Finding Patterns in Liver Function Test results to interpret Well-defined Liver Diseases

Harshil Kananthoor[1], Rajat Kumawat[2], Rahul Mohata[3], Vaishnavi Bisen[4]

[1]Harshil Kananthoor, School Of Engineering And Technology, Dy Patil University Ambi Pune
[2]Rajat Kumawat, School Of Engineering And Technology, Dy Patil University Ambi Pune
[3]Rahul Mohata, School Of Engineering And Technology, Dy Patil University Ambi Pune
[4]Vaishnavi Bisen, School Of Engineering And Technology, Dy Patil University Ambi Pune

---***---

Abstract - Liver function tests (LFTs) are crucial for diagnosing and monitoring liver diseases, but interpreting their results is often challenging due to the complex patterns associated with various conditions. This study aims to identify distinct LFT result patterns that differentiate specific liver diseases, enhancing diagnostic accuracy. We will analyze a comprehensive dataset of LFT results from patients with hepatitis, cirrhosis, fatty liver disease, and liver cancer. Key parameters include ALT, AST, ALP, GGT, total and direct bilirubin, albumin, and prothrombin time. Advanced statistical and machine learning techniques, such as clustering algorithms, principal component analysis (PCA), and decision trees, will be employed to identify correlations and patterns among these parameters. The findings will be validated using a separate dataset to ensure reliability. We anticipate discovering specific combinations and ranges of LFT parameters that correspond to different liver diseases. These patterns will be presented in a simplified format to aid clinicians in making accurate and timely diagnoses. This research aims to enhance the interpretive value of LFTs, providing a robust framework for diagnosing liver diseases and potentially improving patient outcomes through earlier and more precise treatments.

***Key Words*:** Liver Function Tests, Pattern Recognition, Liver Diseases, Diagnostic Accuracy, Machine Learning, Biomarkers.

1.INTRODUCTION

Liver diseases are a major global health issue, causing significant morbidity and mortality. The liver's critical roles in metabolism, detoxification, and protein synthesis make it vulnerable to various disorders, each with unique biochemical profiles. Liver function tests (LFTs) are essential for diagnosing and monitoring these diseases by measuring enzymes, proteins, and other blood substances to assess liver health. However, interpreting LFT results is challenging due to the complex and overlapping patterns of abnormalities. Advancements in data science and machine learning offer powerful tools for analyzing large datasets, identifying patterns, and making predictions. These technologies can uncover subtle biochemical patterns in LFT results that may not be apparent to clinicians, leading to more accurate and timely diagnoses. Python, with its extensive data science libraries like Pandas, NumPy, Scikit-learn, and TensorFlow, is ideal for processing and analyzing LFT data. This study explores the use of machine learning and Python to identify patterns in LFT results, aiming to develop predictive models that distinguish between various liver diseases. By employing techniques such as data preprocessing, feature selection, and machine learning algorithms, this research aims to enhance the interpretive value of LFTs, reduce diagnostic uncertainty, and improve patient care. The ultimate goal is to integrate machine learning into clinical practice, facilitating early detection and better management of liver diseases.

2. Body of Paper

Literature Survey

The integration of machine learning in medical diagnostics, particularly in interpreting liver function tests (LFTs), is a burgeoning field with significant potential. This literature review surveys recent studies and key contributions, highlighting methodologies and their implications for liver disease diagnosis. Joshi, Dhoble, and Bhatnagar provide a comprehensive review of machine learning applications in medical diagnostics, discussing various algorithms and their effectiveness across medical domains. Lee, Wu, and Kuo explore the use of machine learning algorithms to predict liver disease, focusing on feature selection and model accuracy. Shams and Rahman apply machine learning techniques to identify patterns in hepatic disease data, comparing the performance of different algorithms. Patel and Shah discuss the application of various data mining algorithms in diagnosing liver diseases, with an emphasis on decision trees and neural networks. Zhang and Chen evaluate the performance of multiple machine learning models in predicting liver disease, highlighting feature importance and model optimization. Gupta and Kumar investigate the use of pattern recognition and machine learning techniques to analyze liver function test results, identifying key biomarkers. Kim and Lee explore how machine learning enhances liver disease diagnosis by comparing traditional diagnostic methods with machine learning-based approaches. Rao and Singh present predictive modeling approaches for liver disease using data analytics and machine learning, emphasizing practical applications in clinical settings. Wang and Zhao focus on developing automated diagnostic systems for liver diseases using various machine learning techniques, assessing their clinical utility. Finally, Nguyen and Tran analyze liver function test results using machine learning to diagnose liver diseases, emphasizing the importance of feature engineering. These studies collectively underscore the transformative potential of machine learning in the field of hepatology, offering new avenues for improving diagnostic accuracy and patient outcomes.

