Liver cirrhosis prediction

Sina Alba

December 2021

Abstract

The liver is considered an essential organ in the human body.Liver disorders have risen globally at an unprecedented pace due to unhealthy lifestyles and excessive alcohol consumption.Chronic liver disease is one of the principal causes of death affecting large portions of the global population especially in North America.Obesity, an undiagnosed hepatitis infection, alcohol abuse, coughing or vomiting blood, kidney or hepatic failure, jaundice, liver encephalopathy, and many more disorders are responsible for it. Thus, immediate intervention is needed to diagnose the ailment before it is too late.Therefore, this work aims to evaluate several machine learning algorithm outputs, namely logistic regression, K-nearest neighbors (K-NN), support vector machine (SVM), random forest, logistic regression with cross validation, and XGBoost for predicting and diagnosing chronic liver disease. The classification algorithms are evaluated based on various measurement criteria, such as accuracy, precision, recall, F1 score and an area under the curve (AUC). Among the algorithms, the XGboost algorithm showed better performance in liver disease prediction with an accuracy of 83.33%. Furthermore, Random Forest also showed noticeable metrics after XGboost. To put everything into a nutshell, XGboost is considered the best algorithm for early liver disease prediction.

Keywords: Machine learning; Cirrhosis; classification; over-sampling; random forest; XGboost

1 Introduction

Cirrhosis is a late-stage liver disease in which healthy liver tissue is replaced with scar tissue and the liver is permanently damaged. Scar tissue keeps your liver from working properly.

Many types of liver diseases and conditions injure healthy liver cells, causing cell death and inflammation. This is followed by cell repair and finally tissue scarring as a result of the repair process. The scar tissue blocks the flow of blood through the liver and slows the liver's ability to process nutrients, hormones, drugs and natural toxins (poisons). It also reduces the production of proteins and other substances made by the liver. Cirrhosis eventually keeps the liver from working properly. Late-stage cirrhosis is life-threatening. You are more likely to get cirrhosis of the liver if you abuse alcohol for many years, have viral hepatitis, have diabetes, are obese, inject drugs using shared needles and have a history of liver disease. The four stages of cirrhosis stages is shown in **Fig. 1** .

Scientists estimate that cirrhosis of the liver affects about one in 400 adults in the U.S. It affects about 1 in 200 adults age 45 to 54, the age group most commonly affected by cirrhosis. Cirrhosis causes about 26,000 deaths each year in the U.S. and is the seventh leading cause of death in the U.S. among adults 25 to 64 years of age.[2]

Minorities in the United States have higher fatalities related to ongoing liver ailments.[3] Sometimes, liver disease is challenging to diagnose in its early stages. We cannot discover the disease until the liver function is partially damaged. Early diagnosis can be life-saving. Currently, the examination of predicting liver illness has been broadly contemplated.[6]

Globally, liver disease has become an alarming and life-threatening issue. Machine learning algorithms can help in early diagnosis to reduce risk. Hence, this report aims to achieve a simple performance on this problem.

This report aims to determine the accuracy of several popular machine algorithms like logistic regression, K-nearest neighbors (K-NN), support vector machine (SVM), random forest, logistic regression with cross validation, and XGBoost to predict liver diseases by analyzing a data set which was collected from the Mayo Clinic trial.

The paper is organized as follows:Section 2 presents the methodology of the research, Section 3 demonstrates the outcomes of different machine learning algorithms, and finally, the paper concluded in Section 4.

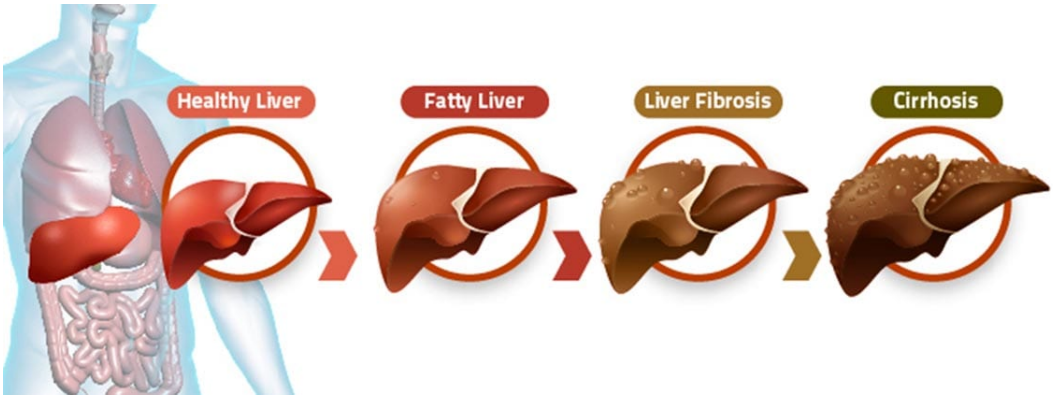

Figure 1: Cirrhosis stages

2 Methodology

The collected data set included 418 instances with 18 attributes and one outcome which is the histologic stage of disease. Instances and data set features, handling missing values, exploratory data analysis and visualizations, and application of the machine learning algorithms are explained in this section. The operational flow of this study is shown in **Fig. 2** .

