

University of Bahrain
College of Science
Department of Postgraduate Programs



Liver Disease Detection using Machine Learning Algorithms

Thesis submitted in partial fulfilment of the requirements for the
degree of M.Sc. In Big Data Science and Analytics

Prepared by:
Burair Hassan Al Telaq
ID Number: 202012510

Supervised by:

Dr. Mustafa Hammad	Dr. Abdulla Alqaddoumi
Associate Professor	Assistant Professor
University of Bahrain	University of Bahrain

Kingdom of Bahrain
June 2022



Thesis Discussion Committee Decision

Student Name: Burair Hassan Al Telaq	Student ID: 202012510
Department: Choose an item.	College: Science
Program: M.Sc. In Big Data Science and Analytics	
Thesis Title: Liver Disease Detection using Machine Learning Algorithms	
Supervisor Name: Dr. Mustafa Hammad and Dr. Abdulla Alqaddoumi	Student GPA: 3.06

The Discussion Committee met on 16-May-22 In College of Science. Upon discussing the student whose details are stated above in public, the discussion committee recommends the following:

The student is awarded a grade A in the thesis subject.

- ☐ Without making any changes to the thesis.
- ☒ After making the modifications attached with this decision.

Discussion Committee Members Approval:

Name

Signature

1. Dr. Mustafa Hammad and Dr. Abdulla Alqaddoumi (Advisor)
2. Dr. Wael Elmedany (Internal Examiner)
3. Dr. Mohamed Bsoul (External Examiner)

Abstract

Liver disease is considered a world health crisis with high mortality every year. This affects the patient's life in addition to the high-cost treatment in most countries. The health field could be helped by different science sectors to reduce the effects of some diseases. Therefore, Machine Learning (ML) could be used in the health care system on diagnosing diseases and help in decision making. That will lead to saving patients' lives, improving services, and reducing the cost. This research has been conducted to help the health care field by exploring and diagnosing liver disease through the potential of ML. The main goal of this study is detecting liver disease using ML algorithms. The dataset used in this research is Indian Liver Patient Dataset (ILPD) which was obtained from a public repository. Two types of ML algorithms implemented in this work are supervised and unsupervised learning. The supervised models are IBK, Random Forest, and Adaptive Boosting with their ensemble learning (voting and stacking). While the unsupervised models are K-Means, DBSCAN, Gaussian Mixture Model, and Expectation Maximization. Different evaluation criteria have been used to compare the models in supervised and unsupervised learning models. They were accuracy, recall, precision, ROC, and RMSE for classification models while two measurements for clustering models were Silhouette Coefficient and Adjusted Rand Index. Moreover, this work applied InfoGainAttributeEval and CorrelationAttributeEval techniques with Ranker search method for feature selection. Also, this research used five types of balancing techniques (ADASYN, SMOTETomek, Random Oversampling, and SMOTEENN). For supervised learning models, the results obtained in this study revealed that stacking technique outperforms other models in detecting liver disease for positive cases which scored 100% using the random oversampling technique. In addition, the IBK model obtained 99.6% in detecting negative cases using SMOTEENN technique. Furthermore, for unsupervised learning, Gaussian Mixture Model and Expectation Maximization have a good performance in detecting negative cases. In addition, this research found that balancing techniques have a positive effect on increasing the model performance. Also, the feature selection technique has a good impact on minimizing the medical tests. However, the results showed that voting technique has a limited impact on improving the accuracy. Different ML algorithms have been used in previous works. They were LR, NB, SVM, IBK, RF, ANN, DT, Boosting, Bagging, Stacking, Voting, K-Means, and DBSCAN. This study proposed stacking models with a random oversampling technique in detecting liver disease and that could help physicians and health care systems with a high rate of detection positive cases. That will lead to saving patients' live, reducing the cost, and improving the services in health care systems. Also, these techniques can increase the performance of weak classifiers (IBK, RF, and adaBoost). Moreover, the proposed models outperform previous works in the literature review in detecting positive cases of liver disease.

Dedication

This research is dedicated to my parents, Hassan Salman and Fatimah Abdullah. To my wife, Zahra Ibrahim. And lastly, to my children Mohammed, Hussain, and Zainab.

Acknowledgements

I would like to express my sincere thanks and gratitude to my thesis supervisors, Dr. Mustafa Hammad and Dr. Abdulla Alqaddoumi for all their guidance, support, and advice. I would like to thank my master's instructors, I learned a lot from their experience. I would also like to thank Dr. Sawsan Hilal for her continuous support and efforts. lastly, I also appreciate my parents and family for all their support and encouragement.

Table of Contents

Thesis Discussion Committee Decision	A
Abstract.....	B
Dedication.....	C
Acknowledgements.....	D
Table of Contents.....	E
List of Tables	H
List of Figures.....	I
List of Abbreviations	J
Chapter One Introduction	2
1.1 Overview	2
1.2 Research Background.....	2
1.3 Research Problem.....	4
1.4 Research Questions.....	5
1.5 Research Objectives	6
1.6 Research Significance	6
1.7 Research Setting	7
1.8 Summary.....	7
Chapter Two Literature Review	9
2.1 Overview.....	9
2.2 Feature Selection	9
2.3 Ensemble Learning	11
2.4 Clustering Models.....	13

2.5 Assessment Approaches	13
2.6 Summary	15
Chapter Three Methods and Procedures.....	17
3.1 Overview.....	17
3.2 Indian Liver Patient Dataset	17
3.3 Preprocessing Techniques	19
3.3.1 Variables Encoding.....	19
3.3.2 Imputation Missing Values.....	20
3.3.3 Normalization	20
3.3.4 Resampling Dataset	20
3.3.5 Feature Selection	23
3.4 Machine Learning Algorithms.....	24
3.4.1 Classification Algorithms	24
3.4.2 Clustering Algorithms	27
3.5 Evaluation Measures.....	29
3.6 Implementation Tools and Parameters Setting.....	31
3.7 Summary	33
Chapter Four Empirical Results	36
4.1 Overview.....	36
4.2 Classification Models Results	36
4.2.1 Applying Classification Models	36
4.2.2 Classification Models Analysis	44
4.3 Clustering Models Results.....	48
4.3.1 Applying Clustering Models	48

4.3.2 Clustering Models Analysis.....	49
4.4 Summary	53
Chapter Five Discussion, Conclusion and Recommendations	55
5.1 Overview.....	55
5.2 Discussing with Related Work	55
5.3 Answering Research Questions	56
5.4 Conclusion and Recommendations	59
5.5 Summary	61
References.....	62
Appendices.....	71
الملخص	76

List of Tables

Table (3.1): Dataset Statistics Information.....	18
Table (3.2): Description of ILPD Features	18
Table (3.3): Dataset Size After Applying Various Resampling Techniques.....	22
Table (3.4): Result by Applying Feature Selection	23
Table (3.5): Parameters Setting in this research on Weka.....	32
Table (4.1): Results After Applying SMOTE Technique	37
Table (4.2): Results After Applying ADASYN Technique	38
Table (4.3): Results After Applying SMOTETomek Technique	40
Table (4.4): Results After Applying Random Oversampling Technique.....	41
Table (4.5): Results After Applying SMOTEENN Technique	43
Table (4.6): Summary of Accuracy, ROC, and RMSE	44
Table (4.7): Summary of Recall and Precision	46
Table (4.8): Summary of Best Confusion Matrix.....	47
Table (4.9): Results of Clustering Models on Oversampling and Undersampling Techniques	48

List of Figures

Fig. 4.1: Comparing of Accuracy, ROC and RMSE	45
Fig. 4.2: Comparing Recall and Precision.....	46
Fig. 4.3: Plotting Clustering Result of K-Means Model	50
using SMOTE Technique	50
Fig. 4.4: Plotting Clustering Result of DBSCAN Model using SMOTE Technique.....	51
Fig. 4.5: Plotting Clustering Result of EM Model	52
using SMOTE Technique	52
Fig. 4.6: Plotting Clustering Result of EM Model	53
using SMOTEENN Technique.....	53
Fig. 5.1: Accuracy Before and After Feature Selection	57
Fig. 5.2: Accuracy using Different Balancing Techniques	58

List of Abbreviations

ML Machine Learning

LR Logistic Regression

DT Decision Tree

RF Random Forest

PSO Particle Swarm Optimization

adaBoost Adaptive Boosting

KNN K-Nearest Neighbour

ANN Artificial Neural Network

DBSCAN Density-based Spatial Clustering of Applications with Noise

SVM Support Vector Machine

MAE Mean Absolute Error

AUC Area Under the Curve

CHIRP Composite Hypercube on Iterated Random Projection

RAE Relative Absolute Error

TPR True Positive Rate

ROC Receiver Operating Characteristics

RMSE Root Mean Square Error

TP True Positive

TN True Negative

FP False Positive

FN False Negative

CLD Chronic Liver Disease

ILPD Indian Liver Patient Dataset

SMOTE Synthetic Minority Oversampling Technique

ADASYN Adaptive Synthetic Sampling Approach

SMOTETomek Synthetic Minority Oversampling Technique Tomek

SMOTEENN Synthetic Minority Oversampling Technique Edited
Nearest Neighbours

Acc. Accuracy

FS Feature Selection

Chapter One

Introduction

1.1 Overview

1.2 Research Background

1.3 Research Problem

1.4 Research Questions

1.5 Research Objectives

1.6 Research Significance

1.7 Research Setting

1.8 Summary

Chapter One

Introduction

1.1 Overview

This chapter introduces a background about the liver which is a part of the human body. Also, it presents the symptoms of liver disease, some causes, and types. Then, the chapter presents the research problem, research questions, research objectives, and the significance of using ML in diagnosing liver disease. That has a positive effect on patients, physicians, and the health care system.

1.2 Research Background

Liver is an organ in the human body located in the upper right side of the abdominal cavity and on top of the stomach that around 3 pounds weight with a cone shape. The liver performs more than 500 vital functions. Most chemical levels in the blood are organized by the liver. Also, ridding the blood of poisonous substances. Some symptoms of liver disease may include a yellowing of the skin and eyes, itching, easy bleeding, fatigue, weakness, and confusion. Furthermore, a variety of factors can cause liver disease such as overexposure of toxins on uncleaned fruits and vegetables or chemicals used around the house, many herbs and supplements, and obesity or high cholesterol. Also, liver disease can be inherited when a family member or close relative has liver disease. There are some types of liver diseases such as Hepatitis A, Hepatitis B, Hepatitis C, alcoholic liver disease, and nonalcoholic fatty liver disease (Johns Hopkins Medicine, 2019).

Liver disease is considered a global health burden. The mortality of liver cirrhosis worldwide increases rapidly every year. Also, the high-cost treatment is inaccessible in most countries of the world (Mokdad et al., 2014). Another study shows that in 2016, about 1.2 million deaths were caused by liver cirrhosis which is close to the number of deaths caused by other diseases which were 1.5 million, 1.3 million for diabetes and tuberculosis, respectively. In addition to deaths impact, chronic liver disease is related to the high cost of health care that affects middle and low-income countries. This study suggests more efforts in prevention, examination, and treatment (Balakrishnan, 2020). Moreover, in 2017 the number of deaths of chronic liver disease have increased by 120,000 cases with a total of 1.32 million compared to 2016 (Sepanlou et al., 2020). The main causes of death by liver disease are Hepatitis B and C. In addition, 257 million people are living with HBV and 71 million living with HCV (World Health Organization, 2017). Globally, according to Moon et al. (2019), the number of people who have CLD is an estimated 1.5 billion.

Many studies suggest increasing the awareness in the public health agencies to allocate more resources to face the spread of liver disease. Also, raising awareness of the economic burden of liver diseases in low and middle-income countries. Therefore, the development of the health care system should consider and focus on the suitable health interventions (Ndugga et al., 2017), (Chung, 2015), and (Cainelli, 2012).

Diagnosing liver disease by a physician requires several processes. Laboratory tests are the main process to determine if the patient has liver disease or not. Those tests may cost the medical sector and considered one of the most expenditures. Reducing the number of medical tests is

significant without effect on patient care or diagnostic accuracy (Attali et al., 2006).

Nowadays, many fields have used machine learning to increase their performance and productivity. One of the common fields that need to benefit from the high potential of ML is the health care system. ML could help in diagnosing disease in the early stage, decreasing the cost, and help decision making and other aspects of the health care system. By using ML, the patients' health care and the physicians' productivity will be improved. In addition, the decision making of the health care system will be developed to allocate the resources in the best way.

In previous works, many ML models have been applied on liver disease detection. The classification models were DT, RF, ANN, SVM, IBK, LR, NB, Boosting, Bagging, Stacking, and Voting. Also, the clustering models were K-Means, DBSCAN, and Affinity Propagation.

1.3 Research Problem

Diagnosing liver disease is a challenging task that requires multiple processes during the physician interview with patients. One of the common and crucial processes in diagnosing liver diseases is medical tests. In addition, collecting relevant data, symptoms, and signs will lead the physician to detect the disease. Those processes have different importance from the perspective of health care system, physician, and patient. The health care system needs to save the cost and leverage the resources in the best way such as reducing medical tests as much as possible. Also, the doctor wants to offer proper diagnosis to save patients' lives with suitable treatment. Furthermore, the patient needs to know if he/she has a specific disease without having to spend days to know the results.

During covid-19 pandemic, there has been a high demand on the health care system around the world. This has shifted the focus to covid-19 patients rather than others. This situation will delay the result of diagnosing disease by doctors in some clinics.

There are lessons learned, such as the importance to speed up detecting liver disease through the technology revolutions. Further, discovering the most important related medical tests that detect liver disease. Also, save the patient's life by notifying the doctor about the disease by just entering the medical tests into the system. That will reduce the time consuming for patient and doctor. Generally, it will help the health care system in decision making and improve the medical services.

1.4 Research Questions

ML has been employed in various fields such as health care system. Decision makers and doctors could benefit of using ML. Doctors can use ML in predicting and detecting disease. This study aims to detect liver disease using ML algorithms. The followings are the research questions:

RQ1. What is the ML potential in detecting liver disease?

RQ2. Can feature selection algorithms improve the detection of liver disease?

RQ3. Can balancing techniques improve the accuracy?

RQ4. How does ensemble learning perform in detecting liver disease?

RQ5. How do clustering algorithms perform in detecting liver disease?

1.5 Research Objectives

The main goal of this research is to select ML model that detects liver disease. The model would help physicians by minimizing the time consuming to detect liver disease. Also, that would save the patients' lives. In addition, saving the cost by reducing the number of medical tests. That could impact the development of the health care system by decision makers. The objectives of this research are the following:

- O1.** Investigate the current state-of-art related ML.
- O2.** Study ML algorithms and feature selection techniques.
- O3.** Determine the most appropriate feature selection techniques.
- O4.** Study the used dataset attributes and extract the subset of the most attributes, which can improve the detecting of liver disease.
- O5.** Study the effect on accuracy by using different balancing techniques on the dataset.
- O6.** Test the performance of the proposed classification model.
- O7.** Test the performance of the proposed clustering model.

1.6 Research Significance

Detecting liver disease by the physician requires multiple processes. Using technology to improve the time consuming in detecting liver disease will have a positive effect on patients, physicians, and the health care system. The patient will be informed about the disease without consuming a long time to start treating if he/she has a disease which could save the patient's life. Also, the doctor's time to read and analyze the result of medical tests will be reduced. In addition, the number of unnecessary medical tests will be reduced and that would reduce costs in

the long term. Therefore, the decision maker could improve the quality of services in the medical sector.

In this study, we propose different ML models. Then, the model with high performance in detecting liver disease will be selected.

