

A Machine Learning-Data Driven Analysis for Non Alcoholic Fatty Liver Disease

A PROJECT REPORT
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD
OF THE DEGREE

OF
MASTER OF TECHNOLOGY IN INDUSTRIAL BIOTECHNOLOGY

Submitted by Jitika Gupta 24/IBT/02

Under the supervision of Dr. Yasha Hasija

BIOTECHNOLOGY
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi 110042
May 2026



CANDIDATE'S DECLARATION

I, Jitika Gupta (Roll No. 24/IBT/02), a student of Master of Technology (Industrial Biotechnology), hereby declare that the Project Dissertation titled “**A Machine Learning-Data Driven Analysis for Non-Alcoholic Fatty Liver Disease**”, submitted to the Department of Biotechnology, Delhi Technological University, Delhi, in partial fulfilment of the requirements for the award of the degree of Master of Technology, is original and has not been copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma, Associateship, Fellowship, or any other similar title or recognition.

Place: Delhi
Gupta
Date: 27.05.2026

Jitika

CERTIFICATE

I hereby certify that the Project Dissertation titled “**A Machine Learning-Data Driven Analysis for Non-Alcoholic Fatty Liver Disease**”, submitted by Jitika Gupta, Roll No. 24/IBT/02, Department of Biotechnology, Delhi Technological University, Delhi, in partial fulfilment of the requirements for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge, this work has not been submitted, in part or full, for any degree or diploma to this University or elsewhere.

Place: Delhi

Prof. Yasha Hasija

Date: 27.05.2026

SUPERVISOR

ACKNOWLEDGEMENT

I would like to show my strong appreciation to Prof. Yasha Hasija who has always guided and mentored me during this project. She not only made every work clear to me but also made me see the real-life usefulness and industrial applicability of the work. Her readiness to discuss my concerns, help me out of the difficulties, and motivate me at each point really helped. Without her strong support and encouragement, this project would not have been completed.

I am also grateful and thankful to all my lab mates PhD's for their continuous assistance during my practical work.

I would like to extend my thanks to Mr C B Sinha and Mr. Jitendra Singh and other non-teaching staff members for supporting me all through the project tenure.

Finally, I am thankful to my family friends for continuous encouragement throughout the process of project and the accomplishment would have not been possible without them.

Place: Delhi
Date: 27.05.2026

Jitika Gupta

TABLE OF CONTENTS

Title Page	1
Candidate's Declaration	2
Certificate	3
Acknowledgement	4
Table of Contents	5
Abstract	6
List of Graphs	7
List of Figures	8
List of Abbreviations	9
Chapter 1: Introduction	11
1.1 Objectives	12
Chapter 2: Review of Literature	13
2.1 NAFLD: Overview	13
2.2 Machine Learning	13
2.3 Supervised Learning	13
2.4 Unsupervised Learning	13
2.5 Reinforcement Learning	13
2.6 Application of ML in Disease Diagnosis	13
Chapter 3: Research Methodologies	14
3.1 Data Acquisition and Datasets	14
3.2 Data Preprocessing and Cleaning	15
3.2.1 Handling Missing Values and Outliers	15
3.2.2 Data Transformation and Scaling	15
3.3 Feature Engineering	16
3.3.1 Correlation Analysis	16

3.4 Data Exploration	18
3.4.1 Liver Marker Feature Distribution	18
3.4.2 Metabolic Feature Distribution	19
3.4.3 Correlation of Metabolic and Liver Marker	20
3.4.4 New Feature Extraction	21
3.4.5 Comparison of Metabolic Feature vs Fibrosis and NASH/NAFL	22
3.5 Machine Learning Modelling and Training	23
3.5.1 Label Encoding	23
3.5.2 Feature Selection	23
3.5.3 Train Test Split	24
3.5.4 Feature Scaling	24
3.6 Algorithms Employed	25
3.6.1 Logistic Regression	25
3.6.2 Random Forest	26
3.6.3 Gaussian Bayes	28
3.6.4 XGBoost	29
3.6.5 KNN	30
3.6.6 SVM	31
3.7 Performance Evaluation	31
3.7.1 Statistical Tools for Evaluation and Comparison	31
Chapter 4: Results and Discussion	32
4.1 NASH vs NAFL Model Comparison	32
4.2 Fibrosis Model Comparison	33
Chapter 5: Conclusion	34
Chapter 6: Future Scope	35
References	39

Abstract

Nonalcoholic fatty liver disease (NAFLD), and its progressiveness of disease, such as nonalcoholic steatohepatitis (NASH) and fibrosis, show worldwide challenge, solely due to the difficulty in achieving early diagnosis given the limited initial symptoms. This research explores the data-driven machine learning (ML) to enhance the accurateness of classification, prediction in fatty liver disease.[1]

Workflow frequently uses datasets such as the Indian Liver Patient Dataset (ILPD), electronic health records (EHR), and multi-omics cohorts (transcriptomics and lipid omics)[2]. Data preprocessing is a initial step, often involving multivariate imputation or iterative imputation for missing values, applying transformations to correct skewedness,[3]. To optimize the model efficiency and performance, feature extraction and selection methods such as the spearman coefficient correlation was applied not linear data and Pearson coefficient to liner data.[4]

Stacking Ensemble Learners and Gradient Boosting algorithms (XGBoost), consistently exhibit higher performance in prediction tasks. But for primary level 6 basic algorithms were used for prognostics.[5], [6] Detection of stages of fibrosis and differentiation between NASH and NAFL was to find disease in early stage without invasive process. Key clinical factors identified as having significant predictive value for progression

include prior mean and maximum AST/ALT ratio , FIB4, and metabolic and liver marker levels, and prior mean and standard deviation[7] . Functional enrichment analysis also emphasized the significant accumulation of fat in liver, scarring and then converting it into cancer[8] . Collectively, these data-driven ML approaches offer efficient means to improve early detection, refine risk stratification, and elucidate the underlying pathogenesis of fatty liver disease.[9]

List of Graphs

Sl.No.	Name of Graph
Graph.1	Fibrosis : Correlation
Graph.1.2	NAFL: Correlation
Graph.2	Liver Marker Feature Distribustion
Graph.3	Metabolic Feature Distribustion
Graph.4	Liver and Metabolic Feature Correlation
Graph.5	Metabolic Risk Score vs Fibrosis and NAFL
Graph.6	Comparison between models for NASH vs NAFL
Graph.7	Comparison between models for Fibrosis

List Of Figures

Sl.no	Name
Fig 1	Describing Datast
Fig 2	Dropping Unnecessary columns
Fig 3	Replacing Missing Value : Iterative Imputation
Fig 4	Outlier Treatment and Standard Scaling of Numerical Column
Fig 5.1	Metabolic Features
Fig 5.2	Liver Markers
Fig 5.3	Target Features
Fig 5.4	Secondary Features
Fig 6	Feature Extraction
Fig 7	Label Encoding
Fig 8	Feature Selection
Fig 9	Importing Model and other necessary modules

List Of Abbreviations

Abbreviation	Full Form
NAFLD	Non-Alcoholic Fatty Liver Disease
NAFL	Non-Alcoholic Fatty Liver
NASH	Non-Alcoholic Steatohepatitis
BMI	Body Mass Index
HOMA	Homeostatic Model Assessment
HOMA-IR	Homeostatic Model Assessment of Insulin Resistance
ALT	Alanine Aminotransferase
AST	Aspartate Aminotransferase
GGT	Gamma-Glutamyl Transferase
ALP	Alkaline Phosphatase
LDH	Lactate Dehydrogenase
APRI	AST to Platelet Ratio Index
FIB4	Fibrosis-4 Index
HDL	High-Density Lipoprotein
LDL	Low-Density Lipoprotein
VLDL	Very Low-Density Lipoprotein
TG	Triglycerides
TC	Total Cholesterol
ROC	Receiver Operating Characteristic
AUC	Area Under the Curve
SVM	Support Vector Machine
KNN	K-Nearest Neighbours
XGBoost	Extreme Gradient Boosting

Abbreviation	Full Form
CK	Creatine Kinase
PT	Prothrombin Time
INR	International Normalized Ratio
HbA1c	Haemoglobin A1c
ML	Machine Learning
EDA	Exploratory Data Analysis
AST/ALT Ratio	Aspartate Aminotransferase to Alanine Aminotransferase Ratio
SBP	Systolic Blood Pressure
DBP	Diastolic Blood Pressure
VD	Vitamin D

Chapter 1

Introduction

Nonalcoholic fatty liver disease (NAFLD), which larger range of hepatic disorders from simple steatosis, NAFL to the advanced NASH to Cirrhosis. IT represents the most prevalent form of; chronic liver disease worldwide. It constitutes a significant global public health concern. Globally, NAFLD is strongly linked to the widespread of obesity and diabetes mellitus .[1] Early diagnosis and effective treatment of NAFLD are severely complicated by the disease's limited symptoms, many times leading to detection only at late stages, which may risks progression to advanced conditions like cirrhosis, Liver cancer, and liver failure.[10] For example, studies says that liver-related deaths account for 3.5% of all deaths in this century, and all over world one million people are diagnosed with liver disease annually.[9], [11]

The pathophysiology of NAFLD is complex and connect to each other like web, involving an interplay between genetic, metabolic, and environmental factors, such as excessive drug use, alcohol, obesity, and diabetes[12]. The initial stages are categorized by the accumulation of triglycerides in Liver cells called steatosis, which may progress to inflammation, fibrosis, and cellular death indicative of NASH.[13] Key pathogenic mechanisms driving progression include insulin resistance, lipotoxicity, and changes in the gut-liver dynamics. Accurate prediction of disease severity or progression are important and, yet traditional diagnostic methods like liver biopsy are invasive and costly. Which can be prevent by solving using liver factors and other available factors. Can also be found out by normal blood tests.[14]

In response to these challenges, the field of data-driven analysis and machine learning (ML) has emerged as a powerful tool to assist healthcare professionals in the diagnostic and prognostic processes for liver diseases. ML models utilize statistical algos to analyze large, complex datasets, identifying hidden patterns and improving the speed and accuracy of disease classification and prediction[15], [16]. The standard methodology in such approaches often involves steps, including data preprocessing , feature selection/extraction , and the application of powerful classification ML models.[17]

