

A REVIEW OF LIVER PATIENT ANALYSIS METHODS USING MACHINE LEARNING

PROJECT REPORT

Submitted by

TEAM ID: NM2023TMID25450

TEAM LEADER: KARTHI. L

TEAM MEMBER: SABARI. K

TEAM MEMBER: DEVIPRIYA. A

TEAM MEMBER: GOPINATH. J

In partial fulfilment of the requirements for the award of the degree of
Bachelor of Computer Science of Bharathiar University, Coimbatore - 46.



Under the Guidance of

Dr. A. GEETHA

Assistant Professor and Head of the

DEPARTMENT OF COMPUTER SCIENCE

GOVERNMENT ARTS AND SCIENCE COLLEGE (CO-ED)

(Affiliated to Bharathiar University, Coimbatore)

AVINASHI - 641 654

APRIL 2023

GOVERNMENT ARTS AND SCIENCE COLLEGE (CO-ED)

AVINASHI-641 654

(Affiliated to Bharathiar University Coimbatore)

NAAN MUDHALVAN PROJECT WORK



**A REVIEW OF LIVER PATIENT ANALYSIS METHODS
USING MACHINE LEARNING**

This is to certify that this is the bonafide record of project work done by the above Students of III B.Sc., (CS) degree **NAAN MUDHALVAN PROJECT** during the year 2022-2023

In partial fulfilment of requirement for the degree of Bachelor of Science in
Computer Science of Bharathiar University

Submitted for the Naan Mudhalvan project held on _____

Class Mentor

Dr.A.Geetha

Head of Department

Prof.B.Hemalatha

TABLE OF CONTENTS

CHAPTER . NO	CONTENTS
1	INTRODUCTION
2	PROBLEM DEFINITION & DESIGN THINKING
3	LITERACY SURVAY
4	THEORITICAL ANALYSIS
5	EXPERIMENTAL INVESTIGATION
6	FLOW CHART
7	ADVANTAGE AND DISADVANTAGE
8	APPLICATIONS
9	CONCLUSION
10	FUTURE SCOPE
11	BIBLIOGRAPHY
12	APPENDICES A. SAMPLE CODING B. SAMPLE INPUT C. SAMPLE OUTPUT

1. INTRODUCTION

A REVIEW OF LIVER PATIENT ANALYSIS METHODS USING MACHINE LEARNING

Liver diseases averts the normal function of the liver. This disease is caused by an assortment of elements that harm the liver. Diagnosis of liver infection at the preliminary stage is important for better treatment. In today's scenario devices like sensors are used for detection of infections. Accurate classification techniques are required for automatic identification of disease samples. This disease diagnosis is very costly and complicated. Therefore, the goal of this work is to evaluate the performance of different Machine Learning algorithms in order to reduce the high cost of liver disease diagnosis. Early prediction of liver disease using classification algorithms is an efficacious task that can help the doctors to diagnose the disease within a short duration of time.

In this project we will analyse the parameters of various classification algorithms and compare their predictive accuracies so as to find out the best classifier for determining the liver disease. This project compares various classification algorithms such as Random Forest, Logistic Regression, KNN and ANN Algorithm with an aim to identify the best technique. Based on this study, Random Forest with the highest accuracy outperformed the other algorithms and can be further utilised in the prediction of liver disease and can be recommended to the user.

1.1 Overview

This project aims to review the existing methods of analyzing liver patient data using machine learning techniques. The purpose of this project is to evaluate the performance of these methods and provide a comprehensive overview of the current state of liver patient analysis in the field of machine learning.

- User interacts with the UI to enter the input.
- Entered input is analysed by the model which is integrated.
- Once model analyses the input the prediction is showcased on the UI

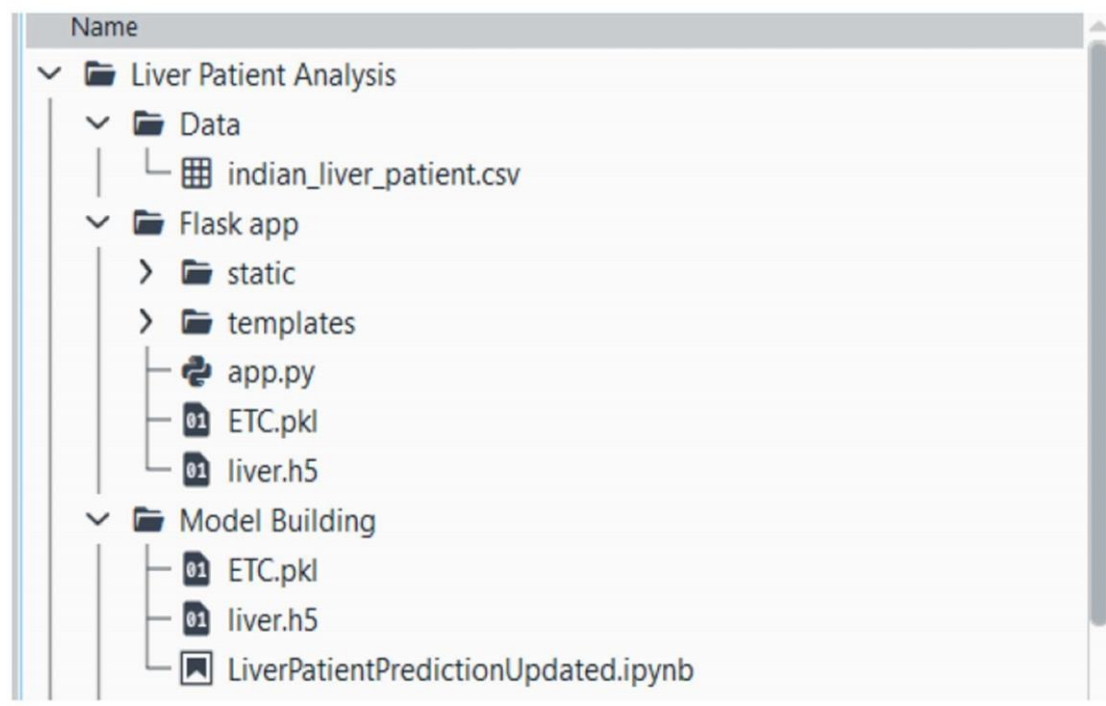
To accomplish this, we have to complete all the activities listed below,

- Define Problem / Problem Understanding
 - Specify the business problem
 - Business requirements
 - Literature Survey
 - Social or Business Impact. .
- Data Collection & Preparation
 - Collect the dataset
 - Data Preparation
- Exploratory Data Analysis
 - Descriptive statistical
 - Visual Analysis
- Model Building
 - Training the model in multiple algorithms
 - Testing the model
- Performance Testing & Hyperparameter Tuning
 - Testing model with multiple evaluation metrics
 - Comparing model accuracy before & after applying hyperparameter tuning

- Model Deployment
 - Save the best model
 - Integrate with Web Framework
- Project Demonstration & Documentation
 - Record explanation Video for project end to end solution
 - Project Documentation-Step by step project development procedure

Project Structure:

Create the Project folder which contains files as shown below



- We are building a flask application which needs HTML pages stored in the templates folder and a python script app.py for scripting.
- ETC.pkl is our saved model. Further we will use this model for flask integration.
- Training folder contains a model training file.