1.7 Research Setting

The aim of this study is to propose a model and measure the performance of the model. Also, this study is quantitative research with an experimental research design and a scientific approach. The thesis will be organized as follows: Chapter 2 reviews literature related in detecting liver disease using machine learning. Then, Chapter 3, presents research methods and procedures. Also, the chapter describes the dataset used, ML algorithms, and description of the used evaluation measures. Chapter 4 evaluates the proposed models by showing the results of the experiments. Finally, chapter 5 answers the research question, concludes the conducted work, and provides recommendations for future work.

1.8 Summary

The chapter presented liver disease with estimations of global mortality every year along with the number of infected people in some types of liver disease. The impact of using ML in health care system. Furthermore, this chapter showed the importance of reducing the time consuming to detect liver disease that saves patients' life, helps physicians and the services offered by the health care system. Also, the chapter stated the research questions and objectives.

Chapter Two

Literature Review

2.1 Overview

2.2 Feature Selection

2.3 Ensemble Learning

2.4 Clustering Models

2.5 Assessment Approaches

2.6 Summary

Chapter Two

Literature Review

2.1 Overview

This chapter presents an overview of the literature related to using ML in detecting liver disease. The chapter highlights the utilization of ML in detecting liver disease. In addition, the use of supervised and unsupervised ML algorithms is related to the study. Moreover, review the impact of using the ensemble model and feature selection on detecting liver disease.

2.2 Feature Selection

Feature selection is one of the important techniques which improve the performance of machine learning algorithms. Different authors have been used various functions to select the best attributes on Indian Liver Patient Dataset (ILPD). Singh et al. (2020) applied different algorithms with feature selection and without. The authors achieved the best accuracy 74.36% on the Logistic Regression (LR) model by using feature selection with an improvement in accuracy by 2%. Another study conducted by Kuzhippallil et al. (2020) used two techniques to improve the performance of the models using a genetic algorithm for feature selection and an isolation forest for eliminating outliers. The maximum accuracy of 86% was obtained by the LightGBM algorithm with increasing rate of 16%. Also, Banu Priya et al. (2018) introduced a paper applying Particle Swarm Optimization (PSO) feature selection on

different algorithms. The paper concludes that J48 Decision Tree (DT) algorithm using PSO achieved a better result with a score of 95.04% compared to other classification algorithms. Meanwhile, Rabbi et al. (2020) implement different algorithms with Adaptive Boosting (adaBoost) for each classifier with Pearson Correlation Coefficient based Feature Selection (PCC-FS). There was a slight increase in the highest accuracy 92.19% which was achieved by adaBoost classifier with Extra Tree as a base estimator. Moreover, Muhuri et al. (2019) applied a model based on Brownian motion model to predict liver disease. The authors studied the attributes variation with respect to the age progresses. Also, the proposed model was used to extract the patterns of the features. The Spearman's correlation coefficient was applied to examine the attribute correlation. The proposed model achieved 94.09% in accuracy and 93.58% in terms of sensitivity.

Some studies in the literature have applied RF model on ILPD by removing the same attributes using different resampling techniques. Sreejith et al. (2020) conducted a study to develop a clinical decision support system using ILPD and two different clinical datasets. The study has applied Synthetic Minority Oversampling Technique (SMOTE) using Orchard's algorithm for balancing the dataset. Also, a wrapper technique was applied for feature subset selection using Chaotic Multi-Verse Optimization. On ILPD, six attributes were selected: DB, SGOT, AAP, Age, TB, and SGPT. In addition, the RF model achieved an accuracy rate of 82.46%. The study concluded that the proposed method has a statistical superiority in the results. However, in 2020, a paper presented by Ambesange et al. used RF to optimize liver disease prediction with different preprocessing techniques. Four attributes were removed: Albumin, Total_protiens, (Albumin and Globulin Ratio), and Gender to improve the performance. Also, it has been checked the performance after

each technique applied on ILPD. The authors concluded that undersampling with total samples of 114 or 116 instances is a better technique than oversampling. Which scored 100% in accuracy measurement. In addition, the paper's findings revealed that, for building effective models the quality and clean data are significant.

2.3 Ensemble Learning

The ensemble model is another technique used by researchers to improve the performance of the models in predicting liver disease. Ensemble model combines different models called base estimators in the prediction process. Raheem et al. (2017) used in their research various ML models: Naïve Bayes (NB), KNN, and C4.5 (A version of DT), also for each model applied Boosting and Bagging. Also, the three algorithms were combined using Stacking and Voting. The highest classification accuracy was obtained by C4.5 DT model with a score of 70.84%. Therefore, the paper concludes that using ensemble models in classifying liver disease did not improve the performance. In contrast, several papers achieved high performance using ensemble learning. Gupta et al. (2020) proposed and recommend a vote majority of three Machine Learning (ML) models: K-Nearest Neighbour (KNN), DT, and Artificial Neural Network (ANN); their accuracy is 84.24%, 88.42%, and 44.76%, respectively. The study revealed a vote majority of three models achieved the best result with an average accuracy of 93.69% on ILPD. Al Telaq and Hewahi presented a recent study in 2021 using PCA to predict liver disease by various algorithms with an ensemble technique. The models implemented were KNN, RF, and SVM on ILPD. The highest TPR was obtained by ensemble technique for negative cases with 99%. Which was

combined by KNN and RF algorithms. The researchers concluded that ensemble learning had a significant impact on increasing the performance. In addition, using PCA reduction technique achieved similar accuracy or had a positive effect on increasing the performance.

In addition, various studies have obtained good performance by applying ensemble learning. Kumar and Thakur conducted a study in 2019 using two models: fuzzy based Adaptive K-NN and Neighbor Weighted (Fuzzy-ANWKNN) and fuzzy based Adaptive K-NN (fuzzy-ADPTKNN). The fuzzy-ANWKNN model obtained better results than fuzzy-ADPTKNN on ILPD. In addition, the proposed model achieved 84.29%, 88.81%, and 91.36% in terms of accuracy, sensitivity, and precision, respectively. Another research presented by Li et al. (2021) implemented Adaptive Boosting with CART as a base learner on ILPD. The model achieved the highest performance on accuracy and precision with 83.06% and 84.31% compared to other models.

Also, ensemble learning has been applied for generating the rules. Hassoon et al. (2017) introduced a new approach to diagnosing liver disease on ILPD. They were using a Genetic Algorithm (GA) for reducing the number of rules produced by the Boosted C5.0 model. The rules were the input data in GA. The study results showed that the total number of rules decreased from 92 to 24 rules. Also, the gender attribute does not have any impact on diagnosing liver disease. Furthermore, the proposed model gained accuracy by 92.93% and sensitivity by 94.16% which is higher than Boosted C5.0.

2.4 Clustering Models

Clustering is an unsupervised ML. The algorithm automatically groups the training instances with similar features into categories. Vats et al. (2019) have been applied three unsupervised ML models on liver disease dataset. The models are K-Means, DBSCAN, and Affinity Propagation, authors compared the models depending on seven factors. The main factor selected to determine the accuracy is Silhouette Coefficient which has a value between 1 and -1. The second factor is the number of clusters produced by the algorithm to compute complexity. K-means is introduced as an optimal method comparing to other methods. Which obtained the highest value of 0.780 on Silhouette Coefficient, In addition to a small number of clusters. Other work presented by Kumar and Katyal (2018) implemented K-Means and other four ML algorithms on ILPD. The study recommends adaptive Boosting of C5.0 DT in diagnosing liver disorder. Moreover, Antony et al. (2021) conducted a study using four unsupervised ML on predicting chronic kidney disease. The models used in the study with features reduction methods are K-Means, DBSCAN, I-Forest, and Autoencoder obtained an accuracy rate of 99%, 97%, 87%, and 71%, respectively. The highest accuracy was scored by K-Means clustering model.

2.5 Assessment Approaches

Many researchers have used different measurements to evaluate the models for predicting or diagnosing liver disease. One of the common measurements is accuracy. ANN model was applied by Jacob et al. (2018) on ILPD and Haque et al. (2018) on BUPA liver dataset. Both papers got an accuracy of 92.8% and 85.29% respectively. Furthermore,

Ramesh and Katheria (2019) implemented Support Vector Machine (SVM) algorithm with an accuracy rate of 89.1%, the same model achieved lower accuracy of 75.04% by Geetha and Arunachalam (2021). Moreover, Simon et al. (2021) have utilized LR method and got a higher accuracy rate of 81% than 74% which was conducted by Adil et al. (2018). In addition, another research conducted by Ambesange et al. (2020) have been applied KNN algorithm. The study scored a 91% accuracy rate. Another study was conducted in 2017 by Abdar et al.. The study used boosted C5.0 and Chi_Square Automatic Interaction Detection (CHAID) algorithms for generating rules on ILPD. Furthermore, the results revealed the importance of gender factor using the rules created by boosted C5.0. which is considered irrelevant in many other research. The highest accuracy was obtained by Boosted C5.0 with 93.75%.

On the other hand, several authors assess ML models for classifying liver disease using other measurements such as Mean Absolute Error (MAE), Area Under the Curve (AUC), and True Positive Rate (TPR) or Sensitivity. Based on an experiment by Khan et al. (2019) proposed Composite Hypercube on Iterated Random Projection (CHIRP) model depends on MAE and Relative Absolute Error (RAE). This model was selected due to decreasing error rate, which obtained better results 0.2870 and 58.8765% in (MAE, RAE), respectively. Another measurement is AUC was utilized by Hartatik et al. (2020) to find the best between the two models. The result was found NB achieved a higher value than KNN in terms of AUC. In Gan et al. (2020), The main goal of the conducted study was improving classification accuracy for imbalanced datasets such as ILPD. The authors combined TAN tree Bayesian classification with the cost-sensitive boosting model (AdaCost) to overcome the misclassification cost problem. The proposed model scored a better outcome on cost-sensitive learning compared with other models. Also,

the researchers concluded that the introduced model is most suitable for clinical imbalanced datasets. The model achieved 68.26% in terms of accuracy and 69.53% in AUC.

Also, True Positive Rate (TPR) or sensitivity as a different measurement was used by Thirunavukkarasu et al. (2018) to compare LR and KNN after they were finding the same accuracy in both models. So, the authors used TPR to compare them. They found LR model had a higher sensitivity rate of 0.952 than KNN 0.904 for predicting liver disease. Nahar et al. (2019) have been used TPR along with other measurements. The study revealed that the LogitBoost model outperformed the other models in terms of TPR with a rate of 93%. The paper presented by Singh et al. (2021) applied different machine learning models on ILPD for detecting liver disease. The RF model obtained the highest performance on most of the evaluation measurements. It scored 88.55%, 91.89%, and 83.95% in terms of accuracy, sensitivity, and precision, respectively. Whereas the lowest performance was achieved by Support Vector Machine (SVM).

2.6 Summary

This chapter introduced some related work of detecting liver disease using ML. That helped our study to explore the latest conducted research on how ML can be applied in detecting liver disease. Also, we achieved our research objective **O1**.

Chapter Three

Methods and Procedures

3.1 Overview

3.2 Used Dataset

3.3 Preprocessing Techniques

3.3.1 Variables Encoding

3.3.2 Imputation Missing Values

3.3.3 Normalization

3.3.4 Resampling Dataset

3.3.5 Feature Selection

3.4 Machine Learning Algorithms

3.4.1 Classification Algorithms

3.4.2 Clustering Algorithms

3.5 Evaluation Measures

3.6 Implementation Tools and Parameters Setting

3.7 Summary

Chapter Three

Methods and Procedures

3.1 Overview

This chapter presents the dataset used in this research. Also, it explains the preprocessing techniques implemented in the dataset. Furthermore, this chapter shows various ML algorithms applied in this study. Moreover, the chapter explains the evaluation criteria used to compare the models. Then, it presents the platforms utilized with a description of parameters implemented in different models.

3.2 Indian Liver Patient Dataset

The dataset used in this research will be explained and analyzed as the main part before implementing ML models. The dataset applied is Indian Liver Patient Dataset (ILPD), which was obtained from University of California, Irvine (UCI) repository which is a public source UCI (Machine Learning Repository, n.d.). It was collected in India from northeast of Andhra Pradesh. Any patient whose age more than 89 is recorded as being of age 90. The class label is classified by the experts. Table 3.1 shows statistics information about the ILPD. The dataset has 583 instances, and 11 features include the class label, which separates the data into two groups (liver patients and non-liver patients). The number of patients with liver disease is 416 cases (71.36%). While the number of patients without liver disease is 167 (28.64%). In addition, the number of males is 441 cases (75.64%) and for females is 142 cases (24.36%). The details of features are illustrated in Table 3.2.

Table (3.1): Dataset Statistics Information

Name	Number
Number of instances	583
Number of features	11
Number of patients with liver disease (positive)	416
Number of patients without liver disease (negative)	167
Percentage of positive cases	71.36%
Percentage of negative cases	28.64%
Number of males	441
Number of females	142
Percentage of males	75.64%
Percentage of females	24.36%

Table (3.2): Description of ILPD Features

No.	Attributes	Description	Type	Range
1	Age	Age of the patient	Numeric	4 – 90
2	Gender	Gender of the patient, Male/Female	Nominal	Male/ Female
3	TB	Total Bilirubin	Numeric	0.4 – 75
4	DB	Direct Bilirubin	Numeric	0.1 – 19.7
5	Alkphos	Alkaline Phosphotase	Numeric	63 – 2110
6	Sgpt	Alamine Aminotransferase	Numeric	10 – 2000
7	Sgot	Aspartate Aminotransferase	Numeric	10 – 4929
8	TP	Total Proteins	Numeric	2.7 – 9.6
9	ALB	Albumin	Numeric	0.9 – 5.5
10	A/G Ratio	Albumin and Globulin Ratio	Numeric	0.3 – 2.8

11	Class	Class label for patient has liver disease 1 or without 2	Nominal	1 or 2
----	-------	--	---------	--------

As shown in Table 3.2 the nominal features in the dataset are Gender and Class while the rest are numeric. The names of numeric attributes are Age, TB, DB, Alkohos, Sgpt, Sgot, TP, ALB, and A/G Ratio. Also, the range column shows the minimum and maximum values of each numeric attribute.

3.3 Preprocessing Techniques

By exploring the dataset using different methods to deeply understand the nature of the data and if it needs preprocessing. The results have shown there were problems in the dataset, such as: a few missing values, imbalance issues, and outliers. This will cause some problems during model training such as noise of the data, overfitting, or generalization. For these reasons, the following steps were executed on ILPD for cleaning and making the dataset more reliable before applying ML algorithms.

3.3.1 Variables Encoding

The performance of ML models affected by the input types from the dataset. Also, only numerical attributes are accepted by most of the models. This means how is an important step to preprocess the data so that the model can train well to achieve better performance. The selected dataset in this research has two attributes that need to be converted from categorical to binary values. The first attribute is Gender attribute was replaced with 1 for males and 0 for females. The second one is Class attribute also converted to 1 for positive cases and 0 for negative cases.

3.3.2 Imputation Missing Values

One of the most problems in real data especially in the health field is missing values. ILPD has a few instances with missing values on the (A/G Ratio) attribute. The missing values were replaced by the average of the (A/G Ratio) attribute.

3.3.3 Normalization

One of the essential steps before fitting the model is transforming the values of the dataset to be on a similar scale. Normalization is the scaler technique used in this research. Which transformed the values into a range between 0 and 1. The formula used to normalize the values as follows:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.1)$$

where X' is the normalized value, X is the value in the feature, X_{min} is the minimum value and X_{max} is the maximum value in the feature.

3.3.4 Resampling Dataset

The ILPD is a binary classification problem. As shown in Table 3.1, the target class has 167 negative cases (28.64%) and 416 positive cases (71.36%). It is clear there is a high difference between the sizes of the two classes which indicates an imbalanced problem. This research has used different techniques to tackle this problem. The following oversampling and under-sampling techniques were applied to the dataset:

1. Random oversampling:

This method randomly duplicates instances from the minority class with replacements to increase the number of samples (Krishnani et al., 2019).

