Fourth Year Project

Finding New Ways to Analyse the Heart using Machine Learning

Ina Hanninger

Hertford College

Supervised by

Professor Vicente Grau

Co-supervised by

Dr Ladislav Valkovic

Department of Engineering Science

May 2020

Abstract

There is a need for healthcare systems to progress towards more preventative, rather than reactive, models of patient care in order to save lives and reduce costs. Most critically is in the case of cardiac disease, which is the leading cause of death globally, claiming 17.9 million lives per year. Novel developments in cardiac magnetic resonance spectroscopy for the quantification of myocardial energy metabolites has shown potential as an informative diagnostic tool for earlier detection, however more evidence is required to prove and further understand its clinical impact.

This report presents the application of machine learning classifiers to improve the predictive accuracy of heart disease diagnosis while simultaneously examining the informativeness of cardiac magnetic resonance spectroscopy as a diagnostic test. To progress away from black-box paradigms of machine learning, interpretable tree-based classification models have been applied alongside interpretability tools such as feature importances and SHAP values. Further emphasising explainability alongside predictive accuracy, Bayesian networks are explored for causal inference. With recent developments in structure learning algorithms which reformulate the NP-Hard combinatorial optimisation problem into a continuous one, Bayesian networks are learned from heart disease data sets to perform rudimentary causal inference techniques such as do-intervention.

Explainable machine learning models, such as Random Forest and Bayesian Network classifiers, have achieved 95.36% and 92.72% test accuracy for the classification of heart disease. These models have also contributed further evidence for a number of causal theories of cardiac pathology, including the informativeness of myocardial metabolite concentrations, supporting the efficacy of cardiac magnetic resonance spectroscopy as a potential diagnostic test.

Acknowledgements

First and foremost I would like to thank Professor Vicente Grau for his guidance and input in this project. I would also like to thank Dr. Ladislav Valkovic and Professor Oliver Rider for helping me understand these new concepts in cardiology and medical imaging.

Contents

Chapter 1

Introduction

1.1 Motivation

Cardiovascular disease is the number one cause of death worldwide. According to the World Health Organisation, this accounts for over 17.9 million lives lost per year, constituting 31% of all global deaths [11]. Identifying at-risk or early-onset cases to administer early treatment has been shown to reduce death rates, and is a primary action point for the NHS long term plan for heart disease. Unfortunately, many of the current diagnostic methods fail to reliably detect cases early enough. For instance, cardiac magnetic resonance (CMR) imaging has become the primary diagnostic tool, detecting morphological changes such as left ventricle mass (LVMass), end diastolic volume (LVEDV) and systolic volume (LVESV) and ejection fraction (LVEF). However, many of these metrics stay in the normal range during the early stages and sometimes even chronic stages of the disease [6]. Thus, there has been an impetus to find alternative tests for the onset of heart disease, such as studying the energy metabolism of the heart to detect abnormalities in how energy is used. One such method is CMR spectroscopy (CMRS), which can quantify concentrations of metabolic molecules in the heart such as phosphocreatine (PCr), adenosine triphosphate (ATP) and myocardial lipids. However the clinical impact of this technology has not yet been demonstrated or fully understood.

By implementing machine learning classification algorithms on patient data sets that include metrics from CMRS scans, this project aims to examine the efficacy of CMRS technology for the diagnosis of heart disease. On a broader scale, in response to the explainability dilemma which is often a criticism of artificial intelligence (AI) in medicine, this project also aims to employ a principled resolution to this problem by reviewing a range of interpretability and causal inference tools which can be applied to clinical machine learning. Through improved predictive accuracy and medical understanding it is hoped that healthcare systems will be able to progress towards more preventative, rather than reactive, models of patient care.

1.2 Background

1.2.1 Medical Theories of Heart Disease

The following section provides background to the 3 types of heart diseases included in the data sets of this project.

Heart Failure

Heart failure is a chronic and progressive condition where the heart muscle is unable to pump sufficient amounts of blood to meet the body's requirements for blood and oxygen. Affecting more than 26 million people worldwide, the condition has been classed as a global pandemic, bearing a 5 year mortality rate of 45-60% [9].

Heart failure may be caused by a range of conditions, including hypertension, cardiomyopathies, valvular and congenital diseases, and most commonly today, coronary heart disease. Some specific risk factor thresholds include having a systolic blood pressure $> 140mmHg$, diastolic blood pressure $>90mmHg$ and BMI $>30kg/m^2$ [45].

Like many heart diseases, heart failure is diagnosed via symptom classification systems, echocardiography or CMRI measuring LVEF, LVEDV and LVMass. A normal range is considered as above 50% for LVEF and below $97ml/m^2$ for LVEDV. However, this is not always an accurate diagnostic. According to Paulus et al. (2007), heart failure with preserved LVEF accounts for more than 50% of all cases [6]. Thus explains the low sensitivity, although high specificity, rates of diagnosis, motivating the need for improved and earlier diagnostic methods.

Aortic Stenosis

Aortic stenosis (AS), defined as the narrowing of the aortic valve, is the most prevalent of all valvular heart diseases. Once assumed to be a degenerative disease, it occurs at a frequency of 2% for people over 65, with risk factors including older age, being male, diabetes, hypertension, and hyperlipidemia[15]. Once the aortic orifice narrows by more than half of its usual $3cm^2$, progressively worse left ventricular pressure overload occurs. As a compensatory mechanism, the left ventricle suffers hypertrophy (thickening of muscle) to preserve normal LVEF. This however impairs coronary blood-reserves, reduces diastolic function, and often leads to heart failure [10].

Further studies in AS suggest connections to impaired energetics, some of which are reliably detected by CMRS. For instance, in a study of 28 patients suffering from AS, PCr/ATP ratios were significantly reduced compared to controls, with values of 1.45 ± 0.21 vs. 2 ± 0.25 [43]. For symptomatic cases, AS bears an extremely poor prognosis with a 5-year mortality rate of 50% [15]. However if detected early enough, there are treatments to prevent the worsening of the condition, such as valve replacement.

Mitral Regurgitation

Mitral regurgitation (MR) is the systolic blood flow reversal of the left ventricle back into the left atrium. This results in increased volume and pressure in the left atrium which can also cause increased pressure in pulmonary veins. In mild cases, symptoms may not be apparent but in more severe cases can lead to palpitations due to atrial fibrillation, and heart failure. Causes include untreated high blood pressure, old age, cardiomyopathy, or congenital heart disease [19].

1.2.2 Novel Methods in Cardiac Magnetic Resonance Spectroscopy

CMRS is a novel technique for the non-invasive study of cardiac metabolism. By applying a powerful magnetic field and radio frequency excitation, atomic nuclei produce resonance signals dependent on the surrounding intramolecular

magnetic fields, thus enabling both the identification and quantification of cardiac metabolites. The 3 most common nuclei being studied for this purpose are phosphorus-31, carbon-13, and hydrogen-1.

The focus of this project primarily concerns phosphorus-31 CMR (31P-CMRS). This method allows for the measurement of phosphocreatine (PCr), adenosine triphosphate (ATP), intracellular pH, inorganic phosphate (Pi), and flux through the creatine kinase reaction. These quantities provide a several insights to the energetic state of the heart.

Applications to cardiac pathology

The energetic state of the heart is governed primarily by the following reactions:

$$ATP + Cr \Longleftrightarrow PCr + ADP + H^{+} \tag{1.1}$$

$$ATP + H_2O \longrightarrow ADP + Pi + energy \tag{1.2}$$

The creatine kinase (CK) enzyme reaction shown by Equation 1.1 serves as a kind of intracellular shuttle for energy. It uses CK to catalyse high energy PCr and transport it from the mitochondria to the myofibrils where it can be converted to ATP. As shown in Equation 1.2, the ATP hydrolysis reaction then provides energy to the myofilaments, where it is used to contract the heart muscles. To ensure the continuity of this, ATP is then resynthesised in the mitochondria via oxidative phosphorylation.

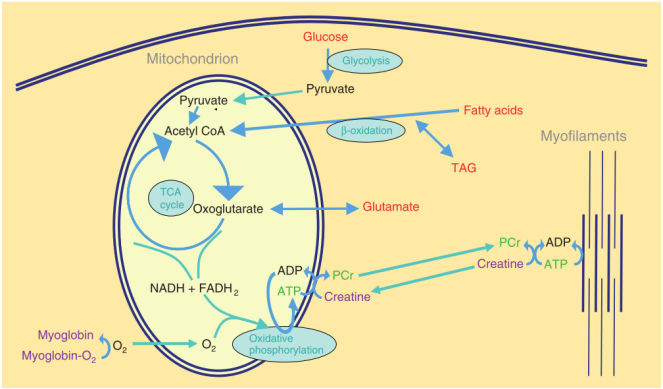


Figure 1.1: Schematic showing myocardial metabolism, with CK reactions and fatty acid oxidation, from [28]

However, when cardiac functions are disturbed – such as ischemia, heart failure or hypertrophy – the rate of ATP resynthesis cannot keep up with ATP hydrolysis and concentrations of ATP decrease. However, acting like a temporal buffer, PCr concentrations decrease at a more rapid rate than ATP. This explains why the overall PCr/ATP ratio decreases, and is quantifiable through 31P-CMRS.

Studies have been performed in vivo, in isolated animal hearts and in a few human studies demonstrating reduced PCr/ATP in heart failure and hypertrophy. For example in dilated cardiomyopathy patients, a correlation with the reduction in LVEF was found, even demonstrating it as a better predictor of long term survival than LVEF [25]. However, the full underlying causal mechanisms behind PCr/ATP are still unknown. It is unclear whether PCr/ATP reduction precedes heart disease and reduced LVEF or if it is an effect of it.

While current research is still sparse, another variable of interest is the **myocardial lipid content** which may be measured through 1H-CMRS (this is included in data set 2 for this project). Paolisso et al. found that myocardial lipid oxidation increased by 50%, and carbohydrate oxidation decreased by 60% in heart failure patients compared to healthy controls [62]. Schulze et al. also found increased lipid accumulation and lipotoxicity in heart disease, diabetes and obesity [57]. It is hoped that the mechanisms underlying myocardial lipids and PCr/ATP in these diseases can be better studied.

Oxford Centre for Magnetic Resonance Research (OCMR)

While CMRS holds potential for enhancing the clinical understanding of cardiac pathology, a limitation has been the low signal-to-noise ratio (SNR) of the PCr, ATP and Pi required for accurate quantification. This is also the reason why PCr/ATP is measured as a ratio instead of in absolute terms, and why studies in Pi and intracellular pH are currently very limited.

Fortunately various methods for enhancing the low SNR have been pioneered and implemented by the Oxford Centre for Clinical Magnetic Resonance Research (OCMR), a research group in the John Radcliffe hospital founded in 2001. The centre is one of the few research groups in the world with access to a 7 Tesla MRI scanner (most are limited to 3 T) and who have as abundant CMRS patient scan data for the study of heart disease, diabetes and obesity [3]. The OCMR has thus been an invaluable source for the data sets used in this project.

1.2.3 The Utility of Machine Learning in Healthcare

The last decade has seen a significant uprising in the application of AI and machine learning to the field of medicine. However, the ethics, reliability and suitability of these tools must be carefully considered if it may replace traditional approaches. In the context of heart disease, machine learning has the potential to benefit healthcare in the following 3 ways which will be considered in this project:

1. **Redefining and justifying risk thresholds for cardiac disorders**

   Currently in diagnosis, thresholds of risk for cardiac disorder tend to be set with only single variable consideration rather than in a multi-variable sense. For pre-symptomatic patients, clinicians arbitrarily combine each factor to predict early onset cases. Presumably, the accuracy of diagnosis would be increased if performed on a case basis with conditional thresholds across multiple variables. For example, given a patient is 40 years old, male, has a systolic blood pressure of 120 mmHg and a total cholesterol of 4 mmol/L, above what threshold of LVMass would indicate the early onset of heart failure? Currently, even the AI rule-based decision support used by the NHS in their Electronic Health Record (EHR) system currently lacks this kind of precision [18]. In Chapter 3, we will show how this objective can be achieved through tree-based classification algorithms.

2. **Developing automatic diagnostic systems with superior accuracy to clinicians**

   With access to larger amounts of data and further robustness testing, machine learning classifiers may be able to determine risk and onset of cardiac disorders sooner and more accurately than clinicians can on their own. For simple and specific imaging diagnosis tasks, such as detecting the malignancy of pigmented skin lesions, Tschandl et al. (2019) found that classifiers outperformed the judgement of 27 human experts with over a decade of experience, achieving a mean number of correct diagnoses of 25.43 vs 18.78 ($P<0.0001$) [64]. However limited data and greater complexity behind the factors of influence currently hinders the accuracy and reliability of machine learning compared to humans. Instead, for now, any application of machine learning ought to collaborate with and

complement human decision making, rather than replace it altogether.

3. **Discovering novel insights in cardiac medical research**

   There is often a dichotomy placed between machine learning and traditional statistics in terms of the predictive performance vs. explainability trade-off - that machine learning was designed with greater emphasis on high accuracy prediction while statistics was primarily designed for inferring relationships between variables. However with new developments in machine learning interpretability and explainability tools (e.g. feature importances, SHAP and Bayesian networks), this report argues that machine learning has the potential to achieve both high predictive accuracy and insightful explanations. Applied to the context of cardiac medical research, this report assesses the types of inferences that can be made from machine learning classification models and scrutinising its validity against medical literature statistics.

1.3 Project Overview

Focusing on applications (1) and (3), this project applies multiple machine learning algorithms using Python to derive novel insights into cardiac pathology, and to achieve high accuracy predictions within the scope of the OCMR data sets. By cross-referencing medical literature, interpretations from the results are used to either refute or further substantiate evidence related to CMRS metrics.

Chapter 2 details the preprocessing, missing data imputation, statistical and correlational analysis of features included in both OCMR data sets, which includes AS, MR, heart failure, diabetes and obesity patient subgroups.

Chapter 3 implements optimised Decision Tree and Random Forest classification models on both OCMR data sets to be able to differentiate between healthy patients and heart disease, diabetes and obesity patients. A range of regularisation methods including cost-complexity pruning, hyperparameter tuning, and recursive feature elimination are employed to reduce generalisation error; thus achieving a test accuracy of 95.36% with 94.69% AUC score for data set 1. Model interpretability tools such as decision tree diagrams, feature importances and SHAP value plots are generated to assist in the medical explanations underpinning the predictions of the model.

Chapter 4 demonstrates the ability of Bayesian networks to learn causal structures from data, mapping the inter-dependencies of variables to further explain mechanisms behind cardiac pathology. For the NP-hard problem of structure learning, an implementation of the K2 score& search algorithm is developed, and then compared to the novel NOTEARS approach which reformulates the combinatorial optimisation problem into a continuous one for a globally optimal evaluation. Bayesian networks are learned from the heart disease and diabetes data sets, turning the former into a classifier and achieving a 92.72% test accuracy. Preliminary techniques in causal inference are applied, using do-calculus interventions to simulate the counterfactual of increasing blood pressure to estimate its causal effect on the probability of heart disease.

Chapter 5 summarises some of the key medical insights, achievements and limitations of the approaches in this project.

Chapter 2

Data Preprocessing & Analysis

Any machine learning problem relies critically on the quality and condition of data which it inputs. "Garbage in, garbage out" is a common phrase in computer science which conceptualises the logic that even the most advanced algorithms are rendered useless if the data input is poor. It is important to be aware of this in a two-fold sense. First in how the format and structure of the data can affect the algorithms computation, sometimes prohibiting it from evaluating correctly, such as with feature scaling and missing data. But just as important is how the statistical composition and bias of the sample vs. the population being modelled influences the interpretation of results and what inferences can be made. Awareness of the latter is especially necessary if the data set is small in relation to the complexity of the system being modelled, which may be true for this project. This motivates the attention placed on the data preprocessing and analysis step of this project.

This chapter introduces the two OCMR data sets which will be used in this project, outlining a semantic description, a statistical and correlational summary of each feature variable, which is also used to clean and validate the data; methods of missing data imputation and feature scaling will then be discussed and implemented.

2.1 OCMR Dataset 1 (multiple heart disease classes)

The first data set received by the OCMR, with 8 features and 535 data points, is an amalgamation of patient data collected from multiple cross-sectional research studies. The groups consist of aortic stenosis patients of varying severity, mitral regurgitation and heart failure patients, obese and healthy control patients. Figure 2.1 illustrates the working data table, followed by a description of each feature.

	Class	Group	PCr/ATP	LVMass	LVEDV	LVEF	SBP	DBP	BMI	Age
0	Unhealthy	1	0.960	170.0	134.0	76.865672	187.0	95.0	23.639692	69.887671
1	Unhealthy	1	1.330	198.0	117.0	86.324786	149.0	71.0	33.491875	69.128767
2	Unhealthy	1	1.375	181.0	123.0	70.731707	135.0	68.0	27.870541	74.887671
3	Unhealthy	1	1.440	132.0	110.0	72.727273	NaN	NaN	28.280724	77.265753
4	Unhealthy	1	1.450	155.0	164.0	60.975610	148.0	72.0	30.094730	74.295890
...	...	...	...	...	...	...	...	...	...	...

Figure 2.1: Data table for data set 1, before missing data imputation

- **PCr/ATP:** The ratio of Phosphocreatine (PCr) to Adenosine Triphosphate (ATP) of heart cells, measured using

31P-MRS imaging localised at the septum of the patient's heart. Healthy values are generally above 2.

- **LVMass:** The estimated mass of the left ventricle, in grams. After using M-mode echocardiograms to measure volume, the mass is calculated via the difference in volume of the left ventricular chamber and the epicardium delimited volume multiplied by the estimated myocardial density.
- **LVEDV:** The left ventricular end diastolic volume, measured in mL. This is the volume of blood in the left ventricle at the end of a diastole when it is completely filled, just before the systole.
- **LVEF:** The left ventricular ejection fraction, calculated as the ratio of stroke volume (or difference between EDV and ESV) to end diastolic volume as a percentage. $LVEF = \frac{LVEDV - LVESV}{LVEDV} \times 100(\%)$
- **SBP:** The systolic blood pressure, measured in $mmHg$. Normal values are less than $120mmHg$
- **DBP:** The diastolic blood pressure, measured in $mmHg$. Normal values are less than $80mmHg$
- **BMI:** The body mass index is the ratio of the patient's body weight in kg to their height in m^2. A healthy range of values is between 18.5 and 24.9.
- **Age:** The age of the patient, in years, at the time of PCr/ATP measurement.
- **Group:** The labelled group of the patient taken from a corresponding study, consisting of: **1:** Severe aortic stenosis (AS); **2:** Severe AS with impaired left ventricle; **3:** Severe mitral regurgitation (MR); **4:** Healthy normal BMI; **5:** Moderate AS; **6:** Athlete; **13:** Obese; **14:** Heart failure reduced ejection fraction (HrEF); **15:** Overweight normal;
- **Class:** Converted into a binary label of whether the patient has heart disease or not. Label 0 indicates no heart disease (groups 4,6,15), 1 indicates heart disease (groups 1,2,3,5,14).

It was decided that the obesity study group would have to be removed from the data set since, although this group suffers significant negative effects on their heart, they do not have an identifiable heart disease yet. Putting the group in either class would likely skew the classifier results due to a distorted sample distribution (e.g. the impact of BMI may be over-represented).

Next, to analyse the distributions of each feature, a facet grid of histograms is plotted as shown in Figure 2.2, and a table of summary statistics is generated. From this, any outliers and duplicate entries have been detected and removed. Most distributions seem to follow a Gaussian distribution except for LVEF which appears to be bi-modal. Many variables such as BMI and LVEDV exhibit a positive skew. The normality of the feature variables is a relevant consideration when deciding the choice of classifier; for example, the Naive Bayes classifier rests on the assumption of feature normality [61]. As a result, Naive Bayes has been ruled out as a potential classifier for this project.

