LIVER PATIENT ANALYSIS

using Logistic Regression

Developed by : Manasa Akinepelly , Yashaswi Neela , Achyutha Reddy Bodapatla , Rakesh Basa

Smart Bridge - Remote Summer Internship Program

1. INTRODUCTION

The liver is the largest organ of the body and it is essential for digesting food and releasing the toxic element of the body. The viruses and alcohol use lead the liver towards liver damage and lead a human to a life-threatening condition. There are many types of liver diseases whereas hepatitis, cirrhosis, liver tumors, liver cancer, and many more. Among them liver diseases and cirrhosis are the main cause of death. Therefore, liver disease is one of the major health problems in the world. Every year, around 2 million people died worldwide because of liver disease. According to the Global Burden of Disease (GBD) project, published in BMC Medicine, one million people died in 2010 because of cirrhosis and millions are suffering from liver cancer. Machine learning has made a significant impact on the biomedical field for liver disease prediction and diagnosis. Machine learning offers a guarantee for improving the detection and prediction of disease that has been made an interest in the biomedical field and they also increase the objectivity of the decision-making process. By using machine learning techniques medical problems can be easily solved and the cost of diagnosis will be reduced. In this study, the main aspect is to predict the results more efficiently and reduce the cost of diagnosis in the medical sector. Therefore, we used different classification techniques for the classification of patients that have liver disease. Problems with liver patients are not easily discovered in an early stage as it will be functioning normally even when it is partially damaged. An early diagnosis of liver problems will increase patient's survival rate.

In our Database of 583 records/entries are taken from the ILPD (Indian Liver Patient Dataset) dataset for the purpose of solving problem of this paper.Entire ILPD dataset contains information about 583 Indian liver patients. In which 416 are liver patient records and 167 non liver patient records .The dataset was collected from north east of Andhra Pradesh, India. Selector is a class label used to divide into groups(liver patient or not).

1.1 Overview

Liver disease is rising day by day and is not discovered easily in its initial stage as liver can work properly even when it is partially damaged. It is important to diagnose liver disease early which can increase the patient's survival rate. Expert physicians are required for various examination tests to diagnose the liver disease, but it cannot assure the correct diagnosis.It is very tough job for the expert physicians to recognize the disease from common symptoms.Most of the symptoms are similar to other fever related ailments which result the wrong diagnosis of disease. As the other diseases dominate the liver disease due to which it is unable to identify.
Computer-aided diagnosis is needed for correct prediction of liver disease and it also helps to deal with tremendous and cumbersome data. Research interest is growing in the field of machine learning and knowledge discovery in order to traverse knowledge in detailed volume. Data stored in databases contains valuable hidden knowledge which helps to enhance decision making.Here a new model, used for classifying Liver Patient Analysis by using Machine Learning concepts. Logistic Regression algorithm is used for analyzing the patients having liver disease or not. The model has been built using dataset to predict the the affected people(liver affected). Six algorithms have been used to build the proposed model : Random Forest, Logistic Regression, Decision Tree, SVM, KNN, NaiveBayes. By using the algorithm a Flask model has been implemented and tested. The results has been discussed and a full comparison between algorithms was conducted.Logistic Regression was selected as best algorithm based on accuracy.

1.2 Purpose

Our aim from the project is to make use of pandas, matplotlib, & seaborn libraries from python to extract the libraries for machine learning for analysis of liver patients. By end of this, we can predict whether the patients are affected with liver disease or not.

2. Review of literature

This seems to be a classic example of supervised learning. We have been provided with a fixed number of features for each data point, and our aim will be to train a variety of Supervised Learning algorithms on this data. So that, when a new data point arises, our best performing classifier can be used to categorize the data point as a positive example or negative. Exact details of the number and types of algorithms used for training is included in the 'Algorithms and Techniques' sub-section of the 'Analysis' part.

2.1. Data Table

Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Aminotransferase	Aspartate_Aminotransferase	Total_Proteins	Albumin	Albumin_and_Globulin_Ratio	Dataset
65	Female	0.7	0.1	187	16	18	6.8	3.3	0.90	1
62	Male	10.9	5.5	699	64	100	7.5	3.2	0.74	1
62	Male	7.3	5.1	490	60	68	7.0	3.3	0.89	1
58	Male	1.0	0.4	182	14	20	6.8	3.4	1.00	1
72	Male	3.9	2.0	195	27	59	7.3	2.4	0.40	1
60	Male	0.5	0.1	500	20	34	5.9	1.6	0.37	2
38	Male	1.0	0.3	216	21	24	7.3	4.4	1.50	2

Context

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. This dataset was used to evaluate prediction algorithms in an effort to reduce burden on doctors.

