eliminating 13duplicate values from the dataset, data integrity is improved, ensuring reliable outcomes for decision-making processes. .The below fig 2 shows the number of duplicates values and how to drop those values.

```
df.duplicated().sum()
```

```
13
```

```
df.drop_duplicates(inplace=True)
df.duplicated().sum()
```

```
0
```

Fig 2..duplicate values

2.Filling missing values:-

Correct analysis and model performance depend on addressing missing values during data preprocessing.

Restoration using median values becomes crucial because the dataset contains 4 missing values for the albumin and globulin ratio. This popular imputation technique maintains the distribution of data for numerical variables by substituting central tendencies such as the mean, median, or mode for missing values.By using median() ,missing values are filled .The below fig 3 shows the missing values and filling those values by using the median().

```
df['Albumin_and_Globulin_Ratio'].fillna(df['Albumin_and_Globulin_Ratio'].median(),inplace=True)
df.isnull().sum()
```

```
Age                         0
Gender                      0
Total_Bilirubin             0
Direct_Bilirubin            0
Alkaline_Phosphotase        0
Alamine_Aminotransferase    0
Aspartate_Aminotransferase  0
Total_Protiens              0
Albumin                     0
Albumin_and_Globulin_Ratio  0
Dataset                     0
dtype: int64
```

Fig 3 . Missing values

3.Elimination of Outliers:-

Removing anomalies from datasets is crucial for preserving the accuracy of modeling and data analysis procedures. To avoid distorting results, outliers—which may be the consequence of mistakes or infrequent occurrences—must be dealt with. By identifying outliers according to how different they are from the dataset mean, methods such as the Standard Deviation Method can be used to effectively remove outliers and guarantee correct analysis. This method is essential for preserving the dependability and quality of data in statistical modeling applications.

```
from scipy import stats
attributes_with_outliers = ['Age', 'Gender','Total_Bilirubin', 'Direct_Bilirubin',
       'Alkaline_Phosphotase', 'Alamine_Aminotransferase',
       'Aspartate_Aminotransferase', 'Total_Protiens',
       'Albumin_and_Globulin_Ratio']
def remove_outliers_zscore(df, attributes):
    for attr in attributes:
        z_scores = stats.zscore(df[attr])
        df = df[(z_scores < 3) & (z_scores > -3)]
    return df

data= remove_outliers_zscore(df, attributes_with_outliers)
```

Fig 4.outlier removal

The above fig 4 provided code for removes outliers from certain characteristics in a pandas DataFrame by using the z-score technique from the scipy library. Z-scores are computed for each attribute in order to detect outliers, or data points that deviate more than three standard deviations from the mean. By filtering the DataFrame to keep data points inside the permitted z-score range, the function essentially eliminates outliers from the dataset.

4.Coefficient of Correlation :-

Indicating the degree and direction of a linear relationship between variables, the correlation coefficient "r" goes from -1 to +1. Perfect positive linear relationships are denoted by a correlation of 1, perfect negative linear relationships are represented by a correlation of -1, and there is no correlation at all. Weaker linear links are indicated by positive but less than perfect correlations ($0 < r < 1$) and negative correlations ($-1 < r < 0$), where the variables either move in opposite directions or increase together. These connections between the variables in the dataset are shown by the correlation coefficients in the table in below Fig 5.

