LIVER DISEASE PREDICTION USING MACHINE LEARNING

A PROJECT REPORT

Submitted by

PRATEEK PANDEY [Reg No: RA1711003030465]
AYUSH GUPTA [Reg No: RA1711003030466]

Under the guidance of
Ms. BHARTI VIDHURY
(Assistant Professor, Department of Computer Sciene & Engineering)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

of

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM
INSTITUTE OF SCIENCE AND TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)
DELHI-NCR CAMPUS, GHAZIABAD (U.P)

SRM INSTITUTE OF SCIENCE & TECHNOLOGY, NCR CAMPUS

MAY 2021

SRM INSTITUTE OF SCIENCE & TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this project report titled “**LIVER DISEASE PREDICTION USING MACHINE LEARNING** ” is the bonafide work of “ **PRATEEK PANDEY [Reg No: RA1711003030465], AYUSH GUPTA [Reg No: RA1711003030466], , ,** ”, who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Ms. BHARTI VIDHURY
GUIDE
Assistant Professor
Dept. of Computer Sciene & Engineering

SIGNATURE

Dr. R.P.MAHAPATRA
HEAD OF THE DEPARTMENT
Dept. of Computer Science & Engineering

Signature of the Internal Examiner

Signature of the External Examiner

ABSTRACT

One of the most lethal diseases in several countries is Liver Diseases. The immoderate consumption of alcohol, pickle, intake of contaminated food, drugs and inhale of harmful gases leads to increase in count of patients with liver Diseases. To predict liver disease the dataset of liver patient are used to investigate and for building classification models. To reduce burden on doctors we used this dataset to assess the prediction algorithm in an effortless manner. A liver disease which lasts over a period of six months is termed as chronic liver diseases. We come up with the various classification which can improve the performance of classification in the situation that training dataset is available. Thus outputs shows from proposed classification model indicate that accuracy in predicting the result.

ACKNOWLEDGEMENTS

I am indebted to my guide, Ms. BHARTI VIDHURY for her guidance and constant supervision throughout the research as well as for providing necessary information everytime we have approached her. All through the work, in her busy schedule, she has extended cordial support to us for completing this research work. I came to know about so many new things which has improved my knowledge and I am very thankful to her.

Prateek Pandey , Ayush Gupta

TABLE OF CONTENTS

LIST OF FIGURES

SYSTEM REQUIREMENTS

1. Anaconda Software version 2020.11.0.0.

2. Pyhton language version 3.8.0.

3. Jupyter Notebook version 6.2.0.

4. Windows 7, 8, 8.1, 10 with x86 and x64 compatibility.

5. Recommended 8 GB RAM.

6. Anaconda prompt for downloading python modules.

7. Recommended 16 GB space.

8. i5 processor or above.

CHAPTER 1

INTRODUCTION

Liver Disease is one of the lethal disease in several countries. It is the disruption of liver function that affect the body. It is also known as hepatic disease. The immoderate consumption of alcohol, pickle, intake of contaminated food, drugs and inhale of harmful gases leads to increase in count of patients with liver diseases.

Indicators of Liver Disease:-

- Eyes and Skin is yellowish
- Legs and ankle will Swell
- Urine Colour will be dark
- Appetite Loss
- Chronic Exhaustion

Nowadays, everyone is facing problem due to shortage of medical professionals which makes the diagnosis delayed. A classic scenario, prevalent somewhat in urban areas and mostly in rural is:

- A patient reach to doctor with definite symptoms.
- The doctor recommend respected tests like complete blood count(CBC), liver function test(LFT), urine test, stool test etc. determined by the symptoms of body.
- The patient carry his reports and go to the hospital, where the doctors analyze the report and identify the disease.

Figure 1.1: A Healthy Liver

In this project we used lot of different algorithms of machine learning to increase the efficiency and predict the most accurate result of liver disease.

- In this paper, we proposed the idea in which we check the patient liver disease using machine learning algorithms.We take 583 patient details for producing different classification to envision liver disease.
- For reference we take chronic liver disease which lasts over six months. So we will take results of positive disease information and negative disease information.
- By using classifiers, we are proceeding liver Disease percentage and values are showing as a confusion matrix.We acquire a various classification scheme which can effectively improve the classification performance in the situation that dataset is available.
- We will get all that details from the dataset and produce outputs shows from proposed classification model indicate that accuracy in predicting the result.

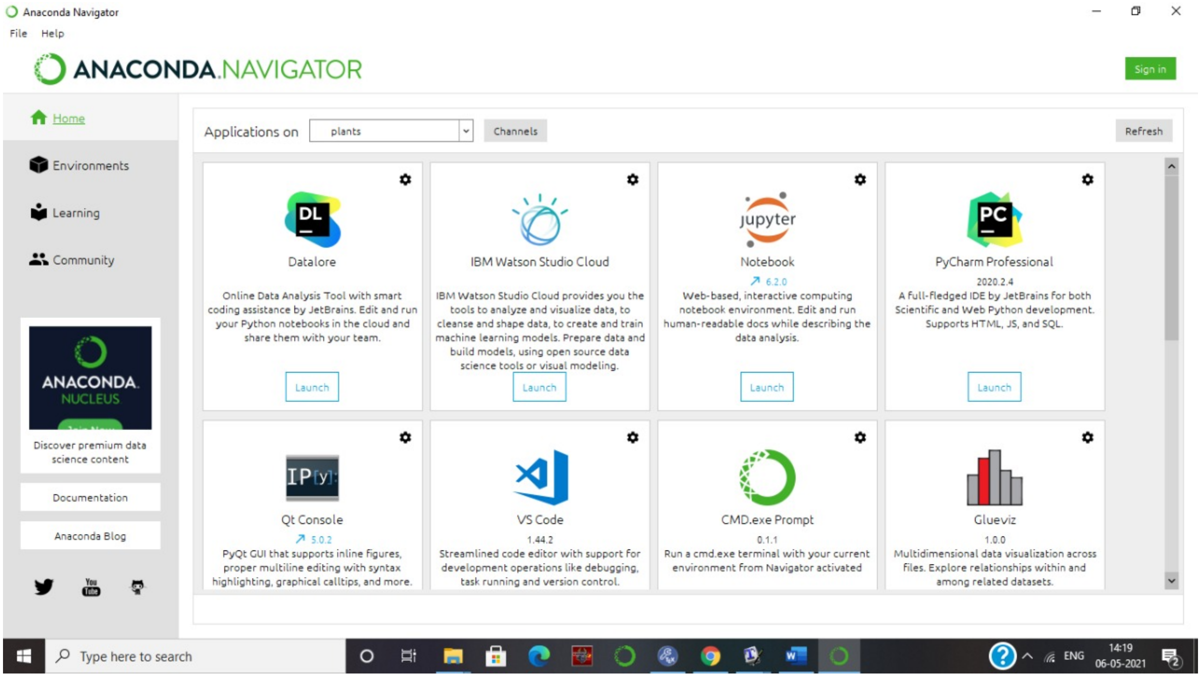
1.1 Anaconda

Anaconda is a software tool which is generally made for the Python and R languages.This tool is basically the birthplace of Data Science.This tool consists of various Python interpreters , R Studio and Visualisation tools as shown in Figure 1.

- One of the graphical user interface is Anaconda that give permission to users to start application and run packages, channels while not defraud command-line commands.It's accessible for Windows, macOS and UNIX operating system.

1.2 Jupyter Notebook

Jupyter Notebook is one of the best Python interpreter provided by the Anaconda which is used for performing machine learning and data science processes. It is available free and comfortable to use. The Jupyter Notebook is am ASCII text file net application that enables us to construct and serve documents that contains code, equations, visualizations and narrative text. Uses include: knowledge improvement and transformation, numerical simulation, applied math modeling, knowledge mental image, machine learning, and far a lot of.



Discover premium data science content

Datalore

Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team.

IBM Watson Studio Cloud

IBM Watson Studio Cloud provides you the tools to analyze and visualize data, to cleanse and shape data, to create and train machine learning models. Prepare data and build models, using open source data science tools or visual modeling.

Notebook

6.2.0

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

PyCharm Professional

2020.2.4

A full-fledged IDE by JetBrains for both Scientific and Web Python development. Supports HTML, JS, and SQL.

Qt Console

5.0.2

PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

VS Code

1.44.2

Streamlined code editor with support for development operations like debugging, task running and version control.

CMD.exe Prompt

0.1.1

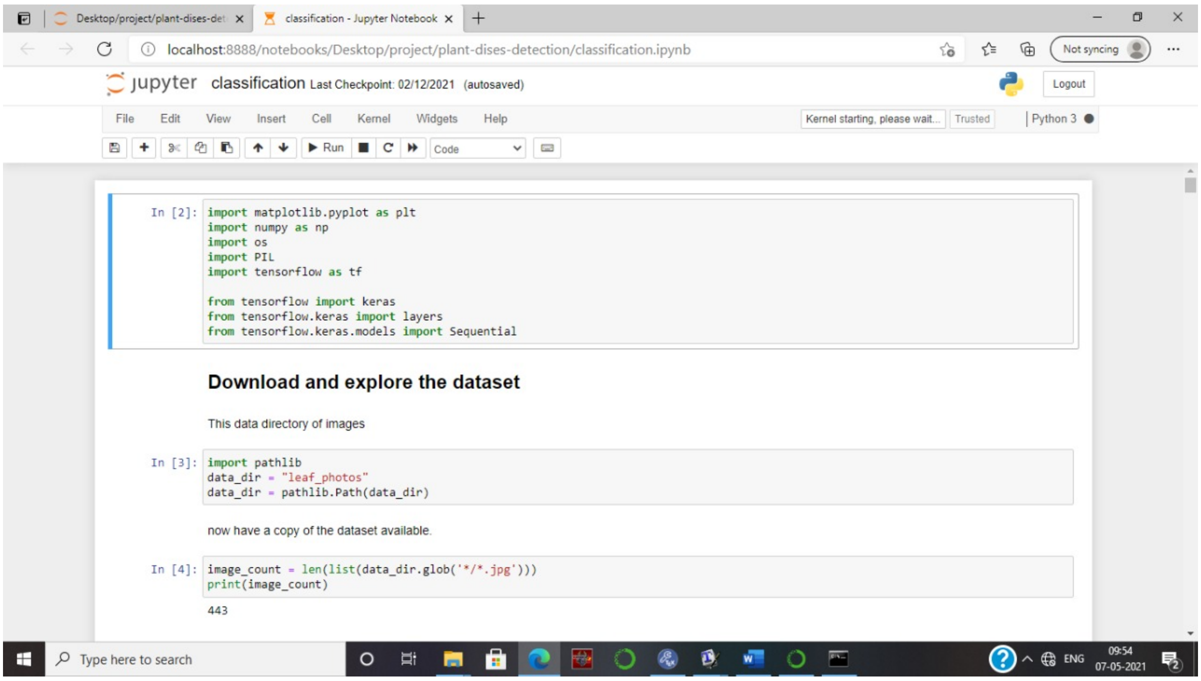
Run a cmd.exe terminal with your current environment from Navigator activated

Glueviz

1.0.0

Multidimensional data visualization across files. Explore relationships within and among related datasets.

