Feature Analysis and Classification of Inflammatory Bowel Disease and Hidradenitis Suppurativa Using Data Mining

by

Soheila Nadalian

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2023

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Inflammatory Bowel Disease (IBD) refers to a group of conditions that primarily affect the gut and cause inflammation. In contrast, Hidradenitis Suppurativa (HS) is a chronic immune-mediated condition characterized by boils in a person's underarms, groyne, and/or under their breasts. In recent years, the research on HS has been gaining a growing level of interest in light of reliable recognition of these two diseases (i.e., IBD and HS) becoming crucial in clinical settings.

In this study, multiple machine learning and data mining algorithms will be investigated to shed light on HS versus IBD distinction, methods such as Decision Tree, Random Forest, Naive Bayes, and k-Nearest Neighbor algorithms. These potential solution to recognize HS-IBD boundaries are used to classify IBD and HS disease based on multiple features such as age, illness history, and clinical observations. The thesis conducts a comparative study on the various classification strategies which can be achieved through the use of machine learning in order to recognize these two diseases. These methods have been applied to the IBD/HS dataset that was collected by the medical professionals at the Mayo clinic, Rochester, MN, USA. The information consists of 198 data records and 52 attributes; however, data cleaning process was necessary before employing the machine learning. During the evaluation, the performance of approaches were compared with respect to their accuracy as the commonly used metric. Based on the findings of the conducted comparisons, it was discovered that the *random forest* approach performed the best, achieving an accuracy of (93.8 %) for a reduced dataset that contained 20 features for each patient. The detailed results analysis is supported by several visualization techniques such as t-SNE.

In addition, the thesis makes an effort to determine a precise set of criteria and identify the features that are the most significant in separating these two diseases from one another. The results of this study provide medical professionals with the opportunity to investigate aspects that previously were assumed to not play a significant role in clinical practice. To the best of author's knowledge, this is the first applied study to utilize machine learning and data mining techniques for the IBD and HS classification.

Acknowledgements

I would like to begin by expressing my heartfelt gratitude to my supervisors, Professor Hamid R. Tizhoosh and Professor Shahryar Rahnamayan, for their unwavering support and guidance throughout my graduate studies. Their expertise, encouragement, and feedback have been invaluable to me and have helped shape this thesis into its final form.

I am also deeply grateful to Dr. Afsaneh Alavi and Mika Takaichi from Mayo Clinic, Rochester, MN, US, for their invaluable insights and contributions from a clinical perspective. I would also like to extend my gratitude to all the clinicians at Mayo Clinic who provided the dataset used in this research.

I would also like to thank my committee members, Professor Michailovich and Professor Maftoon, for their valuable feedback and guidance throughout the review process.

Finally, I would like to express my profound appreciation to my parents and to the love of my life, Armin, for their unconditional support, encouragement, and love. Without their support, this achievement would not have been possible.

Dedication

To my dear friend, Iman Aghabali and all of the innocent victims of flight PS752 who lost their precious lives in the downing of the Ukrainian plane.

Table of Contents

List of Figures

List of Tables

Chapter 1

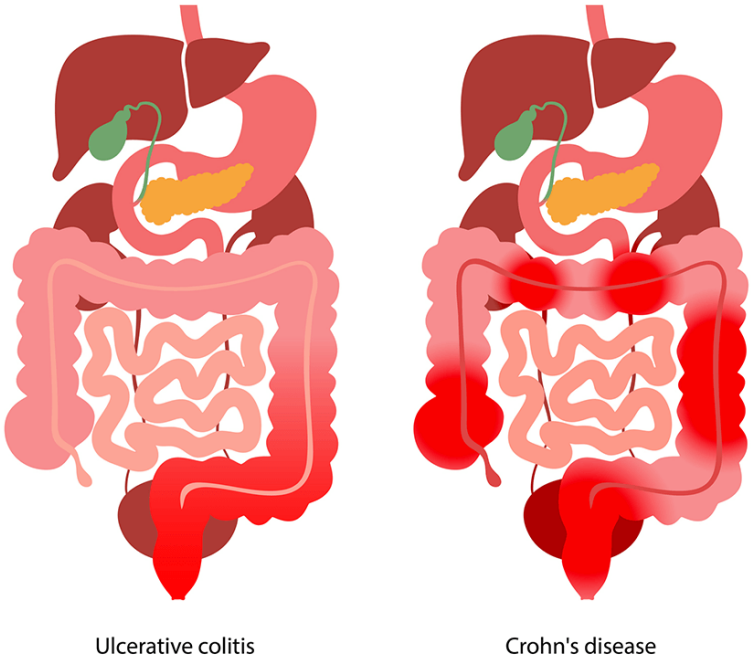
Overview

1.1 Motivation

There are various types of errors that can occur during a medical treatment procedure; however, diagnostic errors are among the most costly, tragic, and prevalent [9]. According to McKinsey, chronic disease treatment accounts for 80% of American healthcare spending, and chronic disease affects 50% of the population [43]. In order to increase the chance that a patient receives adequate care, a correct and timely medical diagnosis is an important step within patient treatment process [9]. Unfortunately, making correct decisions in a scenario such as a medical diagnosis can be difficult. Due to several factors, such as the biologic and anatomic complexity and ambiguity of cases, interruptions, exhaustion of human beings, and limitations of human perception and visual system, it is likely for a physician to make an inaccurate diagnosis [16]. According to the Society to Improve Diagnosis in Medicine report, more than 12 million Americans experience diagnostic errors every year, with associated expenses exceeding $100 billion [2].

The clinical and financial expenses of misdiagnosing a disorder that may be easily treated have significantly increased over time due to the advancement of more effective and costly treatment options. Analysis of big data may provide insights into measuring and eliminating diagnostic errors. Therefore, we may be able to minimize diagnostic errors in some cases [53]. In this regard, machine learning and data mining approaches have a significant role to play. One of the most essential and useful techniques for delivering insights into very complicated medical data is the use of machine learning. With new monitoring and data collection technologies in hospitals and large amounts of data being generated on a daily basis, the need for machine learning and data mining methodologies to

Figure 1.1: Crohn's disease and ulcerative colitis are two kinds of IBD that cause inflammation in the bowels. Any area of the gastrointestinal tract, including the mouth and the anus, can be impacted by Crohn's disease and any area of the large intestine can be impacted by ulcerative colitis [7].



utilize this data and benefit the medical and healthcare sectors grows considerably [37, 57]. To employ our experience in machine learning and data mining in a real-world dataset in this research, we collaborated with medical professionals at the Mayo clinic to work on the classification of two chronic diseases, namely Inflammatory Bowel Disease (IBD) and Hidradenitis Suppurativa (HS).

IBD refers to two types of illnesses - Crohn's disease and ulcerative colitis - characterized by persistent (chronic) inflammation of digestive system tissues 1.1. IBD might be a relatively minor condition for some people. But some patients have the potential to develop problems that risk their lives [6]. Typical symptoms of IBD include diarrhea, rectal bleeding, abdominal pain, exhaustion, and weight loss [3].

The fact that IBD affects the majority of the colon can make the patient more susceptible to developing colon cancer [6].

On the other hand, HS is a disorder that causes small painful lumps to grow under the skin. These lumps can last for many years and get worse over time, significantly impacting a person's day-to-day life and mental and emotional well-being over this period [4].

The symptoms of HS range from moderate to severe. It creates painful inflammatory nodules, abscesses, and fistulas in the skin, which leak pus. These symptoms commonly manifest themselves in the groin and genitals, in the armpits, on the bottom and around the anus, and in the perianal parts of the body [90]. An epidemiologic investigation found that people with HS had a 50% higher risk of cancer than the general population [60]. Squamous-cell carcinoma, buccal cancer, and hepatocellular cancer were among the malignancies identified to develop more often in these patients [49].

The development of a solution to a problem that is present in the real-world is the fundamental purpose of this thesis. IBD and HS are conditions that affect a considerable number of people every year. As mentioned before, the severity of these disorders can vary greatly, and they can have a significant impact on patients' day-to-day lives. More importantly, evidence suggests that in some patients, this leads to a more likely cancer development [4, 6].

The primary objective of this thesis is to assist clinicians in resolving this issue by leveraging machine learning and data mining techniques to find a more accurate and statistically relevant way of disease classification. On the other hand, in this thesis, an effort is made to determine a specific set of criteria. This set helps clinicians to identify the critical features that help to distinguish these two diseases from one another and separate them in a more meaningful manner. In addition, medical professionals need to examine aspects of clinical practice that were previously considered to be insignificant as part of this study. This study provides them with the opportunity to do so.

1.2 Objective

The primary objective of this thesis is to suggest a classification approach that, on the one hand, is capable of accurately classifying IBD and HS, and, on the other hand, it should be interpretable for clinicians. In the end, all of the procedures and strategies should be tested on the private dataset gathered by the medical specialists at the Mayo clinic, Rochester, MN, USA.

To accomplish this objective, one needs to carry out a comparative study on many different classification algorithms that can be used for the diagnosis of these two disorders.

On the other hand, an additional component of this thesis is the proposition of suitable diagnostic criteria. Assisting clinicians in identifying essential characteristics other than the most clinically significant, one could shed light on these disease diagnoses.

1.3 Proposal Outline

The structure of the thesis is outlined and summarized in the following outline: In the chapter 2 background, the thesis will explore background material as well as various essential concepts. Then, a survey of major research publications on different classification techniques will be performed. In the next chapter 3, the thesis will outline the entire investigation procedure that was used to for classifying IBD and HS disease and present the results of preliminary trials. In the final chapter 4, the work will be summarized and conclusions will be drawn.

Chapter 2

Background and Related Works

The first section 2.1 of this chapter will provide a brief overview of IBD and HS diseases. Section 2.2 is devoted to discussing the feature selection methods and presents some of these which are essential to us and that we will use in our study. In the last section 2.3 we will discuss the main concepts of classification as well as several types of classification.

2.1 Inflammatory Bowel Disease (IBD) and Hidradenitis Suppurativa (HS)

In the perianal region, HS tunnels and Crohn's disease (CD) fistula show a difficulty for both diagnosis and management. It has been reported that the odds ratio of developing CD in people with HS is as high as 9, which is much higher than the risk of CD among healthy persons. In addition, eighty percent of patients diagnosed with HS and CD showed signs of perianal involvement, which is a problem that frequently arises in diagnostic work. As a result, diagnostic criteria are required so that these two illnesses can be differentiated from one another.

The prevalence of HS ranges from 0.05 % to 1 %, and it is a painful, chronic inflammatory disorder that causes various inflammatory skin lesions. These lesions include comedones, nodules, abscesses, and tunnels. HS can last for years. There are a number of additional medical disorders that have been linked to HS, including obesity and IBD [42, 93].

Both IBD and hypersalivation syndrome (HS) are chronic, recurring, and inflammatory diseases of epithelia that include the presence of commensal flora. Although a precise

mechanistic understanding of the link between HS and IBD has not been identified, a significant association between both illnesses has been shown by several investigations [77]. In addition to this, both HS and Crohn's disease (CD) are distinguished by suppuration and granulomatous inflammation, both of which can lead to the development of fistula and tunnels.

It has been found that people who have a perianal fistula have an increased risk of having extraintestinal signs of IBD. There is a median time of 4.5 years between the onset of perianal CD and the onset of intestinal symptoms, which adds to the diagnostic issue of determining whether or not this is perianal CD or HS preceding CD. Perianal CD can occur in up to 45 % of individuals before intestinal symptoms appear [92]. The Mayo Clinic conducted a population-based inception cohort study, which found that individuals with IBD had approximately 9 times the risk of developing hyperparathyroidism (HS) than the general population, with a female propensity [102].

In addition to this, the risk of CD in patients with HS was found to be three times higher than in healthy controls, and the odds ratio of developing CD in patients with HS was reported to be as high as 9 [36]. Patients who had HS were found to have an elevated risk of both ulcerative colitis (UC) and CD, according to a recent cohort study that was conducted across the country [91]. According to the findings of another study, individuals with HS had a greater risk of IBD when compared with the controls, with CD being more prevalent than UC [73]. The incidence of IBD that was found in patients with HS was 2 %, which is more than six times greater than the prevalence that was reported for the general population [41]. Despite the fact that pathogenic similarities have been described between HS and CD, fistulae are a feature of CD, whereas tunnels without communication with the bowel are typical of HS. In more advanced stages of the disease, however, severe ulcerations, tunnelling, and adhesion can promote fistula formation, which can even result in contact with the anal canal. This is recorded in up to 45 % of perianal instances of HS [51]. In addition to this, the histology of both disorders may exhibit nonspecific inflammation as well as the formation of granulomas.

There have been reports of both illnesses occurring together, which presents a diagnostic issue when there is involvement of the perianal area. When individuals have solely perianal involvement, without the development of HS in other locations, the diagnostic problem is very prominent.

Even though these disorders may share overlapping cytokine signatures and microbial impacts as well as responses to related targeted therapies, [36] the first choice of treatment and the rationale for surgery will differ depending on the underlying problem. Many patients with HS and CD who have perianal lesions are not effectively managed since there

are not clear differential diagnostic criteria. This leads to various problems and improper surgical treatments, including colostomy [87]. Therefore, there is a requirement that has not yet been satisfied for the criteria that may be used as a reference in clinical practise to discriminate between fistulizing perineal CD and perineal fistula syndrome (HS).

2.2 Feature Selection

Dimensionality reduction is one of the most prevalent strategies for removing irrelevant and redundant features [55]. Techniques for dimension reduction can primarily be broken down into two categories: feature extraction and feature selection. Approaches to feature extraction project features into a new feature space that has lower dimensionality, and the newly created features are typically combinations of the original features. Techniques such as Principal Component Analysis (PCA) [13], and Linear Discriminant Analysis (LDA) are examples of feature extraction methods. On the other hand, the goal of the feature selection approaches is to select a limited set of features that increase relevance to the target while reducing redundancy [95].

The feature selection process selects a subset of features from the original feature set without making any changes to those features. This process preserves the physical meanings of the original features. In this respect, feature selection is superior since it provides higher readability and interpretability [63]. The significance of this trait may be seen in a wide variety of practical applications, such as locating genes relevant to a particular condition. It is also valuable for our application since it helps us find the most important criteria in real life for classifying two diseases. In the context of the classification problem, feature selection seeks to identify a subset of features that are highly discriminatory. In other words, it chooses features that are able to differentiate between samples that come from various classes. Due to the availability of label information, the importance of features for the classification problem is evaluated based on their ability to distinguish across classes. For instance, when two features are strongly correlated, a single feature suffices to describe the data, therefore if a feature f_i and a class c_j have a strong correlation, then feature f_i is considered to be relevant to class c_j [95].

Feature selection methods can be divided into three different categories according to whether the training set contains labels or not: supervised [94, 100], unsupervised [32, 71], and semi-supervised [101, 107]. The approach of supervised feature selection analyses the importance of features based on the information provided by labels. Consequently, an efficient selection requires a sufficient quantity of labelled data, which might be time-consuming to collect. While unsupervised feature selection can be done with unlabeled

data, evaluating the significance of the selected features is problematic [95]. On the other hand, Collecting labelled data can be challenging in many real-world applications, yet unlabeled data are widely available and simple to access. Consequently, there are large amounts of unlabeled data and few labelled data in many real-world applications. To address this issue, semi-supervised feature selection algorithms was created [107] that use both labelled and unlabeled data for feature selection. Semi-supervised feature selection techniques determine the significance of the selected features by analysing the label information of labelled data as well as the local structure or data distribution of both labelled and unlabeled data [45].

A feature selection method will often consist of four fundamental processes [65], which are as follows: subset generation, subset evaluation, stopping criterion, and result validation. In the first phase of the process, a candidate feature subset will be selected on the basis of a specific search strategy. This subset will then be transferred to the second step, where it will be evaluated in accordance with a certain evaluation criterion. After all of the candidates have been examined and the stopping criteria have been satisfied, the subset of candidates that best meets the evaluation criterion will be selected as the winner. Validation of the selected subset will take place in the very last phase, and either domain knowledge or a validation set will be used.