Proposed System

The system for interpreting liver function tests (LFTs) involves several key features to ensure comprehensive data analysis and clinical integration. First, it will gather LFT results from diverse sources such as hospitals, clinics, and research databases, ensuring compatibility and standardization of data formats for seamless integration. During preprocessing, the data will be cleansed to eliminate errors, outliers, and inconsistencies, and missing values will be handled through imputation or deletion while preserving data integrity. Feature extraction will involve identifying and extracting pertinent features from LFT results, including enzyme levels (e.g., ALT, AST), protein concentrations (e.g., albumin, total protein), bilirubin levels, and other relevant biochemical markers indicative of liver function. For pattern analysis, the system will utilize statistical methods (e.g., clustering, regression analysis) and machine learning algorithms (e.g., decision trees, support vector machines) to discern patterns within the LFT data. This will include exploring temporal patterns, correlations between different markers, and anomalies that signify specific liver conditions, with an emphasis on evaluating the robustness and generalizability of identified patterns across diverse patient populations. Visualization tools such as line charts, scatter plots, and heatmaps will be developed to elucidate patterns and trends within the LFT data, enhanced by interactive features that allow users to explore data subsets and drill down into specific details. Finally, the system will be designed to seamlessly integrate with existing clinical workflows, providing decision support tools that assist healthcare professionals in interpreting LFT results and making informed diagnoses. Compliance with regulatory standards (e.g., HIPAA in the United States) will be ensured to safeguard patient privacy and data security. The project's objectives are to develop a comprehensive, scalable database for aggregating LFT results from various sources and to implement preprocessing techniques to cleanse and standardize this data, ensuring high quality and reliability. Feature extraction methods, including traditional statistical and machine learning approaches, will be explored to identify informative features. Machine learning models will be applied to recognize patterns indicative of specific liver diseases, and classifiers will be trained for automated diagnosis and risk stratification. The system's accuracy, sensitivity, and specificity will be rigorously evaluated across diverse patient populations. Collaboration with healthcare professionals will ensure the system integrates seamlessly into clinical workflows, that enhance clinical decision-making and interoperability with electronic health record (EHR) systems.

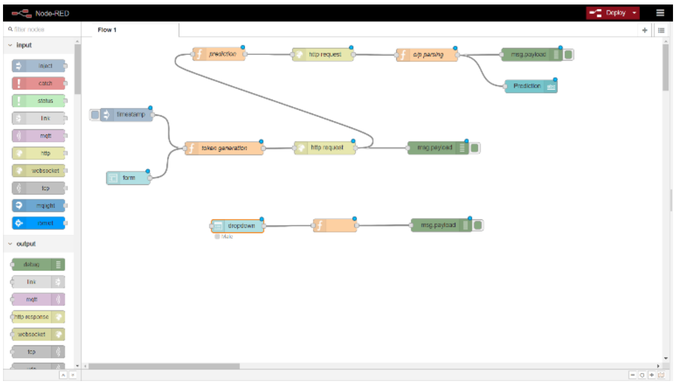


Fig -1: Block Diagram of System

Node-RED, developed by IBM, is a visual programming tool used for connecting hardware devices, APIs, and online services, primarily within the Internet of Things (IoT) ecosystem. Its interface features a palette of nodes on the left, a central workspace, and a sidebar for configuring node properties. Users can create flows by dragging and dropping nodes from the palette to the workspace. The flow illustrated in the image is organized into three main sections: prediction, token generation, and form handling. The prediction section includes nodes for preparing data, sending HTTP requests to an external API for predictions, parsing the API response, and displaying the result. Key nodes include a prediction node for data preparation, an HTTP request node for making API calls, an output parsing node, and a debug node to display results for testing purposes. The token generation section ensures secure API requests by generating unique, time-bound tokens. It includes a timestamp node, a token generation node, and an HTTP request node that uses the generated token, with debug nodes for logging results. The form handling section manages user inputs through forms and dropdown menus. It features nodes for capturing and processing user inputs, such as a form node for input fields, a dropdown node for selections like gender, and a function node for processing the inputs. Debug nodes are used to verify the captured and formatted inputs, ensuring they are correctly prepared for subsequent processing.