Application of these data-driven ML and ensemble learning techniques, datasets to improve the classification and prediction of NAFL[18]. In particular, studies show that models KNN and RF are highly approachable in classifying liver disease accurately and predicting the progression of NAFL to severe conditions such as NASH or fibrosis. This research aims to overlook that ML can be used as clinical tool which can differentiate in NAFL , for further other data science methods can be used.[19]

Objectives

- Relationship between liver and metabolic factors
- Build a classifiers ML model; (e.g., Logistic Regression; Random Forest, XGBoost) to predict NAFLD in early stage.
- Use publicly available datasets for training and validation.
- Work with prime alary for basic understanding
- Prediction of different stages of fibrosis

Chapter 2

Review of Literature

2.1. NAFLD: Overview

NAFLD, with its advance form NASH is worldwide problem. Deaths increases 3.5% every year due to this Finding out early is tough because it does not gives any visible symptoms[20].And late detection , cannot be reversed. Early detection is reversible too. To diagnose this invasive methods are availed only[21].So ML is used to analyses data available to find patterns from records .[11]

2.2 Machine Learning

So, to apply ML we have handle missing data, imbalanced data, remove or replace missing data. Transform data from uneven distribution to normalized one.RF,SVM,LR,KNN, etc. model are used to apply on data[22]. Efficiency is measured by precision recall, f1 score and accuracy. Multiple features like AST<ALT<GGT etc. correlation is seen that how they affect each other and to the target feature. Bias is seen in data, smote is used to remove biasness. Advance method like Principal Component Analysis (PCA), Factor Analysis (FA), Linear Discriminant Analysis (LDA), and Recursive Feature Elimination (RFE) help to find most relevant features.[23]

2.3 Supervised Learning

ML model uses data given which is labeled. Both input feature and output feature is given to model. Then it works on it and understand its pattern and how it relates to each other. After split, training of model is done. Works on test data to find output from the learnt pattern. This is highly used in medical, biotech fields, Some examples are XGBoost, GNB,KNN,SVM.[24]

2.4 Unsupervised Learning

ML model works with no output given and. And main thinking behind this is to understand the concealed pattern, trends. Techniques like clustering, dimensionality reduction is used in this ML model. In medical field it can be used to group patients according to their features. Exploration of data is major part of this model. [25]

2.5 Reinforcement Learning

RL is type of ML, works like give and take, it tries then find error then learn.IT take feedback on its action. This feedback is as rewards or penalties. There is an agent, environment where it works, condition, action and reward. It can also be done online and offline as well. Some drawbacks like need large data to understand patterns. Reward and feedback function gets complex.[26]

2.6 Application of ML in Disease Diagnosis

Early detection, diagnostic accuracy, personalized treatment. Working with complex data , analyzing and understanding it.Using statistical techniques and mathematical approach makes more efficient[18].Core idea is to assist doctor , medical field people to fast diagnosis Delay in understanding the diseases, formation of medicine

base on need. Separation of Nash and NAFL, category of Fibrosis. Revealing important marker to find theses result.[27], [28]

Chapter 3

Research Methodologies

The research methodology understands data related approach for the analysis, classification, and prediction of fatty liver disease. The overall process typically encompasses data collection and describing, preprocessing, feature engineering, applying of machine learning models, and standardized evaluation.

3.1. Data Acquisition and Datasets

Data was taken from Kaggle. Number of columns 62 features and rows 602 records. Metabolic feature 14 and liver related 6 and target 2 columns NASH vs NAFL and Fibrosis.[29]

```
df = pd.read_excel("/Users/jitikagupta/Desktop/NAFLD/NAFLD.xlsx")
df.head()
```

	# Patient No.	# Age	# Gender (Female=1, Male...	# Height	# Weight	# Body Mass Index	# Waist Cir
0	1	60	1	150	80	35.56	
1	2	55	1	154	73	30.78	
2	3	53	2	170	101	34.95	
3	4	33	2	175	95	31.02	
4	5	23	2	162	68	25.91	

5 rows x 62 cols | 10 per page | Page 1 of 1

```
def basic_info(df):
    return {
        "rows": len(df),
        "columns": list(df.columns)
    }

def missing_values(df):
    return df.isnull().sum()

def summary_stats(df):
    return df.describe()

def correlation(df):
    return df.corr(numeric_only=True)

def outliers(df):
    outlier_dict = {}

    for col in df.select_dtypes(include='number').columns:
        q1 = df[col].quantile(0.25)
        q3 = df[col].quantile(0.75)
        iqr = q3 - q1

        outliers = ((df[col] < (q1 - 1.5 * iqr)) |
                    (df[col] > (q3 + 1.5 * iqr))).sum()

        outlier_dict[col] = int(outliers)

    return outlier_dict
```

Fig 1: Describing Dataset

3.2. Data Preprocessing and Cleaning

```
Dropping columns to prevent data leakage and unnecessary cols.

df = df.drop('Patient No.', axis=1)
df=df.drop('Advanced Fibrosis', axis=1)
df=df.drop('Significant Fibrosis', axis=1)
df=df.drop('Fibrosis status', axis=1)
df=df.drop(['Microalbumin/Creatinine Ratio'], axis=1, inplace=True)
df=df.drop('NAS score==4 and Fibrosis==2', axis=1, inplace=True)
df=df.drop('NAS score according to Kleiner', axis=1, inplace=True)
df=df.drop('Cirrhosis', axis=1, inplace=True)
```

Fig 2: Dropping Unnecessary columns

3.2.1 Handling Missing Values and Outliers

Missing data are handled using iterative imputation techniques, found in the Sklearn library, which guess missing values using other features in the dataset. Alternatively, missing observations may be replaced by the median or mean value. Outlier detection method Inter-Quartile Range method, are applied.[30]

```
import pandas as pd
from sklearn.experimental import enable_iterative_imputer # needed
from sklearn.impute import IterativeImputer

def iterative_imputation(df):
    num_cols = df.select_dtypes(include='number').columns

    imputer = IterativeImputer(random_state=0)
    df[num_cols] = imputer.fit_transform(df[num_cols])

    return df
```

Fig 3: Replacing Missing Value: Iterative Imputation

3.2.2 Data Transformation and Scaling

For data features showing skewed distributions, the log1p transformation is applied to achieve a more balanced curve distribution. To standardize features independently and prevent models from being bend over toward attributes with larger numerical ranges, scaling techniques are employed. StandardScaler(0 mean and 1std) was applied to prevent from biasing the ML for all numerical values.[31]

```

# OUTLIER TREATMENT (IQR CAPPING)

def outlier_capping(df, columns):

    for col in columns:

        Q1 = df[col].quantile(0.25)
        Q3 = df[col].quantile(0.75)

        IQR = Q3 - Q1

        lower_bound = Q1 - 1.5 * IQR
        upper_bound = Q3 + 1.5 * IQR

        # capping
        df[col] = np.where(
            df[col] < lower_bound,
            lower_bound,
            df[col]

        df[col] = np.where(
            df[col] > upper_bound,
            upper_bound,
            df[col]

        )

    return df

# apply outlier treatment
df = outlier_capping(df, num_cols)

```

Fig 4. Outlier Treatment

```

# STANDARDIZATION
scaler = StandardScaler()

df[num_cols] = scaler.fit_transform(df[num_cols])

# CHECK RESULT
print(df[num_cols].describe())

```

	Age	Body Mass Index	Waist Circumference	Hip Circumference
count	6.050000e+02	6.050000e+02	6.050000e+02	6.050000e+02
mean	-3.347185e-16	2.701237e-16	-1.027644e-15	-9.982832e-17
std	1.000827e+00	1.000827e+00	1.000827e+00	1.000827e+00
min	-2.060671e+00	-2.549812e+00	-2.744796e+00	-2.430650e+00
25%	-7.805029e-01	-6.668167e-01	-7.370191e-01	-6.208781e-01
50%	6.557281e-02	-1.145563e-01	-1.192414e-01	-1.762087e-02
75%	7.236317e-01	5.885137e-01	6.014992e-01	5.856364e-01
max	2.321775e+00	2.471509e+00	2.609276e+00	2.395408e+00

	Systolic Blood Pressure	Diastolic Blood Pressure	ALT
count	6.050000e+02	6.050000e+02	6.050000e+02
mean	2.671876e-16	-3.082933e-16	1.585509e-16
std	1.000827e+00	1.000827e+00	1.000827e+00
min	-2.114593e+00	-1.883737e+00	-1.499265e+00
25%	-4.888329e-01	-2.578443e-01	-7.507103e-01
50%	-4.165769e-01	-2.578443e-01	-1.953308e-01
75%	5.950075e-01	8.274175e-01	5.532243e-01
max	2.220768e+00	2.454110e+00	2.509126e+00

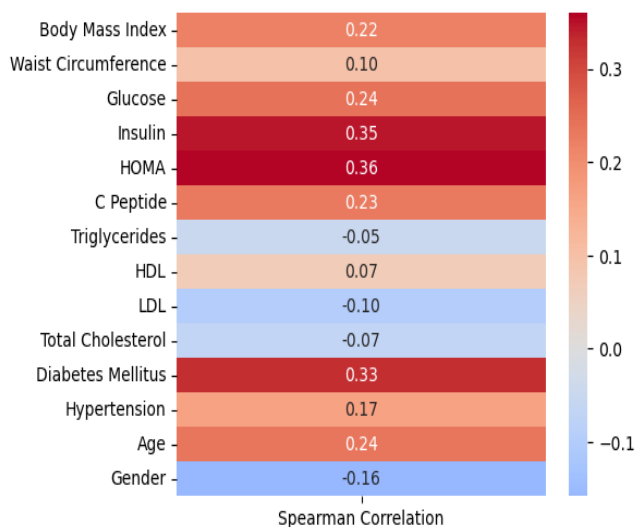
	AST	GGT	LDH	...	Glucose
count	6.050000e+02	6.050000e+02	6.050000e+02	...	6.050000e+02
mean	2.172734e-16	2.290179e-16	-1.879121e-16	...	-3.053572e-16
std	1.000827e+00	1.000827e+00	1.000827e+00	...	1.000827e+00
min	-1.720365e+00	-1.402030e+00	-2.563641e+00	...	-1.905263e+00
25%	-7.416134e-01	-7.493794e-01	-7.143336e-01	...	-6.662240e-01
50%	-1.983502e-01	-3.136788e-01	-3.675885e-01	...	-2.849812e-01
75%	4.930756e-01	4.487972e-01	5.185380e-01	...	5.251596e-01
max	2.345109e+00	2.246062e+00	2.367845e+00	...	2.312235e+00