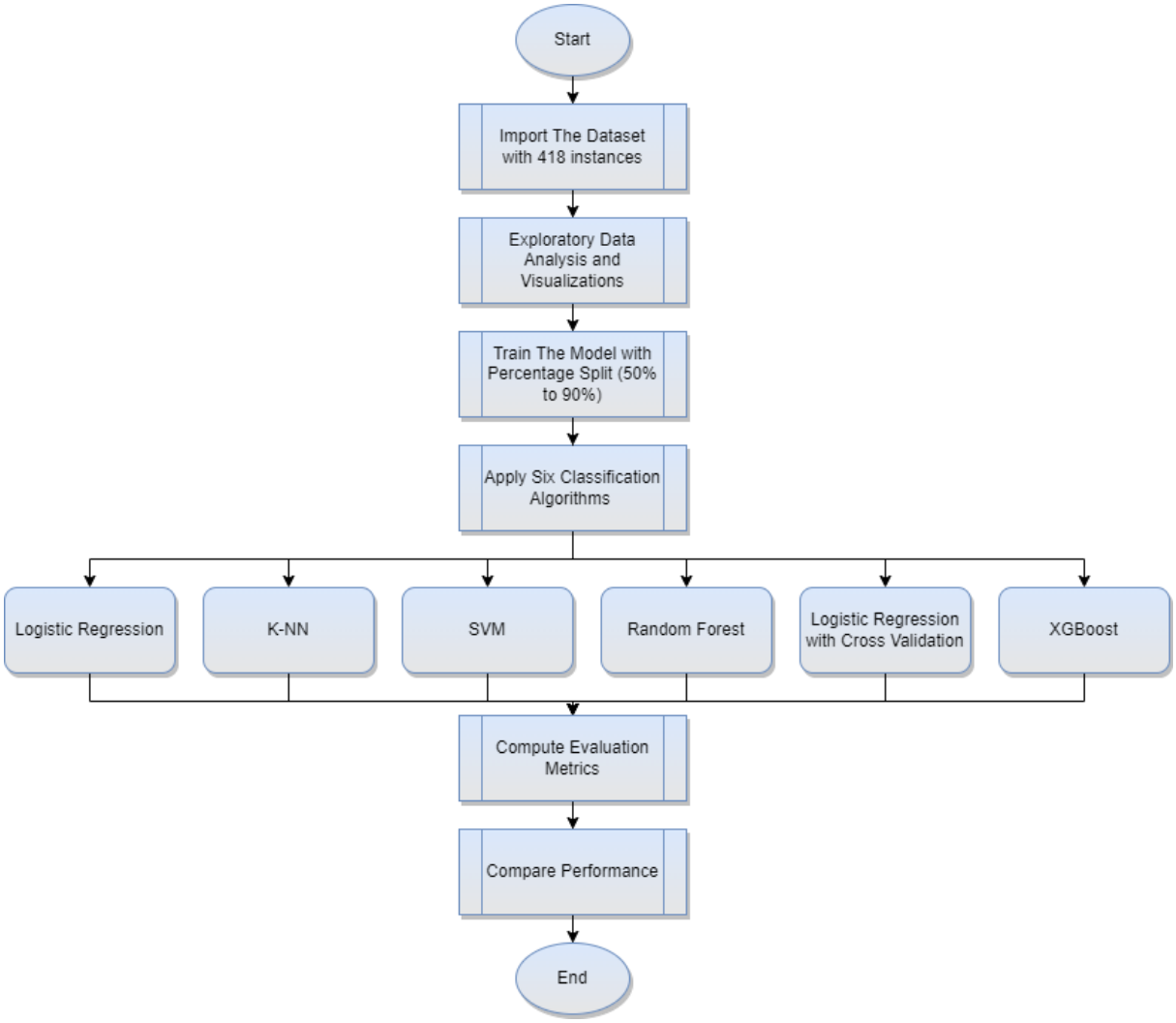

Figure 2: The operational flow chart

2.1 Instances and Data Set

The data contains the information collected from the Mayo Clinic trial in primary biliary cirrhosis (PBC) of the liver conducted between 1974 and 1984. The feature list is shown in **Tab. 1** .

A total of 424 PBC patients, referred to Mayo Clinic during that ten-year interval, met eligibility criteria for the randomized placebo-controlled trial of the drug D-penicillamine. The first 312 cases in the dataset participated in the randomized trial and contain largely complete data. The additional 112 cases did not participate in the clinical trial but consented to have basic measurements recorded and to be followed for survival. Six of those cases were lost to follow-up shortly after diagnosis, so the data here are on an additional 106 cases as well as the 312 randomized participants.

Approximately 37.44 percentage (144) of patients are in fourth stage of liver disease. Out of these data set 374 people are female, and 44 male.

Table 1: Feature List (Note: this feature list relates to the outcome data set of handling missing values)

Serial	**Features**	**Sub Category**	**Data Distributions**	**Data Type**
			Mean +- SD	
1	N_Days	Minimum: 41	Mean: 1917.782	Numeric
		Maximum: 4795	SD: 1104.673	
2	Status	D (death)	38.52%	Categoric
		C (censored)	55.50%	
		CL (censored due to liver tx)	5.98%	
3	Drug	D-penicillamine	63.15	Categoric
		Placebo	36.85	
4	Age	Minimum: 26	Mean: 50.38	Numeric
		Maximum: 78	SD: 10.48	
5	Sex	Female	89.47%	Categoric
		Male	10.53	
6	Ascites	No	94.25%	Categoric
		Yes	5.75%	
7	Hepatomegaly	No	36.36%	Categoric
		Yes	63.64%	

Table 1 (continued).

Serial	**Features Name**	**Sub Category**	**Data Distributions**	2[4]***Data Type**
			Mean +- SD	
8	Spidera	No	78.46%	Categoric
		Yes	21.54	
9	Edema	N	84.68%	Categoric
		Y	4.78%	
		S	40.54%	
10	Bilirubin	Minimum: 0.3	Mean: 3.09	Numeric
		Maximum: 16.44	SD: 3.91	
11	Cholesterol	Minimum: 120	Mean: 340.36	Numeric
		Maximum: 929.64	SD: 140.81	
12	Albumin	Minimum: 2.22	Mean: 3.49	Numeric
		Maximum: 4.64	SD: 0.42	
13	Copper	Minimum: 4	Mean: 88.88	Numeric
		Maximum: 314.73	SD: 63.38	
14	Alk_Phos	Minimum: 289	Mean: 1713.21	Numeric
		Maximum: 7424.51	SD: 1484.64	
15	SGOT	Minimum: 26.35	Mean: 119.63	Numeric
		Maximum: 267.81	SD: 44.97	
16	Tryglicerides	Minimum: 33	Mean: 117.56	Numeric
		Maximum: 281.42	SD: 45.11	
17	Platelets	Minimum: 62	Mean: 256.41	Numeric
		Maximum: 547.94	SD: 95.28	
18	Prothrombin	Minimum: 9	Mean: 10.7	Numeric
		Maximum: 13.78	SD: 0.91	
19	Stage	1	5.02%	Categoric
		2	22%	
		3	38.51%	
		4	34.47%	

2.2 Handling Missing Values and Detect Outliers

As there is some null values in our dataset, we could just get rid of all examples with NA values, but in this case our case of small dataset we cannot afford that. Hence, we will impute the missing entries with some statistical calculations.

We have two different types of data:

1.Numerical data (Age, Cholesterol, Platelets.. etc)

2.Categorical Data (Drug, Sex, Spiders..etc)

We will have to use different imputation for each type:

1.For the numerical type we can use mean or median. In this case we will go with median to avoid skewing in the presence of outliers. Also, the python code is shown in **Fig. 3** .

2.For Categorical type we will impute the most frequent class. Also, the python code is shown in **Fig. 4** .

```
In [19]: df_cat_col = df.select_dtypes(include=('object')).columns
         for c in df_cat_col:
             df[c].fillna(df[c].mode().values[0], inplace=True)
```

Figure 3: Code for handling categorical types of data

```
In [15]: df.select_dtypes(include=(['int64', 'float64'])).isna().sum()
         df_num_col = df.select_dtypes(include=(['int64', 'float64'])).columns
         for c in df_num_col:
             df[c].fillna(df[c].median(), inplace=True)
```

Figure 4: Code for handling numrical types of data

In terms of statistics, Outliers can be defined as,"An Outlier is that observation which is significantly different from all other observations." Hence, this outliers should be detect and replace with an appropriate data. For numerical distribution, the data points which fall below mean-3*(sigma) or above mean+3*(sigma) are outliers. Accordingly, Outliers can be checked by calculating the z-score in python. The distribution plots for the features is shown in **Fig. 5** . Moreover, an example of capping on Bilirubin's outlier code is shown in **Fig. 6**