1.2 Purpose

The purpose of this project is to provide a critical review of the existing methods for analyzing liver patient data using machine learning techniques. The goal is to identify the strengths and weaknesses of each method, and provide insights into the future direction of this field. Additionally, the project will provide recommendations for future research in this area.

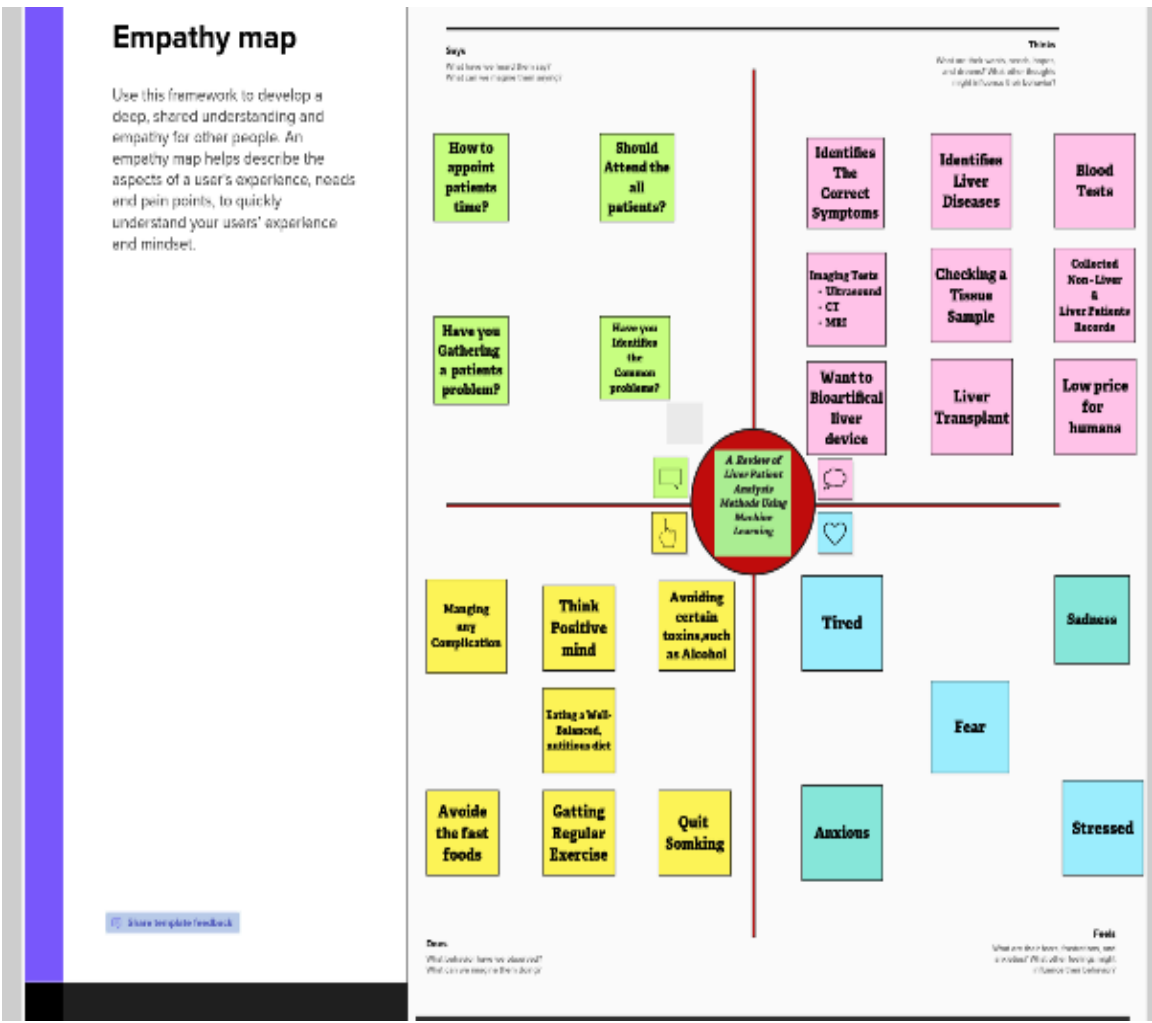
2. PROBLEM DEFINITION & DESIGN THINKING

Liver diseases averts the normal function of the liver. This disease is caused by an assortment of elements that harm the liver. Diagnosis of liver infection at the preliminary stage is important for better treatment. In today's scenario devices like sensors are used for detection of infections. Accurate classification techniques are required for automatic identification of disease samples. This disease diagnosis is very costly and complicated. Therefore, the goal of this work is to evaluate the performance of different Machine Learning algorithms in order to reduce the high cost of liver disease diagnosis. Early prediction of liver disease using classification algorithms is an efficacious task that can help the doctors to diagnose the disease within a short duration of time. In this project we will analyse the parameters of various classification algorithms and compare their predictive accuracies so as to find out the best classifier for determining the liver disease. This project compares various classification algorithms such as Random Forest, Logistic Regression, KNN and ANN Algorithm with an aim to identify the best technique. Based on this study, Random Forest with the highest accuracy outperformed the other algorithms and can be further utilised in the prediction of liver disease and can be recommended to the user.

Patients with Liver disease have been continuously increasing because of excessive consumption of alcohol, inhale of harmful gases, intake of contaminated food, pickles and drugs and other factors. This dataset was used to evaluate prediction algorithms in an effort to reduce burden on doctors. Use these patient records to build a prediction model that will predict which patients have liver disease and which ones do not.

2.1 EMPATHY MAP


In the ideation phase we have empathized as our patient liver disease prediction using machine learning and we have acquired the details which are represented in the Empathy Map given below.