Fig 4. Standard Scaling of Numerical Column

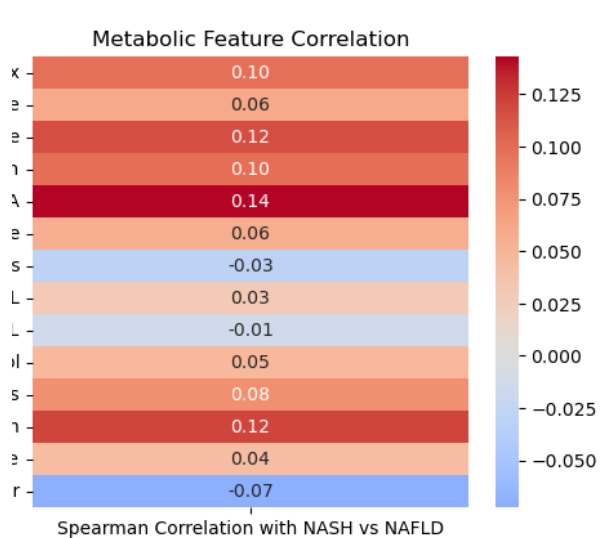
3.3. Feature Engineering.

3.3.1 Correlation Analysis

The heatmap represents the Spearman correlation coefficients between metabolic parameters and fibrosis and NaFL vs NASH outcomes. Positively correlated with Fibrosis: Waist Circumference, ALT, AST, GGT, HOMA, Insulin, C Peptide, BMI .Negatively correlated: Platelet Count, Albumin.[32]



Graph: 1 Fibrosis: Correlation



Graph: 1.1 NAFL: Correlation

Divide Data frame into different category to understand feature that help in disease progression.

```
metabolic_features = df[[  
    'Body Mass Index',  
    'Waist Circumference',  
  
    'Glucose',  
    'Insulin',  
    'HOMA',  
    'C Peptide',  
  
    'Triglycerides',  
    'HDL',  
    'LDL',  
    'Total Cholesterol',  
  
    'Diabetes Mellitus',  
    'Hypertension',  
  
    'Age',  
    'Gender'  
]]
```

Fig 5.1 Metabolic Feature

```
liver_markers = df[[  
  
    'ALT',  
    'AST',  
    'GGT',  
    'ALP',  
  
    'Albumin',  
  
    'Trombositol'  
]]
```

Fig 5.2 Liver Marker Feature

```
target_features = df[[  
    'Fibrosis',  
  
    'Steatosis',  
    'Activity',  
  
    'Nash(1) vs NAFL(2)',  
  
    'Type of Disease'  
]]
```

Fig. 5.3 Target Feature: Fibrosis & NAFL

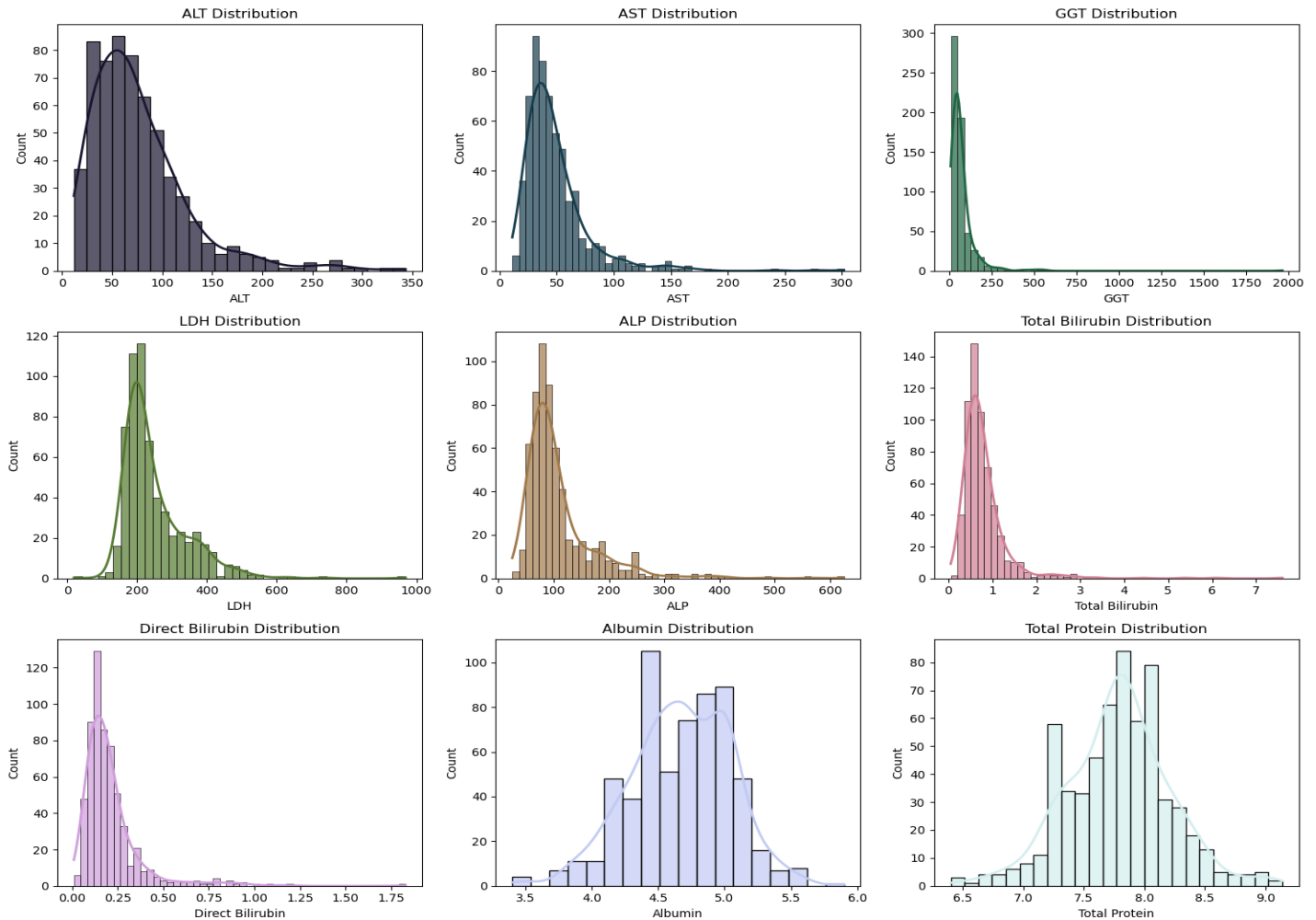
```
secondary_features = df[[  
  
    'Vitamin D',  
  
    'Total Bilirubin',  
    'Direct Bilirubin',  
  
    'BUN',  
    'Creatinine'  
]]
```

Fig. 5.4 Secondary Features

3.4 Data Exploration

3.4.1 Liver marker feature distribution

AST/ALT Ratio increases with fibrosis stage; ratio > 1 indicates advanced fibrosis increases in NASH vs NAFL patients. Platelet Count inversely proportional with fibrosis; hypertension effect. Albumin lower in advanced fibrosis/cirrhosis.



Graph.2 Liver Marker Feature Distribution

3.4.2 Metabolic feature Distribution

Patients with higher Metabolic Risk Scores (4-5) showed higher fibrosis stages. Insulin resistance markers (HOMA, Insulin, C Peptide) correlate with disease severity. Waist circumference is more predictive than BMI alone.