Content

This data set contains 416 liver patient records and 167 non liver patient records collected from North East of Andhra Pradesh, India. The "Dataset" column is a class label used to divide groups into liver patient (liver disease) or not (no disease). This data set contains 441 male patient records and 142 female patient records. Any patient whose age exceeded 89 is listed as being of age "90".

Columns

- Age of the patient
- Gender of the patient
- Total Bilirubin
- Direct Bilirubin
- Alkaline Phosphotase
- Alamine Aminotransferase
- Aspartate Aminotransferase
- Total Protiens
- Albumin
- Albumin and Globulin Ratio
- Dataset: field used to split the data into two sets (patient with liver disease, or no disease)

Methodology

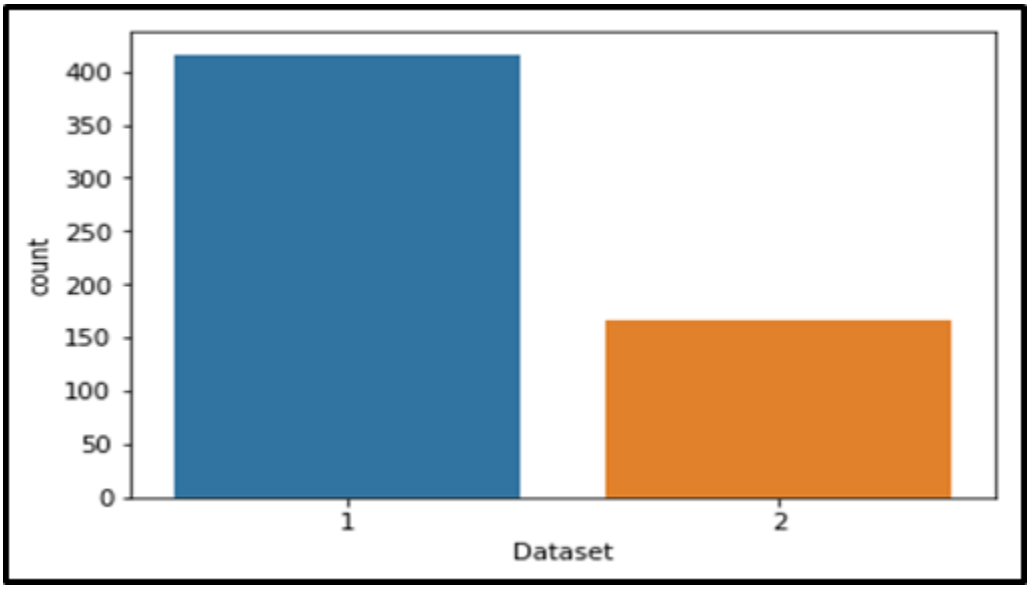
In problems of disease classification like this one, simply comparing the accuracy, that is, the ratio of correct predictions to total predictions is not enough. This is because depending on the context like severity of disease, sometimes it is more important that an algorithm does not wrongly predict a disease as a non-disease. Accuracy can be defined as

$$\text{Accuracy} = TP+TN/(TP+TN+FT+FN)$$

Exploratory Data Analysis

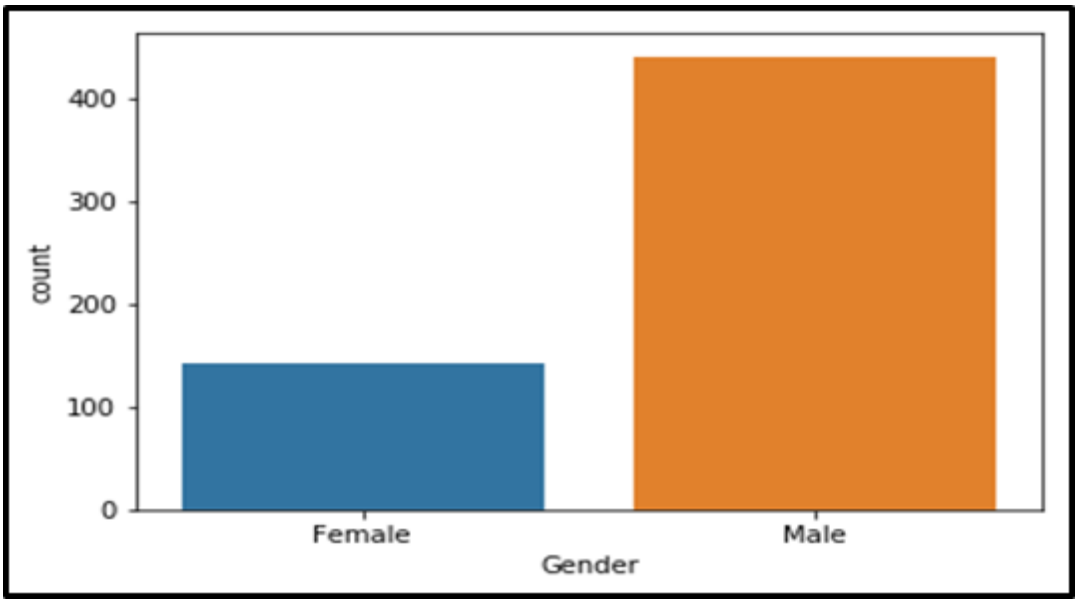
COUNT PLOT OF LIVER PATIENTS DIAGNOISED

Number of patients diagnosed with liver disease: 416
Number of patients not diagnosed with liver disease: 167