Figure 1.2: Anaconda Software


Desktop/project/plant-dises-det
classification - Jupyter Notebook
localhost:8888/notebooks/Desktop/project/plant-dises-detection/classification.ipynb
Not syncing
jupyter
Logout
File Edit View Insert Cell Kernel Widgets Help
Kernel starting, please wait...
Trusted
Python 3
Run
Code
Type here to search
09:54
07-05-2021

classification Last Checkpoint: 02/12/2021 (autosaved)

```
In [2]: import matplotlib.pyplot as plt
        import numpy as np
        import os
        import PIL
        import tensorflow as tf

        from tensorflow import keras
        from tensorflow.keras import layers
        from tensorflow.keras.models import Sequential
```

Download and explore the dataset

This data directory of images

```
In [3]: import pathlib
        data_dir = "leaf_photos"
        data_dir = pathlib.Path(data_dir)
```

now have a copy of the dataset available.

```
In [4]: image_count = len(list(data_dir.glob('*/*.jpg')))
        print(image_count)

        443
```

Figure 1.3: Jupyter Notebook

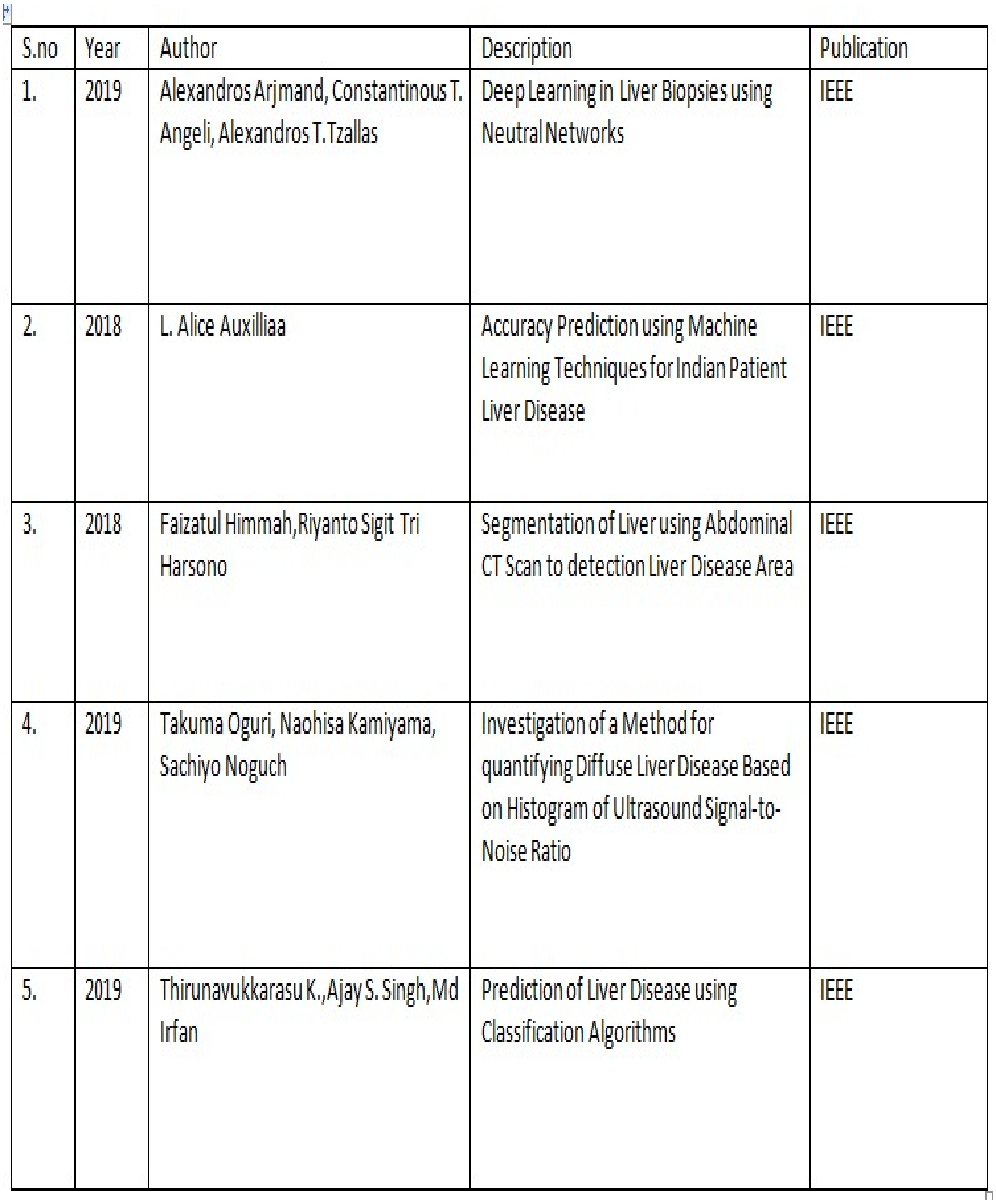
CHAPTER 2

LITERATURE SURVEY

Research Gap

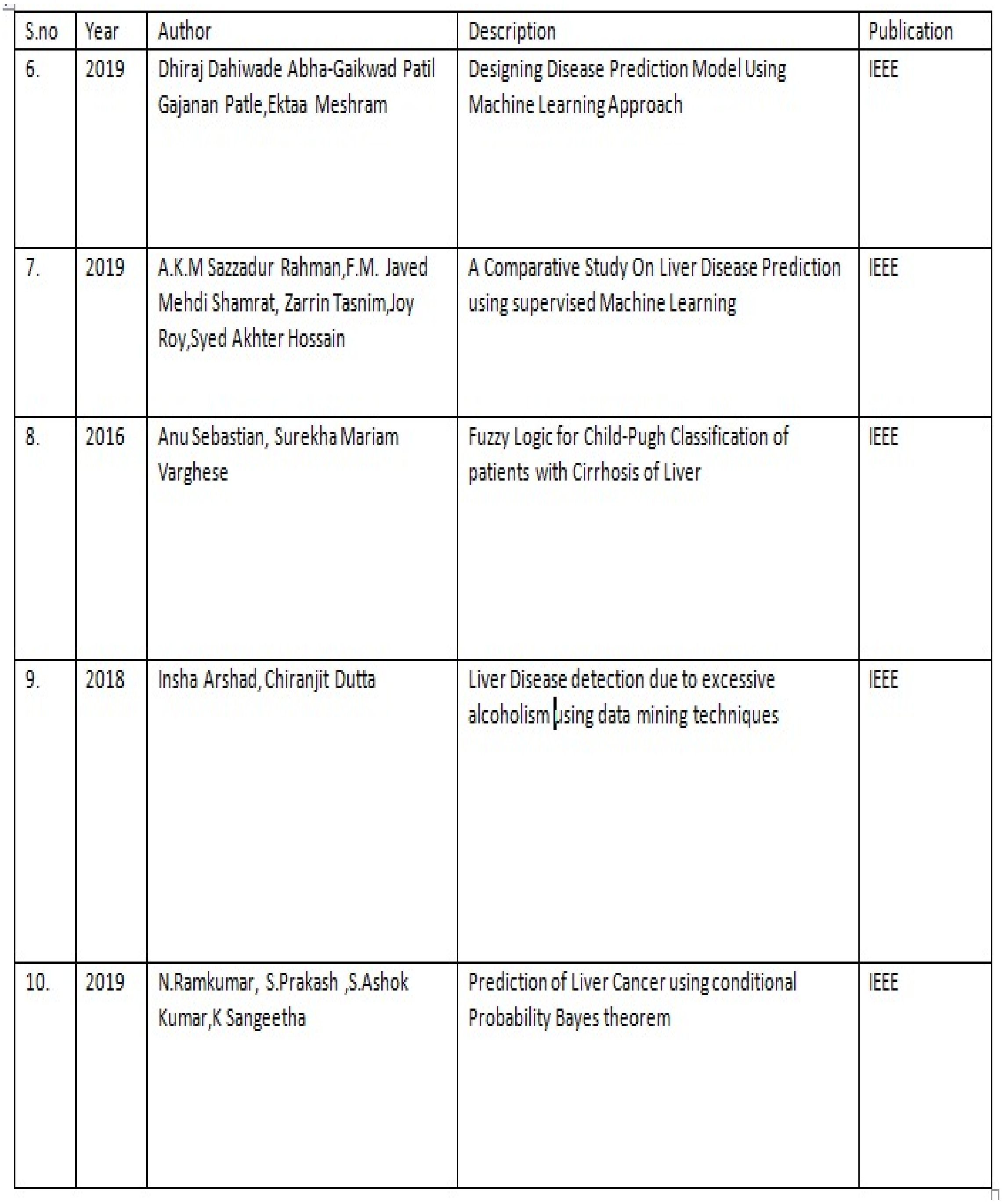
- In some previous research papers, output calculation was done manually which can be done by machine itself nowadays.
- In some research papers, some lengthy algorithms were used like Linear Regression and K-Means ,this can be solved by Neural Networks.
- In some research papers,Artificial Neural Network was used but we have better option as Convolutional Neural Network(CNN).
- In some research rapers, Back Propagation Technique was used which is just like hit and trial method , this can be solved by Convolutional Neural Networks(CNN).

In the proposed system, we must enter the liver patient database. The database should be pre-processed and removed abnormalities and replenish empty cells in the database, so that we can continue to improve the prognosis of liver disease patients, and then use the confusion matrix to find better clarity of positive and negative predictions ROC-AUC model - the model can distinguish between a positive point and a negative point. We will use AUC to select the best model among the learning models of the various machines.



S.no	Year	Author	Description	Publication
1.	2019	Alexandros Arjmand, Constantinous T. Angeli, Alexandros T.Tzallas	Deep Learning in Liver Biopsies using Neutral Networks	IEEE
2.	2018	L. Alice Auxilliaa	Accuracy Prediction using Machine Learning Techniques for Indian Patient Liver Disease	IEEE
3.	2018	Faizatul Himmah,Riyanto Sigit Tri Harsono	Segmentation of Liver using Abdominal CT Scan to detection Liver Disease Area	IEEE
4.	2019	Takuma Oguri, Naohisa Kamiyama, Sachiyo Noguch	Investigation of a Method for quantifying Diffuse Liver Disease Based on Histogram of Ultrasound Signal-to-Noise Ratio	IEEE
5.	2019	Thirunavukkarasu K.,Ajay S. Singh,Md Irfan	Prediction of Liver Disease using Classification Algorithms	IEEE

Figure 2.1: Literature Survey



S.no	Year	Author	Description	Publication
6.	2019	Dhiraj Dahiwade Abha-Gaikwad Patil Gajanan Patle,Ektaa Meshram	Designing Disease Prediction Model Using Machine Learning Approach	IEEE
7.	2019	A.K.M Sazzadur Rahman,F.M. Javed Mehdi Shamrat, Zarrin Tasnim,Joy Roy,Syed Akhter Hossain	A Comparative Study On Liver Disease Prediction using supervised Machine Learning	IEEE
8.	2016	Anu Sebastian, Surekha Mariam Varghese	Fuzzy Logic for Child-Pugh Classification of patients with Cirrhosis of Liver	IEEE
9.	2018	Insha Arshad, Chiranjit Dutta	Liver Disease detection due to excessive alcoholism using data mining techniques	IEEE
10.	2019	N.Ramkumar, S.Prakash ,S.Ashok Kumar,K Sangeetha	Prediction of Liver Cancer using conditional Probability Bayes theorem	IEEE

Figure 2.2: Literature Survey

CHAPTER 3

ARCHITECTURAL DESIGN

3.1 Dataset:-

We will create a model to use predicting results. Here we will use algorithm for model building and segmentation of data. We analyzed 583 patients Details .All features are float and int data type. This model will predict cases of liver diseases. Data sets so the output value contains '1 'for liver disease and '2' because there is no liver disease for it. Data is downloaded in CSV file format as well the target maintains different parameter values

3.2 Feature Extraction:-

It is a process for identifying key features of data sets. We will use liver disease dataset from scikit-learn. This data was collected in the north-eastern state of Andhra Pradesh containing details of 167 non-patient patients and 416 liver patient records. The "Dataset" specificity is a combined category label to distinguish a group into a liver patient (liver disease) or not (no disease). This database contains 441 male patient data and 142 female patient data. Any person over the age of 89 is marked as "90".