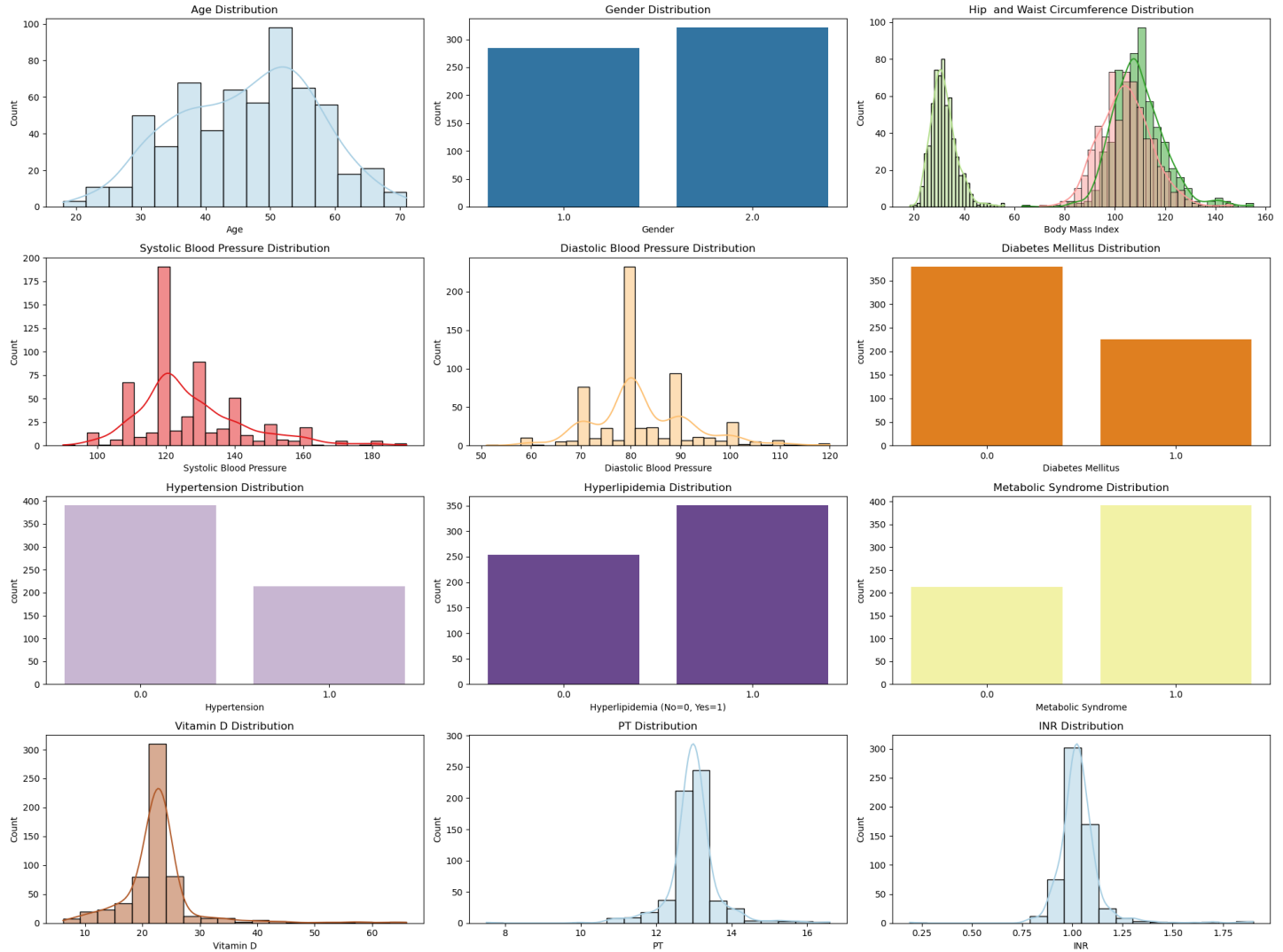
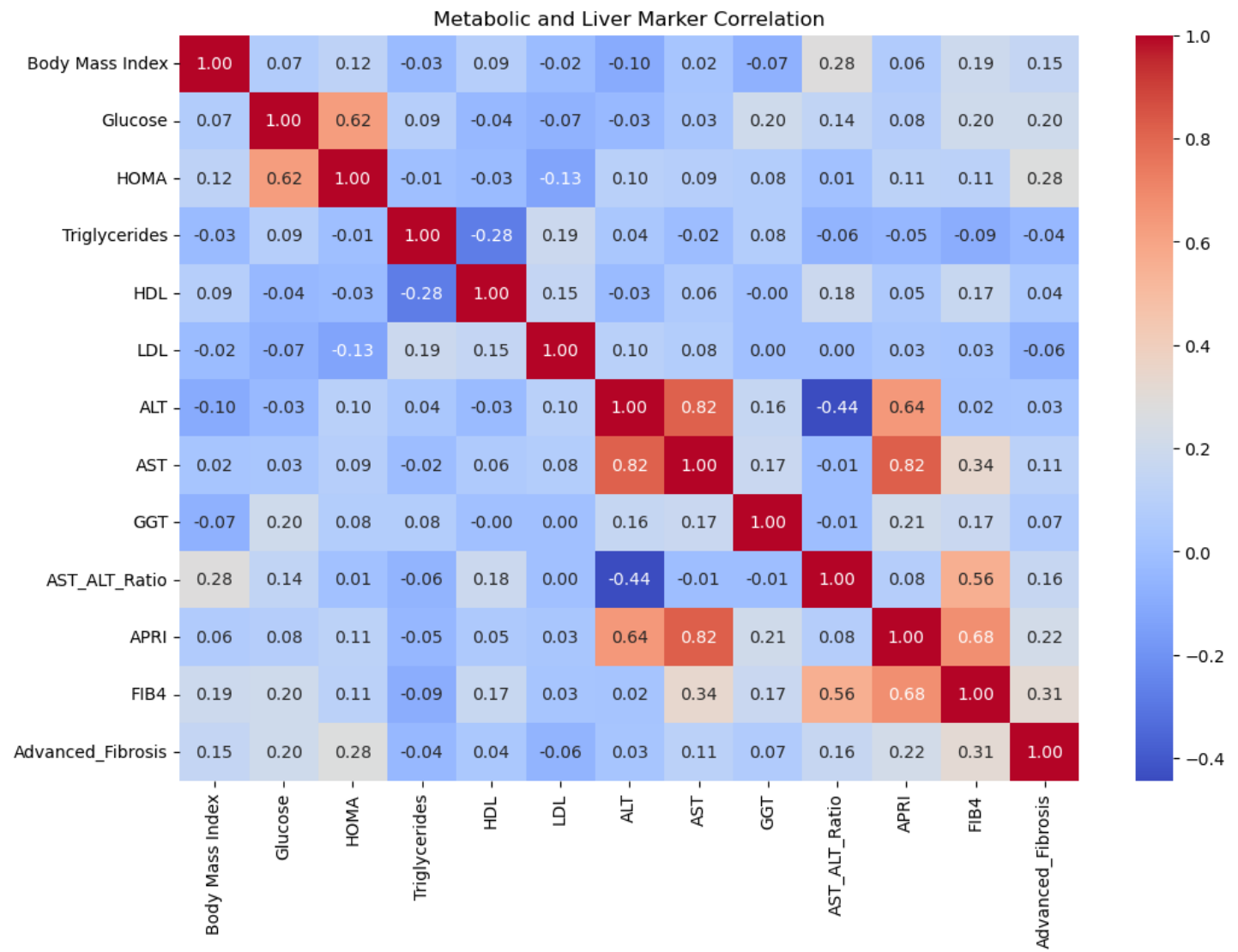


Fig.3 Metabolic Feature Distribution

3.4.3 Correlation of metabolic and liver marker

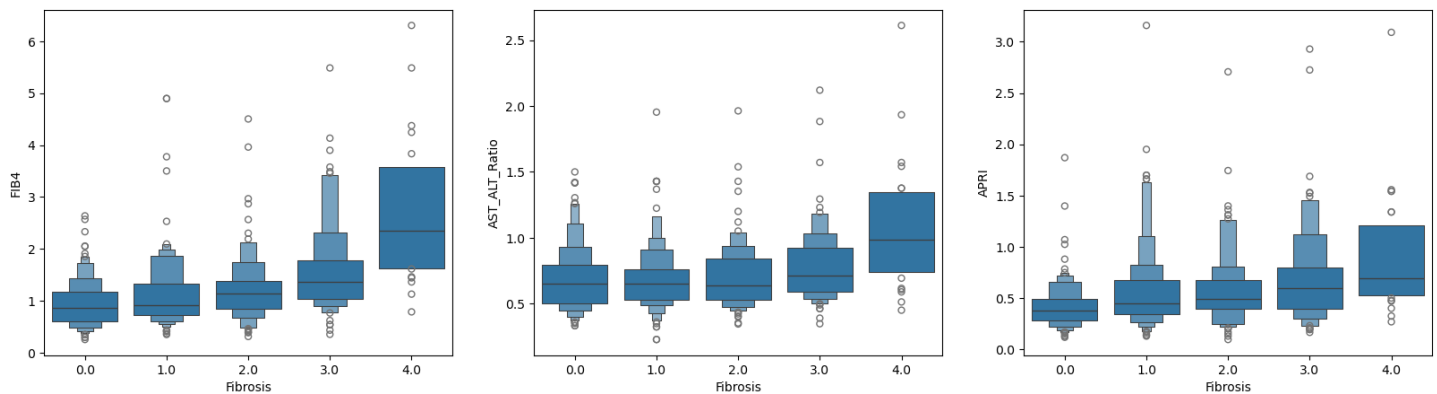
The correlation heatmap illustrates relationships among metabolic markers, liver enzymes, fibrosis stages, and NAFL. Strong positive correlations were observed between AST and ALT, APRI and FIB4, while advanced fibrosis showed moderate correlation with FIB4 and HOMA.



Graph.4 Metabolic and Liver marker Correlation

Derived Feature Analysis:

FIB-4: Higher values in Fibrosis stages 3-4. < 1.3. APRI: Better for cirrhosis detection than intermediate stages. AST/ALT Ratio: Consistent upward across fibrosis stages.



Graph 5: Derived Feature Analysis

3.4.4 New Feature Extraction

This was done in reference to several research paper.

```
# 6. METABOLIC RISK SCORE

# example thresholds
df['High_TG'] = (
    df['Triglycerides'] > 150
).astype(int)

df['Low_HDL'] = (
    df['HDL'] < 40
).astype(int)

df['High_Glucose'] = (
    df['Glucose'] > 100
).astype(int)

df['Obesity'] = (
    df['Body Mass Index'] > 30
).astype(int)

# risk score
df['Metabolic_Risk_Score'] = (
    df['High_TG'] +
    df['Low_HDL'] +
    df['High_Glucose'] +
    df['Obesity'] +
    df['Hypertension']
)
print

df['AST_ALT_Ratio'] = df['AST'] / df['ALT']
df['APRI'] = (
    (df['AST'] / 40) /
    df['Trombosit']
) * 100
df['FIB4'] = (
    (df['Age'] * df['AST']) /
    (df['Trombosit'] * np.sqrt(df['ALT']))
)
```

Fig 6: New Feature Extraction

3.4.5 Comparison of Metabolic Feature vs Fibrosis and NASH N NAFL

The distribution plot demonstrates the relationship between metabolic risk score and fibrosis stages. Higher metabolic risk scores were more frequently associated with fibrosis progression and advanced fibrosis.[33] The metabolic risk score analysis shows that higher metabolic scores were associated with increased fibrosis severity and a greater prevalence of NASH. This indicates that accumulation of metabolic abnormalities contributes to disease progression in NAFLD patients.[34]

