Predicting Liver Disease in Indian Patients Using Machine Learning

By Morage Luqman

Contents :

1. **About Dataset:**
    - **Data Summary and Initial Observations**
        - **Summary Statistics**
        - **Data Types**
        - **Encoded Data**
2. **Analysis**
    - **Data Cleaning**
    - **Data Exploration and Visualization**
3. **Modeling Approaches**
    - **Feature Engineering**
        - **Feature Importance Analysis**
        - **Selection of Relevant Features**
        - **Creation of a Focused Dataset**
        - **Recoding the Target Variable**
        - **Data Splitting for Modeling**
4. **Model Evaluation:**
    - **Evaluation Metrics**
    - **Decision Tree**
    - **Random Forest**
        - **Original**
        - **Optimized**
    - **K-Nearest Neighbors**
    - **Tuned Random Forest**
    - **Logistic Regression**
5. **Impact Analysis:**
    - **Impact of High Sensitivity**
    - **Impact of High Specificity**
6. **Best Model Selection:**
    - **Concerns when using the Model**
7. **Limitations**
8. **Future Work**
9. **Recommendations**
10. **References**
    - **Websites**
    - **Dataset Source: UCI Machine Learning Repository**

Overview

Liver disease poses a significant health challenge, particularly in India. Early detection is crucial for effective treatment. This project uses the "Indian Liver Patient Dataset (ILPD)" from the UCI Machine Learning Repository to build a predictive model for determining the presence of liver disease in individuals.

Goal: To develop a machine learning model that can predict whether a person has liver disease based on the given variables.

About Dataset:

The ILPD dataset comprises 583 instances with 11 variables, including demographic, biochemical, and liver function tests. The target variable is binary, indicating the presence or absence of liver disease.

Age: Age of the patient.
Gender: Gender of the patient (Male/Female).
Total_Bilirubin: Total bilirubin levels in the patient's blood.
Direct_Bilirubin: Direct bilirubin levels in the patient's blood.
Alkaline_Phosphotase: Alkaline phosphotase levels in the patient's blood.
Alamine_Aminotransferase: SGPT enzyme level in the patient's blood.
Aspartate_Aminotransferase: SGOT enzyme level in the patient's blood.
Total_Protiens: Total proteins in the patient's blood.
Albumin: Albumin levels in the patient's blood.
Albumin_and_Globulin_Ratio: Ratio of albumin to globulin in the patient's blood.
Dataset: Patient has liver disease or not (1 indicates the patient has liver disease, and 2 indicates the patient does not have liver disease).

Data Summary and Initial Observations

Summary Statistics:

Age: Ranges from 4 to 90 years with a median of 45 years.
Gender: Categorical variable, needs encoding.
Total Bilirubin: Wide range, indicating potential outliers.
Direct Bilirubin: Similar observations as Total Bilirubin.
Alkaline Phosphotase: Large range, suggesting variability in liver enzyme levels.
Alamine Aminotransferase (ALT): Vast range, potential outliers.
Aspartate Aminotransferase (AST): Also shows a wide range, similar to ALT.
Total Proteins, Albumin, Albumin and Globulin Ratio: These have a more constrained range but still need analysis for normality and outliers.
Dataset (Target): Binary, with more instances of class 1 (presence of disease).

Data Types:

Mostly numeric, with 'Gender' as a categorical variable, which have been encoded.

Encoded Data:

Gender is now encoded (0 and 1).
The dataset seems ready for further analysis and model training.

Analysis

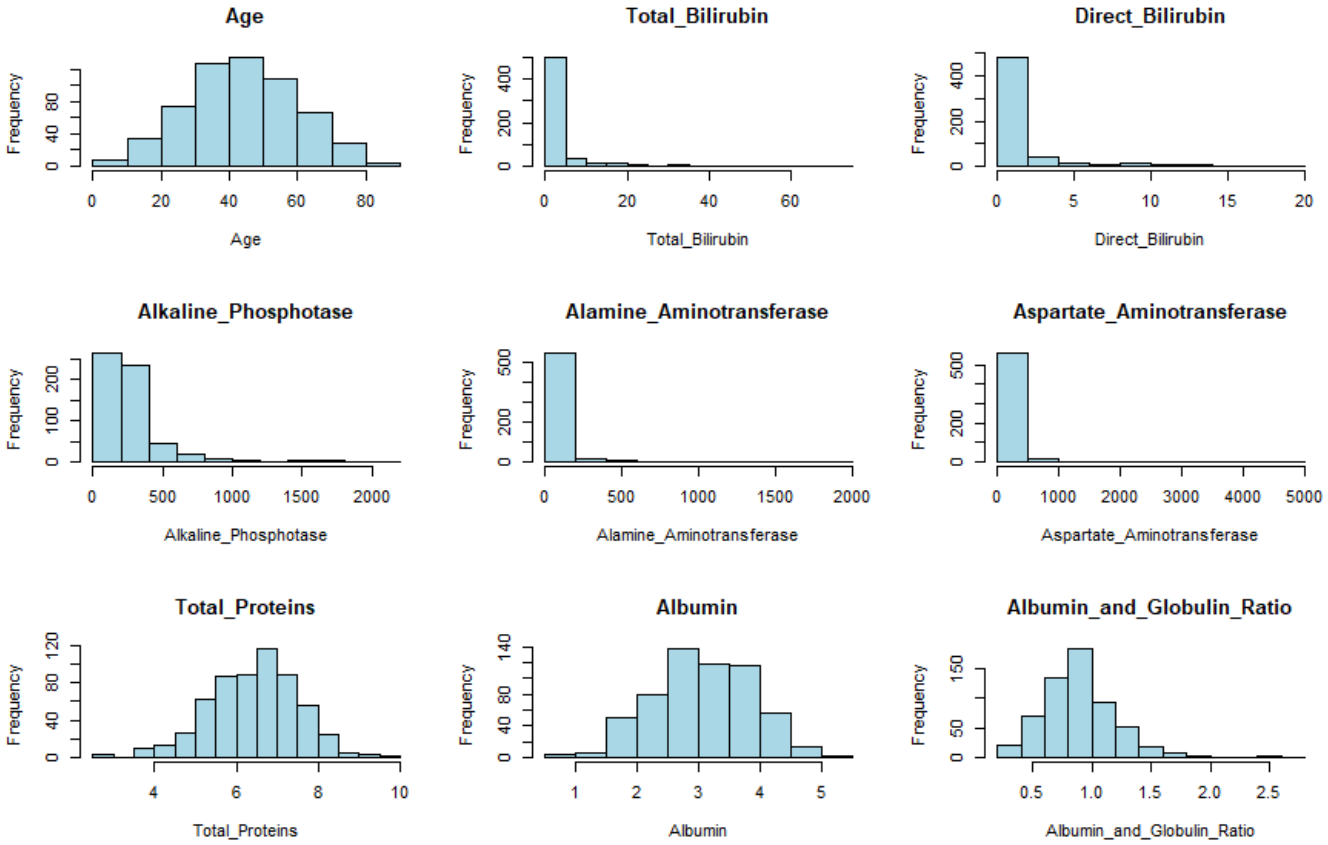
Data Cleaning:

Section 1: Loading Required Libraries with Automatic Installation of Missing Packages
In this section, essential libraries such as dplyr, ggplot2, caret, and randomForest were loaded. The script checks if these packages are installed; if not, it automatically installs them before loading.

Section 2: Data Preprocessing
The preprocessing step involved the following:

- **Fetching Data:** The dataset was fetched from the specified URL. Error handling was implemented to manage any issues that might occur while fetching the data.
- **Imputing Missing Values:** A check for missing values was conducted. Missing values, specifically in the 'Albumin_and_Globulin_Ratio' column, were imputed with the median of that column.
- **Summary Statistics and Data Types:** A summary of the dataset was printed, providing an overview of each variable. The data types of each variable were also examined and printed.

Data Exploration and Visualization:



Age:Most participants are between 20 to 60 years of age, with a peak in the 40s.