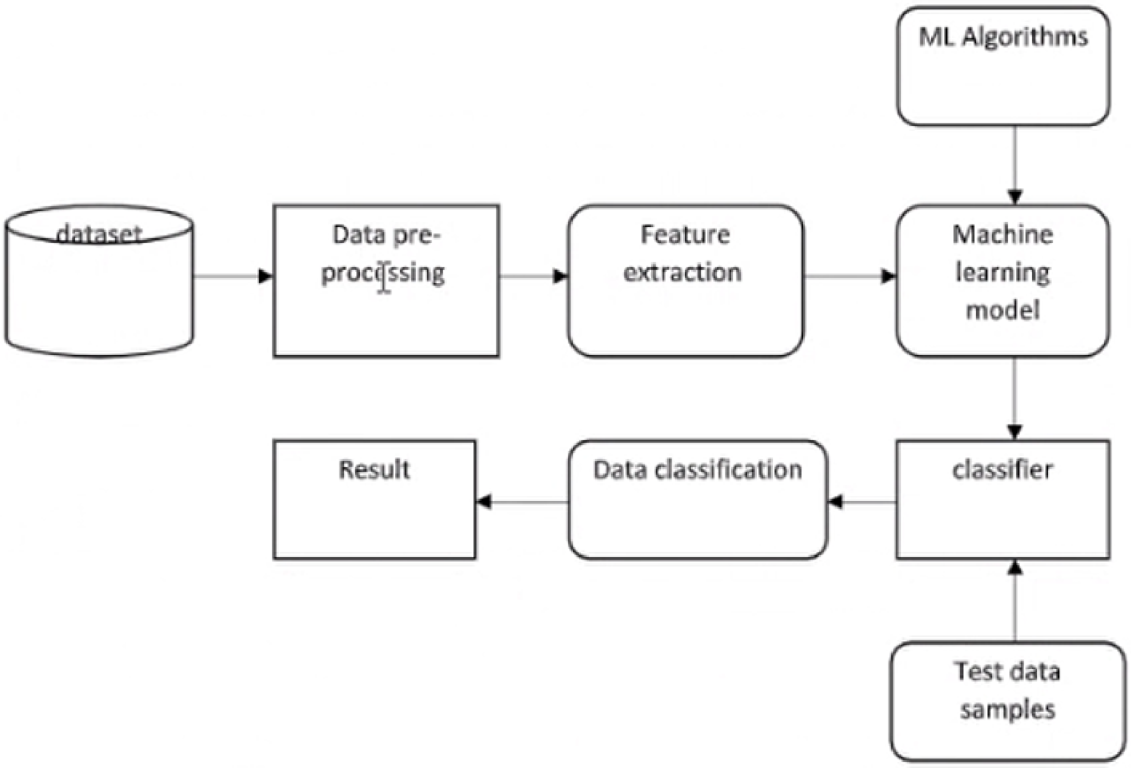


Figure 3.1: Architectural Design of the project

3.3 Attribute Information:-

- Age of the patient
- Gender of the patient
- Total Biliburin
- Direct Biliburin
- Alkaline Phosphotase
- Alamine Aminotransferase
- Aspartate Aminotransferase
- Total Proteins
- Albumin
- Albumin and Globulin Ratio

3.4 Training Stage:-

In this stage, dataset is first converted into numeric values, which are extracted features of cell ,then 80 percent of data is trained with the algorithm used in the process.

3.5 Classification:-

Data is classified into various diseases with the help algorithm and then it is tested with that dataset.

3.5.1 Logistic Regression:-

LR stands for logistic Regression and it will calculated the calculations which are used halfway mainly on traditional experiments and practices from the middle of the 20th century. The asset regression will affect the range of certain analytical factors but clearly, in addition, establishes a separate element elsewhere within the range of zero and one and combine simple preparatory work.

3.5.2 Random Forest :-

Unplanned forest is a method of reversal, segregation, and completely different activities that work by creating a large area of deciduous trees during training and providing the team which is the process of planning or relocating unrelated trees. There has been a rapid relationship between the mixed trees and the effect it will have to encourage continuous effective and accurate predictions, the random forest puts an extra asymmetric sheet to keep it away.

3.5.3 Decision Tree:-

Decision Tree computation contains a site with different learning activities. Contrary to the various supervised learning algorithms, the selected tree rule will be used to address the problems of further procrastination and segregation. A very basic/common approach to the idea is to use Call Tree to create a training model that can be used to predict category or objective limitations by taking selection criteria based on previous information.

3.5.4 Support Vector Machine

Support vector machine will be used for each grouping or relapse problem but typically it's used in characterization problems. SVM performs commendably for a few, human services problems and might comprehend each linear and non-linear problem. SVM classifying approach is an associate degree attempt to role a linearly divisible hyper-plane to arrange the data-set into 2 groups. At enduring, the classifier will without a doubt gauge the target teams for a variety of upcoming cases.

3.5.5 K- Nearest Neighbours:-

KNN is a basic moment-based classification algorithm in Machine Learning. Anyhow, the KNN lay hold of a trial at the idea that illustrates are close to slot in likewise illustration group. KNN types linked degree illustrates to the class that's most resolute between K neighboring. KNN is a restriction to altering the different categorization algorithms.

3.5.6 Naive Bayes:-

The Naive Bayes are one of the most important, leading, and frequently used routes. A possible distinction that separates the use of a limited freedom situation with previously trained data sets. From this point on, Native statistical categories are a means of gaining traditional familiarity with collection problems, for example, spam detection, and appropriate medical problems.

3.6 Testing Stage

We will provide input from our side as a scikit-learn store data in an object bunch like a dictionary and test the dataset and the confidence of the output is its efficiency.

CHAPTER 4

METHODOLOGY

The principal of this analysis is to forsee and quantify disease of liver through the effective management of a selection of strategies and class algorithms for efficiency. Also, compare the results of each selection factor and while it is not a selection process in completely different classifiers for survival that gives us more well-differentiated results for getting liver disease. Classifiers are forced to abuse Ten fold cross confirmation test options in the liver disease database. The feature selection process is forced to reduce the qualification from the liver disease database to obtain higher results at shorter duration of action.

4.1 Database Selection

The data was collected from the North - Eastern state of Andhra Pradesh via Kaggle. The "Dataset" specificity is a combined category label to distinguish a group into a liver patient (liver disease) or not (no disease). This database contains 441 male patient data and 142 female patient data.

4.2 Feature Selection

The patient's liver database is stored in the attribute relation file format. The data was processed in a suitable format for the use of various separators. It is also necessary to delete unnecessary field, lost records and duplicate records if there is a database editing in the most common format.

4.3 Pre-processing and Transformation

The patient's liver database is stored in the attribute relation file format. The data was processed in a suitable format for the use of various separators. jointly it was necessary to eliminate the unnecessary field associated with it, lost records and duplicate records if there is a normal database fix.

4.4 Classifier Implementation:-

We use a variety of layouts such as Logistics Regression, Naive-Bayes, SVC, Random Forest, XG Boost and Decision Tree square measure which is compulsory in the selection of the patient's database data and while not using feature selection. A test opportunity to verify 10 times the hen's tool crossing is used on the database to get the best results.

4.5 Performance Evaluation

We can use the subsequent implementation computation for the demand quantity of defected tilted components as shown by our own demand. We compare the result of different classsifiers like logistic regression, SVC , Naive Bayes, Random Forest.

4.6 Precision of Disease

Based on accurate and well-used classifiers, web-based software development is designed to predict new patient's liver disease.

4.7 Learning and Inference Phase

The core object of the project is learning and inference phase.The machine learn through the uncovering of pattern. The learning phase contain training data, feature vector,algorithm and model. One important part of the data is to choose data to send to the machine. The feature vector is subset of data used for tackling the problem.

4.8 Inferring

When the inferring model is construct, it is attainable to check however powerful it is on never view before information. The latest information square measure reworked into a options vector, bear a model and provides the prediction. This can be all the gorgeous a part of machine learning. there's no have to be compelled to update the principles or train once more the model. we will be able to use a model antecedently trained to form abstract through latest information.

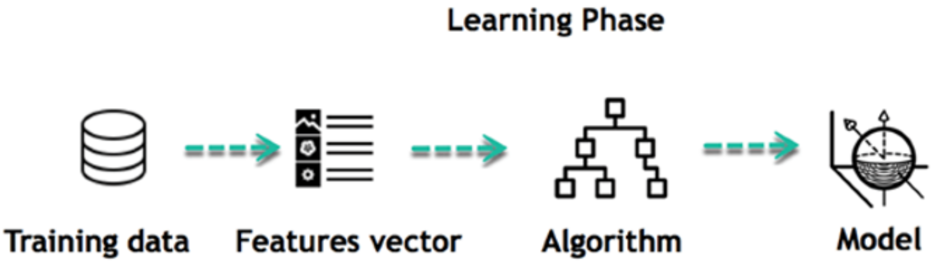


Figure 4.1: Learning Phase

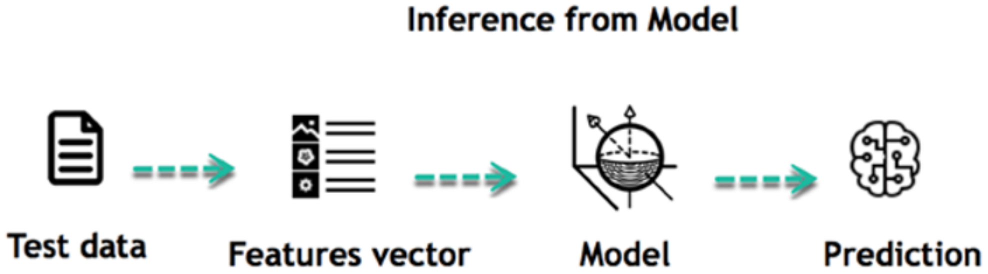
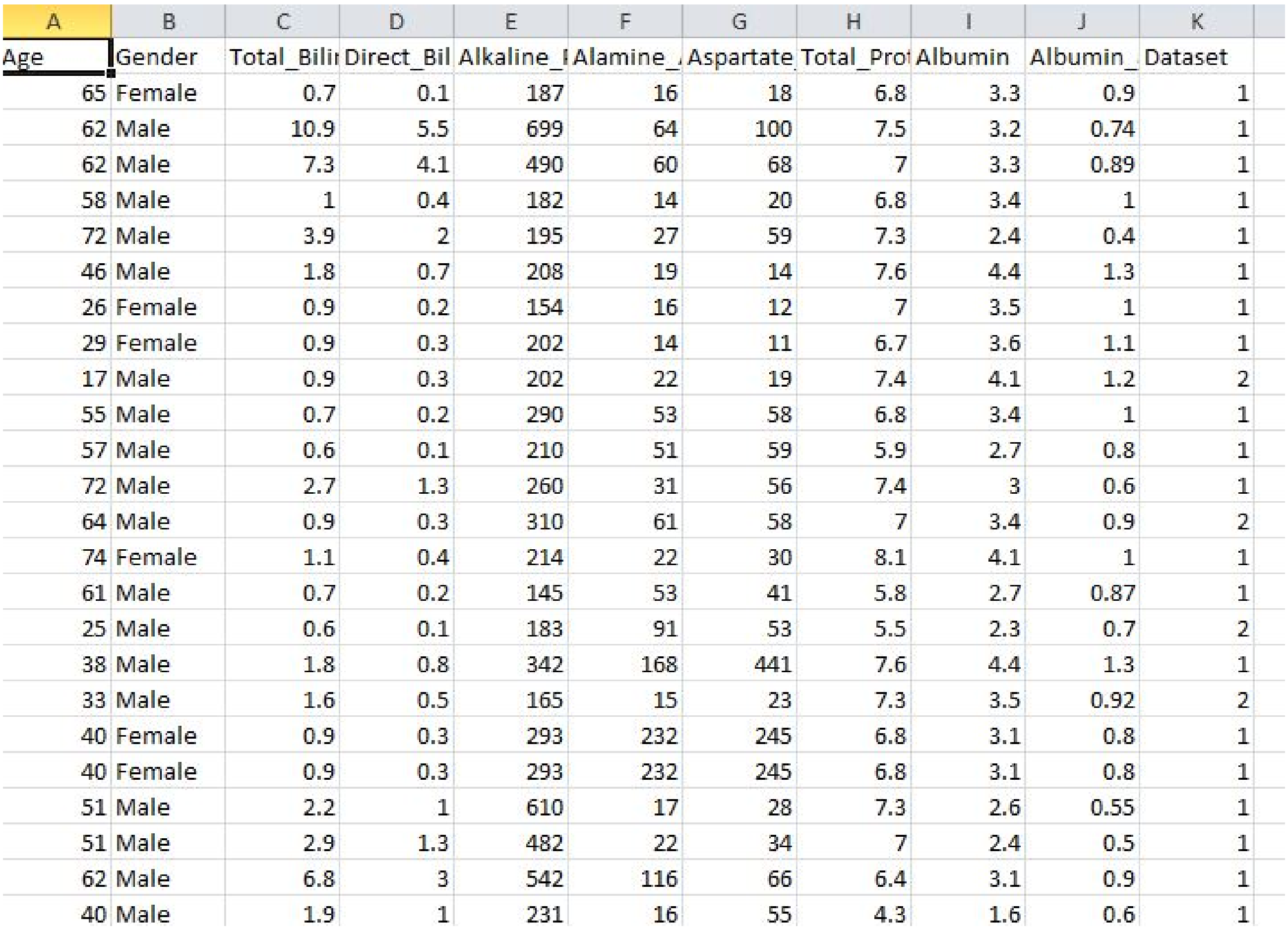


