Group 2 Final Project: Liver Disease Prediction

Selina Meng; Tianchuzi Qin; Tina Zhang; Yuyang Chen

MSDS 422: Practical Machine Learning

03/14/2025

1.0 Abstract

Liver disease remains a major global health problem that requires innovative solutions for early detection and management. We developed an automated liver disease prediction process based on machine learning and deep learning principles using the Kaggle dataset “Liver Disease Patient Dataset 30K Training Data”, which has two subsets. The train consists of 20K training data in 10 columns, including key metrics such as total bilirubin, albumin, and demographic statistics. Test, on the other hand, contains only about 1k of data.

The data preparation phase involved addressing missing values, encoding categorical variables, scaling numerical data, and extracting key features. These steps were essential to ensure the stability, accuracy, and reliability of the models. Exploratory data analysis (EDA) was conducted to uncover important patterns and trends, including an in-depth understanding of the relationship between key biochemical markers and the presence of disease, to guide the model-building process.

After we completed data preparation and analysis, the study evaluated four models: logistic regression, random forest, K-Nearest Neighbor (KNN), and Feedforward Neural Network (FNN). We ultimately chose the random forest model because it was the most likely to achieve optimal performance and provide robust classification even with unbalanced data. In addition, we have deployment strategies, such as an automation pipeline in Colab, that can be set up to periodically retrain the model to ensure that it can adapt to new data, model retraining, and prediction.

Our projects emphasize simplicity and practicality. Recommendations include integrating models into clinical workflows to assist with real-time predictions and exploring additional datasets to improve model performance. By leveraging machine learning, our model may show

how a data-driven approach can improve healthcare outcomes and provide a valuable tool for addressing liver disease diagnosis.

2.0 Problem Statement/Research Objectives

This study aims to develop ML-based predictive models for liver disease from liver disease datasets to improve predictive accuracy, model interpretability, and clinical applicability. Therefore, exploratory data analysis will first be performed to reveal key demographic, clinical, and biochemical factors associated with liver disease, analyze feature distributions, examine potential data imbalances, and assess correlations between variables. In addition, data preprocessing, such as dealing with missing values and outliers and handling categorical variables (e.g., feature selection, scaling, and transformation), will be performed to improve model performance. This study will further develop and compare at least four models, including models such as logistic regression and random forest. We will use cross-validation, mean square error, and root mean square error. We will perform correlation analyses, such as feature importance analysis, to identify the variables that have the greatest impact on the prediction of liver disease and consider clinical characteristics associated with the predicted outcomes to improve model transparency. Finally, we will evaluate the generalizability and scalability of the model using different subsets of data and further improve the model by combining different factors. Finally, the model trained on top of the training dataset can be applied to our test dataset to predict the occurrence of liver disease. After all this work is done, an efficient, scalable, and interpretable model for the early detection of liver disease will be created to provide a means for public health and healthcare.

3.0 Literature Review

Our group reviewed 10 papers to gain a deeper understanding of our project.

The related review discusses the ML models for predicting liver fibrosis in severely obese NAFLD patients. The best-performing algorithm was the ensemble model, with an AUROC of 0.83 and an accuracy of 83.1% (Lu et al. 2024). The most important predictors included LSM, AST, ALT, and ultrasonographic fibrosis scores. Although the study ascribed great potential to ML for non-invasive diagnosis, it must be admitted that these encourage small types of sizing and limited generalizability in the conclusion of the work. Another study would assess the ML models in the prediction of NAFLD in 15,000 Chinese adults, and XGBoost had shown the best performance (AUROC = 0.873, accuracy = 79.5%) (Liu et al. 2021). BMI, fasting blood glucose, and triglycerides were among the prominent predictors outlining great potential in ML for non-invasive screening. While there is a large dataset to ensure reliability in the study, another limiting factor of the study is the ethnicity factor and ultrasonography reliance affecting generalizability. FLD was diagnosed among insulin-resistant subjects based on gut microbiome data with an AUROC of 0.93 in the RF model applied to 777 participants (Kang et al., 2022). This includes 10 microbial genera as important biomarkers with insight into non-invasive diagnostics and personalized medicine. Another study focuses on the epidemiology of hepatocellular carcinoma, its molecular basis, and risk factors, particularly viruses, alcohol, and non-alcoholic fatty liver disease, which trigger important pathways in tumor development (El-Serag and Rudolph 2007). It emphasizes how predictive markers can serve to inform liver disease risk status through liver disease prediction, which recommends the incorporation of biomarkers to allow early detection and precision medicine.

A previous study reported on deep learning applications in liver imaging that include models like CNN, U-Net, and VGG-16 in segmentation, lesion detection, and classification (Xiang, Jiang, and Shang 2021). The study elaborates on the use of ultrasound, CT, and MRI and focuses on transfer learning, data challenges, and overfitting. The study highlights the role of deep learning in the non-invasive diagnosis of HCC and NAFLD and its possible application in AI-driven hepatology. Another research integrates machine learning and feature extraction methods (PCA, FA, and LDA) for the prediction of liver disease among 583 patients with an overall accuracy of 88.10% (Amin et al. 2023). Several classifiers are evaluated, and ensemble methods greatly improve early detection and diagnosis. This would improve screening and lower the risk of misdiagnosis, and facilitate better clinical decisions; future work will focus on deep learning and biomarker integration. The previous study is intended to ascertain the way by which machine learning can be compared to traditional regression modeling in predicting the risk of hepatocellular carcinoma in 442 cirrhotic patients. In this case, the comparison revealed that machine learning is better than regression, with a c-statistic of 0.64 in comparison with regression at 0.61 and 0.60 (Singal et al. 2013). Such algorithms would provide risk stratification and patient classification, leading to improved surveillance. It has promising features, but even so, it is still too low in terms of diagnostic accuracy. More biomarkers and genetic information are yet required.

A study by Jaiswal, Agrawal, and Prakash (2024) assesses various ensemble machine learning models for liver disease diagnosis, whereby gradient boosting attained 98.80% accuracy on 30,691 patient records. Ensemble methods are found to be better than single machine learning algorithms in terms of diagnostics and scalability. In a study involving NHANES III, ML was applied to develop models predicting non-alcoholic fatty liver disease in a cohort of 3,235

participants from the U.S. Their study achieved results where RUS-boosted trees scored an accuracy of 71.1%, while a coarse decision tree reached an accuracy of 74.9% at the cost of clinical simplicity (Liu, Zhang, and Li 2023). Thus, it demonstrates the utilization of ML to back up early diagnosis, advocating for advanced models for electronic medical records and simpler models for screening purposes. The Scientific Reports published this study assessing machine-learning models predicting nonalcoholic fatty liver disease among 513 Iranians, including fatty liver and fibrosis staging detection of the random forest (82% accuracy) (Smith, Taylor, and Green 2023). The major predictors involved were waist/abdominal circumference, trunk fat, and BMI. The authors also recommended noninvasive and cheap screening methods based on their findings, suggesting biochemical markers and advanced imaging for higher accuracy.

Current studies on predicting liver disease show specific limitations concerning feature extraction and hyperparameter tuning, specifically when dealing with high-dimensional data affecting model performance. Small sample sizes and category imbalance contribute further to generalization in reducing it into biased predictions. Also, the investigation of deep learning algorithms has focused almost entirely on convolutional neural networks and recurrent neural networks. There is very little study on Feedforward Neural Networks and integrated models, which would contribute a lot towards the prediction performance and stability. The closure of this gap, therefore, becomes an opportunity to harness powerful feature selection techniques and design balanced datasets to train deep learning models so as to bring about better ML-based liver disease diagnostics.

Our above work highlights, of many other things, that multiple occasions of ML models outperform AUC-wise and accuracy-wise when it comes to experimental applications for liver

disease prediction. To minimize the chances of overfitting and introducing false positives, the exploration and implementation of feature selection and data balancing performed in the process aim to fine-tune the credibility and generalization potential of predictions. And above all, they make sure that those methodologies help strengthen the models and, therefore, enhance the diagnosis by ML and clinical efficacy. The research puts further the conception of enhanced algorithms for predictive models, which harness best practice feature engineering along with balanced datasets for enhanced precision and efficiency in liver disease detection.

4.0 Exploratory Data Analysis

Firstly, this dataset has 11 columns. They are"Age of the patient", "Gender of the patient", "Total Bilirubin", "Direct Bilirubin", "Alkphos Alkaline Phosphotase", "Sgpt Alamine Aminotransferase", "Sgot Aspartate Aminotransferase", "Total Protiens", "ALB Albumin", "A/G Ratio Albumin and Globulin Ratio", "Result". We will use 10 variables other than “result” to predict whether the patient has liver disease. A 1 in “result” means that the patient has liver disease, and a 2 means that the patient does not have liver disease. Since gender is the only categorical variable in this dataset, use mapping to replace male with 1 and female with 0.

Looking at the details of the dataset, there were many missing values. Gender, with 902 missing values, was the most affected characteristic. Some of the assays, such as total bilirubin and direct bilirubin, also have significant missing values. The mean age was around 44 years, and the median age was 45 years, indicating a balanced age distribution of patients in this dataset.

The heatmap shows the relationship between each variable and the result (Figure 1). It is worth proposing that both total and direct bilirubin had a relationship of -0.23 with the result, thus it can be concluded that high bilirubin levels indicate possible underlying liver disease. Meanwhile, the relationship between Sgot aspartate aminotransferase and Sgpt alamine

aminotransferase and the result were both around -0.20, which also suggests that elevated levels of these two indicators are associated with abnormal liver function. While ALB albumin and total protein showed a weak positive correlation, indicating that the lower the level of protein and albumin, the more likely to acquire liver disease. Meanwhile, total bilirubin and direct bilirubin had a strong positive correlation (0.89), suggesting that one of them increases with the other, while Sgot aspartate aminotransferase and Sgpt alamine aminotransferase had the same relationship.