Comparing statistics of the data set to normal populations, the sample distribution of PCr/ATP appears roughly consistent with those stated in literature for that age range, 1.83 ± 0.4 (for this data set) vs. 1.77 ± 0.37. The same generally applies for other variables.

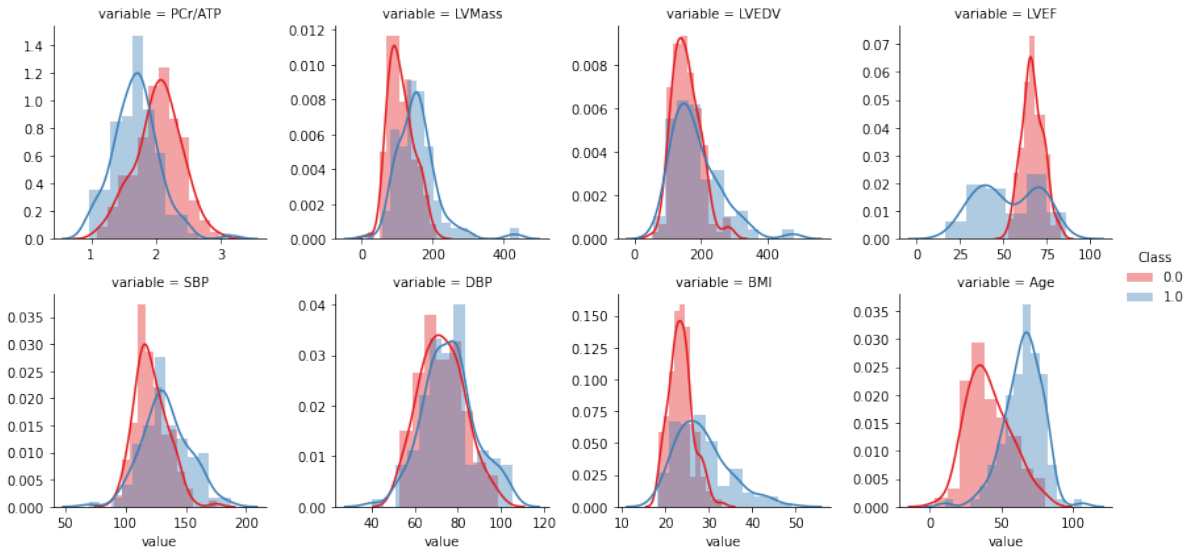


Figure 2.2: Histogram plot of each feature in dataset 1, separated

2.1.1 Feature Correlation

Correlations between features in the data set are analysed. The utility of this stems from two reasons: firstly, it may help infer connections between features (although being wary of the correlation-causation fallacy), and secondly it is important to detect **multicollinearity**. The presence of multicollinearity may be problematic when applying machine learning techniques because it increases the tendency towards overfitting [22].

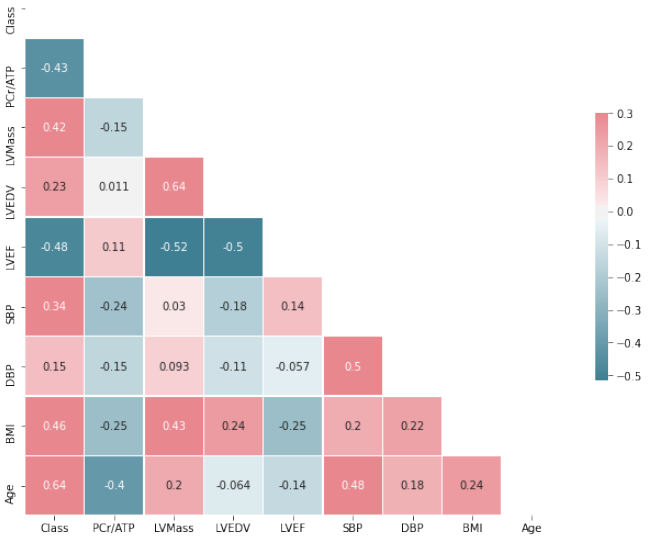


Figure 2.3: A correlation matrix plot of features, representing the Pearson coefficients between each pair of features

As shown in Figure 2.3, no features have a Pearson correlation coefficient greater than 0.65, indicating a low risk of multicollinearity for the dataset. As expected, there is a strong negative correlation between LVEF and LVEDV since LVEF is explicitly calculated from LVEDV as previously shown (given a constant LVESV they are inversely proportional). Between LVMass and LVEDV there is a positive correlation, however it is not obvious which might be a cause of the other. Garg et al. (2017) found that increased LVMass associated with hypertrophy was followed by increases in LVEDV [21], however this remains an interesting question to explore in Chapter 4 using Bayesian networks. As for the correlation to Class, Age, BMI, LVEF and PCr/ATP have the strongest correlation (in descending order), reflecting the separation of distributions in Figure 2.2. It will be of interest to examine how the order of these features change when analysed in terms of classifier feature importance in Section 3.4.2.

2.2 OCMR Dataset 2 (diabetes, obesity and heart failure)

The second data set obtained from the OCMR includes several more patient measurements such as from blood tests and body fat analysis. Although there are more features (17 in total) there are fewer data points, with only 220.

	Class	PCr/ATP	cardiaclipid	BMI	Group	EF	EDV	ESV	SV	Mass	Age	sex	glucose	chol	SBP	DBP	Totalfatmass	Visfat	LVMVR
0	Healthy	2.12	0.19	18.400000	1	81.90	130.90	23.7000	107.2000	96.70	23.0	NaN	5.00	4.3	131.0	80.0	7.9	34.300000	0.738732
1	Unhealthy	1.64	0.89	18.591130	5	33.46	210.23	139.8800	70.3500	177.19	73.0	NaN	6.90	5.8	NaN	NaN	NaN	42.520000	0.842839
2	Healthy	2.80	0.37	18.700000	1	64.30	125.60	44.8392	80.7608	78.50	29.0	NaN	4.55	4.4	113.0	75.0	10.2	16.300000	0.625000
3	Unhealthy	1.41	1.13	18.938776	5	42.73	245.75	140.7400	105.0100	135.22	50.0	2.0	4.40	5.2	NaN	NaN	5.5	22.000000	0.550234
4	Healthy	2.32	0.56	19.000000	1	83.00	95.00	17.0000	78.0000	59.00	81.0	NaN	4.90	5.7	120.0	60.0	NaN	NaN	0.621053
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

Figure 2.4: Data table for dataset 2 before missing data imputation

As can be seen from Figure 2.4, there are multiple features similar to the first data set (PCr/ATP, BMI, EF, EDV, Age, SBP, DBP). The new features distinct to this data set are described below.

- **cardiaclipid:** The myocardial lipid content, measured using 1H-MRS.
- **SV:** Stroke Volume, the amount of blood pumped by the left ventricle out of the heart in one contraction.
- **glucose:** The blood glucose concentration in $mmol/L$. Normal range is between 3.9-7.1 $mmol/L$
- **chol:** The total cholesterol in $mmol/L$ measured from a blood test. Desirable values are below 5.2 $mmol/L$, and high is considered as above 6.2 $mmol/L$.
- **Totalfatmass:** The total body fat mass in kg, measured with bioimpedence analysis.
- **Visfat:** Body fat stored within the abdominal cavity, located near several vital organs. The area, measured in cm^2 is measured on a T1 weighted MRI slice across L5 spinal segment and then manually contoured.
- **LVMVR:** The left ventricular mass to volume ratio

2.2.1 Feature Correlation

From the correlation plot in Figure 2.5, some multicollinearity is revealed between EDV and ESV, ESV and EF, as well as Totalfatmass and BMI. Thus for this data set in particular, feature selection methods will be employed when training classifiers - as will be seen in Section 3.4.3.

What makes this a more difficult data set is the fact that the patient groups for the class 'Unhealthy heart' are composed of more diverse conditions - diabetes, obesity, and heart failure. While medical theory

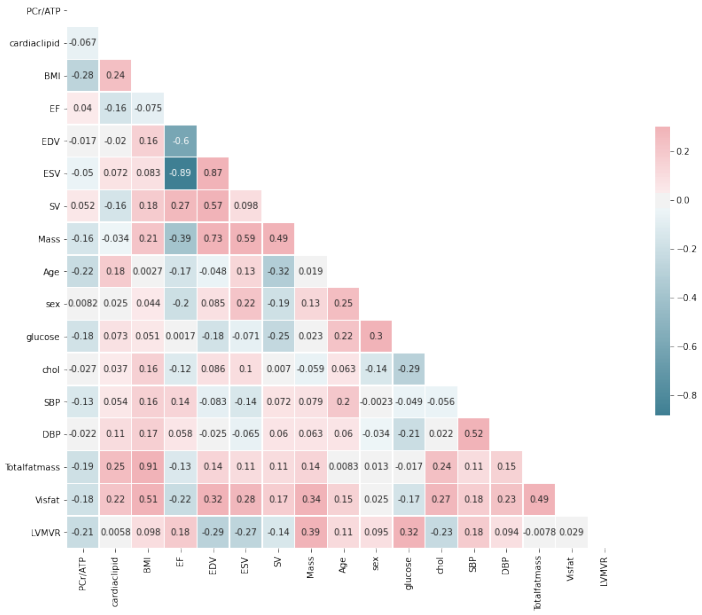


Figure 2.5: Correlation plot for data set 2

suggests each of these disorders can deteriorate the heart in some similar ways, the mechanism is likely to be vastly different, influencing features by varying amounts. Classifying patients in such a general manner would produce too much variance in the model and fail to capture the nuance unique to each condition, as these nuances would be indistinguishable to noise.

Additionally, although the data came labelled as these distinct groups, in fact it is discovered that they are not mutually exclusive. For instance, several patients in the 'diabetes' group are also obese, and several heart failure patients are also diabetic. To deal with this, new indicator variables on the data table for 'Diabetes', 'Obesity', 'Heart Failure' and 'Healthy' were created, including definitional conditions from the glucose and BMI columns. Then, separate subsets of the data set were produced for each case with the 'Healthy' set in each. Resulting subsets of data give the following balance of healthy vs unhealthy data points: Diabetes - 74 vs. 62, Heart failure - 74 vs. 33, Obesity - 74 vs. 103.

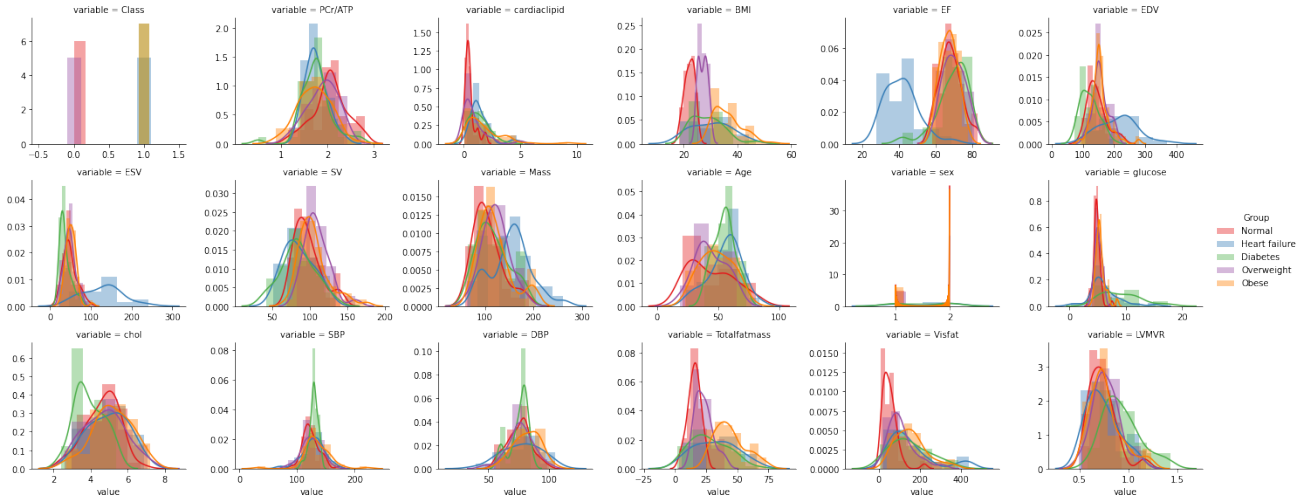


Figure 2.6: Histogram plot of each feature separated by Group

2.3 Missing Data Imputation

For reasons such as patient drop out, medical datasets often have missing values in some rows of the data. Machine learning algorithms generally cannot work with missing features and therefore methods need to be deployed to resolve this problem.

The simplest technique is Complete Case Analysis (CCA), where only patients with no missing data are considered. By simply removing data points with any missing columns, we circumvent the complication of imputation. However, doing so significantly decreases the data set size, losing out on a lot of potentially valuable information. In our case, for the OCMR data set 1 this would mean going from 535 data points to 181.

An important consideration when dealing with missing data is the concept of **missing mechanism** first formalised by Donald Rubin (1976) [54]. This investigates the underlying mechanism which generates the missing data and categorises it into 3 types: *Missing completely at random (MCAR)*, where the occurrence of a missing value in a data point does not

depend on values in the data set, neither missing or observed ones; *Missing at random (MAR)* where the missingness is related to some of the observed data, but not on missing data; and *Not missing at random (NMAR)* where missingness is related to other values in the data set, both missing and observed.

When data is NMAR, there is relevant information contained in the fact that the data point has a value missing; data points with missing values will differ systematically from the completely observed data points. Therefore if data is removed in the CCA method, not only will a large amount of valuable data be lost, it will also introduce bias into the new data set [56]. The data set will no longer be representative of the population. Hence more intelligent methods must be considered.

2.3.1 Mean Imputation vs. Multiple Regression Imputation

One technique that can be performed is mean substitution of missing data. While this has the advantage of being simple and fast and does not change the mean of any variable, it can severely distort statistical distributions of variables - notably, it will underestimate the variance and attenuate any correlations that exist between features [40], thus diminishing the underlying relationships between features which we were intending to discover.

Instead, a more sophisticated approach is to model each missing feature as a function of other features, and estimate the value using linear regression. Using the Python library, Sklearn, and the `IterativeImputer` class, each missing feature column is computed using the observed features in an iterative round-robin fashion; after each missing value is inserted in the previous step, the next value will be computed including that new value. The order of imputation is shuffled and then repeated for `max_iter` imputation rounds.

However, this single imputation does not reflect uncertainty about the regression coefficients in the model. The estimates fit perfectly on the regression line without residual variance, leading to greater precision than is warranted. To resolve this, outcomes across multiple imputed data sets are averaged in a technique known as Multiple Imputation.

Implementing this technique for data set 1 required a trade-off of how many missing features per data point we were willing to impute, balancing bias from the removal of potentially NMAR data and the bias and noise associated with imputation. With the guidance of Lee & Huber (2011) [37], observations with less than 40% missing features were kept leading to a final size of 272 in data set 1. Data set 2 required no removal of data.

2.4 Feature Scaling

In certain scenarios, feature scaling is vital for the performance of machine learning algorithms. Such algorithms include gradient descent based algorithms like neural networks or logistic regression, since different feature ranges cause different step sizes for each feature. Distance-based algorithms like K-nearest neighbours, K-means or SVM are also susceptible as the similarity of the data points are being determined on different scales for different features, leading to higher weighting for higher magnitude features. However, tree-based algorithms are far less sensitive to feature scales since each split of a decision tree considers only a single feature at a time.

Data is usually rescaled via two different methods: normalisation or standardisation. Normalisation, also known as min-max scaling, shifts values such that the range falls between 0 and 1.

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Standardisation instead rescales each features values using its mean and standard deviation, resulting in zero mean and unit standard deviation for each feature.

$$X' = \frac{X - \mu}{\sigma}$$

Standardisation has the advantage of being far less affected by outliers, however due to its unbounded range can be problematic for certain algorithms such as neural networks. This does not apply for any of the experiments of this project and therefore standardisation will be the chosen feature scaling method. Standardisation will be applied where necessary, most prominently during Bayesian network structure learning, using Sklearn's `preprocessing.StandardScaler` class.

2.5 Data set Limitations

As is a common issue in medical machine learning, the size of the data sets provided is less than ideal for drawing high confidence conclusions. This is especially true for data set 2, and particularly for the heart failure subset which only includes 107 data points. Greater caution and scepticism will therefore be practised when interpreting these results. Small data set size also restricts the viability of data-intensive algorithms such as neural networks, thus imposing additional constraints on the choice of classifiers. Although data augmentation methods could be applied to increase sample sizes – random interpolation for example – from an information-theoretic perspective this would be ineffective. No new information could be added via augmentation, only increasing existing biases.

Additionally, the composite nature of both data sets may introduce unintended biases in the results. Although each distribution has been carefully analysed to account for skew, the unknown selection criteria for each patient study may have resulted in an unrepresentative population and the inclusion of confounding variables.

Dataset 1 is highly balanced 133 vs 139, however data set 2 has greater imbalance. Care must therefore be taken when interpreting accuracy metrics, which will be outlined further in Section 3.2.

For the application to Bayesian networks in Chapter 4, although it is still viable, there is decreased amounts of certainty that can be given to causal inference results due to the data set being cross-sectional. Due to the atemporal nature of this observational data, ruling out the possibility that cause and effect is confounded is more difficult than if it were longitudinal (e.g. tracked a group of patients as they developed or recovered from heart disease).

Chapter 3

Tree-based Classification Models

3.1 Introduction

When selecting between the multitude of classification models which could be used in this project, emphasis was placed on 3 criteria.

1. **Interpretability** - any clinical or medical research tool necessitates the ability to understand the reasoning behind its prediction. This establishes user trust, accountability and knowledge discovery.
2. **Performance** - to ensure the predictive accuracy, generalisability and robustness of the model.
3. **Faithfulness to the data generating process** - whether the model correctly reflects the underlying logic of the data generating process, i.e. the mechanism by which heart disease arises.

Despite the superior performance noted in literature of deep learning models such as neural networks, there is little utility of using this for our data sets due to the lack of interpretability (often referred to as a black-box model) and the large data requirements (due to the higher expressivity and parameter complexity of the model)[22]. This applies similarly for another eminent classification algorithm – although less data expensive – known as the Support Vector Machine (SVM), which finds a max-margin hyperplane to optimally separate data points between classes.

However, placing attention to point (3), algorithms such as SVM constructs its classification model in a way that is argued to be oblivious to the logic of cause and effect. For instance, it deals with non-linearity by performing a 'kernel' trick with an arbitrarily chosen function to map its inputs such that a hyperplane can separate it. Essentially these are glorified curve-fitting methods that do not faithfully model the data generating process - and arguably is less useful in the context of medicine.

It may be conjectured that the underlying mechanism of heart disease occurrence depends on a complex set of conditional interrelations and logical rules which apply in some cases and not others. For example, perhaps a high LVMass is only a problem if the patient is above a certain age and with a visceral fat above a particular value. This corresponds to the concepts of patient disease subgroups or precision medicine. With a singular curve-fitting approach, algorithms like SVM, K-nearest neighbours or logistic regression are not able to learn this type of conditional logic unless pre-programmed.