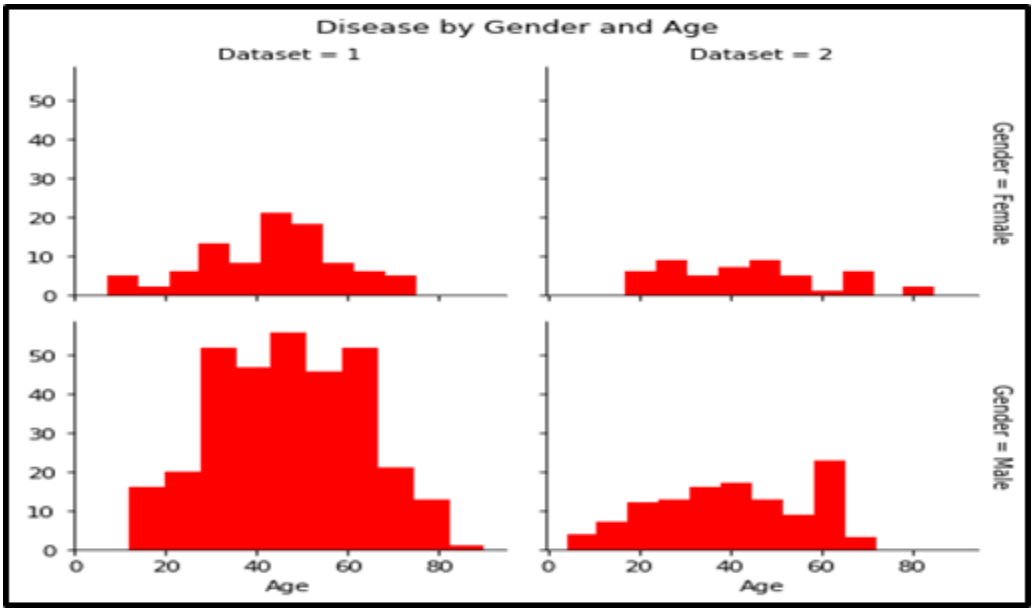


COUNT PLOT OF MALE & FEMALE PATIENTS

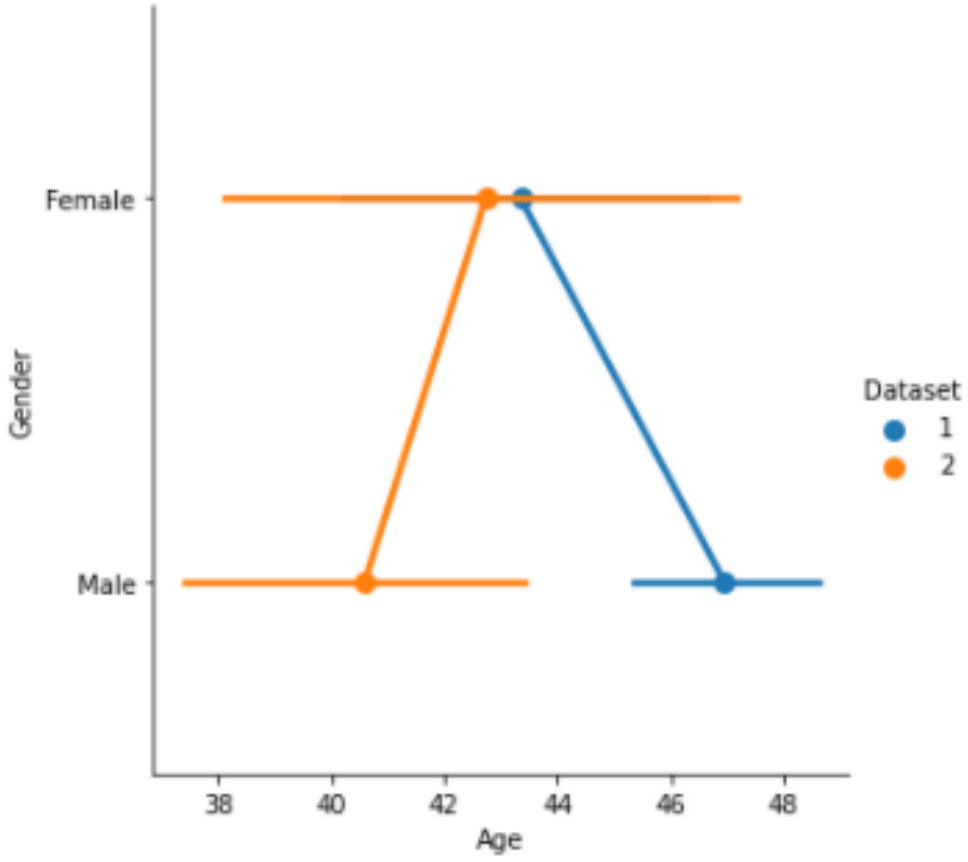
Number of patients that are male: 441
Number of patients that are female: 142