2. Synthetic Minority Oversampling Technique (SMOTE):

It is an oversampling method for the minority class. Instead of making copies from the minority class, this technique creates synthetic examples. This technique works by choosing a random instance then finding its k nearest neighbours. The new instances are created between selected random instance and one or more of its nearest neighbours (Sreejith et al., 2020).

3. Adaptive Synthetic Sampling Approach (ADASYN):

This approach depends on the methodology of SMOTE. The purpose is to oversample the minority class. The difference here is focusing on the distribution of data to generate new instances. The regions that are harder to learn or have low density is selected to generate more new synthetic samples while generating low or none of synthetic examples on the regions that have high density (He et al., 2008).

4. SMOTE and Tomek links (SMOTETomek):

This is a hybrid technique for oversampling which combines SMOTE and Tomek links to optimize the performance of the model. After applying SMOTE technique on the minority class. Then, the Tomek links work in samples done by SMOTE to remove the overlap between the two classes. Therefore, the instances in the

oversampled regions are removed from majority and the minority classes that are closest to each other (Fitriyani et al., 2020).

5. SMOTE and Edited Nearest Neighbours (SMOTEENN):

This is another hybrid technique that combines SMOTE and under-sampling ENN techniques. First, the SMOTE works on the minority class. Second, the ENN method finds the instances in the dataset that are misclassified by its nearest neighbours. Then removing the instances from both majority and minority classes (Manju & Nair, 2019).

Table 3.3 presents the size of the dataset after applying each technique for resampling. It is noticing the same size in random oversampling and SMOTE technique with 832 instances. The increase was in the minority class. While a slight difference in total between ADASYN and SMOTETomek. However, on undersampling technique (SMOTEENN) there was a high decrease by 58.41% in the positive class (majority) and an increase of 62% in the negative class (minority).

Table (3.3): Dataset Size After Applying Various Resampling Techniques

Balancing Technique		Number of Negative cases		Number of Positive cases		before resampling	Total after resampling	Difference on total
		Before	After	Before	After			
1	Random Oversampling	167	416	416	416	583	832	+249
2	SMOTE	167	416	416	416	583	832	+249
3	ADASYN	167	373	416	416	583	789	+206
4	SMOTETomek	167	397	416	397	583	794	+211
5	SMOTEENN	167	270	416	173	583	443	-140

3.3.5 Feature Selection

Feature Selection: this is a process to reduce the irrelevant features from the dataset to improve the performance of the model. This research applied two techniques which are InfoGainAttributeEval and CorrelationAttributeEval as attribute Evaluation. The search method was Ranker which ranks features by their individual measurements (Weka, n.d.). InfoGainAttributeEval evaluates the value of features by measuring information gain according to the class (Bayrak et al., 2019). Another is CorrelationAttributeEval which evaluates the correlation between feature and target class. The values are used to obtain a subset of attributes that are highly relevant (Armya et al., 2021). Table 3.4 presents the results obtained by applying feature selection:

Table (3.4): The Results by Applying Feature Selection

Attributes Evaluation			
InfoGainAttributeEval		CorrelationAttributeEval	
Ranked	Attribute	Ranked	Attribute
0.16015	TB	0.3245	DB
0.1548	DB	0.2918	TB
0.12719	Sgpt	0.2539	Alkphos
0.12451	Sgot	0.2184	Sgpt
0.08926	Alkphos	0.2045	Sgot
0.07845	Agratio	0.1955	Agratio
0.02912	ALB	0.1912	ALB
0.01913	Age	0.1323	Age
0.0066	Gender	0.0955	Gender
0	TP	0.0412	TP

As shown in Table 3.4 the highest ranked achieved by TB and DB attribute in both feature selection techniques with rate 0.16015 and 0.3245 by `infoGainAttributeEval` and `CorrelationAttributeEval`, respectively. This means those attributes are very important for classifying liver disease patients. However, the lowest ranked attribute in both feature selection techniques applied in this study is TP with values 0 and 0.0412 on `InfoGainAttributeEva` and `CorrelationAttributeEval`, respectively. This attribute has very low or no effect for classifying the target class. As a result, the TP attribute was selected to be removed from the dataset to run the classification and clustering algorithms.

3.4 Machine Learning Algorithms

This section describes the machine learning models used in this research. This work attempts to detect liver disease through two techniques of machine learning: classification and clustering algorithms. The next subsection explains the algorithms applied in each phase:

3.4.1 Classification Algorithms

The classification algorithms we used are:

- K-Nearest Neighbour (KNN) classifier: it is a classification and one of the simplest models. This model uses different methods for computing the distance between training elements and new element. The nearest neighbours among the classes are determined based on the majority voting. There are different distance metrics such as Euclidean, Manhattan (City block), Correlation, and Cosine (Liu et al., 2021) and (Poddar et al., 2019). In this work, K equal to 1 was selected as the optimal

value after tested different values of K (1, 3, 5, 7) and Manhattan distance used which computed as follows:

$$d(x, y) = \sum_{i=1}^n |x_i - y_i| \quad (3.2)$$

where, n denotes number of variables, x_i and y_i are the variables of vectors x and y, respectively, in the two-dimensional vector space.

- Random Forest (RF): is a supervised machine learning algorithm and is used for both classification and regression problems. It is randomly generated decision trees and uses bagging where different learning algorithms are combined to improve the predictive accuracy. A refined result obtained by bagging operation after making multiple decision trees and synthesizes them together (Krishnani et al., 2019). The model builds from the best attribute that contributes the most by splitting a node which using a random subset of features. For each feature, adding random threshold values increases the result further. It is one of the best machine learning algorithms and avoids the problem of overfitting. Also, RF is used to score the features and is robust to outlier values.
- Adaptive Boosting (adaBoost): it is used for classification purposes. It combines multiple weak learners to build a strong classifier. AdaBoost is an ensemble classifier that aims to reduce the error rate and can improve classification accuracy. It works by assigning higher weights to misclassified instances. This process of updating weights continues until the criteria of the weak learners are satisfied. The maximum vote received by

the label indicates the final prediction (Kalagotla et al., 2021). In this work we applied different classifiers as base learners and the J48 was selected among them which scored good performance.

- **Decision Tree J48 (DT):** is a machine learning algorithm that can be used on classification and regression problems. Each node refers to a feature and each leaf node refers to a class label. The information gain ratio or Gini index is computed to decide which attribute splits the data. This process is called feature selection (Rabbi et al., 2020). It can be calculated using the following equations:

Information Gain (IG):

$$IG = Entropy(S) - [(Weighted\ Avg) * Entropy\ (each\ feature)] \quad (3.3)$$

where, S is the total number of samples, and Entropy is a metric to measure the impurity in the attribute while Entropy can be calculated as:

$$Entropy = - \sum_{i=1}^n p_i \log_2 p_i \quad (3.4)$$

where, p_i denotes the probability of randomly selecting an example in class i.

Gini Index:

$$Gini\ Index = 1 - \sum_{i=1}^n (P_i)^2 \quad (3.5)$$

Gini Index computes the sum of the squared probabilities of each class p_i and then subtracts it from one.

- **Decision Stump (DS):** this classifier is a one-level decision tree. The root node is the only node on DS which performs the split test based on one attribute. The weighted classification error is

minimized by the selected threshold value. The prediction is made in the algorithm based on one feature (Bereta, 2019).

- **Stacking:** it is an ensemble learning technique which is built from two layers. The first layer contains base learners and the second layer has a meta classifier. Stacking aims to improve the performance of weak learners. In this work, three classifiers have been used as base learners: KNN, adaBoost, and RF. While Decision Stump is used as a meta classifier. The predicted output of base learners is used as the input to train on meta classifier, then obtain the final prediction (Fitriyani et al., 2020). KNN, adaBoost, and RF have been used by many researchers which scored good performance. In this work we have combined those classifiers to increase the performance.
- **Voting classifier:** This is a meta classifier that combines different models to predict the target class. Each classifier makes prediction in a trained dataset then the final prediction is obtained by an aggregating method. The voting technique applied in this work is based on majority voting between base classifiers (Kumari et al., 2021).

3.4.2 Clustering Algorithms

The used clustering algorithms are:

- **K-Means:** is an unsupervised learning algorithm that can group unlabelled datasets. Similar instances are clustered together. It is a centroid-based algorithm that can be used for classification. Each cluster has its own centroid. The purpose of the algorithm

is to minimize the sum of squared distance between the data points and the centroid, which is calculated by a different distance function such as Euclidean Distance. That is an iterative process to assign each data point to the closest centroid. K is the number of clusters and should be predefined in the algorithm (Antony et al., 2021). In this study, the target attribute has two classes for positive and negative cases for liver disease patients. Thus, the K is defined as 2 for patients with liver disease and patients without.

The formula of Euclidean distance as follows:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (3.6)$$

where, d is the distance between two points, (x1, y1) is the coordinates for one point, and (x2, y2) is the coordinates for another point.

- Gaussian Mixture Model (GMM): this is a clustering model which assesses the density of examples in the dataset then calculates the probabilities of each example in a specific cluster. It will make more computations for the algorithm. Each cluster has its own mean and covariance. Also, the example determines a certain cluster after estimating the highest probability. This model depends on the expectation-maximization method to fit the parameters of the distribution (Sbirna & Sbirna, 2020).
- Expectation Maximization (EM): this is a machine learning clustering algorithm. The EM algorithm is an iterative approach that has two steps. In the first step, obtain the expectation of the log-likelihood of the parameters based on the initial values of parameters. In the second step, maximize the parameters of the model obtained in the previous step, then can obtain the

estimations of the parameters. These two steps continue till they reach the optimal estimates of parameters (Zeng & Wang, 2016).

- Density-based Spatial Clustering of Applications with Noise (DBSCAN): The model finds the high density to create a cluster of data points. The point belongs to a cluster when that point is close to many points from the cluster. This algorithm is suitable for data that consist of similar density. There are two main parameters `eps` and `minPts`. `eps` is the distance that specifies the neighborhoods, while `minPts` is the minimum number of neighbours (Antony et al., 2021).

3.5 Evaluation Measures

The models have been applied on the dataset with 10-fold cross-validation. In this work, five measurements were used in classification algorithms to evaluate the algorithms for the detection of liver disease while two measurements are used in clustering algorithms. The classification and clustering metrics are:

➤ Classification Measurements:

The confusion matrix is used to calculate the performance of the classification models. It contains a summary of correct and incorrect predictions. The four types of results in the confusion matrix are as follows:

TP is true positive instances, TN is true negative instances, FP is false positive instances, and FN is false negative instances.

Then the following measures can be calculated as follows:

- Accuracy:

This measure is calculating the percentage of the number of instances that are correctly classified. Calculated as follows:

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (3.7)$$

- Recall:

This measure is called sensitivity or True Positive Rate (TPR). Which is the ratio of positive instances that are correctly classified to the total number of positive instances. Recall is evaluated as:

$$\text{Recall} = (\text{TP}) / (\text{TP} + \text{FN}) \quad (3.8)$$

- Precision:

This is the ratio of correctly classified instances from positive class to the total number of instances that predicted in positive class. Which using the following formula:

$$\text{precision} = (\text{TP}) / (\text{TP} + \text{FP}) \quad (3.9)$$

- Receiver Operating Characteristics (ROC):

This evaluation metric plots a graph on different threshold values between true positive rate and false positive rate. Which means how the classifier can distinguish between classes.

- RMSE:

Defined as the square root of the mean of the squared differences between actual observations and prediction values. RMSE is using the following equation:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{y} - y)^2} \quad (3.10)$$

where, $\hat{y}$ is the predicted value of y , and N is the number of observations.

➤ Clustering Measurement:

▪ Silhouette coefficient:

This measure evaluates how the points in the same cluster are close to each other and how far they are from other clusters. The values range between -1 and 1 where a better score is close to 1. The following formula is used to calculate the score of silhouette coefficient:

$$\text{Silhouette coefficient} = (q-p) / \text{Max}(p, q) \quad (3.11)$$

where, p is the mean distance within a cluster.

q is the average distance between a point and the nearest of different clusters.

▪ Adjusted Rand Index (ARI):

ARI uses actual observations and the predicted values from the cluster model to evaluate the similarity between them. The output of this metric is between 0 and 1 where values close to 1 indicate the better cluster. Using the following equation:

$$\text{ARI} = \text{RI} - \text{expected(RI)} / \text{Max(RI)} - \text{expected(RI)} \quad (3.12)$$

where, RI is the rand index, which calculates a similarity measure between two clusters.

3.6 Implementation Tools and Parameters Setting

This research used Weka software (version 3.8.5) and Python on a public source (google colab) as environments to execute the experiments in this study (Eibe et al., 2016) and (Google Colaboratory, n.d.). The first

platform is Weka which is a machine learning tool that contains algorithms for classification, clustering, regression, and different ML algorithms techniques. This research has applied feature selection technique, classification, and visualizing clustering models on Weka software. Meanwhile, the second platform is Python language which is an open source that consists of different algorithms and various libraries for ML purposes. Python has been used for preprocessing techniques such as variable encoding, imputation missing values, normalization, and balancing. Also, as this research used more evaluation measurements (Silhouette coefficient and Adjusted rand index) for comparing clustering models, the Python tool was selected to run clustering algorithms. Both measurements are not available on Weka software.

To train and evaluate ML models we need to select a procedure for splitting the dataset. The dataset selected in this study has 583 instances (416 with liver disease and 167 without liver disease) which is considered a small dataset. Therefore, we trained the models depending on a 10-fold cross-validation method that would help us for better estimation and to avoid over-fitting in the model.

This work has been applied different ML experiments with specific parameters for each model. Table 3.5 presents the parameters used on Weka for individual model and ensemble algorithms (voting and stacking).

Table (3.5): Parameters Setting in this research on Weka

Techniques	Algorithms	Parameters
classification	IBK	KNN=1 distanceWeighting: weight by 1/distance distance Function: ManhattanDistance -R first-last

		Also, same parameters applied on voting and stacking
	adaBoost	J48 used as a base learner numIteration = 200 Also, same parameters applied on voting and stacking.
	RF	As default
	voting	Different versions of voting 1.classifiers = IBK + adaBoost + RF 2.classifiers = IBK + adaBoost 3.classifiers = IBK + RF 4.classifiers = adaBoost + RF For all voting versions: combinationRule = Majority Voting
	stacking	Different versions of stacking 1.classifiers = IBK + adaBoost + RF 2.classifiers = IBK + adaBoost 3.classifiers = IBK + RF 4.classifiers = adaBoost + RF For all stacking versions: metaClassifier = DecisionStump
clustering	EM	numClusters = 2 Ignore attributes: Gender and class

3.7 Summary

This chapter showed the methodology of this research. The selected dataset was described in detail by showing the source of the

dataset, attributes with description, number of instances, and distribution of target class. Then, a description of preprocessing techniques applied in this work. Which is related to variable encoding, imputation missing values, normalization, various techniques of balancing, and feature selection. Furthermore, this chapter explained two types of ML algorithms executed in this study which are classification and clustering algorithms. In addition, the chapter presented the environments selected to run the experiments with details of the used parameters. Finally, the methods to evaluate the performance of the models are explained in the evaluation measures section.

Chapter Four

Empirical Results

4.1 Overview

4.2 Classification Models Results

4.2.1 Applying Classification Models

4.2.2 Classification Models Analysis

4.3 Clustering Models Results

4.3.1 Applying Clustering Models

4.3.2 Clustering Models Analysis

4.4 Summary

Chapter Four

Empirical Results

4.1 Overview

This chapter presents and analyzes the experimental results for classification and clustering algorithms. Various proposed models have been implemented in the ILPD. Thus, this research has used different evaluation criteria to compare and check the models' performance.

4.2 Classification Models Results

This section shows the results after implementing the feature selection technique. Then, compare and analyze the results between the models. The results before feature selection are present in the appendices.