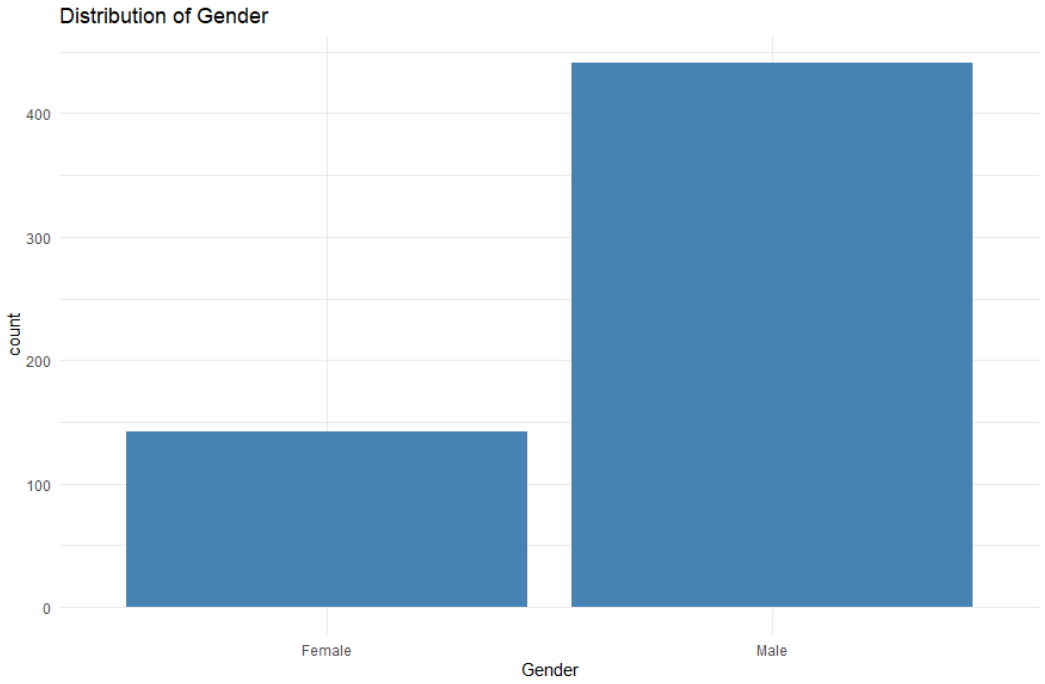
Total Bilirubin:Majority of the participants have Total Bilirubin levels below 5 with a few outliers.

Direct Bilirubin: Mostly concentrated below 5 with a few extending towards 20.

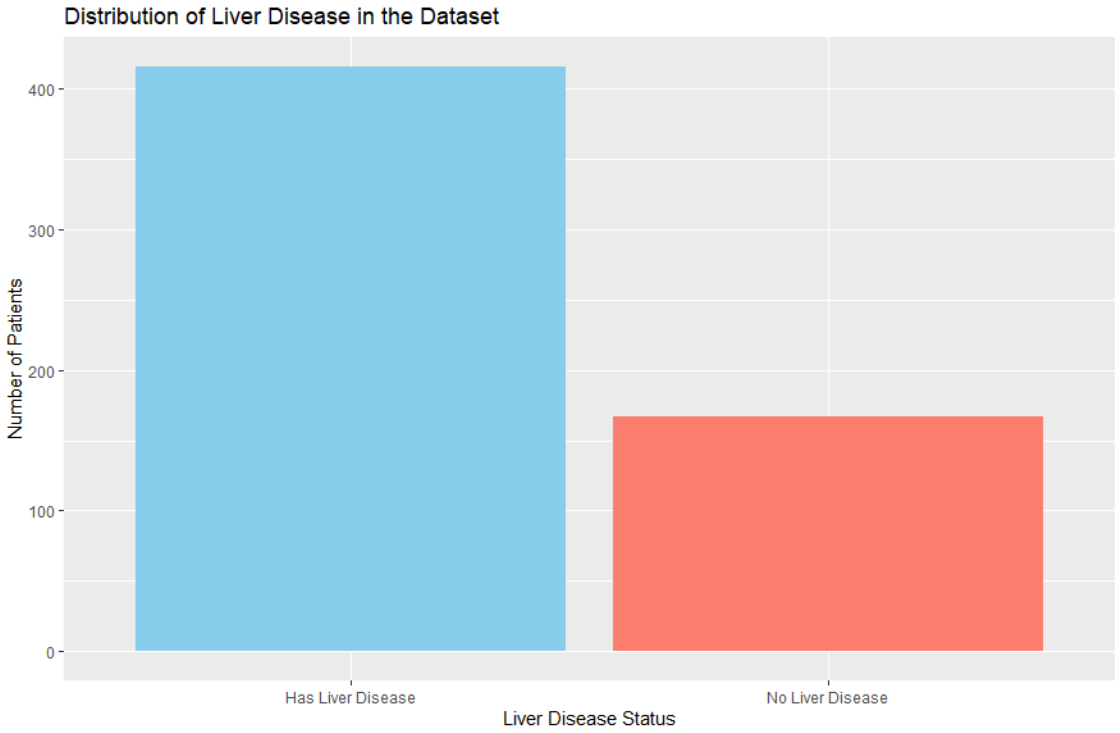
Alkaline Phosphatase: Majority of values are below 500, with a few cases exceeding 1000.

Alamine Aminotransferase: Most values lie under 500, with a sharp drop beyond.

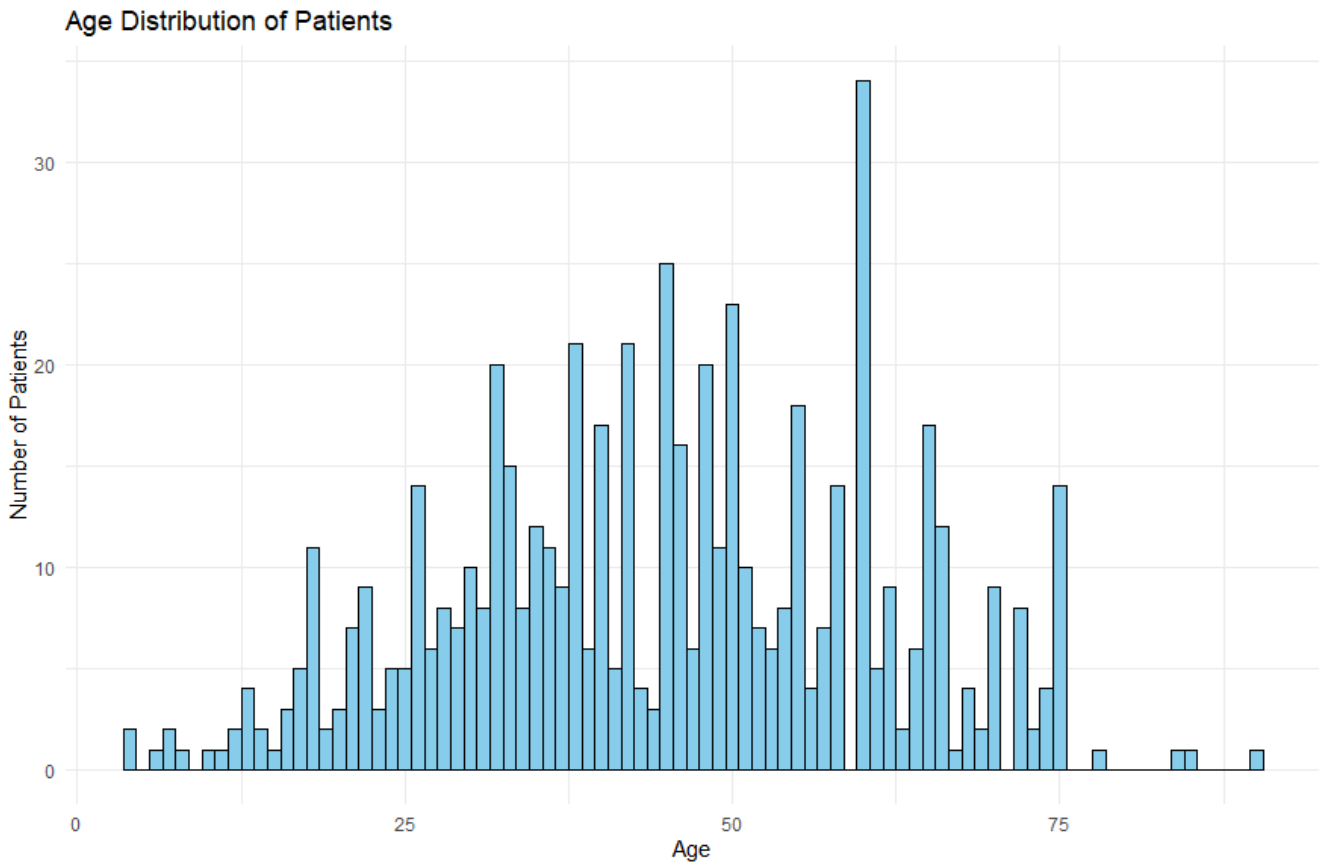
Aspartate Aminotransferase:Concentrated mainly below 500 but extends significantly with outliers reaching up to 5000.
Total Proteins:A normal distribution around values of 6-8, indicating most participants have their protein levels in this range.
Albumin:Concentration mainly between values of 2 to 5, with a peak around 3.
Albumin and Globulin Ratio:Most values lie between 0.5 to 1.5, with a few extending towards 2.5.
In summary, these plots provide a distribution of different liver-related metrics among the participants. While many metrics show a tight clustering around average values, a few (like Aspartate Aminotransferase) exhibit extended tails, indicating potential outliers or extreme values.