	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Aminotransferase	Aspartate_Aminotransferase
Age	1.000000	-0.063376	0.120356	0.115562	0.013648	-0.066961	-0.025955
Gender	-0.063376	1.000000	-0.174467	-0.178576	-0.065860	-0.125357	-0.127567
Total_Bilirubin	0.120356	-0.174467	1.000000	0.948825	0.337643	0.245064	0.330939
Direct_Bilirubin	0.115562	-0.178576	0.948825	1.000000	0.342079	0.292051	0.341458
Alkaline_Phosphotase	0.013648	-0.065860	0.337643	0.342079	1.000000	0.305767	0.253471
Alamine_Aminotransferase	-0.066961	-0.125357	0.245064	0.292051	0.305767	1.000000	0.709476
Aspartate_Aminotransferase	-0.025955	-0.127567	0.330939	0.341458	0.253471	0.709476	1.000000
Total_Protiens	-0.209868	0.082814	-0.100238	-0.069033	-0.017490	0.029430	-0.034154
Albumin	-0.273250	0.083630	-0.264788	-0.237816	-0.150650	0.037628	-0.072413
Albumin_and_Globulin_Ratio	-0.212954	0.022558	-0.294891	-0.286567	-0.251163	-0.002322	-0.075850
Dataset	-0.149688	0.074074	-0.236109	-0.246140	-0.206624	-0.216537	-0.216051

Fig 5. Correlation coefficient table.

4.1 Correlation heatmap :-

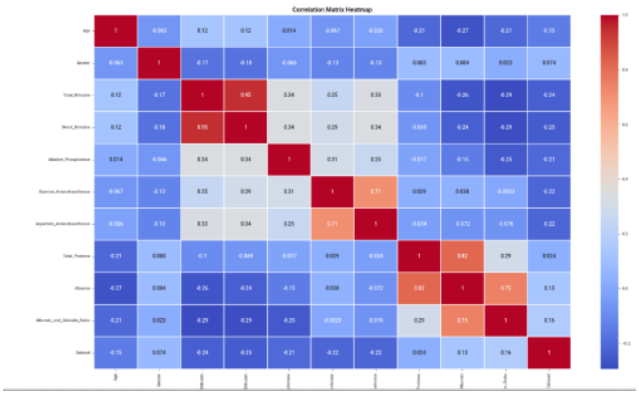

Fig 6. Coorelation heatmap

The above fig 6 shows the correlation heatmap uses color-coded matrix columns to visually represent relationships between variables. Positive correlations are represented by warm hues like red and orange, whereas negative correlations are represented by chilly hues like blue. Finding trends, dependencies, and multicollinearity in datasets is made easier with the help of this heatmap, which facilitates exploratory data analysis and feature selection. It offers perceptions into the ways in which variables interact and impact the target variable, improving comprehension for tasks involving predictive modeling.

4.2 Correlation heatmap after removing correlated colunms:-

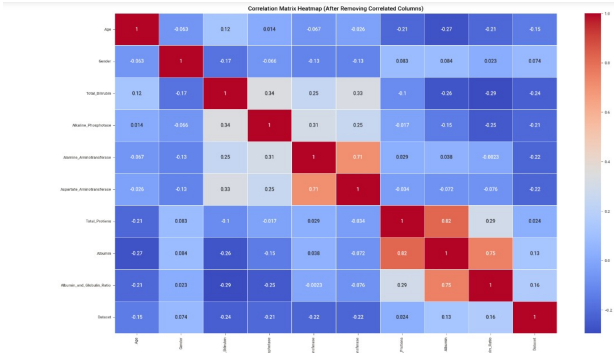

Fig 7. After Coorelation

The fig 7 uses a correlation threshold of 0.85 to find highly connected features in a dataset. A new DataFrame called "df_filtered" is produced when a set of correlated characteristics are created and subsequently eliminated from the dataset, which is represented by the variable "data." The remaining feature correlations are then displayed on a heatmap by the code, which also creates a correlation matrix for the filtered DataFrame. 'Direct_Bilirubin' is one redundant column that is likely highly associated with other columns in the dataset; by removing it, this approach helps reduce multicollinearity and improve model performance.