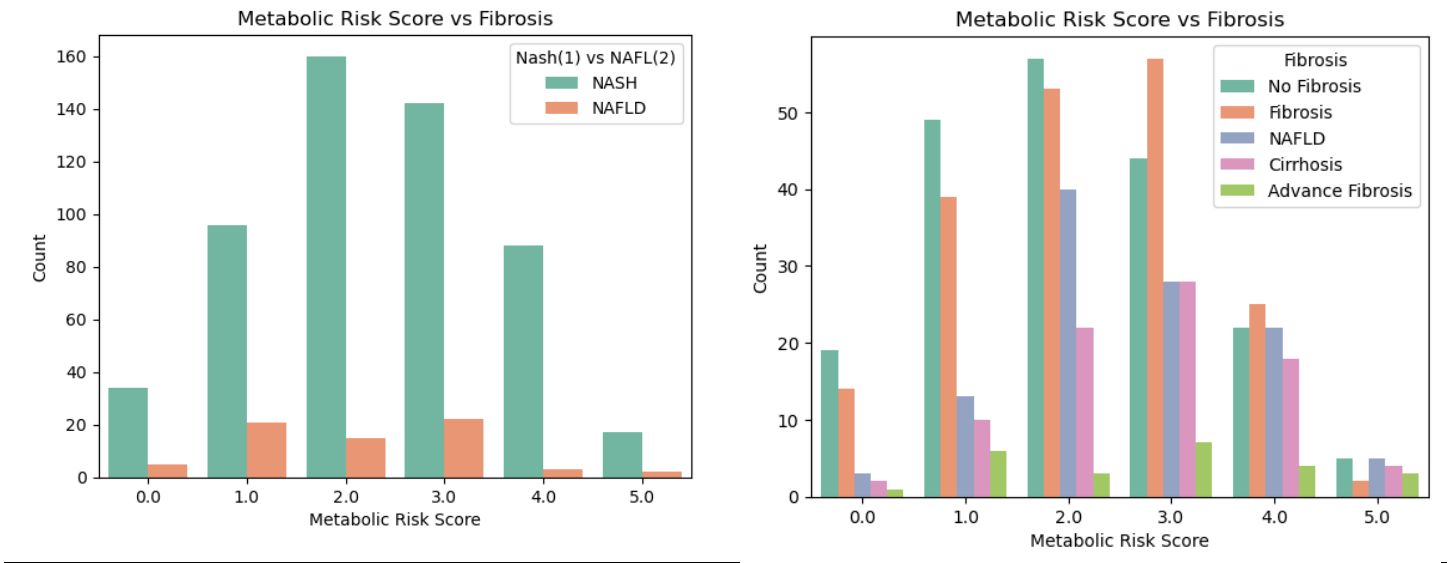


Fig.5 Metabolic Risk Score vs Fibrosis and NAFL

3.5 Machine Learning Modeling and Training

3.5.1 Label Encoding

This step was done because model understands binary data. So, categorical data was change to binary data.

```
#Label encoding for nash(1) vs nash(2) and type of disease
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
df[['Nash(1) vs NAFL(2)', 'Type of Disease']] = df[['Nash(1) vs NAFL(2)', 'Type of Disease']].apply(LabelEncoder().fit_transform)
print(f"Encoded Data: {encoded_data}")

Encoded Data: [[ 0. 26.]
 [ 0.  0.]
 [ 1. 26.]
 ...
 ...
 [ 0.  0.]
 [ 0.  0.]
 [ 1. 30.]]
```

Fig 7: Label Encoding

3.5.2 Feature Selection

On the basis of correlation and multiple research paper these variables where take under consideration to work with. This helps model to work effectively on feature to find efficient target. Age, Gender, BMI, Waist

Circumference, Glucose, Hip Circumference, Insulin,Homa, TG, HDL, LDL, Cholesterol, Diabetes Mellitus,Hypertension, Metabolic Syndrome, ALT, AST, GGT, ALP, LDH, Albumin, Total Bilirubin, Leukocyte, Hemoglobin, Trombosit, AST_ALT_Ratio, APRI, FIB4[33]

```
# FEATURE SELECTION
# Liver + Metabolic + Clinical Score
features = [
    'Age',
    'Gender',
    'Body Mass Index',
    'Waist Circumference',
    'Hip Circumference',
    'Glucose',
    'Insulin',
    'HOMA',
    'Triglycerides',
    'HDL',
    'LDL',
    'Total Cholesterol',
    'Diabetes Mellitus',
    'Hypertension',
    'Metabolic Syndrome',
    'GGT',
    'ALP',
    'LDH',
    'Albumin',
    'Total Bilirubin',
    'Leukocyte',
    'Hemoglobin',
    'Trombosit',
    'AST_ALT_Ratio',
    'APRI',
    'FIB4'
]

# FEATURES AND TARGET
x = df[features]
y = df['Fibrosis']
```

Fig 8: Feature Selection

3.5.3 Train Test Split

The prepared dataset is split into training and testing sets 80% for training and 20% for testing.

3.5.4 Feature Scaling

So, data have different range like BMI ranges from 22 to 30 where Glucose goes from 100 or more than 100. To maintain every data in one level this scaling, model treats every feature with enough and equal importance for learning.

3.6 Algorithms Employed

The machine learning workflow was maintained using Python programming language. Essential libraries were implemented like scikitlearn, was imported to for data preprocessing, model deployment and performance evaluation. The test_train_split function was used to divide dataset into training sub dataset and test sub dataset, while StrandardScaler module was applied to normalize numerical features. Machine learning classifiers including Logistic Regression, Random Forest, Gaussian Naïve, KNN, Support Vector Machine and XGBoost were also imported to compare their predictive capabilities for NAFLD classification. In addition, Python warning messages were suppressed to ensure cleaner program execution.

Below image shows importing necessary model and other library.

```
from sklearn.model_selection import train_test_split, cross_val_score, GridSearchCV
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import (accuracy_score, precision_score, recall_score,
                             f1_score, roc_auc_score, confusion_matrix,
                             classification_report, roc_curve)

# Import all models
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.neural_network import MLPClassifier
from xgboost import XGBClassifier

import warnings
warnings.filterwarnings('ignore')
```

Fig 9: Illustrates the python libraries and modules imported for data preprocessing, model implementations, and evaluation. These libraries provide necessary function for training, testing, and, comparing multiple ML algos.

3.6.1 Logistic Regression

LR is a tool used to how different features effect occurrence of output. It is similar to other model Like Linear Regression, but with made for especially for situation where result is one or two outputs. Working of LR is; It uses probability and uses formula Logarithm of the chance, which includes categorical, numerical features and compare them. And to understand the result better it gives coefficient value. It can handle many different data without category them, it prevents statistical errors and also avoid masking effects.[35]

LR was implemented as base line classifier to distinguish between NAFL and NASH using clinical and metabolic factors. And also, did not give very impressive result for stage differentiation of fibrosis as it can only work with 1 or 2 outputs.

```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix, roc_auc_score
from sklearn.model_selection import train_test_split

y = df['Nash(1) vs NAFL(2)']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
model = LogisticRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
print(classification_report(y_test, y_pred))
print("Confusion Matrix:")
print(confusion_matrix(y_test, y_pred))

# compute ROC AUC appropriately for binary or multiclass target
y_prob = model.predict_proba(X_test)
if y.nunique() == 2:
    auc = roc_auc_score(y_test, y_prob[:, 1])
    print("ROC AUC Score:", auc)
else:
    # multiclass: binarize labels and compute multiclass AUC (one-vs-rest)
    from sklearn.preprocessing import label_binarize
    classes = model.classes_
    y_test_bin = label_binarize(y_test, classes=classes)
    auc = roc_auc_score(y_test_bin, y_prob, multi_class='ovr', average='macro')
    print("Multiclass ROC AUC Score (macro, ovr):", auc)
```

0.0s

	precision	recall	f1-score	support
0	0.00	0.00	0.00	13
1	0.89	1.00	0.94	108
accuracy			0.89	121
macro avg	0.45	0.50	0.47	121
weighted avg	0.80	0.89	0.84	121

Confusion Matrix:

```
..
Confusion Matrix:
[[ 0 13]
 [ 0 108]]
```

ROC AUC Score: 0.6816239316239316

Ln 6, Col 27 Spaces: 4 LF

3.6.2 Random Forest

RF is ML model use to sort data in categories or forecast numerical values. It is called forest because it uses decision trees and finds final result by combining result of all the decision trees. And gives more accurate result. So working is many decision trees, each decision tree work with random sub dataset then give result. This is called Bootstrap Aggregation. Result for classification is but most voted and for Regression average result is used. With this working it avoid overfitting, gives high accuracy and handles messy data well , and can also identify important factors . [36]

It classifies NASH vs NAFL and fibrosis stage differentiation based on metabolic and clinical factors. It was select due to is efficient working with non linear data and finding relationships. The model was trained and evaluated on prepared dataset, and its performance were compared.

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, classification_report

y = df['Fibrosis']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
rf_classifier = RandomForestClassifier(n_estimators=100, random_state=42)
rf_classifier.fit(X_train, y_train)

y_pred = rf_classifier.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
classification_rep = classification_report(y_test, y_pred)

print(f"Accuracy: {accuracy:.2f}")
print("\nClassification Report:\n", classification_rep)

sample = X_test.iloc[0:1]
prediction = rf_classifier.predict(sample)

sample_dict = sample.iloc[0].to_dict()
print(f"\nSample Passenger: {sample_dict}")
print(f"Predicted Fibrosis Stage : {'Stage 1' if prediction[0] == 1 else 'Stage 0'}")
```

✓ 0.1s

Accuracy: 0.44

Classification Report:

	precision	recall	f1-score	support
0.0	0.54	0.54	0.54	41
1.0	0.40	0.61	0.49	41
2.0	0.40	0.18	0.25	22
3.0	0.14	0.08	0.10	13
4.0	1.00	0.25	0.40	4
accuracy			0.44	121
macro avg	0.50	0.33	0.35	121
weighted avg	0.44	0.44	0.42	121

.

Ln 25, Col 1 Spaces: 4 LF

3.6.3 Gaussian Bayes

GaussianNB mathematical tool used to sort information into category. It based in old method called Bayes' Theorem, which uses probability to guess what might happen in future based on past. It treats every piece of information as completely independently. And uses Gaussian distribution to calculate probabilities. Scalable model, easy to use and understand and Fast efficient working.[37]

Applies probabilistic classification for predicting NAFL vs NASH and Fibrosis category. Model is trained using selected features after preprocessing and feature selection. It predicts performance was evaluated and compared with other machine learning models to determine suitability.