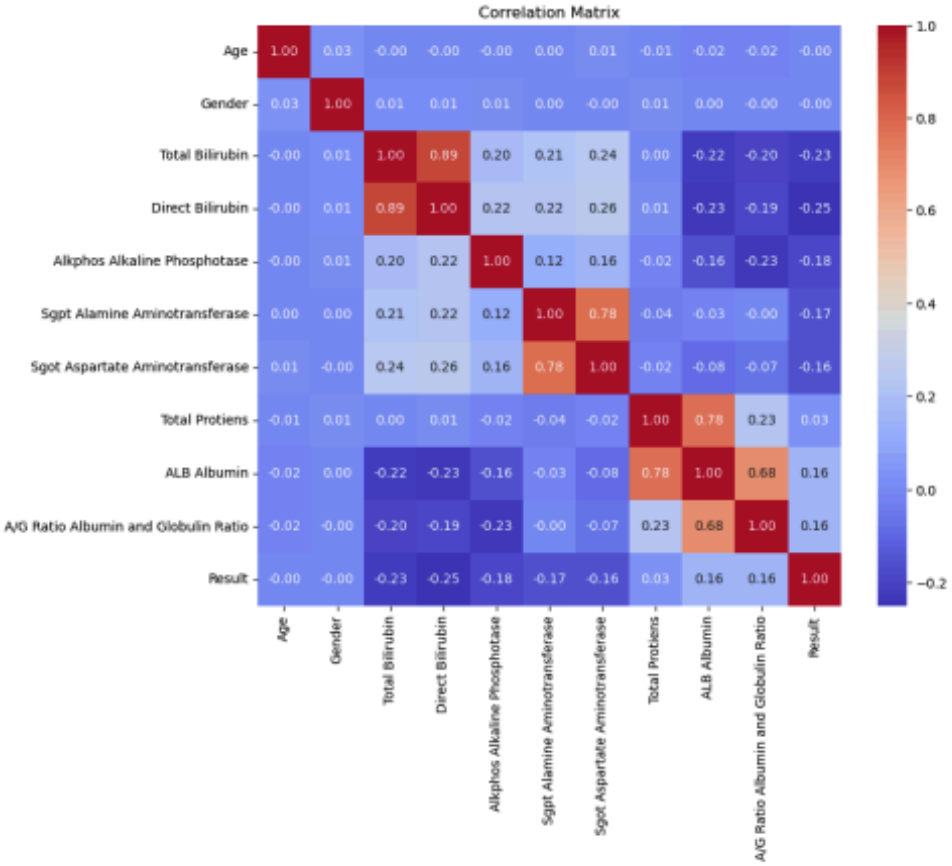

Figure 1.

The distribution of age (Figure 2) shows that the patients were mostly concentrated in the age group of 30-60 years, with the highest number of patients around 40-50 years. This indicates that the highest incidence of liver disease is between 30-60 years of age, while young people are less prone to liver disease due to the absence of bad habits, etc. Figure 2 also shows that the gender distribution is particularly uneven, with many more male than female patients, indicating that men are more likely to develop liver disease. This is because men are more likely to have a higher incidence of alcohol abuse and thus a higher chance of developing the disease (Chinnappan 2020).

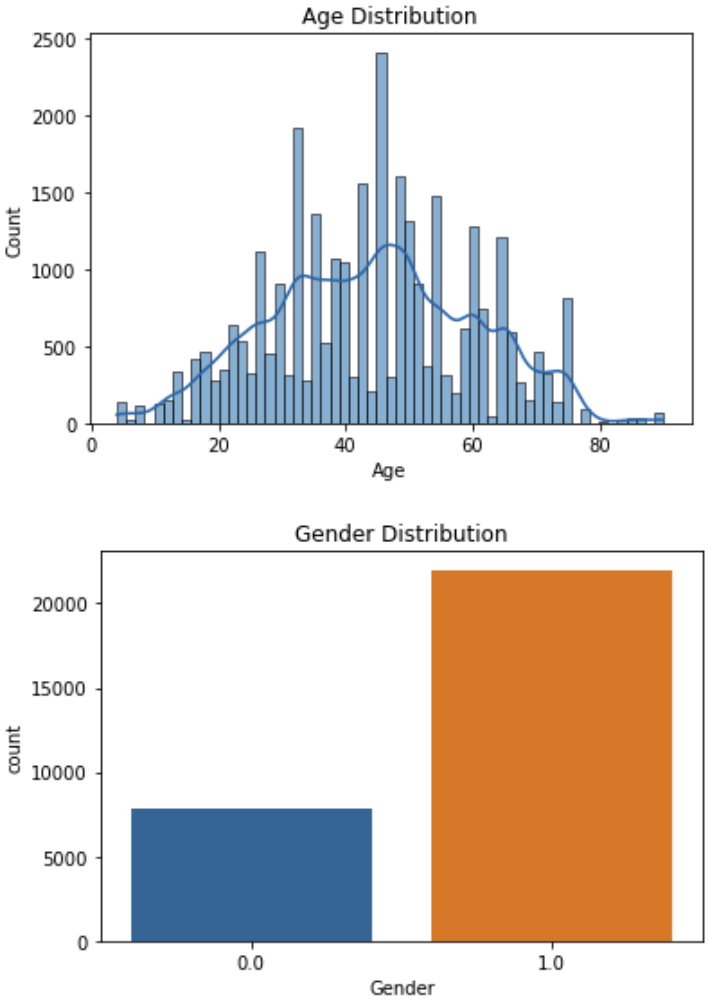

Figure 2.

Figure 3 demonstrates that total bilirubin is right-skewed with most values being clustered around zero. Some patients have extremely high bilirubin levels, indicating that it may be the result of a rare and severe liver disease, suggesting a possible outlier. Also, Figure 3 shows the total bilirubin and the resultant boxplot. it can be seen that the average total bilirubin levels of patients suffering from liver disease are all higher than those of healthy humans. However, it can also be seen that some of their values overlap, proving that total bilirubin is not the only cause of liver disease.

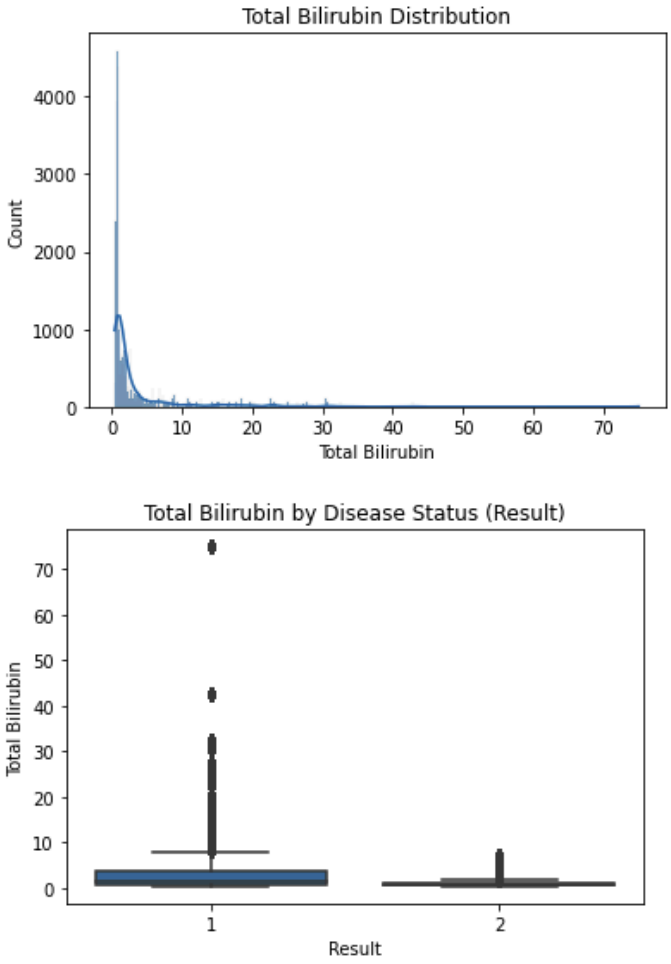

Figure 3.

As can be seen from the boxplot above in Figure 4, the levels of Sgpt are usually higher in patients with liver disease than in healthy humans. It can also be seen that there are many extreme values, even as high as 2000 U/L, indicating that there are certain cases that are very severe and may be advanced. The low and stable levels of Sgpt in healthy people can also show that this is an indicator that can be used to assess liver disease. The distribution of alkaline phosphatase levels is similar in men (1) and women (0), with a wide range of values. However, there were a few extreme outliers, some with values over 2,000 U/L, which may suggest liver or bone disease. The median values for both sexes remained close, indicating that gender did not have a significant effect on the enzyme levels. From the boxplot of alkaline phosphatase and gender, it can be seen that gender and alkaline phosphatase have little influence on each other (Figure 4). The distribution of levels was similar for males and females with some outliers, but the median values for both genders were still very close.