Figure 4.2: Inferring from Model

4.9 Algorithm

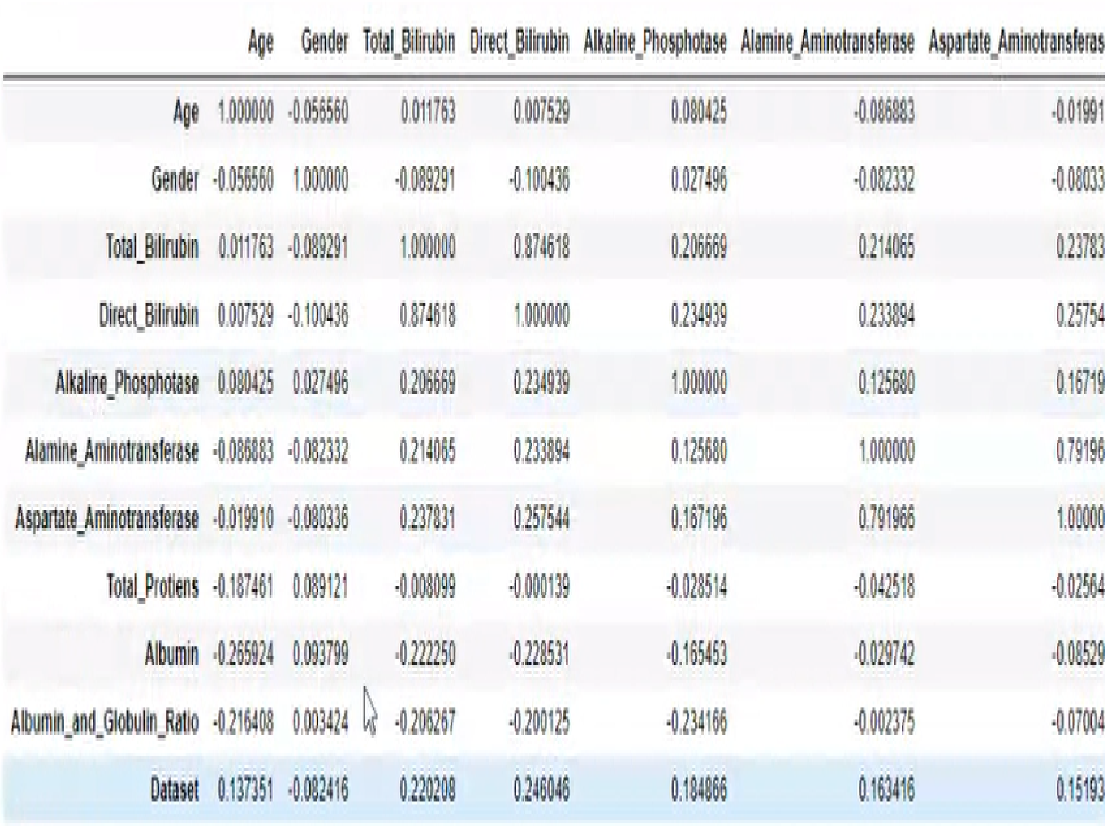
The life of Machine Learning is simple and can be encapsulate in the following points:

- Problem Definition
- Assemble Data
- Envision Data
- Instruct Algorithm
- Check the Algorithm
- Gather Feedback
- Purify the Algorithm
- Loop 4-7 until the output are acceptable
- Model used to predict the output



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

Figure 4.3: Data Visualisation



	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Aminotransferase	Aspartate_Aminotransferas
Age	1.000000	-0.056560	0.011763	0.007529	0.080425	-0.086883	-0.01991
Gender	-0.056560	1.000000	-0.089291	-0.100436	0.027496	-0.082332	-0.08033
Total_Bilirubin	0.011763	-0.089291	1.000000	0.874618	0.206669	0.214065	0.23783
Direct_Bilirubin	0.007529	-0.100436	0.874618	1.000000	0.234939	0.233894	0.25754
Alkaline_Phosphotase	0.080425	0.027496	0.206669	0.234939	1.000000	0.125680	0.16719
Alamine_Aminotransferase	-0.086883	-0.082332	0.214065	0.233894	0.125680	1.000000	0.79196
Aspartate_Aminotransferase	-0.019910	-0.080336	0.237831	0.257544	0.167196	0.791966	1.00000
Total_Protiens	-0.187461	0.089121	-0.008099	-0.000139	-0.028514	-0.042518	-0.02564
Albumin	-0.265924	0.093799	-0.222250	-0.228531	-0.165453	-0.029742	-0.08529
Albumin_and_Globulin_Ratio	-0.216408	0.003424	-0.206267	-0.200125	-0.234166	-0.002375	-0.07004
Dataset	0.137351	-0.082416	0.220208	0.246046	0.184866	0.163416	0.15193

Figure 4.4: Features

CHAPTER 5

PERFORMANCE EVALUATION

5.1 Mean Absolute Error

MAE measures the classic magnitude of errors in an exceedingly predictions set, without in view of their direction. It's the typical above take a look at sample of absolutely the variations between predicted observation and real observation wherever each person variations have same weight.

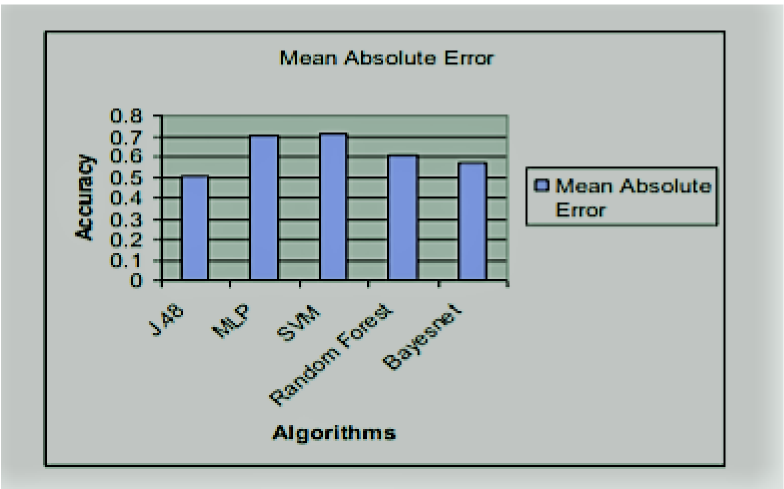


Figure 5.1: Mean Absolute Error Rate

5.2 Root Mean Square Error

It is a primarily used to measure the differences among the predict value and observed value. It gave the average value which is taken as a input in the the project.

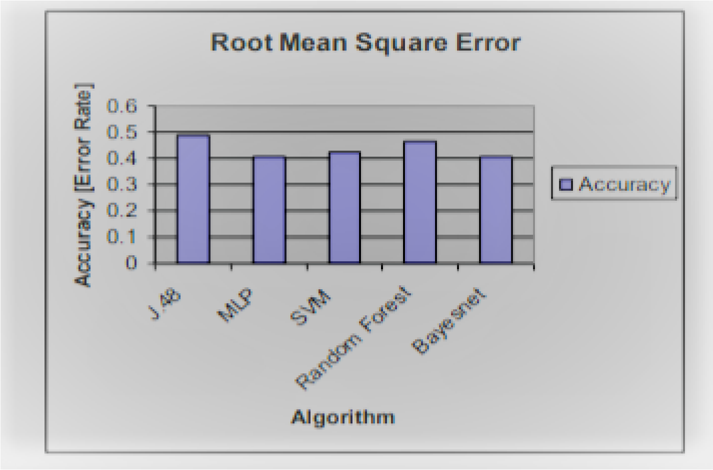
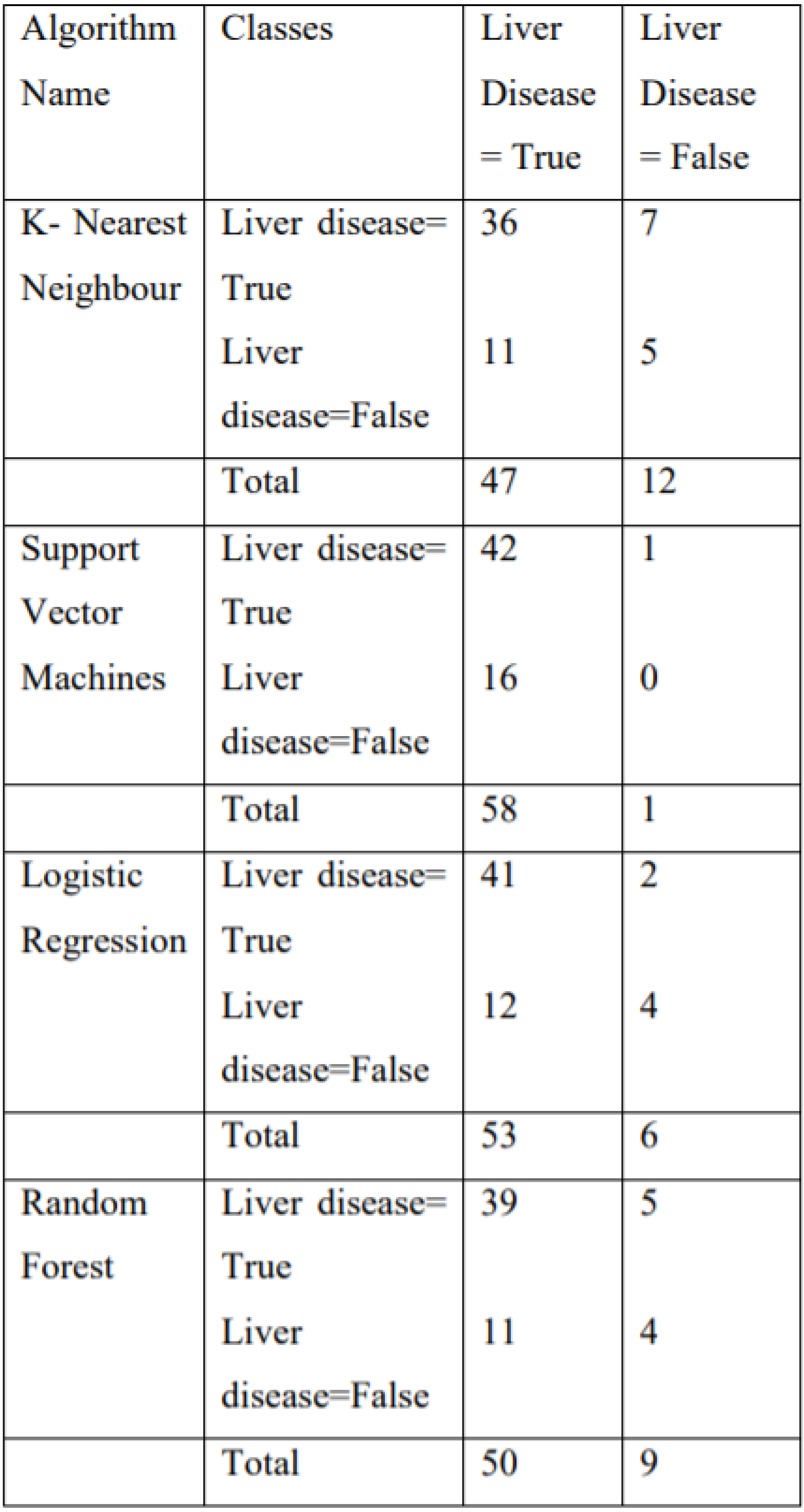