Methodology

In disease classification problems, simply comparing the accuracy—the ratio of correct predictions to total predictions—is insufficient. This is because the context, such as the severity of the disease, often dictates that it is more critical to avoid falsely predicting a disease as non-disease (false negatives), while incorrectly predicting a healthy person as diseased (false positives) carries a relatively less severe penalty. Therefore, we will use the F-beta score as a performance metric, which is the weighted harmonic mean of precision and recall. Precision and recall are defined as follows:

Precision=TP/ (TP+FP), Recall=TP/ (TP+FN)

where TP is True Positive, FP is False Positive, and FN is False Negative. The F-beta score can then be expressed as:

F-beta score = (1+β2)*precision*recall/((β 2 *precision)+recall)

ISSN: 2583-6129
DOI: 10.55041/ISJEM02050

This score allows us to adjust the balance between precision and recall according to the value of β, providing a more nuanced evaluation of the model's performance in disease classification. The software requirements for the project encompass a comprehensive suite of tools and frameworks necessary for data processing, analysis, and application development. Python serves as the primary programming language, offering a vast array of libraries such as NumPy, pandas, and scikit-learn for various tasks including data manipulation, statistical analysis, and machine learning. In addition, visualization tools like matplotlib or seaborn enhance data representation. For web development, options include Flask or Django frameworks, along with frontend technologies like HTML, CSS, and JavaScript. Database management systems such as SQLite, PostgreSQL, or MySQL are utilized for storing LFT data, with SQLAlchemy serving as the ORM for Python database interactions. The hardware requirements entail a robust computing setup with a multi-core processor, sufficient RAM (at least 8GB), and SSD storage for efficient data processing and model training. Graphics Processing Units (GPUs), specifically NVIDIA GeForce GTX or RTX series, offer acceleration for deep learning tasks, though cloud-based GPU instances are viable alternatives. High-speed internet connectivity is essential for accessing online resources and collaboration tools. Supported operating systems, including Windows, macOS, or Linux (preferably Ubuntu), provide the platform for development, ensuring compatibility, resource management, security, performance, and ease of use.