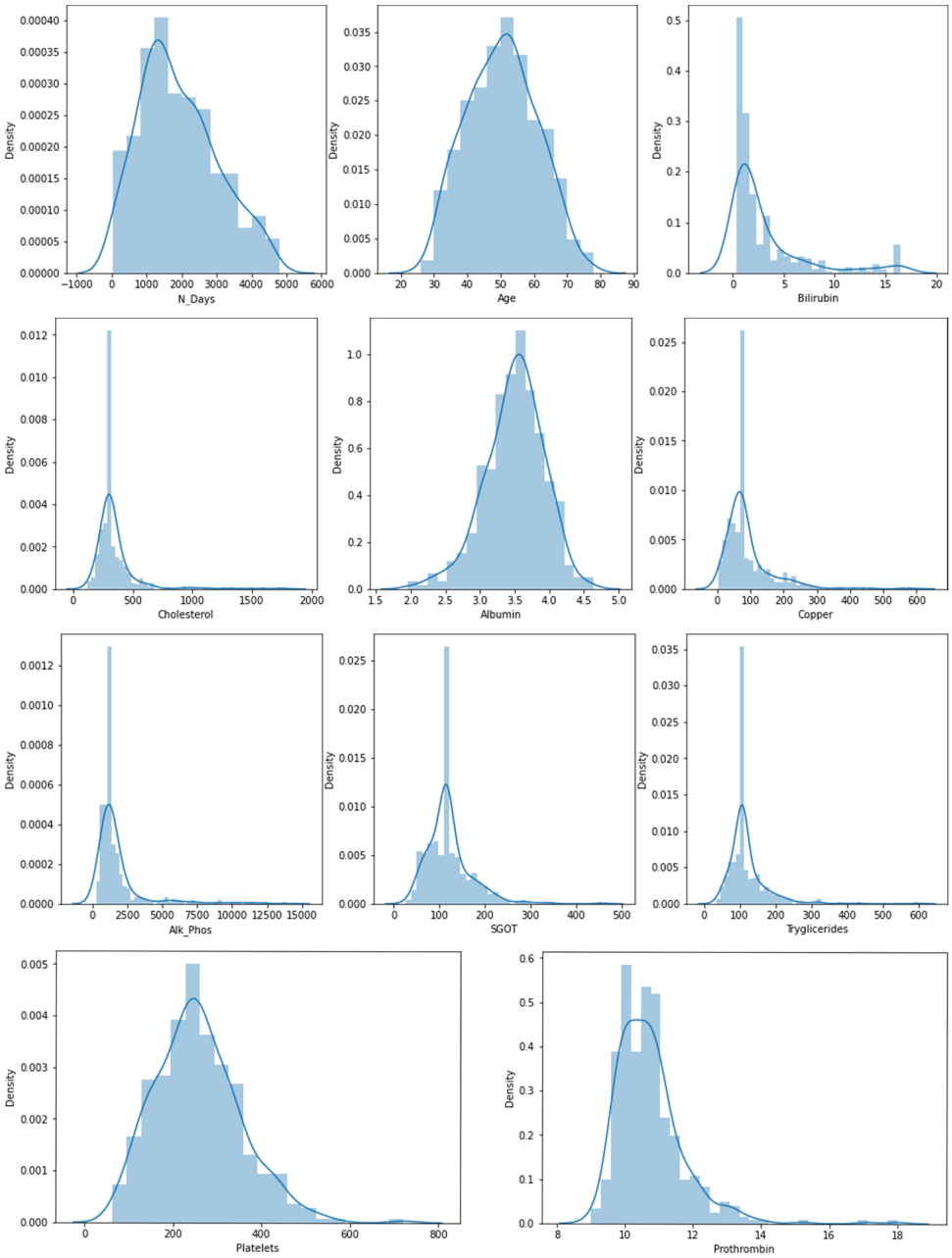

Figure 5: The Distribution Plots for The Features

```
In [48]: upper_limit = df['Bilirubin'].mean() + 3*df['Bilirubin'].std()
         lower_limit = df['Bilirubin'].mean() - 3*df['Bilirubin'].std()

In [49]: df['Bilirubin'] = np.where(
             df['Bilirubin']>upper_limit,
             upper_limit,
             np.where(
                 df['Bilirubin']<lower_limit,
                 lower_limit,
                 df['Bilirubin']
             )
         )

In [50]: df['Bilirubin'].describe()

Out[50]: count    418.000000
         mean       3.040437
         std        3.719671
         min        0.300000
         25%        0.800000
         50%        1.400000
         75%        3.400000
         max       14.841342
         Name: Bilirubin, dtype: float64
```

Figure 6: Capping on Outlier Code. For instance capping on Bilirubin

2.3 Exploratory Data Analysis and Visualizations

Now we should do some exploratory data analysis and visualizations to understand the data better. Hence, let's look at some instances.
Firstly, take a look at how many examples per class do we have in our dataset. It is clearly seen from **Fig. 7**, that the third stage of disease is the most common type among the data frame.

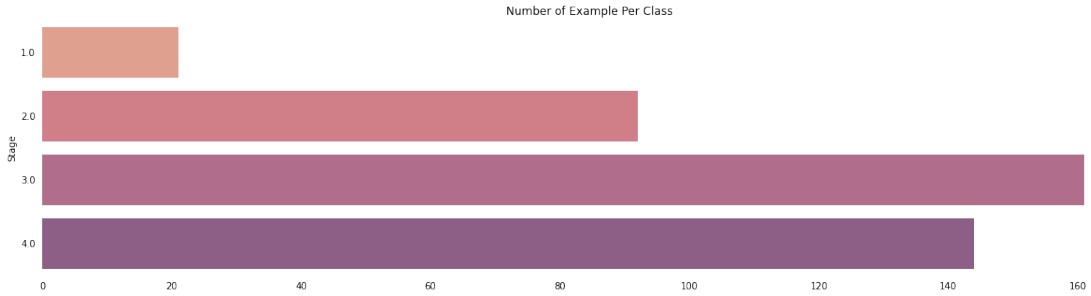

Figure 7: Number of Example Per Class

Also, the proportions of disease stages can be seen in the subcategories of categorical groups, which is shown in **Fig. 8**. Accordingly, the types of diagrams are drawn in the bar plot below. For instance, we find out the number of females is significantly higher than males, however, that most females are in stage two or three of their disease, while males are more in stage one or four.