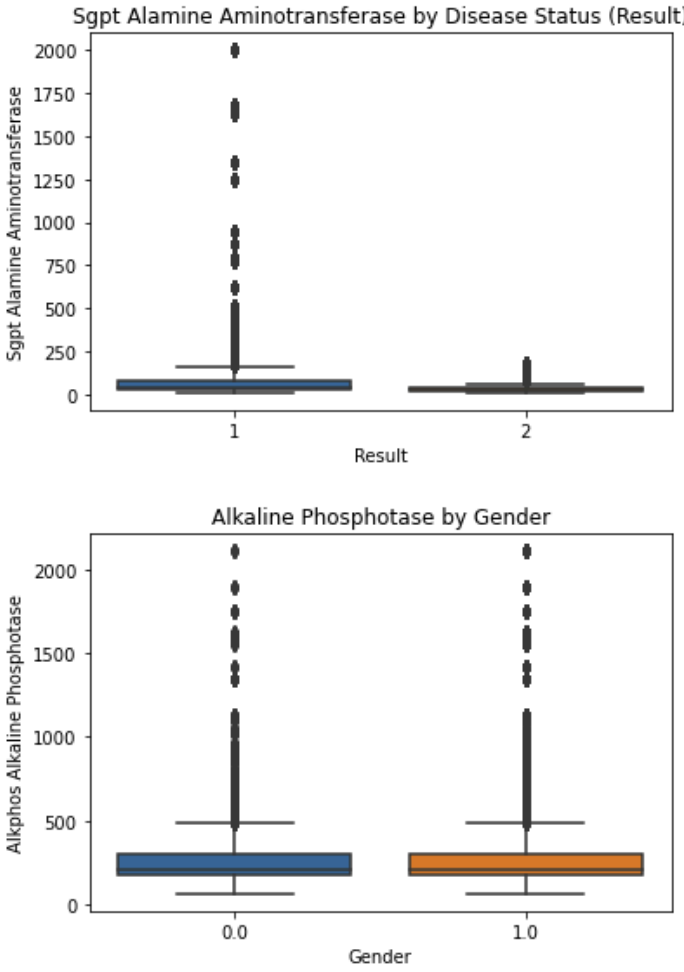

Figure 4.

The correlation between bilirubin and liver enzymes is depicted in paired plots. This correlation suggests that liver dysfunction might affect multiple biochemical markers simultaneously (Figure 5). Patients with increased total bilirubin, alkaline phosphatase, and Sgpt aminotransferase are suggested to have liver disease. The clustering shows these features to be important predictors of liver disease. There was a clear distribution pattern whereby the key majority of patients with normal or low values of these biomarkers fell within the healthy category. However, since overlapping indicators had effects on one another, one feature alone would not suffice in categorizing. It would improve the predictive accuracy if multiple variables

were combined into a single model. In this case, also, certain features such as alkaline phosphatase and Sgpt aminotransferase exhibited a wide range and extreme outliers, thus implying that, amongst the patients, the severity of liver disease differed. Such extreme values might require either a log transformation or scaling, or both, to ameliorate their influence on model training.

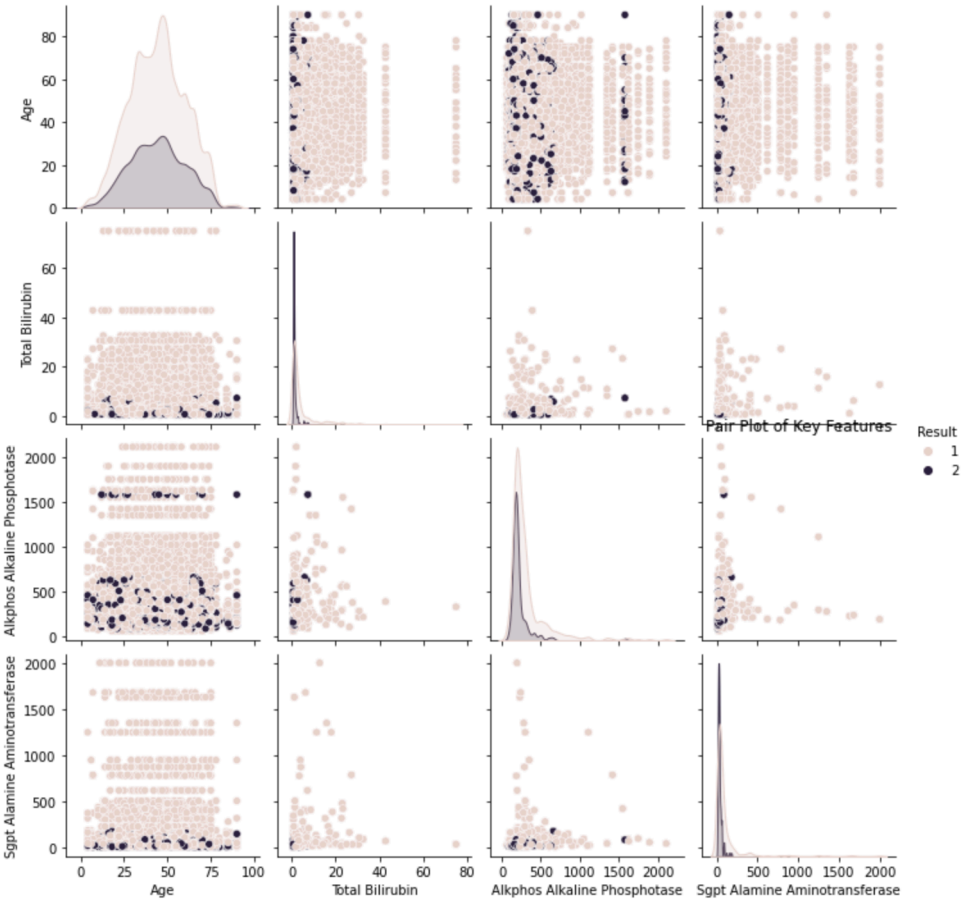

Figure 5.

5.0 Data Preparation/Feature Engineering

We check the missing values in the dataset and fill in the missing values by their category. The numerical variables are filled with the median and the category variable is filled with the mode. In addition, based on the analysis in EDA, we found that Total Bilirubin, Direct Bilirubin,

Sgpt Alamine Aminotransferase, and Sgot Aspartate Aminotransferase existed skewed. Therefore, we use log transformation to ensure their stability. To avoid zero value, we specifically use log(x+1).

Regarding the variables, we focused more on the ratio of some variables. so we created three raito: Bilirubin Ratio, SGOT/SGPT Ratio, and Protein Ratio. The bilirubin ratio was created by total bilirubin divided by direct bilirubin. The SGOT/SGPT ratio was created by Sgpt alanine aminotransferase divided by sgot aspartate aminotransferase. The protein ratio is created by total proteins divided by ALB albumin. Secondly, we labeled the age as four bins. The age from 0 to 18 is labeled as Child. The age from 18 to 40 is labeled as Young. The age from 40 to 60 is labeled as Middle age. The age from 60-100 is labeled as Old. After we encoded our category variables, we also removed the highest correlation variable, Direct Bilirubin, to avoid multicollinearity.

For our new features, we drew the bilirubin ratio histogram in Figure 6. Although we did the log transformation for total Bilirubin and direct bilirubin, the bilirubin ratio still shows skewed. It shows right-skewed, and most data points are clustered between 0 and 1. There are also multiple peaks, which present possible multimodality like different subpopulations. About the SGOT/SGPT Ratio, we did the boxplot by Disease Status in Figure 7. The boxplots for the two groups look very similar in terms of median, distribution, and overall shape. Visually, we see no significant difference in the SGOT/SGPT ratios between the two groups. The median SGOT/SGPT ratios of the two groups were close to 1. Ratios > 1 may indicate hepatic fibrosis, cirrhosis, or alcoholic liver disease, whereas ratios < 1 may indicate acute viral hepatitis(Agarwal, n.d..).

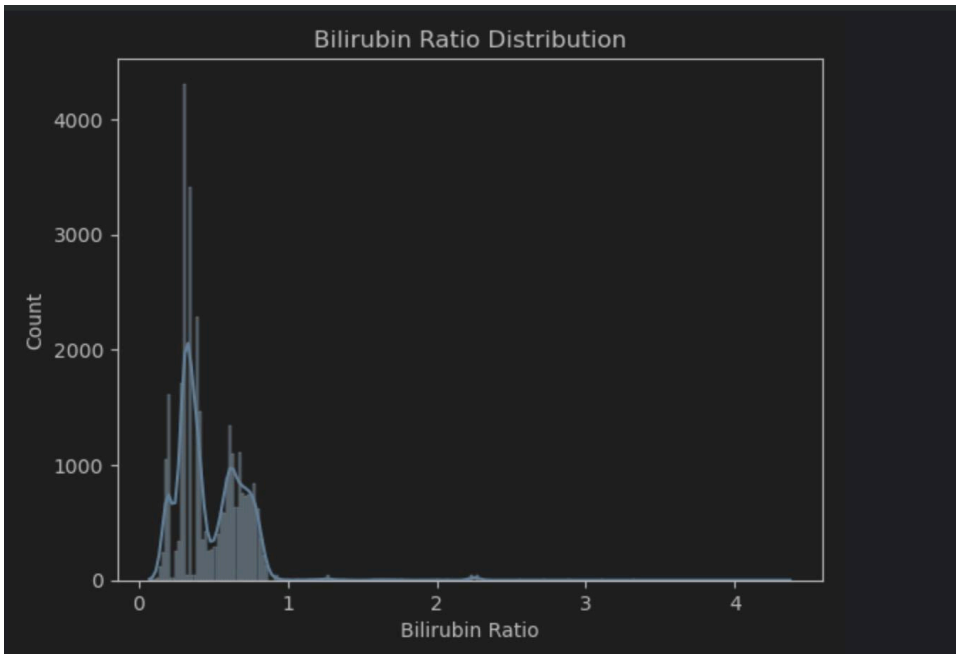

Figure 6.