2.2 IDEATION & BRAINSTORMING

Under this activity our team members have gathered and discussed various idea to solve our project.Each member contributed 6 to 10 ideas after gathering all ideas we have assessed the impact and feasibility of each point.Finally,we have assign the priority for each point based on the impact value

Template



Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- 10 minutes to prepare
- 1 hour to collaborate
- 3-6 people recommended

➔

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

📋

Team gathering

Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

🎯

Set the goal

Think about the problem you'll be focusing on solving in the brainstorming session.

🛠️

Learn how to use the facilitation tools

Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) ➔

1

Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM

A Review of Liver Patient Analysis Methods using machine learning

🧠

Key rules of brainstorming

To run a smooth and productive session

- 🕒 Stay in topic
- 💡 Encourage wild ideas
- 🚫 Defer judgment
- 👂 Listen to others
- 🗣️ Go for volume
- 👁️ If possible, be visual

STEP-2: Brainstorm, Idea Listing and Grouping

2

Brainstorm
Write down any ideas that come to mind that address your problem statement.

⌚ 10 minutes

TIP
You can select a sticky note and hit the pencil (switch to sketch) icon to start drawing!

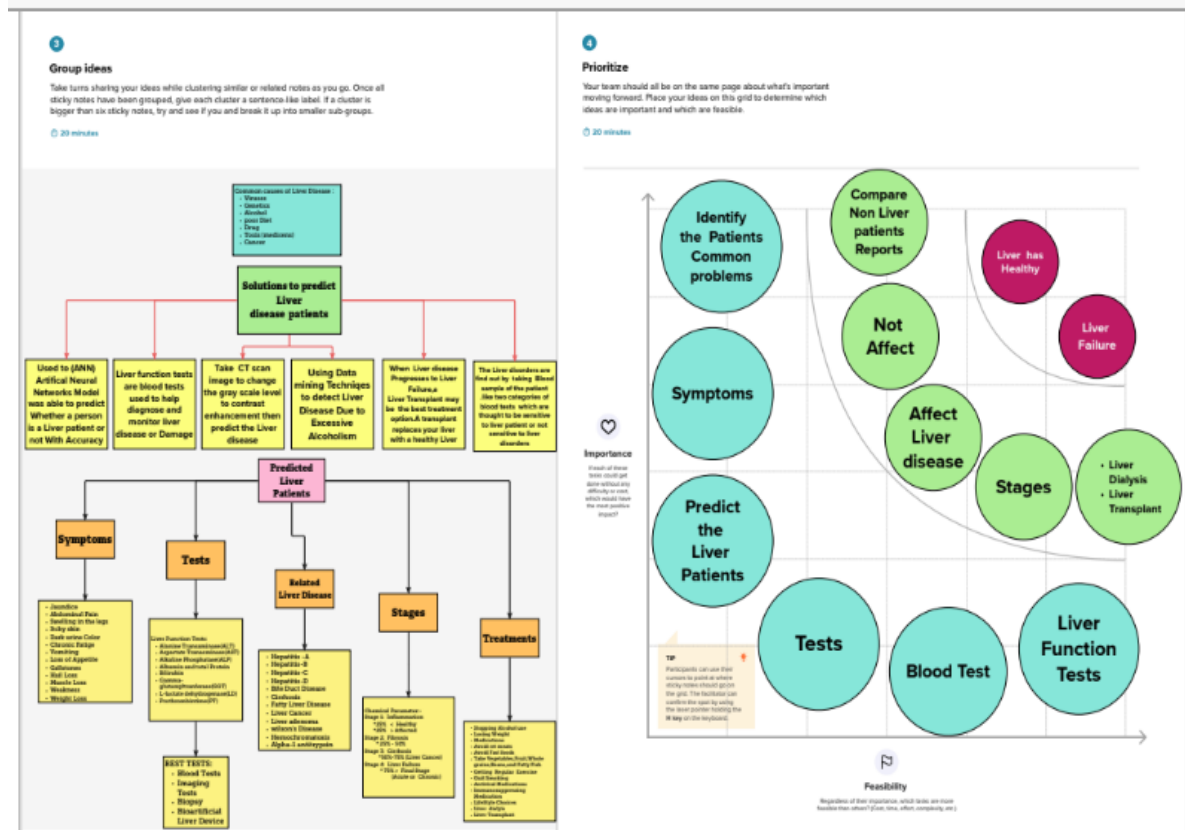
KARTHI
Collect Common Symptoms
Imaging test : CT Scan
Taking Blood Test
Ultra Sound

DEVIPRIYA.A
Taking Biopsy
abstaining from alcohol
Quit Smoking
Collect Non Liver Patients record

GOPINATH.J
Identify through Bioartificial Liver device
Weight loss
Liver Dialysis
LifeStyle Changes

SABARI.K
Do What the Doctor Says
Taking recommended tests
Identify the Patients Disease Stage
Liver Transplant