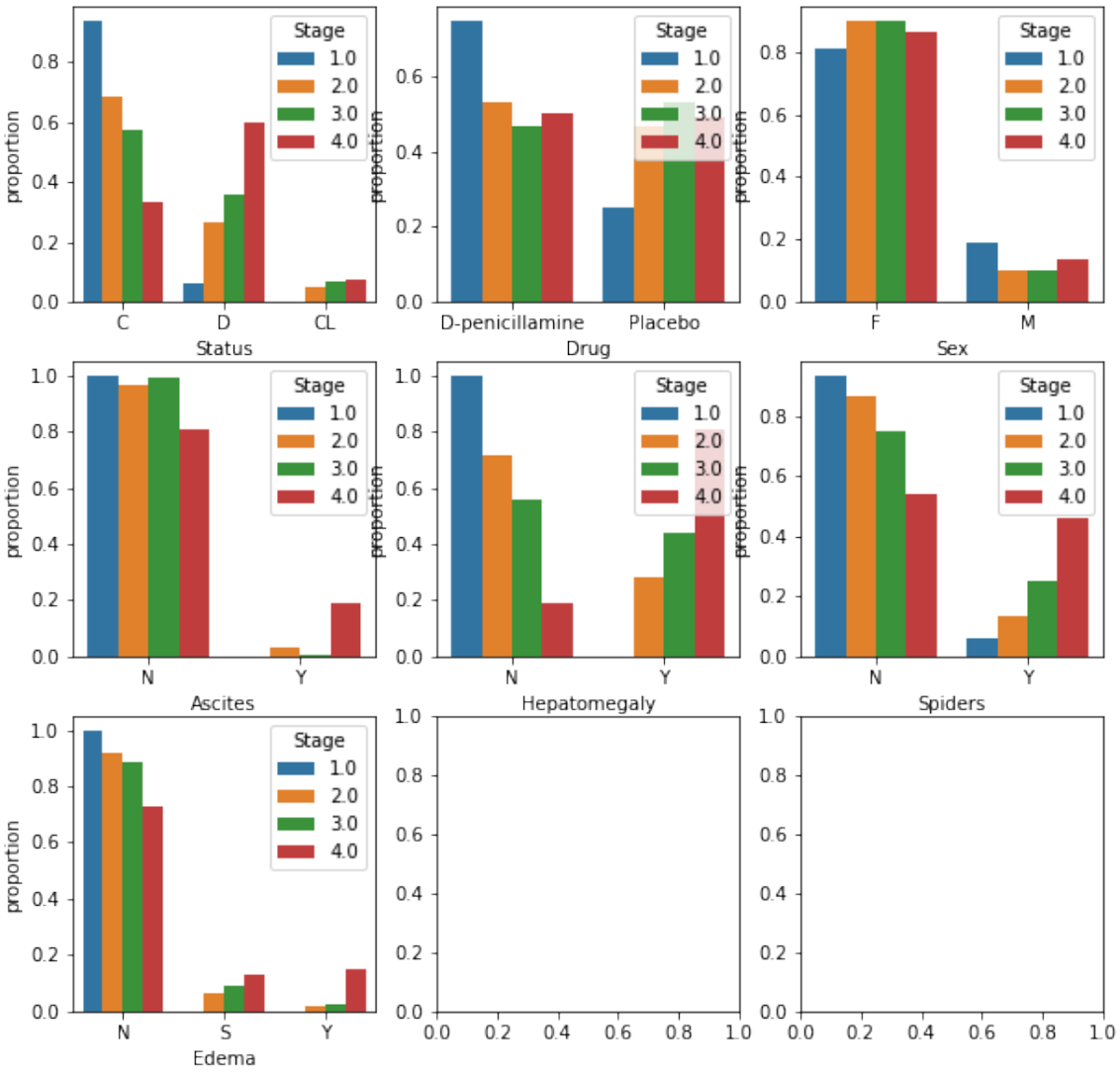

Figure 8: Proportions of Disease Stages

To further determine the path, it is necessary to draw a Correlations Between Variables chart. The relationships between the features can be seen in **Fig. 9** .

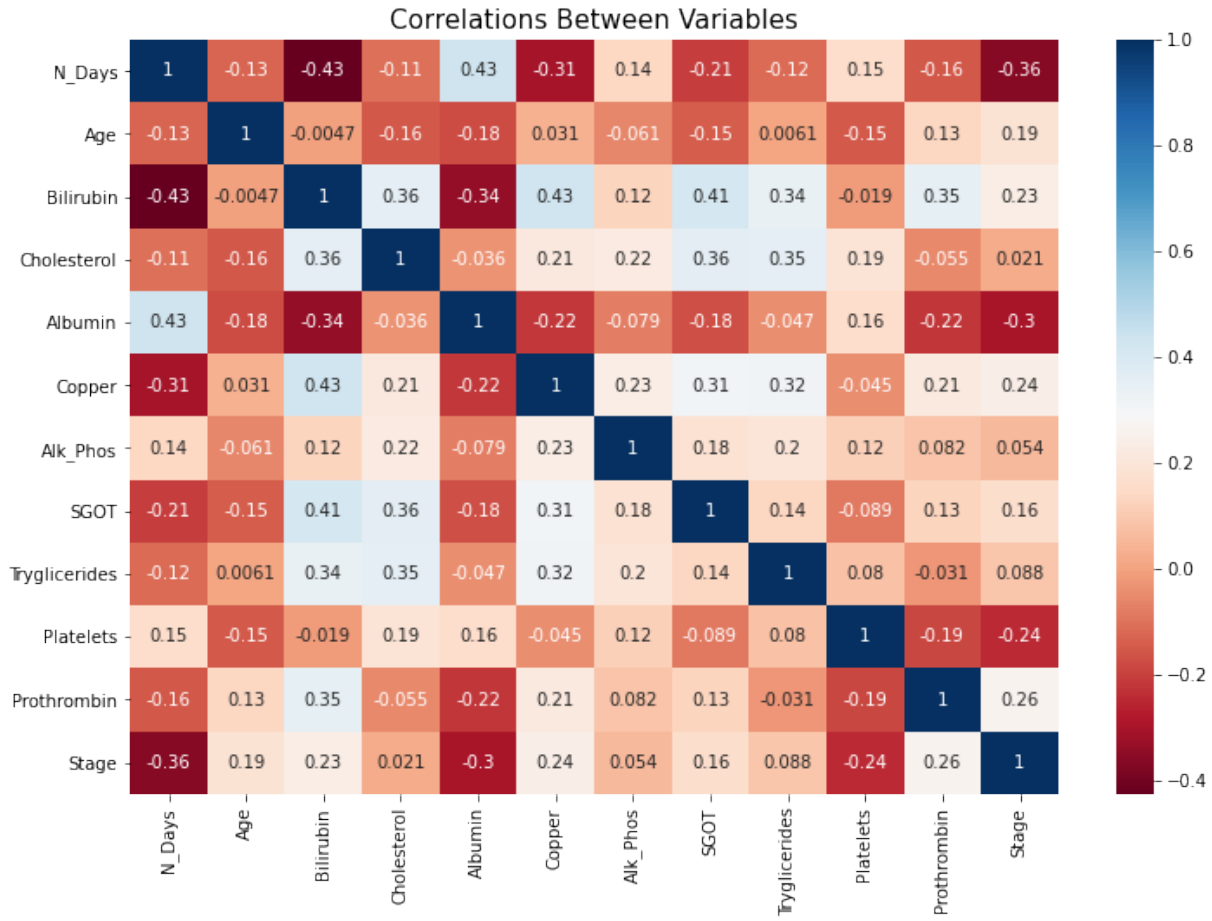

Figure 9: Correlations Between Variables

According to this chart, it is necessary to compare the disease stages with the six characteristics of Age, Prothrombin, Copper, Platelets, Albumin, and Cholesterol. Therefore, we will draw these comparisons using bar plots or box plots. In **Fig. 10** we notice that increasing the cases of Age, Prothrombin, and Copper increases the risk of disease. On the other hand, in **Fig. 11**, we find that increasing the number of Platelets and Albumin reduces the risk of disease. Also in **Fig. 12** Cholesterol levels up to stage 3 are harmful to the disease but are beneficial for stage four. This discrepancy in this diagram could be related to the lack of data.