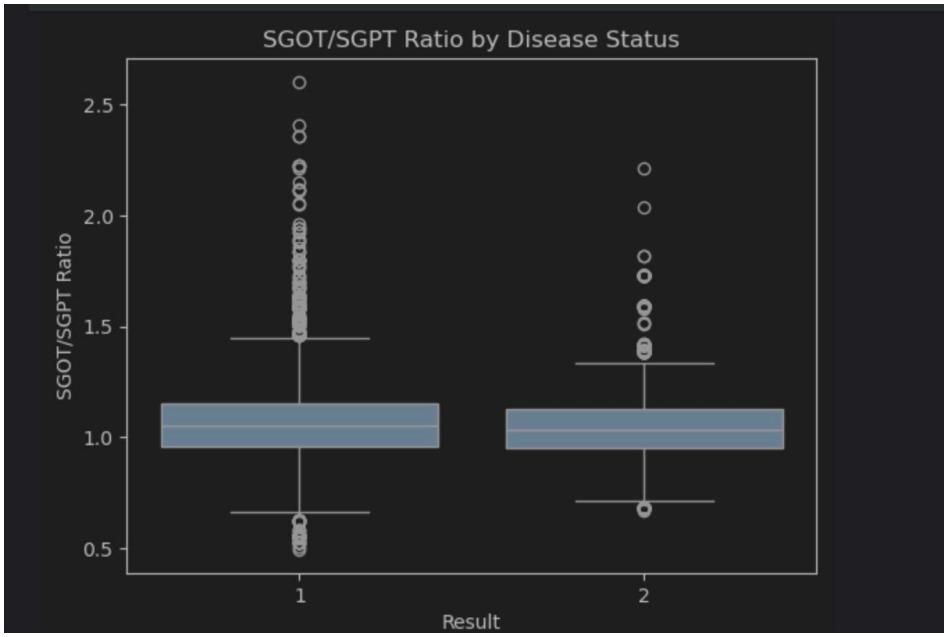

Figure 7.

We first chose SGOT/SGPT Ratio, Bilirubin Ratio, Sgpt Alamine Aminotransferase, and Sgot Aspartate Aminotransferase to PCA. In Figure 8, 2D PCA was plotted for PC1 and PC2, and we observed a significant overlap between the two disease states, in addition to several

outliers on the right side such as PC1 > 6 that may indicate extreme abnormalities in liver function. In Figure 9. 3D PCA can demonstrate PC3, and while overlap remains, some points along PC3 appear more separated, suggesting that it captures additional disease-related differences not visible in 2D. This diffusion may be intervened by highly variable features.

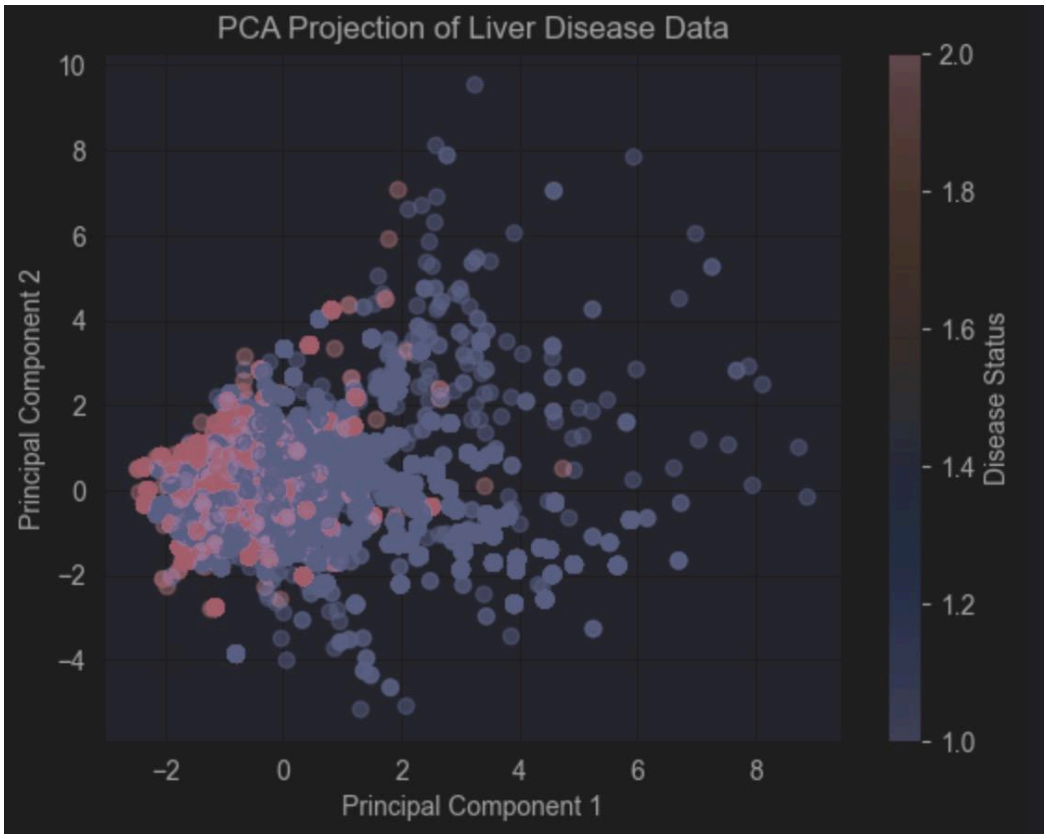

Figure 8.

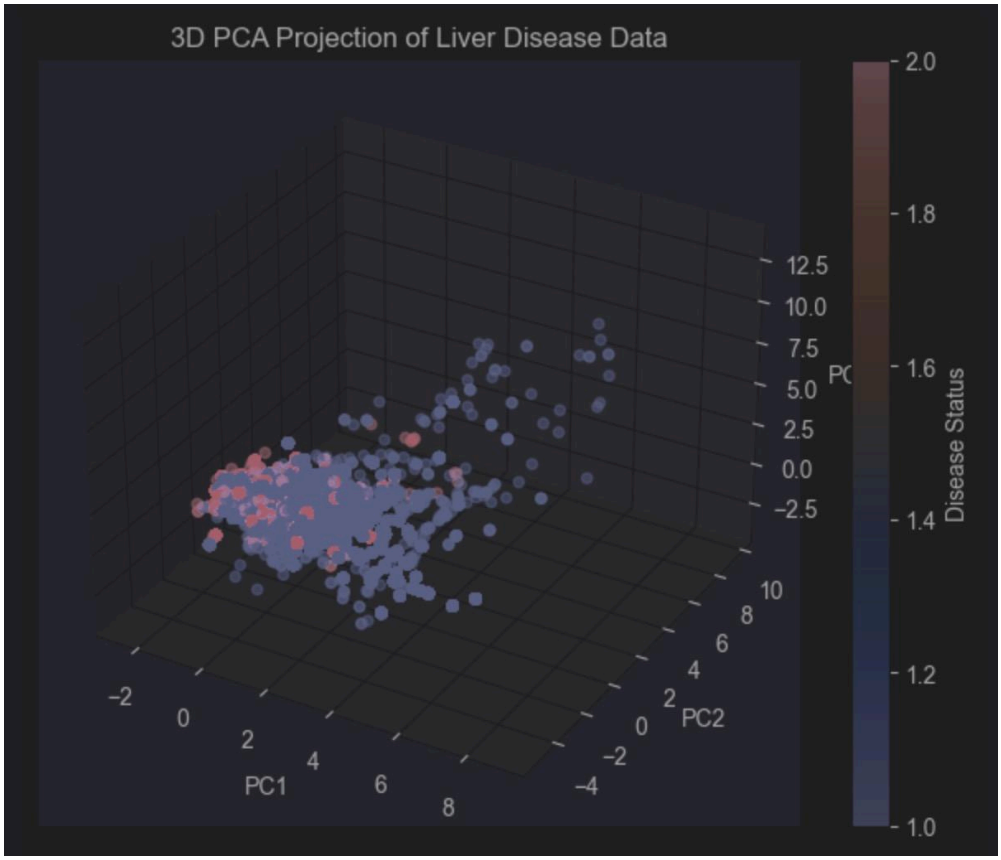

Figure 9.

We also chose features by using random forests as there are some high-dimension relationships (Darst, Malecki, and Engelman 2018). Then we got the ten variables: total bilirubin, alkphos alkaline phosphotase, sgpt alamine aminotransferase, sgot aspartate aminotransferase, total protiens, alb albumin, a/g ratio albumin and globulin ratio, bilirubin ratio, sgot/sgpt ratio, protien ratio. From Figure 10, we can find that Compared to the previous PCA, the optimized PCA showed clearer clustering although there was still overlap between the two disease states. In addition, points indicating health are more densely clustered around the center. Disease cases will show greater dispersion, especially to the right when PC1 > 0. We hypothesize that this indicates that disease status introduces variation in some of the major components.

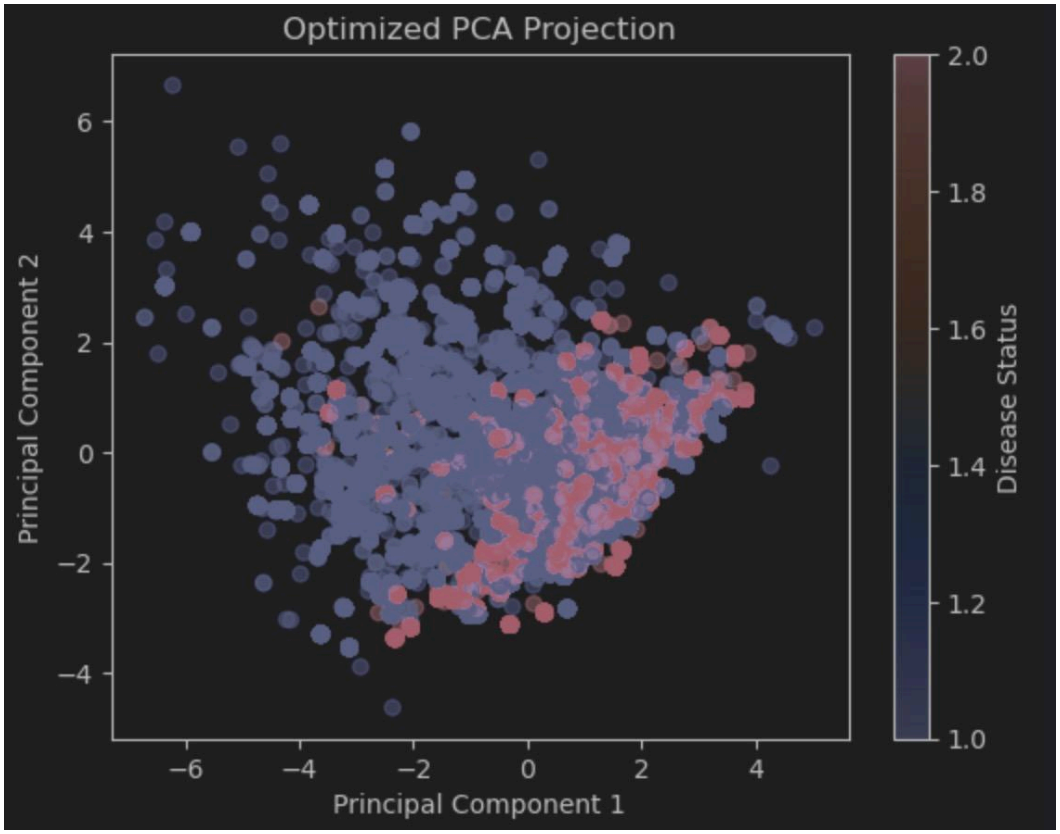

Figure 10.