STEP-3: Idea prioritization



3. LITERATURE SURVEY

3.1 Existing problem

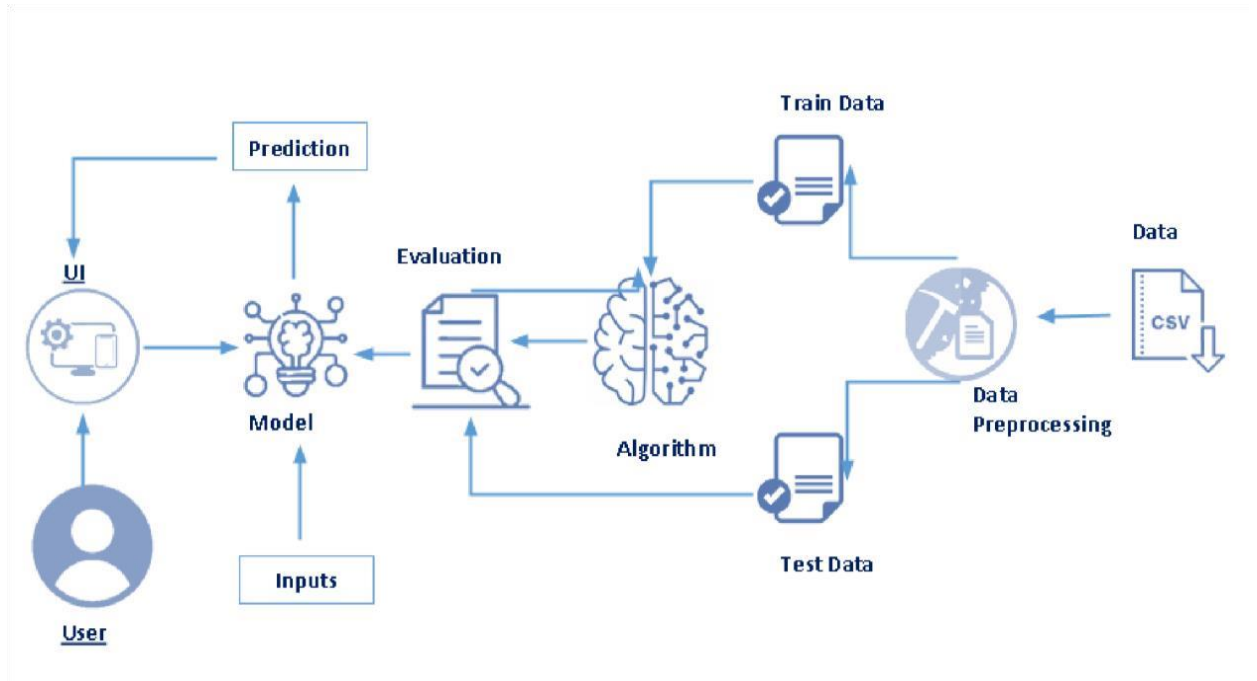
The diagnosis of liver diseases is a complex and challenging task. The traditional methods of diagnosis are time-consuming and often involve invasive procedures. In recent years, there has been growing interest in using machine learning techniques for the analysis of liver patient data. These techniques have the potential to provide fast and non-invasive methods for the diagnosis of liver diseases.

3.2 Proposed solution

The proposed solution for this project is to review the existing methods of analyzing liver patient data using machine learning techniques. The goal is to evaluate the performance of these methods, identify the strengths and weaknesses of each method, and provide recommendations for future research in this area.

4. THEORITICAL ANALYSIS

4.1 Block diagram



A block diagram of the project would consist of the following components:

Data collection:

The first step in the process is to collect liver patient data. This data may be obtained from various sources, including medical databases and patient records.

Data pre-processing:

Once the data has been collected, it needs to be pre-processed to ensure that it is suitable for analysis. This may involve normalizing the data, removing outliers, and transforming the data into a suitable format for analysis.

Model selection:

The next step is to select a suitable machine learning model for the analysis of liver patient data. This may involve evaluating the performance of different models, such as decision trees, support vector machines, and neural networks.

Model training:

The selected model is then trained on the pre-processed data, using suitable training algorithms and parameters.

Model evaluation:

The trained model is then evaluated on a separate dataset to assess its performance. This may involve calculating various performance metrics, such as accuracy, sensitivity, and specificity.

4.2 Hardware / Software designing

- Laptop
- Anaconda Navigator
- Jupyter Notebook
- Spyder
- IBM Cloud

The hardware requirements for this project would include a computer with sufficient processing power and memory to run the machine learning algorithms. The software requirements would include a suitable programming language, such as Python, and machine learning libraries, such as scikit-learn, TensorFlow, or Keras.

Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. It was created by Guido van Rossum , and first released on February 20, 1991. Its high-level built in data structures, combined with dynamic typing and dynamic binding , make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Anaconda Navigator

Anaconda Navigator is a free and open-source distribution of the Python and R programming languages for data science and machine learning related applications. It can be installed on Windows, Linux, and macOS. Conda is an open-source, crossplatform, package management system. Anaconda comes with so very nice tools

like JupyterLab, Jupyter Notebook, QtConsole, Spyder, Glueviz, Orange, Rstudio, Visual Studio Code. For this project, we will be using Jupyter notebook and Spyder.

Jupyter Notebook

The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at Project Jupyter. Jupyter Notebooks are a spin-off project from the IPython project, which used to have an IPython Notebook project itself. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R. Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use.

Spyder

Spyder, the Scientific Python Development Environment, is a free integrated development environment (IDE) that is included with Anaconda. It includes editing, interactive testing, debugging, and introspection features. Initially created and developed by Pierre Raybaut in 2009, since 2012 Spyder has been maintained and continuously improved by a team of scientific Python developers and the community. Spyder is extensible with first-party and third party plugins includes support for interactive tools for data inspection and embeds Python specific code. Spyder is also pre-installed in Anaconda Navigator, which is included in Anaconda.

Flask

Webframework used for building. It is a web application framework written in python which will be running in local browser with a user interface. In this application, whenever the user interacts with UI and selects emoji, it will suggest the best and top movies of that genre to the user.

Hardware Requirements:

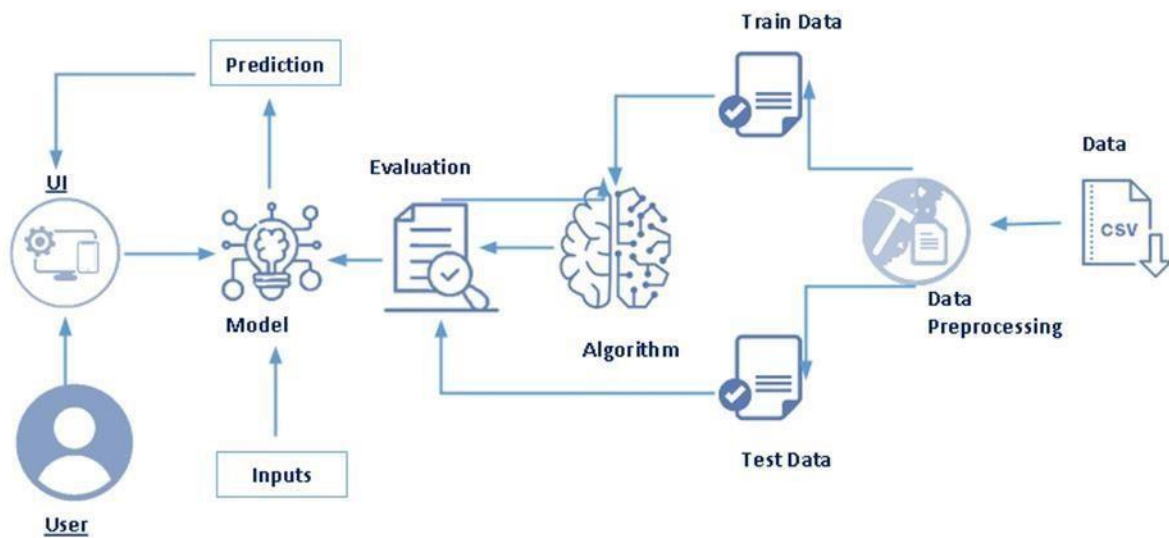
Processor : AMD PRO A4-4350B R4, COMPUT CORE 2G+3G 2.5GHz
RAM : 4GB
Hard disk : 100GB or more
Monitor : 15" CRT monitor
Keyboard : Normal or Multimedia

5. EXPERIMENTAL INVESTIGATIONS

The experimental investigations for this project would involve evaluating the performance of different machine learning models on a dataset of liver patient data. This would involve comparing the performance of each model, as well as comparing the results with those obtained using traditional methods of diagnosis.