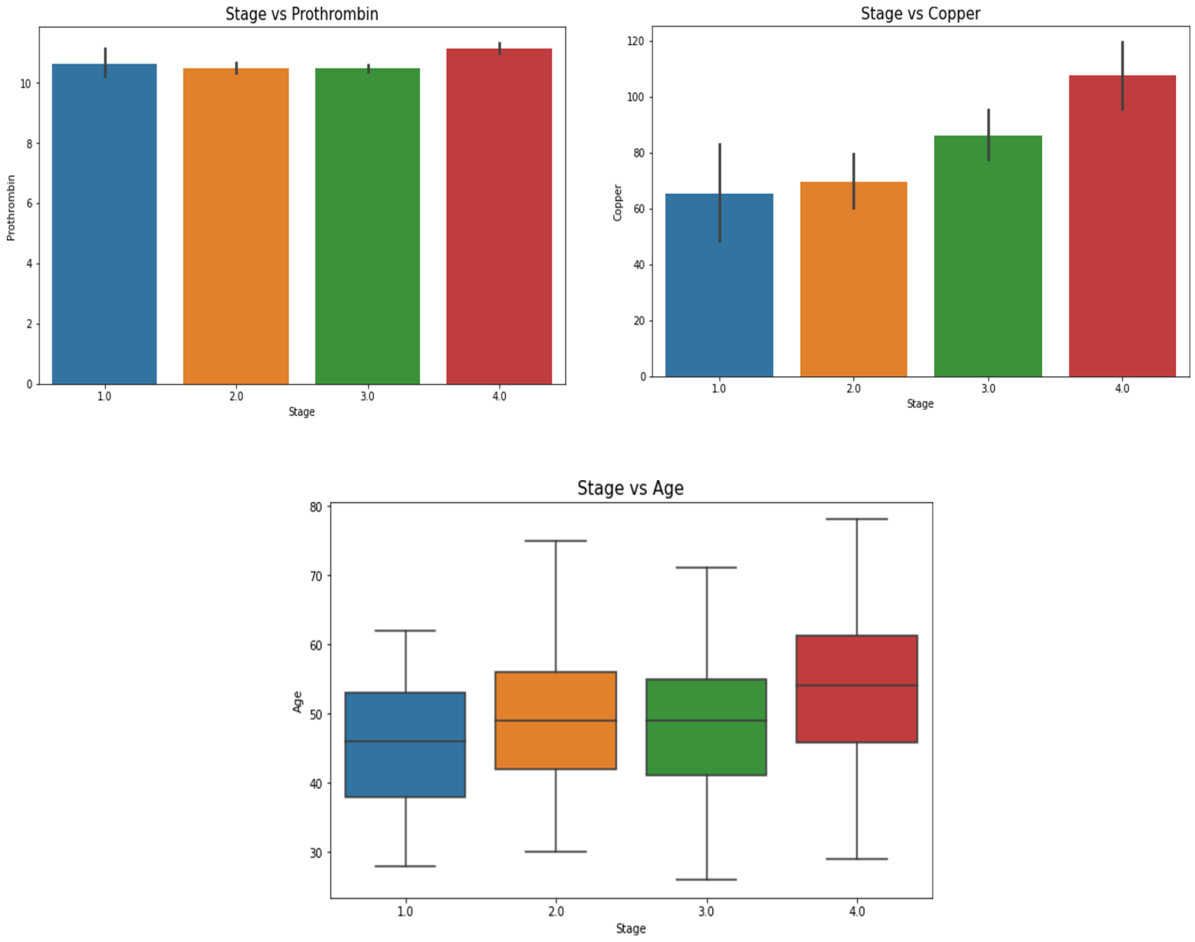

Figure 10: Stage vs Age, Prothrombin, Copper

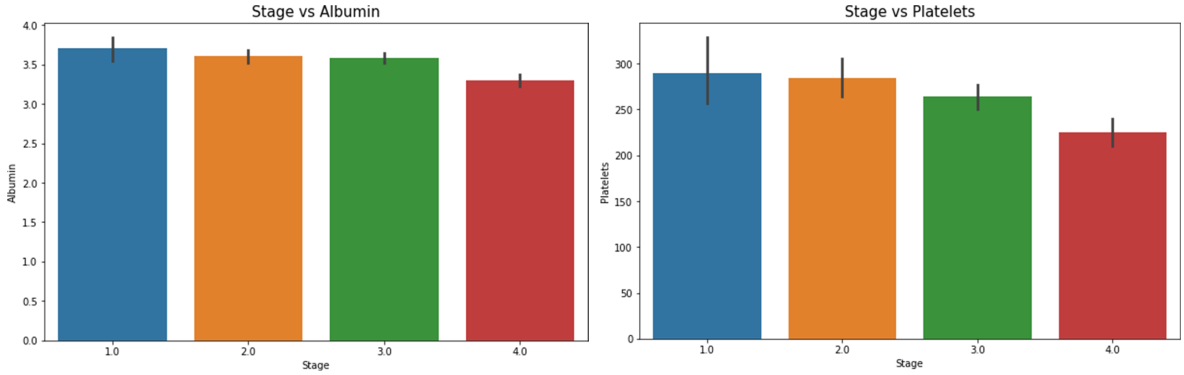

Figure 11: Stage vs Platelets and Albumin

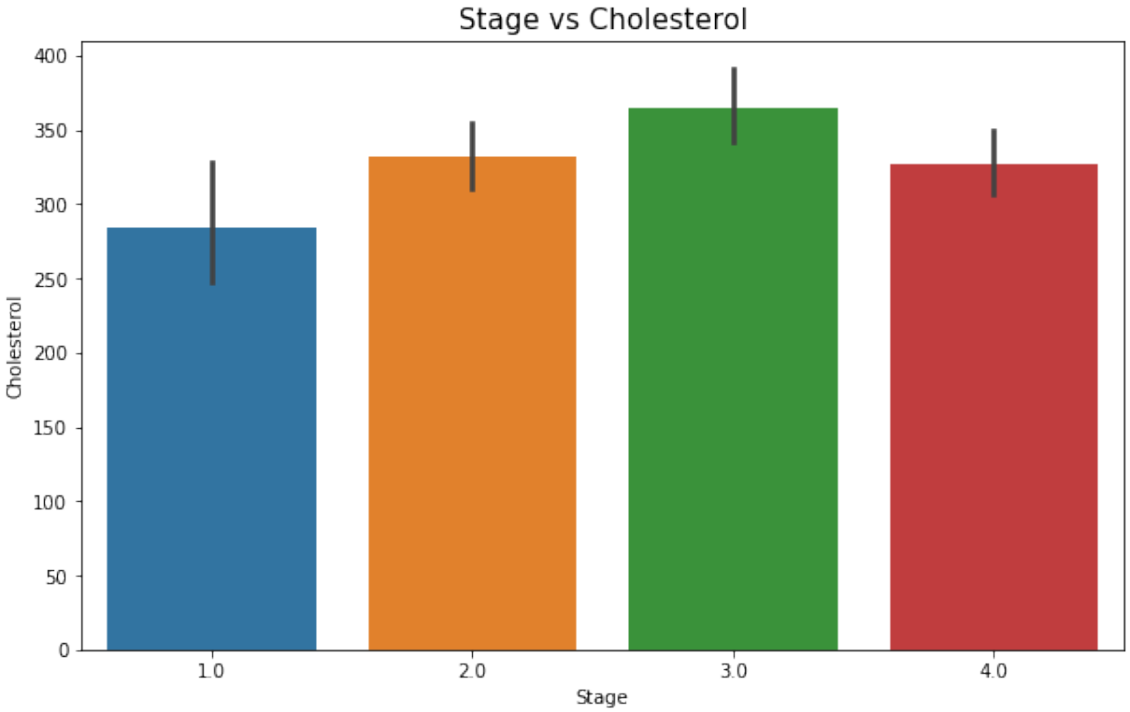

Figure 12: Stage vs Cholesterol

Finally, lets fit a regression plots to check our hypothesises about positive or negative correlated features.
As it can be seen, Regression Plots in **Fig. 13** demonstrate that Age, Prothrombin, and Copper are positively correlated with the possibility of Cirrhosis. In addition, Regression Plots in **Fig. 14** show Platelets and Albumin Negatively Correlated with the possibility of Cirrhosis. Also, the remarkable point about **Fig. 15** is that Cirrhosis is likely to have a completely upward trend with the rising of cholesterol. The reason for this discrepancy with the previous diagram related to cholesterol (**Fig. 12**) is that **Fig. 15** has continuous modes but **Fig. 12** has a discrete mode, which these problems are noticeable in any case included the research data.