FACETGRID ON DISEASE BY GENDER AND AGE

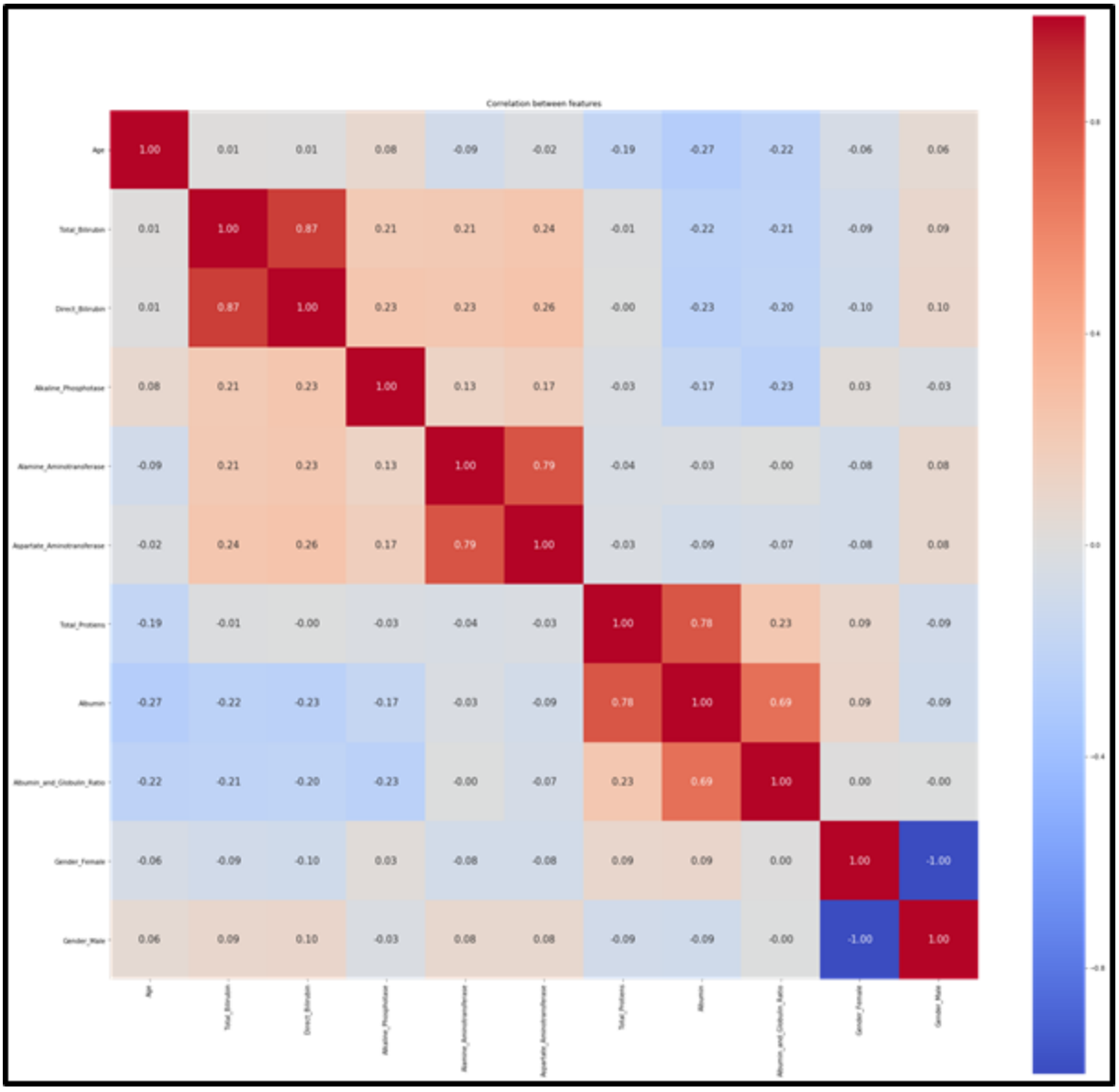


FACTOR PLOT



CO-RELATION GRAPGH

The heatmap is shown appear to have some correlated parameters. Some of these columns have a low correlation. Therefore, we omitted some of the features for better prediction of liver disease.