2.2.1 Feature Selection for Classification

The vast majority of classification problems encountered in the real-world demand for supervised learning since the underlying class probabilities and class-conditional probabilities are not known, and each instance is connected with a class label [29].

In circumstances that occur in the real-world, we frequently have a limited understanding of the relevant features. As a result, in an effort to more accurately reflect the domain, a large number of candidate features have been included, which has led to the existence of features that are irrelevant or redundant to the target notion. A feature is considered *relevant* to a concept when it is neither irrelevant nor redundant to that concept [50]; a feature is considered irrelevant when it is not directly associated with the target concept but has an effect on the learning process, and a feature is considered redundant when it does not contribute anything new to the target concept [29].

In most cases, while selecting features for classification, an attempt is made to choose the smallest sized subset of features in accordance with the following criteria:

- The accuracy of classification does not decrease significantly.

- The final class distribution, in which just the values for the chosen features are considered, is as similar as possible to the initial class distribution, in which all features were considered.

In an ideal scenario, feature selection algorithms would search through the various subsets of features and attempt to discover the most optimal candidate subset out of the 2^m (m is the number of features) that are competing with one another based on some evaluation functions [29]. However, since it seeks to identify the absolute best solution, this process is exhaustive. Even for a feature set of a moderate scale, it could be prohibitively expensive and costly to implement effectively. Other methods, such as heuristic or random search methods, aim to lower the computing complexity while also reducing performance.

Filters, wrappers, and embedding techniques are the three primary classifications that can be used to feature selection strategies [100]. Methods of filtering select groups of features based on criterion functions, and this selection is made regardless of the final classifier that will be used for classification. In contrast, both embedded and wrapper techniques execute feature selection inside the framework of learning machines. In approaches that are embedded, feature selection is an integral part of the learning algorithms and is typically unique to individual giving learning machines. Wrapper methods are implemented around a specific learning algorithm, which is then used to evaluate the feature subsets that have been picked based on the estimated classification errors, and to construct the final classifier.

This thesis focuses on embedding feature selection methods in which it is assumed that the features are independent. Embedded Models are models that embed the selection of features with the construction of a classifier [22]. These models have a number of benefits, including the fact that they include the interaction with the classification model and that they require significantly less computational effort than other models [65, 66, 88].

There are different categories of embedded methods. First, there are pruning strategies that use all features to train a model and then seek to eliminate some features by setting the associated coefficients to 0 while preserving model performance, such as recursive feature elimination using support vector machine [44]. The second category consists of models with an embedded feature selection process, such as ID3 [81] and C4.5 [82]. Following, we will cover these feature selection approaches in more detail.

2.2.2 Decision Trees

Due to the fact that decision trees, such as C4.5 [82], naturally carry out feature selection at each node, they are frequently utilised as embedded approaches. For the purpose of

feature selection [34], single tree models were utilised; however, the quality of the selected features may be reduced due to the fact that the precision of a single tree model may be limited. On the other hand, it is expected that tree ensembles, which are made up of several different trees, are substantially more accurate than a single tree [18].

2.2.3 Random Forest

Random Forest, unlike the vast majority of other classifiers, directly performs feature selection while simultaneously building a classification rule [79]. In other word, Random forest, when used as a classifier, does an implicit feature selection by only employing a small group of "strong variables" for the classification [19]. The Gini significance index and the permutation importance index (PIM) [18] are the two metrics of variable importance that are most frequently employed in Random Forest.

The Gini significance index can serve as a broad indicator of the relevance of the features being considered. This feature relevance score gives a relative ranking of the specific features and is technically a by-product of the training process for the random forest classifier. It is as follows: The Gini impurity is a computationally efficient approximation to the entropy. It measures how well a potential split is separating the samples of the two classes in this particular node. The Gini impurity is used to search for the optimal split at each node within the binary trees of the random forest [68].

Permutation importance measure (PIM) is likely the most often utilised measure of variable importance in random forest. The random forest method does not utilise all training data when building a single tree. This leaves a set of out of bag (OOB) samples, which can be used to test the classification accuracy of the forest. To determine the significance of a particular feature in the tree, permute the values of this feature in the OOB samples and compare the classification accuracy of the intact OOB samples to that of the OOB samples with the feature permuted [68].

2.2.4 Recursive Feature Elimination (RFE)

Recursive feature elimination (RFE) is a feature selection strategy for small sample classification problems [44] among a variety of other feature selection methods. Recursive feature elimination is initially applied to microarray-based cancer classification, where the number of training samples is less than 100 and the number of features is in the tens of thousands, and has evolved into an efficient method for small-sample feature selection. The goal of recursive feature elimination is to improve the performance of generalisation by deleting

the features that are considered to be of the least importance and whose removal will have the least impact on the amount of training errors [23].

2.3 Classifications

One of the tasks that intelligent systems conducts the most frequently is supervised classification. As a result, a significant number of methodologies founded on Artificial Intelligence and Statistics have been established. In supervised learning, the objective is to construct a concise model of the distribution of class labels in terms of predictive variables in order to understand the data better. The resulting classifier is subsequently utilized in order to assign class labels to the testing examples that include known values for the predictor features but an unknown value for the class label [59].

2.3.1 Decision Trees

In the process of building classification models, decision-tree approaches have seen extensive use. Each test in a decision tree compares a numeric attribute to a threshold value or a nominal attribute to a set of possible values. Decision trees are sequential models that logically combine a number of simple tests. The nodes of a decision tree each reflect a feature of an instance that needs to be categorised, and the branches each represent a possible value for the node to take on. The information gain concept is used to determine which aspects of the attribute tree have a greater influence on the classification. These aspects are located closer to the top of the tree. The classification of instances begins at the root node, and the instances are then ordered according to the feature values they possess.

A decision tree will label a data point as belonging to the partitioned region's most frequent class whenever that data point falls within one of the partitioned regions.

In decision tree classification, there are two primary processes. In the first step, we create the tree based on a training set. This is often done by beginning with an empty tree and selecting the suitable test attribute for each decision node with an attribute selection measure. The basic idea is to pick the property that minimizes the mixing of object classes across all of the training subsets generated by the test. This will make it simpler to categorise objects into their appropriate groups. The process is repeated for each sub decision tree until it reaches the leaves, at which point it fixes the classes that are associated with those leaves.

In the subsequent stage, we are able to make use of a new instance, which possesses simply the values of all of its attributes. We begin at the base of the newly constructed tree and go down the path that corresponds to the value of an attribute that has been seen in one of the interior nodes of the tree. This procedure is repeated up to the point where a leaf is found. In the end, we make use of the linked label in order to acquire the value of the predicted class for the current instance [58, 59].

On the other hand, the comprehensibility of decision trees is one of the properties that contributes the most to their utility. People are able to quickly and easily comprehend the rationale behind a decision tree's assignment of a given instance to a particular class.

The comprehensibility of these classifiers is superior than that of black-box models such as neural networks. When compared with the numerical weights of the connections between the nodes in a neural network, the logical rules that are followed by a decision tree are considerably simpler to interpret and understand. When making decisions, people have a tendency to feel better at ease when using models that they can grasp [58, 59].

In conclusion, decision trees are a trustworthy and efficient method for making decisions. They offer a high degree of classification accuracy while maintaining a straightforward representation of the information that has been acquired. Applications in the fields of medicine and health care have made extensive use of decision trees for more than 20 years [78].

The root node of the tree would be the feature that best splits the training data. There are several ways for determining which feature best splits the training data, including information gain [48] and the gini index [20].

While nearsighted methods estimate each feature separately, the Relief algorithm [56] evaluates them in relation to other features. The majority of studies, however, have concluded that there is no one best method [72]. Individual technique comparison may still be useful when determining the metric to utilise in a given dataset.

The same procedure is then repeated until the training data is broken down into subsets of the same class, after which the same procedure is performed on each partition of the divided data.

C4.5 [89] is considered to be one of the most well-known algorithms in the literature for building decision trees. It is an extension of Quinlan's earlier ID3 algorithm that was introduced in 1979 [80]. Taking into account that decision trees are considered one of the most important learning algorithms and they are compared with other learning algorithms in a recent study [64], decision trees are considered to be one of the best learning algorithms. In this study, it was shown that C4.5 is a very good combination of speed as well as error rate in terms of performance.

To sum up, one of the most useful characteristics of decision trees is their comprehensibility. People can easily understand why a decision tree classifies an instance as belonging to a specific class.

2.3.2 Random Forest Classifier

A random forest classifier is an example of an ensemble classifier that, like its name implies, generates numerous decision trees by employing a subset of training examples and features that are chosen at random [18].

The Random Forest classifier has been getting a lot of attention recently due to the outstanding classification results obtained, the speed of processing, and the capacity to deal with a huge attribute space [31, 75, 84].

As a result of these characteristics, random forests have found widespread use across a variety of fields. There have been several studies carried out in real-world application specially in the medical field that fall under the scope of our interests, and these studies made use of random forests in the course of their work [15, 38, 47, 54, 61, 62, 103–105].

Research into machine learning has shown a significant amount of interest in ensemble learning, which refers to systems that generate a large number of models and integrate the outputs of these models. It is a widely held belief that the performance of a set of numerous weak classifiers is typically superior to that of a single classifier when the same quantity of train data is provided [85]. The wisdom of the crowds is the core idea that supports these methods. This idea is deceptively straightforward but incredibly effective. The performance of a large number of models acting as a committee will exceed that of any of the individual constituent models [12].

Boosting [33], bagging [17], and more recently Random Forests [18] are three well-known examples of ensemble approaches.

- **Boosting**

  By iteratively reweighing the occurrences contained in the training set, the boosting strategy generates multiple distinct base learners. At the outset, each instance is given a weighting that is equivalent to the others. Each instance that was misclassified by the previous base learner will receive a larger weight in the subsequent round in an effort to correctly classify it. After computing the error, the weight of the instances that were successfully classified is decreased, whilst the weight of the instances that were not correctly classified is increased. Every single learner's vote carries the same amount of weight relative to their overall performance [33, 99].

- **Bagging**

  In the bagging approach, also known as Bootstrap Aggregation, various training subsets are randomly selected with replacement from the entire training set. Learners at the base level get each training subset as an input. A vote by a majority of learners brings together all of the extracted learners. Bagging is able to build classifiers in parallel, while boosting generates them one at a time in a sequential manner [17, 74].

  Rather than subdividing the training data into smaller chunks and training each tree on a separate chunk, we use bagging to train the trees on a dataset of size N with replacement. It means that if we have a sample that is size N, we will continue to provide each tree with a training set that is of size N. However, rather than using the initial set of training data, we will be using a random sample of size N with replacement [12].

- **Random Forest**

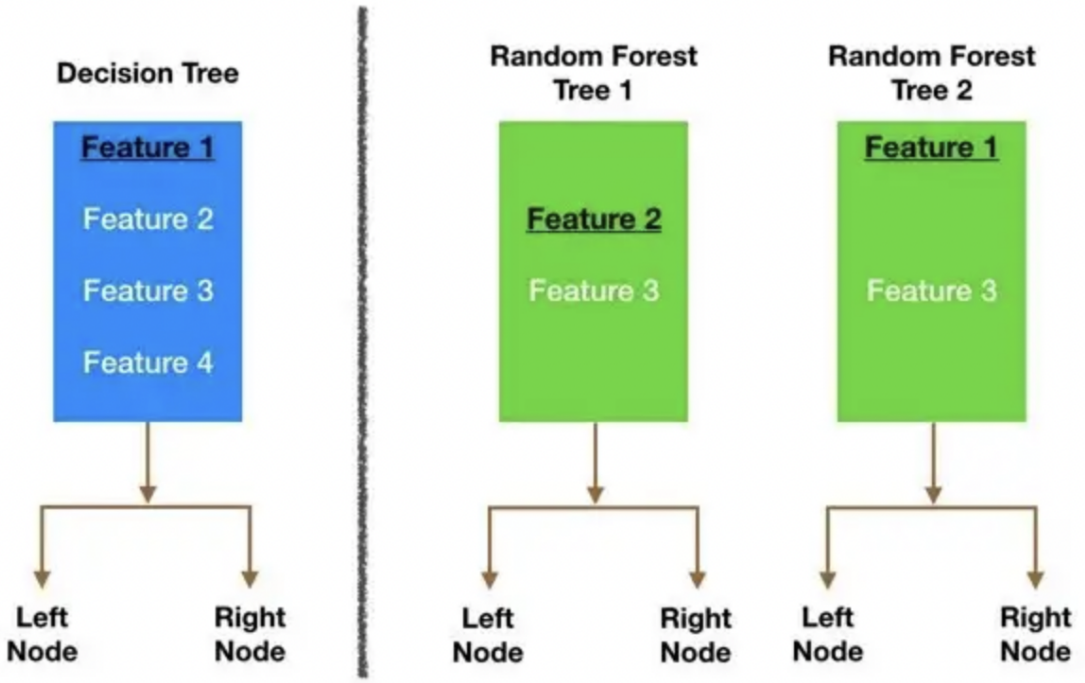
  Another type of ensemble approach is known as Random Forest, and it works by first building a large number of decision trees, which are then used to classify a new instance according to the votes cast by the majority of the trees [18, 74].

  Decision trees are extremely sensitive to the data they are trained on; hence, even slight modifications to the dataset used for training might result in dramatically different tree architectures. The random forest algorithm makes use of this property by permitting each individual tree to randomly pick from the dataset with replacement, so producing a variety of trees. This procedure is referred to as bagging. Random forest takes advantage of this property. The randomization of the features inside a random forest is another distinction that can be made between it and a decision tree 2.1. When it comes time to split a node in a typical decision tree, we take into account all of the alternative features and select the one that creates the greatest amount of differentiation between the observations included in the left node and those contained in the right node. In contrast, only a random subset of characteristics are available for selection by each individual tree in a random forest. This ultimately leads to less correlation between the trees in the model, which allows for a greater degree of diversification, and it forces even more variance among the trees in the model.

  As a result of bagging, the random forest produces trees that are not only trained on diverse sets of data, but also make decisions based on a variety of distinct features [12].

  The random forest is a classification system composed of several uncorrelated decision trees whose aggregated prediction is more accurate than that of any individual tree. This characteristic, along with drawing observations with replacement and dividing

Figure 2.1: In a random forest, the process of node splitting is determined by a subset of random features applied to each tree [12].



nodes based on the optimal split among a random subset of features (instead of whole features) selected at each node, minimises the risk of overfitting in random forest models. In other words, by injecting randomness, random forest reduces overfitting in comparison to decision tree.

Random Forest in Medical Domain

The primary objective of this study [15] is to detect and categorise curvilinear structure in mammograms, as well as to determine whether or not this structure should be considered normal or abnormal. To accomplish their purpose, they employ random forest. In this research, an automatic method for segmenting multiple sclerosis (MS) lesions in three-dimensional magnetic resonance (MR) images was provided. The architecture for this method is based on a discriminative random decision forest, and it offers a probabilistic voxel-by-voxel categorization of the volume [38]. In order to accurately detect acute appendicitis, they built models based on random forests, support vector machines, and

artificial neural networks [47].

The purpose of this research [54] is to offer a method for the effective classification of X-ray images in order to improve both the accuracy and performance. Random Forests is used to perform classification tasks quickly and accurately. An effective strategy for the retrieval of medical images based on keywords is presented in [61], and it involves the use of image classification with Random Forests.

In real-time 3D echocardiography, the automatic delineation of the myocardium has the potential to be employed as a diagnostic aid for a variety of cardiac conditions, including ischaemia. In this study [62], the authors employ a random forests approach and handle the problem at hand as a three-dimensional patch classification assignment with two classes.