Tree based algorithms such as Decision trees and Random Forests explicitly learn and model these subgroupings,

using conditional decision rules that vary between the branches of its tree, providing different sets of logic for different patients. Simultaneously, it facilitates interpretability in the form of feature importances and tree-diagrams. In terms of performance, although the 'no free lunch' concept applies (which states that no one classification algorithm yields superior results for all data sets), several empirical studies have suggested higher accuracy of tree-based algorithms. Comparing 10 different classifiers (including SVM, KNN, neural networks and logistic regression) on 11 data sets using 8 performance metrics, Caruana et al. found that Boosted Decision Trees ranked first followed by Random Forests [12]. Thus, by rationale of its performance, interpretability and theorised suitability to the data generating process, Decision Trees and Random Forests have been chosen.

In the following chapter, first the methods used to evaluate and benchmark different classifier models will be discussed; then Section 3.3 explores Decision Trees both theoretically and experimentally, techniques used for regularisation, hyperparameter tuning, and the implementation to both OCMR data sets; Section 3.4 then aggregates many decision trees into a meta-algorithm known as Random Forests, demonstrating its superior generalisability both theoretically and empirically, introducing feature importances and recursive feature elimination, finally demonstrating the results to both data sets. Lastly, an additional model-agnostic interpretability tool known as SHAP will be applied to the classification of each data set to further extract medical insights.

3.2 Model evaluation methods

When optimising machine learning models, it is important that the metrics used to gauge its success are aligned closely with the goals intended for it. In classification, these goals are more nuanced than a simple accuracy measure, requiring higher granularity for complete interpretation. Furthermore, when data is limited the reliability of evaluation through a single test set is not assured, motivating other techniques that account for variance in the metric. Thus, the performance of each classification algorithm will be evaluated using three sets of methods discussed and justified below.

3.2.1 Cross Validation Accuracy

Typically, accuracy in machine learning is measured using the hold-out method. This means to split the data set into a 'train' set to build the model with, and a 'test' set to evaluate its performance based on unseen data. The ratio is usually 80% and 20% respectively, and split on randomly shuffled data points.

However, this results in an accuracy metric that is dependent on how the data happens to be split at a singular instance. In small data sets, there is a higher risk that the test set might not be representative of the train set or of the population, leading to highly variable estimates of test accuracy - as will be prominent in data set 2. To overcome this, cross-validation is a more robust and informative method of testing.

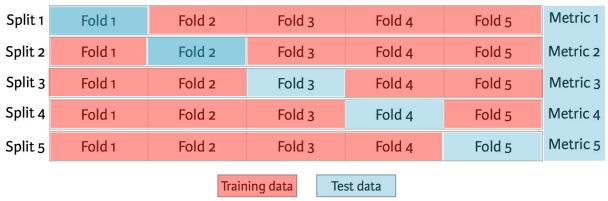


Figure 3.1: 5-fold cross validation [27]

K-fold cross-validation is method which randomly splits the data set into k groups. At each iteration, one group is set aside as a test fold while the model is trained on the other $k-1$ groups. This is repeated until each group has been used as a test set. An example of this with 5 folds is shown in Figure 3.1.

Leave-one-out cross validation (LOOCV) vs. K-fold

Leave-one-out cross validation (LOOCV) is an instance of K-fold cross validation in which $k = n$, the size of the data set. The model is iteratively trained on $n-1$ observations, tested on one single held out observation, and the final test accuracy is the average over the n splits.

Compared to the hold-out method, there is a lower tendency to overestimate the test error, and when performed multiple times produces more consistent results. K-fold also has some of this advantage over the hold-out method. However, as discovered through empirical tests, K-fold generally results in more accurate test error measures than LOOCV due to its tendency to LOOCVs higher variance error [32].

The n fitted models in LOOCV are trained on almost identical observations, leading to highly correlated outputs. Note that the mean of many highly correlated quantities has a greater variance than the mean of quantities that are less correlated. Thus for K-fold, given that there is less overlap in training sets, there will be less correlation and hence less variance [63].

Kohavi (1995) recommends 10-fold cross validation as an optimal compromise between these two opposing effects for a data set of similar size as the OCMR data sets. Furthermore, from empirical tests Kohavi also finds that stratification of K-folds fares better results both in terms of bias and variance. Stratification means that each fold will be guaranteed to have roughly balanced proportions of each class label. Thus, Sklearn's `StratifiedKFold` class will be implemented as a cross-validator using $k = 10$ for the accuracy evaluation and hyperparameter tuning of each classifier.

3.2.2 Confusion Matrix

However, using accuracy as metric even with stratified K-fold cross validation has its limitations. Primarily, it is susceptible to class imbalances in the data. For instance, if a disease occurs in the general population only 10% of the time, having a 90% accuracy in a classification model could plausibly occur in a scenario where all patients are blindly classified as not having a disease. Thus a very poor model may be evaluated in similar performance to another more intelligently trained classifier. Although data set 1 is fairly balanced, data set 2 would be particularly susceptible to this skewed evaluation.

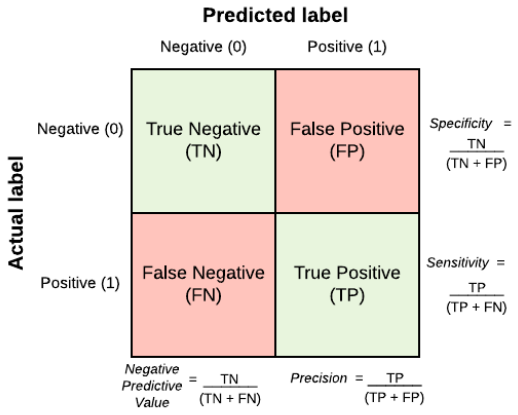


Figure 3.2: Confusion matrix

To avoid this and to ensure more faithful representations of model performance, a classifier should be quantified in terms of number of True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) outcomes. As

shown in Figure 3.2, each element of the confusion matrix displays the number of each outcome in the test set. Based on these, metrics can be defined which capture different intuitions of predictive 'accuracy' which are judged as important. For example, Sensitivity (also known as Recall), Specificity, and Precision whose equations are given above.

3.2.3 ROC Curve

Cross validated accuracy and confusion matrices are both useful tools for evaluating a classifier using the decision threshold of 0.5 (i.e. classify patient in class 1 if $P(C = 1|\, data) > 0.5$). However, it is sometimes desirable to change this threshold to increase the rate of true positives - at the expense of false positives - or vice versa. Or even purely to interpret a classifier with higher granularity, some method is required to visualise this trade-off.

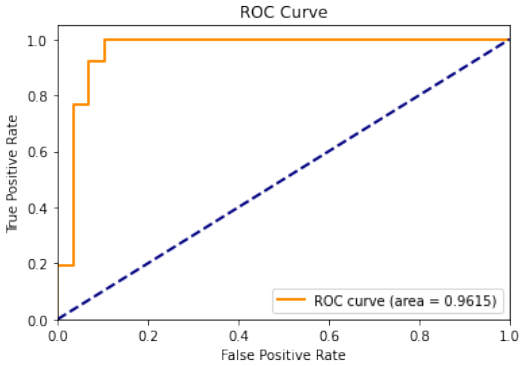


Figure 3.3: An example of an ROC curve

The Receiver Operator Characteristic (ROC) curve is one such method. It is a graph depicting the performance of a classifier at a range of thresholds, plotting True Positive Rate (TPR), a.k.a. the probability of detection, against False Positive Rate (FPR), or equivalently, against 1-Specificity, a.k.a. the probability of false alarm.

$$TPR = \frac{TP}{TP + FN}, \quad FPR = \frac{FP}{FP + TN}$$

From the ROC curve, a metric known as the Area Under Curve (AUC) can be computed by integrating under the curve. This is an aggregate measure for the performance across all decision thresholds. It is essentially equivalent to the probability that the classifier will rank a randomly selected positive observation more highly than a randomly chosen negative one [24]. The proof for this can be shown as follows, taking $FPR = x(s) = 1 - F_0(s)$ and $TPR = y(s) = 1 - F_1(s)$, where $F_{1/0}$ is the CDF for the score, s, of a positive/negative class and $f_{1/0}$ is the pdf for the score of a positive/negative class. A change of variables for the decision threshold, $s = \tau$, is performed:

$$AUC = \int_{x=0}^{1} y(x)dx = \int_{\infty}^{-\infty} y(\tau)x'(\tau)d\tau = \int_{\infty}^{-\infty} (1 - F_1(\tau))(-f_0(\tau))d\tau = \int_{-\infty}^{\infty} (1 - F_1(\tau))f_0(\tau)d\tau$$

Now transforming it in terms of probabilities[29]:

$$\int_{-\infty}^{\infty} P(f_1 > \tau)P(f_0 = \tau)d\tau \;=\; E_x[P(f_1 > f_0)] \;=\; P(f_1 > f_0)$$

Interestingly, it is worth noting that the AUC is essentially an instance of the Mann-Whitney U statistical test, which provides a useful intuitive interpretation for this measure. Furthermore, that the AUC also bears a relation to the Gini coefficient through the formula $G = 2 \times AUC - 1$ [23].

The Precision/Recall curve is another evaluation tool which also evaluates trade-offs along thresholds. However this tends to be preferred only when positive classes are rare in the data set or when there is more concern over FPs than FNs [22]. The data sets in use are fairly balanced and due to the nature of the problem, FNs have greater consequences than

FPs. Undetected heart disease patients without treatment suffer far worse prognosis than a patient wrongly diagnosed, as medication and lifestyle changes do not pose much danger. This, along with the beneficial statistical interpretations of AUC, justifies the preference for using an ROC curve in the following experiments.

3.3 Decision Trees

A Decision Tree is a non-parametric, highly interpretable classification model, composed of the following 3 components:

1. **Nodes:** represents a test or decision criteria for the value of a feature
2. **Edges:** corresponds to the outcome of a test and connects the parent node to a child node or leaf
3. **Leaf nodes:** these are terminal nodes which represent the resulting classification

3.3.1 CART Algorithm for Decision Tree Learning

The Classification and Regression Tree (CART) algorithm is the most common method of learning Decision Trees, and it executes a procedure known as *binary recursive partitioning*. The training set is recursively split into two subsets using a single feature j and threshold t_m, searching for a locally optimal pair which results in the purest subsets weighted by size. The cost function to be minimised is shown below, where n_l or n_r is the number of instances in the left or right subset, and G_l or G_r is the impurity of the left or right subset [22].

$$J(j, t_m) = \frac{n_l}{n} G_l + \frac{n_r}{n} G_r$$

The algorithm continues until all the leaf nodes are pure or until the max depth is reached, as summarised below.

Algorithm 1: CART Decision Tree Algorithm [55]

{Initialise: d=0 } ;
while *d < max_depth AND n < min_samples* **do**
  for *each feature* $j = 1, ...k$ *and threshold value* $t_m \in \mathbb{R}$ *that can be split on* **do**
    $\theta = (j, t_m)$;
    Compute subsets of split ;
    $Q_l(\theta) = \{(x, y) | x_j \leq t_m\}$;
    $Q_r(\theta) = \{(x, y) | x_j > t_m\}$;
    Compute cost function: $J(Q, \theta) = \frac{n_l}{n} G_l + \frac{n_r}{n} G_r$;
  end
  $\theta* = arg \min_{n_l, n_r} J(Q, \theta)$;
  $d \longleftarrow d + 1$;
  Recurse on both children sets $Q_l(\theta*)$ and $Q_r(\theta*)$;
end

The error function. G, also known as 'impurity' can either take the form of the Gini Impurity or Entropy. Both are metrics used to quantitatively evaluate how good a split is.

Entropy is a metric taken from Shannon's information theory, measuring the average information content of a signal or variable. Entropy is zero when the signal is identical. Applied to impurity, entropy is zero when a split contains

instances of only one class, and increases as the representation of other classes. This relationship is defined below:

$$H_i = -\sum_{k=1}^{C} p_{i,k} log_2(p_{i,k})$$

For each split i, $p_{i,k}$ is the frequency of instances of class k in that split.

Gini Impurity instead measures the probability that a randomly chosen instance in the entire set would be incorrectly labelled if it were randomly classified according to the distribution of classes in the split.

$$G_i = \sum_{k=1}^{C} p_{i,k}(1 - p_{i,k})$$

Literature suggests only a marginal difference between these measures on final outcomes, with a rate of disagreement as low as 2% according to Raileanu & Stoffel (2004) [51]. While Gini Impurity is a slightly faster computation, it also tends to isolate the most frequent class into its own branch of the tree, whereas Entropy leads to slightly more balanced trees [22]. For this reason, although minor, Entropy will be chosen as the impurity metric for the following experiments.

3.3.2 Regularisation Hyperparameter Tuning

Unlike other linear models, Decision Trees are non-parametric meaning that the number of parameters are not determined prior to training. By making few presumptions about the training data, the model can adapt more freely to the data, thus increasing the risk of overfitting [22]. To prevent this, regularisation hyperparameters are introduced into the model – namely a maximum depth (`max_depth`), minimum samples required for a split (`min_samples_split`), and a minimum weighted fraction of the sum total of weights required at a leaf node (`min_weight_fraction_leaf`).

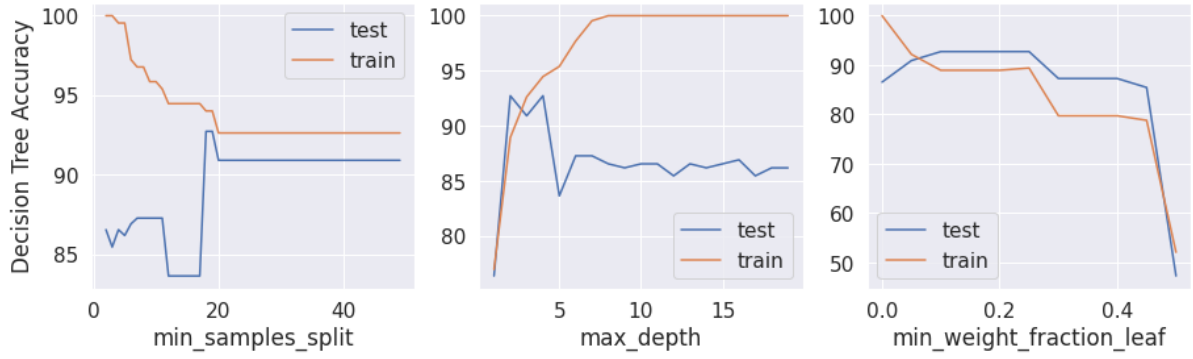


Figure 3.4: Train and test error against hyperparameter values for OCMR dataset 1, generated using matplotlib

To explore the effects of these hyperparameters, a set of graphs plotting both train and test error against hyperparameter values have been generated for data set 1 in Figure 3.4. As expected, increasing `min_samples_split` reduces overfitting which can be seen from the simultaneously decreasing train accuracy and increasing test accuracy. This is essentially the same but the inverse effect of increasing `max_depth`.

Post-pruning

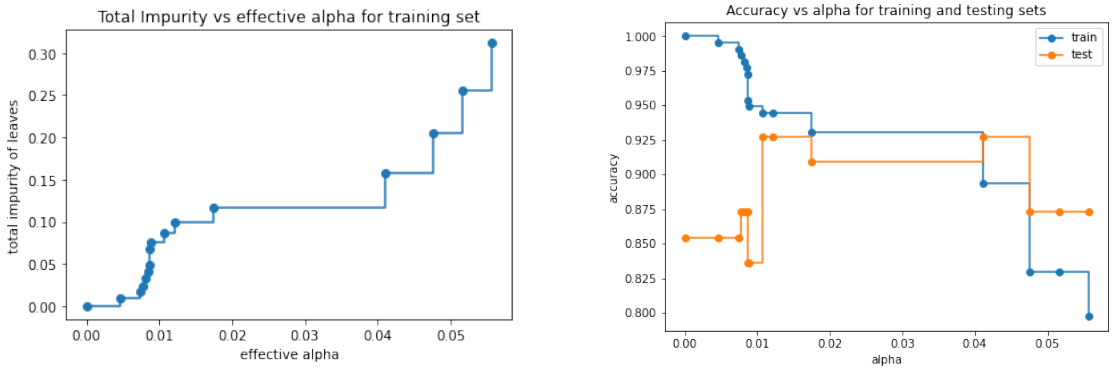
As another method to prevent overfitting, pruning is implemented in the CART algorithm. Pruning means to remove selected branches of the tree which use low importance features, applied after the decision tree has been generated. This can be done either by reduced error pruning or cost complexity pruning. Reduced error pruning simply removes subtrees at each iteration which reduce the misclassification rate $R(T)$ of the given tree T, however this method favours larger trees and will not always guarantee a reduction in overfitting. Instead, the CART algorithm uses cost-complexity pruning, which adds a penalty for the complexity of the tree. Minimal cost-complexity pruning is an optimisation algorithm which defines the following cost-complexity to be minimised:

$$C_\alpha(T) = R(T) + \alpha|\tilde{T}|\,, \qquad \alpha \geq 0$$

$R(T)$ is the total misclassification rate of the terminal nodes of a given tree T, α is the complexity parameter, and $|\tilde{T}|$ is the number of terminal nodes in T. Normally, for a given subtree T_t where t is the root node, the node has a greater impurity than the sum of the terminal nodes, $R(t) > R(T_t)$. Yet, for the cost-complexity measure, these values can be equal for t and T_t depending on the value of α. The effective value of α is defined by this point, and is computed as:

$$\alpha_{eff} = \frac{R(t) - R(T_t)}{|T| - 1}$$

The algorithm recursively finds and prunes non-terminal nodes with the smallest α_{eff} until $min(\alpha_{eff}(t)) > \texttt{ccp\_alpha}$, which is the algorithm's parameter value [1]. To investigate the effects of pruning, experiments have been run using data set 1, measuring the effect of `ccp_alpha`. As shown by Figure 3.5a, increasing alpha increases the total leaf impurity of the decision tree [49]. This occurs up until the tree has only one node, where at this point `ccp_alpha`= 0.187.



(a) Impurity of leaves against alpha

(b) Train and test accuracy against alpha

Figure 3.5: The impact of pruning on total leaf impurity and train/test accuracy

More significantly, Figure 3.5b demonstrates how pruning reduces overfitting. As alpha increases from 0 to 0.02, the train accuracy decreases from 100% to 93% while the test error increases from 85% to 91%, proving its improved generalisation. However, past $\alpha = 0.04$ the test error then decreases again, exemplifying the effect of the bias-variance trade-off.

Grid Search vs. Random Search

The ideal hyperparameters to use for training a model vary between data sets, as these depend on distinct characteristics of the underlying data generating process. As such, this prompts the need for a systematic approach to determining the optimal combination of hyperparameters.

The choice between Grid search – an exhaustive search over combinations of hyperparameters evaluated with cross-validation – and Random Search – which selects these combination randomly – is usually decided based on computational feasibility. Although Random Search will not always find the optimal parameters, impressively it will do so more often than not and in a fraction of the run time [5].

Seeing as the search space for these experiments are not large, grid search will be used. As per Section 3.2.1, Stratified K-fold cross validation will be performed on the training set. Most importantly, during this entire process the test set is held out and excluded. This is to prevent **data leakage** in the final evaluation. Using Sklearn's `DecisionTreeClassifier` class, the hyperparameters `max_depth`, `min_samples_split`, `min_samples_leaf` and `ccp_alpha` will be tuned as these have the most effect on overfitting.