2.2 Proposed Solution

Machine Learning (Logistic Regression)

Since the outcome is binary and we have a reasonable number of examples at our disposal compared to number of features, this approach seems suitable. When

presented with a number of inputs, it assigns different weights to features (based on their relative importance).

Since for this data it already knows the output beforehand, it continuously adjust the weights such that when these weights summed up with their features are introduced in the random forest classifier function, the results are as near as possible to the actual ones. Once presented with a test value, it again inserts the value into our random forest function and returns the output as a number between 0 and 1, which represents the probability of that test value being in a particular class.

Accuracy scored : 0.6837

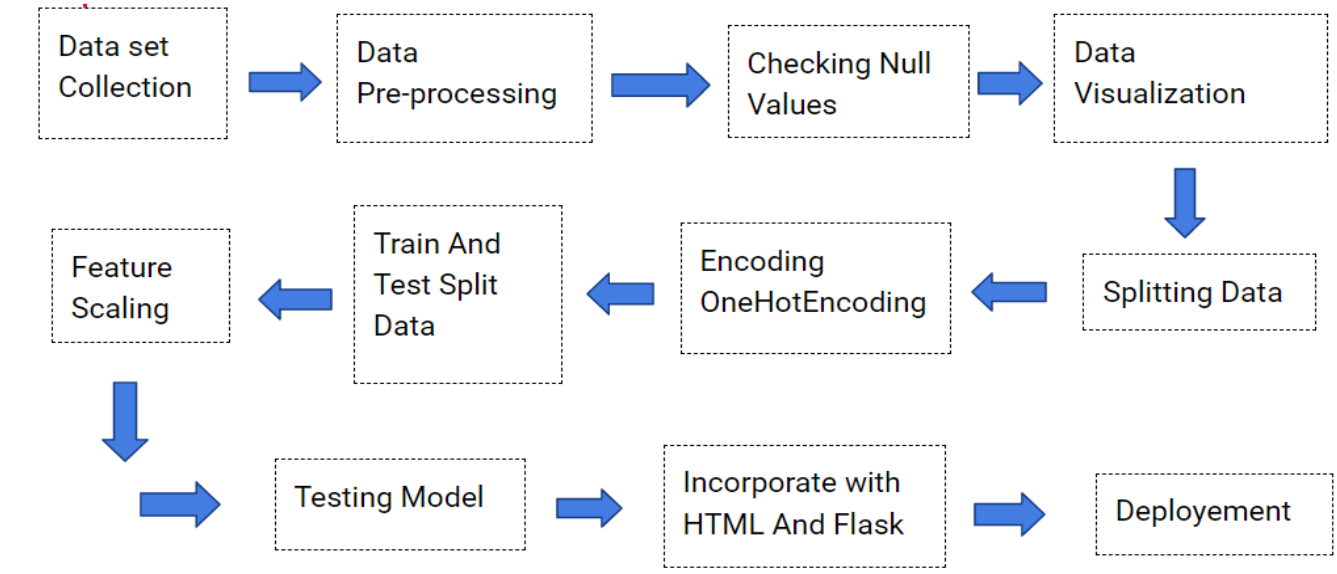
3. THEORETICAL ANALYSIS

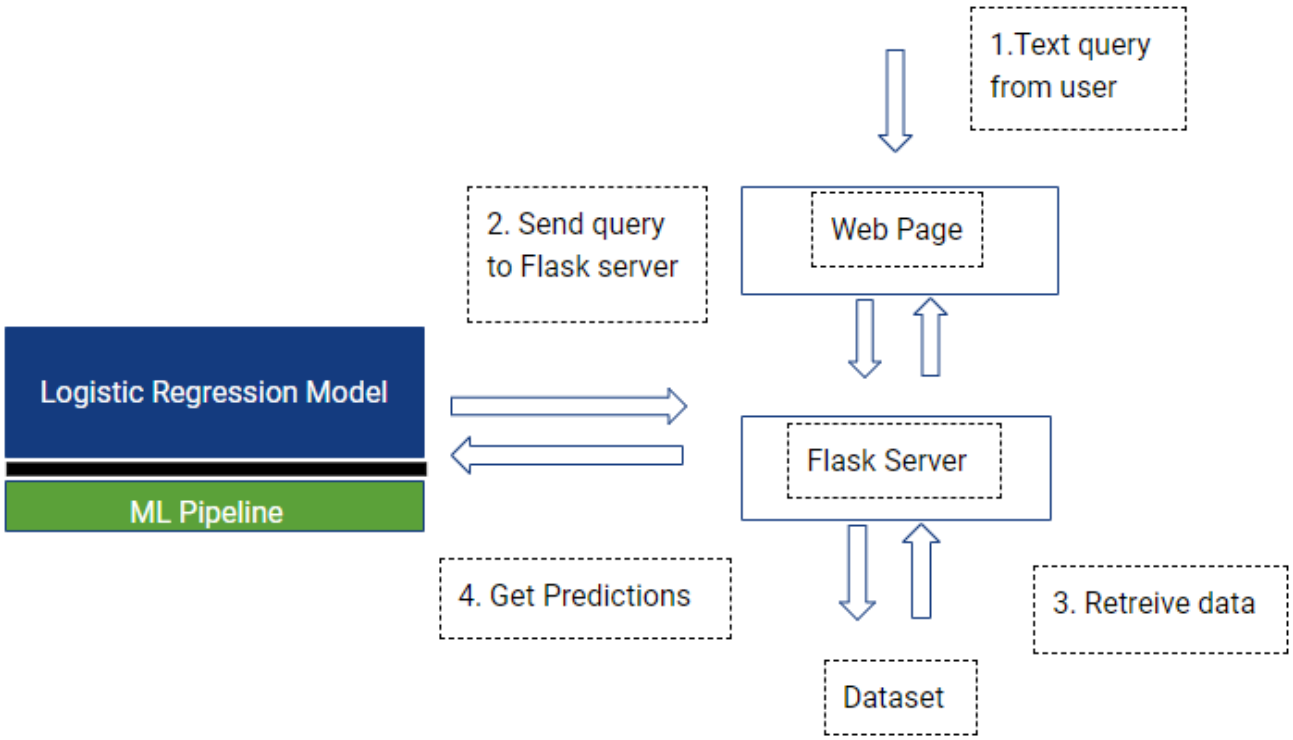
While selecting the algorithm that gives an accurate prediction, we gone through lot of algorithms which gives the results abruptly accurate and from them we selected only one algorithm for the prediction problem that is Random Forest Classifier. It assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Thats how the prediction works great with the Logistic Regression Algorithm.The peculiarity of this problem is collecting the patient's chemical compound details working with the prediction, so we developed an user interface for the people who'll be accesssing for analysis of Liver Patients. Accuracy is defined as the ratio of the number of samples correctly classified by the classifier to the total number of samples for a given test dataset. The formula is as follows :