5.SMOTE:-

Synthetic Minority Oversampling Technique, or SMOTE for short, is a crucial machine learning approach for situations involving class imbalance. In medical datasets, when positive cases are more prevalent than negative ones, class imbalances arise when one class is noticeably smaller

Models may favor the majority class as a result of this imbalance, which would result in subpar performance on the minority class. To counter this, SMOTE rebalances the dataset, generates synthetic samples for the minority class, and improves the model's capacity to learn from the underrepresented class. By guaranteeing that the model is trained on a more balanced dataset, this rebalancing enhances the model's performance, particularly on instances of minority classes.

```
from imblearn.over_sampling import SMOTE
from collections import Counter
smote = SMOTE()
X_train_smote, y_train_smote = smote.fit_resample(X,y)
print("Before SMOTE: ", Counter(y))
print("After SMOTE: ", Counter(y_train_smote))
```

```
Before SMOTE:  Counter({1: 334, 0: 162})
After SMOTE:  Counter({1: 334, 0: 334})
```

Fig 8. Before and After Smote

6.Confusion matrix:-

One important assessment technique for determining how well machine learning models perform, especially in classification tasks, is a confusion matrix. It provides an organized table that makes the degree to which a model's predictions match the target variable's actual values easier to see.

The confusion matrix essentially divides predictions into four groups:

		Predicted	
		Negative (N) -	Positive (P) +
Actual	Negative -	True Negative (TN)	False Positive (FP) Type I Error
	Positive +	False Negative (FN) Type II Error	True Positive (TP)

Fig 9. Confusion matrix

The above fig 9 shows that.

True Positives (TP) are instances where the model accurately predicts positive outcomes for real positive cases.

True Negatives (TN) are situations where the model predicts a negative outcome with accuracy and these predictions align with actual negative cases.

False Positives (FP): Conditions where the model incorrectly identifies negative cases as positive and mispredicts positive outcomes.

False Negatives (FN): Predicting negative outcomes incorrectly, the model confuses positive situations for negative ones.

A.Accuracy :-

A criterion called accuracy is used to assess how well a model predicts the future. It shows the percentage of accurate forecasts among all of the forecasts. A model that performs better is indicated by a greater accuracy value.

In mathematical terms, accuracy is determined by dividin total number of records by the number of accurately predicted records:

$$\text{Accuracy} = ((TP)+(TN)) \ / (TP+TN+FP+FN)$$

The total population is the sum of TP, TN, false positives (FP), and false negatives (FN), where TP stands for true positive predictions and TN for true negative predictions. A clear picture of the model's overall performance in accurately identifying examples across all classes is given by accuracy.

B.Precision:-

Precision measures how accurate positive class predictions are by dividing the number of true positive predictions by the total number of positive predictions generated by the model.

$$\text{Precision} = TP / (TP + FP)$$

C.Recall:-

Recall, sometimes referred to as Sensitivity or True Positive Rate (TPR), quantifies the percentage of real positive cases that the model accurately predicts.

Recall=TP / (TP + FN)

D. F1 score:-

For classification models, the balance between recall and precision is represented by the F1 score.

F1 score =2 * (Recall * Precision) / (Recall + Precision).

E. Specificity:-

Specificity measures the accuracy of negative Class predictions, calculated as Specificity = TN / (TN + FP), with higher values indicating a superior model in handling negative cases.

F. Sensitivity:-

Sensitivity, which is computed as Sensitivity = TP / (TP + FN), indicates how well the classifier can identify positive cases. A high number indicates that the classifier is good at identifying positive examples.

G. False Positive Rate:-

False Positive Rate, which is computed as False Positive Rate = FP / (FP + TN), quantifies the classifier's rate of mistakenly labeling real negative cases as positive. This information is useful in assessing the classifier's performance with respect to negative examples.

H. False Discovery Rate:-

False Discovery Rate, which can be calculated as False Discovery Rate = FP / (FP + TP), measures the percentage of expected positive cases that are really negative and provides information about the accuracy of the classifier's positive predictions.

I.False Omission Rate:-

The classifier's accuracy in making negative predictions for positive instances is highlighted by False Omission Rate, which computes the rate of actual positive instances that are mistakenly categorized as negative by the classifier. False Omission Rate = FN / (FN + TN).