```
from sklearn.naive_bayes import GaussianNB
y = df['Fibrosis']
X_train, X_test, y_train, y_test = train_test_split(
    X,
    y,
    test_size=0.20,
    random_state=23
)
gnb = GaussianNB()
gnb.fit(X_train, y_train)
y_pred = gnb.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(
    y_test,
    y_pred,
    average='weighted'
)
recall = recall_score(
    y_test,
    y_pred,
    average='weighted'
)
f1 = f1_score(
    y_test,
    y_pred,
    average='weighted'
)
print("Accuracy:", accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("F1 Score:", f1)
print("\nConfusion Matrix\n")
print(confusion_matrix(y_test, y_pred))
```

✓ 0.0s

Accuracy: 0.39669421487603307
Precision: 0.38899668622988776
Recall: 0.39669421487603307
F1 Score: 0.3744571606325781

Confusion Matrix

0

Ln 25, Col 1 Spaces: 4 Spaces: 4 LF { }

3.6.4 XGBoost

XGBoost is an advanced ML model used to solve complex problems. It is an Ensemble Technique; means it relies on multiple models for result. It combines weak models to form a strong and accurate model. It works with an iterative process, which means it learns from its mistakes. Uses Similarity score to decide decision like adding new branch or node or removing any. Focuses on loss function to measure distance between result and guess. Exceptional high speed, Supreme accuracy and ability to handle missing values makes it one of the best models to use.[38]

Developed robust predictive model for differentiating NAFL from NASH and identifying Fibrosis stage. The algorithm was trained on pre-processed dataset containing metabolic features and liver markers to find complex relationships. It was evaluated using standard metrics from another model.

```
y = df['Fibrosis']
X_train, X_test, y_train, y_test = train_test_split(
    X,
    y,
    test_size=0.20,
    random_state=23
)

params = {
    'objective': 'multi:softprob',
    'num_class': 5,
    'max_depth': 4,
    'learning_rate': 0.1,
    'n_estimators': 100,
    'alpha': 10
}
model = XGBClassifier(**params)
model.fit(X_train, y_train)
# predictions
y_pred = model.predict(X_test)
# probabilities
y_prob = model.predict_proba(X_test)
# accuracy
accuracy = accuracy_score(y_test, y_pred)
print(f"XGBoost model accuracy: {accuracy:.2f}")
print(
    "\nClassification Report\n"
)
print(
    classification_report(y_test, y_pred)
)
print(
    "\nConfusion Matrix\n"
)
print(
    confusion_matrix(y_test, y_pred)
)
print(
    "\nROC-AUC Score\n"
)
print(
    roc_auc_score(

```

0.1s

Spaces: 4 LF { }

3.6.5 KNN

This method sorts information into categories. Also called as lazy learner, because it stores data simply then waits to see pattern. Then after finding info starts working.” K” represents number of nearest data points to look, measure distance from that point to other points and find shortest line between them.

Majority of voting decides the result. Advantage of using is simple, can solve both classification and regression problem and flexible[39]

Employed to classify patients into NAFL vs NASH and predict fibrosis stages based on the similarity between patients’ records. Algo was trained after feature scaling to ensure accurate distance calculated observation.

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
y = df['Fibrosis']

# =====
# TRAIN TEST SPLIT
# =====
X_train, X_test, y_train, y_test = train_test_split(
    X,
    y,
    test_size=0.20,
    random_state=23,
    stratify=y
)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
knn = KNeighborsClassifier(
    n_neighbors=5
)
knn.fit(X_train, y_train)
y_pred = knn.predict(X_test)
# multiclass probabilities
y_prob = knn.predict_proba(X_test)
print(
    "Accuracy:",
    accuracy_score(y_test, y_pred)
)
print("\nClassification Report\n")
print(
    classification_report(y_test, y_pred)
)
print("\nConfusion Matrix\n")
print(
    confusion_matrix(y_test, y_pred))
print("\nROC-AUC Score\n")
print(
    roc_auc_score(
        y_test,
        y_prob,
        multi_class='ovr',

```

3.6.6 SVM

Powerful computer tool used to sort data like predict or classify. It works with real world task. It draws line between dots and group them. It is a complex model which uses kernel in high dimensional space where boundary separate groups. According boundary, data points predict the results. Using this model is good because works with complex data, also great with small dataset and has strong mathematical background.[40] It was implemented to classify fatty liver disease by separating patients group based on their clinical n metabolic characters. The model was trained on the preprocessed dataset and evaluated. Any results were compared with those of the remaining machine learning model to determine its effectiveness.

```
from sklearn.svm import SVC
X = df[features]
y = df['Fibrosis']
X_train, X_test, y_train, y_test = train_test_split(
    X,
    y,
    test_size=0.2,
    random_state=42,
    stratify=y
)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
svm_model = SVC(
    kernel='rbf',
    C=1.0,
    probability=True,
    random_state=42
)
svm_model.fit(X_train, y_train)
y_pred = svm_model.predict(X_test)
y_prob = svm_model.predict_proba(X_test)
print("Accuracy:",
      accuracy_score(y_test, y_pred))
print("\nROC-AUC Score:")
print(roc_auc_score(
    y_test,
    y_prob,
    multi_class='ovr',
    average='macro'))
print("\nClassification Report\n")
print(
    classification_report(y_test, y_pred)
)
print("\nConfusion Matrix\n")
print(
    confusion_matrix(y_test, y_pred)
)
✓ 0.0s

Accuracy: 0.48760330578512395

ROC-AUC Score:
0
```

Spaces: 4 LF { }

3.7. Performance Evaluation

3.7.1. Statistical Tools for Evaluation and Comparison

- Performance Metrics: Standard statistical metrics are used to evaluate classifier performance:
 - Accuracy: Overall correctness of the model.
 - Precision: Describes the confidence in predicting positive class labels ($\text{True Positive} / (\text{True Positive} + \text{False Positive})$).
 - Recall (Sensitivity): Explains the proportion of actual positive cases predicted correctly ($\text{True Positive} / (\text{True Positive} + \text{False Negative})$).
 - F1-Score: The mean of P and R, useful for comparing model performance when there may be unequal precision and recall values.

Table 1: For NASH vs NAFL

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.89	0.80	0.89	0.84
Random Forest	0.98	0.98	0.98	0.97
Gaussian Bayes	0.80	0.86	0.80	0.83
XGBoost	0.92	0.84	0.92	0.88
KNN	0.98	0.98	0.98	0.98
SVM	0.88	0.78	0.88	0.83