Accuracy=TP+TN/TP+TN+FT+FN

At first we got like lot of worst accuracies because we tried lot of algorithms for the best accurate algorithm, finally after all of that we tried the best suitable algorithm which gives the prediction accurately is Logistic Regression and developed it to use as a real time analysis probelm for Liver Patients.

3.1 Block Diagram





3.2 Software Designing

- Jupyter Notebook Environment
- Spyder Ide
- Machine Learning Algorithms
- Python (pandas, numpy, matplotlib, seaborn, sklearn)
- HTML
- Flask

We developed this Liver Patient Analysis by using the Python language which is a interpreted and high level programming language and usng the Machine Learning algorithms. for coding we used the Jupyter Notebook environment of the Anaconda distributions and the Spyder, it is an integrated scientific programming in the python language.

For creating an user interface for the prediction we used the Flask. It is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions, and a scripting language to create a webpage is HTML by creating the templates to use in th functions of the Flask and HTML.

4. EXPERIMENTAL INVESTIGATION

This dataset contains 416 liver patient records and 167 non liver patient records collected from North East of Andhra Pradesh, India. The "Dataset" column is a class label used to divide groups into liver patient (liver disease) or not(no disease). This dataset contains 441 male patient records and 142 female patient records. Any patient whose age exceeded 89 is listed as being of age "90". Attributes were shown below in the screenshot of the dataset we used.