In the paper [103], the authors build an automatic 3D Random Forests algorithm that can be used to segment the foetal femur in 3D ultrasound images, and they suggest a weighted voting mechanism as a way to generate a probabilistic class label from the segmentation results.

Improvements to Random Forests for the purpose of segmenting three-dimensional objects in various types of three-dimensional medical imaging [104]. By strategically focusing on the "good" features and ignoring the unnecessary ones, it is possible to achieve a voxel classification that is more accurate. This strategy also results in a more efficient learning process. During the testing phase, it is suggested that assigning a weight to each individual tree in the forest will produce a probabilistic judgement that is both objective and more precise.

In this paper [105], a brand novel algorithm for the automatic segmentation and categorization of brain tissue derived from 3D MR data is given. It employs a discriminative Random Decision Forest classification method and takes partial volume effects into consideration.

2.3.3 Extra Tree Classifier

Extremely Randomized Tree, also known as Extra Tree, is an additional ensemble method that can be used for supervised classification and regression issues. Essentially, what it involves is significantly randomising the attribute choice as well as the cut-point choice whenever one is splitting a tree node. In the most extreme scenario, it constructs trees whose architectures are fully random and independent of the output values of the learning sample [39].

Figure 2.2: Algorithm for splitting Extra Trees [39].

Split_a_node(S)
Input: the local learning subset S corresponding to the node we want to split
Output: a split $[a < a_c]$ or nothing
– If **Stop_split**(S) is TRUE then return nothing.
– Otherwise select K attributes $\{a_1, \ldots, a_K\}$ among all non constant (in S) candidate attributes;
– Draw K splits $\{s_1, \ldots, s_K\}$, where s_i = **Pick_a_random_split**(S, a_i), $\forall i = 1, \ldots, K$;
– Return a split s_* such that Score(s_*, S) = $\max_{i=1,\ldots,K}$ Score(s_i, S).

Pick_a_random_split(S,a)
Inputs: a subset S and an attribute a
Output: a split
– Let $a^S_{\max}$ and $a^S_{\min}$ denote the maximal and minimal value of a in S;
– Draw a random cut-point a_c uniformly in $[a^S_{\min}, a^S_{\max}]$;
– Return the split $[a < a_c]$.

Stop_split(S)
Input: a subset S
Output: a boolean
– If $|S| < n_{\min}$, then return TRUE;
– If all attributes are constant in S, then return TRUE;
– If the output is constant in S, then return TRUE;
– Otherwise, return FALSE.

It is similar to Random Forest in that it constructs numerous trees using the traditional top-down procedure, but it differs in the manner in which randomization is injected during the training process. Extra Trees is an alternative to Random Forest. Two of the most important distinctions are that Extra tree does not bootstrap observations (which means that it samples without replacement) and that nodes are split based on random splits rather than the best splits. Algorithm for splitting Extra Trees shown in Figure 2.2. This distinction indicates that the best split at a node is determined by analysing a subset of all of the available features. A single threshold is chosen at random for each feature, as opposed to initially searching for the optimal threshold that corresponds to each feature [40]. The increased level of randomness that occurs during training results in the production of more independent trees, which in return further reduces the variance. Because of this, ExtraTrees typically produce outcomes that are marginally superior than those produced by Random Decision Forests.

2.3.4 K Nearest Neighbors Classifier

Instance-based learning algorithms are lazy learning algorithms [70]. The k-nearest neighbour classifier is the foundation of many lazy learning algorithms [28] since it maintains the training set and delays classification decision making until problem solving [30].

The difference between eager-learning algorithms (such as decision trees, neural and Bayes nets) and lazy-learning algorithms is that lazy-learning approaches require more computation time during the classification process but less during the training phase. The closest neighbour approach is widely recognised as one of the easiest instance-based learning algorithms to comprehend and apply.

The idea behind k-Nearest Neighbor, or kNN, is that instances within a dataset will typically exist in close proximity to other instances that have similar properties and are labelled with a classification label. If this is the case, then the value of the label of an unclassified instance can be determined by observing the class of the instance's nearest neighbours [26].

The k-nearest neighbour algorithm finds the k instances that are the closest to the query instance and determines the class of the query instance by finding the one class label that occurs the most frequently.

In other words, when presented with a new instance to classify, the k-NN method finds the k most similar cases and guesses the class to which the new instance could belong [30].

Instances can be thought of in a general sense as points that exist within an n-dimensional instance space, where each of the n-dimensions corresponds to one of the n-features that are used to characterise an instance. In other words, an instance space can be thought of as having n+1 dimensions. It is less important to focus on the examples' actual positions inside this space and more important to consider their relative distances from one another. The utilisation of a distance measure allowed for the calculation of this relative distance [58, 59].

In an ideal world, the distance metric would reduce the distance that exists between examples that are similarly classed while simultaneously increasing the distance that exists between instances that belong to different classes. Many distinct metrics, such as the Manhattan distance, the Euclidean distance, the Minkowski distance, and many others, have been offered.

2.3.5 Support Vector Machine

Support Vector Machines are one of the most prevalent techniques for supervised machine learning. SVMs are predicated on the concept of a margin, which refers to either side of a hyperplane that divides two data classes. It has been demonstrated that increasing the margin and so generating the biggest possible distance between the separating hyperplane and the instances on either side of it can minimise an upper constraint on the expected generalisation error.

In the situation when the data can be separated linearly, once the optimal separating hyperplane has been identified, the data points that lie on the margin of that hyperplane are referred to as support vector points, and the solution is represented as a linear combination of only these points. Other data points are not taken into consideration. Therefore, the amount of features that are included in the training data does not have an impact on the complexity of the model that an SVM uses (the number of support vectors selected by the SVM learning algorithm is usually small). For this reason, support vector machines are ideally suited to handle learning tasks in which the number of features is rather high in comparison to the number of training examples. Even though the greatest margin enables the SVM to choose between numerous candidate hyperplanes, the SVM may not be able to locate any separating hyperplane at all for many datasets since the data contains instances that were incorrectly classified. The issue can be resolved by employing a flexible margin that allows for certain misclassifications of training examples [98].

Despite this, the vast majority of situations that arise in the actual world include non-separable data for which there is no hyper-plane that can properly differentiate between the positive and the negative instances in the training set. In order to solve the issue of inseparability, one possible option is to map the data onto a space with a higher dimension and then define a separating hyperplane in that space. This higher-dimensional space is known as the feature space, as opposed to the training instances' input space. With a correctly designed feature space of adequate dimension, every consistent training set may be separated [21, 59].

2.3.6 Other Classifiers

- **Passive Aggressive Classifier**

  The passive aggressive classifier is an algorithm used in machine learning for classification tasks. The traditional Perceptron algorithm has been altered to create this new method. It is one of the few algorithms designed for online learning [27]. Passive-Aggressive algorithms are referred to as such because:

  If the prediction is accurate, the model should be retained with no modifications. To put that another way, the data presented in the example are insufficient to produce any discernible shifts in the model.

  Aggressive: If the prediction is inaccurate, modify the model. Therefore, a modification to the model may be required to correct the issue.

- **Perceptron Classifier** The Perceptron [86] is a machine learning technique for

binary classification tasks that is linear. It is regarded to be one of the earliest and simplest types of artificial neural networks. It consists of a single node or neuron that receives as input a row of data and predicts a class label. This can be accomplished by computing a weighted sum of the inputs

- **MultiLayer Perceptrons Classifier** The multilayer perceptron is a type of model that represents a nonlinear mapping between an input vector and an output vector. It is formed of a system of basic neurons or nodes that are interconnected with one another. The weights are a function of the sum of the inputs to the node, and the output signals are functions that are modified by a simple nonlinear activation function. Multilayer perceptrons are able to approximate nonlinear functions due to the superposition of numerous basic nonlinear transfer functions [35].

- **Ridge Classifier** Ridge classification is a method that is utilised in the process of analysing linear discriminant models. In this type of regularisation, model coefficients are penalised in

- **Gaussian Process Classifier** The Gaussian Processes Classifier is a non-parametric technique that can be utilised for binary classification applications. The Gaussian probability distribution serves as the basis for Gaussian Processes, which are a more generalised form of the distribution.

- **Bernoulli Naive Bayes** BernoulliNB is a classification algorithm that implements the naive Bayes algorithm for use with data that is distributed in accordance with multivariate Bernoulli distributions [10].

- **Gaussian Naive Bayes** GaussianNB is a classification algorithm that implements the naive Bayes algorithm for use with data that is distributed in accordance with Gaussian distributions [10].

- **Label Propagation and Label Spreading classifiers** The label propagation algorithm is a semi-supervised form of machine learning that assigns labels to data points that were not labelled in the beginning. At the beginning of the procedure, only a subset of the data points (which is typically quite small) has labels (or classifications). During the course of the procedure, these labels are transferred to the points that have not yet been given labels [108].

  The Label Spreading method is quite comparable to the Label Propagation algorithm, with just a few key distinctions between the two. For instance, the calculations performed by the Label Spreading algorithm utilize a symmetric normalised

graph Laplacian matrix, whereas the calculations performed by the Label Propagation method employs a random walk normalised Laplacian.

Chapter 3

Methodology

3.1 Introduction

As stated in Chapter 1, the purpose of this thesis is to investigate several machine learning and data mining algorithms, such as Decision Tree, Random Forest, Naive Bayes, and K Nearest Neighbor algorithms, in order to classify IBD and HS diseases. Specifically, the goal of this investigation is to determine which of these algorithms is most effective at identifying IBD and HS diseases. In addition, another objective of this research is to come up with a set of criteria and select the features that are the most significant and decisive in distinguishing between these two disorders.

In this chapter, we will walk through the essential steps for data cleaning and pre-processing. The step-by-step instructions for this process can be found in the section 3.3. Following that, the details of each classification approach will be explained. The results of different classification systems are compared and displayed in Table 3.1. Next, the section 3.5.3 will discuss the procedures used in order to extract the most significant feature from the dataset. These characteristics are especially essential for the clinician to consider while searching for precise criteria to use in the classification of IBD and HS. Ultimately, all classification techniques were applied to the newly reduced dataset, which only included the most defining features. Tables 3.2, 3.1, 3.6, 3.5, 3.4, and 3.3 depict the outcome of a comparison of several feature selection and categorization methods.

3.2 Dataset

We have 60 features for each patient, which are divided into the following 6 category:

- **Demographic information:**
  There are 6 features in this category: Gender, age, body mass index (BMI), race, history of smoking, and digestive symptoms.
- **Morphologic information about their perianal lesions:**
  There are 15 features in this category: Bilateral or unilateral lesion, presence or absence of fistula, abscess, nodule, comedo, pustule, induration, plaques, ulcer, erythema, anal tag, scar, tunnel/sinus tract, knife cut ulcers, and genital cutaneous edema.
- **Locations of extra perianal lesion:**
  There are 9 features in this category: Buttock, perineum, axilla, chest, groin, thigh, scrotum, vulva, and back.
- **Lab results:**
  There are 16 features in this category: White blood cell counts (min and max), neutrophil counts (min and max), hemoglobin level (min and max), serum albumin level (min and max), albumin (min and max), erythrocyte sedimentation rate (ESR) (min and max), C-reactive protein (CRP) level (min and max), and fecal calprotectin level (min and max).
- **MRI findings about their perianal lesion:**
  There are 9 features in this category: Presence or absence of transsphincteric /Intersphincteric fistula, subcutaneous tunnels, abscess, inflammation of subcutaneous + skin, inflammation of fat + subcutaneous, inguinal lymphadenopathy, iliac lymphadenopathy, mesoractal lymphadenopathy, and rectal inflammation.
- **Colonoscopy information:**
  There are 5 features in this category: Ulcer, erythema, stricture, fistula, and affected mucosa.

3.3 Data Cleaning

For machine learning, data is the most valuable and indispensable resource. Nonetheless, a flawed dataset may lead to incorrect conclusions. In data analytics, detecting and cleaning

dirty data is a fundamental challenge, and failing to do so can result in incorrect analyses and unreliable conclusions [25]. The act of finding parts of a dataset that are incorrect, incomplete, improperly formatted, duplicate, inaccurate, or missing and afterward, depending on the necessity, altering, replacing, or removing those parts of the dataset that contain incorrect, incomplete, or so is what is known as data cleaning [83]. In this section, several cleaning approaches were used to clean the dataset:.

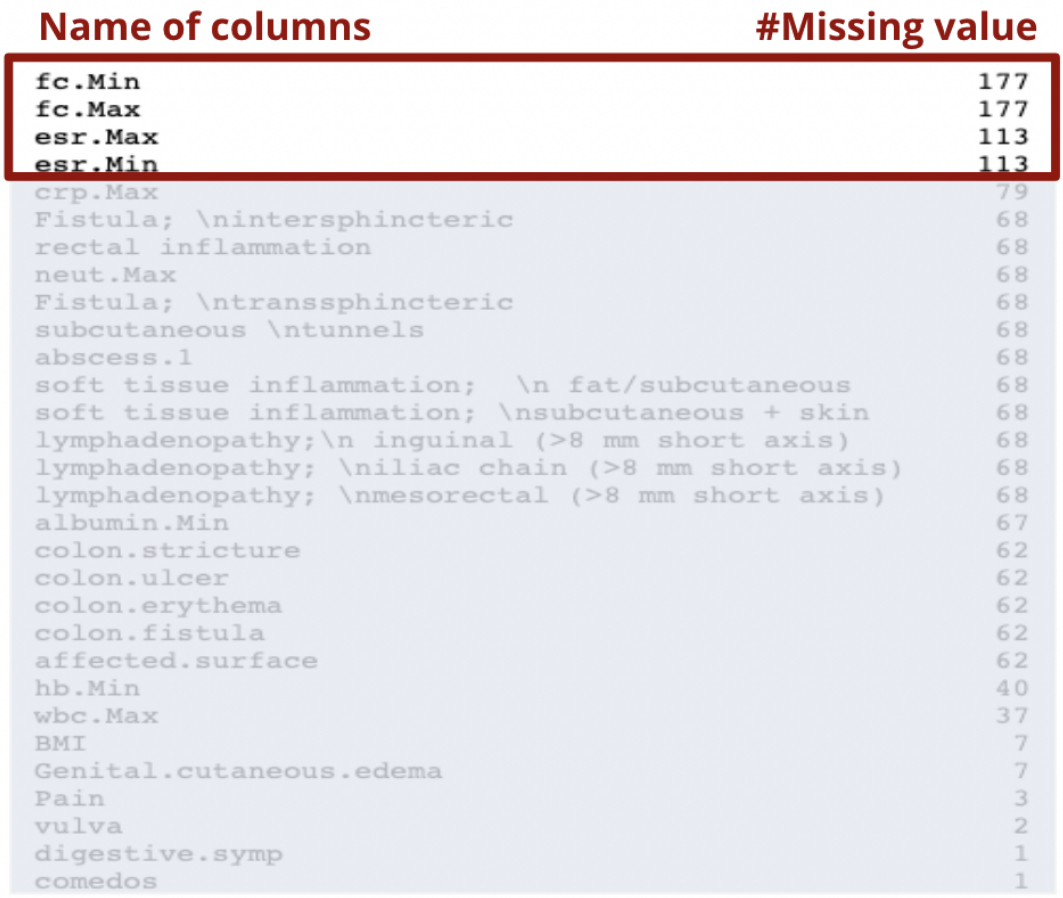
Figure 3.1: A total of 198 records (rows) are included in this dataset. There are many missing values in our dataset which need to be noted. For example, 177 out of 198 data points in the *fc.Min* feature are empty, corresponding to about 89 % of all data points in this feature.