Graph 6: Comparison between models for NASH vs NAFL

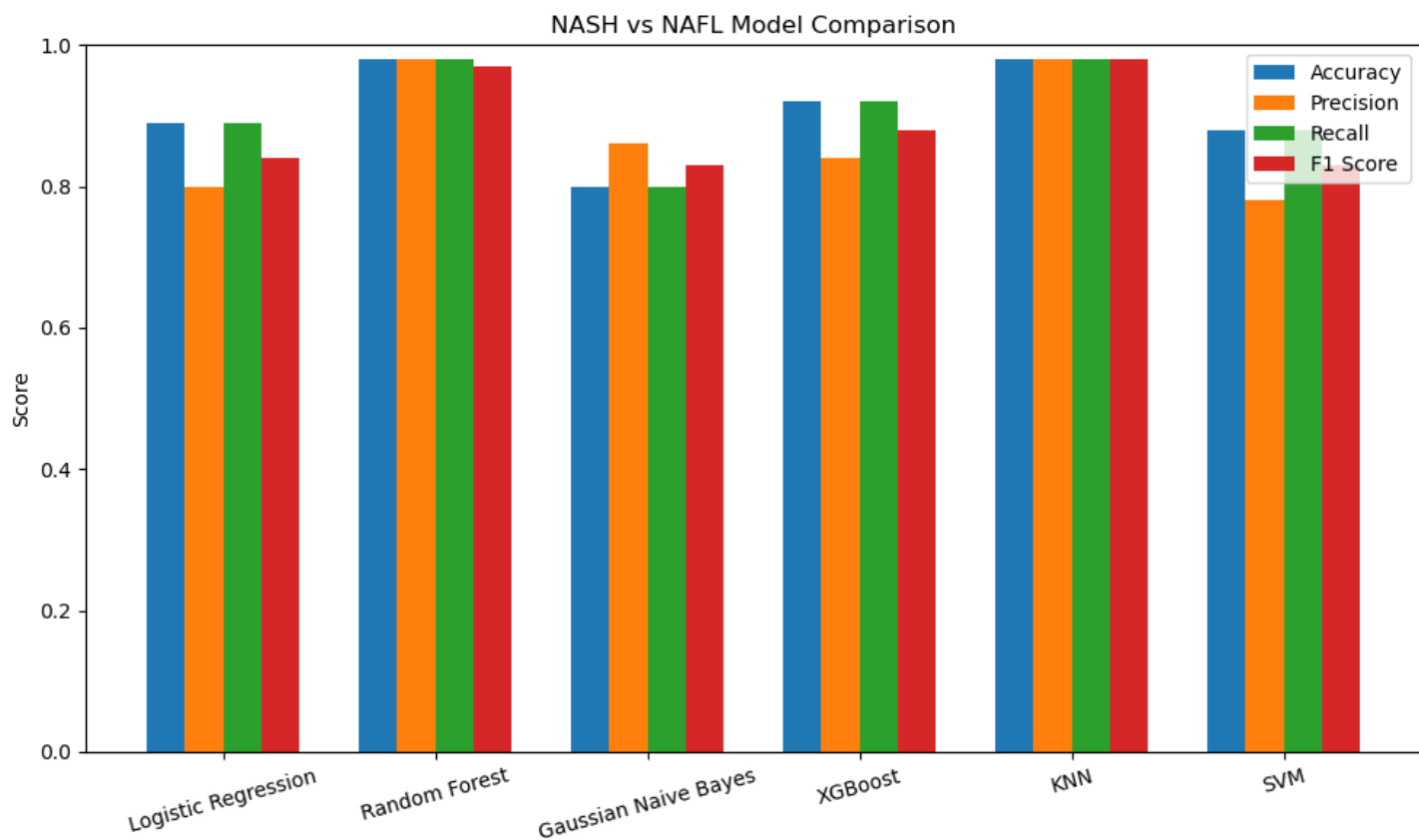
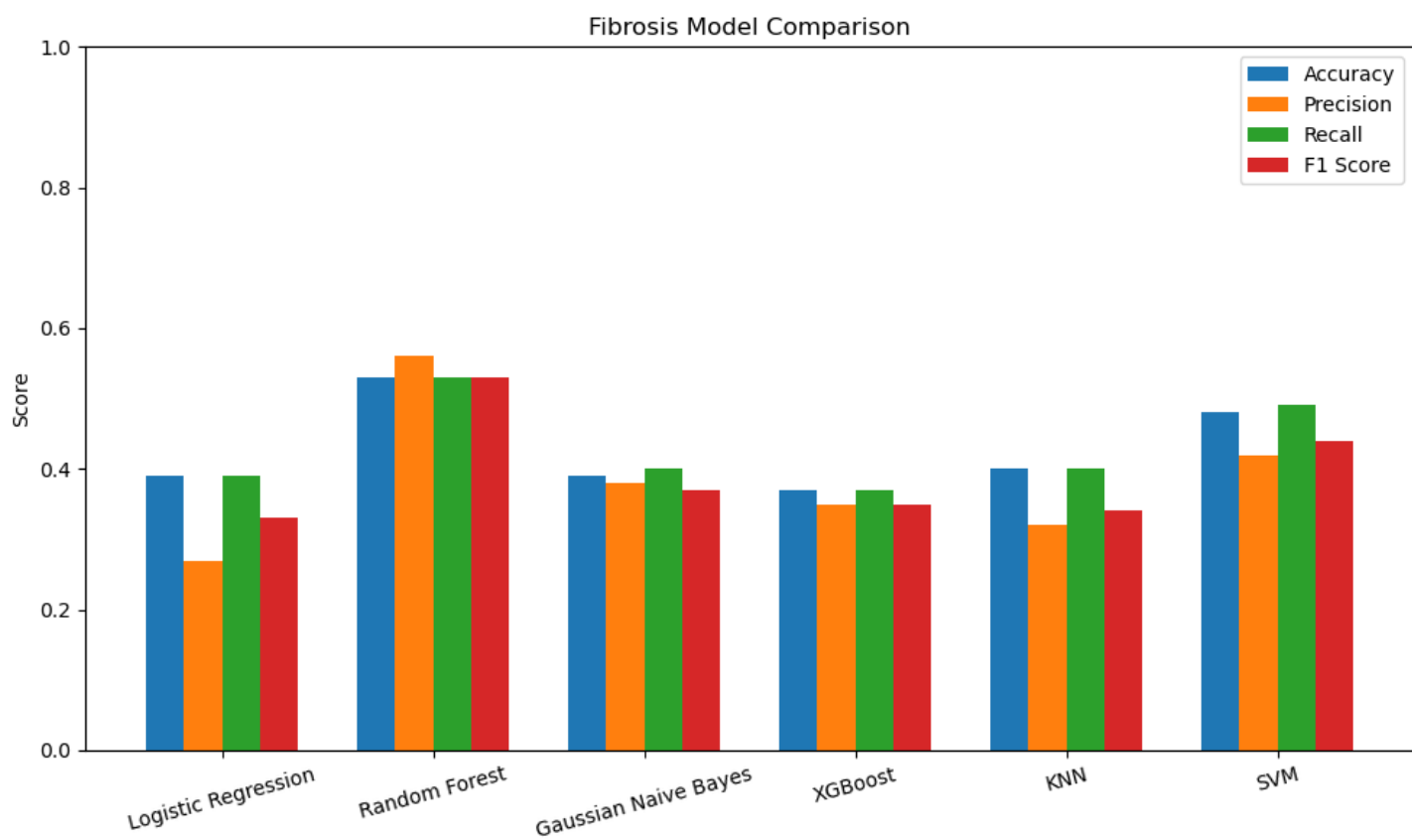


Table 2: For Fibrosis Model Comparison

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.39	0.27	0.39	0.32
Random Forest	0.53	0.56	0.53	0.53
Gaussian Bayes	0.39	0.38	0.40	0.37
XGBoost	0.37	0.35	0.37	0.35
KNN	0.40	0.32	0.40	0.34
SVM	0.48	0.42	0.49	0.44

Graph 7: Comparison between models for Fibrosis



Chapter 4

Result and Discussion

The present study analyzed the relationship between metabolic dysfunction, liver injury markers, and disease progression in NAFLD patients using statistical analysis and machine learning models. Dataset had 62 features in which 28 was finalized and 3 new features were derived from existing features. The source of data was Kaggle, it was not a clinical validated data so variation in result is possible. The type of features where metabolic like hip circumference, bmi, age, gender, etc. and liver related were alt, ast, ggt etc. Other than that, was target feature like fibrosis and nafl column. Preprocessing includes cleaning of data , like removing or replacing nulls, standardizing outlier etc.[23] Standardscaler was use to standardize every numerical data points. New features like ratio of astn alt, fib 4 and apri.Positive relation with HOMA,HOMA with fibrosis, Insulin with fibrosis Diabetes mellitus with fibrosis ,APRI with FIB4 ,AST with ALT.Negative relation HDL showed weak negative association ,LDL showed weak correlation with Spearson and Pearson correlation.[41]

Machine learning models were developed for both fibrosis prediction and NASH vs NAFL classification. In fibrosis prediction, Random Forest achieved is highest performer followed by SVM and KNN[24]. However, overall fibrosis prediction performance remained skeptical due to class imbalance and the complexity of multiclass fibrosis staging. Also, the dataset is not clinically validated so apply these methods may cause more leakages. In contrast, the NASH vs NAFL classification almost all model worked on its best. But Random Forest and KNN models achieved the highest accuracy, precision, recall, and F1-scores, GNB showed comparatively lower performance. This suggest that understanding disease in early stage is easy that to detect it on exact stage without doing invasive process[42]

These results suggest strong association between metabolic dysfunction and NAFLD progression. Elevated insulin resistance, obesity-related parameters, and abnormal lipid profiles were closely associated with fibrosis severity and NASH development. Although fibrosis prediction accuracy was comparatively lower, the developed models still potentially usefulness of ML approaches in non-invasive liver disease prediction. These result show with some advance techniques in ML and other technological field may give chance to predict disease without being invasive.

Chapter 5

Conclusion

This study aims to investigate the relationship between metabolic dysfunction, liver injury markers, and fibrosis severity in patients with Non-Alcoholic Fatty Liver Disease (NAFLD) using statistical analysis and machine learning approaches. Multiple metabolic, biochemical, and clinical parameters were analyzed to evaluate their contribution toward disease progression and predictive modeling.

Metabolic factors like HOMA, BMI, Diabetes and hypertension shows +ve relation with advance Fibrosis stages like cirrhosis and conversion of NASH. Liver related factors and derived features were useful to differentiation. ML model applied to these columns for classification. And best of all was RF and KNN model. KNN is best for small dataset that's why it shows higher result and RF works with complex once this gave benefit in classification.

Overall, findings make us understand interaction between metabolic dysfunction and liver scarring progression in NAFLD. The concatenation of metabolic, liver marker, like AAST, ALT, GGT, HOMA etc. make ML to understand pattern and to understand where progression goes. Limitation here is no clinical data validation, class imbalance, and limited data to detect fibrosis stages.

Further studies may focus on multi-level data, applying advanced hyperparameter optimization, deep learning approaches, for most accuracy and clinical applicability. With all the advancement can be used as first line screen, reducing not needful biopsies. Clinical deployment may take time but with right model and understanding can be achieved.

Chapter 6

References

- [1] F. Nassir, "NAFLD: Mechanisms, Treatments, and Biomarkers," *Biomolecules*, vol. 12, no. 6, p. 824, Jun. 2022, doi: 10.3390/biom12060824.
- [2] A. Ganji, D. Usha, and P. S. Rajakumar, "Enhanced Early Diagnosis of Liver Diseases Using Feature Selection and Machine Learning Techniques on the Indian Liver Patient Dataset," *Scalable Comput. Pract. Exp.*, vol. 26, no. 3, pp. 1104–1115, Apr. 2025, doi: 10.12694/scpe.v26i3.4254.
- [3] A. Pan, S. Mukhopadhyay, and S. Samanta, "RETRACTED: Liver Disease Detection: Evaluation of Machine Learning Algorithms Performances With Optimal Thresholds," *Int. J. Healthc. Inf. Syst. Inform.*, vol. 17, no. 2, pp. 1–19, Jun. 2022, doi: 10.4018/IJHISI.299956.
- [4] B. Mitrovic *et al.*, "Non-alcoholic Fatty Liver Disease, Metabolic syndrome, and Type 2 diabetes mellitus: Where do we stand today?," *Arch. Med. Sci.*, Jun. 2022, doi: 10.5114/aoms/150639.
- [5] J. MacIntyre, *Artificial Intelligence Applications and Innovations: 15th IFIP WG 12. 5 International Conference, AIAI 2019, Hersonissos, Crete, Greece, May 24-26, 2019, Proceedings*. in IFIP Advances in Information and Communication Technology Ser, no. v. 559. Cham: Springer International Publishing AG, 2019.
- [6] B. Pavlyshenko, "Using Stacking Approaches for Machine Learning Models," in *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*, Lviv: IEEE, Aug. 2018, pp. 255–258. doi: 10.1109/DSMP.2018.8478522.
- [7] C. Antunes, M. Azadfard, G. J. Hoilat, and M. Gupta, "Fatty Liver," in *StatPearls*, Treasure Island (FL): StatPearls Publishing, 2026. Accessed: Jul. 16, 2026. [Online]. Available: <http://www.ncbi.nlm.nih.gov/books/NBK441992/>
- [8] A. Bayoumi, M. Shatat, and M. Eslam, "Metabolic associated fatty liver disease and cancer risk: causal role or epiphenomenon?," *Hepatobiliary Surg. Nutr.*, vol. 9, no. 6, pp. 774–776, Dec. 2020, doi: 10.21037/hbsn.2020.03.05.
- [9] T. C.-F. Yip *et al.*, "Non-invasive biomarkers for liver inflammation in non-alcoholic fatty liver disease: present and future," *Clin. Mol. Hepatol.*, vol. 29, no. Suppl, pp. S171–S183, Feb. 2023, doi: 10.3350/cmh.2022.0426.
- [10] S. Pouwels *et al.*, "Non-alcoholic fatty liver disease (NAFLD): a review of pathophysiology, clinical management and effects of weight loss," *BMC Endocr. Disord.*, vol. 22, no. 1, p. 63, Mar. 2022, doi: 10.1186/s12902-022-00980-1.
- [11] C. G. Lall, A. M. Aisen, N. Bansal, and K. Sandrasegaran, "Nonalcoholic Fatty Liver Disease," *Am. J. Roentgenol.*, vol. 190, no. 4, pp. 993–1002, Apr. 2008, doi: 10.2214/AJR.07.2052.
- [12] O. Valencia *et al.*, "Risk Factors Related to the Development of Nonalcoholic Fatty Liver: A Systematic Review," *Can. J. Gastroenterol. Hepatol.*, vol. 2025, p. 9964486, 2025, doi: 10.1155/cjgh/9964486.
- [13] D. H. Fernando, J. M. Forbes, P. W. Angus, and C. B. Herath, "Development and Progression of Non-Alcoholic Fatty Liver Disease: The Role of Advanced Glycation End Products," *Int. J. Mol. Sci.*, vol. 20, no. 20, p. 5037, Oct. 2019, doi: 10.3390/ijms20205037.
- [14] D. Contreras, A. González-Rocha, P. Clark, S. Barquera, and E. Denova-Gutiérrez, "Diagnostic accuracy of blood biomarkers and non-invasive scores for the diagnosis of NAFLD and NASH: Systematic