6.0 Methodology and various tools used in the process

6.1 Logistic Regression

I decided to use Jupyter Notebook and a CPU for my model. I chose an interactive, modular environment due to the ease of debugging, hyperparameter tuning, and stepwise analysis. This interactive testing and visualization in the cells promote reproducibility and teamwork among other partners. Since logistic regression requires almost no computation, and CPU-based Scikit-learn solvers were implemented to optimize the code for CPU execution, there was no need for GPU acceleration. Given that this model does not use deep learning or heavy computations, utilizing a GPU will not bring about any significant improvements regarding speed. Thus, using a CPU makes training and evaluation much more cost-effective.

In this case, I used a logistic regression model to train the data so that observations were classified based on various biochemical markers. To make the model even better, I also did hyperparameter tuning with the aid of a GridSearchCV. With this, I would be able to - via multiple iterations - search among a large number of hyperparameter value combinations to search for the best hyperparameter combination that would yield maximum performance on the model. I optimally focused on tuning such parameters as type of penalty (l1 or l2), inverse of regularization strength (C), solver methods (liblinear and saga), and ways of class weighting (None or balanced). The C values controlled how strong the regularization would be; higher yields a better overall fit for the model but increases the chance of overfitting. I also used l1 and l2 regularization penalties, where l1 promotes sparsity and l2 reduces model complexity. Cross-validation via GridSearchCV assured it chose the best hybrid combination based on accuracy, thus enhancing the trustworthiness or generalizability of the model.

It included five-fold cross-validation, which strengthens the assurance where all the data were divided into five, and the training and testing of the model was repeated five times on various parts. This technique helps me reduce overfitting risks and gives an estimate of the model's performance that I can reflect on with greater assurance. I calculated the average cross-validation accuracy to see the degree of generalization on unseen data. Adding to a decision on overfitting, I computed training accuracy: if training accuracy is really different from cross-validation accuracy, it implies that my model can do well on training data but is not able to generalize. To be sure the model is effective and reasonably reliable before classification, I applied cross-validation.

	Training Accuracy	Testing Accuracy	AUC -ROC	Mean CV Accuracy
Log Regression	73.37%	73.82%	0.77	73.08%

Table 1.

Logistic regression achieved an accuracy of 73.37% in the training set and 73.82% on the testing set, an indication that it was not overfitting and is able to generalize quite well. This is further supported by the model mean of cross-validation (CV) accuracy, which was 73.08%, thus confirming that the CV results were stable across different data splits. Besides, the AUC-ROC score of 0.77 states that the model has a good discriminating power between the classes and thus can be taken as a trustworthy classifier on this dataset. In the end, performance indicators suggest that the model is balanced, with stable outputs across training, testing, and validation.

The automated logistic regression model of risk for liver disease shall include data intake, preprocessing, and inference with automated monitoring. Monitoring shall be performed by the frameworks of performance, drift detection, and pipeline retraining (Mucci and Stryker 2024). This shall ensure scalability, efficiency, and continuous improvement and will, therefore, make the model an even more trustworthy instrument in the field of health-related early diagnosis of diseases.

6.2 Random Forest

I used Google Colab as the primary development environment for this project, leveraging its GPU support to accelerate model training and evaluation. Google Colab provides easy access to pre-installed libraries and scalable hardware resources. Key libraries I used include scikit-learn for machine learning models, numpy for pandas data processing, and matplotlib for seaborn visualization.

My model uses a random forest classifier to classify the liver disease. Random forests can construct multiple decision trees and aggregate their predictions to improve accuracy and reduce overfitting. To analyze the effect of dimensionality reduction on model performance, I trained the random forest model under two conditions: one using the full feature set and the other applying principal component analysis. PCA helps to eliminate redundant information, which potentially improves computational efficiency and generalization. Thus I can compare the differences between the two models.

We uniformly selected 10 key variables to assess model performance. Accuracy was used to assess the overall correctness of the predictions, while the confusion matrix provided a detailed categorization of true positives, false positives, true negatives, and false negatives. Precision, recall, and F1 scores were also analyzed to measure the balance between correctly identified and misclassified instances. To further evaluate the classification effect, I plotted ROC-AUC curves, which represent the model's ability to distinguish between categories, as well as precision-recall curves, which are particularly useful when dealing with category imbalance. In addition, I performed 5-fold cross-validation to ensure the robustness and generalizability of the model across different training and testing splits.

The results in Table 2 show that without PCA, the Random Forest model performs very well, with a training accuracy of 99.99% and a test accuracy of 99.69%. the AUC -ROC score is 1.00, indicating that the model can distinguish between categories perfectly. On the other hand, when PCA is applied, the model's test accuracy decreases slightly to 98.94%, and the AUC-ROC score decreases to 0.998, indicating a slight decrease in performance. In addition, cross-validation of the model without PCA showed consistent performance with an average accuracy of about 99.66%, confirming the stability of the model.

Random Forest	Training Accuracy	Testing Accuracy	AUC -ROC	Mean CV Accuracy
Without PCA	99.99%	99.69%	1.00	99.66%
With PCA	99.99%	98.94%	0.998	

Table 2.

The automated, scalable strategy, I'm considering is that trained models can be saved as serialized files like pkl format and embedded in applications that can make real-time predictions(Raschka and Mirjalili 2019). An automation pipeline in Colab can be set up to periodically retrain the model to ensure that it can adapt to new data. Additionally, continuous evaluation with new test data will help monitor any performance degradation over time so that timely updates and improvements can be made.

6.3 FNN

I employed Jupyter Notebook as the primary development environment, because of its interactive nature and seamless integration with essential machine learning libraries. Its user-friendly interface made it easy to experiment with different models and visualize data. The model was developed and trained using TensorFlow and Keras, with NumPy and Pandas used for data preprocessing. Scikit-learn was employed for feature scaling, class balancing, and evaluation metrics.

The Feedforward Neural Network (FNN) model was constructed with a layered architecture designed for binary classification. The model consists of an input layer corresponding to the number of features, following by two hidden layers with 256 and 128 units respectively, using ReLU activation functions. Batch normalization was used to stabilize training,

and dropout layers were added to prevent overfitting. The final output layer utilized a sigmoid activation function for binary classification.

Hyperparameter tuning was focused on optimizing dropout rates, learning rate adjustments, batch size, and the number of neurons in hidden layers. Through iterative experiments, I tested different dropout values to prevent overfitting while maintaining model stability, ultimately selecting 0.1 dropout rates for the hidden layers. Learning rate tuning was done using a learning rate scheduler, adjusting step sizes dynamically based on validation loss trends. The model was trained for 50 epochs, but early stopping was enabled with a patience level of 5 epochs to prevent overfitting and unnecessary computations. To address class imbalance, I computed class weights dynamically, giving higher weight to the minority class. This ensured that the model did not bias predictions toward the majority class, improving recall and overall fairness across both classes.

Hidden Units	64	128	**256**
Dropout Rate	**0.1**	0.3	0.5
Learning Rate	0.001	**0.0005**	0.0001

Table 3.

Performance evaluation was conducted using accuracy, precision, recall, F1-score, and AUC-ROC metrics. The final training accuracy reached 98.60%, with a test accuracy of 98.79%, demonstrating strong predictive capabilities. The ROC-AUC score of 0.996 indicated high discriminative power, while the precision-recall curves provided additional insights into model performance under class-imbalanced conditions.

	Training Accuracy	Testing Accuracy	AUC -ROC	Mean CV Accuracy
FNN	98.60%	98.79%	0.996	98.72%

Table 4.

6.4 KNN

In this experiment, I chose to use K-Nearest Neighbors (KNN) as a classification model to evaluate its performance on the current dataset.KNN is an instance-based learning algorithm that performs classification by calculating the distance between samples. In the absence of an explicit training process, KNN relies on stored training data and performs nearest-neighbor computation in the inference phase, so its computational overhead is mainly concentrated in the prediction phase.

Since the computation of KNN mainly relies on distance computation and does not involve complex gradient descent or backpropagation, this experiment uses Jupyter Notebook and CPU for training and inference to avoid additional computational resource consumption. In addition, the interactive development environment provided by Jupyter Notebook is helpful for hyperparameter tuning, visualization and analysis, as well as explanatory studies of the model.

KNNs require choosing the appropriate number of neighbors (K-value) as well as the distance metric to ensure that the model is neither overfitted nor overly lax. In this experiment, I used GridSearchCV for hyperparameter search, focusing on optimizing the following parameters:

Number of Neighbors (n_neighbors): multiple K-values were tried, ranging from 5 to 25, to find the optimal K-value.