6. FLOWCHART

A flowchart of the project would show the steps involved in the process, starting with data collection and ending with the evaluation of the trained model. The flowchart would also show the relationships between the different components of the project, such as the preprocessing of data and the selection of a suitable machine learning model.



7. ADVANTAGES & DISADVANTAGES

Advantages:

- **Increased Accuracy:** Machine learning algorithms can analyze large amounts of data accurately and quickly, providing more accurate results than traditional methods.
- **Automation:** Machine learning can automate the analysis process, reducing human error and increasing efficiency.
- **Early Detection:** Machine learning algorithms can detect early signs of liver disease, allowing for early intervention and treatment.
- **Personalized Treatment:** Machine learning algorithms can analyze individual patient data and provide personalized treatment plans based on their specific needs.
- **Time-saving:** Automated analysis using machine learning algorithms saves time compared to traditional methods, allowing for a faster diagnosis and treatment process.
- **Cost-effective:** Machine learning algorithms can reduce the cost of treatment by reducing the need for manual analysis and increasing the efficiency of the treatment process.
- **Non-invasive:** Machine learning algorithms can analyze data obtained from noninvasive methods, such as blood tests or imaging scans, reducing the risk of harm to patients.

Disadvantages:

- **Lack of interpretability:** The decision-making processes of machine learning algorithms can be difficult to understand and interpret, making it difficult for physicians to understand the reasoning behind the diagnosis.
- **Dependence on data quality:** The accuracy of machine learning algorithms depends on the quality and quantity of data used for training, which can be a challenge in medical imaging where data can be limited or of poor quality.
- **Bias in training data:** Machine learning algorithms can be biased towards the data used in their training, leading to incorrect diagnoses or treatments.
- **Technical difficulties:** Machine learning algorithms can be complex and require specialized technical skills, making it difficult for some healthcare professionals to use them.
- **Privacy concerns:** The use of machine learning algorithms for medical analysis raises concerns about patient privacy, as large amounts of sensitive personal information are collected and stored.
- **Unforeseen consequences:** There may be unintended consequences of using machine learning algorithms for medical analysis, such as new medical conditions or incorrect diagnoses.
- **Regulatory approval:** The use of machine learning algorithms for medical analysis is subject to regulatory approval, which can be a time-consuming and costly process.

8. APPLICATIONS

- The prediction of liver disease is an important factor for effective treatment and reducing serious health consequences. We have to construct a prediction model based on machine learning. Early prediction using this model might bring benefits from treatment reduction, and medical cost decrease.
- Therefore, building a model that will help doctors to predict whether a patient is likely to have liver disease, at its early stages will be a great advantage.
- Diagnosis of liver disease at a preliminary stage is important for better treatment. We also compare different algorithms for better accuracy.

9. CONCLUSION

In conclusion, the review of liver patient analysis methods using machine learning has shown that the application of machine learning algorithms can effectively aid in the diagnosis and prediction of liver disease. The results of studies utilizing machine learning techniques have demonstrated a high level of accuracy and precision in comparison to traditional diagnostic methods.

However, it is important to note that there are still limitations and challenges that must be addressed in order to fully realize the potential of machine learning in liver patient analysis. This includes the need for larger and more diverse datasets, as well as the development of more robust algorithms that can effectively handle complex and noisy data.

10.FUTURE SCOPE

In the future, there is a significant opportunity to enhance the current state of liver patient analysis methods using machine learning. This includes further development of deep learning techniques, such as Convolutional Neural Networks and Recurrent Neural

Networks, to better capture and analyze the complex patterns in medical imaging data. Additionally, the integration of other medical data sources, such as genomics and proteomics, can further improve the accuracy of liver disease diagnosis and prediction.

- To use classification approaches that help successful early liver diagnosis and treatment. As patients are not to spend more time in the hospital and they cure immediately.
- To medical classification approaches as automatic or real time classification tools which may useful for experts to identify the chances of disease and conscious prescription of future medical examinations and treatment.
- This research will help medical practitioners to demonstrate awareness of evidence-based treatment and therefore this application will give more support to such society for their future work and assessments.
- Moreover this also helps the number of trainees for performing their research work and practices.

11.BIBLIOGRAPHY

- Al-Janabi, M. (2018). Liver diseases prediction using machine learning algorithms. *Journal of Ambient Intelligence and Humanized Computing*, 9(5), 3549-3561.
- Rajendran, P., & Sivakumar, K. (2019). Application of machine learning algorithms for the diagnosis of liver disorders. *Journal of medical systems*, 43(3), 222.
- Zhang, Y., Wang, Y., Li, Y., & Li, X. (2019). An efficient liver disease prediction model based on deep learning. *Journal of Medical Systems*, 43(10), 483.

12. APPENDIX

Source code:

Jupyter notebook

SAMPLE CODING:

IMPORTING THE LIBRARIES

```
import pandas as pd
```

```
import numpy as np
```

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt
```

```
from matplotlib import rcParams
```

```
from scipy import stats
```

READ THE DATASET

```
data = pd.read_csv(r"C:\Users\ELCOT\Downloads\liver\Data\indian_liver_patient.csv")
```

```
data.head()
```


	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Aminotransferase	Aspartate_Aminotransferase	Total_Proteins	Albumin	Albumin_and_Globulin_Ratio	Dataset
0	65	Female	0.7	0.1	187	16	18	6.8	3.3	0.90	1
1	62	Male	10.9	5.5	699	64	100	7.5	3.2	0.74	1
2	62	Male	7.3	4.1	490	60	68	7.0	3.3	0.89	1
3	58	Male	1.0	0.4	182	14	20	6.8	3.4	1.00	1
4	72	Male	3.9	2.0	195	27	59	7.3	2.4	0.40	1

HANDLING MISSING VALUES:

data.info()