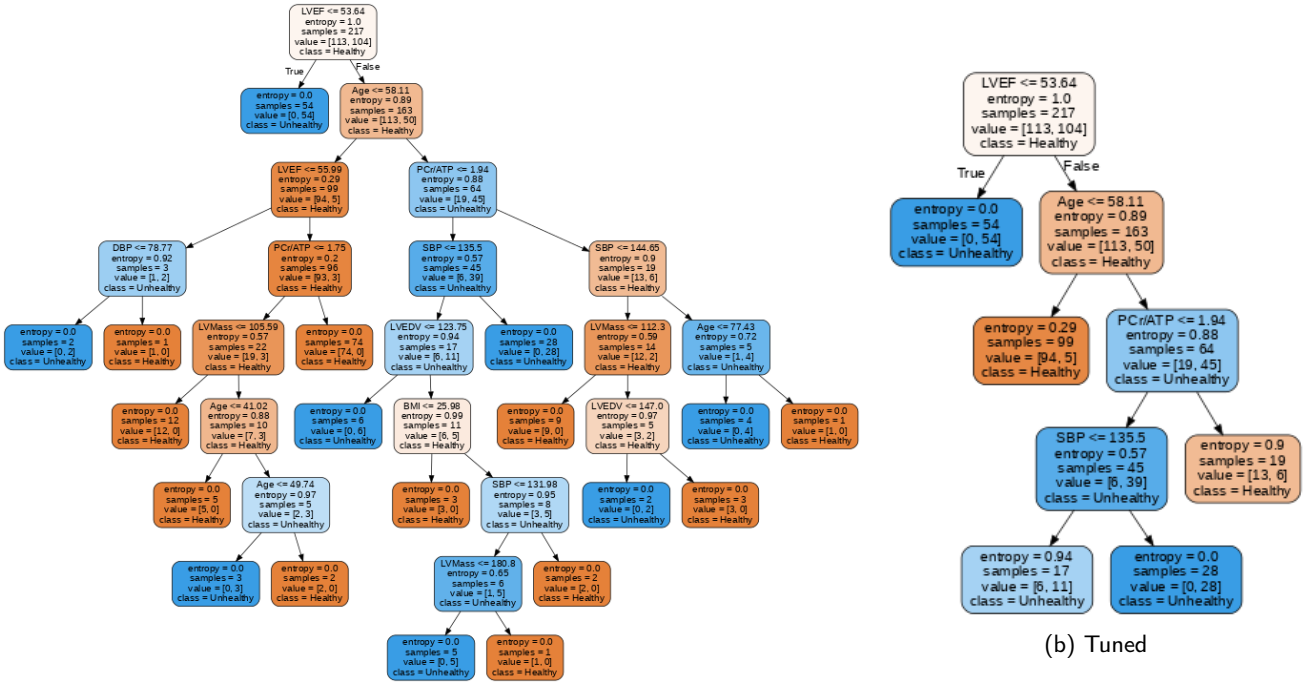
3.3.3 Data set 1 results

Performing a cross validated grid search with `cv=10`, optimal hyperparameters are determined to be {'ccp_alpha': 0.04, 'max_depth': 4, 'min_samples_split': 2, 'min_weight_fraction_leaf': 0}. For this first set of results, to empirically illustrate the significance of hyperparameter tuning (as in Section 3.3.2), decision trees have been trained and evaluated both before and after tuning for comparison.

The resulting decision trees, displayed in Figure 3.6, have been visualised using the `graphviz` library. In this tree, LVEF is shown to be the most important discriminator for the classification of heart disease, where a value below 53.64% is able to detect heart disease with zero entropy for 54/104 of the training samples. Other risk thresholds detected in order of decreasing importance are having an Age ≥ 58.11, PCr/ATP ≤ 1.94 and having an SBP ≥ 135.5. Even as a preliminary attempt, the thresholds learned by the algorithm quite impressively aligns with those found in medical literature as mentioned in Section 1.2.1 (SBP$\geq 140mmHg$ and PCr/ATP ≤ 2).

The rules learned by the untuned version add far greater nuance than the tuned version, but it is to be debated whether these nuances represent a reality beyond the training set. Nodes deeper down in the tree are more obviously non-generalisable. For example, the node of Age ≤ 41.02 being a parent to another node of Age ≤ 49.74 essentially states that - conditioned on the rules of previous nodes - patients older than 49.74 and younger than 41.02 would be healthy but those between these ages are unhealthy. Clearly, the algorithm has started to learn idiosyncrasies in the training set so as to classify every instance perfectly, even if it means interpreting the data noise as a signal.

However, some nodes higher up in Figure 3.6a imply nuances that are a little more plausible. For example, although you can classify with an entropy of 0.89 that a patient is healthy if they have an LVEF above 53.64, to reach zero



(a) Untuned

(b) Tuned

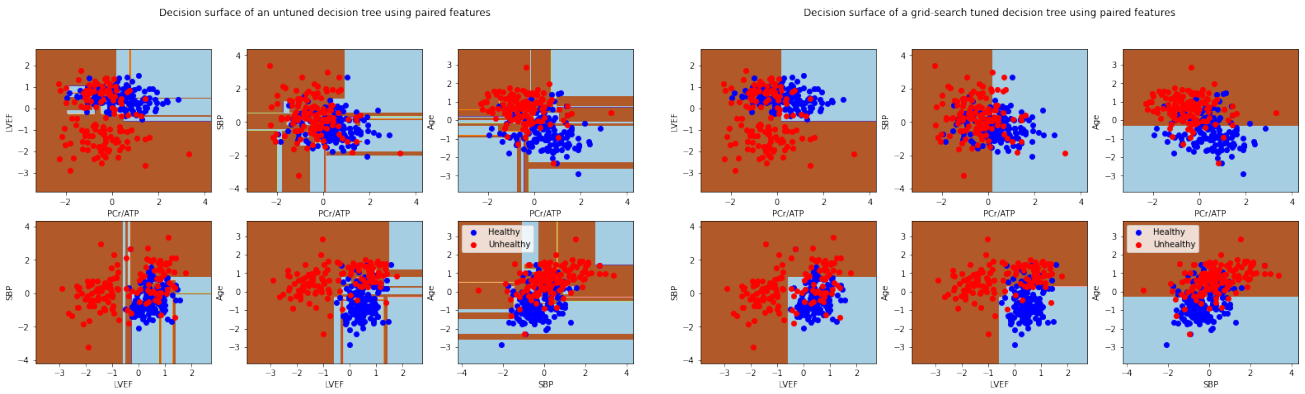
Figure 3.6: Visualisations of learned Decision Trees for OCMR data set 1, using graphviz.

entropy they would need the criteria of being younger than 58, have an LVEF> 56 and PCR/ATP> 1.75. This implies an interesting hypothesis that the PCr/ATP threshold to be unhealthy is lower if the patient is younger. On one hand, this might be opposed to literature's findings that the average PCr/ATP of older patients is lower (e.g. 1.6 ± 0.4 for those of age 60 ± 13 vs. 1.7 ± 0.3 aged 32 ± 3) [7]. However, this could still imply a theory that the hearts of older patients are more sensitive to the decrease of PCr/ATP, perhaps due to an impaired ability for other compensatory mechanisms.

Decision Surface

Every split of a decision node represents the creation of a decision boundary. Due to multidimensionality, these splits between data points typically cannot be studied. However, by selecting pairs of features, one can plot the data points along with the decision surfaces that the classifier uses to distinguish these points. This has been performed for PCr/ATP, LVEF, SBP and Age, as presented in Figure 3.7. Note that feature standardisation has been applied, in accordance with the reasons explained in Section 2.4.

As shown by Figure 3.7a, without tuning the decision tree overfits, creating many implausible tiny surfaces to accommodate for exceptions to the larger surfaces. For example, the splintering of Age between 49.74 and 41.02 as mentioned above. This presents an alternative visualisation for the effect of not imposing a max depth or pruning. Figure 3.7b on the other hand, separates data points in a more simplistically, but with higher impurity, arguably underfitting.



(a) Untuned (b) Tuned

Figure 3.7: Decision surface for variable pairs of data set 1

Accuracy

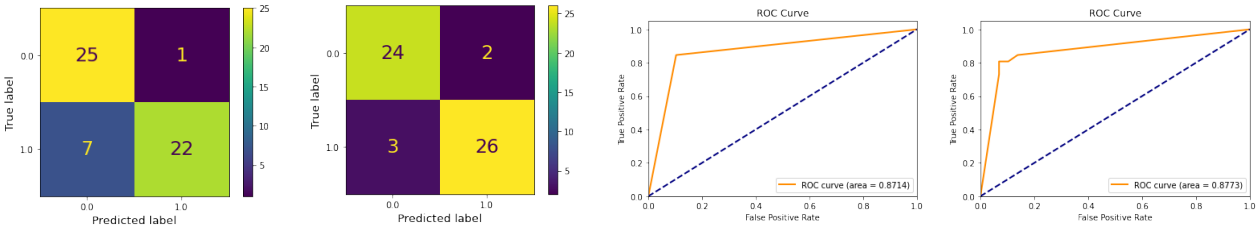
The results shown in Table 3.1 confirms the claim that the untuned decision tree overfits the training data, therefore substantiating the necessity of regularisation.

	Train accuracy % (hold out)	Test accuracy % (hold out)	Test accuracy % (CV)	AUC score %
Untuned Decision Tree	100.00	85.45	78.69 (+/- 7.77)	87.14
Tuned Decision Tree	92.17	90.91	80.20 (+/- 13.87)	87.73

Table 3.1: Accuracy metrics for the untuned and tuned decision tree models for data set 1

Confusion Matrix & ROC curves

Although the tuned version slightly increases the number of FPs (from 1 to 2), there is a far greater decrease the number of FNs (from 7 to 3). This is a much preferred trade-off since FNs impose a more detrimental cost in medical diagnosis than FPs.

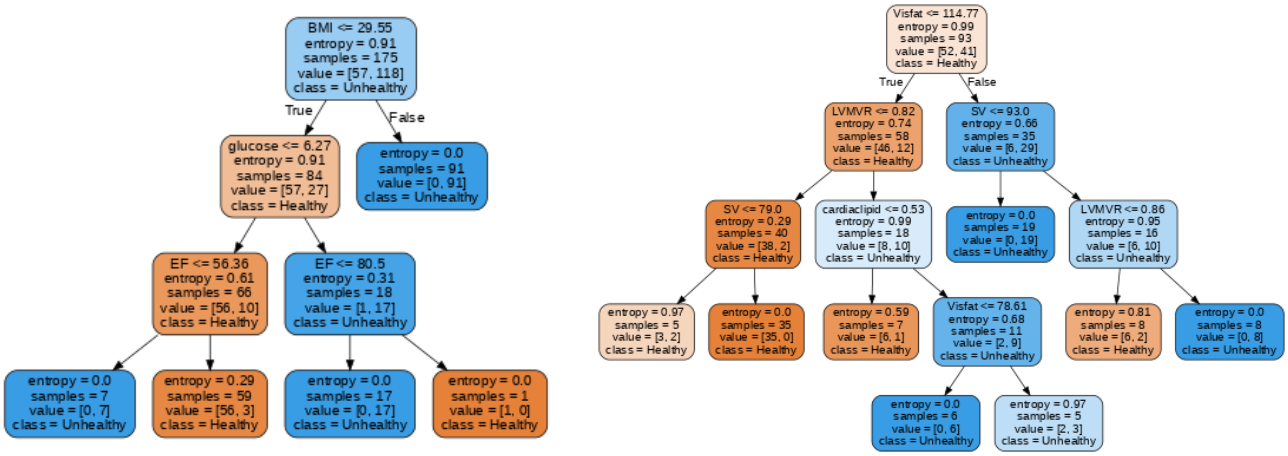


(a) Untuned (b) Tuned (c) Untuned (d) Tuned

Figure 3.8: Confusion matrices and ROC curves for decision tree classifiers for data set 1

3.3.4 Data set 2 results

For data set 2, results will be displayed for each subset of the data classifying Diabetes, Heart Failure, Obesity and the general 'Unhealthy' groups in turn. The classifier has been tuned specific for each case, but for clarity only results for the tuned decision trees will be compared.



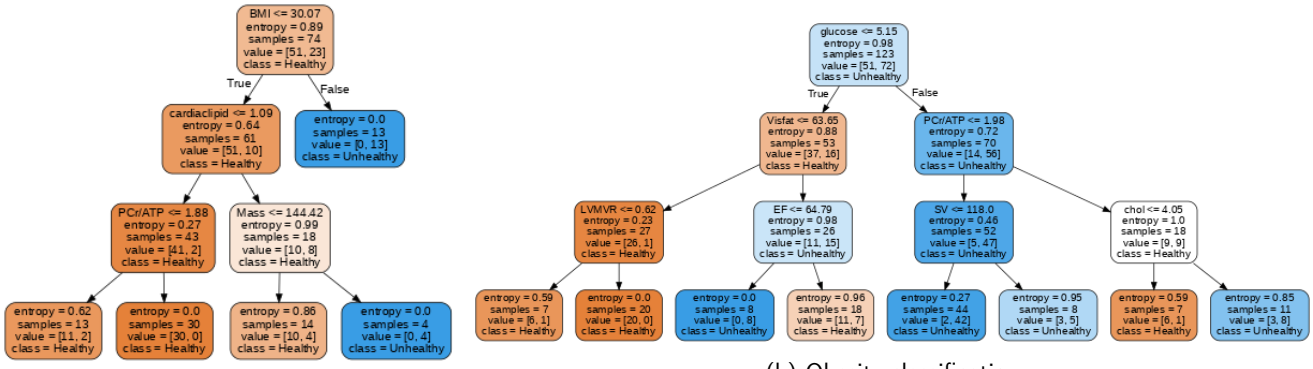
(a) Unhealthy classification

(b) Diabetes classification

Figure 3.9: Decision Trees

Confirming the logic mentioned in Section 2.2, grouping the data too generally - under the semantic of 'Unhealthy' heart - leads to uninformative results. As shown, the decision tree has simply learned the groups the "Unhealthy" class is composed of in order of descending proportion: Obese as $BMI < 29.55$, Diabetic as $glucose > 6.27$, and Heart failure as having $EF < 56.36$. It does not learn any other non-definitional details of what an "Unhealthy heart" entails. This motivates the necessity of reasoned feature selection.

The diabetes classifier reveals far more interesting results once definitional features, i.e. glucose, are removed. Figure 3.9b demonstrates how the decision tree classifier has learned a number of insightful decision rules. For example, that $Visfat \leq 114.77$ and $LVMVR \leq 0.82$. This result is supported by Levelt et al. which found a 31% increase in LVMVR associated with diabetes (0.97 ± 0.17 vs. $0.74 \pm 0.14\ g/mL$), due to concentric remodelling of the heart [39].



(a) Heart failure classification

(b) Obesity classification

Figure 3.10: Decision Trees

Since all heart failure patients in this data set have reduced ejection fraction, EF and related measures (ESV,EDV and SV) must be removed to avoid the trivial result of 100% test accuracy. Besides detecting that heart failure patients of this data set are skewed towards a higher BMI (as seen in Section 2.2), the decision tree also learns that having a cardiaclipid > 1.09 and Mass > 144.42 increases the likelihood of being classed as having heart failure. This reflects the

aforementioned compensatory mechanism of the failing heart in which the left ventricle hypertrophies. The cardiaclipid result is particularly interesting and supports the aforementioned findings of Paolisso et al.

In Figure 3.10b, besides the obvious detection of high glucose and visceral fat being indicative of obesity, the decision tree also picks up an interesting nuance of obese patient subgroups related to PCr/ATP. It implies that, given a patient has a higher blood glucose, they are more likely found to be Obese if their PCr/ATP ratio is less than 1.98 (and further so if they have a SV < 118) - this subgroup represents 42/72 of the obese patients.

	Train accuracy (hold-out) %	Test accuracy (hold-out) %	Test accuracy (CV) %	AUC score %
Unhealthy/Healthy	98.29	86.36	81 (+/- 50.03)	74.34
Diabetes	98.88	82.51	82.53 (+/- 30.73)	72.47
Heart failure	93.24	96.91	83.91 (29.18)	87.39
Obesity	95.12	79.63	73.27 (+/- 24.37)	63.46

Table 3.2: Table of accuracy results for tuned decision tree classification with data set 2

3.3.5 Limitations

In practice, Decision Trees alone are rarely an optimal classifier for several reasons.

- Learning the optimal decision tree is an NP-hard problem so all practical methods rely on heuristics. CART, among others, are 'greedy algorithms' that only determine optimal splits locally at each node. Since leaf impurity of future splits are not considered, globally optimal trees cannot be guaranteed.
- Decision Trees suffer from instability – they are sensitive to small variations in training data due to its propensity towards orthogonal decision boundaries.
- Decision Trees tend to overfit – even with regularisation they still generalise poorly. [1]

3.4 Random Forest Classification

As demonstrated in Section 3.3, Decision Trees suffer from high variance, overfitting and sensitivity to small changes in the training data. While techniques such as pruning and hyperparameter regularisation have improved upon these limitations, the lack of generalisability and instability still poses an issue, especially when applied to small data sets. To overcome this, many decision trees will be applied in tandem to implement a more powerful ensemble meta-algorithm known as Random Forests.

Random Forests, developed by Breiman (2001) [8], uses two forms of randomness to train individual decision trees. First is known as Boostrapping, where random subsets of observations are drawn from the training set with replacement - to ensure each subset is decorrelated. Secondly, each tree is also trained with a randomised subset of features. Finally, the overall classification of the random forest is determined by the majority vote over all the

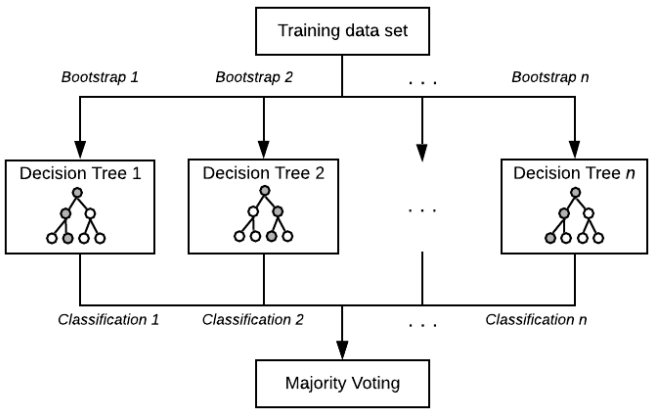


Figure 3.11: Random Forests schematic

individual tree classifier predictions, as depicted by Figure 3.11. Analysing this in terms of bias-variance trade-off, bootstrapping and feature randomisation produces many high variance but low bias trees. Through aggregation and in accordance with the Law of Large Numbers, the ensemble forest will generate classifications that are both low variance and low bias, thus improving generalisability.

3.4.1 Mathematical and empirical proofs for generalisation

This section attempts to explain some of the postulations of the previous section with mathematical rigour, outlining the theoretical proof that random forests converge, which will later be experimentally verified.

Let each k^{th} tree classifier be denoted by $h_k(\mathbf{x}, \Theta_k)$, where $\mathbf{x}$ is the input data vector and Θ_k are independently identically distributed random vectors which represent the parameters of each decision tree. In an ensemble of K classifiers, $h = (h_1(\mathbf{x}), h_2(\mathbf{x}), ..., h_K(\mathbf{x}))$, the empirical margin function is defined as:

$$\hat{m}(\mathbf{x}, y) \equiv \hat{P}_k(h_k(\mathbf{x}) = y) - \max_{j \neq y} \hat{P}_k(h_k(\mathbf{x}) = j) \tag{3.1}$$

$\hat{P}_k(A)$ is the proportion of classifiers where the outcome A occurs. In other words, the margin function is the extent that the average number of votes for the correct class exceeds the average vote for the next best class (for binary classification this is just the alternative class). Thus, the generalisation error for the ensemble h can be defined as:

$$e = P_{\mathbf{x},y}(\hat{m}(\mathbf{x}, y) < 0) \tag{3.2}$$

$P_{\mathbf{x},y}$ denotes the probability as x and y varies (similarly with P_Θ). Combining Equation 3.1 and 3.2, we arrive at the following theorem.