Flowchart and Algorithm

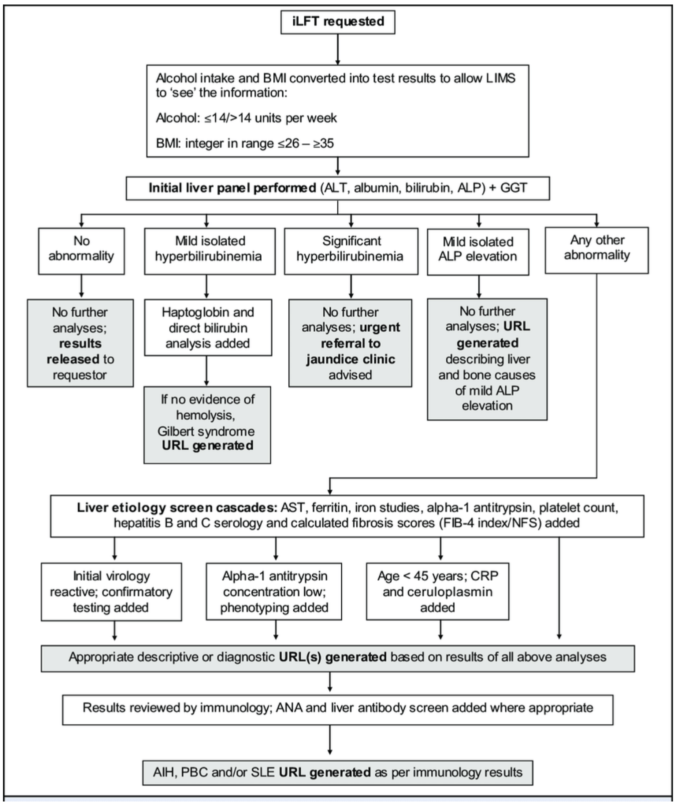

iLFT requested
Alcohol intake and BMI converted into test results to allow LIMS to 'see' the information:
Alcohol: ≤14/>14 units per week
BMI: integer in range ≤26 – ≥35
Initial liver panel performed (ALT, albumin, bilirubin, ALP) + GGT
No abnormality
Mild isolated hyperbilirubinemia
Significant hyperbilirubinemia
Mild isolated ALP elevation
Any other abnormality
No further analyses; results released to requestor
Haptoglobin and direct bilirubin analysis added
No further analyses; urgent referral to jaundice clinic advised
No further analyses; URL generated describing liver and bone causes of mild ALP elevation
If no evidence of hemolysis, Gilbert syndrome URL generated
Liver etiology screen cascades: AST, ferritin, iron studies, alpha-1 antitrypsin, platelet count, hepatitis B and C serology and calculated fibrosis scores (FIB-4 index/NFS) added
Initial virology reactive; confirmatory testing added
Alpha-1 antitrypsin concentration low; phenotyping added
Age < 45 years; CRP and ceruloplasmin added
Appropriate descriptive or diagnostic URL(s) generated based on results of all above analyses
Results reviewed by immunology; ANA and liver antibody screen added where appropriate
AIH, PBC and/or SLE URL generated as per immunology results

Fig -2: Data Flow Diagram

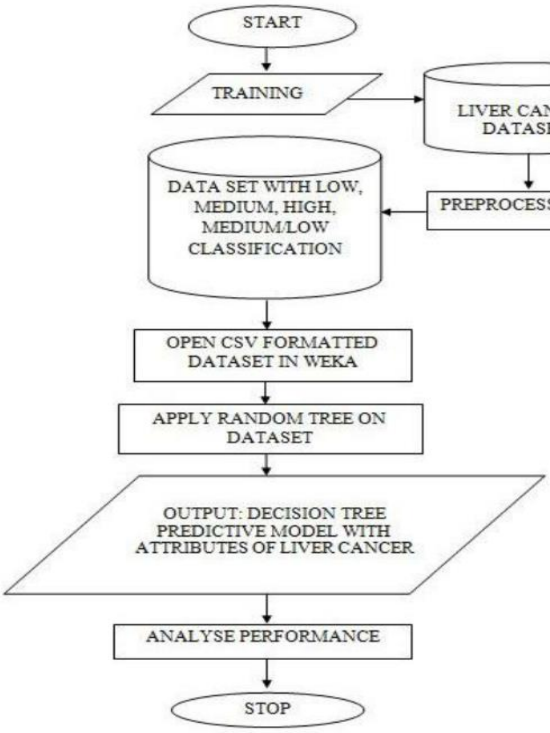

START
TRAINING
LIVER CANCER DATASET
DATA SET WITH LOW, MEDIUM, HIGH, MEDIUM/LOW CLASSIFICATION
PREPROCESS DATA
OPEN CSV FORMATTED DATASET IN WEKA
APPLY RANDOM TREE ON DATASET
OUTPUT: DECISION TREE PREDICTIVE MODEL WITH ATTRIBUTES OF LIVER CANCER
ANALYSE PERFORMANCE
STOP

Fig -3: Entity Relationship Model

Three distinct supervised learning approaches have been selected for solving the problem at hand, ensuring a broad coverage of possible methodologies. To prevent redundancy, approaches from the same family, such as Random Forest and Ada Boost, have been avoided in favor of diverse techniques. Each algorithm undergoes hyperparameter tuning using grid search cross-validation to optimize its performance.

Random Forest Classifier: This ensemble learning method entails the creation of multiple decision trees, each independently trained on a subset of the data. Key hyperparameters, including the number of trees (n_estimators), maximum depth of each tree (max_depth), and the number of features considered for splitting (max_features), are fine-tuned to enhance the classifier's accuracy. Additionally, the out-of-bag score (oob_score) is evaluated to assess prediction error. The achieved accuracy for this classifier stands at 0.68.

Gaussian Naive Bayes Classifier: Unlike Random Forest, Gaussian Naive Bayes relies on probabilistic principles, estimating class probabilities and conditional probabilities of input values given each class. This classifier assumes that features are independent, given the class label, and calculates probabilities accordingly. Despite its simplicity, it achieves a respectable accuracy of 0.5613.

Logistic Regression: Logistic Regression models the probability of a binary outcome using a logistic or sigmoid function. It assigns weights to features based on their importance and continuously adjusts these weights to minimize the difference between predicted and actual values. Logistic Regression yields an accuracy of 0.7143, making it the most accurate among the selected algorithms.

Each algorithm offers unique strengths and weaknesses, contributing to a comprehensive exploration of the problem space.