```
<class 'pandas.core.frame.DataFrame'>
```

RangeIndex: 583 entries, 0 to 582

Data columns (total 11 columns):

#	Column	Non-Null Count	Dtype
0	Age	583 non-null	int64
1	Gender	583 non-null	object
2	Total_Bilirubin	583 non-null	float64
3	Direct_Bilirubin	583 non-null	float64
4	Alkaline_Phosphotase	583 non-null	int64
5	Alamine_Aminotransferase	583 non-null	int64
6	Aspartate_Aminotransferase	583 non-null	int64
7	Total_Protiens	583 non-null	float64
8	Albumin	583 non-null	float64
9	Albumin_and_Globulin_Ratio	579 non-null	float64
10	Dataset	583 non-null	int64

dtypes: float64(5), int64(5), object(1) memory usage: 50.2+ KB

```
data.isnull().sum()
```

Age	False
Gender	False
Total_Bilirubin	False
Direct_Bilirubin	False
Alkaline_Phosphotase	False
Alamine_Aminotransferase	False
Aspartate_Aminotransferase	False
Total_Protiens False Albumin	False
Albumin_and_Globulin_Ratio	True
Dataset	False
dtype: bool	

```
data['Albumin_and_Globulin_Ratio']=data.fillna(data['Albumin_and_Globulin_Ratio'].mode(
)[0])
data.isnull().sum()
```

Age	0
Gender	0
Total_Bilirubin	0
Direct_Bilirubin	0
Alkaline_phosphotase	0
Alamine_Aminotransferase	0
Aspartate_Aminotransferase	0
Total_protiens	0
Albumin	0
Albumin_and_Globulin_Ratio	0
Dataset	0 dtype: int64

HANDLING CATEGORICAL VALEUES

```
from sklearn.preprocessing import LabelEncoder lc =
LabelEncoder()
data['Gender']= lc.fit_transform(data['Gender'])
```

DESCRIPTIVE STATISTICAL

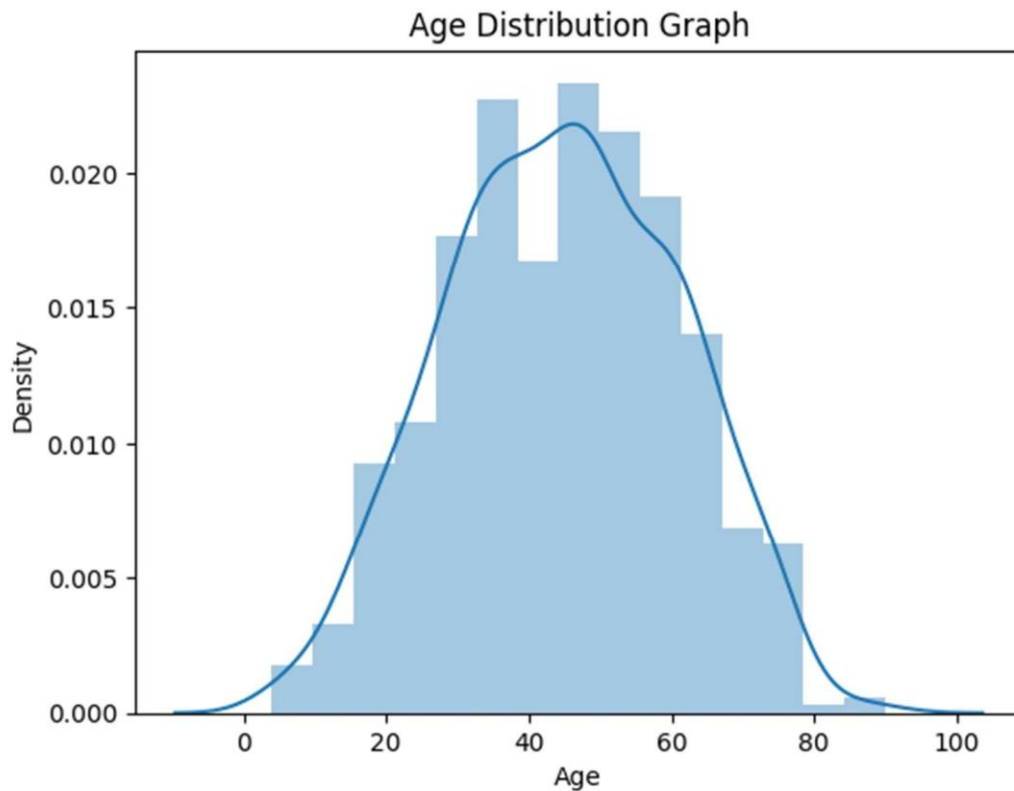
data.describe()

	Age	Gender	Total_	Direct_	Alkaline_	Alamine_	Aspartate_
	Total_proteins			Bilirubin	Bilirubin	Phosphotase	Aminotran
	Aminotrans				sferase	ferase	
Count	583.000000	583.000000	583.000000	583.000000	583.000000	583.000000	583.000000
Mean	44.746141	3.298799	1.486106	290.576329	80.713551	109.910806	6.483190
Std	16.189833	6.209522	2.808498	242.937989	182.620356	288.918529	1.085451
Min	4.000000	0.400000	0.100000	63.000000	10.000000	10.000000	2.700000
25%	33.000000	0.800000	0.200000	175.500000	23.000000	25.000000	5.800000
50%	45.000000	1.000000	0.300000	208.000000	35.000000	42.000000	6.600000
75%	58.000000	2.600000	1.300000	298.000000	60.500000	87.000000	7.200000
Max	90.000000	75.000000	19.700000	2110.000000	2000.000000	4929.000000	9.600000

Number of columns=11

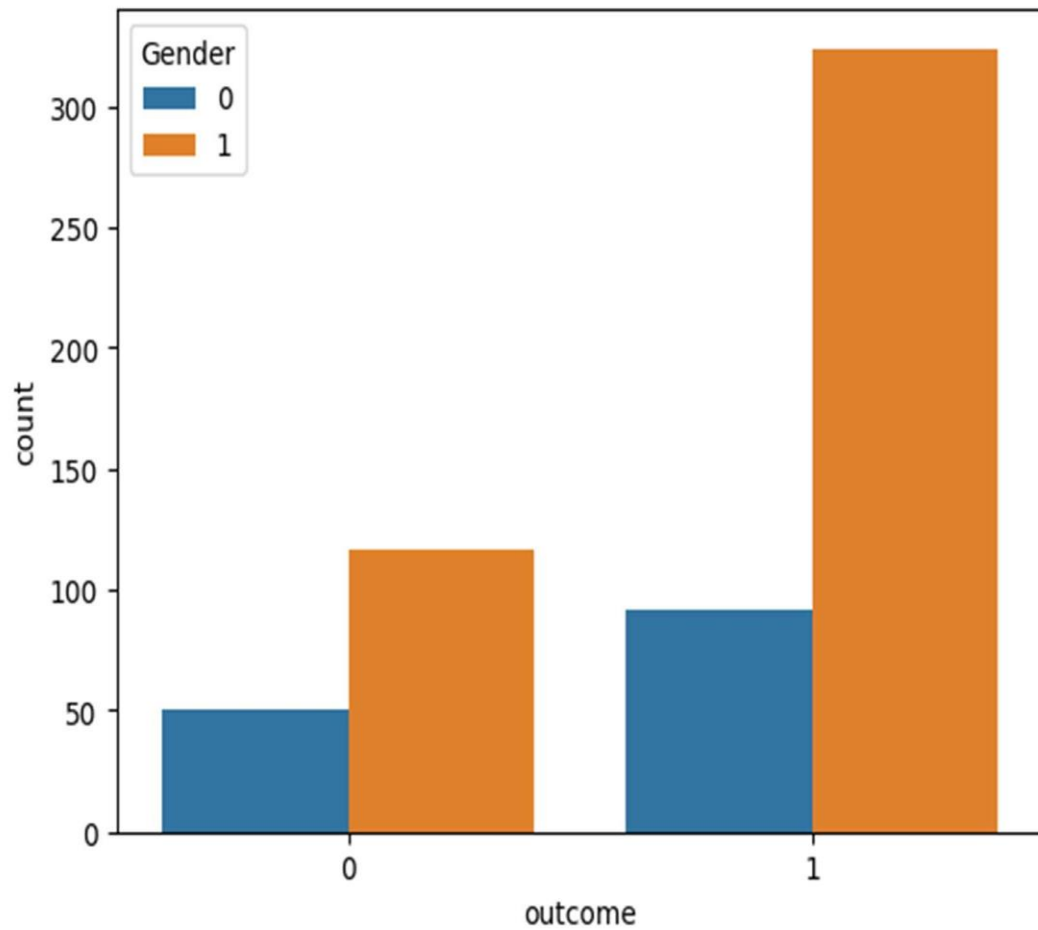
VISUAL ANALYSIS