Name of columns	#Missing value
fc.Min	177
fc.Max	177
esr.Max	113
esr.Min	113
crp.Max	79
Fistula; \nintersphincteric	68
rectal inflammation	68
neut.Max	68
Fistula; \ntranssphincteric	68
subcutaneous \ntunnels	68
abscess.1	68
soft tissue inflammation; \n fat/subcutaneous	68
soft tissue inflammation; \nsubcutaneous + skin	68
lymphadenopathy;\n inguinal (>8 mm short axis)	68
lymphadenopathy; \niliac chain (>8 mm short axis)	68
lymphadenopathy; \nmesorectal (>8 mm short axis)	68
albumin.Min	67
colon.stricture	62
colon.ulcer	62
colon.erythema	62
colon.fistula	62
affected.surface	62
hb.Min	40
wbc.Max	37
BMI	7
Genital.cutaneous.edema	7
Pain	3
vulva	2
digestive.symp	1
comedos	1

- **Missing values**

  When it comes to the data from the real-world, it is hard to find a dataset that is complete and does not contain missing variables. Handling missing values is partic-

Figure 3.2: Deleting the entire column (feature) is one of the method for dealing with missing value. This strategy is utilized when more than 25 % or 30% of the data is missing. By using this method we will remove some specific features. The highlighted features in this table are those that were chosen to be eliminated.



Name of columns	#Missing value
fc.Min	177
fc.Max	177
esr.Max	113
esr.Min	113
crp.Max	79
Fistula; \nintersphincteric	68
rectal inflammation	68
neut.Max	68
Fistula; \ntranssphincteric	68
subcutaneous \ntunnels	68
abscess.1	68
soft tissue inflammation; \n fat/subcutaneous	68
soft tissue inflammation; \nsubcutaneous + skin	68
lymphadenopathy;\n inguinal (>8 mm short axis)	68
lymphadenopathy; \niliac chain (>8 mm short axis)	68
lymphadenopathy; \nmesorectal (>8 mm short axis)	68
albumin.Min	67
colon.stricture	62
colon.ulcer	62
colon.erythema	62
colon.fistula	62
affected.surface	62
hb.Min	40
wbc.Max	37
BMI	7
Genital.cutaneous.edema	7
Pain	3
vulva	2
digestive.symp	1
comedos	1

ularly crucial because the majority of algorithms will not accept missing values in their inputs. As can be seen in Table 3.1, there are numerous missing values in our data that required to be addressed.

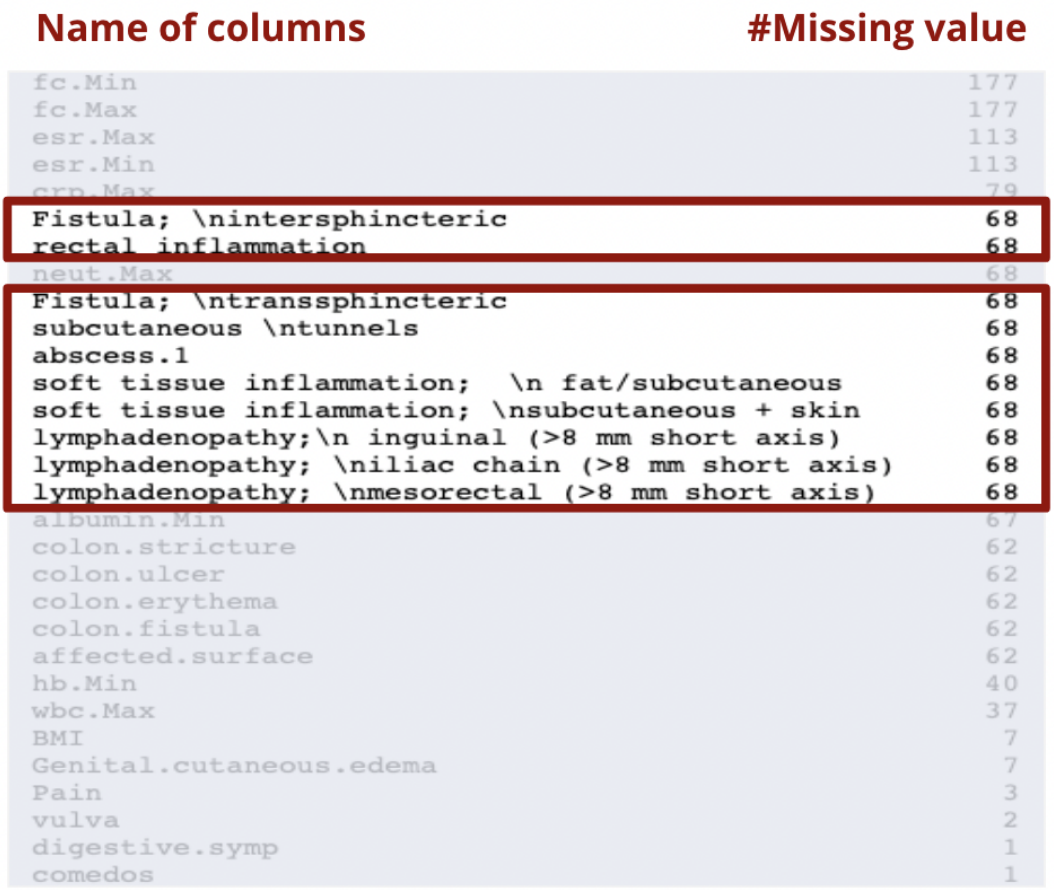
When dealing with missing data, there are several methods that can be used. One method is to delete the entire column (feature) [1]. This method is used when more than 25 % or 30% of the data is missing [5]. However, in some circumstances, we do not want to lose a specific feature, so we can delete the related row. In this case, we will lose the entire data point associated with a certain patient. However, this strategy is not appropriate when the majority of the cells in a column are empty. For example, if we wish to use this approach for *fc.Min*, we must eliminate 177 rows out of 198, which is not reasonable.

Another technique is to replace the missing value with another value, which could be the mean or most frequent value, or it could come from field knowledge in some circumstances [52]. However, we should proceed cautiously when employing this method because it has the potential to mislead us.

As previously indicated, this dataset has 60 columns (features). During the initial round of data cleaning, we asked medical professionals to eliminate any irrelevant columns, which included the *chest*, *thigh*, *back*, *wbc.Min*, *neut.Min*, *hb.Max*, *crp.Min*, and *albumin.Max* columns. We have 198 rows and 52 features after this procedure.

- **Drop column**
  If the majority of a column's values are missing (more than 25 to 30 percent), that column (feature) should be removed; nevertheless, this will result in the loss of information, thus more evaluation is required. Due to the fact that the vast majority of *fc.Min*, *fc.Max*, *esr.Max*, and *esr.Min* features have been missed (about more that 60 percent), we dropped these 4 columns that are displayed in Table 3.2. At the end of this process, we have 198 rows and 48 features.
- **Drop row**
  As previously said, we ought to think carefully before eliminating part of data. Given that more than 25 percent of the data in the highlighted columns in Table 3.3 are missing, we should have deleted these columns. However, after discussing with clinicians, we discovered that these columns are related to the results of MRI images, which are important for this study; therefore, we have decided to keep these features so that we can conduct further analysis. As a solution to the problem of missing values, we remove the rows that correspond to those values; as a consequence, some of the missing values of other features are also removed. 130 rows and 48 columns remain after some rows were removed.

Figure 3.3: Sometimes only a small percentage of the data in a particular column is missing, in which case it would not be appropriate to eliminate the entire column for those few data points. Additionally, there are situations when we do not want to lose a certain feature. In this case, eliminating a row is preferred. In this scenario, the entire data point related to a certain patient will be lost.



Name of columns	#Missing value
fc.Min	177
fc.Max	177
esr.Max	113
esr.Min	113
crp.Max	79
Fistula; \nintersphincteric	68
rectal inflammation	68
neut.Max	68
Fistula; \ntranssphincteric	68
subcutaneous \ntunnels	68
abscess.1	68
soft tissue inflammation; \n fat/subcutaneous	68
soft tissue inflammation; \nsubcutaneous + skin	68
lymphadenopathy;\n inguinal (>8 mm short axis)	68
lymphadenopathy; \niliac chain (>8 mm short axis)	68
lymphadenopathy; \nmesorectal (>8 mm short axis)	68
albumin.Min	67
colon.stricture	62
colon.ulcer	62
colon.erythema	62
colon.fistula	62
affected.surface	62
hb.Min	40
wbc.Max	37
BMI	7
Genital.cutaneous.edema	7
Pain	3
vulva	2
digestive.symp	1
comedos	1

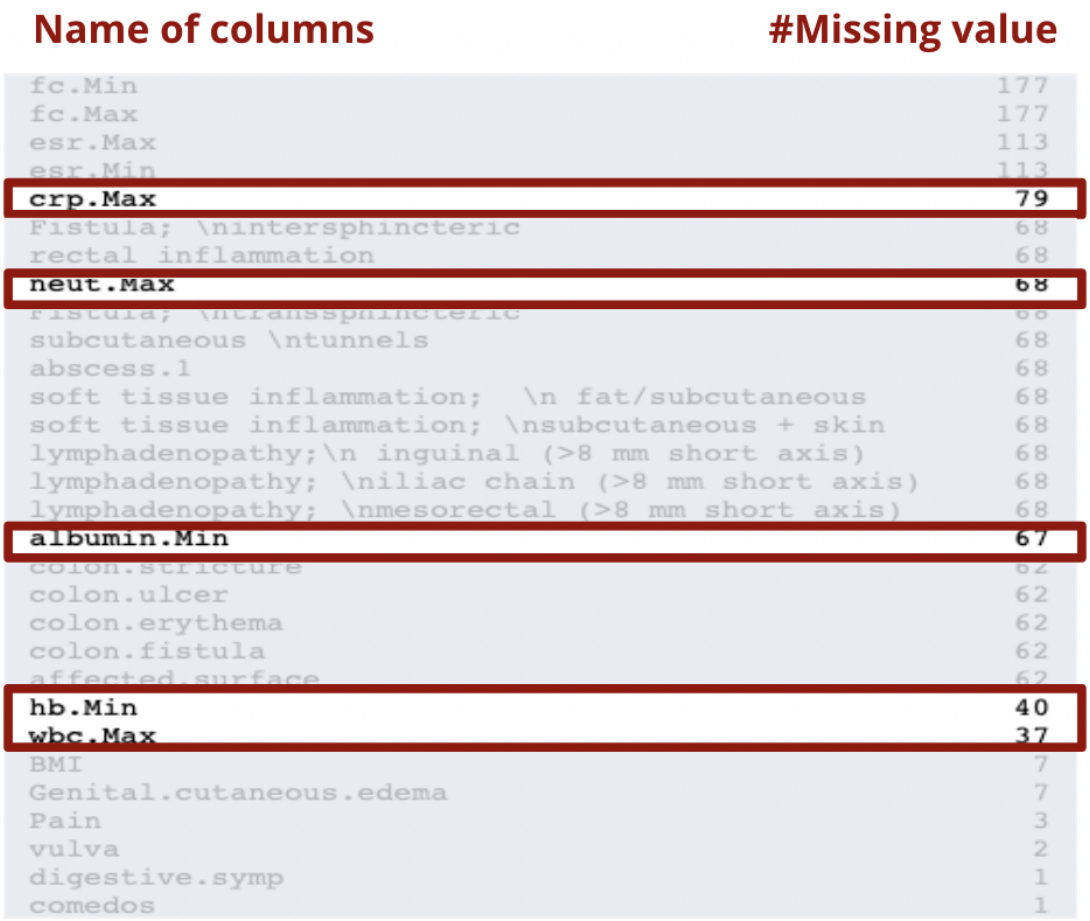
- **Fill missing values**
  Filling in missing values based on other observations is an alternative method for tackling the missing value problem; however, there is a risk of compromising the data integrity, as you may be relying on assumptions rather than actual observations. We are able to fill in the missing values with a variety of statistical approaches, depending on our requirements. The process of filling in missing values is performed differently for categorical data compared to numerical data. We will utilize the statistical mean method to fill in missing values for numerical data. On the other hand, we are able to use the most frequent group when dealing with categorical data. We replaced the empty values for the numerical features specified in Table 3.4 with the mean of each column. The number of rows and columns, 130 rows and 48 columns, remains unchanged. On the contrary, given that the highlight features in Table 3.5 contain categorical data, we substitute the missing values with the value that occurred the most frequently in that column.

At the end of this process, we deleted colonoscopy data as well since it was not in the focus of this study, thus we removed 5 highlighted columns in Table 3.6 related to colonoscopy data. Therefore, in the end, **130 rows and 43 features** remain for additional analysis.

- **Data Normalization**
  One of the fundamental processes that must be completed before moving on to any other attempt is known as data normalization. In our study, we intend to categorize two diseases using distinct features from our dataset; however, these numerical feature ranges (scale) may vary. For instance, the range of the feature *albumin.Min* is between (1.8 and 4.8), yet the range of the feature age is between (16, 89). It is possible that comparing these two variables using different scales will give us misleading results. In this circumstance, it is impossible for us to compare these features. Therefore, in order to prevent this problem and ensure that all of our features are treated with the same level of importance, our dataset needs to be normalised and scaled to the same level.

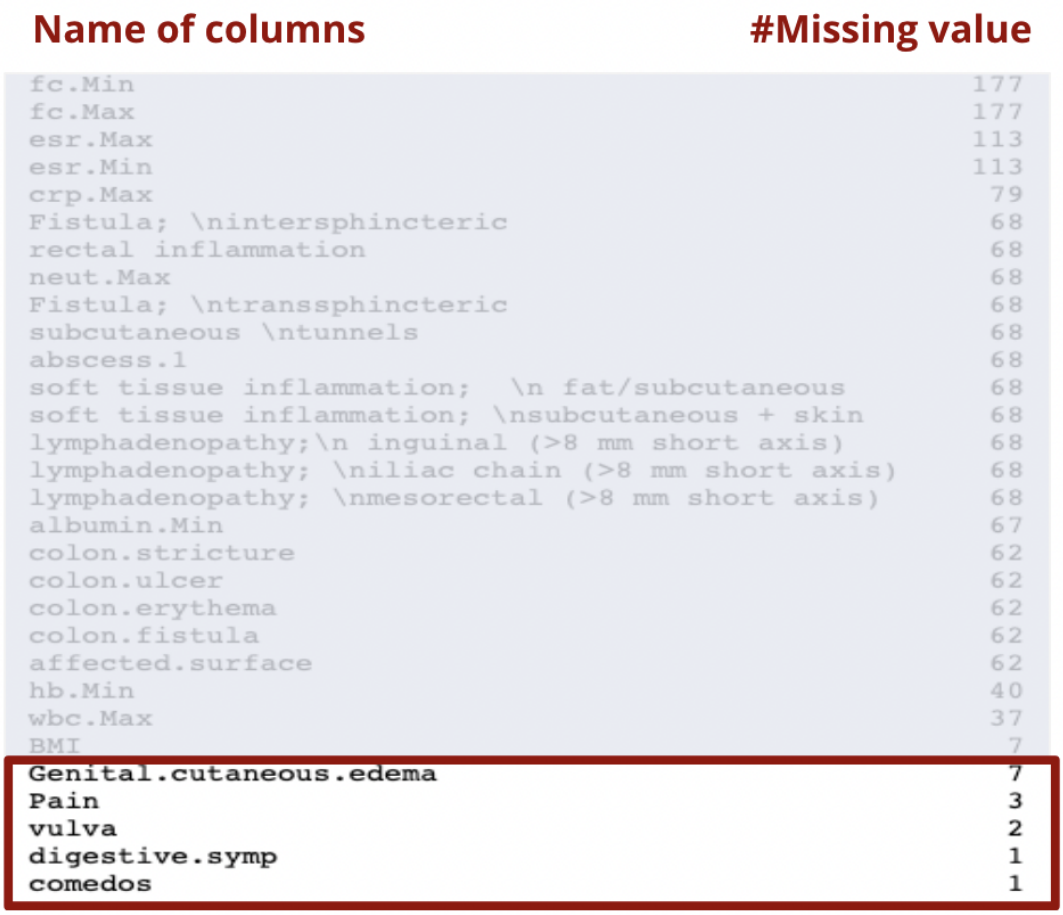
  We utilized Min-Max Normalization (i.e., re-scaling) in our experiments. Min-Max Normalization is one of the most prevalent approaches to re-scale data into a desired range. The data can be linearly transformed using this method while maintaining their original range, and it implies that the relationship between the original data is

Figure 3.4: Another method for handling missing data is to replace a different value for the one that is missing. For categorical data as opposed to numerical data, the process of filling in missing values is carried out differently. For the numerical data highlighted in this table, we will use the statistical mean approach to fill in the missing values.