Theorem 1 *As $K \longrightarrow \infty$ (as number of trees increases), the generalisation error converges to a fixed value:*

$$e \longrightarrow P_{\mathbf{x},y}[P_\Theta(h(\mathbf{x}, \Theta_k) = y) - \max_{j \neq y} P_\Theta(h(\mathbf{x}, \Theta_k) = j) < 0]$$

This result follows from the fact that $\hat{P}_k(h_k(\mathbf{x}) = j) = E_k[I(h_k(\mathbf{x}) = j)] \equiv \frac{1}{K} \sum_{k=1}^{K} I[h_k(\mathbf{x}) = j]$, where I is the indicator variable. For the random sequence of Θ, where $h_k(\mathbf{x}) \equiv h(\mathbf{x}, \Theta_k)$, using the Law of Large Numbers, it can be shown that:

$$\frac{1}{K} \sum_{k=1}^{K} I[h(\mathbf{x}, \Theta_k) = j] \longrightarrow P_\Theta(h(\mathbf{x}, \Theta) = j)$$

Theorem 1 explains why random forests do not overfit as more trees are added, which one would typically associate with models that grow in complexity.

Theorem 2 *The upper bound for generalisation error in a Random Forest is given by:*

$$e^* \leq \frac{\bar{\rho}(1 - s^2)}{s^2}$$

$\bar{\rho}$ is the mean value of correlation of Θ, while s is the strength of the set of classifiers which is calculated using $s = E_{x,y}[\hat{m}(x, y)]$. This can be derived from first computing Chebychev's inequality for the generalisation error, which,

assuming $s \geq 0$ is given by: $e^* \leq \frac{var(\hat{m}(x,y))}{s^2}$ Then expanding out an upper bound of the variance of the margin function, $var(\hat{m}) \leq \bar{\rho} E_\Theta var(\Theta)$. The complete detailed proof is quite lengthy but may be found in [8] and [34].

The significance of this result is that the upper bound on the generalisation of a random forest depends on the strength of individual trees along with the correlation between trees. This proves why Bootstrapping and feature randomisation, which decorrelates classifiers, robustly improves generalisation.

Empirically, this can be tested by iteratively generating a set of random forests for a varying numbers of trees, `n_estimators`, and varying numbers of features used in the random feature subset, `max_features`, and averaging the accuracy (inversely proportional to generalisation error) over each set and plotting the corresponding graphs. As shown by Figure 3.12, both train and test accuracy plateaus to a fixed value as the number of trees increases. Holding `n_estimators` constant, increasing `max_features` up until `max_features = n_features` has the effect of slightly decreasing accuracy since this increases the correlation between each tree and thus increasing the overall variance (at the expense of a slightly lower bias). The green lines in each graph demonstrates the effect of not using Bootstrapping, setting `bootstrap = False` and instead using the entire data set to build each tree. Consistent with theory, the accuracy of these models prove be lower, therefore worse at generalising, than with Bootstrapping.

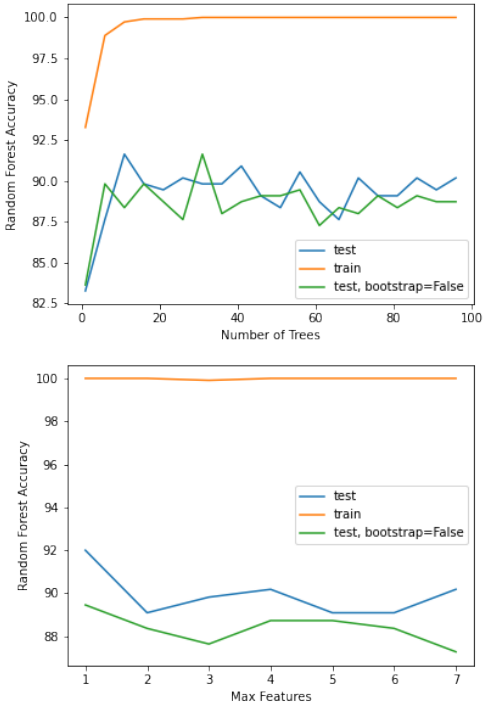


Figure 3.12: Accuracy against no. of trees and max features

3.4.2 Feature Importance

An added advantage of Random Forests is the interpretability derived from feature importances. With the mass of decision trees generated, feature importance is essentially the expected fraction of samples which a feature contributes to in predicting the output class and the total reduction of impurity it brings. For a feature variable, X_m, its importance is calculated by adding up the weighted reductions in impurity, ΔG, from all nodes t which uses X_m, averaged over all N_T in the forest:

$$FI(X_m) = \frac{1}{N_T} \sum_{T} \sum_{t \in T : f(s_t) = X_m} p(t) \Delta G(s_t, t) \tag{3.3}$$

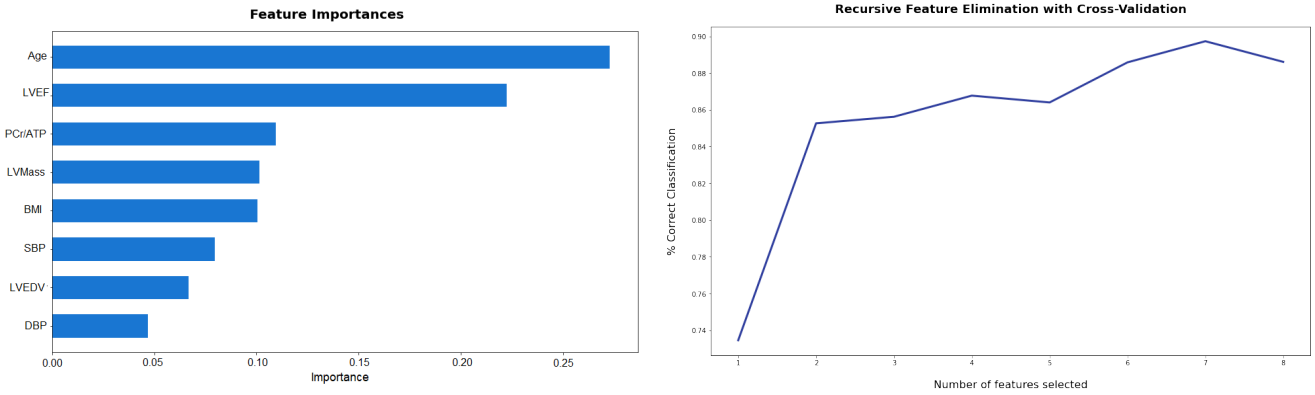
The weight, $p(t)$, is the fraction of samples reaching node t. $f(s_t)$ is the feature used in the split s_t [41]. Thus, features used at the top of a decision tree will contribute to the final prediction for a larger fraction of the input samples than those lower down. However, there are limitations to impurity-based feature importances. For instance, its bias to high cardinality features and the fact that it is computed based on train set statistics only [48]. To increase the reliability of

the importance results, another type of importance metric known as SHAP will also be evaluated in Section 3.5.

3.4.3 Recursive Feature Elimination (RFE)

Due to the possibility of redundancy and irrelevance, certain features may actually worsen the performance of a model when included. Analogous to removing noise from a signal, recursive feature elimination (RFE) will be implemented as a form of dimensionality reduction.

RFE works by recursively considering smaller and smaller subsets of features. A classifier is first trained on a set of features, feature importances are computed and then the least important features are removed from the set. This process is then recursively repeated on the set until the desired number of features is reached [2]. Scikit-learn's `RFECV` class uses the same principle but with a cross-validation loop to automatically find the optimal number of features.



(a) Feature importances

(b) Accuracy against number of features with RFECV

Figure 3.13: Feature importances and the impact of RFECV on accuracy for OCMR data set 1

For data set 1, using Stratified K-fold cross validation where K=10, RFECV has found the optimal number of features to be 7, eliminating DBP. The feature importances displayed in Figure 3.13b indicate that Age, LVEF, PCr/ATP and LVMass, in descending order, are the 4 factors of highest predictive power in classifying heart disease.

3.4.4 Data set 1 results

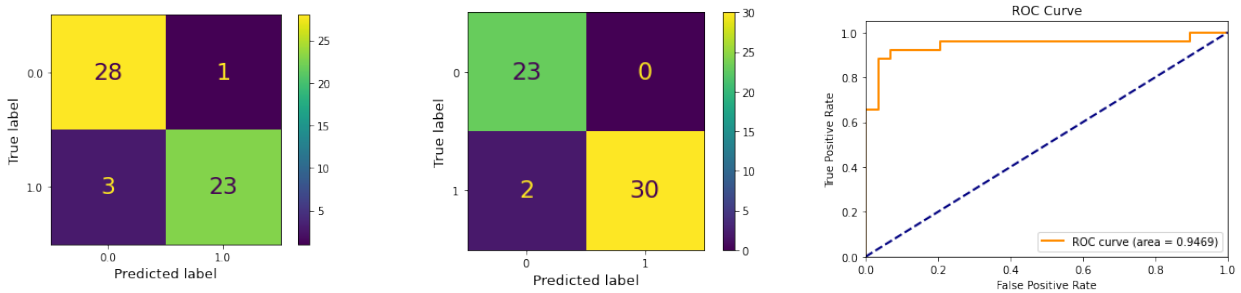
Using similar hyperparameter tuning methods as for decision trees, the optimal hyperparameters are determined as: `{'max_depth': 4, 'max_features': 2, 'n_estimators': 60}`. To evaluate the premise made in Section 3.4.3, a Random Forest classifier will be trained before and after performing feature selection via RFECV. This means evaluating the model when trained on all the features versus all but DBP. As shown in Table 3.3, in both cases the Random Forest

	Train accuracy % (hold out)	Test accuracy % (hold out)	Test accuracy % (CV)	AUC score %
Random Forest (all features)	96.77	94.55	87.87 (+/- 5.04)	94.33
Random Forest (after RFECV)	96.39	95.36	90.08(+/- 5.73)	94.69

Table 3.3: Random Forest classification results both before and after RFECV for data set 1

classifier performs far better than the optimally tuned decision tree with an increase of CV test accuracy of 9.88%. Note

that the 95% confidence interval has substantially reduced, from 13.87 with the tuned decision tree to 5.04. This further validates the claim made that Random Forests - as with other bootstrap aggregation methods - perform classifications with significantly lower variance than when using a single estimator. As a result of RFECV, misclassifications have decreased, resulting an increase in Sensitivity from 88.46% to 93.75%, and increase in Specificity from 96.55% to 100%.



(a) Confusion Matrix, all features

(b) Confusion matrix after RFECV

(c) ROC after RFECV

Figure 3.14: The impact of RFECV on the confusion matrices of random forest models for data set 1

3.4.5 Data set 2 results

For data set 2, although RFECV detected several features to remove for each subset, testing reveals that this does not always lead to an increase in accuracy. From Table 3.4, only the Obesity subset benefited from RFECV. This is likely due to the small data set sizes (of which Obesity is the largest), therefore resulting in high variance during the algorithms cross validation. The results also demonstrate a notable increase in all metrics of test accuracy for Random Forests compared to Decision Tree classification. For example, yielding increases in CV test accuracy of 7.52%, 1.18% and 8.58% for each respectively, along with decreases in variance once again.

	Results of RFECV		Final best results for Random Forest Classification			
	Removed features from RFECV	Increase in accuracy %	Train accuracy (hold-out) %	Test accuracy (hold-out) %	Test accuracy (CV) %	AUC score %
Diabetes	SBP, sex	-2.24	98.01	97.51	90.05 (+/26.71)	93.56
Heart Failure	sex, chol, SBP, DBP, Visfat, LVMVR	-4.53	95.95	96.97	85.09 (+/24.83)	96.30
Obesity	EDV, ESV, SV, Mass, Age, sex, DBP, LVMVR	+1.78	100.00	87.04	81.85 (+/21.09)	85.27

Table 3.4: Results for RFECV and Random Forest classification for OCMR data set 2

3.5 SHAP (SHapely Additive exPlanations) Interpretability Tools

In many cases, as in medical diagnosis, explaining why a model makes a prediction may be as important as the prediction itself. As mentioned, it supports user trust, provides a sense of accountability, hints at ways to improve the model, and facilitates understanding the underlying mechanism being modelled. This chapter introduces SHAP as a model-agnostic interpretability tool which improves upon the limitations stated in 3.4.2.

SHAP is a method of explaining the prediction of a particular observation by computing the contribution of each feature to that prediction. This is a development from Shapley values, an approach originating from cooperative game

theory, where a selection of features in a data instance are modelled as 'players' in a coalition, and the computed values tell us how to fairly distribute the 'payout' i.e the prediction among the features. The Shapely value is the average marginal contribution of a player across all possible coalitions.

In the context of machine learning, to compute the Shapely value of a particular feature, i, different subsets, S, are formed by permuting the set of all remaining features, $N/\{i\}$. Then the model output value of the S features is calculated, first without the feature i, then with feature i included. The difference is given by $v(S \cup \{i\}) - v(S)$. Since the effect of withholding a feature depends on the other features in the model, this difference needs to be calculated for all possible subsets. Summarised by equation 3.4, the Shapely value is the weighted average of all possible differences.

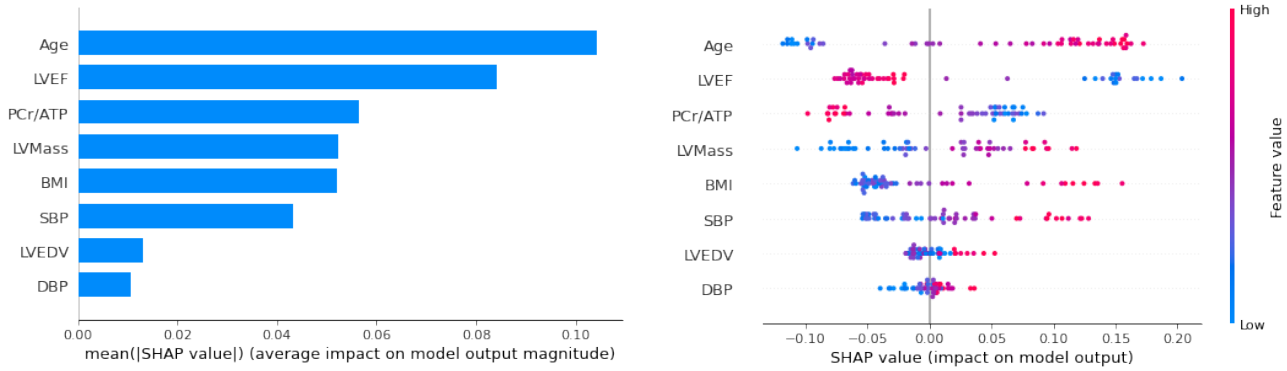
$$\phi_i(v) = \sum_{S \subseteq N/\{i\}} \frac{|S|!(N - |S| - 1)!}{N!}(v(S \cup \{i\}) - v(S)) \tag{3.4}$$

SHAP, by Lundberg & Lee (2016) [42], evolves this as an additive feature attribution method, essentially a linear function of binary variables shown in equation 3.5, where $z' \in \{0, 1\}^M$ and M is the number of simplified input features.

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i, \tag{3.5}$$

A benefit of SHAP is its simultaneous global and local interpretability. Local in the sense that each observation gives the contributions of each feature in predicting the target variable. Global in the sense that it shows how much each feature contributes - whether positively or negatively - to the target variable. This quality sets it apart from other interpretability tools such as LIME, which is inherently local, and justifies our choice of use in this project.

3.5.1 Data set 1 interpretation



(a) SHAP feature importances

(b) SHAP plot

Figure 3.15: SHAP plots generated from Random Forest classification models of OCMR data set 1

Using the `shap` library, SHAP values are computed for each feature and plotted in terms of mean magnitude (Figure 3.15a) and its raw value for each data point (Figure 3.15b). Importance rankings using SHAP are consistent with those from Figure 3.13 except that SHAP yields a far lower relative importance of LVEDV and DBP. The real utility of SHAP values compared to other feature importance measures is exemplified by Figure 3.15b. The SHAP values are capable of indicating not just a feature's impact on the model, but also the direction of that impact in terms of high or low

feature values. The model has intelligently learned that high values of Age, LVMass, BMI, SBP, LVEDV and DBP, but low LVEF and PCr/ATP increase the likelihood of having heart disease. These findings are consistent with literature and the Decision Tree results, indicating robustness of our methodology. Information may also be inferred from the skew of these plots. For instance, it appears that having a low LVEF is more impactful for predicting heart disease than a high LVEF is for predicting a healthy patient. This would aptly be explained by the distribution of the LVEF histogram plot of Figure 2.2, which showed that due to bi-normality, high LVEF could occur in both classes but not for low LVEF. A similar skew occurs also with BMI and SBP.

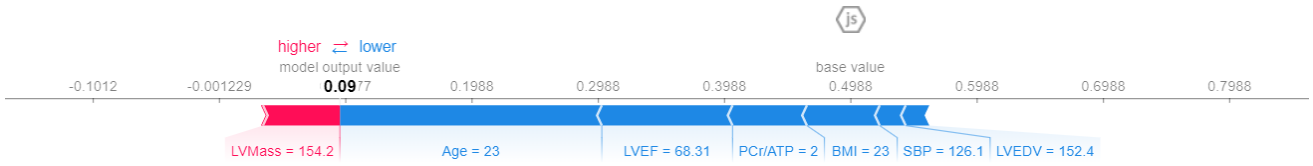
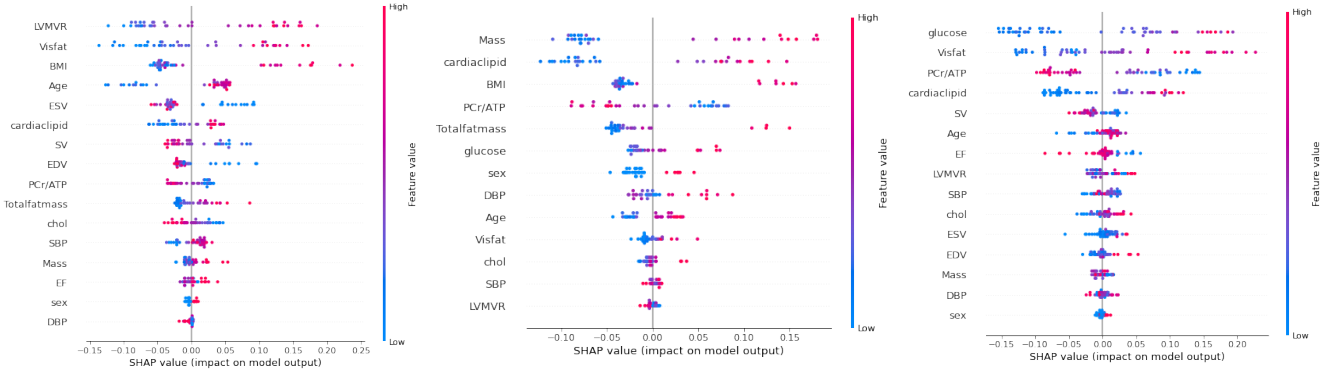


Figure 3.16: SHAP value plot for the interpretation of individual patient predictions for heart disease

Figure 3.16 displays a JavaScript visualisation tool which showcases the aforementioned local interpretability of SHAP. For a given patient, the plot shows the relative impact that the value of each feature has in its output prediction of heart disease, y; red indicates a positive impact (greater probability of heart disease) and blue indicates negative impact.