4.2.1 Applying Classification Models

In this section, the results present in detail after applying the models in different techniques of oversampling and undersampling.

A. SMOTE Technique

The finding after implementing the models with SMOTE technique summarizes in Table 4.1.

Table (4.1): Results After Applying SMOTE Technique

Algorithms		Acc.	Recall		Precision		ROC	RMSE
			0	1	0	1		
MODELS	IBK	81.37%	0.91	0.72	0.76	0.89	0.89	0.43
	RF	82.33%	0.89	0.76	0.79	0.87	0.90	0.36
	adaBoost	82.69%	0.88	0.78	0.80	0.86	0.89	0.41
STACKING	IBK+RF	84.13%	0.84	0.85	0.85	0.84	0.85	0.37
	IBK+adaBoost	81.13%	0.76	0.87	0.85	0.78	0.80	0.39
	RF+adaBoost	80.65%	0.78	0.83	0.82	0.79	0.82	0.40
	RF+IBK+adaBoost	80.77%	0.76	0.85	0.84	0.78	0.81	0.40
VOTING	IBK+RF	81.37%	0.91	0.72	0.76	0.89	0.81	0.43
	IBK+adaBoost	83.41%	0.88	0.79	0.81	0.87	0.83	0.41
	RF+adaBoost	82.45%	0.88	0.77	0.80	0.86	0.83	0.42
	RF+IBK+adaBoost	83.41%	0.90	0.77	0.80	0.88	0.83	0.41

As shown in Table 4.1, there is a good performance on applying SMOTE technique on the dataset. The highest accuracy 84.13% achieved by stacking technique (IBK+RF) with slight improvement. However, there are negative effects on the rest of stacking models which include the minimum accuracy of 80.65% scored by (RF+AdaBoost). Furthermore, voting techniques obtained slight increases on two models (IBK+AdaBoost) and (RF+IBK+AdaBoost). IBK model scored the highest rate 91% on the negative class by recall measurement, the same as voting (IBK+RF) model without effect of using RF. Also, both models obtained the highest score of 89% by precision on positive class and by RMSE with 43%. Whereas they achieved the lowest rate (72%,76%) by recall measurement on positive class and precision on negative class,

respectively. In another measurement, ROC obtained the highest rate of 90% by RF. The same model scored the lowest RMSE with 36%. Moreover, stacking (IBK+AdaBoost) produced the lowest rate of 76% and 78% in terms of recall on negative class and precision on positive class, respectively. In addition, the highest rate of 87% and 85% were obtained by the same model in terms of recall on positive class and precision on negative class, respectively.

Comparing the result of stacking with voting technique. It's clear the stacking technique achieved better outcomes in terms of accuracy, recall on positive class, precision on negative class, and RMSE. However, the voting outperforms stacking models in two measurements: recall on negative class and precision on positive class.

B. ADASYN Technique

The finding after implementing the models with ADASYN technique summarizes in Table 4.2.

Table (4.2): Results After Applying ADASYN Technique

Algorithms		Acc.	Recall		Precision		ROC	RMSE
			0	1	0	1		
MODELS	IBK	80.61%	0.90	0.72	0.74	0.89	0.89	0.44
	RF	80.48%	0.85	0.76	0.76	0.85	0.89	0.37
	adaBoost	82.26%	0.85	0.80	0.79	0.86	0.88	0.42
STACKING	IBK+RF	83.65%	0.83	0.84	0.82	0.85	0.84	0.37
	IBK+adaBoost	83.65%	0.83	0.84	0.82	0.85	0.84	0.37
	RF+adaBoost	82.13%	0.87	0.78	0.78	0.87	0.82	0.38
	RF+IBK+adaBoost	83.65%	0.83	0.84	0.82	0.85	0.84	0.37

VOTING	IBK+RF	80.61%	0.90	0.72	0.74	0.89	0.81	0.44
	IBK+adaBoost	82.00%	0.86	0.79	0.78	0.86	0.82	0.42
	RF+adaBoost	82.26%	0.86	0.79	0.79	0.86	0.82	0.42
	RF+IBK+adaBoost	82.51%	0.88	0.78	0.78	0.88	0.83	0.42

In general, Table 4.2 shows the same performance as the previous one after applying ADASYN technique with a slight change in some models. In terms of accuracy, stacking models produced the highest rate with 83.65% in three models. Also, the same models achieved the maximum rate of 84% in recall measurement on positive class and got 82% in precision on negative class. However, they obtained the lowest score of 83% on recall for negative class and achieved 85% on precision for positive class. Another ensemble technique, voting classifier (IBK+RF) achieved the lowest performance 80.61%, 72%, 74%, 81%, and 44% in terms of accuracy, recall on positive class, precision on negative class, ROC, and RMSE, respectively. Also, the same results are scored by IBK model on all criteria unless ROC. This means RF did not affect the performance on voting (IBK+RF) rather it has the negative effect of decreasing ROC. In addition, both models obtained the same highest rate of recall for negative class with 90% and precision for positive class with 89%. Moreover, the RF model got the best results 89% in terms of ROC.

From the result in Table 4.2, we can notice that the voting classifier didn't improve the performance while the stacking classifier could make a slight increase. Applying staking models on ADASYN oversampling technique in the ILPD could make a good difference in recall for positive cases and precision measurement for negative cases.

C. SMOTETomek Technique

The finding after implementing the models with SMOTETomek technique summarizes in Table 4.3.

Table (4.3): Results After Applying SMOTETomek Technique

Algorithms		Acc.	Recall		Precision		ROC	RMSE
			0	1	0	1		
MODELS	IBK	82.62%	0.95	0.71	0.76	0.93	0.93	0.42
	RF	81.23%	0.87	0.76	0.78	0.85	0.91	0.35
	AdaBoost	83.75%	0.87	0.80	0.82	0.86	0.90	0.40
STACKING	IBK+RF	87.15%	0.87	0.87	0.87	0.87	0.87	0.34
	IBK+AdaBoost	87.15%	0.87	0.87	0.87	0.87	0.87	0.34
	RF+AdaBoost	84.63%	0.83	0.86	0.86	0.84	0.86	0.36
	RF+IBK+AdaBoost	87.15%	0.87	0.87	0.87	0.87	0.87	0.34
VOTING	IBK+RF	82.87%	0.95	0.71	0.77	0.93	0.83	0.41
	IBK+AdaBoost	84.63%	0.89	0.81	0.82	0.88	0.85	0.39
	RF+AdaBoost	83.75%	0.87	0.80	0.82	0.86	0.84	0.40
	RF+IBK+AdaBoost	84.51%	0.90	0.79	0.81	0.89	0.85	0.39

Table 4.3 presented the result after applying SMOTETomek technique on the ILPD. There is a better improvement in the result compared with two previous oversampling techniques (SMOTE and ADASYN). As shown in Table 4.3, most of the stacking models got the maximum rate of accuracy with 87.15 and could increase the performance by more than 3%. Whereas, voting models got a slight increase in accuracy. In contrast, RF produced the minimum rate of 81.23%. In terms of recall, the highest rate achieved by stacking models was 87% on the positive class while IBK and voting classifier (IBK+RF) obtained the

maximum value of 95% on the negative class. On another measurement, the same voting model and IBK scored the highest rate on precision with 93% for positive class. However, in the same measurement three stacking models produced the best value of 87% for the negative class. Furthermore, the lowest performance was by voting classifier with 83% and 41% in terms of ROC and RMSE, respectively. Whereas the best performance 34% achieved by three stacking models in terms of RMSE. In addition, IBK obtained the highest rate of 93% on ROC measurement.

It is very clear from Table 4.3 the ability of stacking classifiers on improving the performance on the accuracy, the weighted average of (recall, precision), and RMSE except for ROC. Whereas the voting classifiers made a slight increase of only one measurement which is the accuracy.

D. Random Oversampling Technique

The finding after implementing the models with Random Oversampling technique summarizes in Table 4.4.

Table (4.4): Results After Applying Random Oversampling Technique

Algorithms		Acc.	Recall		Precision		ROC	RMSE
			0	1	0	1		
MODELS	IBK	85.82%	0.96	0.76	0.80	0.95	0.96	0.38
	RF	86.90%	0.96	0.78	0.81	0.95	0.98	0.29
	adaBoost	89.30%	0.95	0.84	0.86	0.94	0.95	0.33
STACKING	IBK+RF	95.43%	0.91	1	1	0.92	0.94	0.21
	IBK+adaBoost	95.43%	0.91	1	1	0.92	0.95	0.21
	RF+adaBoost	95.31%	0.91	0.99	0.99	0.92	0.94	0.21
	RF+IBK+	95.43%	0.91	1	1	0.92	0.95	0.21

	adaBoost							
VOTING	IBK+RF	85.82%	0.96	0.76	0.80	0.95	0.86	0.38
	IBK+adaBoost	89.30%	0.95	0.84	0.85	0.94	0.89	0.33
	RF+adaBoost	89.30%	0.95	0.84	0.86	0.94	0.89	0.33
	RF+IBK+ adaBoost	88.82%	0.96	0.82	0.84	0.95	0.89	0.33

As shown in Table 4.4, there are significant improvements in the performance using all versions of stacking models. The accuracy rate was increasing by 6% and 9.6%. In addition, the positive cases could be classified by 100% by the same models. This means the models have the precision on negative cases by 100%. Also, stacking techniques could decrease the RMSE. However, stacking classifiers did not record any increment on ROC, rather they had a negative effect with a high decrease in recall for negative cases and precision for positive cases. On the other hand, implementing four versions of voting classifiers on the dataset using random oversampling technique have no change in the performance in all evaluations criteria. Moreover, the lowest performance scored by IBK and voting classifier (IBK+RF) on the accuracy, recall for positive class, precision for negative class, and RMSE by 85.82%, 76%, 80%, and 38%, respectively. While both and RF classifiers produced the highest rate of recall for negative class and precision for positive class.

It's obvious from the result, in general stacking outperforms voting classifiers in the performance for classifying positive cases. Whereas each model IBK, RF, and adaBoost run individually have the best performance on classifying negative cases. This indicates that voting technique has no effect on classifying negative cases.

E. SMOTEENN Technique

The finding after implementing the models with SMOTEENN undersampling technique summarizes in Table 4.5.

Table (4.5): Results After Applying SMOTEENN Technique

Algorithms		Acc.	Recall		Precision		ROC	RMSE
			0	1	0	1		
MODELS	IBK	96.39%	0.996	0.91	0.95	0.99	0.997	0.19
	RF	91.20%	0.96	0.83	0.9	0.94	0.97	0.25
	adaBoost	93.45%	0.98	0.87	0.92	0.96	0.95	0.25
STACKING	IBK+RF	97.07%	0.97	0.97	0.98	0.96	0.98	0.17
	IBK+adaBoost	97.07%	0.97	0.97	0.98	0.96	0.98	0.17
	RF+adaBoost	91.87%	0.95	0.87	0.92	0.92	0.92	0.27
	RF+IBK+adaBoost	97.07%	0.97	0.97	0.98	0.96	0.98	0.17
VOTING	IBK+RF	96.39%	0.996	0.91	0.95	0.99	0.96	0.19
	IBK+adaBoost	93.68%	0.98	0.87	0.92	0.96	0.93	0.25
	RF+adaBoost	93.45%	0.98	0.87	0.92	0.96	0.92	0.26
	RF+IBK+adaBoost	93.45%	0.98	0.87	0.92	0.96	0.92	0.26

The results after applying different models on the ILPD with SMOTEENN undersampling technique are shown in Table 4.5. There was no significant increase in the performance of this experiment except for the recall of positive cases. This increase was by stacking classifiers. In addition, the same classifiers have slight increases in the accuracy rate. Also, they were making a small reduction on RMSE 17%. Moreover, the best model that could detect negative cases is IBK algorithm by 99.60%. Furthermore, the IBK model produced the highest rate of precision for positive cases 99% and ROC with 99.70%. However, RF obtained the lowest rate of accuracy, recall for positive cases and precision for negative cases with 91.20%, 83%, and 90%, respectively. While the rest

measurements have obtained the lowest rate by stacking classifier (RF+adaBoost).

Comparing the result between stacking with voting technique. The latter did not improve the performance of the models similarly to voting classifiers in Table 4.4. Regarding stacking technique, three classifiers have a positive effect on the performance while one classifier (RF+adaBoost) has a negative effect on four criteria: recall of negative class, precision of positive class, ROC, and RMSE.

4.2.2 Classification Models Analysis

In this part, comparing and analyzing the best classification models were running in different oversampling and under-sampling techniques. Table 4.6 and Fig. 4.1 summarize the results of accuracy, ROC, and RMSE.

Table (4.6): Summary of Accuracy, ROC, and RMSE

Preprocessing Technique	Models	Accuracy	ROC	RMSE
SMOTE	IBK	81.37%	0.89	0.43
	Stacking	81.13%	0.80	0.39
	Voting	81.37%	0.81	0.43
ADASYN	IBK	80.61%	0.89	0.44
	Stacking	83.65%	0.84	0.37
	Voting	80.61%	0.81	0.44
SMOTETomek	IBK	82.62%	0.93	0.42
	Stacking	87.15%	0.87	0.34
	Voting	82.87%	0.83	0.41
Random Oversampling	IBK	85.82%	0.96	0.38
	Stacking	95.43%	0.94	0.21
	Voting	85.82%	0.86	0.38
SMOTEENN	IBK	96.39%	0.997	0.19
	Stacking	97.07%	0.98	0.17
	Voting	96.39%	0.96	0.19