Figure 5.2: Root Mean Square Error

5.3 Confusion Matrix

The confusion matrix is used to calculate the formation of the size of the data group of a given education. The appropriate category for result which is true and postive or the one which is true but negative piece reports adequately reports examples with FP (false) and FN (false) Invalid Request Conditions. The Confusion Matrix accurately classifies Instance TP + TN randomly setting the conditions. The real benefits include the courage of a well-meaning couple.



Algorithm Name	Classes	Liver Disease = True	Liver Disease = False
K- Nearest Neighbour	Liver disease= True	36	7
	Liver disease=False	11	5
	Total	47	12
Support Vector Machines	Liver disease= True	42	1
	Liver disease=False	16	0
	Total	58	1
Logistic Regression	Liver disease= True	41	2
	Liver disease=False	12	4
	Total	53	6
Random Forest	Liver disease= True	39	5
	Liver disease=False	11	4
	Total	50	9

Figure 5.3: Confusion Matrix of Various Algorithm

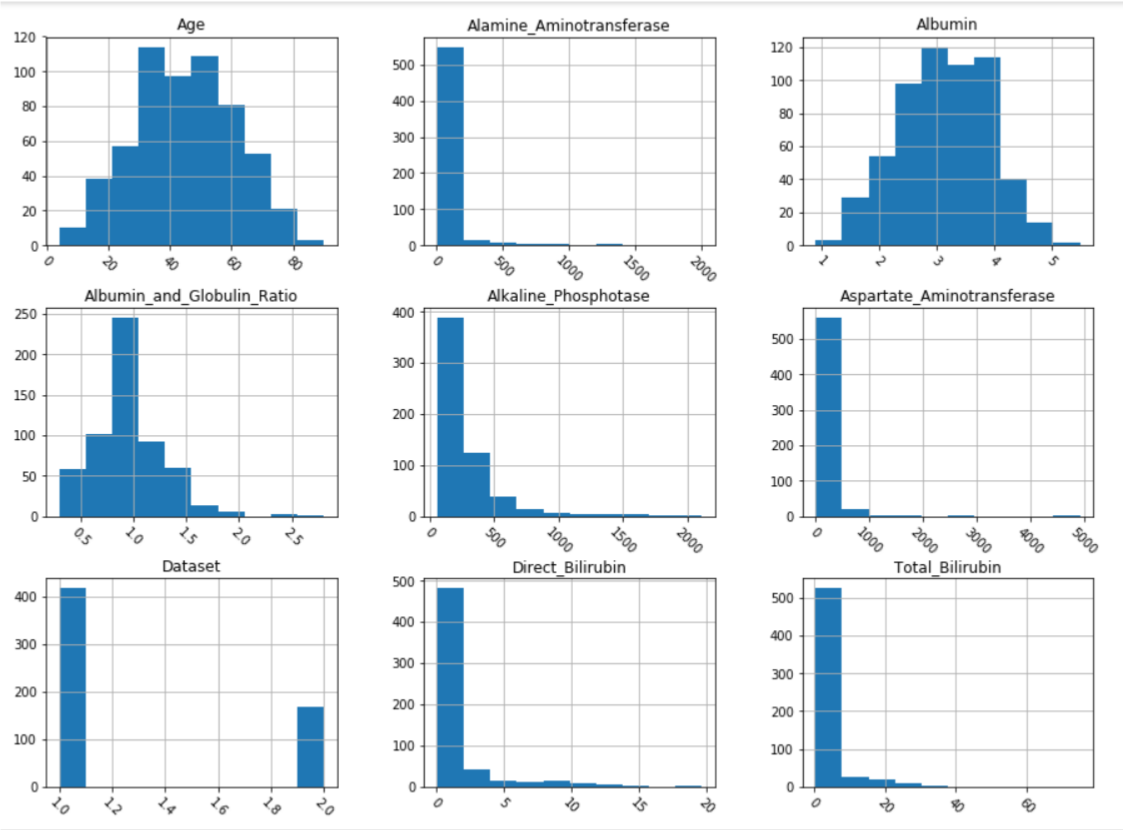


Figure 5.4: Distribution of Numerical Feature

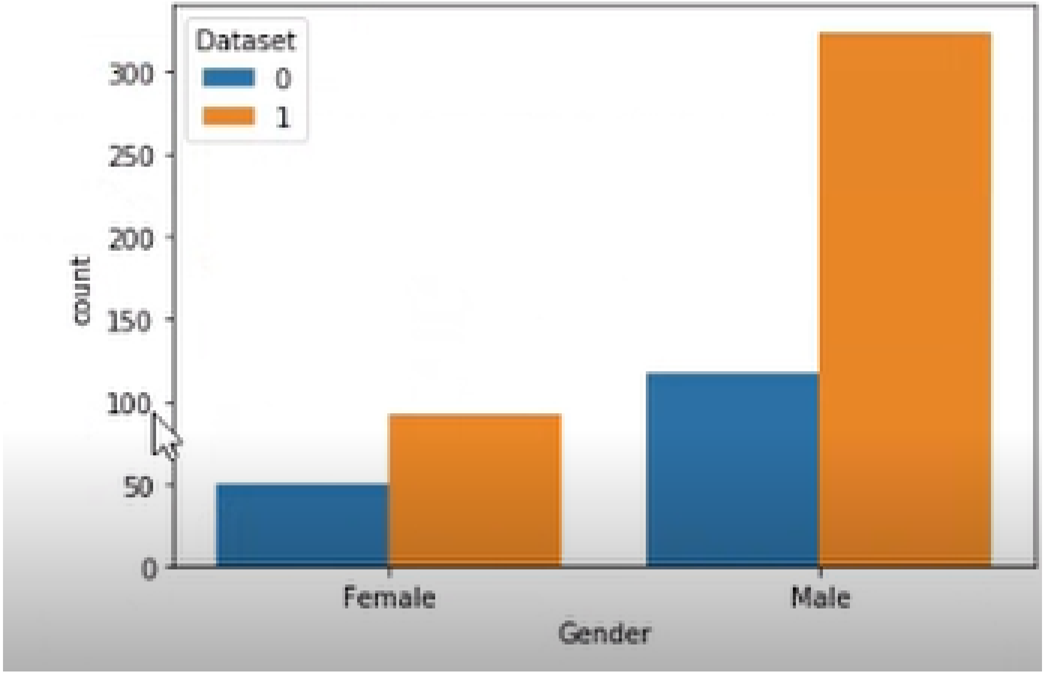


Figure 5.5: Bar Plot of Categorical Feature

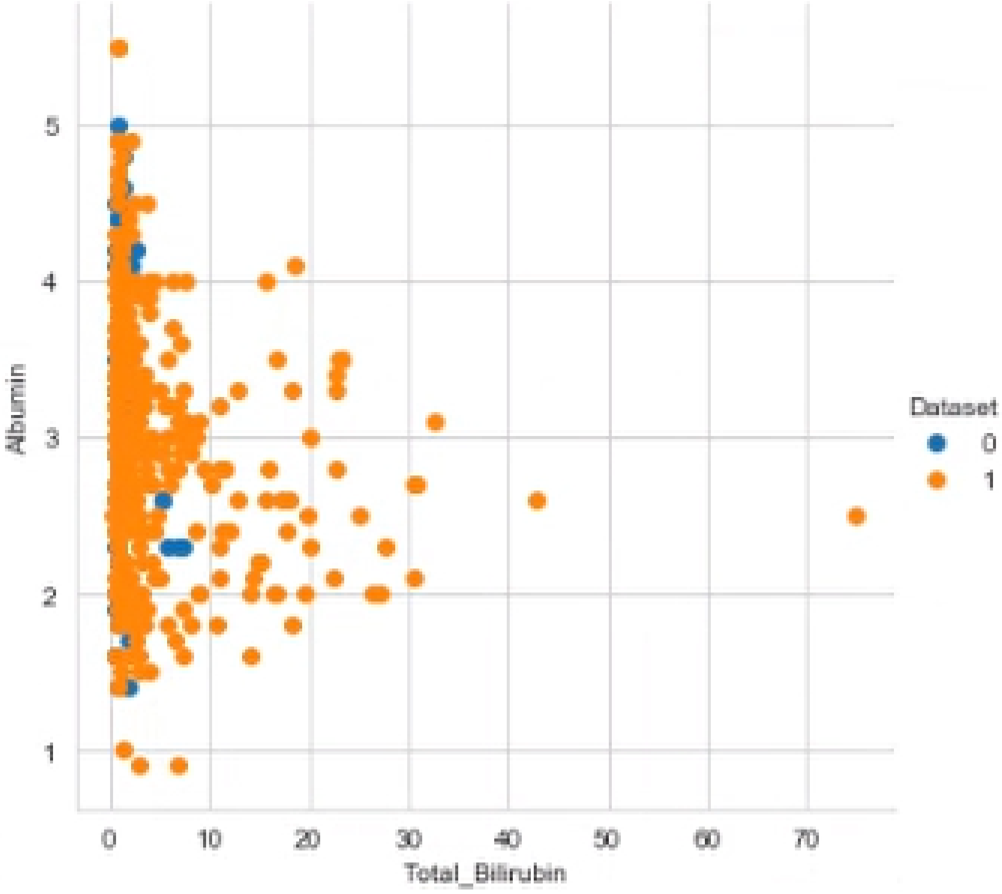
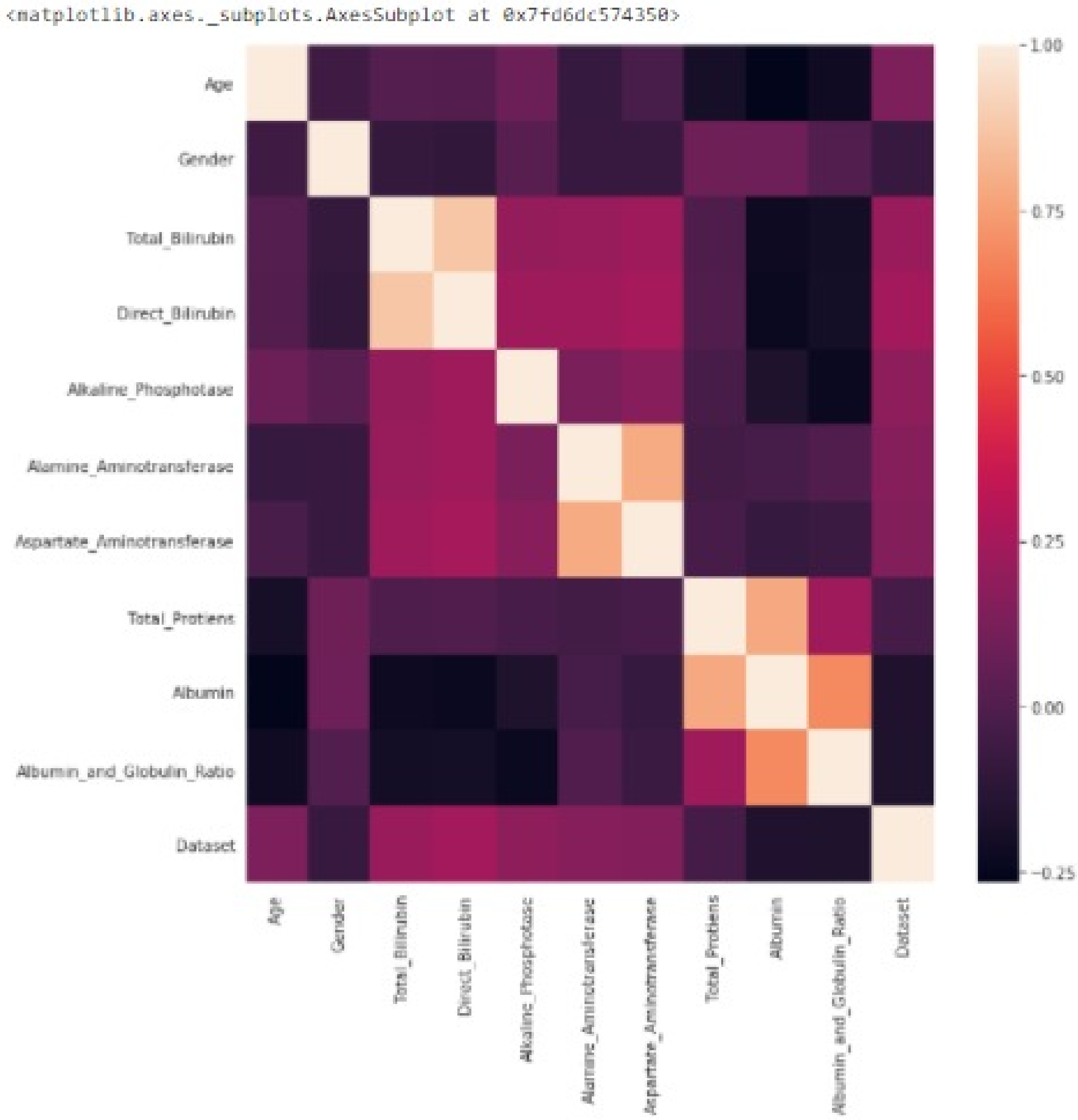


Figure 5.6: 2-D Scatter Plot



```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd6dc574350>
```

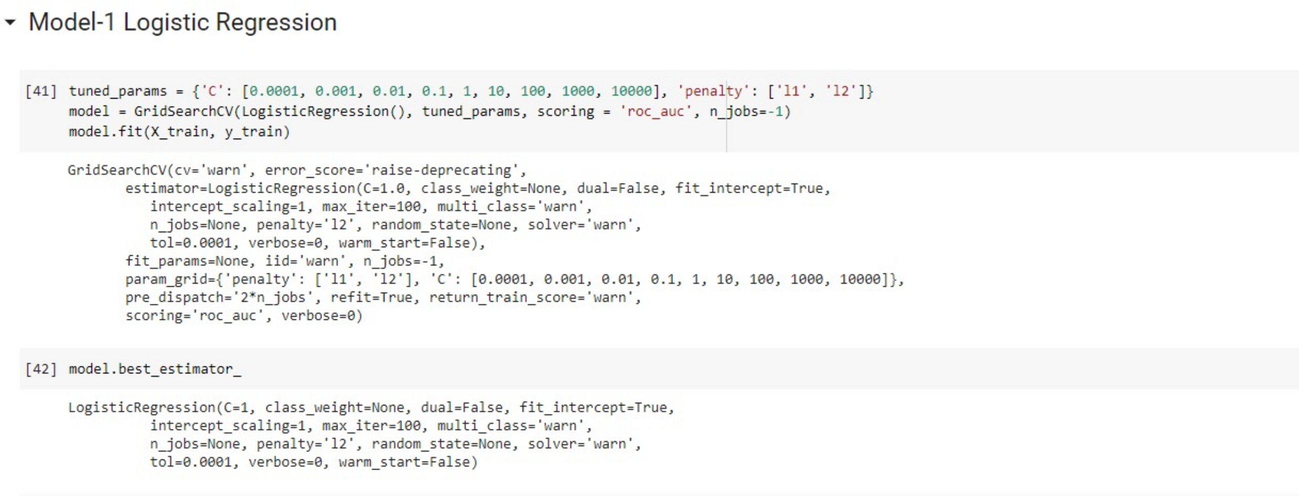
Figure 5.7: Correlation of different parameters

CHAPTER 6

CODING TESTING

In coding phase , We will code our project in Jupyter Notebook python interprete. We will apply all modules in the code first , then we will load our dataset in the python then create the model of the dataset then we will test it.It is a procedure of identifying whether the model is working accurately. In this procedure we have to identify the error and reconstruct the code. A document named test case that consist