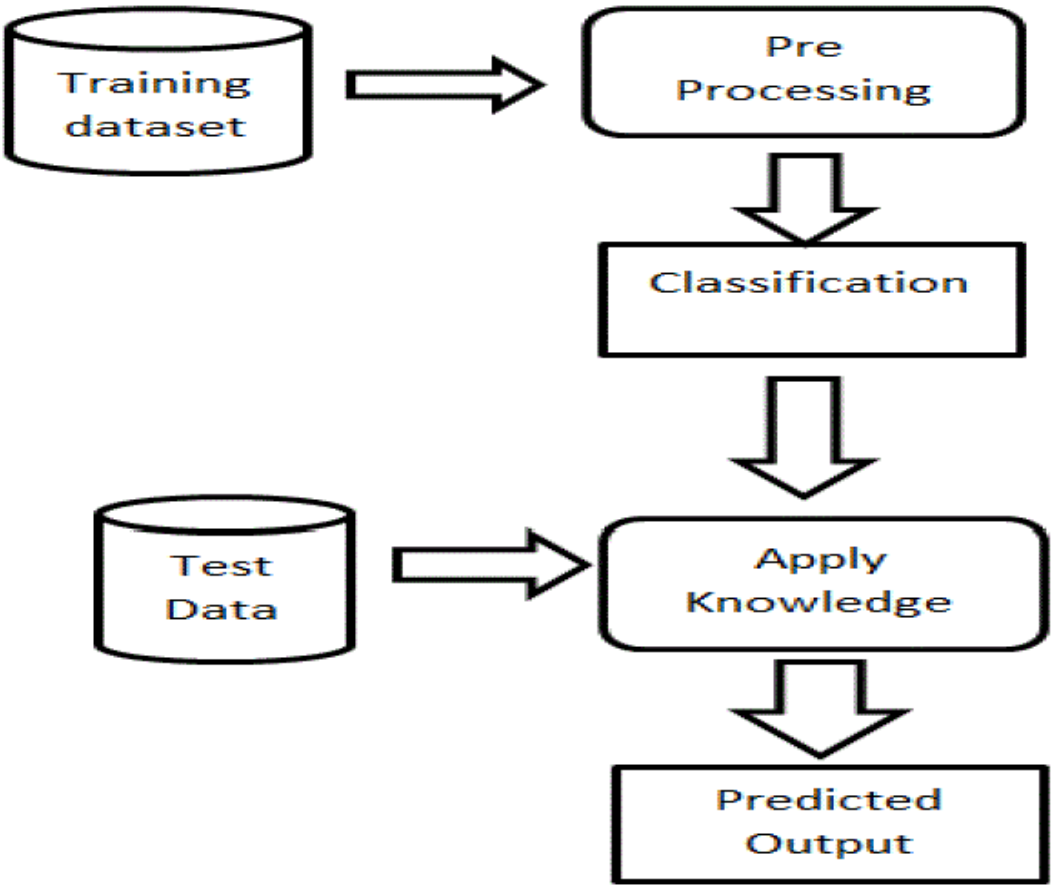
	A	B	C	D	E	F	G	H	I	J	K	L
1	Age	Gender	Total_Bili	Direct_Bil	Alkaline_I	Alamine_A	Aspartate_	Total_Pro	Albumin	Albumin_	Dataset	
2	65	Female	0.7	0.1	187	16	18	6.8	3.3	0.9	1	
3	62	Male	10.9	5.5	699	64	100	7.5	3.2	0.74	1	
4	62	Male	7.3	4.1	490	60	68	7	3.3	0.89	1	
5	58	Male	1	0.4	182	14	20	6.8	3.4	1	1	
6	72	Male	3.9	2	195	27	59	7.3	2.4	0.4	1	
7	46	Male	1.8	0.7	208	19	14	7.6	4.4	1.3	1	
8	26	Female	0.9	0.2	154	16	12	7	3.5	1	1	
9	29	Female	0.9	0.3	202	14	11	6.7	3.6	1.1	1	
10	17	Male	0.9	0.3	202	22	19	7.4	4.1	1.2	2	
11	55	Male	0.7	0.2	290	53	58	6.8	3.4	1	1	
12	57	Male	0.6	0.1	210	51	59	5.9	2.7	0.8	1	
13	72	Male	2.7	1.3	260	31	56	7.4	3	0.6	1	
14	64	Male	0.9	0.3	310	61	58	7	3.4	0.9	2	
15	74	Female	1.1	0.4	214	22	30	8.1	4.1	1	1	
16	61	Male	0.7	0.2	145	53	41	5.8	2.7	0.87	1	
17	25	Male	0.6	0.1	183	91	53	5.5	2.3	0.7	2	
18	38	Male	1.8	0.8	342	168	441	7.6	4.4	1.3	1	
19	33	Male	1.6	0.5	165	15	23	7.3	3.5	0.92	2	
20	40	Female	0.9	0.3	293	232	245	6.8	3.1	0.8	1	
21	40	Female	0.9	0.3	293	232	245	6.8	3.1	0.8	1	
22	51	Male	2.2	1	610	17	28	7.3	2.6	0.55	1	
23	51	Male	2.9	1.3	482	22	34	7	2.4	0.5	1	
24	62	Male	6.8	3	542	116	66	6.4	3.1	0.9	1	
25	40	Male	1.9	1	231	16	55	4.3	1.6	0.6	1	

datasets_2607_4342_indian_liver

Our dataset accuracy values after applying different types of algorithms in machine learning.