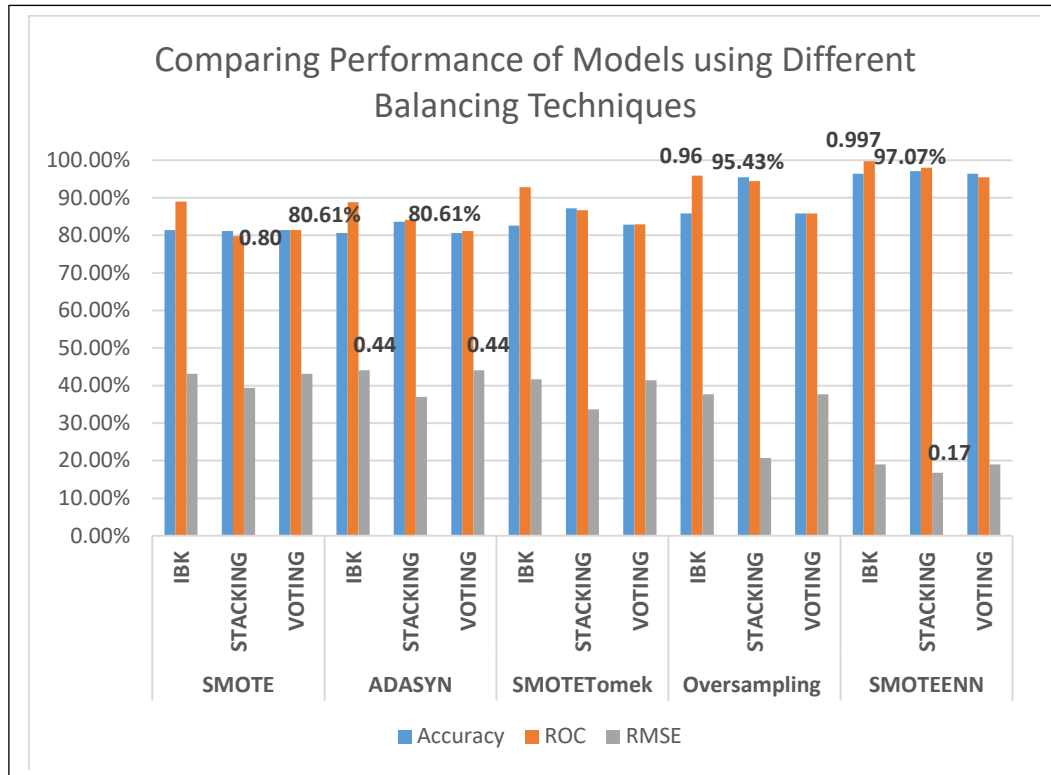
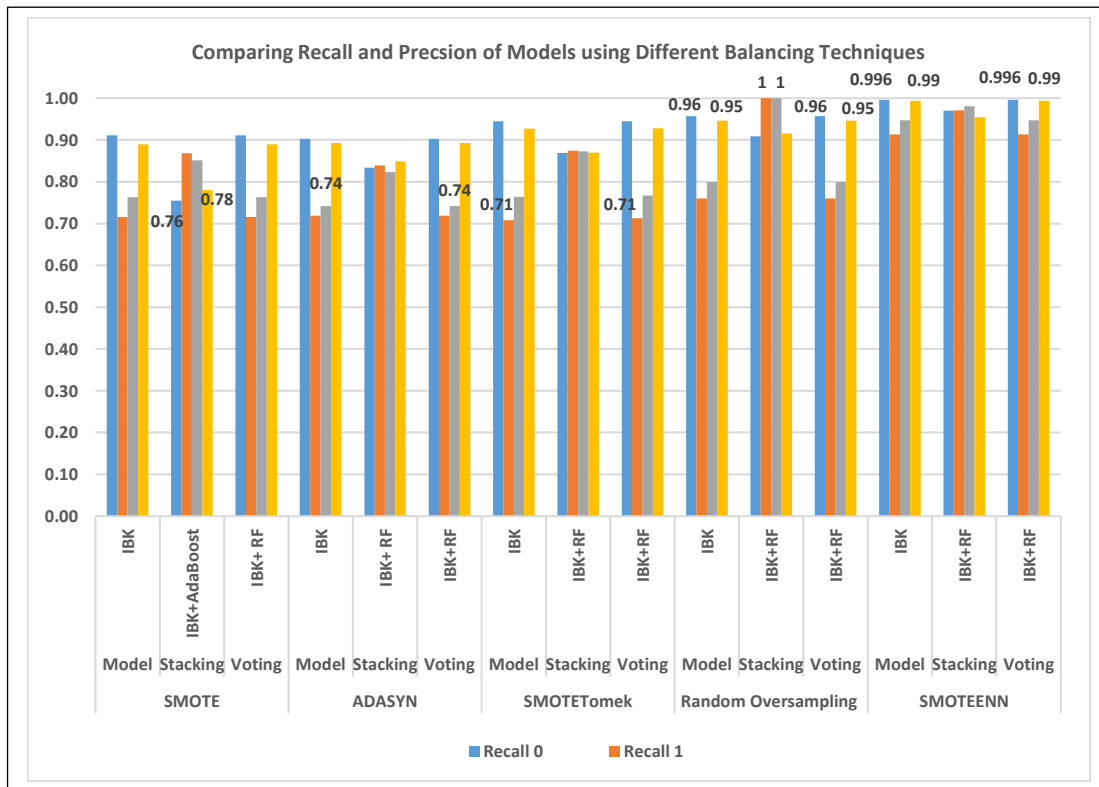


Fig. 4.1: Comparing of Accuracy, ROC and RMSE

As shown in Table 4.6 and Fig. 4.1, the highest performance was achieved in this work by undersampling technique (SMOTEENN) in terms of accuracy, ROC, and RMSE with 97.07%, 99.70%, and 0.17, respectively. However, the lowest accuracy 80.61% and RMSE 0.44 was produced in oversampling technique (ADASYN) by IBK and voting models. Whereas the stacking model using SMOTE technique got the lowest rate 0.80 in terms of ROC. Furthermore, among oversampling techniques the highest accuracy 95.43% scored by stacking algorithms using random oversampling. In addition, the same technique of random oversampling obtained the maximum rate of 0.96 in terms of ROC. The most important finding is that stacking technique could improve the accuracy in all cases while voting did not improve. In contrast, there was a decrease in ROC by stacking algorithms. Table 4.7 and Fig. 4.2 conclude the outcome of recall and precision in this work.

Table (4.7): Summary of Recall and Precision

Preprocessing Technique	Algorithms		Recall		Precision	
			0	1	0	1
SMOTE	Model	IBK	0.91	0.72	0.76	0.89
	Stacking	IBK+AdaBoost	0.76	0.87	0.85	0.78
	Voting	IBK+RF	0.91	0.72	0.76	0.89
ADASYN	Model	IBK	0.90	0.72	0.74	0.89
	Stacking	IBK+RF	0.83	0.84	0.82	0.85
	Voting	IBK+RF	0.90	0.72	0.74	0.89
SMOTETomek	Model	IBK	0.95	0.71	0.76	0.93
	Stacking	IBK+RF	0.87	0.87	0.87	0.87
	Voting	IBK+RF	0.95	0.71	0.77	0.93
Random Oversampling	Model	IBK	0.96	0.76	0.80	0.95
	Stacking	IBK+RF	0.91	1	1	0.92
	Voting	IBK+RF	0.96	0.76	0.80	0.95
SMOTEENN	Model	IBK	0.996	0.91	0.95	0.99
	Stacking	IBK+RF	0.97	0.97	0.98	0.96
	Voting	IBK+RF	0.996	0.91	0.95	0.99

**Fig. 4.2:** Comparing Recall and Precision

As presented in Table 4.7 and Fig. 4.2, the maximum rate of 100% was produced by stacking algorithms in recall for positive cases and precision for negative cases. While for recall of positive cases in SMOTETomek technique was scored the minimum value 0.71 by IBK and voting model. Also, by the same models in ADASYN technique was achieved the lowest rate of 0.74 for precision of negative cases. Another measurement, using SMOTEENN technique produced the highest rate of 0.996 of recall for negative cases and precision for positive cases. However, the lowest value for the same evaluation was achieved in SOMTE technique by a staking algorithm (IBK+adaBoost).

Table 4.8 provides an overview using a confusion matrix of the best results for classification models of patients with liver disease and without in this work. Stacking models using random oversampling did not record any misclassified cases for patients with liver disease but there are 38 cases misclassified as patients with liver disease. However, among oversampling techniques, RF model achieved the best result in this study for negative cases which misclassified only 16 cases using random oversampling technique. Also, the same model misclassified 93 cases as patients without liver disease. Whereas, using an undersampling technique could achieve the best result for patients without liver disease with only one case misclassified. In addition, 15 patients were misclassified without liver disease.

Table (4.8): Summary of Best Confusion Matrix

Model	class	0	1	Accuracy
Three versions of Stacking models using Random Oversampling Technique (IBK+RF, IBK+adaBoost,	0	378	38	95.43%
	1	0	416	

and RF+IBK+adaBoost)				
RF model using Random Oversampling Technique	0	400	16	86.90%
	1	93	323	
IBK model and Voting (IBK+RF) using Under sampling Technique (SMOTEENN)	0	269	1	96.39%
	1	15	158	

4.3 Clustering Models Results

4.3.1 Applying Clustering Models

The results produced by three clustering algorithms on different versions of ILPD (oversampling and undersampling) techniques are presented in Table 4.9.

Table (4.9): Results of Clustering Models on Oversampling and Undersampling Techniques

Balancing Technique	Cluster model	Silhouette Coefficient	Adjusted Rand Index (ARI)	cluster 0	cluster 1	Un-clustered
SMOTE 832 Instances	k-Means	0.62	0.01	613	219	
	DBSCAN	0.6	0.01	209	597	26
	GMM	0.07	0.19	574	258	
ADASYN 789 Instances	k-Means	0.61	0.01	585	204	
	DBSCAN	0.59	0.01	194	569	26
	GMM	0.07	0.15	537	252	
SMOTETomek 794 Instances	k-Means	0.61	0	205	589	
	DBSCAN	0.59	0.01	195	573	26
	GMM	0.07	0.2	250	544	
SMOTEENN	k-Means	0.6	-0.02	345	98	

443 Instances	DBSCAN	0.57	0.02	333	88	22
	GMM	0.12	0.49	298	145	
Random Oversampling 832 Instances	k-Means	0.61	0.01	597	235	
	DBSCAN	0.59	0.02	225	579	28
	GMM	0.07	0.16	573	259	

As illustrated in Table 4.9, the highest value of silhouette coefficient is 0.6 obtained by K-means and DBSCAN. Which indicates how the instances are close to each other in the same cluster. However, the same models achieved the lowest score in terms of ARI. Furthermore, K-Means, DBSCAN, and GMM clusters have the same performance on all versions of the dataset except in SMOTEENN by GMM cluster. Which has the highest result in ARI 0.5 compared to others but got the lowest in terms of silhouette coefficient 0.12. Other findings from the results, in general, K-means and DBSCAN have the same values in all versions of the dataset even when the number of instances is different in each cluster. Also, un-clustered instances in the DBSCAN cluster did not change the performance compared with K-means result. It is noticed that using different techniques of oversampling does not make any difference in the result of clustering between the three algorithms. Also, by using undersampling technique except with the GMM cluster.

4.3.2 Clustering Models Analysis

Because of the similar results on the performance between the clustering models, the plotting was made for 3 models using SMOTE technique and only one model GMM using SMOTEENN technique due to having the highest ARI. The following Fig. 4.3, 4.4, 4.5, and 4.6 show the distribution of patients with liver disease and without in each cluster.

GMM depends on the expectation-maximization (EM) method to fit the parameters of the distribution. Thus, the plotting was done on Weka software using the EM model.

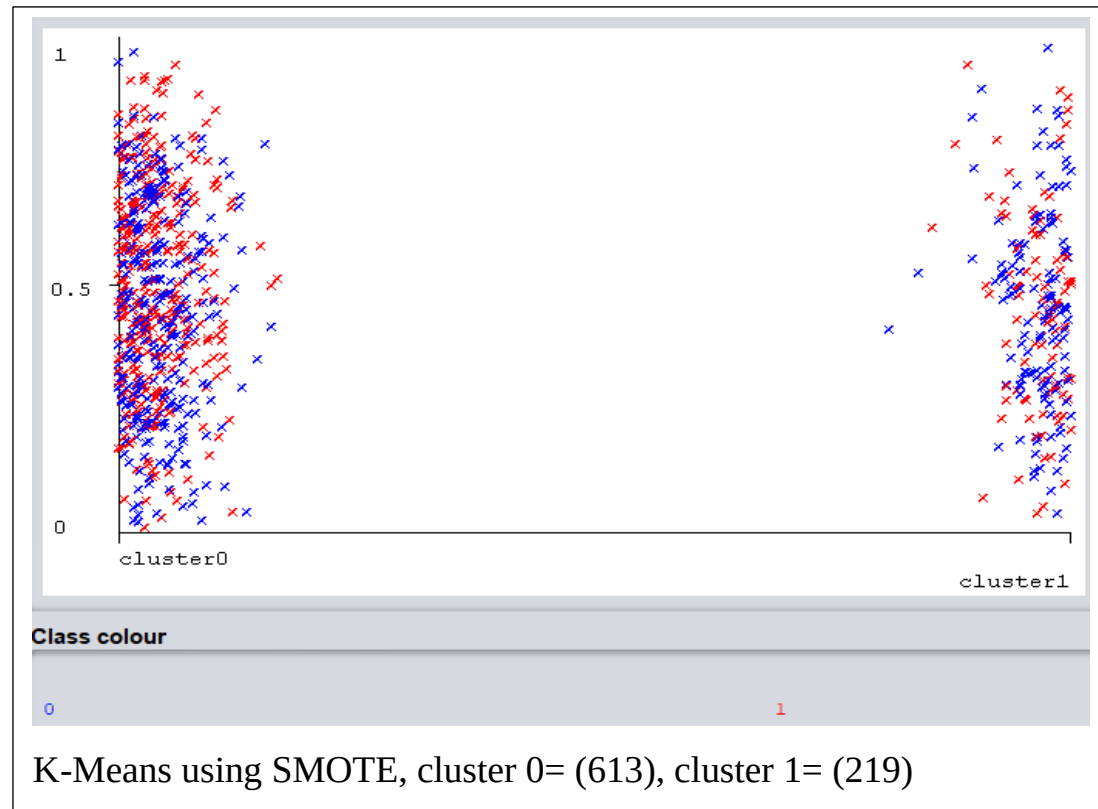


Fig. 4.3: Plotting Clustering Result of K-Means Model
using SMOTE Technique

As shown on Fig. 4.3, in the K-Means cluster there is one big cluster (0) with the majority instances from both cases. While the other cluster (1) has a small number and consists of both cases. It's clear the misclassified cases in this model in both clusters. Generally, DBSCAN on Fig. 4.4 has the likeness of K-Means cluster in the size of cluster, distribution of cases, and misclassification of patients. Whereas there is a reverse of cluster location between cluster 0 and 1. It is apparent from Fig. 4.3 and Fig. 4.4 the difficulty of detection of liver disease by K-Means and DBSCAN clusters. This refers to the small value of ARI 0.01 in both clusters in Table 4.9.

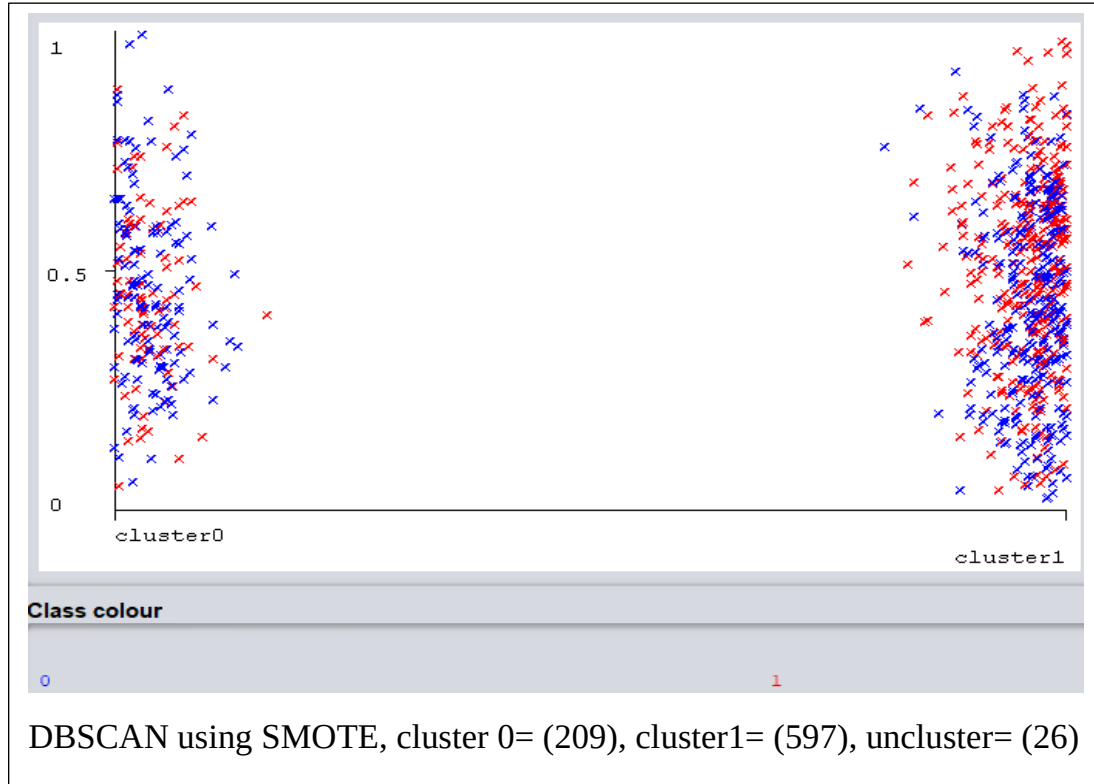


Fig. 4.4: Plotting Clustering Result of DBSCAN Model using SMOTE Technique

In Fig. 4.5, the EM model shows better performance than other cluster models. There is a clear density of red in cluster (0) and blue density in cluster (1). Regarding the size, cluster 1 is bigger than cluster 0. In addition, cluster 0 has a few blue misclassified instances. While cluster 1 consists of many red misclassified instances. This means the EM model has a good ability to detect negative cases without liver disease. This point is related to the increasing value of ARI 0.19 in this cluster. In the same way in Fig. 4.6, the EM model has a small number of misclassified cases in cluster 0. Also, this model has the highest value of ARI 0.49. which indicates the power of this model to classify patients without liver disease. This reveals the performance strength of the EM model on ILPD for negative cases, unlike K-Means and DBSCAN. Also, undersampling technique (SMOTEENN) could help the EM model to

increase ARI by a large rate. While oversampling techniques (SMOTE, ADASYN, SMOTETomek, and random oversampling) could increase ARI with a small rate in the EM model.