3.5.2 Data set 2 interpretation



(a) Diabetes (b) Heart failure (c) Obesity

Figure 3.17: SHAP plots for OCMR data set 2 generated from Random Forest classification models

The SHAP values generated for data set 2 reaffirm many of the interpretations from Section 3.3.4. Namely, the associations of increased LVMVR and Visfat with diabetes; increased LVMass and cardiaclipid with heart failure; and increased glucose, decreased PCr/ATP and SV with obesity. Most prominently, each plot indicates high impacts of high cardiaclipid feature values, strongly supporting the claims of Schultze et al. about myocardial lipid accumulation and altered fatty acid oxidation both heart failure, diabetes and obesity. Another inference can be made , in the case of both diabetes and heart failure, the skew reveals that higher BMI is more indicative of these diseases than low BMI is indicative of healthiness.

Chapter 4

Bayesian Networks for Causal Inference

In the previous chapter, tree-based algorithms have proven to be successful for making accurate predictions and providing a rationale for these predictions. However, there is a distinction to be made between interpretability and explainability in machine learning models. For example, feature importances may indicate which variables influence the prediction of heart disease most, but they do not provide information about *why* they are influential, or how they relate to other variables. Similarly in the case of traditional statistics where, famously, "correlation is not causation."

Results thus far have shown that decreased PCr/ATP is both correlated and important in the prediction of heart disease. However, no information is given about the direction of the causality (i.e. is it a cause or a symptom), or if they are even causally linked in the first place. Scientists address this problem by hypothesising conceptual theories, given empirical data, that could logically explain cause and effect relations. But what if computers could also help to theorise cause and effect by learning it through data, constructing a model to be used to perform highly specific predictions?

This chapter explores Bayesian Networks, a probabilistic graphical model developed by Turing award winner Judea Pearl, as a viable means of achieving this. Although Bayesian networks have been applied to medical diagnostics since the early 90s, limitations in the computational approaches to the NP-hard problem of structure learning from data have hindered its progress and application to causal inference. However, recent developments (NOTEARS algorithm) have found solutions to this intractability by reformulating the combinatorial optimisation into a continuous one.

As a preliminary exploration, this chapter examines the theory behind Bayesian networks, structure learning from data, and foundational concepts for causal inference. The K2 score algorithm is implemented and compared to the NOTEARS algorithm using both a validation data set and the OCMR data set to further our medical interpretation. Finally the learned Bayesian network is transformed into a classifier to be compared with those from Chapter 3.

4.1 Introduction to Bayesian networks

A Bayesian network is a directed acyclic graph (DAG), $G = (V, E)$, where each of the n nodes in V represent a random variable of interest, and the directed edges E encode informational or causal influences in the form of conditional probability dependence.

As a probabilistic graphical model, the crux of a Bayesian network is its efficient representation of joint probability functions.

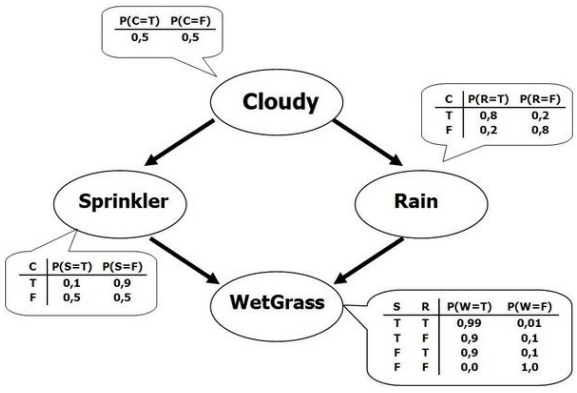


Figure 4.1: Example of a Bayesian network[60]

Given a distribution P of n discrete variables, $X_1, X_2, ...X_n$, the chain rule of probability calculus allows the decomposition of a joint probability into the product of its conditional distributions. However, Bayesian networks allow further simplification of this decomposition such that the conditional distribution of each node X_i is only a function of X_i's parents, π_i. π_i is the minimal set of predecessors of X_i which renders X_i independent of all its other predecessors.

$$P(X_1, ..., X_n) = \prod_{i=1}^{n} P(X_i|X_1, ..., X_{i-1}) = \prod_{i=1}^{n} P(X_i|\pi_i) \tag{4.1}$$

For example in Figure 4.1, the Bayesian network represents a joint distribution that can simply be factorised into $P(C, R, S, W) = P(C)P(R|C)P(S|C)P(W|R, S)$. Thus, the parameters defined by the Bayesian network model are the conditional probability distributions (CPDs) for each variable node, and these are only conditioned on its parents. These parameters are often represented in the form of a conditional probability table, as in Figure 4.1.

Equation 4.1 follows from the Local Markov Property of Bayesian networks, which states that each node is conditionally independent of all non-descendent nodes given its parents [35]. Note the definition of conditional independence as follows:

Definition 4.1.1 (Conditional Independence) *A is conditionally independent of B given C (denoted as $A \perp\!\!\!\perp B|C$) if and only if:*

$$P(A, B|C) = P(A|C)P(B|C) \tag{4.2}$$

Equivalently, another more useful definition can be derived from this via Bayes rule:

$$P(A|B, C) = \frac{P(A, B|C)P(C)}{P(B, C)} = \frac{P(A|C)P(B|C)P(C)}{P(B|C)P(C)} = P(A|C) \tag{4.3}$$

This is to say that *once C is known, learning B would not influence the belief in A*. Thus, once the values of a nodes parents are observed, the node is independent of all other predecessors.

Definition 4.1.2 (Markov Compatibility) *If a joint probability function, P, can be factorised in the manner as in Equation 4.1 relative to a DAG, G, then G and P are said to be Markov compatible.*

Ascertaining this compatibility means that a DAG can adequately represent the underlying data generating process [46]. Bayesian networks are thus capable of performing predictions such that it also models the data generating mechanism, as was desired by criteria (3) in Chapter 3 for the choice of classifiers. Notably, this factorisation in Equation 4.1 also significantly reduces the number of parameters required for computation, computation which would otherwise be intractable. Given n binary variables, the non-simplified joint distribution would require $O(2^n)$ computations, whereas a Bayesian network with at most k parents per node would only $O(n \times 2^k)$.

In order for a DAG to be deemed compatible with a probability distribution, a set of conditional independencies must be satisfied. Equivalently, to determine the presence of edges, a set of conditional independence tests on the data must be satisfied. To determine this from a DAG, a graphical criterion known as d-separation has been developed [46].

4.1.1 D-separation

To understand d-separation, the principles behind how information is transferred or blocked between two variables in a Bayesian network must first be outlined. There are 3 elementary types of connections that a Bayesian network is composed of: the Cascade, Fork and Collider [35]. To contextualise each type, examples of tri-node graphs with tenable medical evidence for have been learned from data set 2.

Cascade (X -> Z -> Y)

If Z is observed, X and Y are independent. Conversely, if Z is not observed, X and Y and dependent and information can be transmitted between X and Y. Hence, observing Z blocks the information path between X and Y.

In the context of Figure 4.2, if `Mass` (LVMass) is observed, `sex` and `LVMVR` are independent. This accurately represents relations noted in literature: in the general population, men have a larger LVMass than women [53], and by definition, LVMVR is dependent on the LVmass. If Mass is not observed, LVMVR would then be dependent on sex.

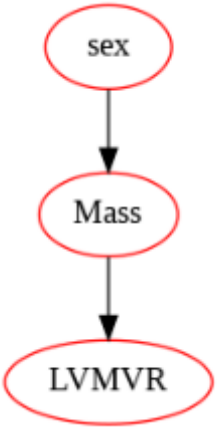


Figure 4.2: Cascade

Fork (X <- Z -> Y)

Just as before if Z is observed, X and Y are independent. If it is not observed, information can be transmitted among any of the children of Z. Thus, observing Z blocks the information path between X and Y.

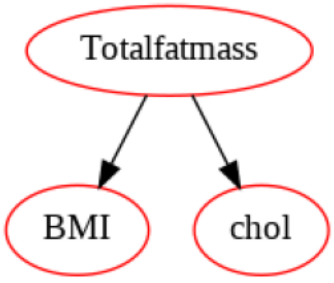


Figure 4.3: Fork

This is essentially the definition of confounding variables. Across population data, it would appear that BMI and cholesterol are correlated; given no other information, one might assume a high BMI causes a higher cholesterol. However when conditioned on the variable Totalfatmass, it is learned that the two variables are actually independent. Since BMI only takes into account height and weight, athletes, for example, can have a high BMI due to high muscle mass [31]. Athletes also tend to have a lower than average cholesterol due to their exercise [44]. With conditional independence relations in Bayesian networks, this information can be appropriately learned and represented, as shown by Figure 4.3. If Totalfatmass is not known, cholesterol is dependent on BMI (due to the general correlation between total fat mass and BMI), however once Totalfatmass is observed, they are independent since Totalfatmass has the stronger causal link.

Collider (X -> Z <- Y)

On the other hand, a collider connection is the inverse of the previous two types. If Z or its descendants are observed, X and Y become dependent on each other. Conversely, if neither Z nor its descendants are observed, X and Y become independent. Hence, observing Z or its descendants opens up the information path.

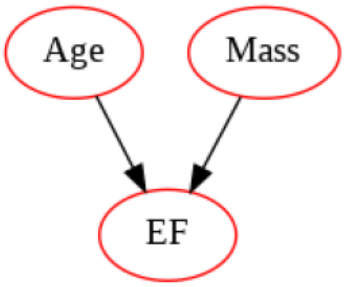


Figure 4.4: Collider

Referring to Figure 4.4, the learned graph states that EF is influenced by both a patients Age and LV Mass. It states that when EF is observed, Age and LV Mass depend on each other (observing a patients Age influences our belief about their LV Mass). Interestingly, it also states that in the absence of knowing EF, they are independent. This is a particularly informative inference to make as this relation has been controversial in medical literature - does age cause ones LV mass to increase, or is age simply associated with higher risk factors for heart disease, which themselves cause LV mass to increase? The graph learned from data set 2 states the latter, and is supported also by the subgroup findings from the Framingham study [17]. Although more data is required to confirm these three relations conclusively, one can appreciate how these representations can be useful in the context of medical research and in inferring the causal relations of observational data. More on this will be discussed in Section 4.2.

Aggregating the rules of these 3 elementary connections, conditional independence can be determined between any two selected nodes in a larger network via d-separation. D-separation, as delineated by Pearl (2009), is defined as follows:

Definition 4.1.3 *A path p is said to be **d-separated** (or blocked) by a set of nodes Z if and only if:*

1. *p contains either a chain $l \longrightarrow m \longrightarrow m$ or a fork $l \longleftarrow m \longrightarrow n$ such that the middle node m is in Z, or*
2. *p contains a collider $l \longrightarrow m \longleftarrow n$ such that node m is not in Z and no descendants of m is in Z.*

A set Z is said to d-separate X from Y if and only if Z blocks every path from a node in X to a node in Y [46].

The concept of graphical d-separation in a DAG has implications on whether a joint probability distribution is compatible with a particular Bayesian network. If a particular DAG contains sets X and Y that are d-separated by Z, then X must be independent of Y conditional on Z for every compatible distribution.

4.1.2 Observational Equivalence

A problem that results from Bayesian networks as a representation of conditional independences (or d-separation) is the fact that two Bayesian networks may be empirically indistinguishable. If one interprets the arcs in a DAG as causal influences, then each DAG would be distinct. However if a DAG is only to represent independence/dependence relations between variables, then many DAGs are capable of representing and factorising the same joint probability distribution.

Theorem 3 (Observational Equivalence) *Two DAGs are observationally equivalent if and only if they have the same skeleton (the undirected version of the graph) and the same set of collider structures*

For example, considering Figure 4.1 again, reversing the direction of the nodes $Cloudy \longrightarrow Rain$ would neither destroy nor introduce a collider structure. Thus, the reversal would yield a DAG that is observationally equivalent, and one would not be able to determine the directionality simply from probabilistic information. The edge $Rain \longrightarrow WetGrass$, however, is fixed in the sense that there would be no way to reverse the direction without creating a new collider structure. Observational equivalence is one of the obstacles preventing the direct inference of causality given a learned structure.

4.2 Causal Inference

Causality, as defined by philosopher David Hume, requires the fulfilment of two criteria:

1. *Regularity:* that a cause is regularly followed by an effect
2. *Counterfactual:* that if the first event had not been, the second had not have happened

The second criterion is often the missing piece preventing correlations derived from observational data to truly imply causation. For example, regularity can be derived from the association between visiting the hospital and lung cancer occurrence (hospital visits are regularly followed by the 'effect' of lung cancer diagnosis).

However, it is obvious that visiting the hospital is not the 'cause' of lung cancer because of criterion (2) since, had they not have gone to the hospital, they would still have had lung cancer regardless (the counterfactual does not hold true). On the contrary, we can know that smoking cigarettes *causes* cancer from the inferred counterfactual that if a person had not have smoked X cigarettes, they would not have developed lung cancer. Although, even this is difficult to rigorously prove since comparing a group of individuals who smoke vs. those who do not, it is hard be sure that it was not another confounding variable that is common to both groups. In fact, in the 1950's this very instance sparked mass controversy within the medical community; for example, some groups were convinced that there was a confounding gene expression. It was only after multiple controlled studies and other causal inference techniques (e.g. sensitivity analysis) that the controversy was resolved [47].

In absence of controlled variables, statistics may be susceptible to an even worse error: **Simpson's paradox**. For example, as in Figure 4.5 a correlation between exercise and cholesterol may exist when only these variables are observed - implying from this data set that 'exercise regularly follows an increased cholesterol'. However, when conditioning upon a third variable, Age, this correlation inverts itself, thus revealing a counterfactual that would have been false.

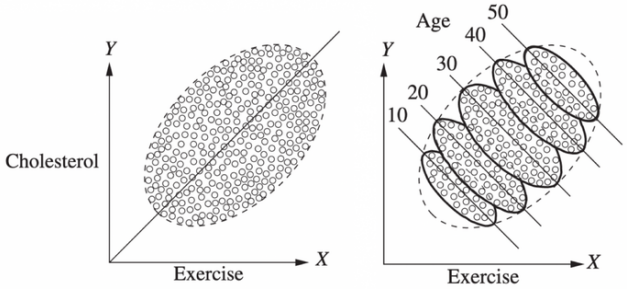


Figure 4.5: Example illustrating Simpson's Paradox [36]

Today, randomised controlled trials (RCTs) are the gold-standard approach for the proof of causality in medicine. Through randomisation, a way of holding all else equal besides the variable of interest (e.g. Age, sex etc.), one is able to infer the counterfactual by observing that the levels of cholesterol decrease with exercise. However, RCTs cannot always be applied due to its high cost, size requirements and ethics, rendering a multitude of observational data in medical research useless for casual inference. Thus entails two significant benefits of Bayesian networks for medical data: (1) the ability to detect the presence of confounders (as in Figure 4.3) to avoid Simpson's paradox. (2) Through the manipulation of a Bayesian network to simulate an 'intervention' and the subsequent probabilistic inference, a counterfactual can also be inferred in an equivalent way to RCTs. Pearl has also developed an entire framework of causal inference which

mathematically evaluates the effect of an intervention purely on observational data - this is known as do-calculus.

4.2.1 Do-calculus

Do-calculus enables the computation of probability beyond mere observation, instead into intervention. For instance, how would the probability of having an LVEF < 50 change if we were to *'do'*, or set, the value of BMI to 30, all else being equal? Note that this is different to the question of what is the probability of LVEF < 50 given I observe BMI as 30, since this would also change many other variables that would be associated with that particular state.

$$P(X|Y = y) \neq P(X|\, do(Y = y)\,)$$

Do probability is instead an intervention on the actual data generating process, and is performed **by removing the edges of any parents of the intervened node**. An example of this will be shown in Section 4.6.1. However, note that for the results of do operations to truly imply causation as in RCTs, it must be true that all relevant variables are identified by the network, i.e. that there are no hidden confounders that haven't been modelled. Methods exist to detect and compensate for this (e.g. back-door criterion/adjustment), however are beyond the scope of this report.

4.3 Learning Bayesian Networks from raw data

Besides the convenient probabilistic representation of Bayesian networks and the efficient probabilistic inference computations, arguably the most significant advantage is the ability to use a set of data to learn a causal structure. This task of constructing a Bayesian Network is composed of two steps:

1. Structure learning – to identify the optimal topology of the network from the data
2. Parameter learning – estimating the conditional probabilities of the variable states given the data

Computation of conditional probabilities are conditional on the structure, thus the structure must be learned first. However, structure learning is an arduous task. The search space for DAGs is combinatorial and scales super-exponentially with the number of nodes, making this an NP-hard problem. Current methods for structure learning generally fall under two categories, constraint based or score & search based methods, which will be reviewed below.

4.3.1 Constraint based methods

Constraint based methods involve specifying a set of conditional independence constraints between variables in the data, such that the output graph contains only edges that would satisfy its d-separation requirements. Examples of conditional independence tests include mutual information and Pearson's test, typically used for discrete data. The two most widely used are the PC Algorithm, which starts with fully connected graph and then iteratively removes edges based on pairwise independence tests, and the Incremental Association Markov Blanket (IAMB), which uses Markov blankets to restrict the subset of variables to test for independence.

However, the major limitation of constraint-based methods is the assumption of **faithfulness**; essentially requiring that graphical d-separation and independence are equivalent, which is to assume the given data was generated by a Bayesian

network distribution. This may not hold because of sampling variance or if variables are determined simultaneously.

As evaluated in Koski & Noble (2012), data analysis shows this assumption does not hold in a variety of real-world applications [33]. Additionally, constraint-based methods often produce solutions with undirected edges due to observational equivalence and provide no procedure for resolving this. Hence, it was decided that this class of methods will not be attempted in our experiments with the OCMR data.

4.3.2 Score-based methods

This class of methods use a scoring function to measure the goodness of fit of many DAGs with respect to the data set. Taking the score as an objective function, a search procedure is employed to find a structure which maximises this score. Unlike the constraint-based methods, conditional independence tests are not explicitly conducted, but through the optimisation of the score are encompassed implicitly. These score functions are commonly classed into two categories: Information-theoretic scoring functions and Bayesian scoring functions. The former is based on maximising the likelihood of the set of data, D, given the model, G, while the latter finds the maximum a posteriori model. For this project, we will focus on Bayesian scoring functions.

Bayesian scoring functions

Bayesian scoring functions maximise the posterior probability distribution of a particular network given the data, $P(G|D)$, using a prior probability distribution for all possible networks $P(G)$. Bayes rule states that $P(G|D) = \frac{P(D|G)P(G)}{P(D)} = \frac{P(G,D)}{P(D)}$. However, the probability of the data, $P(D)$, does not depend on the structure and cannot be evaluated, so it suffices to compute the joint probability, aka. the 'relative posterior probability', since $P(G|D) \propto P(G,D)$. The original instance of this type of score function is known as the **Bayesian Dirichlet (BD) score**, proposed by Heckerman, Geiger and Chickering (1995).

$$BD(G,D) = log(P(G)) + \sum_{i=1}^{n}\sum_{j=1}^{q_i}\left(log(\frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij}+N_{ij})}\right) + \sum_{k=1}^{r_i} log\left(\frac{\Gamma(\alpha_{ij}+N_{ij})}{\Gamma(\alpha_{ij})}\right) \tag{4.4}$$

r_i is the number of states of the random variable X_i, q_i is the number of possible configurations of the parent set of X_i, N_{ijk} is the number of observations in the data D where variable X_i takes on the k^{th} state where its parents are in their j^{th} configuration. The hyperparameter, α_{ijk} gives the a priori probability of X_i taking on its k^{th} state where its parents are in their j^{th} configuration [13]. However, setting these values for $\alpha_{ijk} \forall i,j,k$ is not feasible. Some simplification needs to be made in order for this score to be tractable; two methods that exist are the K2 Score, proposed by Cooper &Herskovits (1992), and the BDe score.