Weights: Compare uniform (uniform weights) and distance (distance-based weighting), and observe their effects on the classification performance.

Distance metric: Compare euclidean and manhattan to choose the best metric.

Ultimately, the best combination of hyperparameters chosen for the model is:

n_neighbors=17;weights=distance;metric=manhattan

This configuration achieves an optimal accuracy of 99.65% in cross-validation, indicating that the KNN is well adapted to the task at hand.

In order to evaluate the generalization ability of the model, I used cross-validation learning curves to show the performance of the model under different training set sizes. The experimental results show that as the proportion of the training set increases, the test accuracy gradually stabilizes, indicating that the KNN model converges well on the current data.

The final test accuracy reaches 100%, indicating that the model has excellent classification results in this task. The area of the AUC-ROC curve is 1.00, which means that the model's ability of distinguishing between the categories reaches a perfect level.

	Training Accuracy	Testing Accuracy	AUC -ROC	Mean CV Accuracy
FNN	99.98%	99.56%	0.998	99.02%

Table 5.

7.0 Findings and Conclusions

7.1 Findings

7.1.1 Logistic Regression

The testing confusion matrix provides the performance assessment of logistic regression over the two classes (Figure 11). It classified 514 observations into class 0 and 4,018 into class 1, while it misclassified 1,241 of class 0 as class 1 (false positives) and misclassified 366 of class 1 as class 0 (false negatives). The relatively high number of false positives suggests that the model was biased toward predicting class 1 with a higher frequency, possibly due to class imbalance or simply a more favorable threshold. The performance was fairly aligned with the AUC-ROC score of 0.77, where the model performs well, although it seems further adjustment would be needed, from tuning the decision threshold to looking at different regularization techniques to reduce false positives and improve precision.

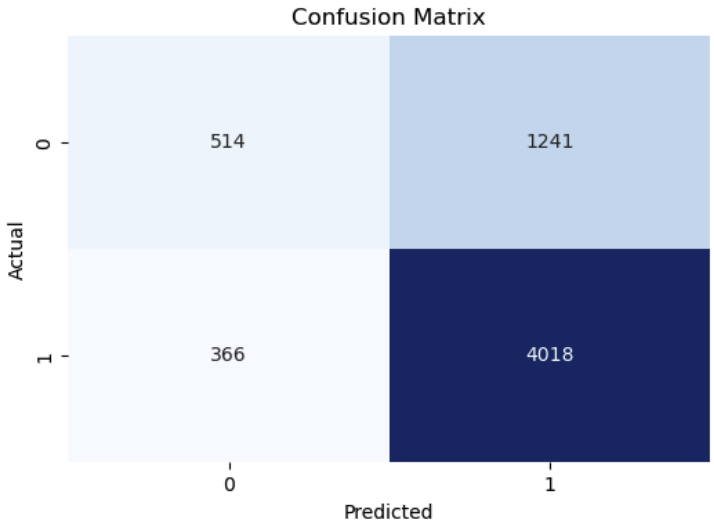

Figure 11.

The PR curve shows the trade-off of precision and recall for our logistic regression model for the testing set (Figure 12). For the majority of the recall values, however, the curve is still high, which means that precision is mostly good, and that is the most critical aspect of reducing false positives. The average precision score of 0.90 represents the trade-off between capturing positive cases and predictions with false alarms at a low rate. However, with increasing recall, precision becomes less and less, indicating that with more actual positives captured, more false positives are also misclassified in the model. This indicates that the high AP score represents that the model works very well in this situation, where correctly identifying positive cases is more important than capturing all possible instances.

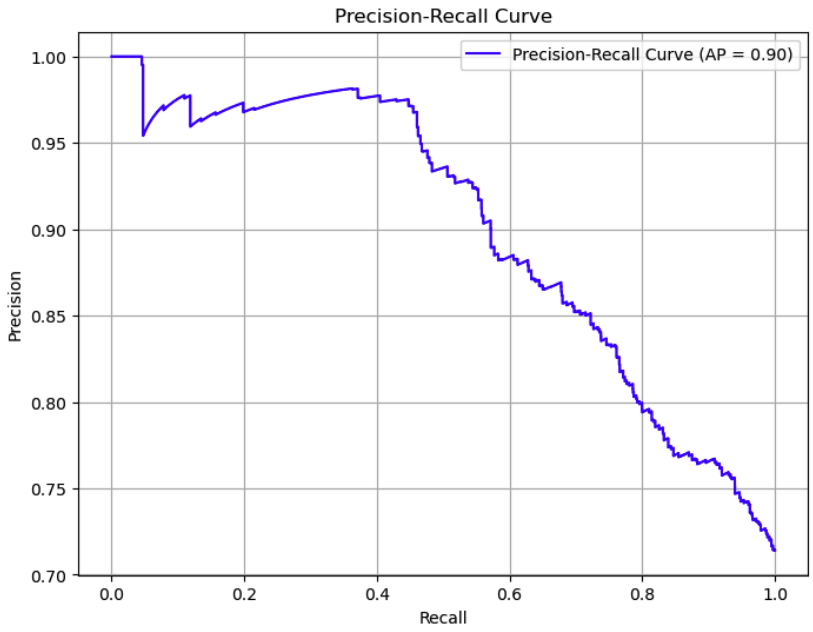

Figure 12.

7.1.2 Random Forest

To critically assess model performance, we evaluated the confusion matrix because the confusion matrix helps to identify potential biases in the model, especially when distinguishing between patients with and without liver disease. From Figure 13 , we observed that the false

negative rate was extremely low since there were 6 cases out of 1755, which means that very few liver disease cases were incorrectly categorized as disease-free. This is particularly important in a clinical setting where false negatives can lead to misdiagnosis. In addition, the false positive rate was also extremely low 13 out of 4384 cases, thus reducing the likelihood of unnecessary medical interventions.

After using PCA in Figure 14, the number of false negatives was 21 vs. 6 cases and false positives was 44 vs. 13 cases with a slight increase. Although PCA helped to reduce model complexity, it resulted in a slight decrease in predictive power, especially when distinguishing liver disease from non-disease cases.

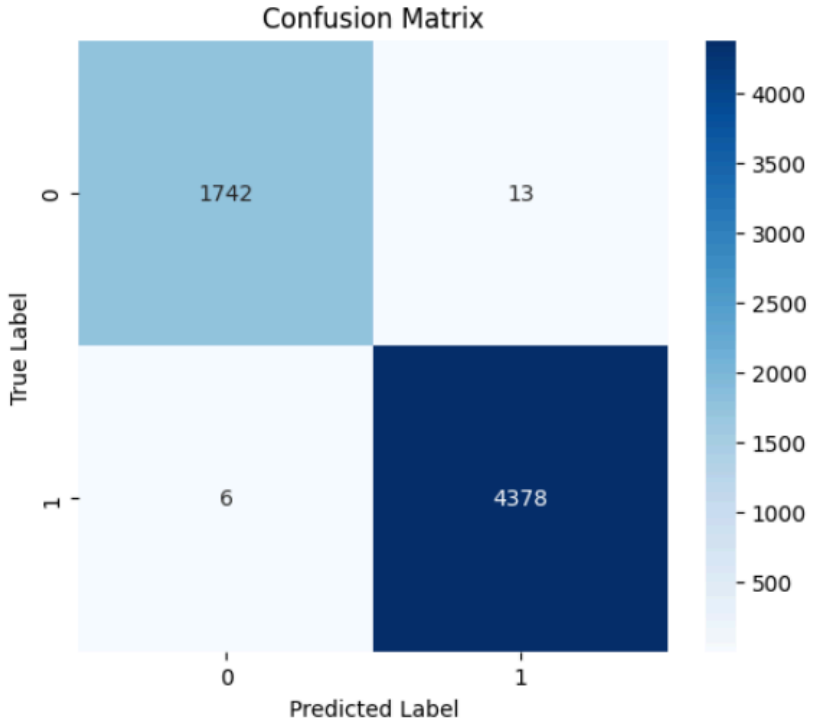

Figure 13. Confusion Plot of Random Forest without PCA

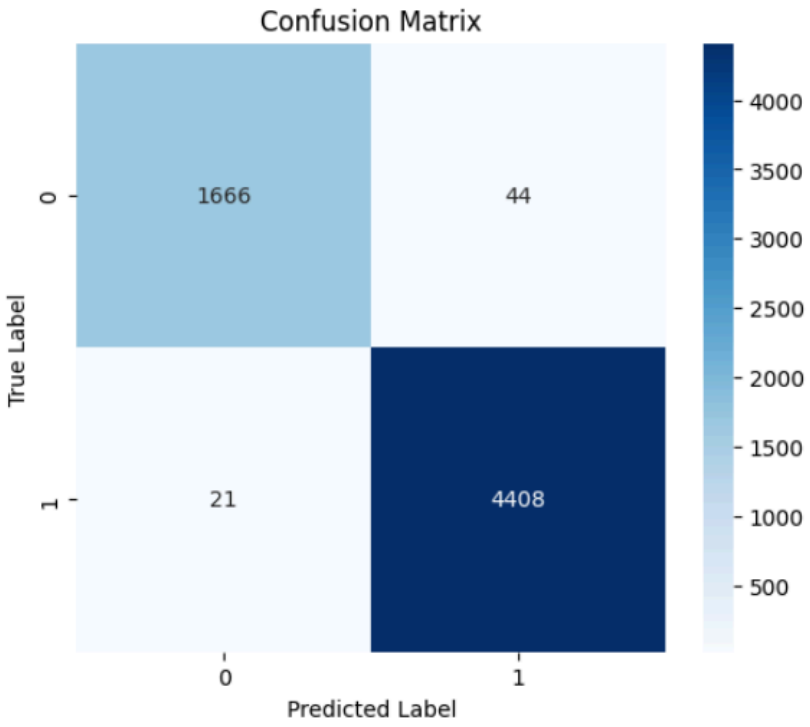

Figure 14. Confusion Plot of Random Forest with PCA

In conclusion, for our liver disease data, the random forest model exhibits excellent classification performance and is well-suited for deployment in real-world applications. While PCA slightly reduces computational complexity, it also causes a slight decrease in classification accuracy. Extensive evaluation with multiple metrics, cross-validation, and visualization confirmed that the model is highly effective in distinguishing liver disease categories.