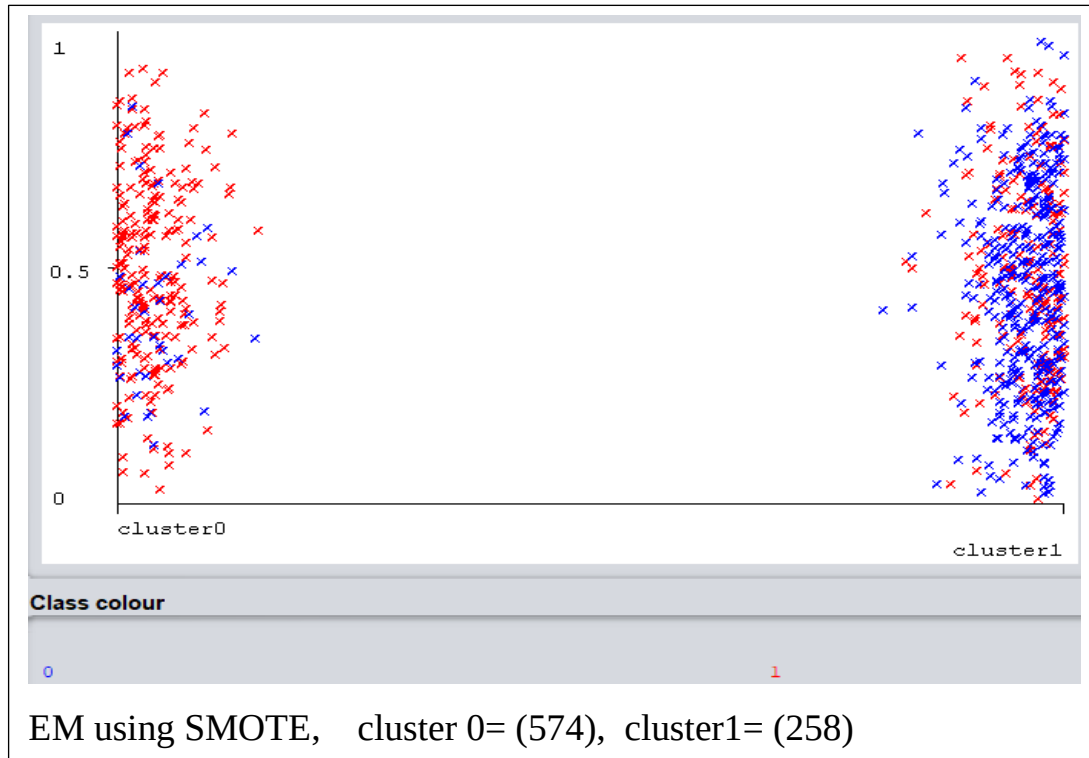


Fig. 4.5: Plotting Clustering Result of EM Model
using SMOTE Technique

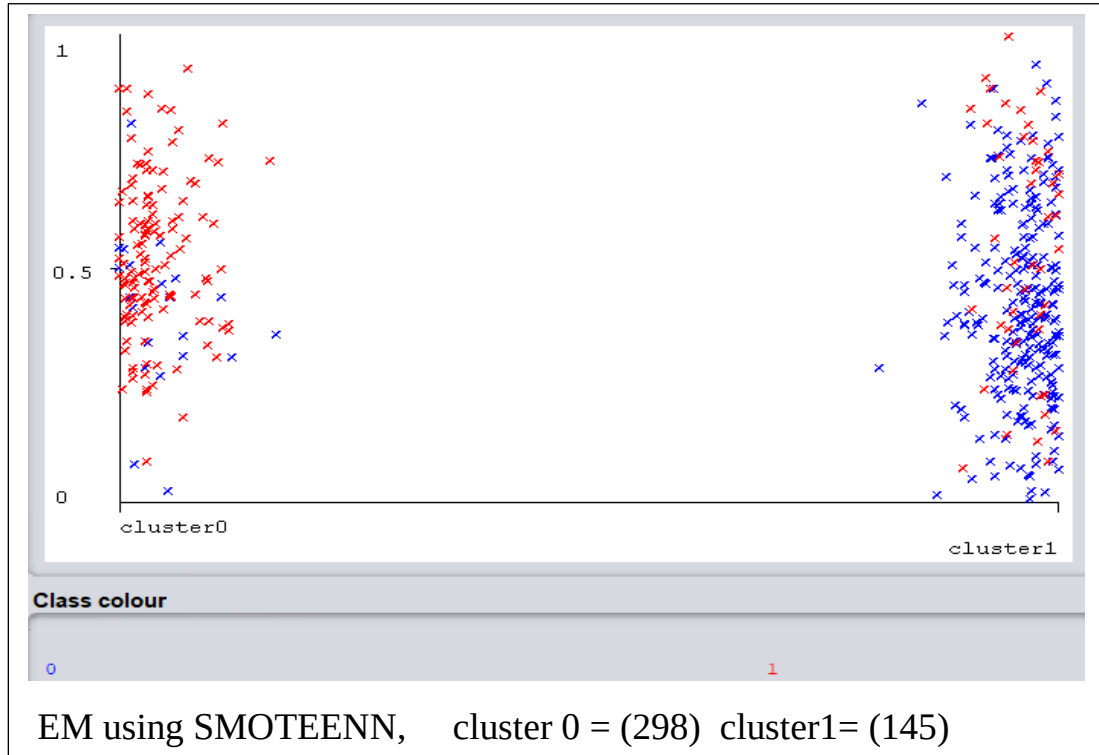


Fig. 4.6: Plotting Clustering Result of EM Model
using SMOTEENN Technique

4.4 Summary

This chapter presented the results of supervised and unsupervised machine learning after implementing feature selection. Also, the results presented in each balancing technique considered in this study. In addition, the chapter analyzed the results and compared the algorithms based on different evaluation metrics in classification and clustering algorithms.

Chapter Five

Discussion, Conclusion and Recommendations

5.1 Overview

5.2 Discussing with Related Work

5.3 Answering Research Questions

5.4 Conclusion and Recommendations

5.5 Summary

Chapter Five

Discussion, Conclusion and Recommendations

5.1 Overview

This chapter discusses and compares the results of this study with related work. Then answers the research questions. Finally, we summarize the research followed by the recommendations and future work.

5.2 Discussing with Related Work

In this section, we discuss the results achieved in this work with the related works. Stacking models in this research achieved the highest recall by 100% for positive cases which outperform Thirunavukkarasu et al. (2018) and Nahar et al. (2019) with a rate of 95.2% using LR and 93% using LogitBoost, respectively. This indicates using ensemble learning with stacking technique was a good tool applied in this research to improve the recall performance. According to ROC measurement, this study obtained 95.9% using random oversampling and 99.7% using undersampling (SMOTEENN) by IBK model which is higher than 72.5% in the study conducted by Hartatik et al. (2020) using NB model. Furthermore, in terms of accuracy, this research applied a stacking technique which scored 95.43% using random oversampling and 97.07% using undersampling (SMOTEENN). That outperform three studies conducted by Ambesange et al. (2020), Ramesh and Katheria (2019), and Simon et al. (2021) with values 91% by KNN, 89.1% by SVM, and 81%

by LR, respectively. Moreover, Gupta et al. (2020) obtained an accuracy of 93.69% after combining three models KNN, DT, and ANN using vote majority. Meanwhile, using a voting technique in this study scored 89.30% in random oversampling while achieving 96.39% in undersampling. Another study conducted by Raheem et al. (2017) concluded that using stacking and voting did not make a significant improvement in the performance. This is different from this research that was revealed from the results obtained using both techniques. According to adaBoost classifier, this research achieved 89.30% using J48 as a base learner which is less than 92.19% in a study conducted by Rabbi et al. (2020) by using extra tree as a base learner.

5.3 Answering Research Questions

This section aims to answer the research questions. The following answers the questions regarding the proposed ML model for liver disease detection.

RQ1. What is the ML potential in detecting liver disease?

According to Tables 4.1-4.8, Fig. 4.5, and Fig. 4.6, the results showed that ML models scored various results from bad to outstanding results in detecting liver disease. Therefore, ML models have the potential to detect patients with liver disease using classification and clustering models. By answering this question, we have achieved the research objective **O2**.

RQ2. Can feature selection algorithms improve the detection of liver disease?

According to Fig. 5.1, the results revealed that the feature selection technique could minimize the number of medical tests since the accuracy of stacking algorithms did not change before and after implementing feature selection (all performance results before feature selection are present in appendices). Two feature selection techniques were applied in this research: InfoGainAttributeEval and CorrelationAttributeEval to detect the most relevant attributes. Furthermore, the attribute has been removed from the dataset is TP. By answering this question, we have accomplished research objectives **O2, O3, and O4**.

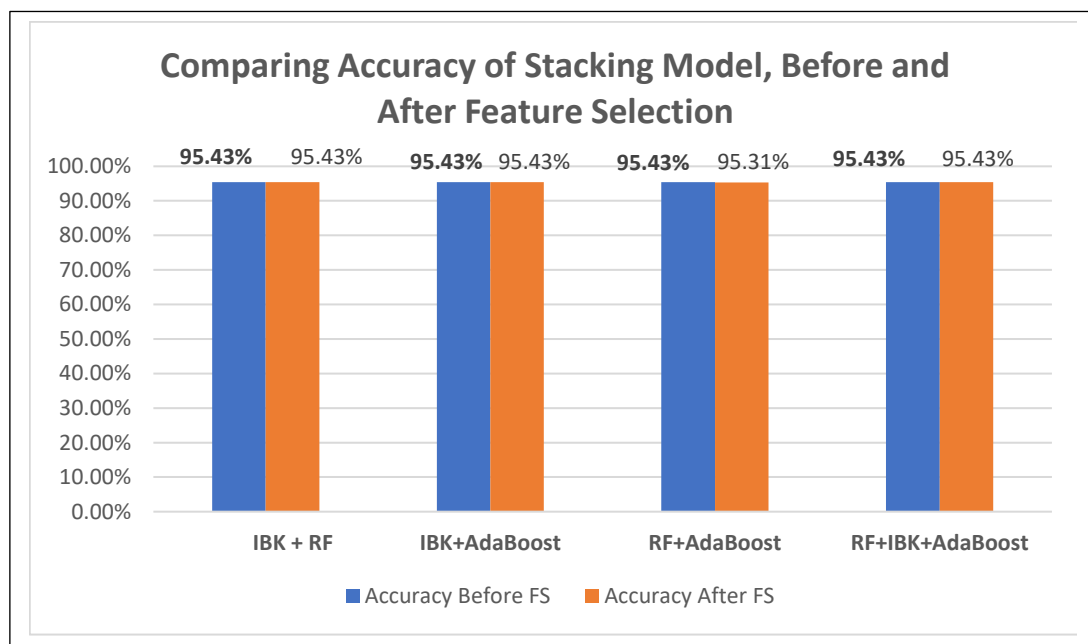


Fig. 5.1: Accuracy Before and After Feature Selection

RQ3. Can balancing techniques improve the accuracy?

To check the effect of the balancing techniques on the improvement of accuracy. We have used five versions of oversampling and undersampling techniques then we compare the accuracy. The balancing techniques used in this research are four

versions of oversampling (SMOTE, ADASYN, SMOTETomek, and random oversampling) and one version of undersampling (SMOTEENN). According to Fig. 5.2, the accuracy has significantly improved by using different versions of balancing techniques. The lowest accuracy was scored using SMOTE while the highest was scored using SMOTEEN technique. By answering this question, we have achieved the research objective **O5**.

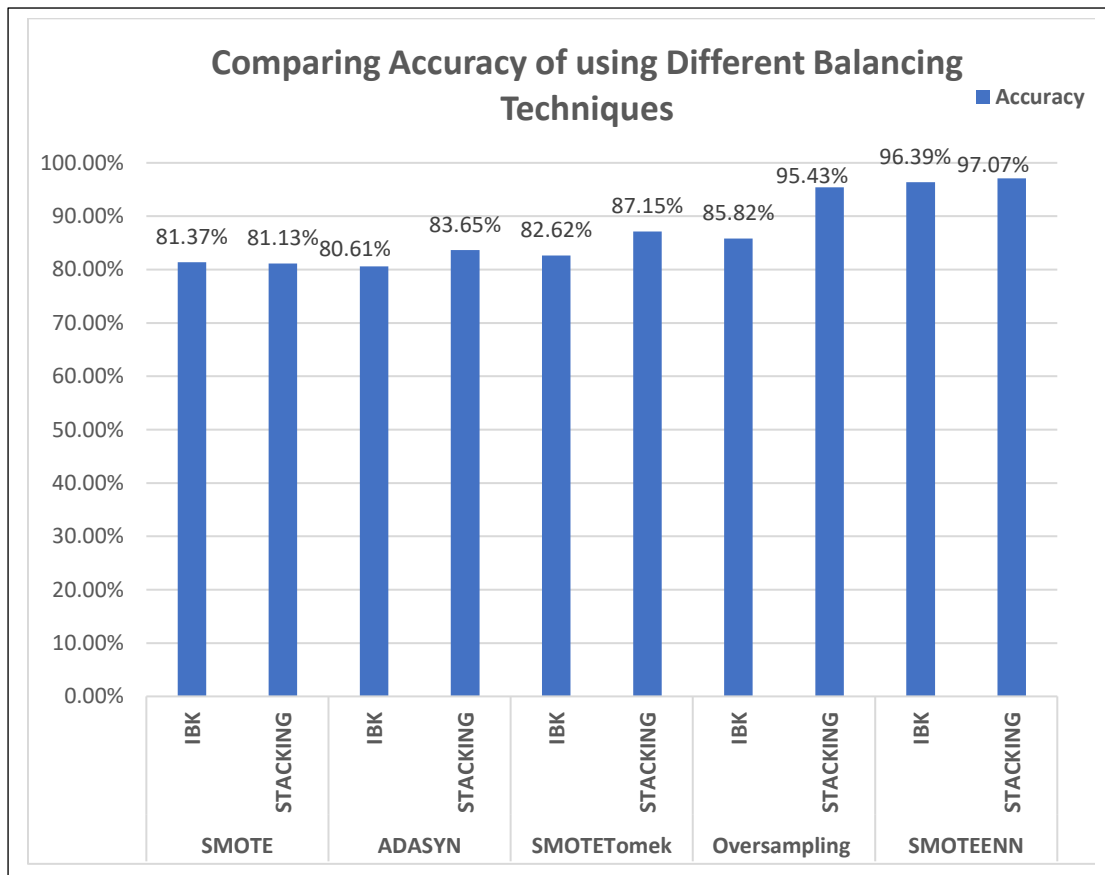


Fig. 5.2: Accuracy using Different Balancing Techniques

RQ4. How does ensemble learning perform in detecting liver disease?