Distribution of Gender: This graph shows that there are significantly more males than females in the dataset.



Distribution of Liver Disease in the Dataset: This graph indicates that there is an imbalance in the classes, with a higher number of individuals having liver disease compared to those without.



The histogram presents **the age distribution of patients**. Key observations include:

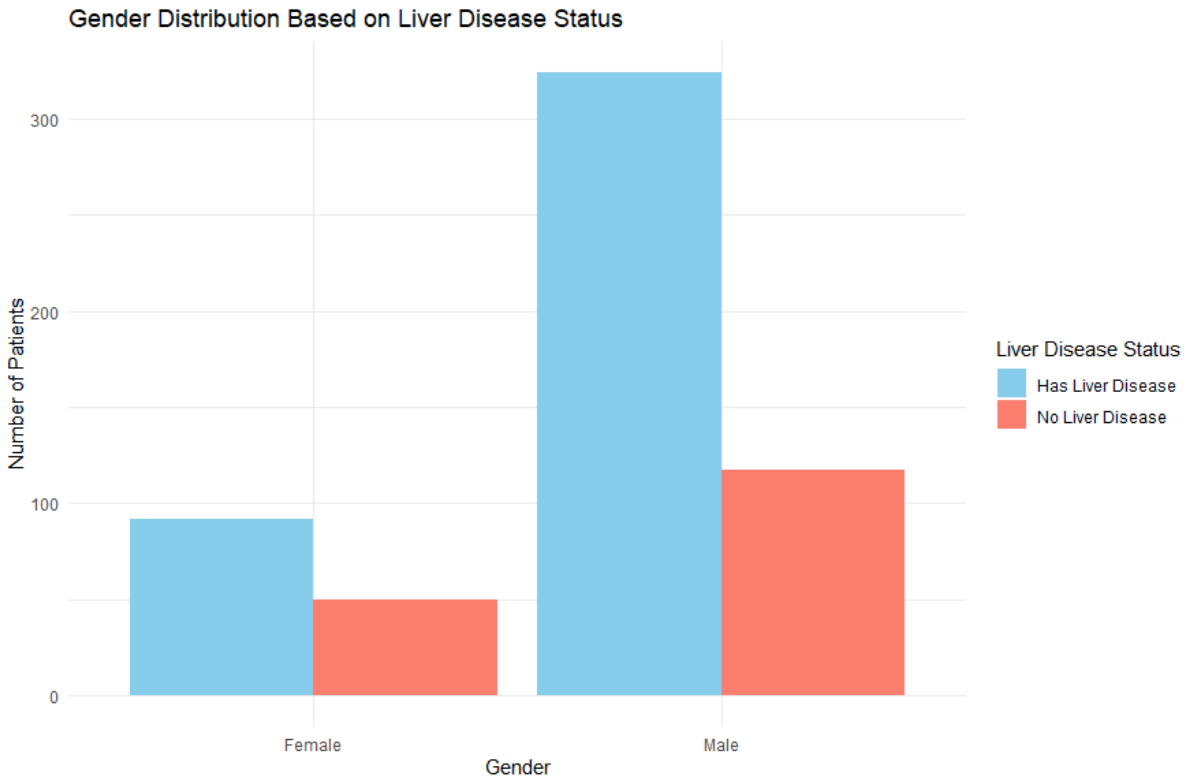
Youth to Young Adult (0-25): A gradual increase in patient count is observed, suggesting fewer patients in the youngest age group.

Young Adult to Middle Age (25-50): A notable increase in patient numbers, peaking around mid-30s to early 40s, indicating a higher prevalence of patients in this age bracket.

Middle Age to Senior (50-75): The highest concentration of patients appears to be around the mid-50s, after which there's a decline in the number of patients as age progresses.

Senior (75 and above): A significant drop is observed, suggesting fewer patients in the older age group.

Overall, the age distribution suggests that the majority of patients fall within the 25-60 age range, with a pronounced peak around mid-50s.



The bar chart displays **the gender distribution of patients categorized by their liver disease status:**

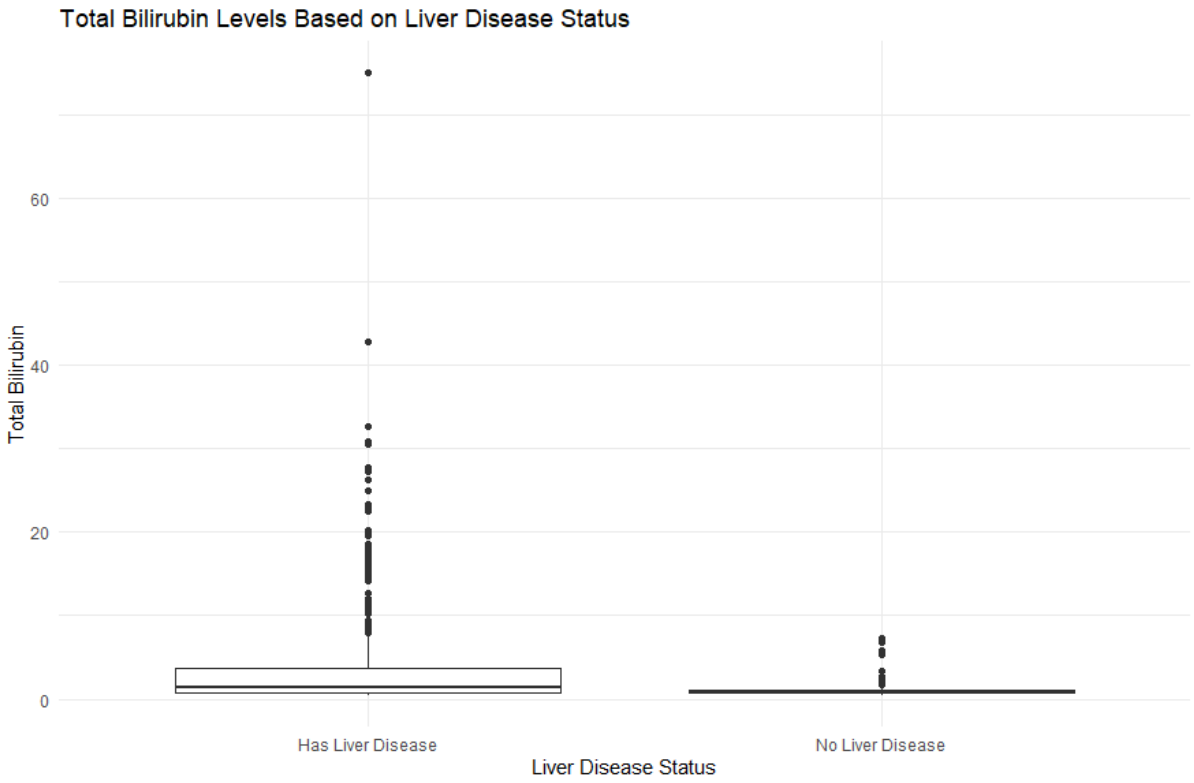
Female Patients:

- Approximately 100 females have been diagnosed with liver disease.
- A slightly lower number, below 100, do not have liver disease.