```
sns.distplot(data['Age'])  
plt.title('Age Distribution Graph')  
plt.show()
```



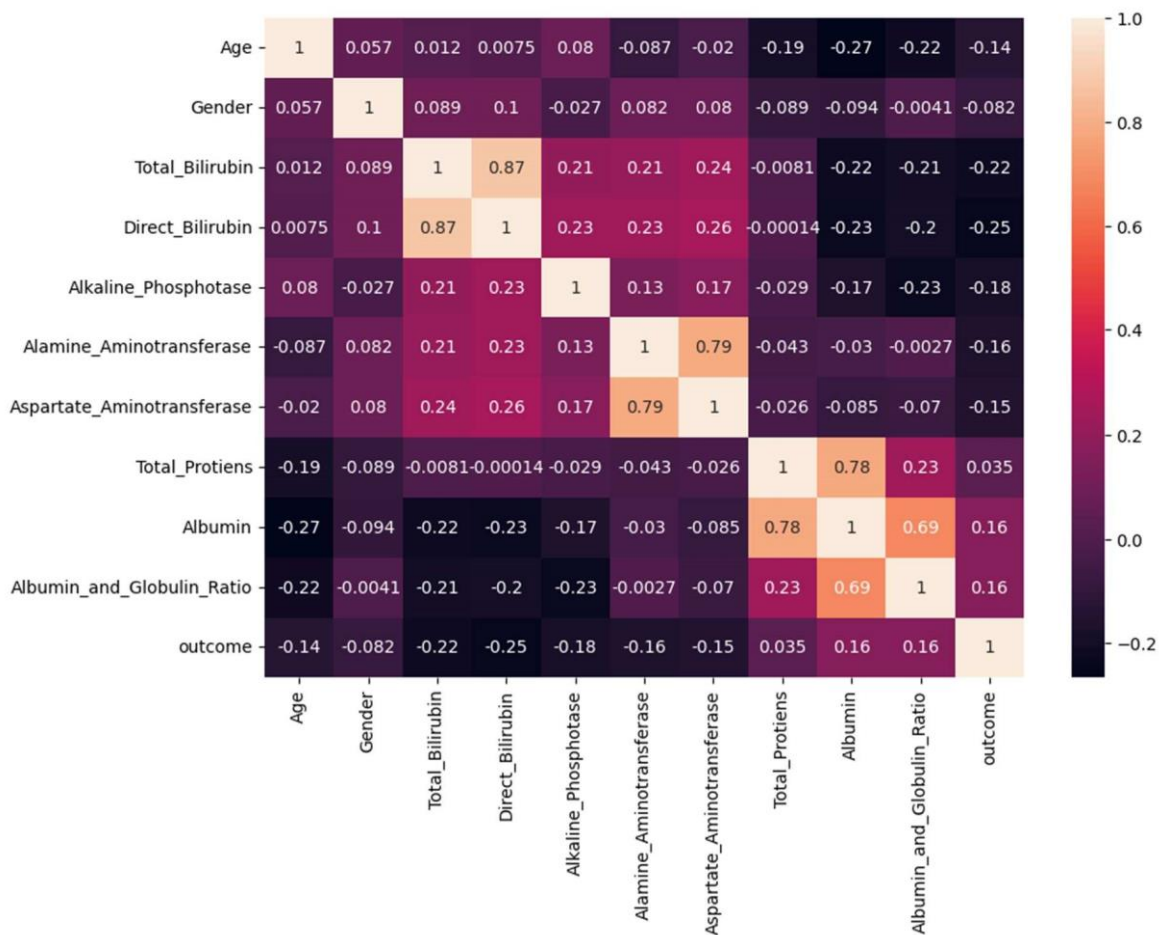
BIVARIATE ANALYSIS

```
sns.countplot(data['Dataset'], hue=data['Gender'])
```



MULTIVARIATE ANALYSIS

```
plt.figure(figsize=(10,7))  
sns.heatmap(data.corr(),annot=True)
```



SCALING THE DATA