According to Table 4.4 and Table 4.5, the results showed that ML models particularly ensemble stacking models scored 100% by recall measurement in detecting positive cases for patients with liver disease. Therefore, ML models have the ability to detect patients with liver disease using stacking models. Meanwhile, in general, voting

techniques did not improve the performance of the individual models compared to the ability and strength of stacking models. This means the ensemble learning especially stacking technique can increase with high rate the performance of weak classifiers using different versions of resampling techniques. Moreover, the three individual models (IBK, RF, and adaBoost) scored high results in detecting negative cases for patients without liver disease. Also, IBK achieved the highest result among them with 99.6% in terms of recall measurement using SOMTEEN technique. By answering this question, we have achieved research objective **O6**.

RQ5. How do clustering algorithms perform in detecting liver disease?

Referring to the results plotted in Fig. 4.3 – 4.6, the K-Means and DBSCAN clustering models have been scored bad performance in detecting liver disease. There are a large number of misclassified cases between two classes (positive and negative cases) in each cluster. Which indicates that both clusters are unable to detect liver disease. Meanwhile, the clustering EM model scored interesting results in detecting negative cases which outperform both clustering models used in this study. By answering this question, we have achieved research objective **O7**.

5.4 Conclusion and Recommendations

In conclusion, this research aims to detect liver disease using ML. In addition, the study examines the effect of using feature selection and different balancing preprocessing techniques on improving accuracy.

Also, different ML classification and clustering models have been applied to ILPD. The classification models are IBK, RF, and adaBoost with various versions of voting and stacking while the clustering models are K-Means, DBSCAN, GMM, and EM. Moreover, the balancing techniques used in this work are oversampling (SMOTE, ADASYN, SMOTETomek, random oversampling), and undersampling (SMOTEENN). The research found that ML has the potential in detecting liver disease. Also, this study has shown stacking models could detect positive cases by 100% using random oversampling technique. Another interesting finding is the same models could increase the accuracy rate by 6% to 9.6% using oversampling technique. However, voting models did not have a significant effect in increasing the performance of weak classifiers on ILPD. Another finding is that the IBK model could detect negative cases for patients without liver disease by 99.6% using SMOTEENN technique. In addition, considering the drawback of a high decrease in the number of positive cases by 58.41% in SMOTEENN undersampling technique. Furthermore, this research revealed the impact of improving the accuracy by using different balancing preprocessing techniques. The lowest accuracy was by SMOTE while the highest accuracy was by SMOTEENN technique. One more finding is feature selection could improve the performance of the proposed models. The TP attribute was selected as irrelevant in the ILPD which will reduce the cost of medical tests. On the other hand, for clustering models, this study found that GMM and EM models achieved the best performance among them in detecting negative cases. Those algorithms might compete with individual classification models or stacking models for classifying negative cases. Finally, in this research, the proposed model for detecting liver disease is the stacking model using random oversampling technique.

This research could help physicians in detecting liver disease for positive cases with minimal medical tests. That could save patients' lives and reduce the cost. Also, the findings could contribute to improving the services by decision makers in health care systems. For future work, we recommend applying ML in a big dataset for liver disease since the selected dataset is considered a small dataset. Also, focusing on using stacking technique with different parameters and more than two levels. Another direction in the clustering algorithm is applying EM model with various parameters in different liver disease datasets.

5.5 Summary

This chapter compared the finding results with the results of the previous work. Also, it answered the research questions of the current study. Finally, the chapter showed the conclusion and recommendations.

References

- Abdar, M., Zomorodi-Moghadam, M., Das, R., & Ting, I. H. (2017). Performance Analysis of Classification Algorithms on Early Detection of Liver Disease. *Expert Systems with Applications*, 67, 239–251. <https://doi.org/10.1016/j.eswa.2016.08.065>
- Adil, S. H., Ebrahim, M., Raza, K., Saad, S., Ali, A., & Hashmani, M. A. (2018). Liver Patient Classification using Logistic Regression.
- Al Telaq, B. H., & Hewahi, N. (2021). Prediction of Liver Disease using Machine Learning Models with PCA. *2021 International Conference on Data Analytics for Business and Industry, ICDABI 2021*, 250–254. <https://doi.org/10.1109/ICDABI53623.2021.9655897>
- Ambesange, S., Nadagoudar, R., Uppin, R., Patil, V., Patil, S., & Patil, S. (2020). Liver Diseases Prediction using KNN with Hyper Parameter Tuning Techniques. *Proceedings of B-HTC 2020 - 1st IEEE Bangalore Humanitarian Technology Conference*, 1–6. <https://doi.org/10.1109/B-HTC50970.2020.9297949>
- Ambesange, S., Vijayalaxmi, A., Uppin, R., Patil, S., & Patil, V. (2020). Optimizing Liver Disease Prediction with Random Forest by Various Data Balancing Techniques. *Proceedings - 2020 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM 2020*, 98–102. <https://doi.org/10.1109/CCEM50674.2020.00030>
- Antony, L., Azam, S., Ignatious, E., Quadir, R., Beeravolu, A. R., Jonkman, M., & De Boer, F. (2021). A Comprehensive Unsupervised Framework for Chronic Kidney Disease Prediction. *IEEE Access*, 9, 126481–126501. <https://doi.org/10.1109/ACCESS.2021.3109168>
- Armya, R. E. A., Abdulazeez, A. M., Sallow, A. B., & Zeebaree, D. Q. (2021). Leukemia Diagnosis using Machine Learning Classifiers

Based on Correlation Attribute Eval Feature Selection. *Asian Journal of Research in Computer Science*, July, 52–65.

<https://doi.org/10.9734/ajrcos/2021/v9i330225>

Attali, M., Barel, Y., Somin, M., Beilinson, N., Shankman, M., Ackerman, A., & Malnick, S. D. H. (2006). A Cost-Effective Method for Reducing the Volume of Laboratory Tests in a University-Associated Teaching Hospital. *The Mount Sinai Journal of Medicine*, New York, 73(5), 787–794.

<https://europepmc.org/article/med/17008940>

B. R. Manju and A. R. Nair, "Classification of Cardiac Arrhythmia of 12 Lead ECG Using Combination of SMOTEENN, XGBoost and Machine Learning Algorithms," 2019 9th International Symposium on Embedded Computing and System Design (ISED), 2019, pp. 1-7, doi: 10.1109/ISED48680.2019.9096244.

Balakrishnan, M. (2020). Public Health Perspectives on Combating Liver Disease. *Clinical Liver Disease*, 16(5), 186.

<https://doi.org/10.1002/cld.1049>

Banu Priya, M., Laura Juliet, P., & Tamilselvi, P. R. (2018). Performance Analysis of Liver Disease Prediction Using Machine Learning Algorithms. *International Research Journal of Engineering and Technology(IRJET)*, 5(1), 206–211.

Bayrak, E. A., Kirci, P., & Ensari, T. (2019). Performance Analysis of Machine Learning Algorithms and Feature Selection Methods on Hepatitis Disease. *International Journal of Multidisciplinary Studies and Innovative Technologies*, 3(2), 135–138.

Bereta, M. (2019). Regularization of Boosted Decision Stumps using Tabu Search. *Applied Soft Computing Journal*, 79, 424–438.

<https://doi.org/10.1016/j.asoc.2019.04.003>

Cainelli, F. (2012). Liver Diseases in Developing Countries. *World*

- Journal of Hepatology*, 4(3), 66–67.
<https://doi.org/10.4254/wjh.v4.i3.66>
- Chung, W. (2015). The Cost of Liver Disease in Korea: Methodology, Data, and Evidence. *Clinical and Molecular Hepatology*, 21(1), 14–21. <https://doi.org/10.3350/cmh.2015.21.1.14>
- Eibe, F., Mark A. Hall, and Ian H. Witten (2016). The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, Fourth Edition, 2016.
- Fitriyani, N. L., Syafrudin, M., Alfian, G., Fatwanto, A., Qolbiyani, S. L., & Rhee, J. (2020). Prediction Model for Type 2 Diabetes using Stacked Ensemble Classifiers. *2020 International Conference on Decision Aid Sciences and Application, DASA 2020*, 399–402. <https://doi.org/10.1109/DASA51403.2020.9317090>
- Gan, D., Shen, J., An, B., Xu, M., & Liu, N. (2020). Integrating TANBN with Cost Sensitive Classification Algorithm for Imbalanced Data in Medical Diagnosis. *Computers and Industrial Engineering*, 140(December 2019), 106266. <https://doi.org/10.1016/j.cie.2019.106266>
- Geetha, C., & Arunachalam, A. R. (2021). Evaluation Based Approaches for Liver Disease Prediction using Machine Learning Algorithms. *2021 International Conference on Computer Communication and Informatics, ICCCI 2021*, 55–58. <https://doi.org/10.1109/ICCCI50826.2021.9402463>
- Google Colaboratory. (n.d.). Colab.research.google.com. https://colab.research.google.com/?utm_source=scs-index
- Gupta, S., Karanth, G., Pentapati, N., & Prasad, V. R. B. (2020). A Web Based Framework for Liver Disease Diagnosis using Combined Machine Learning Models. *Proceedings - International Conference*

- on *Smart Electronics and Communication, ICOSEC 2020, Icosec*, 421–428. <https://doi.org/10.1109/ICOSEC49089.2020.9215454>
- Haibo He, Yang Bai, E. A. Garcia and Shutao Li, "ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning," *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 2008, pp. 1322-1328, doi: 10.1109/IJCNN.2008.4633969.
- Haque, M. R., Islam, M. M., Iqbal, H., Reza, M. S., & Hasan, M. K. (2018). Performance Evaluation of Random Forests and Artificial Neural Networks for the Classification of Liver Disorder. *International Conference on Computer, Communication, Chemical, Material and Electronic Engineering, IC4ME2 2018*. <https://doi.org/10.1109/IC4ME2.2018.8465658>
- Hartatik, H., Tamam, M. B., & Setyanto, A. (2020). Prediction for Diagnosing Liver Disease in Patients using KNN and Naïve Bayes Algorithms. *2020 2nd International Conference on Cybernetics and Intelligent System, ICORIS 2020*, 0–4. <https://doi.org/10.1109/ICORIS50180.2020.9320797>
- Hassoon, M., Kouhi, M. S., Zomorodi-Moghadam, M., & Abdar, M. (2017). Rule Optimization of Boosted C5.0 Classification Using Genetic Algorithm for Liver disease Prediction. *2017 International Conference on Computer and Applications, ICCA 2017*, 299–305. <https://doi.org/10.1109/COMAPP.2017.8079783>
- Jacob, J., Chakkalakal Mathew, J., Mathew, J., & Issac, E. (2018). Diagnosis of Liver Disease Using Machine Learning Techniques. *International Research Journal of Engineering and Technology*, 5(4), 4011–4014.
- Johns Hopkins Medicine. (2019). Conditions and Diseases. <https://www.hopkinsmedicine.org/health/conditions-and-diseases>

- Kalagotla, S. K., Gangashetty, S. V., & Giridhar, K. (2021). A Novel Stacking Technique for Prediction of Diabetes. *Computers in Biology and Medicine*, 135(June), 104554.
<https://doi.org/10.1016/j.compbiomed.2021.104554>
- Khan, B., Naseem, R., Ali, M., Arshad, M., & Jan, N. (2019). Machine Learning Approaches for Liver Disease Diagnosing. *International Journal of Data Science and Advanced Analytics (ISSN 2563-4429)*, 1(1), 27–31.
- Krishnani, D., Kumari, A., Dewangan, A., Singh, A., & Naik, N. S. (2019). Prediction of Coronary Heart Disease using Supervised Machine Learning Algorithms. *IEEE Region 10 Annual International Conference, Proceedings/TENCON, 2019-Octob*, 367–372.
<https://doi.org/10.1109/TENCON.2019.8929434>
- Kumar, P., & Thakur, R. S. (2019). Diagnosis of Liver Disorder Using Fuzzy Adaptive and Neighbor Weighted K-NN Method for LFT Imbalanced Data. *6th IEEE International Conference on "Smart Structures and Systems", ICSSS 2019*, 17–21.
<https://doi.org/10.1109/ICSSS.2019.8882861>
- Kumar, S., & Katyal, S. (2018). Effective Analysis and Diagnosis of Liver Disorder by Data Mining. *Proceedings of the International Conference on Inventive Research in Computing Applications, ICIRCA 2018*, 1047–1051.
<https://doi.org/10.1109/ICIRCA.2018.8596817>
- Kumari, S., Kumar, D., & Mittal, M. (2021). An Ensemble Approach for Classification and Prediction of Diabetes Mellitus using Soft Voting Classifier. *International Journal of Cognitive Computing in Engineering*, 2(November 2020), 40–46.
<https://doi.org/10.1016/j.ijcce.2021.01.001>

- Kuzhippallil, M. A., Joseph, C., & Kannan, A. (2020). Comparative Analysis of Machine Learning Techniques for Indian Liver Disease Patients. *2020 6th International Conference on Advanced Computing and Communication Systems, ICACCS 2020*, 778–782. <https://doi.org/10.1109/ICACCS48705.2020.9074368>
- Li, X., Chen, X., & Yuan, Z. (2021). Applicable Model of Liver Disease Detection Based on the Improved CART-AdaBoost Algorithm. *2021 IEEE International Conference on Artificial Intelligence and Computer Applications, ICAICA 2021*, 1177–1181. <https://doi.org/10.1109/ICAICA52286.2021.9498046>
- Liu, S., Reviriego, P., Montuschi, P., & Lombardi, F. (2021). Less-is-Better Protection (LBP) for Memory Errors in KNNs classifiers. *Future Generation Computer Systems*, 117, 401–411. <https://doi.org/10.1016/j.future.2020.12.015>
- Mokdad, A. A., Lopez, A. D., Shahrzaz, S., Lozano, R., Mokdad, A. H., Stanaway, J., Murray, C. J. L., & Naghavi, M. (2014). Liver Cirrhosis Mortality in 187 Countries Between 1980 and 2010: A systematic analysis. *BMC Medicine*, 12(1), 1–24. <https://doi.org/10.1186/s12916-014-0145-y>
- Moon, A. M., Singal, A. G., & Tapper, E. B. (2019). Contemporary Epidemiology of Chronic Liver Disease and Cirrhosis. *Clinical Gastroenterology and Hepatology*, 18(12). <https://doi.org/10.1016/j.cgh.2019.07.060>
- Muhuri, S., Sarkar, A., Chakraborty, S., & Chakraborty, S. (2019). A Statistical Method for Prediction of Liver Disease Based on the Brownian Motion Model. *Proceedings of 2019 IEEE Region 10 Symposium, TENSYP 2019*, 7, 157–161. <https://doi.org/10.1109/TENSYP46218.2019.8971234>
- Nahar, N., Ara, F., Nelay, M. A. I., Barua, V., Hossain, M. S., &

- Andersson, K. (2019). A Comparative Analysis of the Ensemble Method for Liver Disease Prediction. *ICIET 2019 - 2nd International Conference on Innovation in Engineering and Technology*, 23–24. <https://doi.org/10.1109/ICIET48527.2019.9290507>
- Ndugga, N., Lightbourne, T. G., Javaherian, K., Cabezas, J., Verma, N., Barritt, A. S., & Bataller, R. (2017). Disparities Between Research Attention and Burden in Liver Diseases: Implications on uneven advances in pharmacological therapies in Europe and the USA. *BMJ Open*, 7(3). <https://doi.org/10.1136/bmjopen-2016-013620>
- Poddar, M. G., Birajdar, A. C., Virmani, J., & Kriti. (2019). Automated Classification of Hypertension and Coronary Artery Disease Patients by PNN, KNN, and SVM Classifiers Using HRV Analysis. In *Machine Learning in Bio-Signal Analysis and Diagnostic Imaging*. Elsevier Inc. <https://doi.org/10.1016/b978-0-12-816086-2.00005-9>
- Rabbi, M. F., Hassan, S. ., Arifa, Asif, & Kamrul. (2020). Prediction of Liver Disorders using Machine Learning Algorithms: A Comparative Study. 111–116.
- Raheem, I. O., Babatunde, A. O., ARO, T. O., & MUSA, A. Y. (2017). Effect of Ensembles on Selected Data Mining Techniques in Classification of Liver Diseases.
- Ramesh, D., & Katheria, Y. S. (2019). Ensemble Method Based Predictive Model for Analyzing Disease Datasets: A Predictive Analysis Approach. *Health and Technology*, 9(4), 533–545. <https://doi.org/10.1007/s12553-019-00299-3>
- Sbirna, S., & Sbirna, L. S. (2020). Angiographic Prognosis and Diagnosis of Heart Disease by using Unsupervised and Supervised Machine Learning Techniques. *2020 24th International Conference on System Theory, Control and Computing, ICSTCC 2020 - Proceedings*, 10,