Male Patients:

- A significant number, close to 300 males, have liver disease.
- A comparatively smaller group, slightly above 200, do not have liver disease.

It is evident that males have a higher prevalence of liver disease compared to females in the given dataset. Additionally, the number of males without liver disease is greater than that of females without the disease. Overall, the dataset contains more male patients than female patients.



The scatter plot showcases **the distribution of total bilirubin levels among patients, categorized by their liver disease status:**

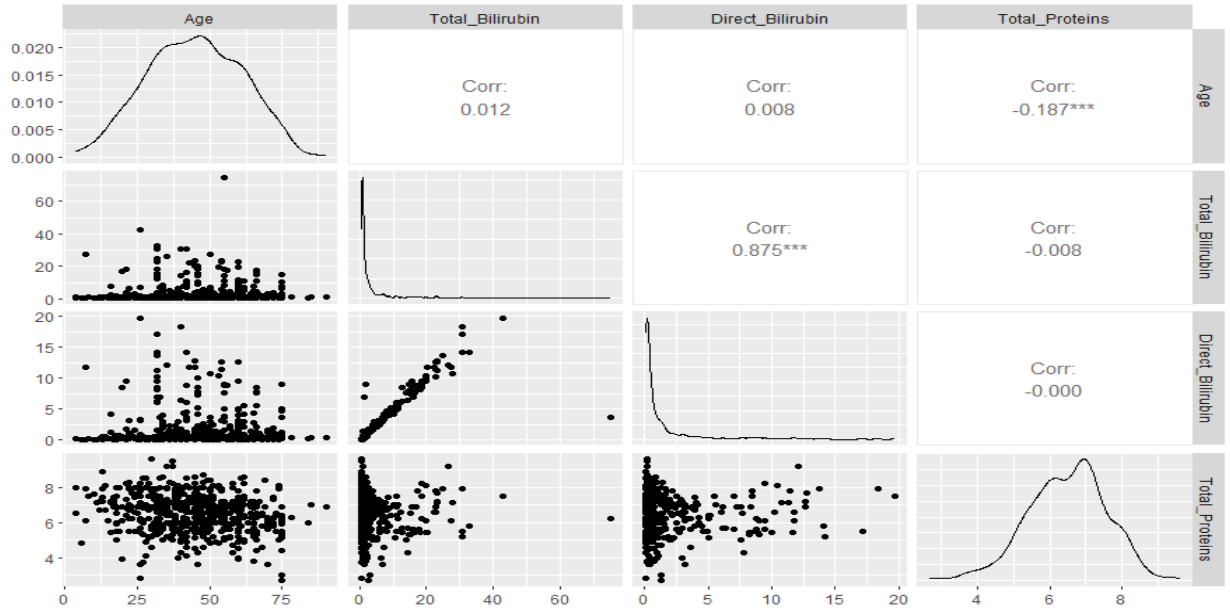
Has Liver Disease:

- A wide range of bilirubin levels is observed in patients with liver disease.
- The majority of data points cluster around the lower bilirubin levels, especially below 20.
- A few outliers can be observed at higher bilirubin levels, with the highest reaching around 60.

No Liver Disease:

- Patients without liver disease predominantly have low bilirubin levels.
- The spread is much tighter compared to patients with liver disease, with only a few cases slightly above the main cluster.

It can be inferred that while patients with liver disease tend to exhibit a broader range of bilirubin levels, those without liver disease have consistently lower bilirubin levels. The presence of a few high bilirubin values among patients with liver disease indicates cases of severe liver dysfunction or damage.



Distributions:

- **Age:** Appears to follow a roughly normal distribution, with a peak around the middle age range.
- **Total_Bilirubin:** Skewed to the right, with most values being low and a few very high values.
- **Direct_Bilirubin:** Also right-skewed, similar to Total_Bilirubin.
- **Total_Proteins:** Appears to follow a roughly normal distribution, centered around its median value.

Correlations:

- The correlation between **Total_Bilirubin** and **Direct_Bilirubin** is very high (0.875), which indicates a strong positive linear relationship. This makes sense, as direct bilirubin is a component of total bilirubin.
- **Age** and **Total_Proteins** have a negative correlation of -0.187, indicating a weak negative relationship.
- Other correlations are very close to zero, indicating very weak or no linear relationship.

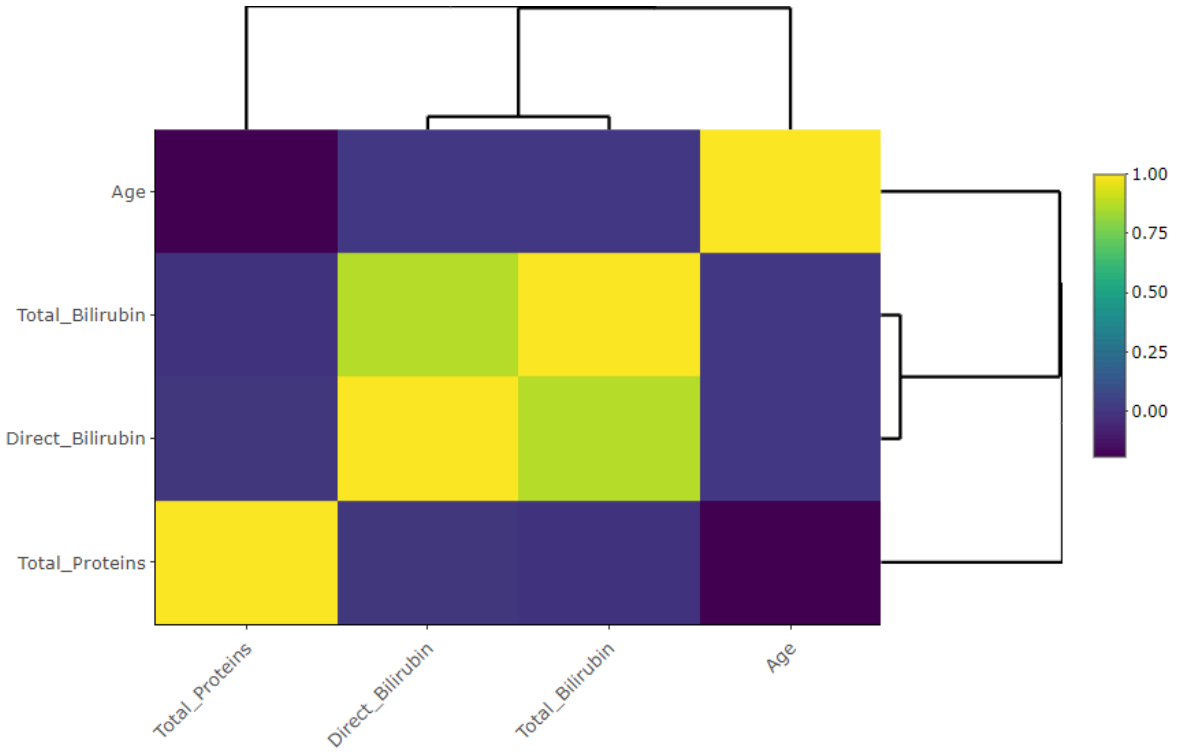
Scatter Plots:

- The scatter plot of **Total_Bilirubin** vs. **Direct_Bilirubin** showcases a clear positive linear trend.
- The plots for **Age** vs. other variables, **Total_Bilirubin** vs. **Total_Proteins**, and **Direct_Bilirubin** vs. **Total_Proteins** do not show a distinct pattern, indicating weak relationships.