- review and meta-analysis," *Ann. Hepatol.*, vol. 28, no. 1, p. 100873, Jan. 2023, doi: 10.1016/j.aohp.2022.100873.
- [15] C. S. Ivan Belcic, "Machine Learning Pipeline." [Online]. Available: <https://www.ibm.com/think/topics/machine-learning-pipeline>
- [16] A. Rajagopal *et al.*, "Machine Learning Operations in Health Care: A Scoping Review," *Mayo Clin. Proc. Digit. Health*, vol. 2, no. 3, pp. 421–437, Sep. 2024, doi: 10.1016/j.mcpdig.2024.06.009.
- [17] H. Habebh and S. Gohel, "Machine Learning in Healthcare," *Curr. Genomics*, vol. 22, no. 4, pp. 291–300, Dec. 2021, doi: 10.2174/1389202922666210705124359.
- [18] S. Ghosh, *Machine Learning in Biological Sciences: Updates and Future Prospects*, 1st ed. Singapore: Springer, 2022.
- [19] R. Amin, R. Yasmin, S. Ruhi, M. H. Rahman, and M. S. Reza, "Prediction of chronic liver disease patients using integrated projection based statistical feature extraction with machine learning algorithms," *Inform. Med. Unlocked*, vol. 36, p. 101155, 2023, doi: 10.1016/j.imu.2022.101155.
- [20] Z. M. Younossi, P. Golabi, J. M. Paik, A. Henry, C. Van Dongen, and L. Henry, "The global epidemiology of nonalcoholic fatty liver disease (NAFLD) and nonalcoholic steatohepatitis (NASH): a systematic review," *Hepatology*, vol. 77, no. 4, pp. 1335–1347, Apr. 2023, doi: 10.1097/HEP.0000000000000004.
- [21] Y. Xue, J. Xu, M. Li, and Y. Gao, "Potential screening indicators for early diagnosis of NAFLD/MAFLD and liver fibrosis: Triglyceride glucose index-related parameters," *Front. Endocrinol.*, vol. 13, p. 951689, 2022, doi: 10.3389/fendo.2022.951689.
- [22] P. Li, E. A. Stuart, and D. B. Allison, "Multiple Imputation: A Flexible Tool for Handling Missing Data," *JAMA*, vol. 314, no. 18, pp. 1966–1967, Nov. 2015, doi: 10.1001/jama.2015.15281.
- [23] M. Hassan Mohamed *et al.*, "Toward an Accurate Liver Disease Prediction Based on Two-Level Ensemble Stacking Model," *IEEE Access*, vol. 12, pp. 180210–180237, 2024, doi: 10.1109/ACCESS.2024.3459429.
- [24] V. Sheth, U. Tripathi, and A. Sharma, "A Comparative Analysis of Machine Learning Algorithms for Classification Purpose," *Procedia Comput. Sci.*, vol. 215, pp. 422–431, 2022, doi: 10.1016/j.procs.2022.12.044.
- [25] I. D. Mienye and Y. Sun, "A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects," *IEEE Access*, vol. 10, pp. 99129–99149, 2022, doi: 10.1109/ACCESS.2022.3207287.
- [26] A. R. Kamble, M. M. Jiet, I. N. Weerarathna, S. A. Dhande, S. V. Izankar, and C. G. Puri, "Reinforcement learning model, algorithms, and its application," presented at the INTERNATIONAL CONFERENCE ON EMERGING MATERIALS, SMART MANUFACTURING, AND COMPUTATIONAL INTELLIGENCE, Rajpura, India, 2025, p. 050012. doi: 10.1063/5.0241925.
- [27] A. R. Kamble, M. M. Jiet, I. N. Weerarathna, S. A. Dhande, S. V. Izankar, and C. G. Puri, "Reinforcement learning model, algorithms, and its application," presented at the INTERNATIONAL CONFERENCE ON EMERGING MATERIALS, SMART MANUFACTURING, AND COMPUTATIONAL INTELLIGENCE, Rajpura, India, 2025, p. 050012. doi: 10.1063/5.0241925.
- [28] C. Vandever, "Introduction to Research Statistical Analysis: An Overview of the Basics," *HCA Healthc. J. Med.*, vol. 1, no. 2, Apr. 2020, doi: 10.36518/2689-0216.1062.
- [29] "non alcoholic liver disease." [Online]. Available: <https://www.kaggle.com/datasets/sunilsah905/non-alcoholic-fatty-liver-disease>
- [30] M. S. Ali, M. K. Islam, A. A. Das, D. U. S. Duranta, Mst. F. Haque, and M. H. Rahman, "A Novel Approach for Best Parameters Selection and Feature Engineering to Analyze and Detect Diabetes: Machine Learning Insights," *BioMed Res. Int.*, vol. 2023, no. 1, p. 8583210, Jan. 2023, doi: 10.1155/2023/8583210.

- [31] A. Q. Md, S. Kulkarni, C. J. Joshua, T. Vaichole, S. Mohan, and C. Iwendi, "Enhanced Preprocessing Approach Using Ensemble Machine Learning Algorithms for Detecting Liver Disease," *Biomedicines*, vol. 11, no. 2, p. 581, Feb. 2023, doi: 10.3390/biomedicines11020581.
- [32] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Glob. Transit. Proc.*, vol. 3, no. 1, pp. 91–99, Jun. 2022, doi: 10.1016/j.gltp.2022.04.020.
- [33] A. Kaur, K. Guleria, and N. Kumar Trivedi, "Feature Selection in Machine Learning: Methods and Comparison," in *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, Greater Noida, India: IEEE, Mar. 2021, pp. 789–795. doi: 10.1109/ICACITE51222.2021.9404623.
- [34] F. Mostafa, E. Hasan, M. Williamson, and H. Khan, "Statistical Machine Learning Approaches to Liver Disease Prediction," *Livers*, vol. 1, no. 4, pp. 294–312, Dec. 2021, doi: 10.3390/livers1040023.
- [35] S. Sperandei, "Understanding logistic regression analysis," *Biochem. Medica*, vol. 24, no. 1, pp. 12–18, 2014, doi: 10.11613/BM.2014.003.
- [36] H. A. Salman, A. Kalakech, and A. Steiti, "Random Forest Algorithm Overview," *Babylon. J. Mach. Learn.*, vol. 2024, pp. 69–79, Jun. 2024, doi: 10.58496/BJML/2024/007.
- [37] K. Shashank, A. Sahithya, S. Shakeel, and D. R. Lakshmipriya, "REVIEWS ANALYSIS USING GAUSSIAN NAÏVEBAYES IN MACHINE LEARNING," vol. 10, no. 01, 2023.
- [38] S. Fatima, A. Hussain, S. B. Amir, S. H. Ahmed, and S. M. H. Aslam, "XGBoost and Random Forest Algorithms: An in Depth Analysis," *Pak. J. Sci. Res.*, vol. 3, no. 1, pp. 26–31, Oct. 2023, doi: 10.57041/pjosr.v3i1.946.
- [39] Dr. G. Vijaya Lakshmi, "A Review on K nearest Neighbour Classification Technique in Machine Learning," *Int. J. Sci. Res. Sci. Eng. Technol.*, vol. 12, no. 1, pp. 257–260, Feb. 2025, doi: 10.32628/IJSRSET25121168.
- [40] R. S. Priya, M. E. Student, and N. Soms, "A Review on Support Vector Machine: a Classification Method".
- [41] Z. Noroozi, A. Orooji, and L. Erfannia, "Analyzing the impact of feature selection methods on machine learning algorithms for heart disease prediction," *Sci. Rep.*, vol. 13, no. 1, p. 22588, Dec. 2023, doi: 10.1038/s41598-023-49962-w.
- [42] R. Shafiha, B. Bahcivanci, G. V. Gkoutos, and A. Acharjee, "Machine Learning-Based Identification of Potentially Novel Non-Alcoholic Fatty Liver Disease Biomarkers," *Biomedicines*, vol. 9, no. 11, p. 1636, Nov. 2021, doi: 10.3390/biomedicines9111636.