For simplicity of implementation, the K2 score is chosen in the following experiments. The K2 score is a particular instance of the BD score with the uninformative assignment of $\alpha_{ijk} = 1$, implying that, a priori, all structures are considered equally likely. Such an assignment simplifies this into the following function [14]:

$$K2(G,D) = P(G)\prod_{i=1}^{n}\prod_{j=1}^{q_i}\frac{(r_i-1)!}{(N_{ij}+r_i-1)!}\prod_{k=1}^{r_i}N_{ijk}! \tag{4.5}$$

A computational benefit of Equation 4.5 is that, under the assumption of mutual independence between variables, it is decomposable. This means that it may be expressed as a product of independent subscores for each node and its set of parents, $g(X_i, \pi_i)$ [38]. This decomposed score function will thus be implemented in Section 4.4.

$$K2(G,D) = P(G)\prod_{i=1}^{n}g(X_i,\pi_i) \tag{4.6}$$

Search Methods

Once a score function has been selected, a search procedure moves through the space of all possible networks to optimise this score. However as mentioned previously, this search space scales super-exponentially with the number of nodes, making it an NP-hard problem. To represent this, the following equation is a recursive function for computing the number of possible DAGs with d nodes [33]:

$$N(d) = \sum_{i=1}^{d}(-1)^{i+1}\binom{d}{i}2^{i(d-1)}N(d-i)$$

To conceptualise the magnitude of this, given a data set with $d = 5$ nodes, this would result in a search space of 29000 possible DAGs. For $d = 10$, however, this grows to 4.2×10^{18} DAGs. Thus, only heuristic search methods are applicable for optimising the score. Most search procedures rely on neighbourhood structure which defines a set of operators that can be used to move within the search space. These include edge addition, deletion, and reversal. It is worth noting that these search algorithms are efficient because of the decomposability property that many scoring functions exhibit, such as that found in Equation 4.6 for the K2 score. By allowing this local search procedure that only changes one arc at a time, computations can be reused for the subscores of variables in previous iterations [30].

For example, the Greedy Hill Climb Search iteratively adds, removes or reverses edges in the graph under the constraint that there are no directed cycles, checks the score and updates the graph until an optimum is found. Unfortunately, on its own this method tends to get stuck at local optima. A common solution is to include random restarts to avoid this. Another search method that will be implemented is the K2 heuristic search. This starts by assuming that a node has no parents, and then incrementally adds a parent from a given topological ordering which increases the score of the structure most. The exact implementation will be outlined in detail in Section 4.4.

4.3.3 Parameter Learning

Once an optimal structure is found, the parameters of the model can be learned - in this case it is the set of conditional probability distributions for each variable estimated using the data samples and the DAG. Using relative frequencies of each state of each variable, one approach could be to use the maximum likelihood estimation for $P(Data|DAG)$. However, this tends to overfit the data because there is unlikely to be enough observations of each state to rely on observed

frequencies. Even with a large sample size, since state counts are done conditionally for each configuration of parent states, the fragmentation results in scarce observations. Instead, Bayesian Parameter Estimation will be used. This addresses the problem by beginning with pre-existing prior CPDs, then updating these using state counts from observed data. A BDeu prior will be used in the following experiments due to its superior empirical performance [4].

4.4 An Implementation of the K2 Algorithm

The following pseudo-code summarises the implementation of the K2 score search algorithm for structure learning using manually written code with slight procedural alterations (links to the full scripts are given in Chapter 5). The code implemented will first be validated on a synthetic data set, then on both OCMR data sets.

Algorithm 2: K2 Search Algorithm, adapted from [14]

{**Input:** A set of n nodes, an initial ordering of the nodes, an upper bound, u, and lower bound, l, on the number of parents per node, and a data set D containing m observations};
{**Output:** For each node, the parents of the node is returned} ;
for $i := 1$ *to n* **do**
    $\pi_i := \emptyset$;
    $P_{old} := g(i, \pi_i)$;
    OKToProceed := True ;
    while *OkToProceed AND* $l < |\pi_i| < u$ **do**
        $z \in pred(i)$ where $z = argmax_k g(i, \pi_i \cup \{k\})$;
        $P_{new} := g(i, \pi_i \cup \{z\})$;
        if $P_{new} > P_{old}$ **then**
            $P_{old} := P_{new}$;
            $\pi_i := \pi_i \cup \{z\}$;
        else
            OKToProceed := False ;
        end
    end
end

4.4.1 Data Discretisation

Each score-based structural learning algorithm discussed thus far has been designed for discrete data. Although methods exist for dealing with continuous data, such as in Gaussian Bayesian networks which models each node as multivariate normal and $X_i|\pi_i$ as univariate normal [58]. However this method assumes a linear dependence between each node and its parents and is also computationally expensive.

Instead, since we are more interested in identifying causal dependencies, the data may be interpreted with a lower granularity by discretising it. Although methods of how best to partition the discretisation bins exist given the influence of each unit on the other variables, such as the MDL [20] this extra complexity lies beyond the scope of this project. Instead, given uniform priors are assumed it would make sense to use a quantile cut such that equal samples occur in each bin. For the K2 algorithm, the pandas `qcut` function will be used for each data set.

4.4.2 Validation data set (ASIA network) results

As many of the variables in the OCMR data sets have never been analysed in combination, especially given the novelty of PCr/ATP and cardiaclipid measures, there is no conceivable ground truth to evaluate the Bayesian network on. Therefore, in order to validate the accuracy of our structural learning algorithms, a synthetic data set generated from the ASIA Bayesian network from Lauritzen & Spiegelhalter (1988) will be used, shown in Figure 4.6. This data set consists of 10000 samples for 8 discrete variables generated using a Monte Carlo technique called probabilistic logic sampling [26].

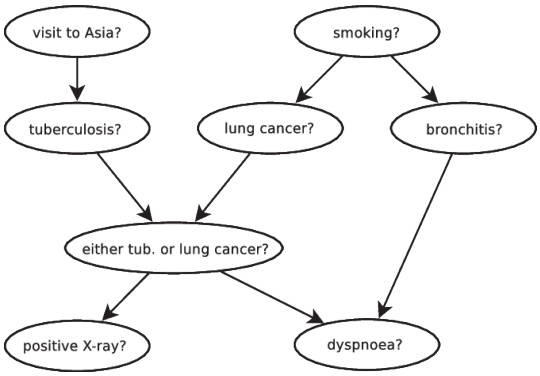
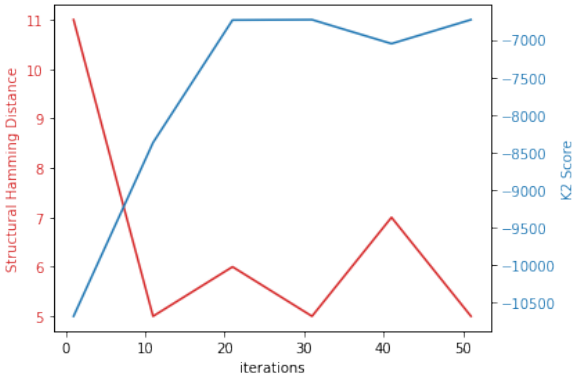
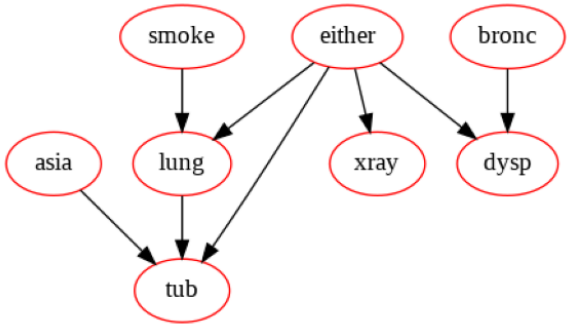


Figure 4.6: ASIA network by Lauritzen & Spiegelhalter (1988), from [16]

Performance can thus be evaluated by determining the structural hamming distance between the adjacency matrix of the generated DAG and the ground truth network. The structural hamming distance (SHD) is the number of edge insertions, deletions or reversals required to transform one graph to another. Note that code for this has also been hand-written.



(a) SHD and K2 score against no. iterations



(b) Learned network (iters=200, K2 score = -5734, SHD = 4)

Figure 4.7: Structure learning results for the ASIA network data set

As expected, as the number of iterations of the program are increased (i.e. the number of random restarts), the higher the K2 score and the lower the SHD (apart from some fluctuations due to stochasticity). This reflects how increasing range of topological orderings of the variables widens the search space, improving the score and accuracy of the output graph. After 200 iterations (arbitrarily chosen), the resulting network (Figure 4.7b) has been produced with a SHD of 4. Edge transformations required are the reversal of `(either, tub)`, `(either,lung)`, the removal of `(lung,tub)` and the addition of `(smoke,bronc)`.

4.4.3 OCMR data set 1 results

Applying this algorithm to OCMR data set 1, running the algorithm for a total of 100 iterations, the generated structure is shown in Figure 4.8. Note that two obvious edge constraints were implemented into the algorithm to speed up optimisation, namely that Age and BMI can only be child nodes in this network (no other variable in the network could be causes of these). With a K2 score of -1746.7683 and a run time of 590.63 seconds, preliminary structure learning results appear to roughly match intuitions and literature.

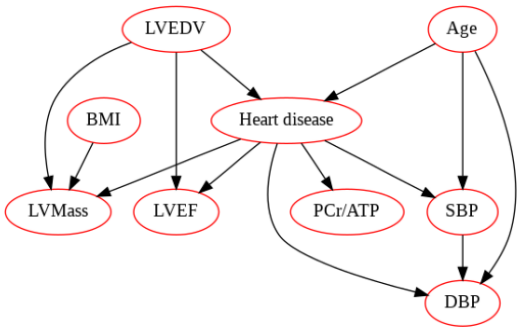


Figure 4.8: Learned Bayesian network for OCMR dataset 1 using the K2 algorithm

As previously mentioned, conditional probability distributions can be computed via a Bayesian network, i.e. parameter learning. A table has been generated for each node conditioned on its parent values, representing $P(X_i|\pi_i)$. As an example, Table 4.1 displays the conditional probabilities of each state of $P(Heart\ disease|Age,\ LVEDV)$. As expected probabilities of having heart disease generally increase with Age and LVEDV. Interestingly, when conditioned on fixed Ages, increases in LVEDV have more impact on the probabilities of the middle two quantiles of Age. Further comparison and analysis will be conducted as a whole in Section 4.5.5.

	Age(1)	Age(1)	Age(1)	Age(1)	Age(2)	Age(2)	Age(2)	Age(2)	Age(3)	Age(3)	Age(3)	Age(3)	Age(4)	Age(4)	Age(4)	Age(4)
	LVEDV(1)	LVEDV(2)	LVEDV(3)	LVEDV(4)	LVEDV(1)	LVEDV(2)	LVEDV(3)	LVEDV(4)	LVEDV(1)	LVEDV(2)	LVEDV(3)	LVEDV(4)	LVEDV(1)	LVEDV(2)	LVEDV(3)	LVEDV(4)
Heart disease (0)	0.9848	0.9923	0.9926	0.9909	0.9923	0.8176	0.6839	0.4020	0.3873	0.5969	0.2547	0.0053	0.0851	0.1011	0.0755	0.0213
Heart disease (1)	0.0151	0.0076	0.0073	0.0090	0.0076	0.1823	0.3160	0.5979	0.6126	0.4030	0.7452	0.9946	0.9148	0.8988	0.9244	0.9786

Table 4.1: Conditional probability table for P(Heart disease | Age, LVEDV)

4.5 NOTEARS Continuous Optimisation Algorithm

As explored in previous sections, traditional structural learning methods optimise scores over a combinatorial search space of DAGs which scales superexponentially. Local heuristic methods have been used to enforce acyclicity constraints, however results show this to be inefficient, generally yielding suboptimal structures. To address this problem, Zheng et al. (2018) proposed a novel method of reformulating the combinatorial structure learning problem into a continuous optimisation problem as follows.

Let F be a scoring function which inputs a DAG represented by a weighted adjacency matrix W, that is, a $d \times d$ matrix where the presence of a non-zero value indicates the presence of an edge directed from the variable of the i^{th} row to the variable of the j^{th} column. As shown in Equation 4.7, the minimisation of the score function in the constraint of being a DAG (left) is reformulated into an equality constrained continuous program (right):

$$\min_{W \in \mathbb{R}^{d\times d}} F(W) \quad \text{subject to } G(W) \in \text{DAGs} \iff \min_{W \in \mathbb{R}^{d\times d}} F(W) \quad \text{subject to } h(W) = 0 \tag{4.7}$$

Here, h is a value quantifying the extent to which a graph is DAG. $h(W) = 0$ if and only if the graph represented by W is acyclic ($G(W) \in DAGs$). As h is a smooth function, the program can then be easily solved by standard numerical solvers, in this case using an augmented Lagrangian. Referred to as NOTEARS (*Non-combinatorial Optimisation via*

Trace Exponential and Augmented lagRangian for Structure learning), the method results in a global optima, yielding superior empirical results, while also circumventing the previous limitations associated with discretising continuous data.

The following sections delve into the mathematical background, specifically the formulation of each node as structural equation models and the characterisation of acyclicity for h; then using the open source code for the NOTEARS algorithm [65], experiments are conducted on the ASIA validation data set and then finally on the OCMR data sets.

4.5.1 Random variables as Structural Equation Models (SEM)

To be able to work with continuous data, each j^{th} random variable X in the data matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ is modelled as a linear structural equation model (SEM) defined by the column vectors of $W = [w_1|...|w_d]$:

$$X_j = w_j^T X + z_j$$

Here, $z = (z_1, ...z_d)$ is a random noise vector, not assumed to be Gaussian. Applying least squares loss to this SEM, along with l1 regularisation (as we are working with sparse DAGs), the following regularised score function is defined:

$$F(W) = \frac{1}{2n}||X - XW||_F^2 + \lambda||W||_1 \tag{4.8}$$

4.5.2 The characterisation of acyclicity

In a binary adjacency matrix, $B \in \{0,1\}^{d \times d}$, where d is the dimension of the graph, acyclicity can be characterised by the following theorem:

Theorem 4 *Given that $B \in \{0,1\}^{d \times d}$ and $r(B) < 1$, then B is a DAG if and only if* $tr(I - B)^{-1} = d$

$tr()$ denotes the trace of a square matrix, defined as the sum of the elements on the main diagonal, and $r(B)$ is the spectral radius, which is the maximum absolute eigenvalue of B. The proof for this is derived from the fact that the trace, $tr(B^k)$, counts the number of closed walks of length k in a directed graph. Therefore, if a graph is acyclic then $tr(B^k) = 0 \; \forall \; k = 1, ..., \infty$. This translates to $\sum_{k=1}^{\infty} \sum_{i=1}^{d} (B^k)_{ii} = 0$. Thus, producing the following result:

$$tr(I - B)^{-1} = tr(\sum_{k=0}^{\infty} B^k) = tr(I) + tr(\sum_{k=1}^{\infty} B^k) = d + \sum_{k=1}^{\infty} \sum_{i=1}^{d} (B^k)_{ii} = d$$

However, $r(B) < 1$ is a very restrictive condition that does not hold true in most cases, and using a finite series instead would lead to numerical instability. To avoid this, an exponential form is used, reformulating it as below:

Theorem 5 *A binary matrix $B \in \{0,1\}^{d \times d}$ is a DAG if and only if* $tr(e^B) = d$

In order for h to be a smooth function, this characterisation must be extended from a discrete case to a continuous one, for weighted adjacency matrices instead of just binary matrices. Applying Theorem 5 would only work for non-negative weighted matrices. To extend this to any arbitrary W, the Hadamard product (an element-wise multiplication),$W \circ W$, is used. Thus resulting in the final characterisation used in NOTEARS:

Theorem 6 *A matrix $W \in \mathbb{R}$ is DAG if and only if:*

$$h(W) = tr\left(e^{W \circ W}\right) - d = 0 \tag{4.9}$$

Note that by replacing B with $W \circ W$, this is means to count the number of weighted closed walks instead. This function quantifies acyclicity while being smooth, generalisable, numerically stable, and possessing easily computable derivatives:

$$\nabla h(W) = (e^{W \circ W})^T \circ 2W$$

4.5.3 The optimisation algorithm

Once the smooth function h is devised, existing methods of constrained optimisation can be used; specifically the augmented Lagrangian method (Nemirovski, 1999) which adds a quadratic penalty to the score function (left of 4.10) with the penalty constant $\rho > 0$. This method approximates the constrained problem as a solution of unconstrained problems. To do so, the dual function is computed from the augmented Lagrangian L^ρ:

$$D(\alpha) = \min_{W \in \mathbb{R}^{d \times d}} L^\rho(W, \alpha)$$

Thus the program is transformed from the primal (left) to into the maximisation of the dual (right):

$$\min_{W \in \mathbb{R}^{d \times d}} F(W) + \frac{\rho}{2}|h(W)|^2 \quad \text{subject to } h(W) = 0 \iff \max_{\alpha \in \mathbb{R}} \min_{W \in \mathbb{R}^{d \times d}} L^\rho(W, \alpha), \quad \text{where } L^\rho(W, \alpha) = F(W) + \frac{\rho}{2}|h(W)|^2 + \alpha h(W) \tag{4.10}$$

Dual gradient ascent is then performed on the minimiser of the Lagrangian, denoted as W_α^*, with step size ρ. The overall procedure is outlined below in Algorithm 3.

Algorithm 3: NOTEARS Algorithm, from [66]

Input: Initial guess (W_0, α_0), progress rate $c \in (0, 1)$, tolerance $\epsilon > 0$, threshold $\omega > 0$;

for *t = 0,1,2...* **do**

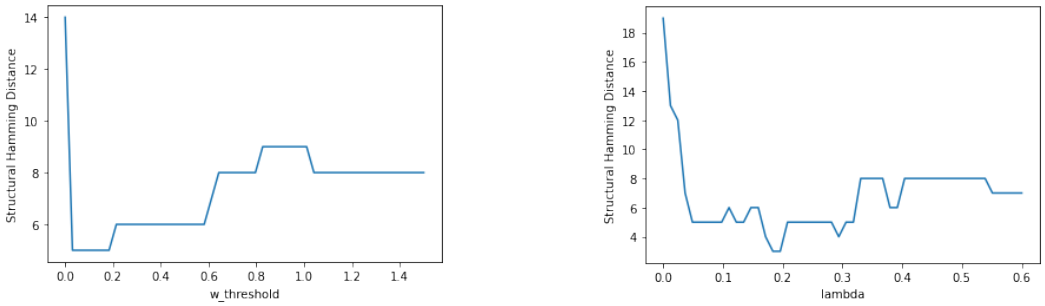
- Solve the primal $W_{t+1} \longleftarrow_W L^\rho(W, \alpha_t)$ selecting ρ such that $h(W_{t+1}) < ch(W_t)$;
- Perform dual ascent: $\alpha_{t+1} \longleftarrow \alpha_t + \rho h(W_{t+1})$;
- **if** $h(W_{t+1}) < \epsilon$ **then**
  - $\tilde{W}_{ECP} = W_{t+1}$;
  - break ;
- **end**
- Return $\hat{W} := \tilde{W}_{ECP} \circ 1(|\tilde{W}_{ECP}| > \omega)$;

end

Note that h and its gradient only involve evaluating a matrix exponential which is widely known to be $O(d^3)$, therefore the overall complexity of this algorithm is cubic with respect to the number of nodes.

The threshold, ω, is a hyperparameter which determines the threshold above which an edge will be included. To determine this, along with the regularisation constant λ, methods from Section 3.3.2 will be used. Figure 4.9 displays the effect that each of these hyperparameters has on the structural hamming distance (accuracy) of the learned structure.