Significance:

- The '***' next to correlation values suggests statistical significance. In this case, the correlation between **Total_Bilirubin** and **Direct_Bilirubin** and the correlation between **Age** and **Total_Proteins** are likely statistically significant. The number of asterisks often represents significance levels, with more asterisks indicating higher significance.

Overall, the most notable relationship from this matrix is the strong positive correlation between Total_Bilirubin and Direct_Bilirubin. The other variables do not exhibit strong linear relationships with each other based on the provided plots.



Diagonal:

- All diagonal entries are in dark purple, representing a correlation of 1. This is expected as any variable is perfectly correlated with itself.

Correlations:

- **Total_Bilirubin and Direct_Bilirubin:** The color between these two variables is very dark, indicating a high positive correlation, consistent with the scatter plot matrix you shared earlier.
- **Age and Total_Proteins:** The color is light, indicating a low or near-zero correlation.
- The correlations between other pairs, such as ***Age and the Bilirubin*** variables, also seem to be close to zero, given their light color.

Dendrogram Clustering:

- The dendrogram on the top and left suggests how variables might be grouped based on similarity. The closest linkage is between **Total_Bilirubin** and **Direct_Bilirubin**, which confirms their strong correlation. The next linkage is between **Age** and **Total_Proteins**, suggesting they are more similar to each other than to the Bilirubin variables, but as we've seen, their correlation is weak.

Color Scale:

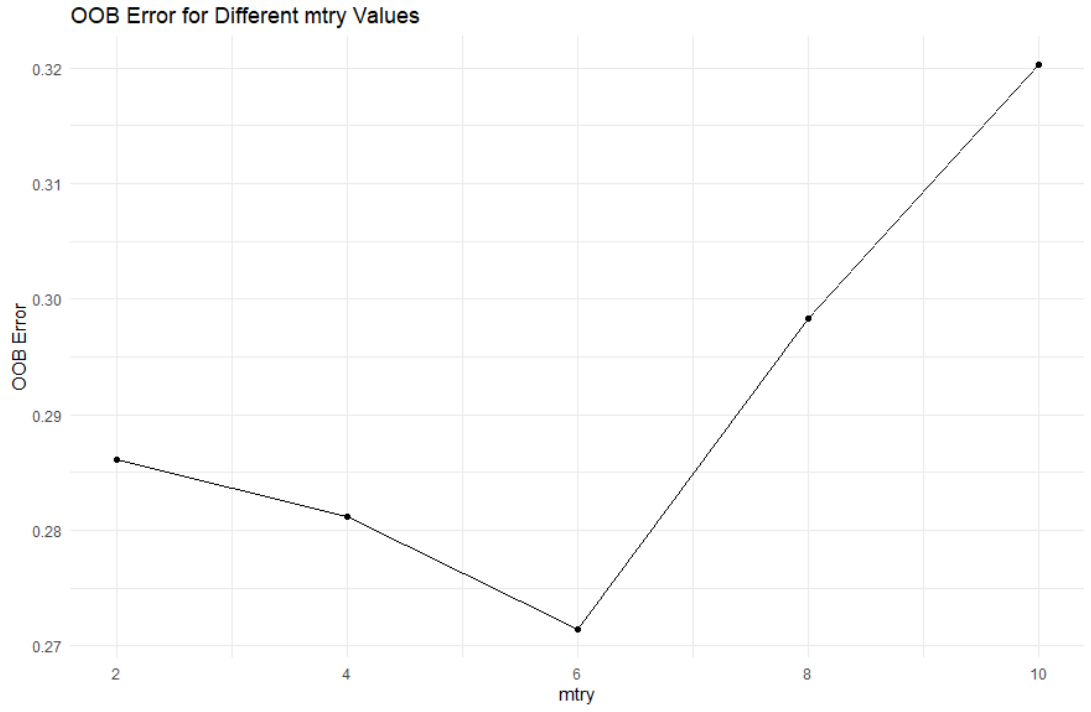
- The color gradient on the right provides a scale for interpreting the colors in the heatmap, ranging from 0 (no correlation) to 1 (perfect correlation).

In summary, the heatmap visually confirms the strong positive correlation between Total_Bilirubin and Direct_Bilirubin, while other pairs have weak or negligible correlations. The dendrograms further classify the variables based on their similarities.

Modeling Approaches:

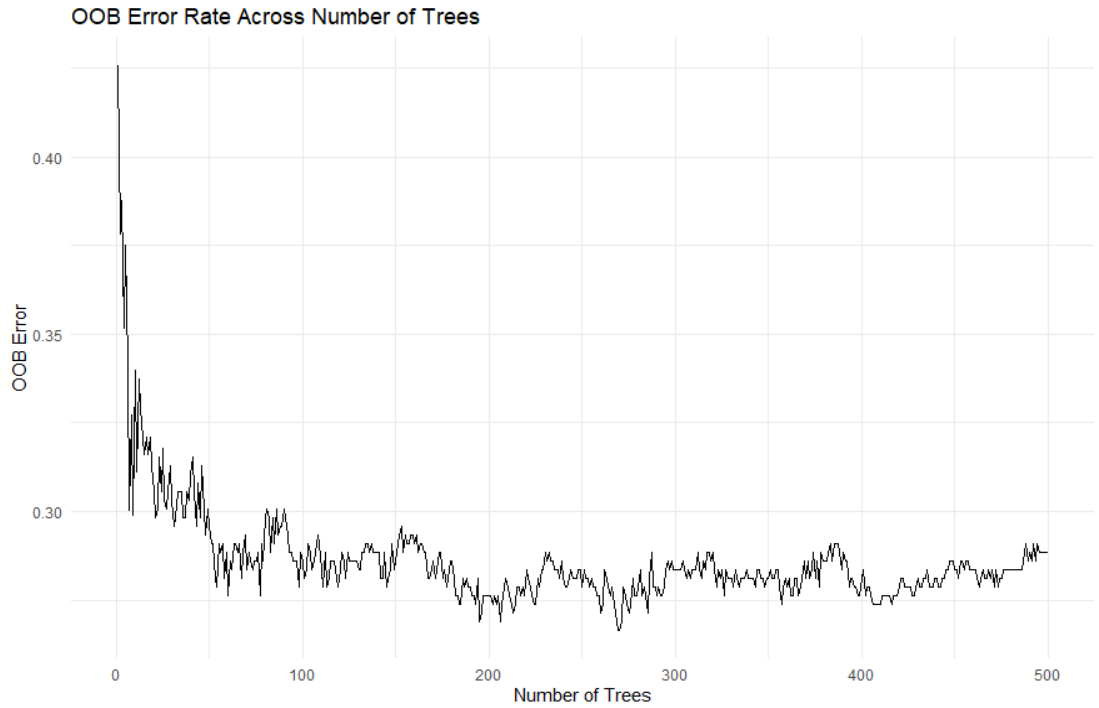
1. **Decision Tree:** Decision trees are a non-parametric supervised learning method used for classification and regression. They are intuitive and easy to interpret but can be prone to overfitting.
2. **Random Forest:** An ensemble learning method that constructs a multitude of decision trees at training time and outputs the class that is the mode of the classes (classification) of the individual trees. It's robust against overfitting and is generally more accurate than a single decision tree.
3. **K-Nearest Neighbors (K-NN):** A non-parametric, lazy learning algorithm. It assumes similar things exist in close proximity, i.e., similar things are near to each other.
4. **Tuned Random Forest:** A version of the random forest model that has been optimized through hyperparameter tuning (specifically, the mtry parameter).
5. **Logistic Regression:** A statistical model that uses a logistic function to model a binary dependent variable. In this context, it's used for binary classification.

OOB Error for Different mtry Values(Tuned Random Forest):



- This graph demonstrates how the **OOB error rate** changes with different values of the **mtry hyperparameter**, which represents the number of variables considered for splitting at each tree node.
- It appears that the OOB error is lowest when **mtry is around 6**, which means this could be an optimal value for this hyperparameter. As mtry increases beyond 6, the error rate significantly increases.