of tested data and expected result which are generated for a specific kind of test. We are using clean data to build the Machine learning model. The output is in a categorical format as we are using different type of supervised classification machine learnin algorithms. To build the model, we have trained and tested the dataset with multiple Machine Learning algorithms then on the basis of output percentage we have find the best ML model.

1. Logistic Regression
2. Random Forest
3. Decision Tree Classifier
4. SVC
5. Gradient Boosting



Model-1 Logistic Regression

```
[41] tuned_params = {'C': [0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000], 'penalty': ['l1', 'l2']}
     model = GridSearchCV(LogisticRegression(), tuned_params, scoring = 'roc_auc', n_jobs=-1)
     model.fit(X_train, y_train)

     GridSearchCV(cv='warn', error_score='raise-deprecating',
            estimator=LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
               intercept_scaling=1, max_iter=100, multi_class='warn',
               n_jobs=None, penalty='l2', random_state=None, solver='warn',
               tol=0.0001, verbose=0, warm_start=False),
            fit_params=None, iid='warn', n_jobs=-1,
            param_grid={'penalty': ['l1', 'l2'], 'C': [0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000]},
            pre_dispatch='2*n_jobs', refit=True, return_train_score='warn',
            scoring='roc_auc', verbose=0)

[42] model.best_estimator_

     LogisticRegression(C=1, class_weight=None, dual=False, fit_intercept=True,
               intercept_scaling=1, max_iter=100, multi_class='warn',
               n_jobs=None, penalty='l2', random_state=None, solver='warn',
               tol=0.0001, verbose=0, warm_start=False)
```

Figure 6.1: Logistic Regression

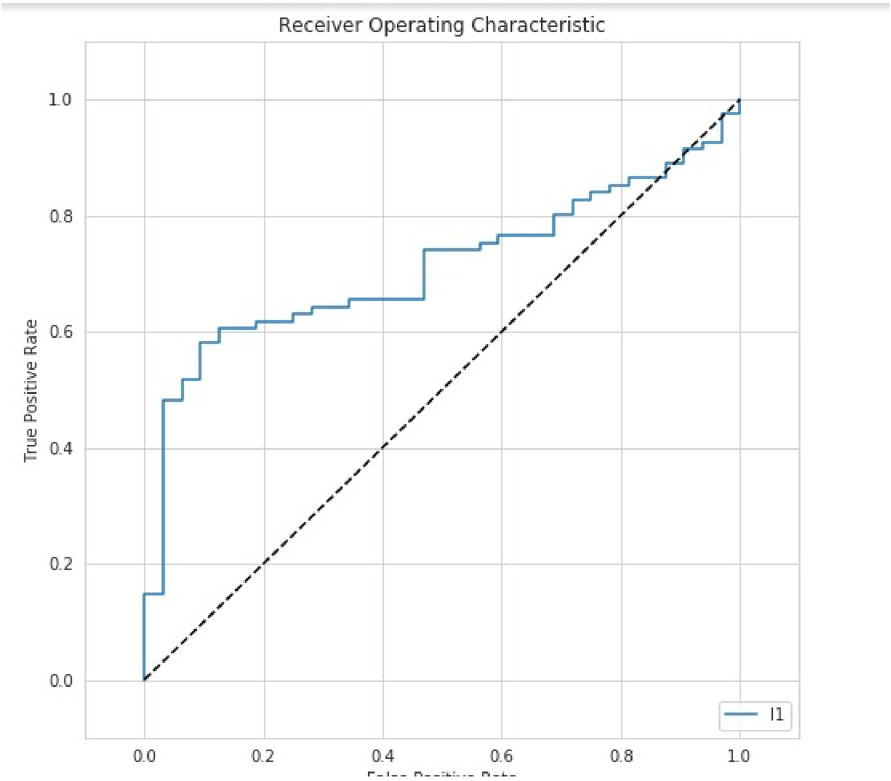
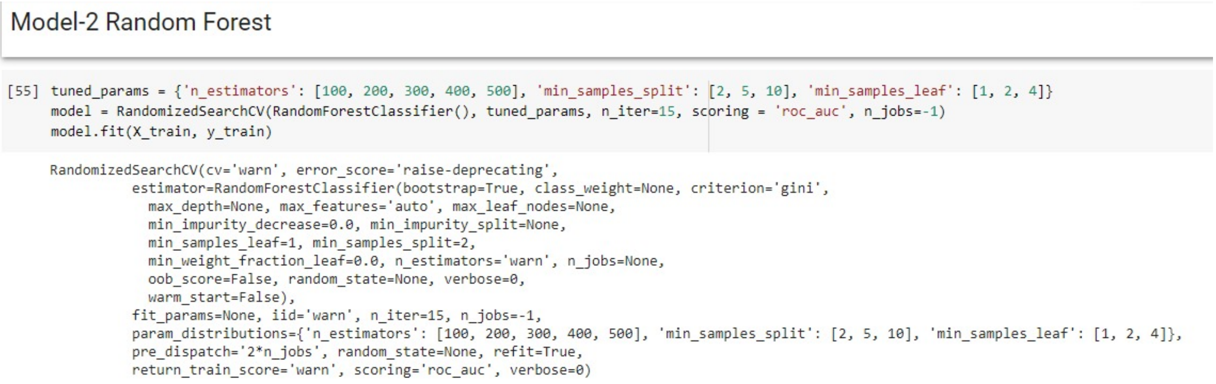