J.ROC CURVE:-

To help with evaluating the performance of the model, the ROC curve graphically depicts the trade-off between the true positive rate (sensitivity) and false positive rate across various categorization thresholds.

7. Performance analysis:-

The metrics for the best model is:-

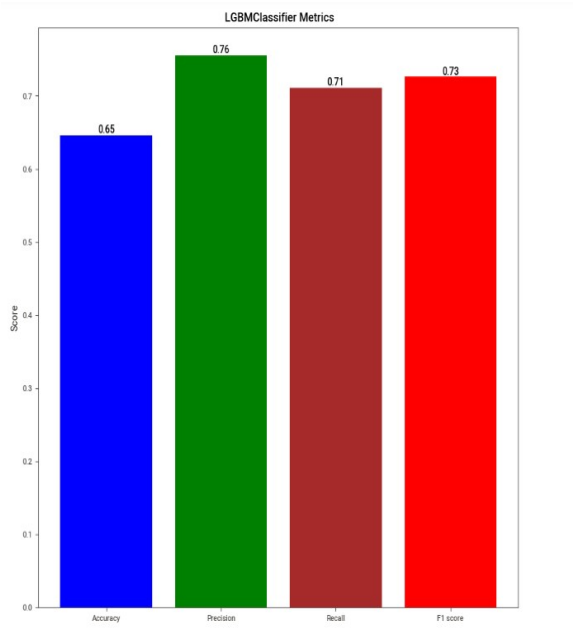

Fig 10.Best model metrics

The above fig 10. Shows that The LGBM classifier model outperformed all other machine learning classification models in terms of performance metrics. It obtained an F1 score of 73%, recall of 71%, accuracy of 65%, and precision of 76%. These metrics emphasize the model's efficacy in liver illness prediction by showing its high precision in recognizing positive cases, large recall in catching pertinent instances, and balanced F1 score taking both precision and recall into account.

The Roc curve for the best model is:-

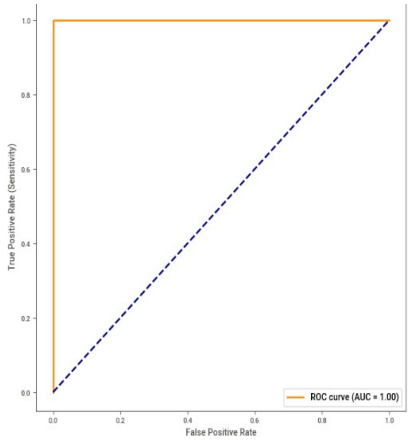

Fig 11.Roc curve for LGBM classifier

The above fig 11. Shows that With an astounding Area Under the Curve (AUC) value of 1.00, the ROC curve

for the LGBM classifier demonstrated flawless discriminating power between true positive and false positive rates. This remarkable AUC value indicates that the model may perform optimally in terms of classification across a range of threshold values. The model's strong ability to discriminate between cases of liver disease and non-disease is demonstrated by the curve's closeness to the top-left corner, which also represents the model's high sensitivity and specificity.

The performance analysis between all the machine learning models / classification techniques is:-

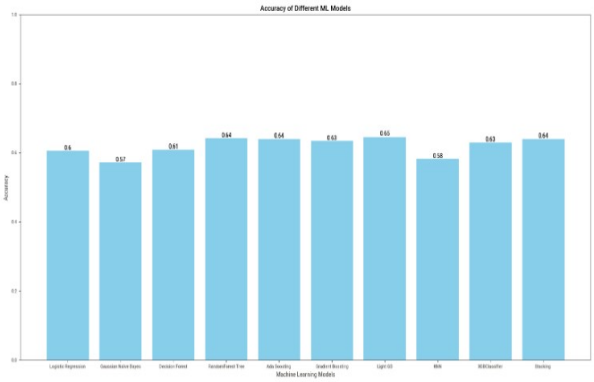

Fig 12. Accuracy of diff models.