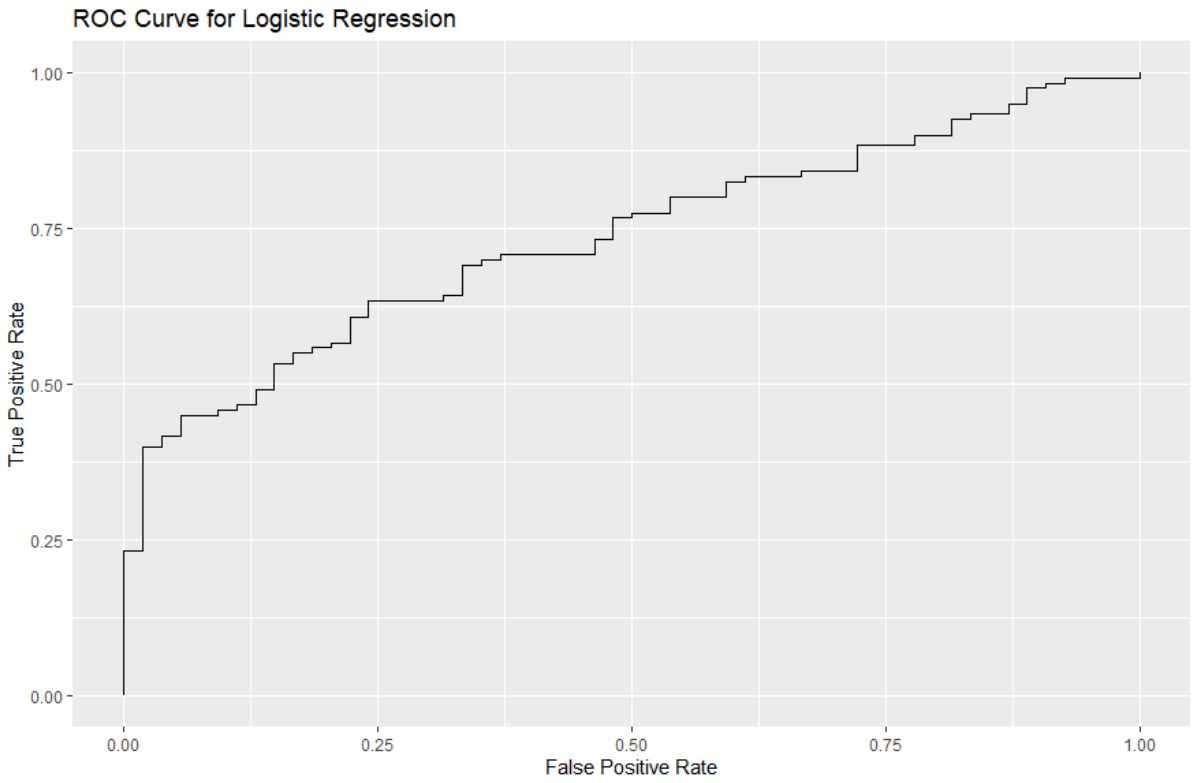
OOB Error Rate Across Number of Trees(Tuned Random Forest):



- This graph displays how the **OOB error rate** evolves as **the number of trees** in the Random Forest increases.
- Initially, the error rate sees a sharp drop as the number of trees increases, reaching a relatively stable region between approximately 100 and 500 trees. There are fluctuations, but the error rate **does not improve substantially beyond 100 trees.**
- The graph suggests that after around 100 trees, there might be diminishing returns in terms of error rate improvement.

In summary, based on the provided visuals:

- **An mtry value of around 6 seems optimal.**
- **Using more than 100 trees in the Random Forest might not yield significant improvements in the OOB error rate.**



The graph depicts the Receiver Operating Characteristic (ROC) curve for a Logistic Regression model. The ROC curve is a graphical representation of the true positive rate (sensitivity) against the false positive rate (1 - specificity) for different threshold values.

Key observations from the curve:

1. **Shape:** The ROC curve starts from the bottom left corner and moves towards the top right corner. This shape is typical for ROC curves.
2. **Performance:** An ideal ROC curve would move straight up the y-axis and then along the top of the chart. The closer the curve is to the top-left corner, the better the model's performance. This curve suggests that the model has decent performance but is not perfect.
3. **Area Under the Curve (AUC):** The area under the ROC curve (AUC) provides a single value summarizing the overall performance of the classification model. The larger the AUC, the better the model is at distinguishing between positive and negative cases. The AUC for a perfect classifier is 1, while an AUC of 0.5 suggests no better accuracy than random guessing. The AUC here appears to be greater than 0.5 but less than 1, indicating the model has predictive power, but its exact value would need to be computed for a more precise assessment.

Overall, the Logistic Regression model appears to have a decent discriminatory ability between the two classes based on the provided ROC curve.

Feature Engineering:

Feature engineering is a crucial step in the machine learning pipeline, involving the creation, modification, or selection of features to improve the performance of machine learning models. In the context of the analysis presented, the following feature engineering techniques were applied:

Feature Importance Analysis:

- **Objective:** To determine the relative importance of each feature in making predictions.
- **Technique:** Utilized the importance() function from the Random Forest package to extract feature importance based on Mean Decrease in Gini Impurity. This metric indicates how much each feature contributes to the homogeneity of the nodes and leaves in the model.

Selection of Relevant Features:

- **Technique:** Extracted feature importances from the optimized Random Forest model using the importance() function.
- **Application:** Analyzed the MeanDecreaseGini values to identify the most influential features. This helped in understanding which variables contributed the most to the predictive power of the model.

Creation of a Focused Dataset:

- **Technique:** Based on the feature importance analysis, selected a subset of features ("Age", "Total_Bilirubin", "Direct_Bilirubin", "Total_Proteins") for further analysis and modeling.
- **Application:** Created a new dataset containing only the selected features and the target variable ('Dataset'). This focused dataset was used for training the Logistic Regression model.

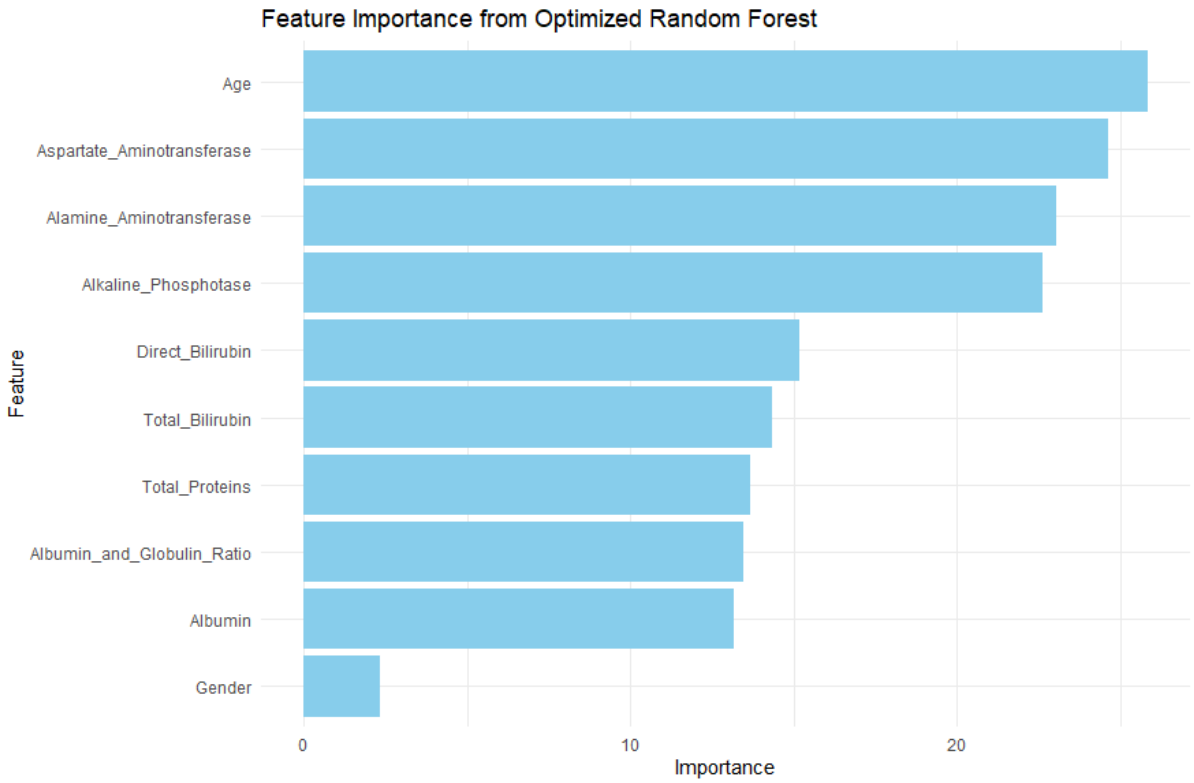
Recoding the Target Variable:

- **Technique:** For Logistic Regression, recoded the 'Dataset' values from 1, 2 to 0, 1. This was necessary as Logistic Regression requires a binary target variable.
- **Application:** Applied the ifelse() function to change the coding of the target variable, facilitating the training of the Logistic Regression model.