Figure 6.2: ROC of Logistic Regression



Model-2 Random Forest

```
[55] tuned_params = {'n_estimators': [100, 200, 300, 400, 500], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4]}
     model = RandomizedSearchCV(RandomForestClassifier(), tuned_params, n_iter=15, scoring = 'roc_auc', n_jobs=-1)
     model.fit(X_train, y_train)

     RandomizedSearchCV(cv='warn', error_score='raise-deprecating',
               estimator=RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
                 max_depth=None, max_features='auto', max_leaf_nodes=None,
                 min_impurity_decrease=0.0, min_impurity_split=None,
                 min_samples_leaf=1, min_samples_split=2,
                 min_weight_fraction_leaf=0.0, n_estimators='warn', n_jobs=None,
                 oob_score=False, random_state=None, verbose=0,
                 warm_start=False),
               fit_params=None, iid='warn', n_iter=15, n_jobs=-1,
               param_distributions={'n_estimators': [100, 200, 300, 400, 500], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4]},
               pre_dispatch='2*n_jobs', random_state=None, refit=True,
               return_train_score='warn', scoring='roc_auc', verbose=0)
```

Figure 6.3: Random Forest

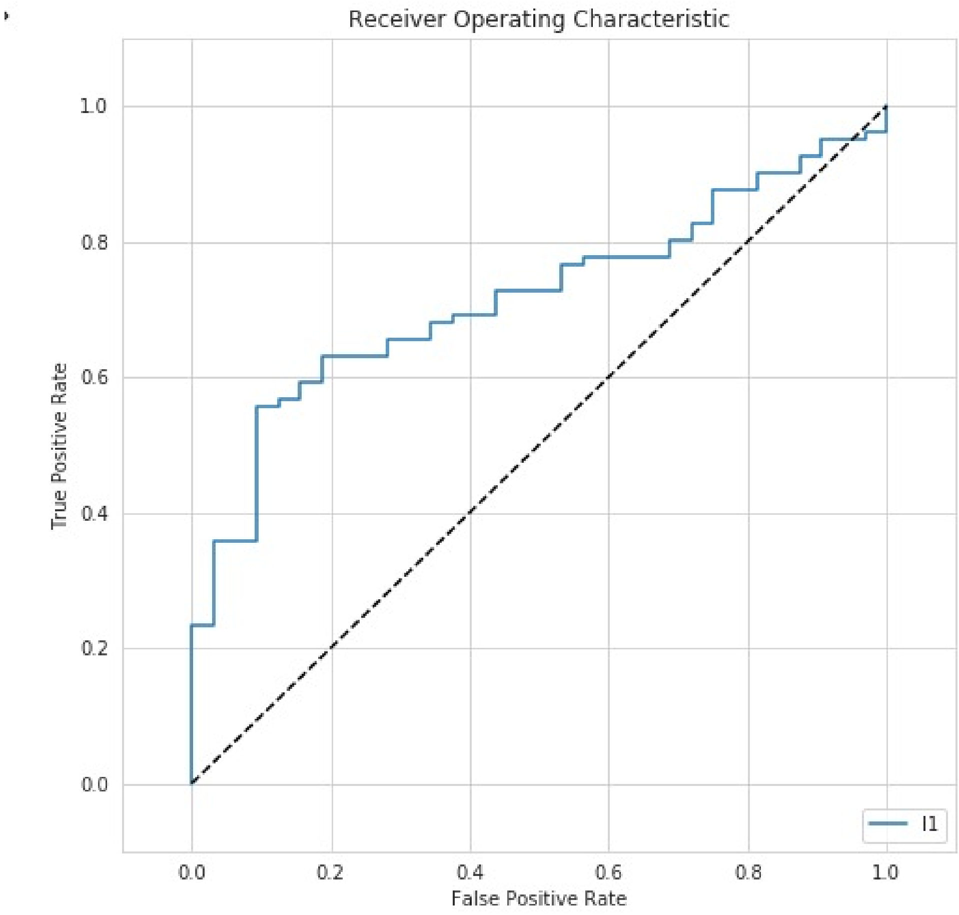
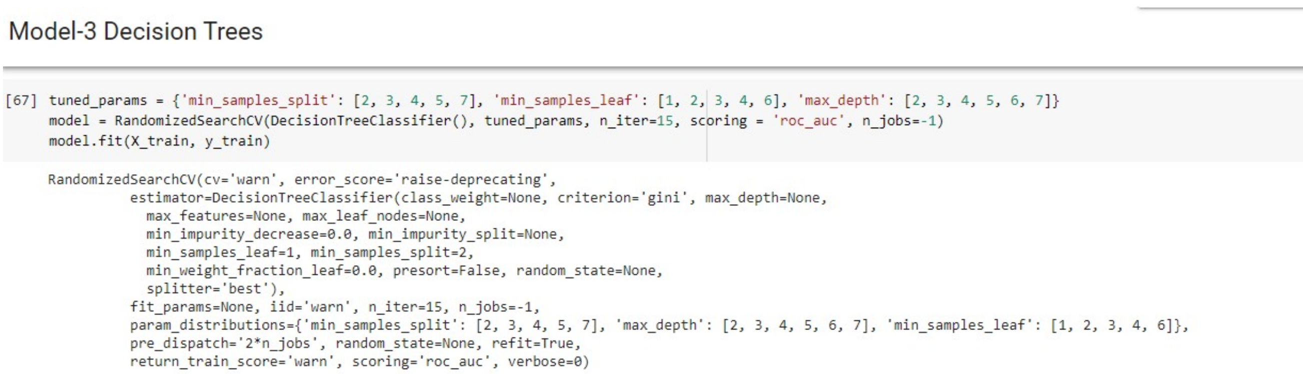


Figure 6.4: ROC of Random Forest



Model-3 Decision Trees

```
[67] tuned_params = {'min_samples_split': [2, 3, 4, 5, 7], 'min_samples_leaf': [1, 2, 3, 4, 6], 'max_depth': [2, 3, 4, 5, 6, 7]}
     model = RandomizedSearchCV(DecisionTreeClassifier(), tuned_params, n_iter=15, scoring = 'roc_auc', n_jobs=-1)
     model.fit(X_train, y_train)
```

```
RandomizedSearchCV(cv='warn', error_score='raise-deprecating',
          estimator=DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
            max_features=None, max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, presort=False, random_state=None,
            splitter='best'),
          fit_params=None, iid='warn', n_iter=15, n_jobs=-1,
          param_distributions={'min_samples_split': [2, 3, 4, 5, 7], 'max_depth': [2, 3, 4, 5, 6, 7], 'min_samples_leaf': [1, 2, 3, 4, 6]},
          pre_dispatch='2*n_jobs', random_state=None, refit=True,
          return_train_score='warn', scoring='roc_auc', verbose=0)
```

Figure 6.5: Decision Tree Classifier

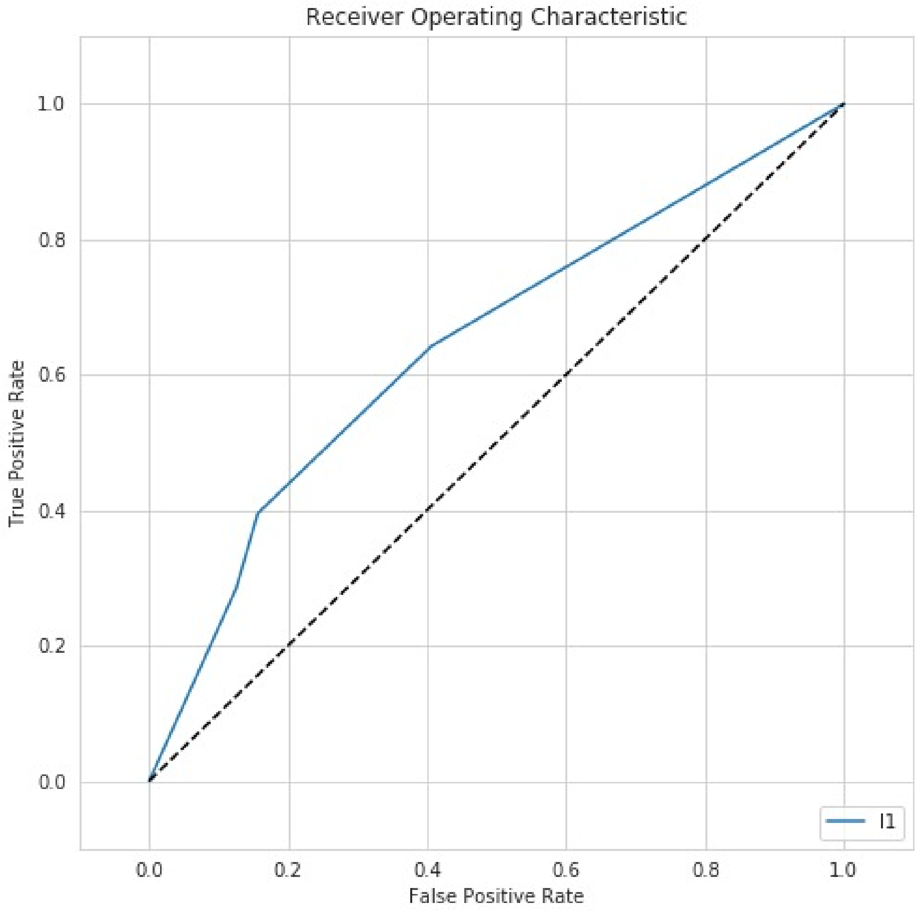
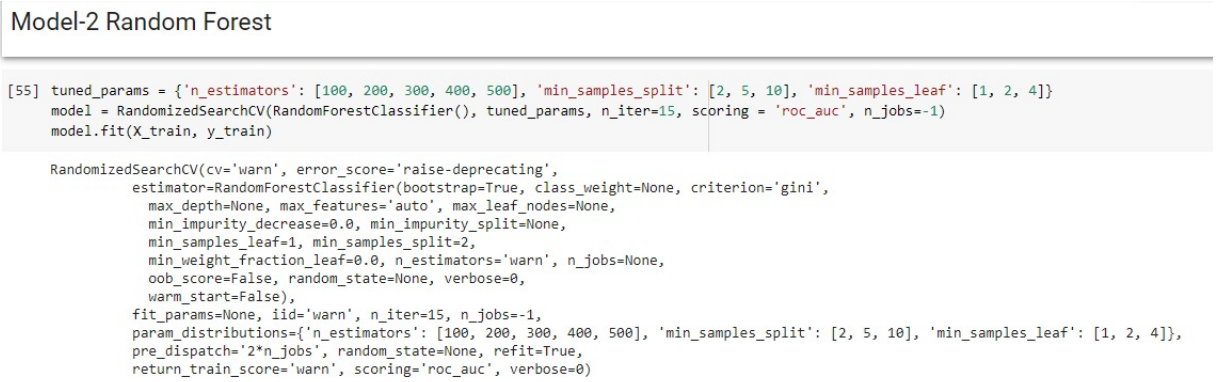


Figure 6.6: Decision Tree Classifier



Model-2 Random Forest

```
[55] tuned_params = {'n_estimators': [100, 200, 300, 400, 500], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4]}
     model = RandomizedSearchCV(RandomForestClassifier(), tuned_params, n_iter=15, scoring = 'roc_auc', n_jobs=-1)
     model.fit(X_train, y_train)

     RandomizedSearchCV(cv='warn', error_score='raise-deprecating',
               estimator=RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
                 max_depth=None, max_features='auto', max_leaf_nodes=None,
                 min_impurity_decrease=0.0, min_impurity_split=None,
                 min_samples_leaf=1, min_samples_split=2,
                 min_weight_fraction_leaf=0.0, n_estimators='warn', n_jobs=None,
                 oob_score=False, random_state=None, verbose=0,
                 warm_start=False),
               fit_params=None, iid='warn', n_iter=15, n_jobs=-1,
               param_distributions={'n_estimators': [100, 200, 300, 400, 500], 'min_samples_split': [2, 5, 10], 'min_samples_leaf': [1, 2, 4]},
               pre_dispatch='2*n_jobs', random_state=None, refit=True,
               return_train_score='warn', scoring='roc_auc', verbose=0)
```