Name of columns	#Missing value
fc.Min	177
fc.Max	177
esr.Max	113
esr.Min	113
crp.Max	79
Fistula; \nintersphincteric	68
rectal inflammation	68
neut.Max	68
Fistula; \ntranssphincteric	68
subcutaneous \ntunnels	68
abscess.1	68
soft tissue inflammation; \n fat/subcutaneous	68
soft tissue inflammation; \nsubcutaneous + skin	68
lymphadenopathy;\n inguinal (>8 mm short axis)	68
lymphadenopathy; \niliac chain (>8 mm short axis)	68
lymphadenopathy; \nmesorectal (>8 mm short axis)	68
albumin.Min	67
colon.stricture	62
colon.ulcer	62
colon.erythema	62
colon.fistula	62
affected.surface	62
hb.Min	40
wbc.Max	37
BMI	7
Genital.cutaneous.edema	7
Pain	3
vulva	2
digestive.symp	1
comedos	1

Figure 3.5: As previously indicated, the procedure of filling in missing values for categorical data differs from that of numerical data. When working with categorical data, we can fill in the empty cells by using the most frequent value in each column.



Name of columns	#Missing value
fc.Min	177
fc.Max	177
esr.Max	113
esr.Min	113
crp.Max	79
Fistula; \nintersphincteric	68
rectal inflammation	68
neut.Max	68
Fistula; \ntranssphincteric	68
subcutaneous \ntunnels	68
abscess.1	68
soft tissue inflammation; \n fat/subcutaneous	68
soft tissue inflammation; \nsubcutaneous + skin	68
lymphadenopathy;\n inguinal (>8 mm short axis)	68
lymphadenopathy; \niliac chain (>8 mm short axis)	68
lymphadenopathy; \nmesorectal (>8 mm short axis)	68
albumin.Min	67
colon.stricture	62
colon.ulcer	62
colon.erythema	62
colon.fistula	62
affected.surface	62
hb.Min	40
wbc.Max	37
BMI	7
Genital.cutaneous.edema	7
Pain	3
vulva	2
digestive.symp	1
comedos	1

Figure 3.6: After employing all of the aforementioned approaches for dealing with missing values, we come to the table below. After consulting with medical professionals at Mayo Clinic, we decided to remove the entire highlighted columns to avoid losing any additional patient data.

Name of columns	#Missing value
fc.Min	177
fc.Max	177
esr.Max	113
esr.Min	113
crp.Max	79
Fistula; \nintersphincteric	68
rectal inflammation	68
neut.Max	68
Fistula; \ntranssphincteric	68
subcutaneous \ntunnels	68
abscess.1	68
soft tissue inflammation; \n fat/subcutaneous	68
soft tissue inflammation; \nsubcutaneous + skin	68
lymphadenopathy;\n inguinal (>8 mm short axis)	68
lymphadenopathy; \niliac chain (>8 mm short axis)	68
lymphadenopathy; \nmesorectal (>8 mm short axis)	68
albumin.Min	67
colon.stricture	**62**
colon.ulcer	**62**
colon.erythema	**62**
colon.fistula	**62**
affected.surface	**62**
hb.Min	40
wbc.Max	37
BMI	7
Genital.cutaneous.edema	7
Pain	3
vulva	2
digestive.symp	1
comedos	1

maintained after min-max normalization.

$$x_{MinMax} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.1}$$

In other words, the minimum value of each feature is changed to a zero, the highest value is changed to a one, and every other value is changed to a decimal between zero and one. This process is repeated for each feature [76].

3.4 Evaluation

Evaluation is a critical stage of machine learning and data mining studies. In this section, we will go over how to properly evaluate our model. There are several methods for splitting data to avoid data leaks, overfitting, and underfitting that will be described. Following that, we will discuss the evaluation metric that is critical for this study.

3.4.1 Split into Train, Validation and Test Datasets

One of the most significant aspects of evaluation is that we must evaluate performance metrics based on unseen data. To accomplish this, we keep different fragments of data for each purpose. In this case, we divided the data into train, validation, and test datasets. Then, initially, the model is fit to the training dataset. Subsequently, the fitted model is used to predict the responses for the data in the validation dataset. The test dataset is utilised to provide an unbiased evaluation of the model's final fit to the training dataset. If the data in the test dataset have never been used for training, the test dataset is also referred to as a holdout dataset.

The motivation for this technique is the concept of preventing data leaks. For this reason, we must preserve a part of the data from the whole model selection and training phase for the final evaluation [11].

3.4.2 k-Fold Cross-Validation

In the preceding approach for validating models, candidate models could only be evaluated once. If we would like to test each model multiple times with various datasets, we may use the k-fold approach to resample the same dataset multiple times by generating distinct subsets. Due to the fact that we are evaluating the model, the model must be trained

from scratch each time, without reusing previous training results. This is referred known as cross-validation [11].

In truth, it is rather similar to the train/test split, but it is applied to a larger number of subsets. In other words, we partition our data into k subsets to perform k train/validate/test cycles.

3.4.3 Leave-One-Out Cross-Validation

Leave-one-out cross validation, also known as LOOCV, is a variation of k-fold cross validation in which the value of k is equal to the total number of samples in the dataset and all test subsets consist of a single instance. In other words, a single observation is subjected to testing. The model is assessed for each observation withheld. The ultimate result is then computed by averaging all of the individual scores [8].

This technique overcomes the drawback of adopting limited training sets by fitting the model to nearly all of the training data.

This technique is computationally demanding and should only be applied to small datasets, as it would require a large number of training sets (equal to the number of samples). This method is useful, however, when the most precise estimation of a classifier's error rate is necessary in light of scarce data availability.

When all of these methods are considered, the LOOCV strategy emerges as the most reasonable solution for validating our solutions. On the one hand, our dataset is quite limited, and on the other, we lay a strong emphasis on reaching high levels of accuracy.

3.4.4 Metric

One parameter for evaluating classification models is accuracy. Accuracy is the proportion of correct predictions made by our model. Accuracy is defined as follows:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

3.5 Classification

There are several classification algorithms now in use in the medical field, each of which holds the potential to improve both the accuracy and the level of confidence [14, 69]. It is

not difficult to choose a classifier based on the data, regardless of whether the classifier is parametric or non-parametric [106].

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Due to the fact that tree-based approaches such as Decision Tree and Random Forest are non-parametric classifiers, they are able to be applied to data with unknown distribution. Since it is hard or unlikely to acquire normally distributed data in the medical field, non-parametric approaches are virtually always helpful [24]. However, certain alternative classifiers, such as MLP, naive bayes, and support vector machines, have also proven highly popular in classifying diagnostic data [67]. These classifiers, however, hold assumptions to simplify the learning process, which can sometimes result in a higher error rate. We have utilized a set of different classification algorithms, such as decision trees, random forests, naive Bayes, and KNN, within this particular instance. The objective is to determine the optimal classification model for accurately classifying these two diseases.

3.5.1 Classification with 43 features

In this step, the cleaned data in the previous section 3.3 are utilized. As previously indicated, 43 out of 52 features and 130 out of 198 patients remained following the cleaning. Table 3.1 displays the results of all classifiers on these 43 features. All of the classifiers

delivered solid results, with all of them scoring well above 80 % accuracy. Ridge, SVC, Ridge CV, BernouliNB, NuSVC, and Random Forest are among the strongest classifiers, according to the results shown in Table 3.1. However, because we are working with clinicians, it is crucial for us to develop a solution that is comprehensible and interpretable for clinician and more conducive to the previous common diagnosis process. The classifiers in Table 3.1 that are most compatible with clinical practise include Random Forest, Decision Tree, and KNN. Figure 3.7 shows the interpretability of a decision tree. KNN and Decision Tree, on the other hand, do not provide accuracy results as high as Random Forest. The best classifier, according to this experiment, is random forest, which has a high accuracy (very comparable and competitive with others) and strong interpretability.

Table 3.1: The table below shows the accuracy of various classifiers applied to our cleaned dataset. The dataset contained 130 rows and 43 characteristics. The Ridge classifier has the best performance (91.5 %), SVC, Ridge CV, BernouliNB, and NuSVC are second with 90.8% accuracy, and Random Forest is third with 90.5%. A 0.3 difference, on the other hand, is not statistically significant and could be due to randomness.

Classifier	Accuracy
Extra Tree	89.3
Random Forest	**90.5**
XG Boosting	90.0
Gradient Boosting	87.6
KNN	86.9
SVC	**90.8**
Perceptron	87.7
Decision Tree	81.1
MLP	90.0
SGD	86.6
Ridge CV	**90.8**
Ridge	**91.5**
Passive Aggressive	87.4
Gaussian Process	86.9
Ada Boost	86.2
Bagging	85.9
BernoulliNB	**90.8**
Calibrated CV	88.5
GaussianNB	80.0
Label Propagation	84.6
Label Spreading	84.6
Linear Discriminant Analysis	88.5
Logistic Regression	89.2
NuSVC	**90.8**

3.5.2 Classification with 28 clinically important features

Feature selection by expert is a profound way of feature selection. Despite the fact that all of the classifiers in the previous section 3.5.1 performed well, using domain knowledge is an essential part of data cleaning. Since this project is based on a real-world problem, it is critical that we incorporate medical knowledge into our own practise in order to produce more reliable results. As a result, we consulted with the clinical professionals at the Mayo Clinic to apply this knowledge, and after that we deleted several features that were not clinically important. For instance, based on the recommendations of clinic professionals, we deleted *albumin* features from the dataset. Only 28 of the 52 clinically significant features were left after this process. The Ridge, Ridge CV, and Random Forest classifiers performed better than other classifiers, same as in the previous section 3.5.1. However, there are some aspects to which we should pay particular attention. First and foremost, we can see an improvement in the result compared to the previous section 3.5.1 when we did not remove the non-clinically significant features. This outcome demonstrates how using domain expertise aided us in enhancing our performance. Another aspect worth mentioning is that whereas linear discriminant analysis' accuracy in the previous section 3.5.1 was 88.5, it increased to 93.3 in this section. However, compared to the preceding section 3.5.1, KNN and SVC performance drastically decreased.

Table 3.2: The table below displays the accuracy of different classifiers used on our cleaned dataset, which only contains clinically significant features. There were 130 rows in the dataset, along with 28 features. Ridge and Ridge CV classifiers with a performance of 93.3% are the top classifiers, followed by Random Forest with a performance of 92.3%. Contrary to the results of the prior experiment (See in Table 3.1), some classifiers in this experiment perform poorly and have accuracy below 70 %.

Classifier	Accuracy
Extra Tree	90.8
Random Forest	**92.3**
XG Boosting	90.0
Gradient Boosting	91.3
KNN	66.9
SVC	66.9
Perceptron	75.3
Decision Tree	82.4
MLP	84.7
SGD	77.6
Ridge CV	**93.3**
Ridge	**93.3**
Passive Aggressive	79.8
Gaussian Process	69.1
Ada Boost	91.0
Bagging	89.8
BernoulliNB	91.6
Calibrated CV	90.4
GaussianNB	88.8
Label Propagation	69.7
Label Spreading	69.7
Linear Discriminant Analysis	**93.3**
Logistic Regression	92.7
NuSVC	92.7

3.5.3 Feature Selection

Techniques for feature selection have a capacity to identify the features that are most important for classification. In this study, the classification performance is enhanced by the feature selection.

For feature selection, we utilised four distinct methods:

- Random Forest (RF)
- Decision Tree (DT)
- XGBoosting (XGB)
- Recursive Feature Elimination (RFE)

Nonetheless, due to randomization, these methods produce a subset that differs from runtime to runtime. As a result, for each feature selection approach, we conducted 100 iterations of the experiment and calculated the most common features over all 100 iterations; the most frequent features were chosen as the final subset.

3.5.4 Classification with 20 features

In this section, first we selected the 20 most important features from the 43 that were cleaned in the section 3.3 section. After extracting the most important features, we trained the classifier using the new dataset which only contain the 20 most important features. The result of these classifiers on the extracted dataset reported in Table 3.3. Using the exact same procedure, 20 features out of 28 clinically significant features presented in the section 3.5.2 were extracted. The outcomes of all classifiers on these features are reported in Table 3.4.

Table 3.3: The classifier is displayed in the first column of this table, while the other four columns each indicate a different feature selection technique. We can determine the optimum feature selection method and classifier for our task by using the results of these classifiers on various feature selection methods. According to the results, the SVC classifier employing the XBG feature selection and Ridge CV classsifier using RFE feature selection approaches achieved the highest result (93.1 %), which is still inferior to the result shown in Table 3.2. However, it is noticeable that every outcome is higher than 80 %, which is really noteworthy.

Classifier	20 features out of 43 entire features			
	DT	RF	XGB	RFE
KNN	85.4	89.2	87.7	90.8
Random Forest	87.7	91.5	91.5	92.3
SVC	88.5	90.8	**93.1**	89.2
Ridge	90.8	90.0	90.0	92.3
BernoulliNB	89.2	92.3	90.8	92.3
Gradient Boosting	84.6	89.2	90.0	89.2
MLP	88.5	90.8	91.5	92.8
Extra Tree	85.4	89.2	91.5	90.8
XG Boosting	89.2	**93.1**	92.3	90.0
Perceptron	86.9	90.0	85.4	90.0
Decision Tree	85.4	84.6	81.5	83.1
SGD	90.0	83.1	88.5	87.7
RidgeCV	**91.5**	90.0	90.8	**93.1**
Passive Aggressive	85.4	88.5	86.2	87.7
Gaussian Process	84.6	90.8	92.3	90.0
Ada Boost	89.2	89.2	90.0	86.2
Bagging	85.4	83.8	86.9	86.2
Calibrated CV	90.0	89.2	88.5	88.5
GaussianNB	88.5	87.7	87.7	90.0
Label Propagation	86.9	83.1	85.4	85.4
Label Spreading	87.7	83.1	86.2	85.4
LDA	91.5	91.5	90.8	90.0
Logistic Regression	89.2	90.0	90.8	92.3
NuSVC	89.2	90.0	92.3	91.5

Table 3.4: In this table, the first column represents the classifier, and the remaining four columns represent various feature selection methods. According to the results, the Ridge CV classifier using the DT feature selection strategy produced the highest result (94.4 %), which is the best result thus far. In spite of the low accuracy in the results below (which are similar to Table 3.2), it is noteworthy that every result in the RFE column is higher than 80, and in most situations, the classifier's best performance is in the RFE selection technique. This demonstrates the reliability of the RFE method for choosing critical features.