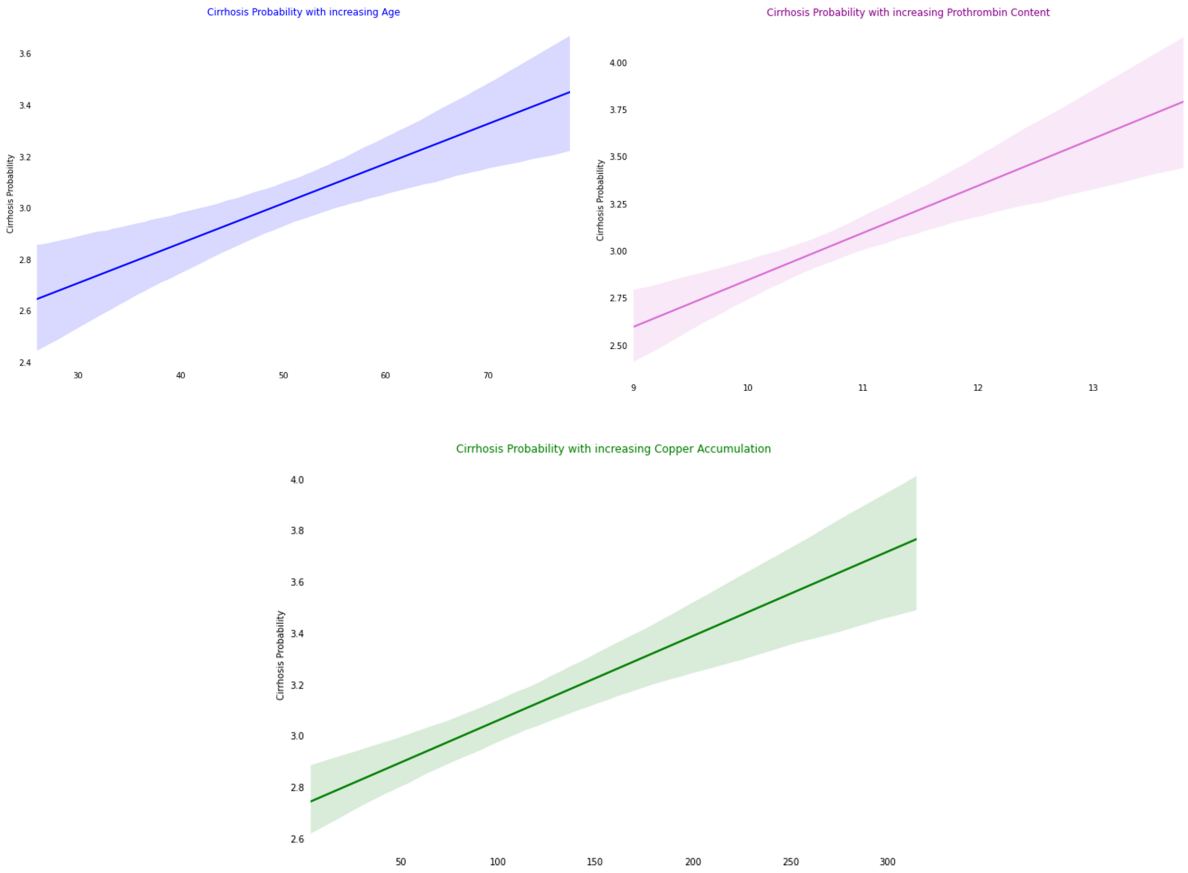

Figure 13: Cirrhosis Probability with increasing Age, Prothrombin, and Copper

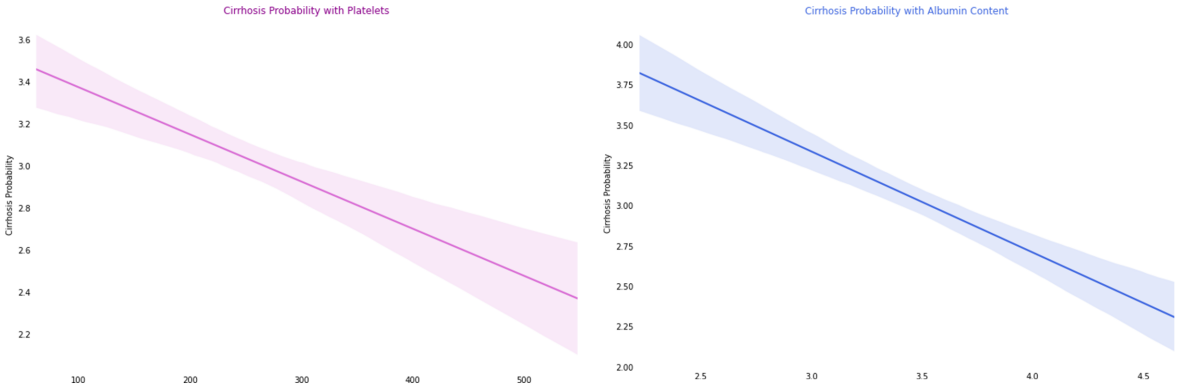

Figure 14: Cirrhosis Probability with increasing Platelets and Albumin

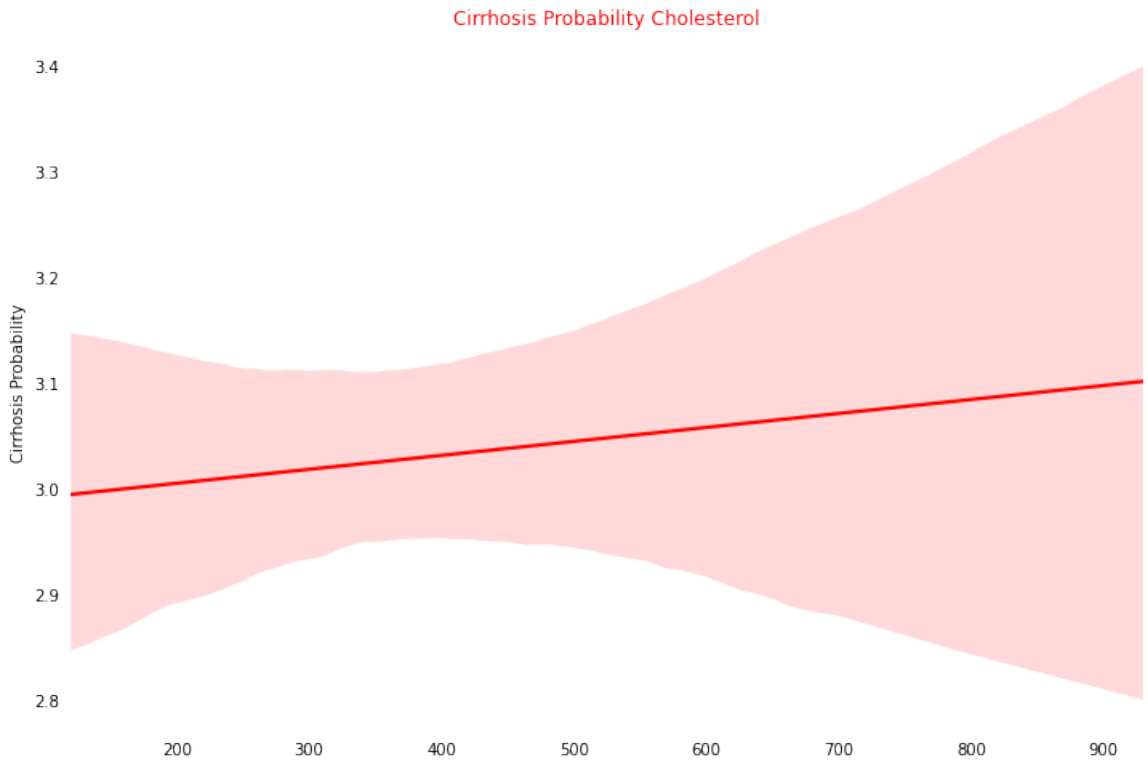

Figure 15: Cirrhosis Probability with increasing cholesterol

2.4 application of the machine learning algorithms

This study applied logistic regression, K-nearest neighbors (K-NN), support vector machine (SVM), random forest, logistic regression with cross validation, and XGBoost algorithms. The algorithms are briefly described as follows.It should also be noted that in the Preprocessing data stage, firstly, the categorical data should be replaced with integers (zero, one, and negative one). Then for specifying our model, steps one to three of disease are put in class zero and stage four of disease is put in class one. It can also be seen that our dataset classes are imbalanced. Therefore, it is necessary to use the Over Sampling method to make our models more accurate.

2.4.1 Logistic Regression (LR)

Logistic regression (LR) is generally a linear model that is used for predicting binary variables. LR technique is used for classifying a new observation of an unknown group. LR is a unique system for gathering data into two random and exhaustive data collections. [4]

2.4.2 K-Nearest Neighbors (K-NN)

K-NN is an elementary classification algorithm of machine learning. It gives an example of the most preferred class among the neighboring K. K is a constraint for changing the classification algorithms. [8]

2.4.3 Support Vector Machine (SVM)

SVM is a supervised measure of learning that can be used for classification and relapse problems, but it is commonly used for characterization problems. SVM functions work admirably and can grasp linear and nonlinear problems. [7]

2.4.4 Random Forest (RF)

Random forest (RF) is a versatile, convenient model that exhibits different outputs. RF obstructs the overfitting problem. It is one of the main versions of ensemble learning. Ensemble learning is defined by using the same algorithm multiple times or using numerous algorithms. [1]