Result and Discussion

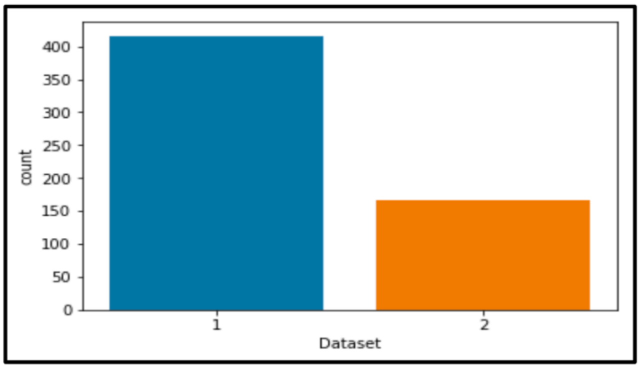


Fig -4: COUNT PLOT OF LIVER PATIENTS DIAGNOSED

Number of patients diagnosed with liver disease: 416

Number of patients not diagnosed with liver disease: 167

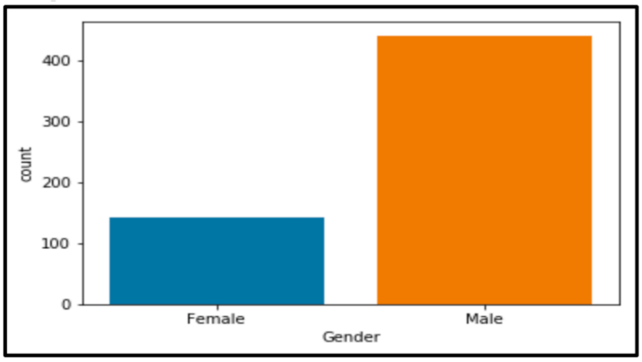


Fig -5: COUNT PLOT OF MALE & FEMALE PATIENTS

Number of patients that are male: 441

Number of patients that are female: 142

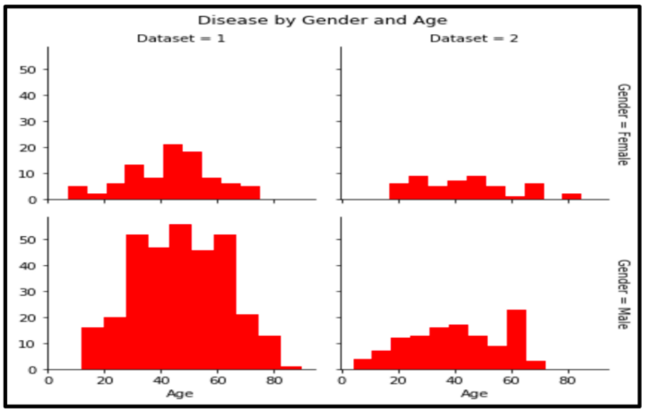