Classifier	20 features out of 28 clinically important features			
	DT	RF	XGB	RFE
KNN	69.1	69.7	69.1	86.0
Random Forest	92.7	91.6	**93.8**	91.0
SVC	66.9	66.9	66.9	**90.4**
Ridge	92.7	92.7	92.7	92.1
BernoulliNB	93.8	92.7	92.7	91.6
Gradient Boosting	91.6	89.9	90.4	89.3
MLP	83.1	83.7	84.3	91.6
Extra Tree	91.0	91.0	91.0	88.8
XG Boosting	92.1	92.1	89.9	88.8
Perceptron	68.0	68.5	74.7	**91.0**
Decision Tree	83.7	79.8	82.0	84.8
SGD	76.4	72.5	73.0	91.6
RidgeCV	**94.4**	92.7	92.7	92.1
Passive Aggressive	79.8	84.3	79.8	90.4
Gaussian Process	69.7	68.0	70.8	92.1
Ada Boost	91.6	89.9	91.6	90.4
Bagging	89.3	89.9	92.1	88.8
Calibrated CV	93.3	91.0	91.6	93.3
GaussianNB	92.7	90.4	91.0	90.4
Label Propagation	68.5	67.4	66.9	84.8
Label Spreading	68.5	66.9	66.9	85.4
LDA	92.7	92.7	92.7	92.1
Logistic Regression	92.1	92.1	92.1	93.8
NuSVC	92.7	92.7	91.6	92.7

3.5.5 Classification with 10 features

This section is similar to the preceding section 3.5.4, with the exception that we have only included the top 10 features here rather than 20. Reducing the number of features might be advantageous in general, but in our situation it is especially crucial because collecting medical data is a laborious task, and fewer features can reduce the amount of effort we must do to obtain data.

In this part, at first we chose the 10 most important features ones from the 43 features that were cleaned in the section 3.3. Then, we select 10 clinically significant features from a total of 28 in section 3.5.2. In Table 3.5, and Table 3.6, the results of all classifiers on these features are displayed. As stated in Table 3.6 the highest result (94.9 %) was achieved by the BernouliNB classifier using the RFE feature selection approach. This result demonstrates the value of employing domain knowledge when choosing features, as well as demonstrates how having fewer features can be advantageous since it simplifies the task of classification. Furthermore, having fewer features allows us to visualise data more easily which is quite important. On the other hand, it is important for medical professionals because collecting data is a time-consuming process.

Table 3.5: In the following table, the first column indicates the classifier, and the next four columns reflect various methods of feature selection. Based on the results of this experiment, the Gaussian Process classifier with the XGB feature selection method produced the best results (94.6 %). Following that, the SVC classifier with RFE earned the best performance in this experiment, with 93 % accuracy. In addition, as you can see, most of the classifiers perform well on this data, indicating that the selected features are accurate representations of the data and are capable of separating it.

Classifier	10 features out of 43 entire features			
	DT	RF	XGB	RFE
KNN	**90.8**	**90.0**	89.2	90.8
Random Forest	89.2	87.7	91.5	90.0
SVC	87.7	90.0	92.3	**93.1**
Ridge	87.7	89.2	90.0	90.0
BernoulliNB	**90.8**	**90.0**	92.3	91.5
Gradient Boosting	88.5	87.7	92.3	89.2
MLP	90.8	89.2	92.3	90.8
Extra Tree	87.7	87.7	90.0	90.0
XG Boosting	88.5	89.2	92.3	91.5
Perceptron	86.2	88.5	90.8	90.8
Decision Tree	86.9	83.8	89.2	89.2
SGD	83.8	87.7	88.5	89.2
RidgeCV	87.7	89.2	91.5	88.5
Passive Aggressive	84.6	86.9	86.2	86.2
Gaussian Process	90.0	90.0	**94.6**	92.3
Ada Boost	87.7	88.5	90.8	85.4
Bagging	82.3	88.5	89.2	83.8
Calibrated CV	86.9	89.2	91.5	90.8
GaussianNB	89.2	87.7	89.2	88.5
Label Propagation	90.8	86.2	90.0	90.8
Label Spreading	91.5	86.2	90.8	90.8
LDA	88.5	89.2	90.0	90.0
Logistic Regression	90.0	89.2	91.5	90.8
NuSVC	89.2	87.7	92.3	90.0

Table 3.6: The classifier is shown in the first column of the following table, and the feature selection techniques are shown in the following four columns. Based on the outcomes of this experiment, the best result (94.9 %), which is also the highest results so far, were achieved by the BernouliNB classifier using the RFE feature selection approach. However, we should keep in mind that the difference between this result and the one obtained in Table 3.5 is not statically important. We may therefore continue with a classifier that provides decent results and can be interpreted by clinical professionals.

Classifier	10 features out of 28 clinically important features			
	DT	RF	XGB	RFE
KNN	70.2	71.3	88.2	89.3
Random Forest	92.1	92.7	91.6	89.3
SVC	65.2	65.7	91.0	91.0
Ridge	90.4	91.6	92.1	**93.3**
BernoulliNB	92.7	93.3	94.4	**94.9**
Gradient Boosting	90.4	91.6	91.6	92.1
MLP	82.0	82.0	92.1	93.3
Extra Tree	89.9	91.0	90.4	88.8
XG Boosting	90.4	92.1	91.6	91.6
Perceptron	62.9	68.5	89.9	92.1
Decision Tree	83.1	82.0	88.2	87.1
SGD	66.9	72.5	90.4	88.8
RidgeCV	92.1	92.7	92.7	93.8
Passive Aggressive	71.3	79.2	89.9	92.7
Gaussian Process	69.1	68.5	92.7	93.8
Ada Boost	90.4	88.2	92.1	92.7
Bagging	87.6	88.2	87.6	90.4
Calibrated CV	90.4	93.3	92.1	92.7
GaussianNB	92.1	92.1	89.3	92.7
Label Propagation	66.3	68.0	91.0	89.9
Label Spreading	66.3	68.0	91.0	90.4
LDA	90.4	91.6	92.1	93.3
Logistic Regression	92.1	**94.4**	93.8	93.8
NuSVC	91.0	89.3	92.1	93.8

3.6 Visualization

The visualisation of high-dimensional data is a critical subject in many different disciplines, and it deals with data of various dimensionality. For example, in the dataset that we use for IBD and HS classification, we often have more than 30 features.

We can use dimensionality reduction methods to visualise this dataset with such a high dimension. These methods convert the high-dimensional dataset $X = x_1, x_2, ..., x_n$ to two or three-dimensional data $Y = y_1, y_2, ..., y_n$. The low dimensional Y represents as a map, and each element of Y which are y_i $(y_1, y_2, ..., y_n)$ represent the individual datapoints presented in a scatterplot [97]. Several solutions to this problem have been presented, each with a different type of structure preserved. Traditional methods for reducing dimensionality such as Principal Components Analysis [46] and conventional multidimensional scaling [96], are linear procedures that emphasise on maintaining dissimilar datapoints' low-dimensional representations as far apart as possible. It is usually more critical for high-dimensional data that lies on or near a low-dimensional, non-linear manifold to keep the low-dimensional representations of extremely comparable datapoints close together, which is normally not attainable with a linear mapping.

In this section, we desire to see how separable our dataset is, but due to its high dimension, it is impossible for us to display it simply. To overcome this issue, we attempt to reduce dimension so that the low-dimensional map retains as much of the significant structure of the high-dimensional data as possible. We perform PCA and TSNE [97] algorithms on our IBD/HS dataset, and the results are shown in Figures 3.8, 3.9, and 3.10.

We can plot TSNE and PCA visulization for the best feature selection approach in 10 features (Section 3.5.5) to ensure that our feature selection is advantageous. Figure 3.11 depicts the TSNE visualization outcome of selecting 10 features using RFE feature. Based on the result in section 3.5.5 the best result was achieved by Bernouli NB classifier using RFE feature selection.The PCA visualisation in Figure 3.12, like the TSNE visualisation, confirms that our feature selection method was effective.

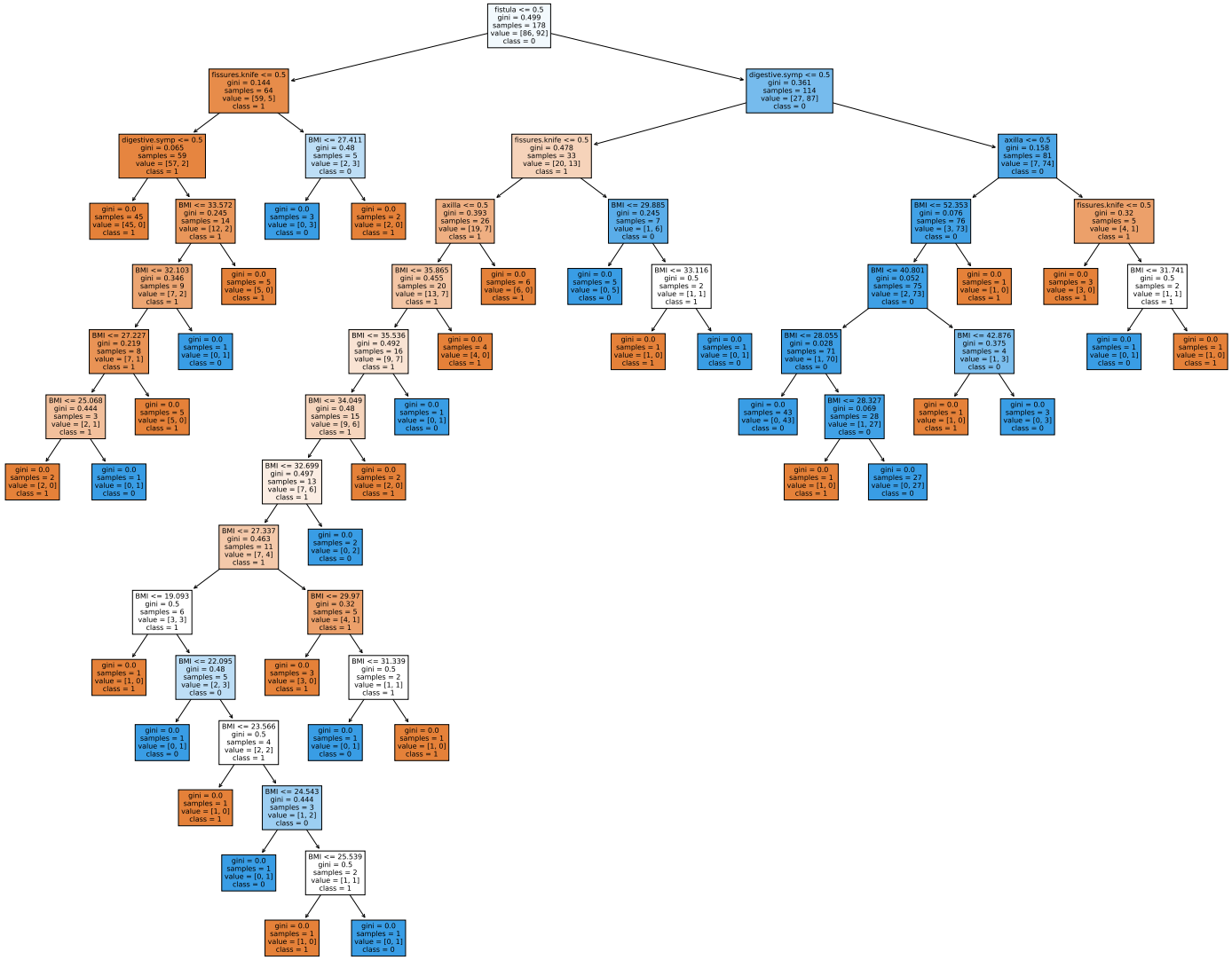


Figure 3.7: Each level has a unique feature, as can be seen in the figure below. We may determine the class associated to our datapoint by traversing this tree.

Figure 3.8: The figure below depicts the outcome of applying TSNE to our dataset. The blue dots represent IBD patients, while the red dots represent HS patients. As illustrated in the figure below, the dataset is clearly separable, and we may draw a line to split these two classes and achieve high classification accuracy. Indeed, this strategy assists us in ensuring that our results which are presented in classification section 3.5 are acceptable. However, there are several data points that appear to be outliers. For example, the blue dot in the lower left corner of the figure (about (-10, -7.5)) is relatively distant from its own group. This data point was shown to Mayo Clinic professionals for additional analysis.

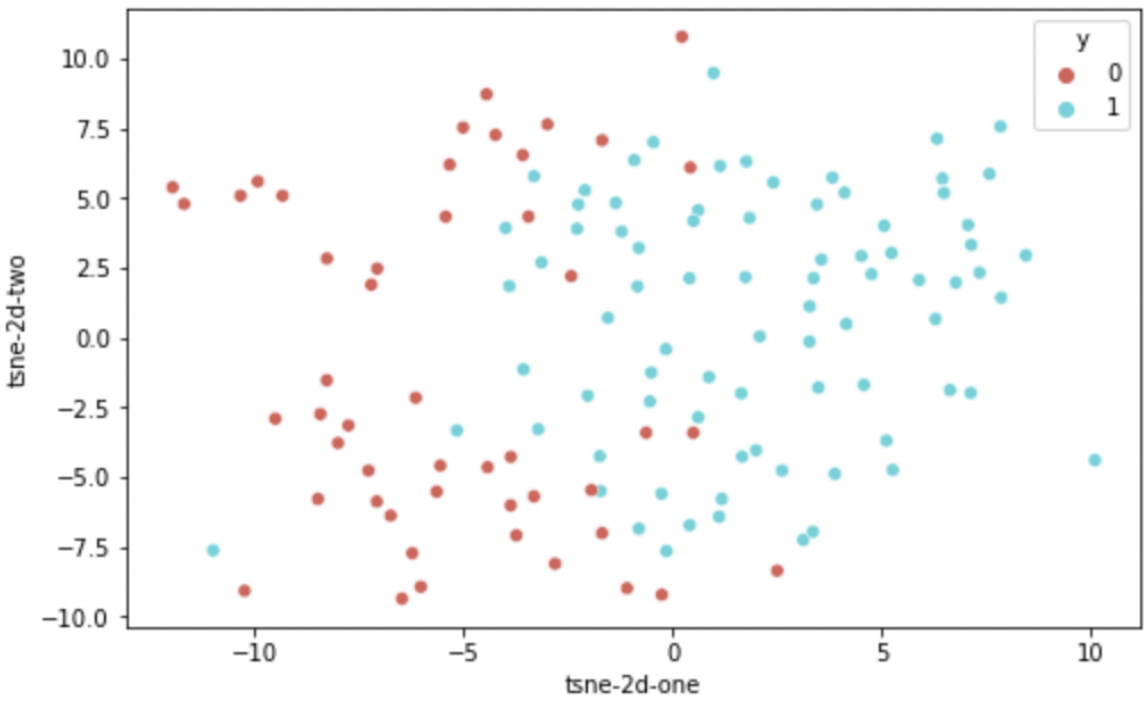


Figure 3.9: The results of using 2D PCA to analyse our dataset are shown in the figure below. IBD patients are represented by blue dots, while HS patients are represented by red dots. As seen in the figure below, the dataset is separable, but it is less separable when compared to TSNE (See in Figure 3.8). This is because PCA applies a linear transformation, but TSNE performs a non-linear transformation and it can capture much of the high-dimensional data's local structure while also exposing global structure.