Figure 6.7: SVC

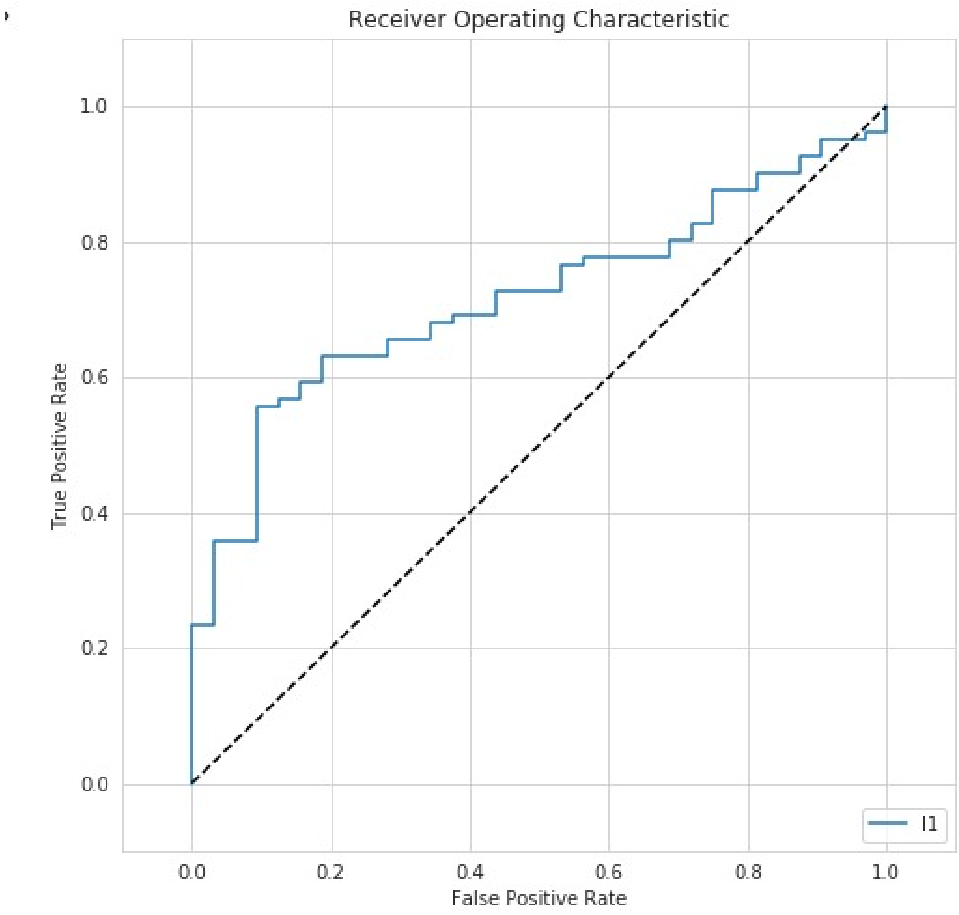
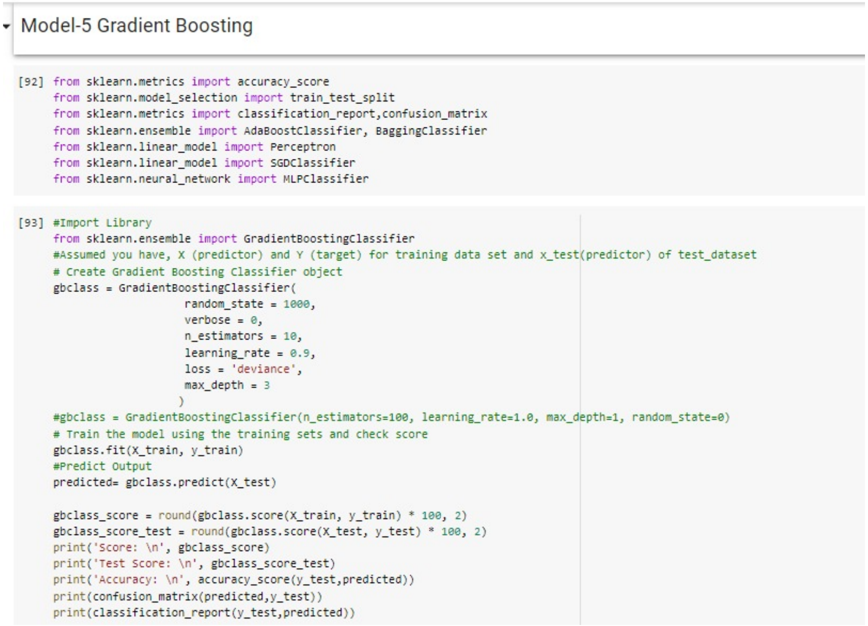


Figure 6.8: ROC of SVC



Model-5 Gradient Boosting

```
[92] from sklearn.metrics import accuracy_score
     from sklearn.model_selection import train_test_split
     from sklearn.metrics import classification_report,confusion_matrix
     from sklearn.ensemble import AdaBoostClassifier, BaggingClassifier
     from sklearn.linear_model import Perceptron
     from sklearn.linear_model import SGDClassifier
     from sklearn.neural_network import MLPClassifier
```

```
[93] #Import Library
     from sklearn.ensemble import GradientBoostingClassifier
     #Assumed you have, X (predictor) and Y (target) for training data set and x_test(predictor) of test_dataset
     # Create Gradient Boosting Classifier object
     gbclass = GradientBoostingClassifier(
                         random_state = 1000,
                         verbose = 0,
                         n_estimators = 10,
                         learning_rate = 0.9,
                         loss = 'deviance',
                         max_depth = 3
                        )
     #gbclass = GradientBoostingClassifier(n_estimators=100, learning_rate=1.0, max_depth=1, random_state=0)
     # Train the model using the training sets and check score
     gbclass.fit(X_train, y_train)
     #Predict Output
     predicted= gbclass.predict(X_test)

     gbclass_score = round(gbclass.score(X_train, y_train) * 100, 2)
     gbclass_score_test = round(gbclass.score(X_test, y_test) * 100, 2)
     print('Score: \n', gbclass_score)
     print('Test Score: \n', gbclass_score_test)
     print('Accuracy: \n', accuracy_score(y_test,predicted))
     print(confusion_matrix(predicted,y_test))
     print(classification_report(y_test,predicted))
```

Figure 6.9: Gradient Boosting

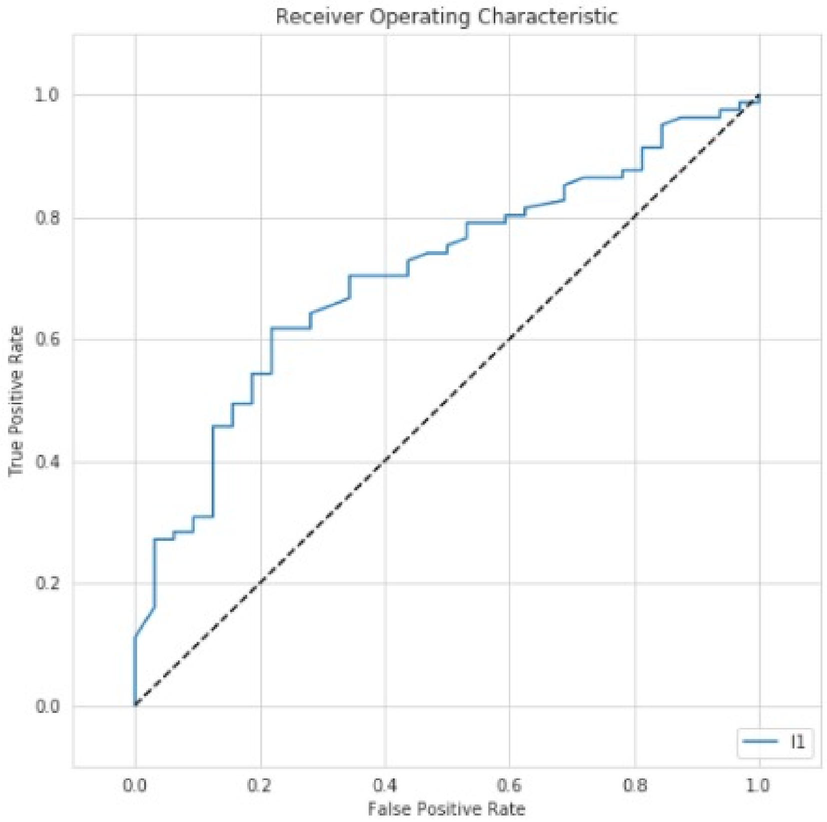


Figure 6.10: ROC of Gradient Boosting

CHAPTER 7

CONCLUSION

In conclusion, the use of statistics breaks down predictive assessment systems is significant in the social context because it empowers us to deal with past disruptions and equally preserves human life in the belief of correction. In project, we use a few statistics to study KNN, SVM, Logistic Regression, Random Forest and Naive bayes to anticipate the patient of liver with chronic hepatitis B infection, as well as patients who do not develop the disease. The results of the redesign show that the classification of the system shows its presentation in anticipation of excellent results that respect the accuracy and short operating time.

CHAPTER 8

FUTURE ENHANCEMENT

In our future work , We will try to improve the efficiency of the project as compared to current obtained efficiency. We will try some other algorithms that will come in the future for the advancement of the technology. We will apply that later.

CHAPTER 9

REFERENCES

1. Alexandros Arjmand, Constantinos T.Angeli, Alexandros T. Tzallas, Deep Learning in Liver Biopsies using Neural Networks,2019.
2. L. Alice Auxilia, Accuracy Prediction using Machine Learning Techniques for Indian Patient Liver 2018.
3. Faizatul Himmah, Riyanto Sigit, Tri Harsono, Segmentation of Liver using Abdominal CT Scan to Detection Liver Disease Area,2018.
4. Takuma Oguri, Naohisa Kamiyama, Sachiyo Noguchi, Investigation of a Method for quantifiying Diffuse Liver Disease Based on Histogram of Ultrasound Signal-to-Noise Ratio, 2019.
5. Thirunavukkarasu K, Ajay S. Singh, Md Irfan, Prediction of Liver Disease using Classification Algorithms,2018
6. Dhiraj Dahiwade Abha-Gaikwad Patil Gajanan Patle, Ektaa Meshram, Designing Disease Prediction Model Using Machine Learning Approach,2019.
7. A.K.M Sazzadur Rahman,F.M.Javed Mehedi Shamrat, Zarrin Tasnim, Joy Roy, Syed Akhter Hossain, A Comparative Study On Liver Disease Prediction Using Supervised Machine Learning Algorithms, 2019.
8. Anu Sebastian, Surekha Mariam Varghese, Fuzzy Logic for Child-Pugh classification of patients with Cirrhosis of Liver,2017.
9. . Insha Arshad , Chiranjit Dutta, Liver Disease Detection Due to Excessive Alcoholism Using Data Mining Techniques,2018
10. N. Ramkumar ; S. Prakash ; S. Ashok Kumar ; K Sangeetha, : Prediction of liver cancer using Conditional probability Bayes theorem,2017