Fig -6: FACETGRID ON DISEASE BY GENDER AND AGE

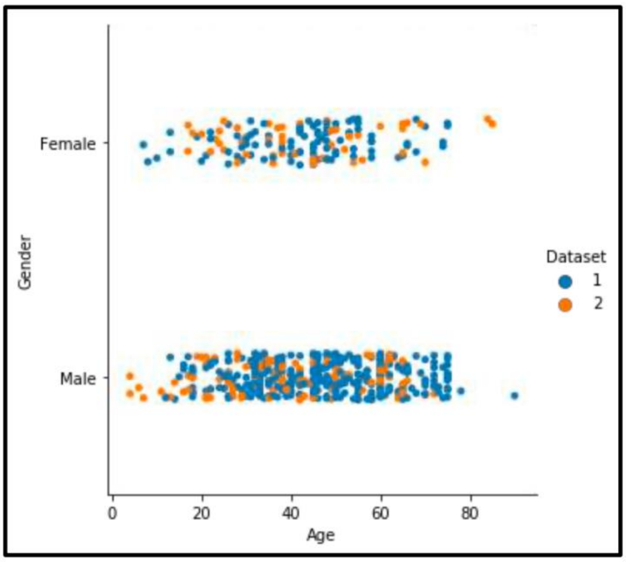


Fig -7: CATPLOT BASED ON AGE, GENDER & DATASET

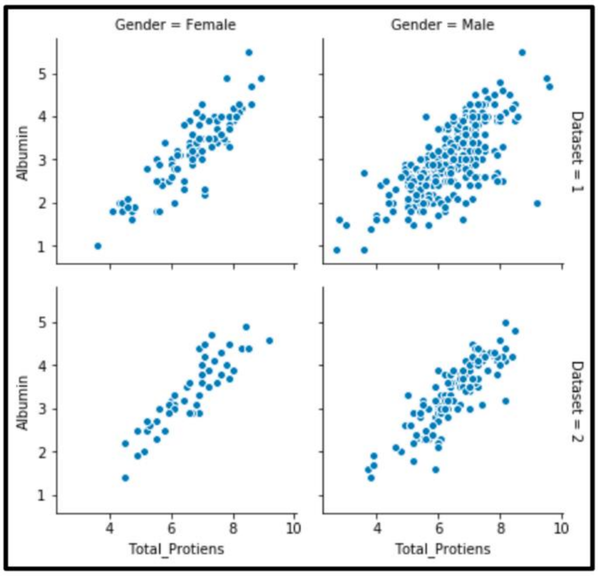


Fig -8: FACETGRID ON DIRECT_BILIRUBIN &TOTAL_BILIRUBIN

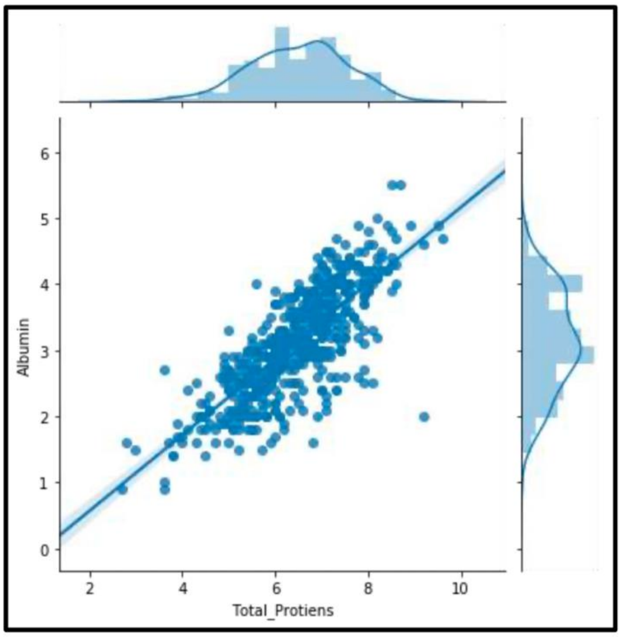


Fig -9: JOINTPLOT ON DIRECT_BILIRUBIN &TOTAL_BILIRUBIN

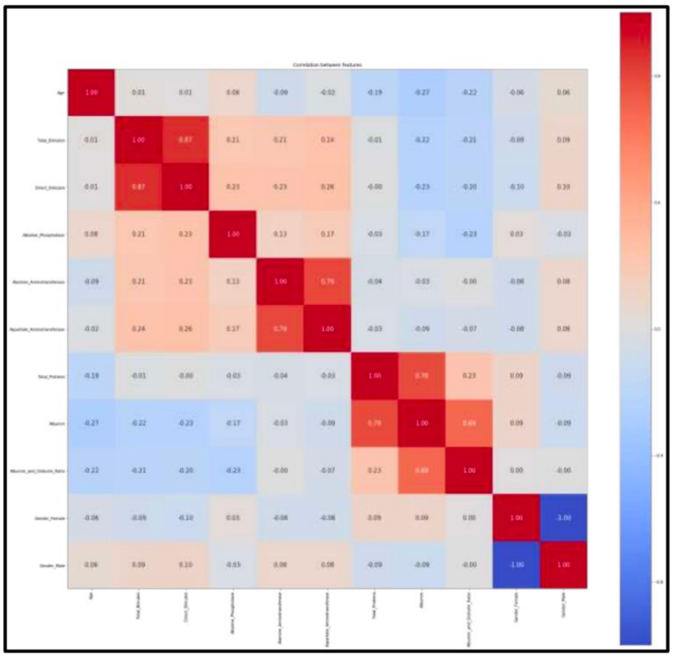
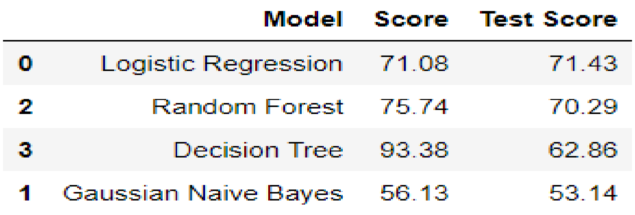