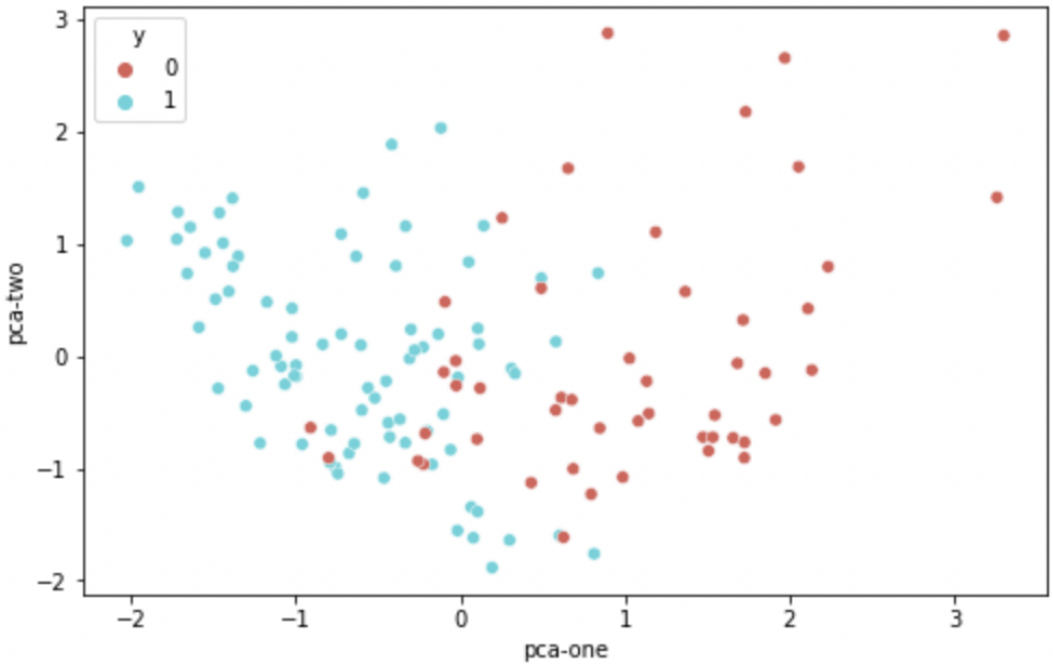


Figure 3.10: The results of using 3D PCA to analyse our dataset are shown in the figure below. IBD patients are represented by blue dots, while HS patients are represented by red dots. There are situations when a dataset appears to be non-separable in two dimensions but is actually separable in higher dimensions. We are able to notice that the dataset appears to be more separable owing to the 3D visualisation. Therefore, in higher dimensions, we may get more accuracy.

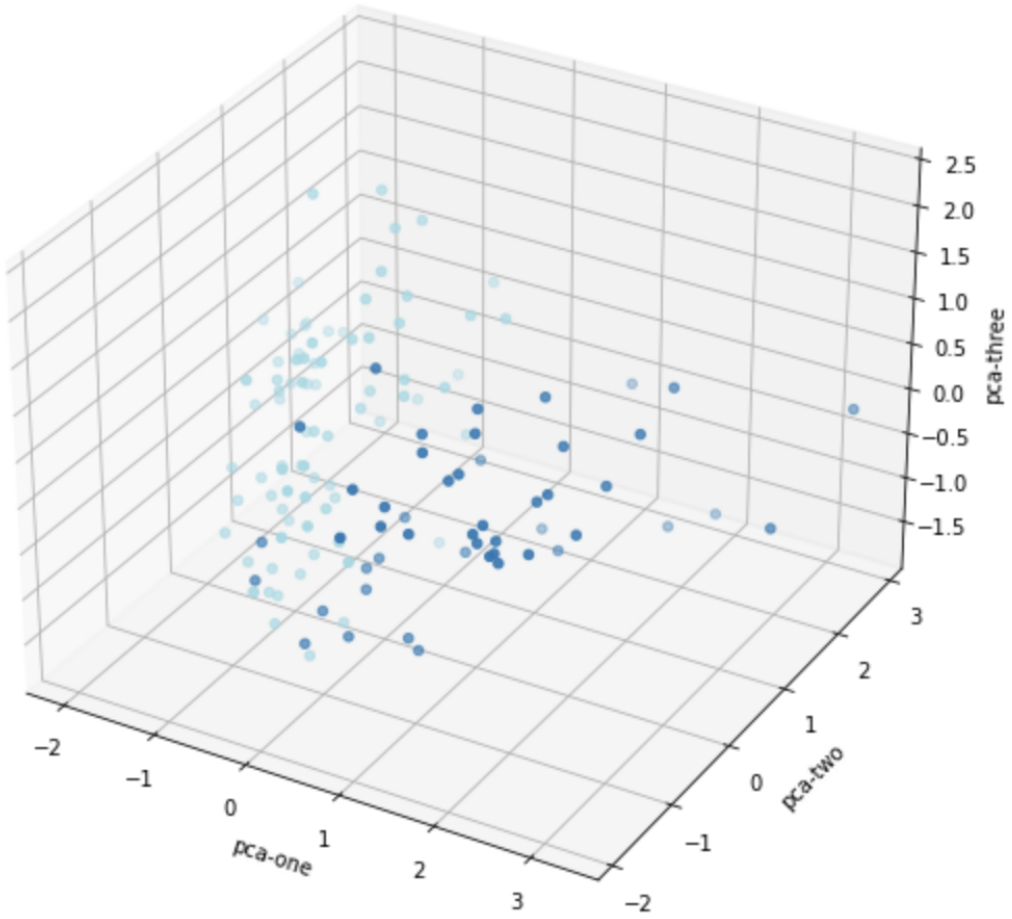


Figure 3.11: The result of applying TSNE to the dataset with 10 important features selected by RFE is depicted in the figure below. IBD patients are represented by blue dots, while HS patients are represented by red dots. The dataset is clearly separable, as demonstrated in the figure below, and we may draw a line to separate these two classes and achieve high classification accuracy. It demonstrates that our feature selection strategy was effective.

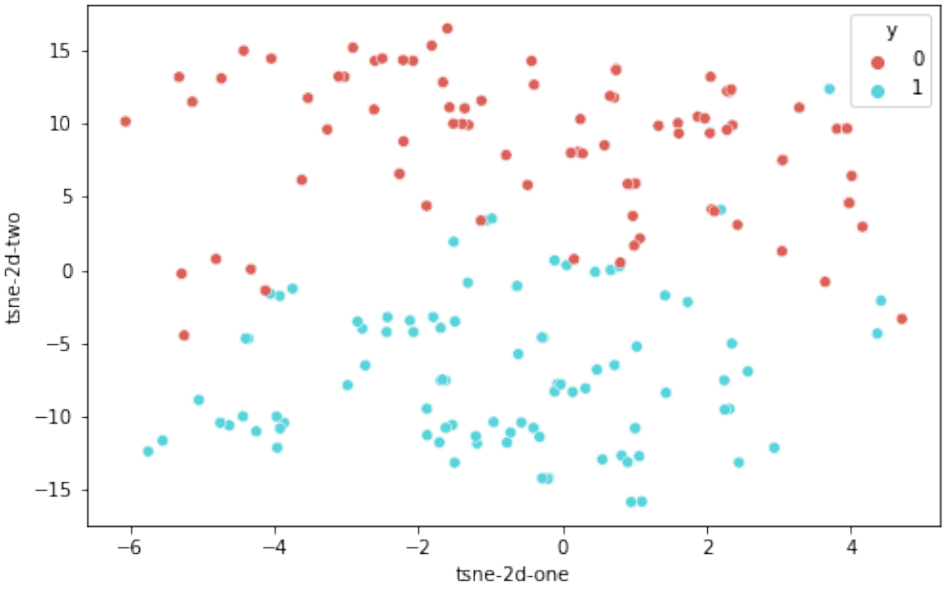
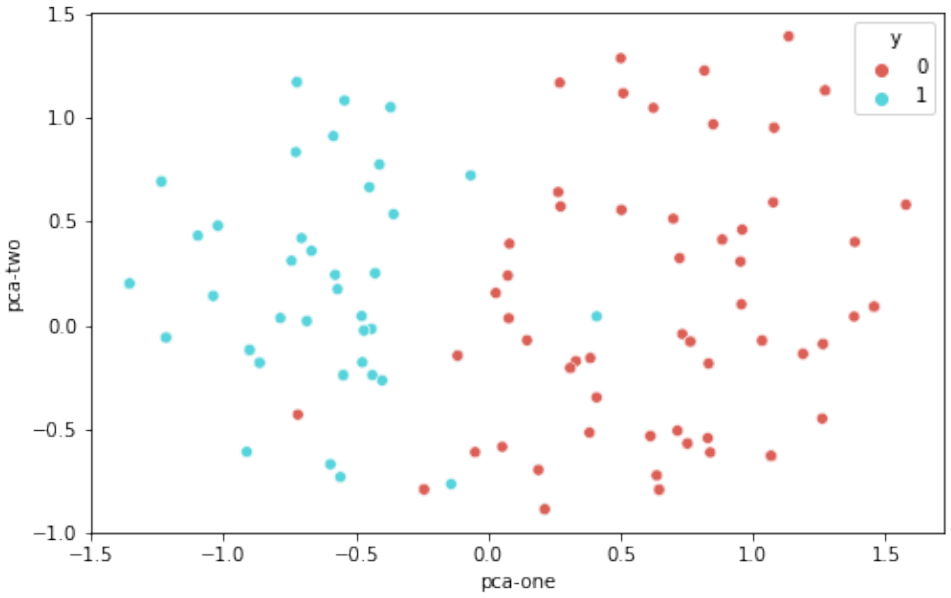


Figure 3.12: The figure below illustrates the outcome of applying PCA to the dataset with 10 important features chosen by RFE. Patients with IBD are shown by blue dots, whereas those with HS are represented by red dots. This figure, indicates the effectiveness of our feature selection technique.



Chapter 4

Summary and Conclusion

4.1 Summary

The objective of this thesis was to use machine learning and data mining techniques to classify two diseases, HS and IBD. Several algorithms were used in this work, including Decision Trees, Random Forest, and k nearest neighbours. To determine the most important features, however, various feature selection strategies have been applied including both computer-based and expert knowledge-based techniques. Feature selection is important for lowering the dimensionality of the problem and making classification easier. It is also useful for clinicians to determine the importance of different features.

Extensive experiments were conducted to determine the method with the best accuracy and with the highest level of interpretability. Approximately 24 different classifiers were trained in this thesis. Experiments started by training the classifiers on a dataset with 43 features. Then, classifiers were trained on a dataset including 29 clinically significant features. Following that, four different feature selection methods were used to pick the 20 most important features and subsequently the 10 most important features. In total, over 240 different experiments were conducted to determine the best result.

Among all experiments, Bernouli NB with 10 selected features using RFE had the highest accuracy (94.9 %); nevertheless, as previously indicated, there are other factors that require our attention aside from accuracy. Explainability of a decision-making process is an important aspect of machine decision making in the medical field. Physicians need to know why a specific disease was chosen as a diagnosis, so the Random Forest technique was used as our primary method since it is more similar to the clinician's decision process.

To the best of our knowledge, this was the first study to use machine learning and data mining approaches to classify IBD and HS.

4.2 Conclusion

Among all experiments that used a Random Forest classifier, the Random Forest classifier with 20 features using he XGB feature selection approach generated the highest accuracy of 93.8 %. Overall, experiments on the IBD and HS dataset provided by medical specialists at the Mayo Clinic indicated high accuracy in disease classification. Meanwhile, extensive feature selection strategies have been used to assist the physician in finding the most significant features. The combination of classification and feature selection seems to be the most promising approach to establsih a computerized decision-making support system.

4.3 Future Works

As discussed previously in the chapter *Methodology*, a part of the data had been derived from MR images. However, the clinician extracts these data based on visual inspection. This means that the clinician will fill out a list of features based on what he or she observes in the MR images. Nonetheless, some data may have been missed due to the subjective nature of visual inspections. One obvious issue is that for some features, the clinician only checks whether or not he or she observes that feature; nevertheless, by processing the image itself, there are many additional aspects, such as slice thickness, that may need to be considered. In addition to these kinds of characteristics, there may also exist a collection of disguised characteristics that can aid in distinguishing between HS and IBD, visual clues and subtleties that can be identified by machine learning. Working on MR images to classify these two diseases is a promising technique that could yield new insights and better results.

References

[1] 4 techniques to deal with missing data in datasets — by egor howell — towards datascience. https://towardsdatascience.com/4-techniques-to-deal-with-missing-data-in-datasets-841f8a303395. (Accessed on 11/11/2022).

[2] Artificial intelligence in health care — gao. https://www.gao.gov/products/gao-22-104629. (Accessed on 11/11/2022).

[3] Crohn's disease — hopkins medicine. https://www.hopkinsmedicine.org/health/conditions-and-diseases/crohns-disease. (Accessed on 11/11/2022).

[4] Hidradenitis suppurativa (hs) — by mayo clinic — mayo clinic. https://www.mayoclinic.org/diseases-conditions/hidradenitis-suppurativa/symptoms-causes/syc-20352306. (Accessed on 11/11/2022).

[5] How to treat missing values — hopkins medicine — by mohit sharma — data science beginners. https://datasciencebeginners.com/2018/11/06/09-how-to-treat-missing-values/. (Accessed on 11/11/2022).

[6] Inflammatory bowel disease (ibd) — by mayo clinic — mayo clinic. https://www.mayoclinic.org/diseases-conditions/inflammatory-bowel-disease/symptoms-causes/syc-20353315. (Accessed on 11/11/2022).

[7] Inflammatory bowel disease (ibd) — hopkins medicine. https://www.hopkinsmedicine.org/health/conditions-and-diseases/inflammatory-bowel-disease. (Accessed on 11/11/2022).

[8] Loocv for evaluating machine learning algorithms— by jason brownlee— machine learning mastery. https://machinelearningmastery.com/loocv-for-evaluating-machine-learning-algorithms/. (Accessed on 11/11/2022).

[9] Machine learning's potential to improve medical diagnosis — gao. https://www.gao.gov/blog/machine-learnings-potential-improve-medical-diagnosis#:~:text=How%20could%20machine%20learning%20affect,learning%20could%20detect%20diseases%20earlier. (Accessed on 11/11/2022).

[10] Naive bayes — by scikit learn developers — scikit learn. https://scikit-learn.org/stable/modules/naive_bayes.html. (Accessed on 11/11/2022).

[11] Training-validation-test split and cross-validation done right— by adrian tam— machine learning mastery. https://machinelearningmastery.com/training-validation-test-split-and-cross-validation-done-right/. (Accessed on 11/11/2022).

[12] Understanding random forest — by tony yiu — towards data science. https://towardsdatascience.com/understanding-random-forest-58381e0602d2. (Accessed on 11/11/2022).

[13] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[14] Andrew L Beam and Isaac S Kohane. Big data and machine learning in health care. *Jama*, 319(13):1317–1318, 2018.

[15] Michael Berks, Zezhi Chen, Sue Astley, and Chris Taylor. Detecting and classifying linear structures in mammograms using random forests. In *Biennial International Conference on Information Processing in Medical Imaging*, pages 510–524. Springer, 2011.

[16] Kaustubh Arun Bhavsar, Jimmy Singla, Yasser D Al-Otaibi, Oh-Young Song, Yousaf Bin Zikria, and Ali Kashif Bashir. Medical diagnosis using machine learning: a statistical review. *Computers, Materials and Continua*, 67(1):107–125, 2021.

[17] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[18] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[19] Leo Breiman. Consistency for a simple model of random forests. 2004.

[20] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone. *Classification and regression trees*. Routledge, 2017.

[21] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998.

[22] Chih-Wen Chen, Yi-Hong Tsai, Fang-Rong Chang, and Wei-Chao Lin. Ensemble feature selection in medical datasets: Combining filter, wrapper, and embedded feature selection results. *Expert Systems*, 37(5):e12553, 2020.

[23] Xue-wen Chen and Jong Cheol Jeong. Enhanced recursive feature elimination. In *Sixth International Conference on Machine Learning and Applications (ICMLA 2007)*, pages 429–435. IEEE, 2007.

[24] Seok-Hwan Choi, Jin-Myeong Shin, and Yoon-Ho Choi. Dynamic nonparametric random forest using covariance. *Security and Communication Networks*, 2019, 2019.

[25] Xu Chu, Ihab F Ilyas, Sanjay Krishnan, and Jiannan Wang. Data cleaning: Overview and emerging challenges. In *Proceedings of the 2016 international conference on management of data*, pages 2201–2206, 2016.

[26] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

[27] Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. Online passive aggressive algorithms. 2006.

[28] Belur V Dasarathy. Nearest neighbor (nn) norms: Nn pattern classification techniques. *IEEE Computer Society Tutorial*, 1991.