7.1.3 FNN

The FNN model demonstrated strong predictive performance in classifying liver disease cases, as indicated by the confusion matrix and evaluation metrics. The model correctly classified 4,344 cases of liver disease and 1,721 cases of no disease, with only 34 false positives and 40 false negatives (Figure 15). These results reflect a high classification accuracy, confirming the model's reliability in distinguishing between healthy individuals and those with liver disease.

The confusion matrix visualization provides additional insights into the model's predictive behavior. The low false positive rate (34 cases) suggests that the model rarely misclassified healthy individuals as having liver disease, which is crucial for avoiding unnecessary medical treatments. Similarly, the low false negative rate (40 cases) indicates that only a small percentage of individuals with liver disease were misclassified as healthy, reducing the risk of undetected cases. These results suggest that the model achieves a good trade-off between sensitivity and specificity, making it a valuable tool for clinical applications.

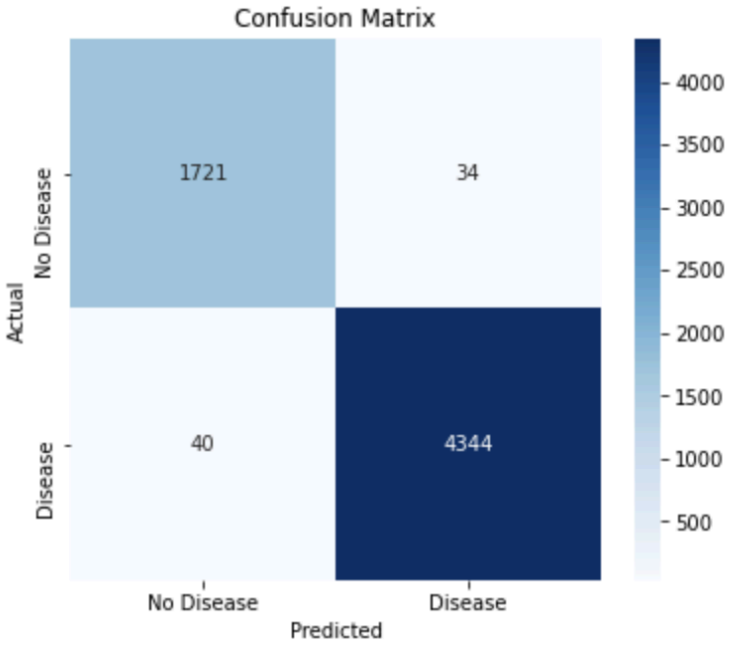

Figure 15.

7.1.4 KNN

The confusion matrix for the K-Nearest Neighbors (KNN) model provides a comprehensive assessment of its classification performance (Figure 16). The model correctly classified 1,747 cases as class 0 and 4,363 cases as class 1. It misclassified 8 instances of class 0 as class 1 (false positives) and 21 instances of class 1 as class 0 (false negatives). These results

indicate that the model achieved an exceptionally high classification accuracy of 100% on the test set, with minimal misclassification errors.

The extremely low false positive and false negative rates suggest that the KNN model is highly reliable in distinguishing between the two classes. This is further corroborated by the AUC-ROC score of 1.00, which demonstrates that the model has perfect discrimination ability between positive and negative cases. The learning curve (Figure 17) illustrates that both training and test accuracy stabilize as the training data size increases, indicating that the model generalizes well. Additionally, hyperparameter tuning through cross-validation (Figure 18) confirmed that the optimal parameters—Manhattan distance metric, 17 neighbors, and distance-based weighting—were effective in maximizing performance.

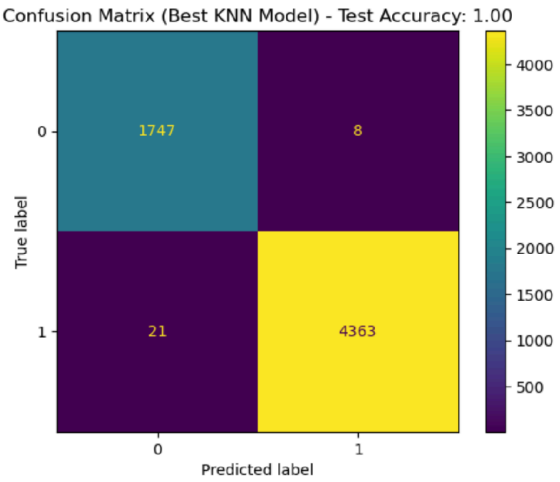

Figure 16. Confusion Matrix for KNN Model

The ROC curve (Figure 17) provides further insight into the model's classification effectiveness, showing a perfect separation between the two classes. The model's true positive rate reaches 1.0 at all points, further confirming its superior performance in this dataset. Given its

perfect AUC-ROC score, the KNN model outperforms logistic regression and random forest, demonstrating its ability to handle the classification task with remarkable precision.

However, despite its outstanding accuracy, KNN has computational drawbacks, particularly in larger datasets. Since it is a lazy learning algorithm, it requires storing and computing distances for all training points at prediction time, which can be inefficient for real-time applications. While it performs exceptionally well on this dataset, its scalability might be a concern for deployment in environments with large-scale data.

In summary, the KNN model achieved near-perfect classification accuracy with minimal misclassifications, making it a highly effective choice for this task. However, future implementations may benefit from optimized nearest neighbor search techniques, such as KD-Tree or Ball-Tree, to enhance computational efficiency without compromising classification performance.

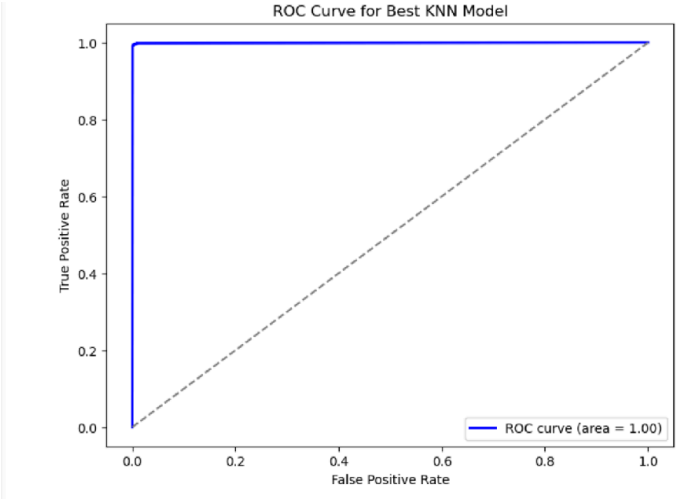

Figure 17. ROC Curve for KNN Model

7.2 Conclusions

Hypothesis testing and impact analysis in this study confirmed that logistic regression modeling was not applicable due to the nonlinear relationship between the characteristics and the target variables. The low recall and F1 score of logistic regression made it clear that it struggled to capture the complexity of liver disease prediction. Given the limitations of linear models, we explored more advanced machine learning techniques, including random forests, K nearest neighbors, and feedforward neural networks.

Through rigorous model evaluation, including confusion matrix analysis, cross-validation, and performance metrics, Random Forest without PCA emerged as the best-performing model. The model demonstrated excellent precision, recall, and F1 scores in both categories, ensuring the lowest false-negative rate, which is critical in the clinical setting where early detection of liver disease can significantly impact patient outcomes and treatment costs.

From an impact perspective, selecting the most accurate model ensures efficient and reliable prediction of liver disease, which is critical for medical decision-making. High recall means that the model can correctly identify most patients with liver disease, thus minimizing missed diagnoses. The confusion matrix results further validate the robustness of the model as the false-negative rate is significantly lower compared to other models, confirming its reliability in practical applications.

The high accuracy and reliability of models make it a valuable tool for healthcare businesses and medical services. This model can help doctors and hospitals detect liver disease earlier and more accurately, reducing misdiagnosis and improving patient care. By integrating

this technology into hospital systems, telemedicine platforms, or diagnostic centers, healthcare providers can offer faster and data-driven diagnoses, making medical services more efficient.

From a business perspective, this model can be developed into a digital health tool for insurance companies and pharmaceutical firms. Insurance providers can use it to better assess health risks, allowing them to offer personalized health plans and fairer policy pricing. Pharmaceutical companies can benefit from identifying high-risk patients for clinical trials and targeted treatments, leading to better drug research and development.

Additionally, companies focused on workplace health and employee wellness could use this model to track employee health risks and promote preventive healthcare programs. Early detection of health issues can help businesses reduce health-related costs and support employee well-being.

8.0 Lessons Learned and Recommendations

8.1 Limitation

Although our model performs well in classifying liver disease, some limitations must be recognized:

8.1.1 Data imbalance

There are significantly more non-disease cases than disease cases in the dataset, which may lead to a bias towards predicting non-disease cases more frequently. This imbalance reduces the sensitivity of the model in detecting liver disease and may lead to higher false-negative rates in real-world applications.

8.1.2 Feature Availability

The dataset lacks important clinical and lifestyle-related factors such as alcohol consumption, medication history, genetic predisposition, and dietary habits, which are known to

influence the progression of liver disease. The absence of these key features may limit the ability of the model to make fully informed predictions.