84–91. <https://doi.org/10.1109/ICSTCC50638.2020.9259719>

- Sepanlou, S. G., Safiri, S., Bisignano, C., Ikuta, K. S., Merat, S., Saberifiroozi, M., Poustchi, H., Tsoi, D., Colombara, D. V., Abdoli, A., Adedoyin, R. A., Afarideh, M., Agrawal, S., Ahmad, S., Ahmadian, E., Ahmadpour, E., Akinyemiju, T., Akunna, C. J., Alipour, V., ... Malekzadeh, R. (2020). The global, Regional, and National Burden of Cirrhosis by Cause in 195 Countries and Territories, 1990–2017: A Systematic Analysis for the Global Burden of Disease Study 2017. *The Lancet Gastroenterology and Hepatology*, 5(3), 245–266. [https://doi.org/10.1016/S2468-1253\(19\)30349-8](https://doi.org/10.1016/S2468-1253(19)30349-8)
- Simon, M. A. M., Saradhi, N. K., P.Sahithi, Kiran, R. S., & A. Aparna, M. T. (2021). Liver Disease Prediction Using Machine Learning. XIV(Vi), 589–596. https://doi.org/10.1007/978-981-16-0171-2_56
- Singh, J., Bagga, S., & Kaur, R. (2020). Software-based Prediction of Liver Disease with Feature Selection and Classification Techniques. *Procedia Computer Science*, 167(2019), 1970–1980. <https://doi.org/10.1016/j.procs.2020.03.226>
- Singh, V., Gourisaria, M. K., & Das, H. (2021). Performance Analysis of Machine Learning Algorithms for Prediction of Liver Disease. 2021 *IEEE 4th International Conference on Computing, Power and Communication Technologies, GUCON 2021*, 1–7. <https://doi.org/10.1109/GUCON50781.2021.9573803>
- Sreejith, S., Khanna Nehemiah, H., & Kannan, A. (2020). Clinical Data Classification using an Enhanced SMOTE and Chaotic Evolutionary Feature Selection. *Computers in Biology and Medicine*, 126(September), 103991. <https://doi.org/10.1016/j.compbimed.2020.103991>
- Thirunavukkarasu, K., Singh, A. S., Irfan, M., & Chowdhury, A. (2018).

Prediction of Liver Disease using Classification Algorithms. *2018 4th International Conference on Computing Communication and Automation, ICCCA 2018*, 6(9), 18–20.

<https://doi.org/10.1109/CCAA.2018.8777655>

UCI Machine Learning Repository. (n.d.). Retrieved July 15, 2021, from [https://archive.ics.uci.edu/ml/datasets/ILPD+\(Indian+Liver+Patient+Dataset\)](https://archive.ics.uci.edu/ml/datasets/ILPD+(Indian+Liver+Patient+Dataset))

Vats, V., Zhang, L., Chatterjee, S., Ahmed, S., Enziama, E., & Tepe, K. (2019). A Comparative Analysis of Unsupervised Machine Techniques for Liver Disease Prediction. *2018 IEEE International Symposium on Signal Processing and Information Technology, ISSPIT 2018*, 486–489.

<https://doi.org/10.1109/ISSPIT.2018.8642735>

Weka. (n.d.). Class Ranker. Retrieved February 1, 2022, from <https://weka.sourceforge.io/doc.dev/weka/attributeSelection/Ranker.html>

World Health Organization. (2017, April 19). Global Hepatitis Report. <https://www.who.int/publications/i/item/global-hepatitis-report-2017>

Zeng, F., & Wang, Y. (2016). Dynamic Model with Time Varying Delay for Type 1 Diabetes Mellitus Identified by using Expectation Maximization Algorithm. *Chinese Control Conference, CCC, 2016-Augus*, 9376–9381. <https://doi.org/10.1109/ChiCC.2016.7554848>

Appendices

The results using different balancing techniques before implementing feature selection on ILPD.

Table (1): Results using SMOTE Technique Before Applying Feature Selection

Algorithms		Acc.	Recall		Precision		ROC	RMSE
			0	1	0	1		
MODELS	IBK	79.69%	0.93	0.67	0.74	0.90	0.90	0.45
	RF	81.97%	0.89	0.75	0.78	0.87	0.90	0.36
	adaBoost	82.81%	0.88	0.77	0.80	0.87	0.89	0.41
STACKING	IBK+RF	83.89%	0.84	0.84	0.84	0.84	0.84	0.37
	IBK+adaBoost	82.45%	0.82	0.83	0.83	0.82	0.84	0.38
	RF+adaBoost	84.50%	0.82	0.87	0.86	0.83	0.86	0.36
	RF+IBK+adaBoost	82.45%	0.82	0.83	0.83	0.82	0.84	0.38
VOTING	IBK+RF	79.81%	0.93	0.67	0.74	0.90	0.80	0.45
	IBK+adaBoost	82.81%	0.89	0.77	0.79	0.87	0.83	0.41
	RF+adaBoost	82.81%	0.88	0.77	0.80	0.87	0.83	0.41
	RF+IBK+adaBoost	83.65%	0.90	0.77	0.80	0.89	0.84	0.40

Table (2): Results using ADASYN Technique Before Applying
Feature Selection

Algorithms		Acc.	Recall		Precision		ROC	RMSE
			0	1	0	1		
MODELS	IBK	79.34%	0.92	0.68	0.72	0.91	0.91	0.45
	RF	80.10%	0.86	0.75	0.75	0.86	0.90	0.37
	adaBoost	82.76%	0.87	0.79	0.79	0.87	0.88	0.41
STACKING	IBK+RF	84.41%	0.81	0.88	0.85	0.84	0.84	0.36
	IBK+adaBoost	84.41%	0.81	0.88	0.85	0.84	0.84	0.36
	RF+adaBoost	82.76%	0.86	0.80	0.79	0.86	0.83	0.38
	RF+IBK+ adaBoost	84.41%	0.81	0.88	0.85	0.84	0.84	0.36
VOTING	IBK+RF	79.47%	0.92	0.68	0.72	0.91	0.80	0.45
	IBK+adaBoost	82.76%	0.88	0.78	0.78	0.88	0.83	0.42
	RF+adaBoost	82.76%	0.87	0.79	0.79	0.87	0.83	0.42
	RF+IBK+ adaBoost	82.38%	0.89	0.77	0.78	0.88	0.83	0.42

Table (3): Results using SMOTETomek Technique Before
Applying Feature Selection

Algorithms		Acc.	Recall		Precision		ROC	RMSE
			0	1	0	1		
MODELS	IBK	83.38%	0.97	0.70	0.76	0.96	0.95	0.41
	RF	82.24%	0.89	0.75	0.78	0.87	0.92	0.35
	adaBoost	85.39%	0.89	0.82	0.83	0.88	0.91	0.38
STACKING	IBK+RF	89.29%	0.88	0.90	0.90	0.89	0.88	0.31
	IBK+adaBoost	89.29%	0.88	0.90	0.90	0.89	0.88	0.31
	RF+adaBoost	84.63%	0.80	0.89	0.88	0.82	0.85	0.36
	RF+IBK+ adaBoost	89.29%	0.88	0.90	0.90	0.89	0.88	0.31
VOTING	IBK+RF	83.25%	0.97	0.70	0.76	0.96	0.83	0.41
	IBK+adaBoost	86.15%	0.90	0.82	0.83	0.90	0.86	0.37
	RF+adaBoost	85.39%	0.89	0.82	0.83	0.88	0.85	0.38
	RF+IBK+ adaBoost	85.14%	0.92	0.78	0.81	0.91	0.85	0.39

Table (4): Results using Random Oversampling Technique
Before Applying Feature Selection

Algorithms		Acc.	Recall		Precision		ROC	RMSE
			0	1	0	1		
MODELS	IBK	83.41%	0.96	0.71	0.77	0.94	0.96	0.41
	RF	86.66%	0.96	0.77	0.81	0.95	0.98	0.29
	adaBoost	89.42%	0.95	0.84	0.86	0.94	0.95	0.32
STACKING	IBK+RF	95.43%	0.91	1	1	0.92	0.94	0.21
	IBK+adaBoost	95.43%	0.91	1	1	0.92	0.94	0.21
	RF+adaBoost	95.43%	0.91	1	1	0.92	0.94	0.21
	RF+IBK+ adaBoost	95.43%	0.91	1	1	0.92	0.94	0.21
VOTING	IBK+RF	83.41%	0.96	0.71	0.77	0.94	0.83	0.41
	IBK+adaBoost	89.66%	0.95	0.84	0.86	0.95	0.90	0.32
	RF+adaBoost	89.42%	0.95	0.84	0.86	0.94	0.89	0.33
	RF+IBK+ adaBoost	88.70%	0.95	0.82	0.84	0.95	0.89	0.34

Table (5): Results using SMOTEENN Technique Before
Applying Feature Selection

Algorithms		Acc.	Recall		Precision		ROC	RMSE
			0	1	0	1		
MODELS	IBK	96.16%	0.99	0.91	0.94	0.99	0.99	0.20
	RF	93.00%	0.98	0.86	0.91	0.96	0.98	0.25
	adaBoost	94.36%	0.99	0.87	0.92	0.98	0.96	0.24
STACKING	IBK+RF	97.52%	0.98	0.97	0.98	0.97	0.97	0.16
	IBK+adaBoost	97.52%	0.98	0.97	0.98	0.97	0.97	0.16
	RF+adaBoost	94.13%	0.98	0.88	0.93	0.97	0.92	0.23
	RF+IBK+ adaBoost	97.52%	0.98	0.97	0.98	0.97	0.97	0.16
VOTING	IBK+RF	96.16%	0.996	0.91	0.94	0.99	0.95	0.20
	IBK+adaBoost	94.58%	0.99	0.87	0.92	0.99	0.93	0.23
	RF+adaBoost	94.36%	0.99	0.87	0.92	0.98	0.93	0.24
	RF+IBK+ adaBoost	94.36%	0.99	0.87	0.92	0.98	0.93	0.24

الملخص

يعتبر مرض الكبد أزمة صحية عالمية مع ارتفاع معدل الوفيات كل عام. مما يؤثر على حياة المريض بالإضافة إلى ارتفاع تكلفة العلاج في معظم الدول. يمكن أن تساعد القطاعات العلمية المختلفة في المجال الصحي للحد من آثار بعض الأمراض. لذلك ، يمكن استخدام تعلم الآلة في نظام الرعاية الصحية في تشخيص الأمراض والمساعدة في اتخاذ القرار. سيؤدي ذلك إلى الحفاظ على حياة المريض وتحسين الخدمات وتقليل التكلفة. تم إجراء هذا البحث لمساعدة مجال الرعاية الصحية من خلال استكشاف وتشخيص أمراض الكبد من خلال إمكانات تعلم الآلة. الهدف الرئيسي من هذه الدراسة هو استخدام خوارزميات تعلم الآلة في الكشف عن أمراض الكبد. مجموعة البيانات المستخدمة في هذا البحث هي مجموعة بيانات مرضى الكبد الهندي (ILPD) والتي تم الحصول عليها من موقع بيانات إنترنت عام. نوعان من خوارزميات التعلم الآلي المنفذة في هذا العمل التعلم الخاضع للإشراف والتعلم غير الخاضع للإشراف. النماذج الخاضعة للإشراف هي (IBK و Random Forest و Adaptive Boosting) مع تعلم المجموعة التصويت والتكديس (voting و stacking). بينما النماذج غير الخاضعة للإشراف هي (K-Means و DBSCAN و Gaussian Mixture Model و Expectation Maximization). تم استخدام معايير تقييم مختلفة لمقارنة النماذج في نماذج التعلم تحت الإشراف وغير الخاضعة للإشراف. كانت الدقة والتذكر والدقة و ROC و RMSE لنماذج التصنيف بينما كان قياسان لنماذج التجميع هما معامل صورة ظليلة ومؤشر راند المعدل. علاوة على ذلك ، طبق هذا العمل تقنيات Ranker لاختيار الميزة. CorrelationAttributeEval و InfoGainAttributeEval مع طريقة البحث Ranker لاختيار الميزة. علاوة على ذلك ، طبق هذا العمل خمسة أنواع من تقنيات الموازنة. كانت تقنيات الموازنة هي (SMOTE و ADASYN و SMOTETomek و Random Oversampling و SMOTEENN). بالنسبة لنماذج التعلم تحت الإشراف ، أوضحت النتائج المتحصل عليها في هذه الدراسة أن تقنية التكديس تفوقت على النماذج الأخرى في الكشف عن أمراض الكبد للحالات الإيجابية والتي سجلت 100٪ باستخدام تقنية العشوائية الزائدة (Random Oversampling). بالإضافة إلى نموذج IBK حصل على 99.6٪ في الكشف عن الحالات السلبية باستخدام تقنية (SMOTEENN). علاوة على ذلك ، بالنسبة للتعلم غير الخاضع للإشراف ، فإن نموذج (Expectation Maximization و Gaussian Mixture Model) لديهما أداء جيد في اكتشاف الحالات السلبية. بالإضافة إلى ذلك ، وجد هذا البحث أن تقنيات الموازنة لها تأثير إيجابي على زيادة الأداء. كما أن لتقنية اختيار الميزة تأثير جيد على تقليل الفحوصات الطبية. ومع ذلك ، أظهرت النتائج أن أسلوب التصويت (voting) له تأثير محدود على تحسين الدقة. تم استخدام خوارزميات ML المختلفة في الأعمال السابقة. كانت LR و NB و SVM و IBK و RF و ANN و DT والتعزيز والتعبئة والتكديس والتصويت و K-Means و DBSCAN. اقترحت هذه الدراسة نماذج التراص باستخدام تقنية عشوائية مفرطة في الكشف عن أمراض الكبد والتي يمكن أن تساعد الأطباء وأنظمة الرعاية الصحية مع ارتفاع معدل الكشف عن الحالات الإيجابية. سيؤدي ذلك إلى إنقاذ حياة المرضى وتقليل التكلفة وتحسين الخدمات في أنظمة الرعاية الصحية. أيضًا ، يمكن أن تزيد هذه التقنيات من أداء المصنفات الضعيفة (IBK و RF و adaBoost). علاوة على ذلك ، تفوقت النماذج المقترحة على الأعمال السابقة المذكورة في هذا البحث في الكشف عن الحالات الإيجابية لأمراض الكبد.

جامعة البحرين
كلية العلوم
قسم برامج الدراسات العليا



الكشف عن أمراض الكبد باستخدام خوارزميات التعلم الآلي

إطروحة مقدمة كجزء من متطلبات الحصول على درجة الماجستير
في علوم وتحليل البيانات الضخمة

إعداد

الطالب: برير حسن آل طلاق

الرقم الجامعي: 202012510

إشراف

د. عبدالله القدومي
أستاذ مساعد

جامعة البحرين

د. مصطفى حماد
أستاذ مشارك

جامعة البحرين

مملكة البحرين
يونيو 2022