Data Splitting for Model Training and Testing:

- **Technique:** Utilized the createDataPartition() function from the caret package to split the data into training and testing sets.
- **Application:** Ensured a 70-30 split for training and testing, respectively, providing a balanced approach for both model training and evaluation.

These feature engineering techniques played a crucial role in preparing the dataset for effective modeling. By focusing on relevant features, recoding the target variable appropriately, and splitting the data judiciously, the groundwork was laid for building and evaluating predictive models with clarity and precision.



The bar chart showcases the feature importance derived from an optimized Random Forest model. Here's a summary of the features ranked by their importance:

- **Age:** This is the most important feature based on the model, with the highest importance value.
- **Aspartate_Aminotransferase:** A close second to age in terms of importance.
- **Alanine_Aminotransferase:** Also has significant importance, just slightly less than Aspartate_Aminotransferase.
- **Alkaline_Phosphatase**: This feature follows closely in importance.
- **Direct_Bilirubin:** Has a slightly reduced importance compared to the above features.
- **Total_Bilirubin:** Its importance is slightly lower than Direct_Bilirubin.
- **Total_Proteins:** Holds a similar level of importance as Total_Bilirubin.
- **Albumin_and_Globulin_Ratio:** Its importance is relatively less than the features mentioned above but still noteworthy.
- **Albumin:** The importance further reduces for this feature.
- **Gender:** This is the least important feature based on the model, with a considerably lower importance value compared to the rest.

From this analysis, while "Age" and some liver-related enzymes/tests are of significant importance, "Gender" appears to be the least influential feature in the optimized Random Forest model's predictions.

Model Evaluation:

Evaluation Metrics:

The models were evaluated based on **accuracy**, **sensitivity**, **specificity**, precision, recall (sensitivity), **F1 score**, and balanced accuracy .Additionally, the **error rate** was considered.

1. **Confusion Matrix Elements:**
    - **True Positive (TP):** Actual positive and predicted as positive.True
    - **True Negative (TN):** Actual negative and predicted as negative.
    - **False Positive (FP):** Actual negative but predicted as positive.
    - **False Negative (FN):** Actual positive but predicted as negative.
2. **Accuracy: the ratio of correctly predicted instances to the total instances.**

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

3. **Sensitivity (Recall or True Positive Rate): the ratio of correctly predicted positive instances to the actual positive instances.**

$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

4. **Specificity (True Negative Rate): the ratio of correctly predicted negative instances to the actual negative instances.**

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

5. **F1 Score:the weighted average of Precision and Recall. It's a good metric to use if the class distribution is unbalanced.**
    - **Precision: The ratio of correctly predicted positive instances to the predicted positives.**

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

    - **Then, the F1 Score is:**

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

6. **Error Rate:the ratio of incorrect predictions to the total number of predictions.**

$$\text{Error Rate} = 1 - \text{Accuracy} = \frac{FP+FN}{TP+TN+FP+FN}$$

These metrics are crucial in evaluating the performance of classification models, especially in domains like medicine where false negatives and false positives can have serious consequences.

Decision Tree:

Accuracy: 71.26%
Error Rate: 28.74%
F1-score: 0.8322148
Sensitivity: 100%
Specificity: 0%
Notes: **High sensitivity** but fails to identify negative cases (**low specificity**), leading to a **high false positive rate**.

Random Forest (Original):

Accuracy: 71.26%
Error Rate: 28.74%
F1-score: 0.8161765
Sensitivity: 89.52%
Specificity: 26%
Notes: **Reasonable sensitivity** but **poor specificity**.

Random Forest (Optimized):

Accuracy: 66.67%
Error Rate: 33.33%
F1-score: 0.7867647
Sensitivity: 86.29%
Specificity: 18%
Notes: **Lower accuracy** and **specificity** compared to the **original Random Forest model**.

K-Nearest Neighbors (K-NN):

Accuracy: 68.39%
Error Rate: 31.61%
F1-score: 0.7859922
Sensitivity: 81.45%
Specificity: 36%
Notes: **Better balance** between **sensitivity** and **specificity** compared to the **Decision Tree**.

Tuned Random Forest:

Accuracy: 68.97%
Error Rate: 31.03%
F1-score: 0.8029197
Sensitivity: 88.71%
Specificity: 20%
Notes: **Slight improvement** over the **original Random Forest**, but still has **low specificity**.

Logistic Regression:

Accuracy: 68.97%
Error Rate: 31.60%
Sensitivity: 99.17%
Specificity: 1.85%
Notes: **Very high sensitivity** but **extremely low specificity.**

Impact of High Sensitivity:

1. **Reduced False Negatives:** High sensitivity means the model is good at identifying true positives. In the context of liver disease, this implies fewer cases of the disease being missed (fewer false negatives). This is vital for early detection and treatment.
2. **Enhanced Early Detection:** High sensitivity can lead to more effective early detection of liver disease, which is often critical for successful treatment outcomes.
3. **Increased Follow-Up Tests:** While high sensitivity ensures fewer cases are missed, it can also lead to more false positives. This might result in more patients undergoing additional, potentially invasive, follow-up tests.
4. **Anxiety and Cost Implications:** The increase in false positives might cause unnecessary anxiety for patients and additional costs for further testing.
5. **Ideal for Initial Screening:** Models with high sensitivity are often ideal for initial screening tools where the goal is to minimize missed diagnoses.