[29] Manoranjan Dash and Huan Liu. Feature selection for classification. *Intelligent data analysis*, 1(1-4):131–156, 1997.

[30] Ramon Lopez De Mantaras and Eva Armengol. Machine learning from examples: Inductive and lazy methods. *Data & Knowledge Engineering*, 25(1-2):99–123, 1998.

[31] Peijun Du, Alim Samat, Björn Waske, Sicong Liu, and Zhenhong Li. Random forest and rotation forest for fully polarized sar image classification using polarimetric and spatial features. *ISPRS Journal of Photogrammetry and Remote Sensing*, 105:38–53, 2015.

[32] Jennifer G Dy and Carla E Brodley. Feature selection for unsupervised learning. *Journal of machine learning research*, 5(Aug):845–889, 2004.

[33] Yoav Freund, Robert E Schapire, et al. Experiments with a new boosting algorithm. In *icml*, volume 96, pages 148–156. Citeseer, 1996.

[34] Lewis Frey, Douglas Fisher, Ioannis Tsamardinos, Constantin F Aliferis, and Alexander Statnikov. Identifying markov blankets with decision tree induction. In *Third IEEE International Conference on Data Mining*, pages 59–66. IEEE, 2003.

[35] Matt W Gardner and SR Dorling. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric environment*, 32(14-15):2627–2636, 1998.

[36] Amit Garg, Jessica Hundal, and Andrew Strunk. Overall and subgroup prevalence of crohn disease among patients with hidradenitis suppurativa: a population-based analysis in the united states. *JAMA dermatology*, 154(7):814–818, 2018.

[37] Arunim Garg and Vijay Mago. Role of machine learning in medical research: A survey. *Computer Science Review*, 40:100370, 2021.

[38] Ezequiel Geremia, Olivier Clatz, Bjoern H Menze, Ender Konukoglu, Antonio Criminisi, and Nicholas Ayache. Spatial decision forests for ms lesion segmentation in multi-channel magnetic resonance images. *NeuroImage*, 57(2):378–390, 2011.

[39] Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine learning*, 63(1):3–42, 2006.

[40] Michael Goetz, Christian Weber, Josiah Bloecher, Bram Stieltjes, Hans-Peter Meinzer, and Klaus Maier-Hein. Extremely randomized trees based brain tumor segmentation. *Proceeding of BRATS challenge-MICCAI*, pages 006–011, 2014.

[41] Daniel A Gold, Cynthia Nicholson, Gordon Jacobsen, and Iltefat H Hamzavi. International classification of diseases–based analysis is inaccurate in assessing the prevalence of inflammatory bowel disease in patients with hidradenitis suppurativa. *Journal of the American Academy of Dermatology*, 85(2):495–497, 2021.

[42] Ryan Daniel Gotesman, Charles Choi, and Afsaneh Alavi. Hidradenitis suppurativa in east and southeast asian populations: a systematic review and meta-analysis. *International Journal of Dermatology*, 60(11):e433–e439, 2021.

[43] Peter Groves, Basel Kayyali, David Knott, and Steve Van Kuiken. The'big data'revolution in healthcare: Accelerating value and innovation. 2016.

[44] Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1):389–422, 2002.

[45] Yahong Han, Yi Yang, Yan Yan, Zhigang Ma, Nicu Sebe, and Xiaofang Zhou. Semisupervised feature selection via spline regression for video semantic recognition. *IEEE Transactions on Neural Networks and Learning Systems*, 26(2):252–264, 2014.

[46] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933.

[47] Chung-Ho Hsieh, Ruey-Hwa Lu, Nai-Hsin Lee, Wen-Ta Chiu, Min-Huei Hsu, and Yu-Chuan Jack Li. Novel solutions for an old disease: diagnosis of acute appendicitis with random forest, support vector machines, and artificial neural networks. *Surgery*, 149(1):87–93, 2011.

[48] Earl B Hunt, Janet Marin, and Philip J Stone. Experiments in induction. 1966.

[49] Gregor BE Jemec. Hidradenitis suppurativa. *New England Journal of Medicine*, 366(2):158–164, 2012.

[50] George H John, Ron Kohavi, and Karl Pfleger. Irrelevant features and the subset selection problem. In *Machine learning proceedings 1994*, pages 121–129. Elsevier, 1994.

[51] Astrid-Helene Ravn Jørgensen, Simon Francis Thomsen, Katrine Elisabeth Karmisholt, and Hans Christian Ring. Clinical, microbiological, immunological and imaging characteristics of tunnels and fistulas in hidradenitis suppurativa and crohn's disease. *Experimental Dermatology*, 29(2):118–123, 2020.

[52] Hyun Kang. The prevention and handling of the missing data. *Korean journal of anesthesiology*, 64(5):402–406, 2013.

[53] Dhruv Khullar, Ashish K Jha, and Anupam B Jena. Reducing diagnostic errors—why now? *The New England journal of medicine*, 373(26):2491, 2015.

[54] Seong-Hoon Kim, Ji-Hyun Lee, Byoungchul Ko, and Jae-Yeal Nam. X-ray image classification using random forests with local binary patterns. In *2010 International Conference on Machine Learning and Cybernetics*, volume 6, pages 3190–3194. IEEE, 2010.

[55] Daphne Koller and Mehran Sahami. Toward optimal feature selection. Technical report, Stanford InfoLab, 1996.

[56] Igor Kononenko. Estimating attributes: Analysis and extensions of relief. In *European conference on machine learning*, pages 171–182. Springer, 1994.

[57] Igor Kononenko. Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in medicine*, 23(1):89–109, 2001.

[58] Sotiris B Kotsiantis. Decision trees: a recent overview. *Artificial Intelligence Review*, 39(4):261–283, 2013.

[59] Sotiris B Kotsiantis, Ioannis D Zaharakis, and Panayiotis E Pintelas. Machine learning: a review of classification and combining techniques. *Artificial Intelligence Review*, 26(3):159–190, 2006.

[60] Jan Lapins, Weimin Ye, Olof Nyrén, and Lennart Emtestam. Incidence of cancer among patients with hidradenitis suppurativa. *Archives of dermatology*, 137(6):730–734, 2001.

[61] Ji-Hyeon Lee, Deok-Yeon Kim, Byoung Chul Ko, and Jae-Yeal Nam. Keyword annotation of medical image with random forest classifier and confidence assigning. In *2011 Eighth International Conference Computer Graphics, Imaging and Visualization*, pages 156–159. IEEE, 2011.

[62] Victor Lempitsky, Michael Verhoek, J Alison Noble, and Andrew Blake. Random forest classification for automatic delineation of myocardium in real-time 3d echocardiography. In *International Conference on Functional Imaging and Modeling of the Heart*, pages 447–456. Springer, 2009.

[63] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P Trevino, Jiliang Tang, and Huan Liu. Feature selection: A data perspective. *ACM computing surveys (CSUR)*, 50(6):1–45, 2017.

[64] Tjen-Sien Lim, Wei-Yin Loh, and Yu-Shan Shih. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. *Machine learning*, 40(3):203–228, 2000.

[65] Huan Liu and Lei Yu. Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on knowledge and data engineering*, 17(4):491–502, 2005.

[66] Shuangge Ma and Jian Huang. Penalized feature selection and classification in bioinformatics. *Briefings in bioinformatics*, 9(5):392–403, 2008.

[67] Sadaf Malik, Nadia Kanwal, Mamoona Naveed Asghar, Mohammad Ali A Sadiq, Irfan Karamat, and Martin Fleury. Data driven approach for eye disease classification with machine learning. *Applied Sciences*, 9(14):2789, 2019.

[68] Bjoern H Menze, B Michael Kelm, Ralf Masuch, Uwe Himmelreich, Peter Bachert, Wolfgang Petrich, and Fred A Hamprecht. A comparison of random forest and its gini importance with standard chemometric methods for the feature selection and classification of spectral data. *BMC bioinformatics*, 10(1):1–16, 2009.

[69] D Douglas Miller and Eric W Brown. Artificial intelligence in medical practice: the question to the answer? *The American journal of medicine*, 131(2):129–133, 2018.

[70] Tom M Mitchell and Tom M Mitchell. *Machine learning*, volume 1. McGraw-hill New York, 1997.

[71] Pabitra Mitra, CA Murthy, and Sankar K. Pal. Unsupervised feature selection using feature similarity. *IEEE transactions on pattern analysis and machine intelligence*, 24(3):301–312, 2002.

[72] Sreerama K Murthy. Automatic construction of decision trees from data: A multi-disciplinary survey. *Data mining and knowledge discovery*, 2(4):345–389, 1998.

[73] Siew C Ng, Hai Yun Shi, Nima Hamidi, Fox E Underwood, Whitney Tang, Eric I Benchimol, Remo Panaccione, Subrata Ghosh, Justin CY Wu, Francis KL Chan, et al. Worldwide incidence and prevalence of inflammatory bowel disease in the 21st century: a systematic review of population-based studies. *The Lancet*, 390(10114):2769–2778, 2017.

[74] Thais Mayumi Oshiro, Pedro Santoro Perez, and José Augusto Baranauskas. How many trees in a random forest? In *International workshop on machine learning and data mining in pattern recognition*, pages 154–168. Springer, 2012.

[75] Mahesh Pal. Random forest classifier for remote sensing classification. *International journal of remote sensing*, 26(1):217–222, 2005.

[76] S Patro and Kishore Kumar Sahu. Normalization: A preprocessing stage. *arXiv preprint arXiv:1503.06462*, 2015.

[77] Kevin Phan, Artiene Tatian, Jane Woods, Geoffrey Cains, and John W Frew. Prevalence of inflammatory bowel disease (ibd) in hidradenitis suppurativa (hs): systematic review and adjusted meta-analysis. *International Journal of Dermatology*, 59(2):221–228, 2020.

[78] Vili Podgorelec, Peter Kokol, Bruno Stiglic, and Ivan Rozman. Decision trees: an overview and their use in medicine. *Journal of medical systems*, 26(5):445–463, 2002.

[79] Yanjun Qi. Random forest for bioinformatics. In *Ensemble machine learning*, pages 307–323. Springer, 2012.

[80] J Ross Quinlan. Discovering rules by induction from large collections of examples. *Expert systems in the micro electronics age*, 1979.

[81] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[82] J Ross Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.

[83] Erhard Rahm and Hong Hai Do. Data cleaning: Problems and current approaches. *IEEE Data Eng. Bull.*, 23(4):3–13, 2000.

[84] Victor Francisco Rodriguez-Galiano, Bardan Ghimire, John Rogan, Mario Chica-Olmo, and Juan Pedro Rigol-Sanchez. An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS journal of photogrammetry and remote sensing*, 67:93–104, 2012.

[85] Lior Rokach. Ensemble-based classifiers. *Artificial intelligence review*, 33(1):1–39, 2010.

[86] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[87] Muskaan Sachdeva, Asfandyar Mufti, Hiba Zaaroura, Abrahim Abduelmula, Rafael Paolo Lansang, Ahmed Bagit, and Raed Alhusayen. Squamous cell carcinoma arising within hidradenitis suppurativa: a literature review. *International Journal of Dermatology*, 60(11):e459–e465, 2021.

[88] Yvan Saeys, Inaki Inza, and Pedro Larranaga. A review of feature selection techniques in bioinformatics. *bioinformatics*, 23(19):2507–2517, 2007.

[89] Steven L Salzberg. C4. 5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993, 1994.

[90] Ditte Marie Lindhardt Saunte and Gregor Borut Ernst Jemec. Hidradenitis suppurativa: advances in diagnosis and treatment. *Jama*, 318(20):2019–2032, 2017.

[91] Maria C Schneeweiss, Julien Kirchgesner, Richard Wyss, Yinzhu Jin, Cassandra York, Joseph F Merola, Arash Mostaghimi, Jonathan I Silverberg, Sebastian Schneeweiss, and Robert J Glynn. Occurrence of inflammatory bowel disease in patients with chronic inflammatory skin diseases: a cohort study. *British Journal of Dermatology*, 187(5):692–703, 2022.

[92] David A Schwartz, Edward V Loftus Jr, William J Tremaine, Remo Panaccione, W Scott Harmsen, Alan R Zinsmeister, and William J Sandborn. The natural history of fistulizing crohn's disease in olmsted county, minnesota. *Gastroenterology*, 122(4):875–880, 2002.

[93] Olayemi Sokumbi, David O Hodge, Sophia A Ederaine, Afsaneh Alavi, and Ali Alikhan. Comorbid diseases of hidradenitis suppurativa: a 15-year population-based study in olmsted county, minnesota, usa, 2022.

[94] Le Song, Alex Smola, Arthur Gretton, Karsten M Borgwardt, and Justin Bedo. Supervised feature selection via dependence estimation. In *Proceedings of the 24th international conference on Machine learning*, pages 823–830, 2007.

[95] Jiliang Tang, Salem Alelyani, and Huan Liu. Feature selection for classification: A review. *Data classification: Algorithms and applications*, page 37, 2014.

[96] Warren S Torgerson. Multidimensional scaling: I. theory and method. *Psychometrika*, 17(4):401–419, 1952.

[97] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[98] Konstantinos Veropoulos, Colin Campbell, Nello Cristianini, et al. Controlling the sensitivity of support vector machines. In *Proceedings of the international joint conference on AI*, volume 55, page 60. Stockholm, 1999.

[99] Gang Wang, Jinxing Hao, Jian Ma, and Hongbing Jiang. A comparative assessment of ensemble learning for credit scoring. *Expert systems with applications*, 38(1):223–230, 2011.

[100] Jason Weston, André Elisseeff, Bernhard Schölkopf, and Mike Tipping. Use of the zero norm with linear models and kernel methods. *The Journal of Machine Learning Research*, 3:1439–1461, 2003.

[101] Zenglin Xu, Irwin King, Michael Rung-Tsong Lyu, and Rong Jin. Discriminative semi-supervised feature selection via manifold regularization. *IEEE Transactions on Neural networks*, 21(7):1033–1047, 2010.

[102] Siddhant Yadav, Siddharth Singh, Jithinraj Edakkanambeth Varayil, W Scott Harmsen, Alan R Zinsmeister, William J Tremaine, Mark Denis P Davis, David A Wetter, Jean-Frederic Colombel, and Edward V Loftus Jr. Hidradenitis suppurativa in patients with inflammatory bowel disease: a population-based cohort study in olmsted county, minnesota. *Clinical Gastroenterology and Hepatology*, 14(1):65–70, 2016.

[103] M Yaqub, P Mahon, MK Javaid, C Cooper, and JA Noble. Weighted voting in 3d random forest segmentation. *Proc. Medical Image Understanding and Analysis, Warwick, UK*, pages 261–266, 2010.

[104] Mohammad Yaqub, M Kassim Javaid, Cyrus Cooper, and J Alison Noble. Improving the classification accuracy of the classic rf method by intelligent feature selection and weighted voting of trees with application to medical image segmentation. In *International Workshop on Machine Learning in Medical Imaging*, pages 184–192. Springer, 2011.

[105] Zhao Yi, Antonio Criminisi, Jamie Shotton, and Andrew Blake. Discriminative, semantic segmentation of brain tissue in mr images. In *International conference on medical image computing and computer-assisted intervention*, pages 558–565. Springer, 2009.

[106] Ya-Hui Yu, Lisa M Bodnar, Maria M Brooks, Katherine P Himes, and Ashley I Naimi. Comparison of parametric and nonparametric estimators for the association between incident prepregnancy obesity and stillbirth in a population-based cohort study. *American journal of epidemiology*, 188(7):1328–1336, 2019.

[107] Zheng Zhao and Huan Liu. Semi-supervised feature selection via spectral analysis. In *Proceedings of the 2007 SIAM international conference on data mining*, pages 641–646. SIAM, 2007.

[108] Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. 2002.