Sl.no	Algorithm used	Accuracy
1.	Random Forest Classifier	68
2.	Logistic Regression	68
3.	Decision Tree Classifier	64

5. FLOWCHART



This project examines data from liver patients concentrating on relationships between a key list of liver enzymes, proteins, age and gender using them to try and predict the likeliness of liver disease. Here we are building a model by applying various machine learning algorithms to find the best accurate model and integrate to flask based web application. User can predict the disease by entering parameters in the web application, so that easily anyone can analyse.

6. RESULT

When we run the program, we get the attributes of the csv file of the patients. We can see that all the attributes are not-null, this helps us to identify any null values or missing values. This also shows us the data type of each attribute. So that we can easily identify the patients affected with Liver disease. Using this model we can classify the affected people from a given large dataset. Here Logistic Regression algorithm is used to predict its performance compared with other machine learning algorithms namely the Naive Bayes, KNN, Random Forest and the SVM.

7. ADVANTAGES AND DISADVANTAGES

Advantages:

- Easy and simple User Interface for the liver patients for analysis.
- Logistic Regression is the algorithm we used for analysis to give the accurate result upto 68% .
- It is widely used in Health Sector.
- It is composed using the HTML and Python for the web usage in real time.
- It can work and predict as soon as the necessary details for prediction are given to the model.

Disadvantages:

- Gives only 68% accuracy for the given dataset and if other info is given it may change accordingly.
- It could not work anywhere like an web-application, if one is using other should be quiet.
- Needs more than a single value for the prediction.

8. CONCLUSION

Initially, the dataset was explored and made ready to be fed into the classifiers. This was achieved by removing some rows containing null values, transforming some columns which were showing skewness and using appropriate methods (Label-encoding) to convert the labels so that they can be useful for classification purposes. Performance metrics on which the models would be evaluated were decided. The dataset was then split into a training and testing set.

Firstly, a naive predictor and a benchmark model ('Logistic Regression') were run on the dataset to determine the benchmark value of accuracy. The greatest difficulty in the execution of this project was faced in two areas- determining the algorithms for training and choosing proper parameters for fine-tuning. Initially, I found it very vexing to decide upon 3 or 4 techniques out of the numerous options available in sklearn.

This exercise made me realize that parameter tuning is not only a very interesting but also a very important part of machine learning. I think this area can warrant further improvement, if we are willing to invest a greater amount of time as well as computing power.

APPENDIX

HTML:

```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>ML API</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300'
```

```
rel='stylesheet' type='text/css'>
<link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">

<style>
.login{
top: 20%;
}
</style>
</head>

<body>
 <div class="login">
<h1>LIVER DISEASE ANALYSIS</h1>

    <!-- Main Input For Receiving Query to our ML -->
   <form action="{{ url_for('y_predict')}}"method="post">
      <input type="number" name="Age" placeholder="Age" title="enter age less than 100"
min="0" max="100"required="required" />
            <input type="number" name="Gender" placeholder="gender"
title="0-female,1-male"  min="0" max="1" required="required" />
            <input type="number" name="TB" placeholder="Total_Bilirubin" step="0.01"
min="0" max="85" required="required" />
            <input type="number" name="AP" placeholder="Alkaline_Phosphotase"  min="0"
required="required" />
            <input type="number" name="AM" placeholder="Aspartate_Aminotransferase"
min="0" required="required" />

            <input type="number" name="Tp" placeholder="Total_proteins" step="0.01"
min="0" max="10" required="required" />

            <input type="number" name="agr" placeholder="Albumin_and_Globulin_Ratio"
step="0.01" min="0" max="5"required="required" />
      <button type="submit" class="btn btn-primary btn-block btn-large">Predict</button>

   </form>

  <br>
```

```
  <br>
  {{ prediction_text }}

 </div>


</body>
</html>
```

APP.PY:

```
import numpy as np
from flask import Flask, request, jsonify, render_template
import pickle
from joblib import load
app = Flask(__name__)
model = pickle.load(open('liver.pkl', 'rb'))

@app.route('/')

def home():
    return render_template('index.html')

@app.route('/y_predict',methods=['POST'])
def y_predict():
    '''
    For rendering results on HTML GUI
    '''
    x_test = [[float(x) for x in request.form.values()]]
    print(x_test)
    sc = load('minmax.save')
```

```
    prediction = model.predict(sc.transform(x_test))
    print(prediction)
    output=prediction[0]
    if(output==1):
        pred="has LIVER DISEASE"
    else:
        pred=" does NOT HAVE LIVER DISEASE"

    return render_template('index.html', prediction_text='Patient {}'.format(pred))

if __name__ == "__main__":
    app.run(debug=True)
```