Fig -10: CO-RELATION GRAPGH

The dataset utilized for liver patient classification, sourced from the UCI Machine Learning Repository and representing Andhra Pradesh in 2017, presents certain limitations that impact the model's performance and generalizability. These limitations include a small dataset size and similarity between training and test datasets, which hinders the model's ability to generalize to larger datasets or different populations. To address these challenges and enhance classification accuracy, several recommendations are proposed. Firstly, acquiring a more precise dataset with larger sample sizes and a greater number of attributes can improve the model's predictive capabilities. Additionally, collecting data from diverse regions and over different time periods can capture broader trends and variations, leading to a more robust and representative model. Furthermore, exploring feature engineering techniques can enhance the discriminative power of features, improving the model's ability to differentiate between liver patients and non-patients. Finally, experimenting with more sophisticated machine learning algorithms or ensemble methods can further enhance classification performance. By implementing these recommendations, the accuracy and reliability of the liver patient classification model can be significantly improved, enabling better predictions and insights into liver disease diagnosis and management.



	Model	Score	Test Score
0	Logistic Regression	71.08	71.43
2	Random Forest	75.74	70.29
3	Decision Tree	93.38	62.86
1	Gaussian Naive Bayes	56.13	53.14

Fig -11: Resultant score of models used

Initially, the dataset was explored and made ready to be fed into the classifiers. This was achieved by removing some rows containing null values, transforming some columns which were showing skewness and using appropriate methods (one-hot encoding) to convert the labels so that they can be useful for classification purposes.

3. CONCLUSIONS

In conclusion, the development of a system to analyze patterns in liver function test (LFT) results holds immense potential for advancing liver disease diagnostics. By addressing the limitations of traditional diagnostic methods, this project aims to provide a more accurate, efficient, and personalized approach to liver disease diagnosis and management. The integration of data collection, pre-processing, feature extraction, pattern analysis, and visualization into a cohesive system ensures that healthcare professionals have access to powerful diagnostic tools that can improve patient outcomes. The significance of this project extends beyond individual patient care. By creating a centralized database of LFT results and uncovering new diagnostic markers, the system can contribute to the broader field of hepatology research. This can lead to the development of novel diagnostic and therapeutic strategies, ultimately benefiting patients worldwide. The expected results at each stage of the project highlight the practical benefits of this approach. From enhanced diagnostic accuracy to improved clinical efficiency and personalized medicine, the system promises to make a substantial impact on the diagnosis and treatment of liver diseases. By leveraging advanced computational techniques and collaborating with medical professionals, this project represents a significant step forward in the quest for better liver health.

ACKNOWLEDGEMENT

We express our sense of gratitude towards our project guide **Prof.Priyadarshini Patil** for the valuable guidance at step of study of this project and also for the necessary guidelines and timely co-operation during the completion of project. We are thank full to our Project Coordinator Dr. Moresh Mukhedkar, Head of the Department Prof. Malayaj Kumar and all the staff member who extended their support for the project. We are very much thank full to respected Dean Prof. Dr. Pranav Charkha for their support and providing all facilities to complete the Project. Finally, we want to thank to all of my friends for their support and suggestions. Last but not the least we want to express thanks to our family for giving us support and confidence at each and every stage of completion of Project.

REFERENCES

1. Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. MIT Press.
2. Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems* (2nd ed.). O'Reilly Media.
3. Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.
4. Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.
5. Bates, D. W., & Gawande, A. A. (2003). Improving Safety with Information Technology. *New England Journal of Medicine*, 348(25), 2526-2534.
6. Schardt, C., Adams, M. B., Owens, T., Keitz, S., & Fontelo, P. (2007). Utilization of the PICO framework to improve searching PubMed for clinical questions. *BMC Medical Informatics and Decision Making*, 7, 16.
7. Node-RED Official Documentation. (n.d.). Retrieved from https://nodered.org/docs/
8. Node-RED Cookbook. (n.d.). Retrieved from https://cookbook.nodered.org/
9. GitHub. (n.d.). Node-RED Projects and Repositories. Retrieved from https://github.com/node-red
10. Stack Overflow. (n.d.). Retrieved from https://stackoverflow.com/
11. Coursera. (n.d.). Machine Learning and Data Science Courses. Retrieved from https://www.coursera.org/