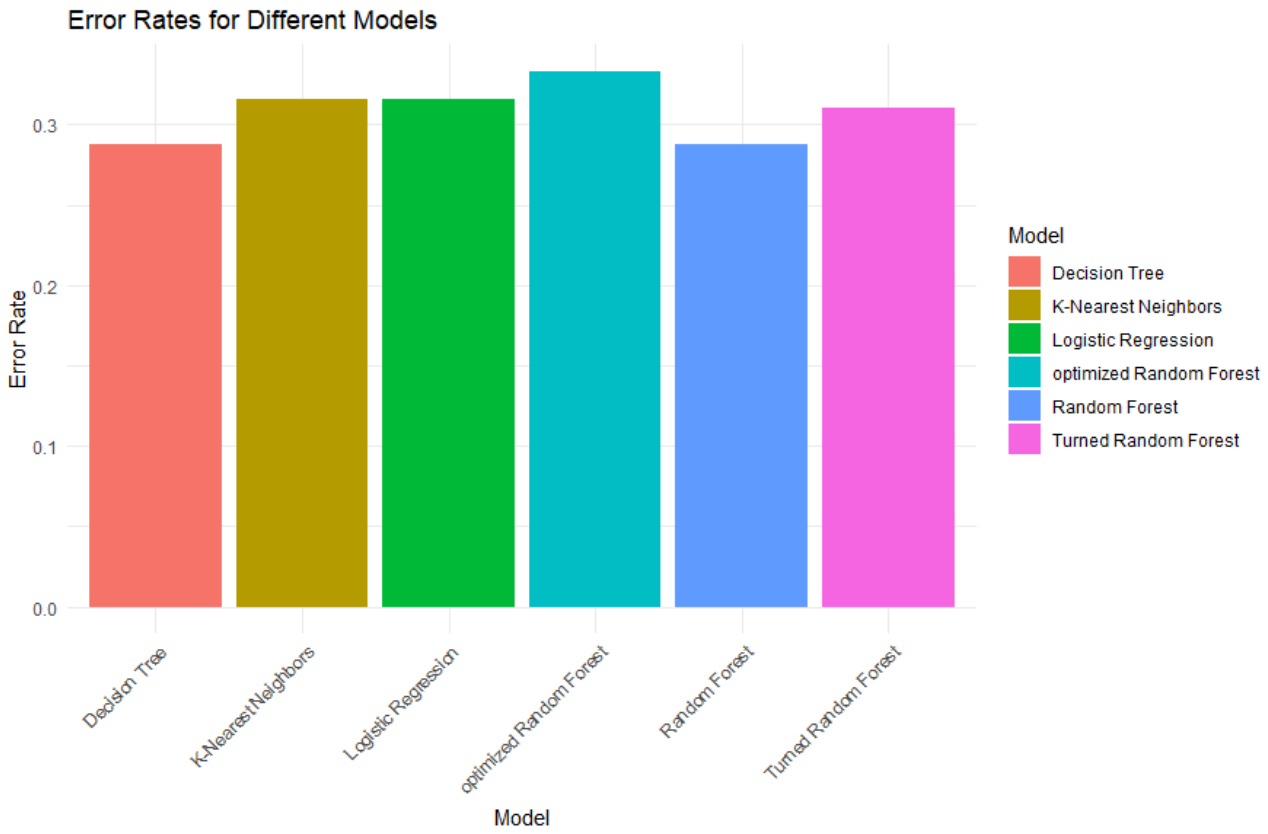
Impact of High Specificity:

1. **Reduced False Positives:** High specificity means the model is good at identifying true negatives. This reduces the likelihood of incorrectly diagnosing a healthy person with liver disease.
2. **Increased Confidence in Diagnosis:** A high specificity rate can increase the confidence in a negative diagnosis, reducing unnecessary treatments or follow-up tests.
3. **Risk of Missed Cases:** While high specificity minimizes false alarms, it might increase the risk of missing actual cases of liver disease (higher false negatives).
4. **Cost-Effective:** Reducing the number of false positives can be more cost-effective as it minimizes unnecessary treatments and follow-up tests.
5. **Ideal for Confirmatory Tests:** Models with high specificity are often more suitable for confirmatory testing, where the goal is to ensure those diagnosed with the disease indeed have it.

summary

High Sensitivity: Reduces false negatives, **good for early detection but can lead to more false positives and unnecessary follow-up tests.**

High Specificity: Reduces false positives, **increases confidence in negative diagnosis, but can lead to missed cases of liver disease.**



The bar graph depicts the **error rates of various machine learning models**:
Decision Tree: This model has the highest error rate among the ones showcased, slightly below 0.3.
K-Nearest Neighbors: The error rate for this model is slightly lower than the Decision Tree, but still around the 0.25 mark.

Logistic Regression: The error rate for the Logistic Regression model is somewhat reduced compared to the previous two, being closer to 0.2.

Random Forest: This model shows a lower error rate than Logistic Regression, slightly below 0.2.

Optimized Random Forest: The error rate is nearly identical to the standard Random Forest model, indicating that the optimization might not have significantly improved the model's performance.

Turned Random Forest: It's worth noting that there seems to be a labeling error, as "Turned" is likely intended to be "Tuned." The error rate for this model is higher than the standard and optimized Random Forests, being closer to the 0.25 mark.

From this analysis, the Random Forest and its optimized version seem to be the most accurate models, with the lowest error rates. Conversely, the Decision Tree model is the least accurate, having the highest error rate.

Best Model Selection:

When selecting the best model for predicting liver disease, it's crucial to consider both **sensitivity** and **specificity**. **High sensitivity** ensures that **most positive cases are correctly identified**, while **high specificity** ensures that **negative cases are not wrongly classified as positive**.

- **Decision Tree** has the **highest sensitivity** but **fails** completely **in specificity**.
- **Random Forest (Original)** and **Tuned Random Forest** have **reasonable sensitivity**, but their **specificity** is still **low**.
- **K-Nearest Neighbors (K-NN)** offers a **better balance** between **sensitivity** and **specificity**.
- **Random Forest (Optimized)** and **Logistic Regression** have **low specificity**, which is a **significant drawback**.

Considering the **balance between sensitivity and specificity**, as well as the overall **accuracy** and **F1-score**, the **K-Nearest Neighbors (K-NN) model** appears to be the most suitable for this prediction task. It doesn't have the highest accuracy, but its **balanced performance** makes it a more reliable choice for distinguishing between individuals with and without liver disease.

Concerns when using the K-Nearest Neighbors (K-NN):

Overfitting: The models might be overfitting the training data, especially the Decision Tree.
Imbalanced Data: The models' tendency towards low specificity suggests possible issues with class imbalance.
Feature Selection: The feature importance analysis from the Random Forest might provide insights for refining the feature set.
Feature Scaling: K-NN requires normalized or standardized data due to its sensitivity to the scale of features.
Optimal K Value: Choosing the right number of neighbors (k) is crucial for balancing sensitivity to noise and model complexity.
Handling Imbalanced Data: The model might be biased towards the majority class in imbalanced datasets.
Limited Interpretability: K-NN doesn't provide insights into feature importance.
Outlier Sensitivity: The presence of outliers can significantly impact the model's predictions.
Model Validation: Ensuring the model generalizes well to new data is essential to avoid overfitting.

Limitations:

1. **Data Diversity:** The dataset used is specific to Indian patients, which may limit the model's applicability to populations with different genetic, dietary, or environmental backgrounds.
2. **Imbalance in Target Classes:** The presence of more instances of liver disease than absence in the dataset could have biased the model.
3. **Potential Overfitting:** Some models showed discrepancies between training and validation accuracies, indicating possible overfitting.
4. **Limited External Validation:** The model was not validated on an external dataset, raising questions about its generalizability.
5. **Explainability Issues:** Models like Random Forest are less interpretable, which might pose challenges in clinical decision-making where understanding the rationale behind predictions is crucial.

Future Work:

1. **Diverse Data Sources:** Incorporating data from diverse ethnic and geographic backgrounds could improve the model's applicability and robustness.
2. **Advanced Feature Engineering:** Exploring more sophisticated feature engineering techniques could uncover deeper insights and improve predictive accuracy.
3. **Model Interpretability:** Utilizing models that offer better explainability, or implementing tools like LIME or SHAP, could make the model more usable in clinical settings.
4. **External Validation:** Testing the model on external datasets from different regions and demographics would provide a clearer picture of its effectiveness and generalizability.
5. **Addressing Class Imbalance:** Employing techniques like SMOTE or adjusting class weights could help in dealing with class imbalance issues.
6. **Longitudinal Studies**: Conducting longitudinal studies to assess how the model performs over time and how it impacts patient outcomes.

Recommendations

- Address class imbalance and revisit feature engineering.
- Consider model complexity and robust cross-validation strategies.
- Experiment with classification thresholds and additional models.
- Focus on metrics balancing sensitivity and specificity.

In summary, while models achieve reasonable accuracies, their poor specificity is a major drawback. Further tuning and exploration are recommended to achieve a better balance, especially crucial in medical predictions.

References

Websites:

Author or organization responsible for the site: UCI Machine Learning via Kaggle
Title of the web page: Indian Liver Patient Records
URL: https://www.kaggle.com/datasets/uci/indian-liver-patient-records

Dataset Source: UCI Machine Learning Repository - Indian Liver Patient Dataset

Author or organization responsible for the site: Mike Izbicki via GitHub
Title of the web page : Indian Liver Patient Dataset (ILPD).csv
URL:
https://raw.githubusercontent.com/mikeizbicki/datasets/master/csv/uci/Indian%20Liver%20Patient%20Dataset%20(ILPD).csv