The above fig 12 shows that the Various machine learning models were compared based on their accuracy scores in predicting liver disease. Logistic Regression and Decision Tree models achieved an accuracy of 60%, while KNN and Gaussian Naïve Bayes models scored lower at 58% and 57%, respectively. Random Forest algorithm showed improved accuracy at 64%, followed closely by Gradient Boosting and Ada Boosting at 63%. The highest accuracy was achieved by Light GBM with 65%.

FINAL RESULT:-

Algorithm	Accuracy	precision	recall	F1 score
LR	60	78	57	65
GNB	57	89	42	55
DF	60	72	67	69
RFT	64	74	71	72
Ada Boost	63	78	65	70
GB	63	76	67	70
Light GB	64	75	71	72
KNN	58	74	57	63
XG Boost	62	73	71	72
Stacking	63	75	69	71

Fig 13. Result

[LR= logistric regression, GNB=Gaussian Naïve Bayes,DF=Decision Tree,RFT= Random Forest Tree, GB= Gradient Boosting]

The above fig 13 compare different machine learning models to predict liver illness, and find that their performance measures differ. With an F1 score of 65%, recall of 57%, precision of 78%, and accuracy of 60%, Logistic Regression was successful. KNN demonstrated a 58% accuracy rate, 74% precision rate, 57% recall rate, and 63% F1 score. With a precision of 72%, recall of 67%, accuracy of 60%, and F1 score of 69%, the decision tree performed well. With an accuracy of 64%, precision of 74%, recall of 71%, and F1 score of 72%, Random Forest performed well.

Switching to Gaussian Naïve Bayes, it showed 57% accuracy, 78% precision, 42% recall, and a 55% F1 score. Gradient Boosting obtained an F1 score of 70%, recall of 67%, accuracy of 63%, and precision of 76%. With Ada Boosting, the results were as follows: 63% accuracy, 78% precision, 65% recall, and 70% F1 score. With an accuracy of 65%, precision of 75%, recall of 71%, and F1 score of 72%, Light GB demonstrated the highest performance. XGBoost obtained an F1 score of 72%, recall of 71%, accuracy of 62%, and precision of 73%.

The assessment focuses on trade-offs between different models' accuracy, precision, recall, and F1 score. Models with balanced performance include Logistic Regression and Decision Trees, whereas Gaussian Naïve Bayes exhibits differences in recall and precision. Random Forest, Gradient Boosting, Ada Boosting, Light GB, and XGBoost are examples of ensemble approaches that perform well on a variety of metrics. Light GB stands out in terms of accuracy, highlighting the significance of taking certain task needs into account for the best model selection.

Conclusion:-

Our conclusion explores the use of rigorous data analysis and model evaluation methodologies to predict liver sickness in patients. To improve model performance and accuracy, the preprocessing of the data included removing outliers, using dummy encoding for categorical variables, and imputing missing values using the median.

The preprocessed data was subjected to a number of classification techniques, including LightGBM, Gradient Boosting, Extreme Gradient Boosting (XGBoost), K-Nearest Neighbors (KNN), Decision Tree, Random Forest, and Logistic Regression. After extensive testing and analysis, it was found that the Random Forest, LightGBM, and AdaBoosting algorithms produced results with greater accuracy than the others.

Based on the properties of the dataset and metrics for evaluating the models, the results indicate that Random Forest and LightGBM models, in conjunction with AdaBoosting, showed greater performance in predicting liver illness. Based on the research and analysis done for the paper, the study concludes that LightGBM is a good

algorithm for accurate liver disease prediction, demonstrating its robustness and effectiveness in tasks involving medical diagnosis.

