Revolutionizing Liver Care: Predicting Liver Cirrhosis Using Advanced Machine Learning Techniques

1. Introduction

1.1. Project Overview

This project aims to build a machine learning model to predict liver cirrhosis based on various patient features. By analysing these features, the model will classify patients into risk categories, aiding in early diagnosis and treatment.

1.2. Objectives

- Collect and prepare a dataset of liver health characteristics.
- Perform exploratory data analysis (EDA) and visualize the data.
- Build and evaluate multiple machine learning models.
- Optimize the best-performing model using hyperparameter tuning.
- Deploy the final model for practical use.

2. Project Initialization and Planning Phase

2.1. Define Problem Statement

The goal is to classify patients' risk levels for liver cirrhosis based on their medical data. Accurate prediction will support better management and early intervention for liver health.

2.2. Project Proposal (Proposed Solution)

The solution involves developing several machine learning models to predict liver cirrhosis. We will select and optimize the best model based on performance metrics to achieve the highest accuracy.

2.3. Initial Project Planning

Initial planning included setting up the project environment, defining objectives, and outlining the workflow for data collection, preprocessing, model development, and evaluation.

3. Data Collection and Preprocessing Phase

3.1. Data Collection Plan and Raw Data Sources Identified

The dataset for this project was sourced from Kaggle, containing patient data relevant to liver cirrhosis prediction (Dataset link: https://www.kaggle.com/datasets/bhavanipriya222/liver-cirrhosis-prediction).

3.2. Data Quality Report

- **Data Shape:** The dataset initially comprised [number of rows, number of columns] rows and columns.
- **Missing Values:** Handled by dropping rows with missing values.

3.3. Data Exploration and Preprocessing

- **Univariate Analysis:** Histograms were plotted for numerical features.
- **Bivariate Analysis:** Scatter plots and pair plots explored relationships between features. • **Outlier Handling:** Outliers were detected and managed using the IQR method.

4. Model Development Phase

4.1. Feature Selection Report

Features relevant to liver cirrhosis prediction were selected, and data scaling was applied to standardize the input.

4.2. Model Selection Report

- **Models Tested:** Naive Bayes, Random Forest, Logistic Regression, Ridge Classifier, Support Vector Classifier, KNN, XG Boost.
- **Evaluation Metrics:** Accuracy, Confusion Matrix, Classification Report.

4.3. Initial Model Training Code, Model Validation and Evaluation Report

- **Code:** Model training and evaluation steps were implemented for each algorithm.
- **Validation:** Models were validated using a test set, with performance metrics recorded. The KNN model achieved the highest accuracy of 86.32%.

5. Model Optimization and Tuning Phase

5.1. Hyperparameter Tuning Documentation

- **KNN:** Optimized by tuning the number of neighbors and distance metrics.

- **XG Boost:** Hyperparameters tuned for learning rate, max depth, and n_estimators.

5.2. Performance Metrics Comparison Report

Model	Accuracy	Precision	Recall	F1-Score
Naive Bayes	35.79%	0.00	0.00	0.00
Random Forest	35.79	0.00	0.00	0.00
Logistic Regression CV	81.58%	91.80	79.43%	86.49
Ridge Classifier	84.21%	93.44	83.82	88.37
Support Vector Classifier	35.79%	0.00	0.00	0.00
Logistic Regression	79.47%	91.80	79.43	85.58
KNN	86.32%	94.26	85.82	89.84
XG Boost	35.79%	3.28	50.00	6.15

5.3. Final Model Selection Justification

The K-Nearest Neighbors (KNN) algorithm achieved the highest overall performance with 86.32% accuracy and strong precision-recall balance, making it the selected model for production deployment.

But Using K-Nearest Neighbors (KNN) for your liver cirrhosis prediction project can be an option, but it is *not the best choice* in this case — and here's why:

Why KNN Is Not Ideal for OUR Dataset

- High Dimensionality - You have 40 features. KNN suffers in high-dimensional spaces due to the curse of dimensionality, which weakens distance-based decisions.
- Class Imbalance - We have very few negative samples (NO = 5 in test set). KNN tends to be biased toward the majority class unless specifically tuned.
- Performance - KNN is slow at prediction time since it needs to compute distances from all training points. Not ideal for real-time predictions.
- Feature Scaling Needed - KNN requires normalized/scaled data (e.g., StandardScaler or MinMaxScaler) for meaningful distance calculation.

Better Alternatives for our Dataset:

Given our medical prediction context with imbalanced data:

- Random Forest – handles imbalance & noise well
- XGBoost – powerful and accurate
- Logistic Regression (with class_weight='balanced') – interpretable
- SVM (with class_weight='balanced') – robust to imbalance

The **Random Forest** algorithm achieved the second highest overall performance with 43.16% accuracy and strong precision-recall balance, making it the selected model for production deployment.

6. Results

6.1. **Output Screenshots**

The source code and output screenshots are available in the accompanying files.

7. Advantages & Disadvantages

- **Advantages:** High accuracy, effective at handling local data variations, robust performance.
- **Disadvantages:** Can be computationally intensive, requires careful tuning.

8. Conclusion

The project successfully developed a machine learning model to predict liver cirrhosis with high accuracy. The Random Forest model, after hyperparameter tuning, provided the best results and was chosen for its robustness.

9. Future Scope

- Further data collection to include more features and increase dataset size.
- Exploration of additional features and engineering techniques.
- Experimentation with deep learning models to potentially outperform traditional models.
- Integration with a real-time prediction system for practical deployment.

10. Appendix

10.1. Source Code

Code File: Predicting_Liver_Currhosis.ipynb

10.2. GitHub & Project Demo Link

GitHub Repository: https://github.com/harshinimallela/Predicting-Liver-Cirrhosis-Using-Advanced-Machine-Learning-Techniques

Project Demo Link: https://predicting-liver-cirrhosis.netlify.app/