2.4.5 logistic regression with cross validation

Cross-validation is largely used in settings where the target is prediction and it is necessary to estimate the accuracy of the performance of a predictive model. The prime reason for the use of cross-validation rather than conventional validation is that there is not enough data

available for partitioning them into separate training and test sets. This results in a loss of testing and modeling capability.

2.4.6 Extreme Gradient Boosting (XGBoost)

XGBoost is a supervised learning algorithm that implements a method to generate accurate models called boosting. Supervised learning applies from a series of notable training examples to the task of inferencing a predictive model.[5]

3 Outcome

3.1 Accuracy

Accuracy is the measurement used for classification evaluation. Accuracy is usually the portion of forecasting authentic prediction.

3.2 Precision

Precision is the dimension of positive predictions that define good predictions.

3.3 Recall

The recall is the component of the accumulated number of actual examples that have already been retrieved.

3.4 F1 Score

The F1 score is an acceptable measure to take advantage of the event. An analogy between precision and recall is found for observing the F1 score, and there remains a jagged class propagation.

3.5 AUC-ROC

The greater the ROC of the AUC, the stronger the portrayal of the model for separating the positive from the negative classes.

3.6 Experimented Analysis

Fig. 16 shows the results and accuracies of the implemented machine learning algorithms. Fig. 16 represents the accuracy of different trained data sets. XGboost shows the best performance, with an accuracy of 83.33% while the 80% of our data was in train dataset.Fig. 17 shows the mutation of trained data with accuracy. We trained different data with different size ranges of 50% to 90%. Fig. 17 shows that, at the 80% range of data training, the peak value is obtained for XGboost.