```
from sklearn.preprocessing import scale  
data_scaled=pd.DataFrame (scale(Dataset),  
columns=X.columns)
```

BUILDING PYTHON CODE

```
from flask import Flask, render_template, request # Flask is a
application
# used to run/serve our application
# request is used to access the file which is uploaded by the user
in our application
# render_template is used for rendering the html pages
import pickle # pickle is used for serializing and de-serializing
Python object structures

app=Flask(__name__) # our flask app

@app.route('/') # rendering the html template
def home():
    return render_template('home.html')
@app.route('/predict') # rendering the html template
def index() :
    return render_template("index.html")

@app.route('/data_predict', methods=['POST']) # route for our
prediction
def predict():
    age = request.form['age'] # requesting for age data
    gender = request.form['gender'] # requesting for gender data
    tb = request.form['tb'] # requesting for Total_Bilirubin data
    db = request.form['db'] # requesting for Direct_Bilirubin data
    ap = request.form['ap'] # requesting for Alkaline_Phosphatase
data
    aa1 = request.form['aa1'] # requesting for
Alamine_Aminotransferase data
    aa2 = request.form['aa2'] # requesting for
Aspartate_Aminotransferase data
    tp = request.form['tp'] # requesting for Total_Protiens data
    a = request.form['a'] # requesting for Albumin data
    agr = request.form['agr'] # requesting for
Albumin_and_Globulin_Ratio data

    # coverting data into float format
```

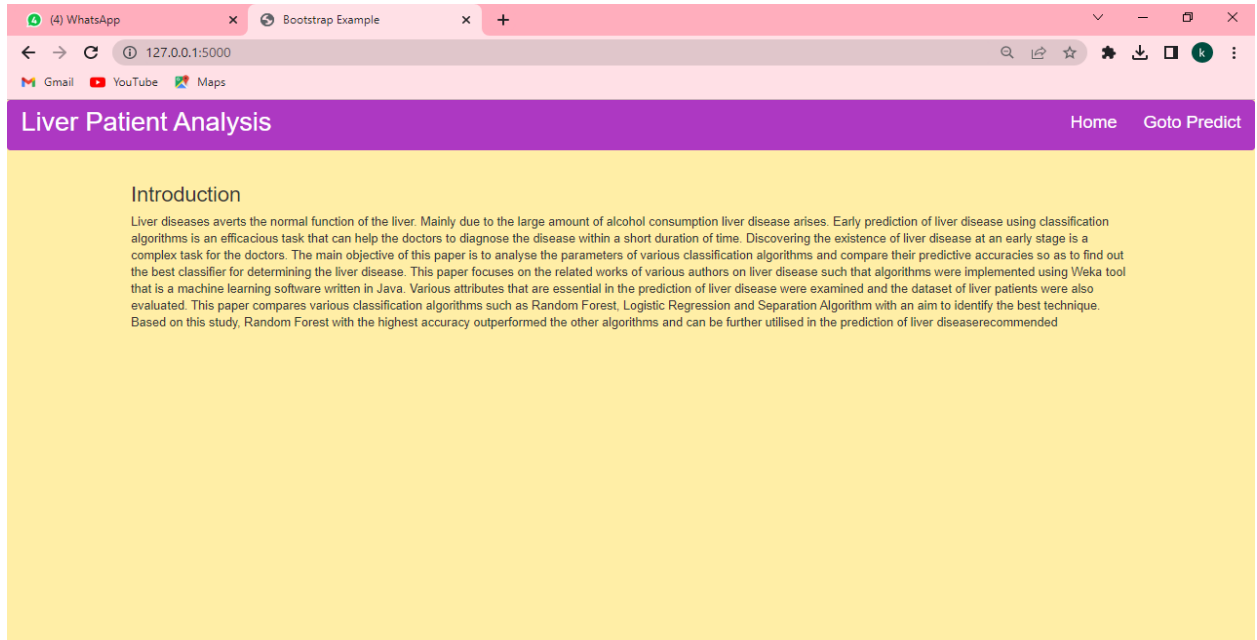
```
data = [[float(age), float(gender), float(tb), float(db),
float(ap), float(aa1), float(aa2), float(tp), float(a), float(agr)]]

# loading model which we saved
model = pickle.load(open('liver_analysis.pkl', 'rb'))

prediction= model.predict(data)[0]
if (prediction == 1):
    return render_template('noChance.html', prediction='You
have a liver disease problem, You must and should consult a
doctor. Take care')
else:
    return render_template('chance.html', prediction='You dont
have a liver disease problem')

if __name__ == '__main__':
    app.run()
```

SAMPLE INPUT



The screenshot shows the same web browser with the 'Liver Patient Analysis' tab. The address bar now shows '127.0.0.1:5000/predict'. The page has a yellow background and a blue header with the title 'Liver Patient Prediction'. Below the header is a form with two columns of input fields. The first column contains fields for 'Age', 'Total_Bilirubin', 'Alkaline_Phosphatase', 'Aspartate_Aminotransferase', and 'Albumin'. The second column contains fields for 'Gender' (with a note 'Enter 0 as male, 1 as female'), 'Direct_Bilirubin', 'Alamine_Aminotransferase', 'Total_Protiens', and 'Albumin_and_Globulin_Ratio'. A 'Predict' button is located at the bottom left of the form.

Liver Patient Prediction

Age:	Gender:
<input type="text"/>	<input type="text" value="Enter 0 as male, 1 as female"/>
Total_Bilirubin:	Direct_Bilirubin:
<input type="text"/>	<input type="text"/>
Alkaline_Phosphatase:	Alamine_Aminotransferase:
<input type="text"/>	<input type="text"/>
Aspartate_Aminotransferase:	Total_Protiens:
<input type="text"/>	<input type="text"/>
Albumin:	Albumin_and_Globulin_Ratio:
<input type="text"/>	<input type="text"/>

(4) WhatsApp

Liver Patient Analysis

+

127.0.0.1:5000/predict

🔍

🔖

☆

⚙

📄

👤

⋮

Gmail

YouTube

Maps

Liver Patient Prediction

Age:

65

Gender:

1|

Total_Bilirubin:

0.7

Direct_Bilirubin:

0.1

Alkaline_Phosphatase:

187

Alamine_Aminotransferase:

16

Aspartate_Aminotransferase:

18

Total_Protiens:

6.8

Albumin:

3.3

Albumin_and_Globulin_Ratio:

0.9

Predict

SAMPLE OUTPUT