8.1.3 Overfitting Risk

While techniques such as cross-validation and hyper-parameter tuning are used to improve generalization, complex models such as FNN and Random Forests without PCA may still be at risk of overfitting, especially on small datasets. Regularization techniques, early stopping, and culling layers are necessary to reduce this risk in deep learning models.

8.1.4 Interpretability

Some models, especially FNNs and ensemble-based methods (Random Forest, XGBoost, etc.), lack transparency in their decision-making process. This can be a limitation in medical applications where interpretability is crucial for trust and clinical adoption. Techniques such as SHAP (Shapley Additive Explanations) or LIME (Local Interpretable Model-Agnostic Explanations) should be explored to improve interpretability.

8.1.5 Computational Limitations

The KNN model is computationally expensive, requiring a large amount of memory storage and a lot of processing time to compute distances for large datasets. This makes it less practical for real-time prediction.

Deep learning models such as FNN require GPU acceleration for efficient training and hyperparameter tuning. Without sufficient computational resources, large-scale datasets can lead to long training times and resource constraints.

8.2 Lessons Learned

Through this study, we gained some important insights into the classification of liver disease and the challenges faced by machine learning models in medical applications:

8.2.1 The Importance of Data Balancing in Medical Prediction

The robust performance of our model highlights how unbalanced datasets can skew classification results, leading to biased predictions. While models such as RF and KNN achieve high accuracy rates, their reliability may be compromised in real-world clinical settings where false-negative results can have serious consequences. Future work should emphasize data resampling techniques and cost-sensitive learning to improve disease detection.

8.2.2 Feature selection and data integrity are important

The lack of certain key clinical and lifestyle features, such as alcohol consumption, genetic predisposition, and medication history, limits the potential performance of the model. This emphasizes the need for feature engineering and dataset extension to improve prediction quality. The integration of external medical databases and real-world clinical data should be prioritized in future studies.

8.2.3 Model complexity must be balanced with interpretability

While deep learning models such as FNNs provide powerful classification results, their black-box nature makes interpretability challenging. In healthcare applications, interpretable tools (e.g., SHAP, LIME) should be used to improve transparency and ensure that predictions can be trusted by healthcare professionals. Traditional models such as logistic regression, although less accurate, are more interpretable and still have value in medical decision-making.

8.2.4 Computational efficiency is crucial for scalability

The KNN model achieves perfect accuracy but is computationally expensive due to its reliance on distance calculations. This suggests that in practical applications, the choice of algorithms should not only consider accuracy but also scalability and efficiency. Tree-based

models (Random Forest, XGBoost) and dimensionality reduction techniques (PCA) provide alternatives that balance performance and computational feasibility.

8.2.5 Cross-validation and regularization are crucial for generalization

The risk of overfitting complex models highlights the need for appropriate regularization techniques, culling layers in deep networks, and extensive cross-validation. Ensuring that a model generalizes well to new, unseen data is more valuable than achieving high accuracy on a specific dataset.

8.3 Third-Party Datasets That Can Add Value to the Existing Analysis

To improve liver disease prediction and analysis incorporating external datasets could provide a more comprehensive view of the disease's progression and potential risk factors. Some recommended third-party datasets include:

8.3.1 ILPD (Indian Liver Patient Dataset) - UCI Machine Learning Repository

A widely used dataset contains clinical parameters such as albumin, bilirubin, and enzyme levels. It can be used for cross-dataset validation, ensuring our model generalizes well across different populations.(Dua, Dheeru, and Casey 2011)

8.3.2 BioBank UK Liver Disease Data

Contains genetic and lifestyle factors affecting liver disease. It can help improve feature availability, addressing gaps in our current dataset.(Sudlow, Cathie and Jill, 2015)

8.3.3 NHANES (National Health and Nutrition Examination Survey) Liver Function Dataset

Includes alcohol consumption, diet, lifestyle choices, and medication history, which are crucial in understanding liver disease risks. It can help in addressing missing feature availability and improving real-world applicability.(CDC, 2023)

8.3.4 EHR (Electronic Health Records) Liver Disease Cohorts

Many medical research institutions have liver disease cohorts derived from hospital EHR systems. It can provide a real-world clinical validation for our predictive models.(Johnson, Alistair, and Tom, 2016)

References

Agarwal, Monika. n.d. “Knowing SGPT & SGOT Tests: Indications, Normal Ranges, & Procedures.” Dr. B. Lal. https://blallab.com/blogs/sgpt-and-sgot-tests-indications-normal-ranges-and-procedures.

Amin, Ruhul, Rubia Yasmin, Sabba Ruhi, Md Habibur Rahman, and Md Shamim Reza. 2023. “Prediction of Chronic Liver Disease Patients Using Integrated Projection Based Statistical Feature Extraction with Machine Learning Algorithms.” Informatics in Medicine Unlocked 36: 101155. https://doi.org/10.1016/j.imu.2022.101155.

BioBank UK Liver Disease Data. Sudlow, Cathie, Jill Gallacher, Naomi Allen, Vince Beral, Paul Burton, Rory Collins, John Danesh, et al. 2015. "UK Biobank: An Open Access Resource for Identifying the Causes of a Wide Range of Complex Diseases of Middle and Old Age." PLOS Medicine 12 (3): e1001779. https://doi.org/10.1371/journal.pmed.1001779.

Centers for Disease Control and Prevention (CDC). 2020. National Health and Nutrition Examination Survey (NHANES): Liver Function Tests Data. U.S. Department of Health and Human Services. https://wwwn.cdc.gov/nchs/nhanes.

Chinnappan, Shivani. 2020. “Exploring the Role of Sex and Gender Differences in Liver Health.” Society for Women’s Health Research, November 19. https://swhr.org/exploring-the-role-of-sex-and-gender-differences-in-liver-health/.

Darst, Burcu, Kristen Malecki, and Corinne Engelman. 2018. “Using Recursive Feature Elimination in Random Forest to Account for Correlated Variables in High Dimensional Data.” BMC Genetics 19 (9): 65. https://doi.org/10.1186/s12863-018-0633-8.

Dua, Dheeru, and Casey Graff. 2011. Indian Liver Patient Dataset (ILPD). UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/ILPD+(Indian+Liver+Patient+Dataset).

El–Serag, Hashem B., and K. Lenhard Rudolph. 2007. “Hepatocellular Carcinoma: Epidemiology and Molecular Carcinogenesis.” Gastroenterology 132 (7): 2557–76. https://doi.org/10.1053/j.gastro.2007.04.061.

Jaiswal, Ankur, Ritika Agrawal, and Surya Prakash. 2024. “Improved Liver Disease Prediction from Clinical Data through an Ensemble Learning Model.” BMC Medical Informatics and Decision Making 24 (1): 1–12. https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-024-02550.

Johnson, Alistair E. W., Tom J. Pollard, Lu Shen, Li-Wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, et al. 2016. "MIMIC-III, a Freely Accessible Critical Care Database." Scientific Data 3: 160035. https://doi.org/10.1038/sdata.2016.35.

Kang, Baeki E., Aron Park, Hyekyung Yang, Yunju Jo, Tae Gyu Oh, Seung Min Jeong, and Yosep Ji, et al. 2022. “Machine Learning-Derived Gut Microbiome Signature Predicts Fatty Liver Disease in the Presence of Insulin Resistance.” Scientific Reports 12 (1): 21842. https://doi.org/10.1038/s41598-022-26102-4.

Liu, Mei, Fei Zhang, and Ping Li. 2023. “Machine Learning Models for Predicting Non-Alcoholic Fatty Liver Disease.” World Journal of Hepatology 13 (10): 1500–1510. https://pmc.ncbi.nlm.nih.gov/articles/PMC8568572.

Liu, Yuan-Xing, Xi Liu, Chao Cen, Xin Li, Ji-Min Liu, Zhao-Yan Ming, Song-Feng Yu, et al. 2021. “Comparison and Development of Advanced Machine Learning Tools to Predict Nonalcoholic Fatty Liver Disease: An Extended Study.” Hepatobiliary & Pancreatic Diseases International 20 (5): 409–15. https://doi.org/10.1016/j.hbpd.2021.08.004.

Lu, Chien-Hung, Weu Wang, Yu-Chuan Jack Li, I-Wei Chang, Chi-Long Chen, Chien-Wei Su, Chun-Chao Chang, and Wei-Yu Kao. 2024. “Machine Learning Models for Predicting Significant Liver Fibrosis in Patients with Severe Obesity and Nonalcoholic Fatty Liver Disease.” Obesity Surgery 34 (12): 4393–4404. https://doi.org/10.1007/s11695-024-07548-z.

Mucci, Tim, and Cole Stryker. 2024. “What Is MLOps?” IBM Think, April 5. Accessed March 14, 2025. https://www.ibm.com/think/topics/mlops.

Raschka, S., and V. Mirjalili. 2019. Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-Learn, and TensorFlow 2. Packt Publishing.

Singal, Amit G., Ashin Mukherjee, B. Joseph Elmunzer, Peter D. R. Higgins, Anna S. Lok, Ji Zhu, Jorge A. Marrero, and Akbar K. Waljee. 2013. “Machine Learning Algorithms Outperform Conventional Regression Models in Predicting Development of

Hepatocellular Carcinoma." The American Journal of Gastroenterology 108 (11): 1723–30. https://doi.org/10.1038/ajg.2013.332.

Shrivastava, Abhishek. n.d. "Liver Disease Patient Dataset." Kaggle. Accessed March 10, 2025. https://www.kaggle.com/datasets/abhi8923shriv/liver-disease-patient-dataset/data.

Xiang, Kailai, Baihui Jiang, and Dong Shang. 2021. "The Overview of the Deep Learning Integrated into the Medical Imaging of Liver: A Review." Hepatology International 15 (4): 868–80. https://doi.org/10.1007/s12072-021-10229-z.