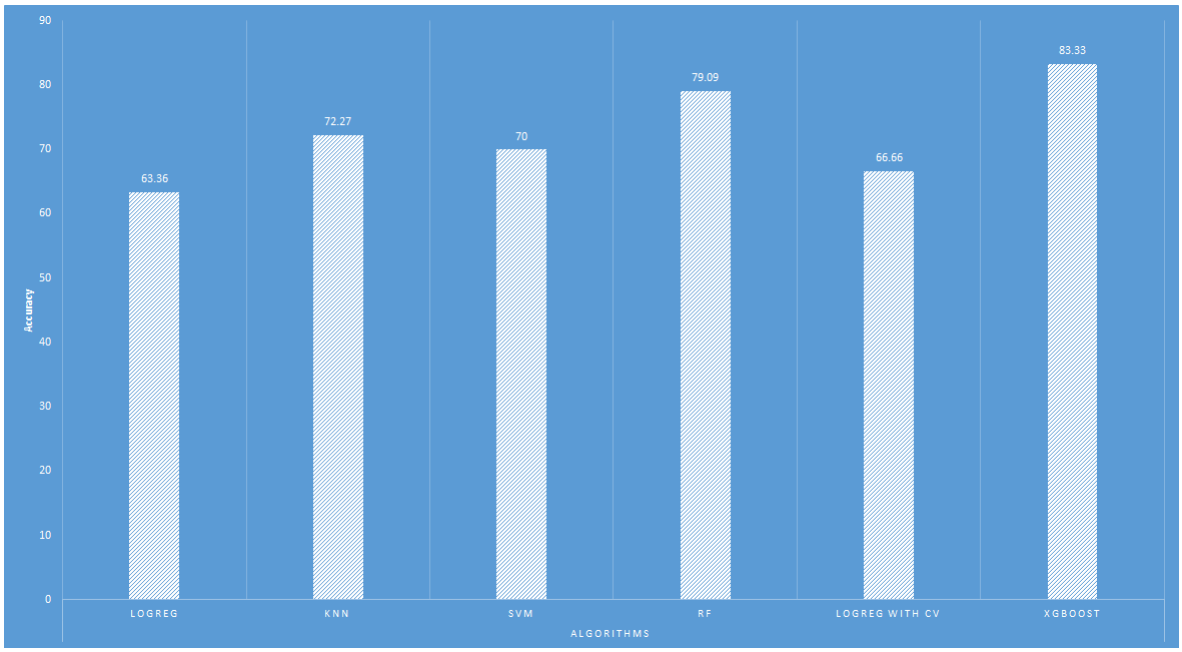

Figure 16: accuracy of applied algorithms

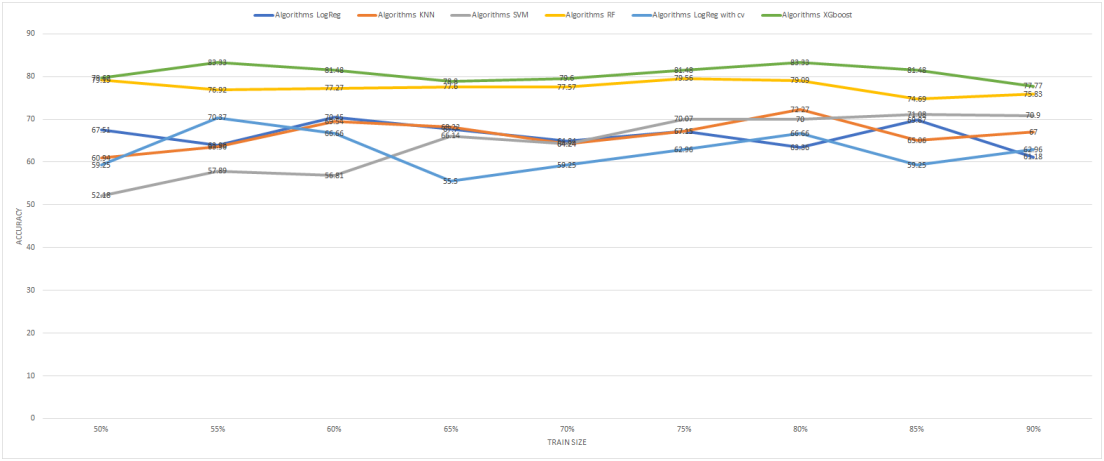

Figure 17: Train size (%) vs. accuracy (%)

From **Tab. 2**, we obtain the accuracy, precision, recall, F1 score, and AUC for those proposed algorithms. The accuracy, F1 score, precision, recall, and AUC were 63.63%, 63.62%, 63.65%, 63.63%, and 63.63% respectively, in the LogReg algorithm. K-NN showed an accuracy, F1 score, precision, recall, and AUC of 72.27%, 72.39%, 73.86%, 72.72%, and 72.72% respectively. SVM had an accuracy, F1 score, precision, recall, and AUC of 70%, 69.97%, 70.05%, 70%, and 70%, respectively. The RF exhibits an accuracy, F1 score, precision, recall, and AUC of 79.09%, 78.94%, 79.89%, 79.09%, 79.09%, respectively. LogReg with CV shows an accuracy, F1 score, precision, recall, and AUC of 66.66%, 66.39%, 67.8%, 66.66%, 71.97%, respectively. The accuracy, F1 score, precision, recall, and AUC were 83.33%, 83.32%, 83.37%, 83.33%, and 90.12& respectively, for the XGBoost algorithm.

Table 2: Statistical results (80% train size) of this study

Algorithms	Accuracy	F1_score	Precision	Recall	AUC
LogReg	63.63%	63.62%	63.65%	63.63%	63.63%
KNN	72.27%	72.39%	73.86%	72.72%	72.72%
SVM	70%	69.97%	70.05%	70%	70%
RF	**79.09%**	**78.94%**	**79.89%**	**79.09%**	**79.09%**
LogReg with CV	66.66%	66.39%	67.80%	66.66%	71.97%
XGboost	**83.33%**	**83.32%**	**83.37%**	**83.33%**	**90.12%**

Comparing the statistical results, it can be seen that XGboost has the best criteria among the algorithms. However, the Random Forest algorithm has significant outputs. Also, the use of the Over Sampling method has caused the F1 score, precision, recall, and AUC criteria to give very close results to each other.

4 Conclusion

This report demonstrated multiple prediction algorithms to predict and diagnose liver disease at an early stage. The data set showed different input parameters gathered, and we verified and trained the models for the input parameters given. The prediction of liver disease was tested with greater precision by evaluating the algorithms with an attribute collection and data set training. These findings identify novel factors to be used specifically at an early stage by classifiers to detect liver disease. LogReg, K-NN, SVM, RF, LogReg with CV, and XGBoost algorithms are constructed to predict liver disease. These findings showed that the XGBoost model accurately predicted patients with liver disease. Hence, XGBoost is considered the best and most promising algorithm for liver disease prognosis.

References

[1] Sneha Grampurohit and Chetan Sagarnal. Disease prediction using machine learning algorithms. In *2020 International Conference for Emerging Technology (INCET)*, pages 1–7. IEEE, 2020.

[2] Anna Khesin. *Living Well with Hemochromatosis: A Healthy Diet for Reducing Iron Intake, Managing Symptoms, and Feeling Great*. Simon and Schuster, 2019.

[3] Donghee Kim, Andrew A Li, Chiranjeevi Gadiparthi, Muhammad Ali Khan, George Cholankeril, Jeffrey S Glenn, and Aijaz Ahmed. Changing trends in etiology-based annual mortality from chronic liver disease, from 2007 through 2016. *Gastroenterology*, 155(4):1154–1163, 2018.

[4] Xingan Li. Application of data mining methods in the study of crime based on international data sources. 2014.

[5] Rory Mitchell and Eibe Frank. Accelerating the xgboost algorithm using gpu computing. *PeerJ Computer Science*, 3:e127, 2017.

[6] Sumedh Sontakke, Jay Lohokare, and Reshul Dani. Diagnosis of liver diseases using machine learning. In *2017 International Conference on Emerging Trends & Innovation in ICT (ICEI)*, pages 129–133. IEEE, 2017.

[7] Chih-Chia Yao and Pao-Ta Yu. Effective training of support vector machines using extractive support vector algorithm. In *2007 International Conference on Machine Learning and Cybernetics*, volume 3, pages 1808–1814. IEEE, 2007.

[8] Min-Ling Zhang and Zhi-Hua Zhou. Ml-knn: A lazy learning approach to multi-label learning. *Pattern recognition*, 40(7):2038–2048, 2007.