Reference:-

[1] M. Sameer and B. Gupta, "Beta Band as a Biomarker for Classification between Interictal and Ictal States of Epileptical Patients," in *2020 7th International Conference on Signal Processing and Integrated Networks (SPIN)*, 2020, pp. 567–570, doi: 10.1109/SPIN48934.2020.9071343.

[2] Indian Liver Patient Records. Available in http://www.kaggle.com/datasets/uciml/indian-liver-patient-records

[3] M. Sameer, A. K. Gupta, C. Chakraborty, and B. Gupta, "Epileptical SeizureDetection: Performance analysis of gamma band in EEG signal Using Short-Time Fourier Transform," in *2019 22nd International Symposium on Wireless Personal Multimedia Communications (WPMC)*, 2019, pp. 1–6, doi: 10.1109/WPMC48795.2019.9096119.

[4] Rajeswari, P., & Reena, G.S. (2010). Analysis of liver disorder using data mining algorithm. Global journal of computer science and technology, 10(14), 48-52.

[5] N. Afreen, R. Patel, M. Ahmed, and M. Sameer, "A Novel Machine Learning Approach Using Boosting Algorithm for Liver Disease Classification," in 2021 5th International Conference on Information Systems and Computer Networks (ISCON), 2021, pp. 1–5.

[6] Vijayarani, S., & Dhayanand, S. (2015). Liver disease prediction using SVM and Naïve Bayes algorithms. International Journal of Science, Engineering and Technology Research (IJSETR), 4(4), 816–820.

[7] Barnaghi, P.M., Sahzabi, V.A., & Bakar, A.A. (2012). A comparative study for various methods of classification. In International Conference on Information and Computer Networks, 27(2), 875-81.

[8] Shaheamlung, G., Kaur, H., & Kaur, M. (2020). A survey on machine learning techniques for the diagnosis of liver disease. Proceedings of International Conference on Intelligent Engineering and Management, ICIEM 2020, 337–341. https://doi.org/10.1109/ICIEM48762.2020.9160097

[9] Joloudari, J. H., Saadatfar, H., Dehzangi, A., & Shamshirband, S. (2019). Computer- aided decision-making for predicting liver disease using PSO-based optimised SVM with feature selection. Informatics in Medicine Unlocked, 17, 100255. https://doi.org/10.1016/j.imu.2019.100255

[10] Ramana, B.V. (2010). Prof. MS Prasad Babu and BR Sarath Kumar: "New Automatic Diagnosis of Liver Status Using Bayesian Classification". In IEEE International Conference on Intelligent Network and Computing (ICINC 2010), 26-29.

[11] Afzal, S., Masroor, I., & Beg, M. (2013). Evaluation of chronic liver disease: does ultrasound scoring criteria help?. International journal of chronic diseases, 2013. http://dx.doi.org/10.1155/2013/326231

[12] El-Shafeiy, E. A., El-Desouky, A. I., & Elghamrawy, S. M. (2018). Prediction of liver diseases based on machine learning technique for big data. International Conference on Advanced Machine Learning Technologies and Applications, 362–374.

[13] Wu, C.-C., Yeh, W.-C., Hsu, W.-D., Islam, M. M., Nguyen, P. A. A., Poly, T. N., Wang, Y.-C., Yang, H.-C., & Li, Y.-C. J. (2019). Prediction of fatty liver disease using machine learning algorithms. Computer Methods and Programs in Biomedicine, 170, 23–29. https://doi.org/10.1016/j.cmpb.2018.12.032

[14] Arbain, A. N., & Balakrishnan, B. Y. P. (2019). A comparison of data mining algorithms for liver disease prediction on imbalanced data. International Journal of Data Science and Advanced Analytics (ISSN 2563-4429), 1(1), 1–11.Retrieved from http://ijdsaa.com/index.php/welcome/article/view/2

[15] Carvalho, J.R.; Machado, M.V. New insights about albumin and liver disease. *Ann. Hepatol.* 2018, *17*, 547–560. [Google Scholar] [CrossRef] [PubMed]