The optimal parameters were found to be $\lambda = 0.18$ and $\omega = 0.03$.



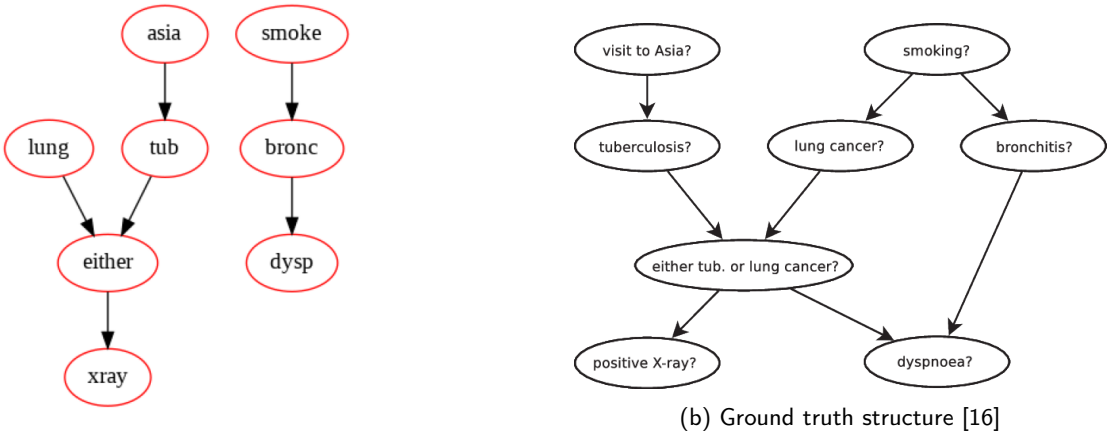
(a) Threshold ω against SHD

(b) Threshold λ against SHD

Figure 4.9: Hyperparameters against SHD for the ASIA network data set

4.5.4 Results for validation set

The resulting structure proves superior in accuracy to the previous K2 scoring method, with an SHD of 2 and missing out only the edges `(either, dysp)` and `(smoke, lung)`.



(a) NOTEARS learned structure (SHD=2)

(b) Ground truth structure [16]

Figure 4.10: Comparison of NOTEARS vs ground truth for the ASIA network data set

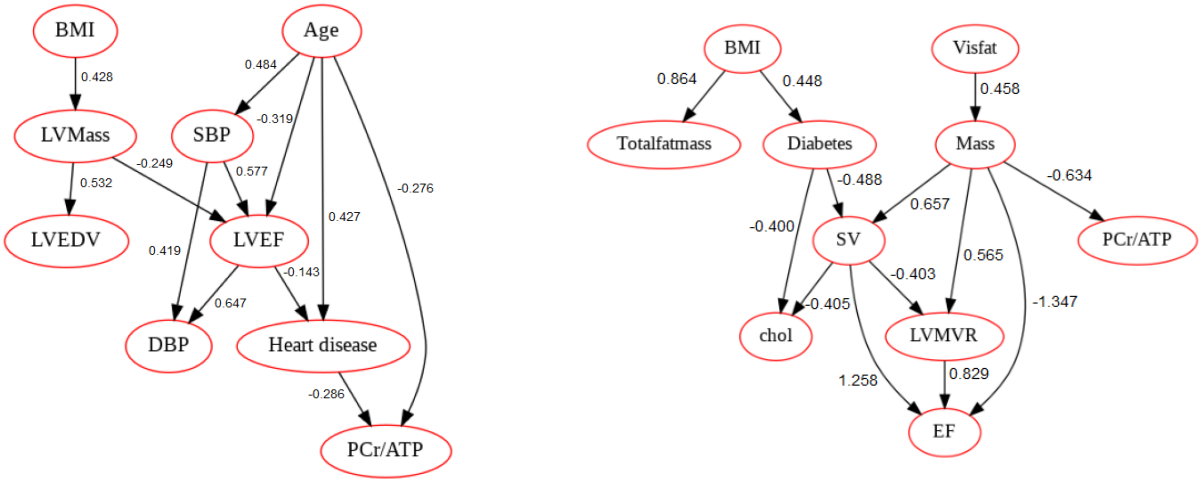
4.5.5 Results for OCMR data sets

For data set 1,a range of interesting interpretations can be formed. The structure implies that a change in BMI causes a change in LVMass, which causes a change in LVEDV and LVEF. These relationships, widely backed up by literature, provides additional evidence for the causality of the positive correlation between BMI and LVMass [52], and the previously disputed directionality (as discussed in Section 2.1.1) of increased LVMass causing increased LVEDV [21].

The weights of the resulting weighted adjacency matrix, $\hat{W}$, have also been displayed along the edges. Comparing the NOTEARS structure with the previous K2 structure, both detect heart disease to be a cause of the decrease in PCr/ATP, Age as a cause of heart disease, and BMI to be a cause of an increased LVMass. However, this interprets decreased LVEF to be a cause of heart disease instead of an effect. In reality though this is a conceptually difficult and maybe indiscernable distinction to make, i.e. in some cases heart disease is defined by decreased LVEF and in some cases

it stays constant due to other other compensatory mechanisms.

In the Diabetes data set, the extracted insights further develop the theory behind its association with LVMVR and Visfat as introduced in Section 3.3.4. The structure implies that higher Visfat and Diabetes are, in fact, independent but co-occuring variables. The cause of the increased LVMVR, as expected, is due to increased Mass and decreased SV. However, it tells us that high Visfat is the cause of the increased LVMass rather than any mechanism to do with dysfunctional insulin response. Interestingly, it also implies that increased LVMass is the direct cause of decreased PCr/ATP. This is supported by literature's claims that cardiac steatosis (associated with visceral fat) contributes to LV concentric modelling in diabetic patients and impaired myocardial energetics [39]. The chain between BMI, Diabetes and chol also reflects studies showing that diabetes modulates cholesterol metabolism more than obesity alone [59].



(a) Data set 1

(b) Data set 2 (Diabetes subset)

Figure 4.11: Learned Bayesian network structures for OCMR datasets with NOTEARS, edge weights included from $\hat{W}$

4.6 Bayesian Networks as a Classifier

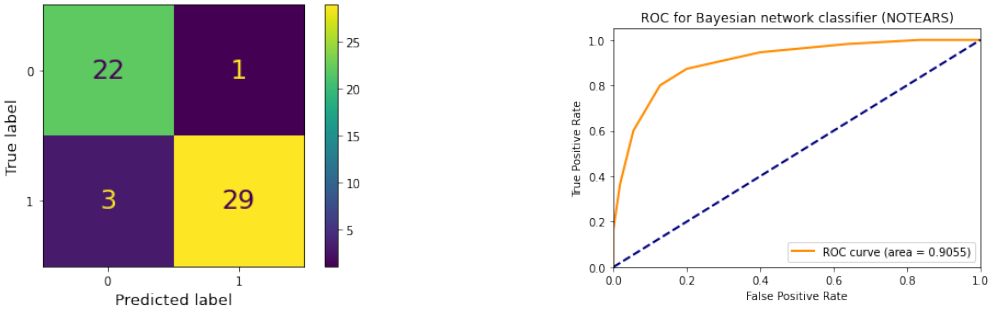
Just as was achieved in Chapter 3, a Bayesian network can also be applied to classification problems by performing probabilistic inference. Inference in Bayesian networks is essentially the task of calculating the conditional probability distribution of a subset of nodes in the DAG ('hidden' nodes), given another subset ('observed' nodes) [38]. In classification, the hidden node is the class variable. Unlike other classification models, Bayesian networks allow for predictions which do not require all data since missing values can be marginalised out as hidden variables; thus removing some of the bias attributed with the imputation methods of Section 2.3.

Denoting the class variable with the first variable $X_1 = C$ and defining $\mathbf{X} \setminus C$ and $\pi_i \setminus C$ as the set of graph nodes and parents of X_i which exclude C, the following equation is applied to classification given the graph G [38]:

$$P(C|\mathbf{X} \setminus C, G) = \prod_{i=2}^{n} P(X_i|(C, \pi_i \setminus C), G) \frac{P(C|G)}{P(\mathbf{X}|G)} \tag{4.11}$$

Transforming the Bayesian network of Figure 4.11a into a classifier, evaluating the performance on a 80:20 train-

test split, 92.72% test hold out accuracy is achieved on the OCMR data set 1 for classifying heart disease. The resulting confusion matrix and ROC curve are displayed in Figure 4.12. The ROC results indicate lower but comparable classification performance to the Random Forest in the previous chapter; this is impressive given that the input data has been discretised.



(a) Confusion Matrix

(b) ROC curve

Figure 4.12: Classification results for OCMR data set 1 using the Bayesian network learned using NOTEARS

4.6.1 Do-Intervention

A do-intervention on SBP is applied to the Bayesian network in Figure 4.11a, which essentially removes the edges from any parent nodes and sets the variable to a fixed value with all else remaining equal. By applying do-intervention, then marginalising on that variable we can calculate the Average Causal Estimate (ACE) of SBP on heart disease [47]:

$$ACE(SBP) = P(\ Heart\ disease \mid do(\ SBP\ = 10)\ )\ -\ P(\ Heart\ disease \mid do(\ SBP\ = 1)\ ) \tag{4.12}$$

This is the equivalent of inferring a counterfactual, i.e. what would happen to the probability of getting heart disease if, all else being equal, a patient's blood pressure increased or decreased greatly (e.g through blood pressure medication). Note that for the parameter learning in this experiment we have increased discretisation bin number has increased to 10, therefore this would be equivalent to increasing or decreasing the patient's SBP from above 148 to below 108. A similar computation has been performed for this same intervention on the marginal probability of PCr/ATP being below 1.38. For convenience, these computations were performed using the CausalNex python library by QuantumBlack [50].

	Before do	After do SBP = high	After do SBP = low	Average Causal Effect of SBP
Marginal probability of heart disease	0.49733	0.49850	0.49663	0.00187
Marginal probability of low PCr/ATP	0.10064	0.10081	0.10058	0.00023

Table 4.2: Marginal probabilities of heart disease and low (1) PCr/ATP before and after do-interventions on SBP

Given the very small effect of the do-interventions, the above result implies that SBP is only a very minor cause of heart disease, and less so of low PCr/ATP. In this data set, it is likely there are many important missing variables for a complete, accurate and highly predictive causal structure, e.g. lifestyle factors. Furthermore, given the exponentiality of states in parameter learning, this dataset size is likely to be insufficient. Overall, however, the capabilities of Bayesian networks have been demonstrated and suggests great potential for the application of causal inference in medicine.

Conclusion

The first half of this project implemented Decision Tree and Random Forests using Python and Sklearn libraries, achieving a test accuracy of 95.36% and an AUC of 94.69% to classify heart disease with data set 1. In line with the theoretical proofs, the generalisability of Random Forests was empirically demonstrated through experiments, as well as the utility of regularisation methods in Decision Trees. Tree diagrams and SHAP were found to be a useful and informative means of inferring medical insights, with results coinciding closely with risk thresholds from literature. However, the interpretation often lacked information about the causal interrelation to other variables.

The second half of this project developed a set of Bayesian network models to study possible causal interrelations, and apply do-intervention to simulate a counterfactual inference. Custom code for the K2 algorithm was written and implemented, which gave an SHD of 2/8 edges for the validation set. NOTEARS was then implemented from open source code, giving an SHD of 4/8. Learned structures of the OCMR datasets appear to align with medical intuition but it is likely these results are inconclusive due to limitations in the data.

A link to the full scripts for this project can be found via: https://bit.ly/2XcZJ0m

Novel contributions of this project are as follows:

- This is the first known instance of machine learning and causal inference methods being applied to patient data sets containing information about the cardiac energetics, PCR/ATP and cariac lipids (from novel 31P-MRS and 1H-MRS technology).
- The first known set of Bayesian networks for the purpose of heart disease diagnosis constructed via structure learning with data. Other attempts exist for Bayesian networks for heart disease decision support (Ghosh et al., 2000), but only through arbitrary construction via domain experts.
- This is the first found application of the NOTEARS continuous optimisation algorithm for structure learning applied to the field of medical research.

Throughout this project, an abundance of medical inferences have been made, some of which could not fit within the limits of this report. However, some notable inferences include the increased sensitivity of PCr/ATP and LVEDV to heart disease given a patient is older, higher cardiac lipids in heart failure, obesity and diabetes implying lipotoxicity, increased LVMVR and visceral fat in diabetics, and a small but detected causal relationship between SBP and heart disease. Many inferences have thus supported the informativeness of CMRS metrics substantiating its potential as a diagnostic tool.

Future Work

Unfortunately, due to the time constraints, more advanced techniques for causal inference could not be attempted. As a suggestion for future work, methods for the identification and adjustment for unobserved confounders could be performed - for example, the back-door criterion and adjustment methods. Through this, more confident conclusions may be drawn about causal relationships.

Bibliography

[1] *1.10. Decision Trees*. URL: https://scikit-learn.org/stable/modules/tree.html#tree.

[2] *1.13. Feature selection*. URL: https://scikit-learn.org/stable/modules/feature_selection.html#rfe.

[3] *About OCMR*. URL: https://www.rdm.ox.ac.uk/about/our-clinical-facilities-and-mrc-units/oxford-centre-for-clinical-magnetic-resonance-research/about-ocmr.

[4] Ankur Ankan. *Learning Bayesian Networks*. URL: https://github.com/pgmpy/pgmpy_notebook/blob/master/notebooks/9.%20Learning%20Bayesian%20Networks%20from%20Data.ipynb.

[5] James Bergstra and Yoshua Bengio. "Random Search for Hyper-Parameter Optimization". In: *J. Mach. Learn. Res.* 13.null (Feb. 2012), pp. 281–305. ISSN: 1532-4435.

[6] Barry A. Borlaug and Walter J. Paulus. "Heart failure with preserved ejection fraction: pathophysiology, diagnosis, and treatment". In: *European Heart Journal* 32.6 (Dec. 2010), pp. 670–679. ISSN: 0195-668X. DOI: 10.1093/eurheartj/ehq426. eprint: https://academic.oup.com/eurheartj/article-pdf/32/6/670/1338872/ehq426.pdf. URL: https://doi.org/10.1093/eurheartj/ehq426.

[7] Paul A. Bottomley. "NMR Spectroscopy of the Human Heart". In: *eMagRes*. American Cancer Society, 2009. ISBN: 9780470034590. DOI: 10.1002/9780470034590.emrstm0345.pub2. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/9780470034590.emrstm0345.pub2. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470034590.emrstm0345.pub2.

[8] Leo Breiman. "Random Forests". In: *Mach. Learn.* 45.1 (Oct. 2001), pp. 5–32. ISSN: 0885-6125. DOI: 10.1023/A:1010933404324. URL: https://doi.org/10.1023/A:1010933404324.

[9] Ibadete Bytyçi and Gani Bajraktari. "Mortality in heart failure patients." In: *Anatolian Journal of Cardiology* (). DOI: 10.5152/akd.2014.5731.

[10] Blase A Carabello and Walter J Paulus. "Aortic stenosis". In: *The Lancet* 373.9667 (2009), pp. 956–966. ISSN: 0140-6736. DOI: https://doi.org/10.1016/S0140-6736(09)60211-7. URL: http://www.sciencedirect.com/science/article/pii/S0140673609602117.

[11] *Cardiovascular diseases (CVDs)*. URL: https://www.who.int/en/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds).

[12] Rich Caruana and Alexandru Niculescu-Mizil. "An Empirical Comparison of Supervised Learning Algorithms". In: *Proceedings of the 23rd International Conference on Machine Learning*. ICML '06. Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, 2006, pp. 161–168. ISBN: 1595933832. DOI: 10.1145/1143844.1143865. URL: https://doi.org/10.1145/1143844.1143865.

[13] Alexandra M Carvalho. *Scoring functions for learning Bayesian networks*. URL: http://www.lx.it.pt/~asmc/pub/talks/09-TA/ta_pres.pdf.

[14] Gregory F. Cooper and Edward Herskovits. "A Bayesian Method for the Induction of Probabilistic Networks from Data". In: *Mach. Learn.* 9.4 (Oct. 1992), pp. 309–347. ISSN: 0885-6125. DOI: 10.1023/A:1022649401552. URL: https://doi.org/10.1023/A:1022649401552.

[15] Matthew J. Czarny and Jon R. Resar. "Diagnosis and Management of Valvular Aortic Stenosis". In: *Clinical Medicine Insights: Cardiology* 8s1 (2014), CMC.S15716. DOI: 10.4137/CMC.S15716. eprint: https://doi.org/10.4137/CMC.S15716. URL: https://doi.org/10.4137/CMC.S15716.

[16] Rónán Daly, Qiang Shen, and Stuart Aitken. "Learning Bayesian networks: Approaches and issues". In: *Knowledge Eng. Review* 26 (June 2011), pp. 99–157. DOI: 10.1017/S0269888910000251.

[17] Andrew L. Dannenberg, Daniel Levy, and Robert J. Garrison. "Impact of age on echocardiographic left ventricular mass in a healthy population (the Framingham study)". In: *The American Journal of Cardiology* 64.16 (1989), pp. 1066–1068. ISSN: 0002-9149. DOI: https://doi.org/10.1016/0002-9149(89)90816-3. URL: http://www.sciencedirect.com/science/article/pii/0002914989908163.

[18] Thomas Davenport and Ravi Kalakota. "The potential for artificial intelligence in healthcare". In: *Future Healthcare Journal* 6.2 (2019), pp. 94–98. ISSN: 2514-6645. DOI: 10.7861/futurehosp.6-2-94. eprint: https://www.rcpjournals.org/content/6/2/94.full.pdf. URL: https://www.rcpjournals.org/content/6/2/94.

[19] Maurice Enriquez-Sarano, Vuyisile T. Nkomo, and Hector I. Michelena. "Mitral Regurgitation". In: *Valvular Heart Disease*. Ed. by Andrew Wang and Thomas M. Bashore. Totowa, NJ: Humana Press, 2009, pp. 221–246. ISBN: 978-1-59745-411-7. DOI: 10.1007/978-1-59745-411-7_10. URL: https://doi.org/10.1007/978-1-59745-411-7_10.

[20] Nir Friedman and Moisés Goldszmidt. "Discretizing Continuous Attributes While Learning Bayesian Networks". In: *ICML*. 1996, pp. 157–165.

[21] Sonia Garg et al. "Association of Concentric Left Ventricular Hypertrophy With Subsequent Change in Left Ventricular End-Diastolic Volume: The Dallas Heart Study". In: *Circulation: Heart Failure* 10 (Aug. 2017), e003959. DOI: 10.1161/CIRCHEARTFAILURE.117.003959.

[22] Aurlien Gron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 1st. O'Reilly Media, Inc., 2017. ISBN: 1491962291.

[23] David Hand. "Measuring Classifier Performance: A Coherent Alternative to the Area Under the ROC Curve". In: *Machine Learning* 77 (Oct. 2009), pp. 103–123. DOI: 10.1007/s10994-009-5119-5.

[24] J A Hanley and B J McNeil. "The meaning and use of the area under a receiver operating characteristic (ROC) curve." In: *Radiology* 143.1 (1982). PMID: 7063747, pp. 29–36. DOI: 10.1148/radiology.143.1.7063747. eprint: https://doi.org/10.1148/radiology.143.1.7063747. URL: https://doi.org/10.1148/radiology.143.1.7063747.

[25] Christopher J. Hardy et al. "Altered myocardial high-energy phosphate metabolites in patients with dilated cardiomyopathy". In: *American Heart Journal* 122.3, Part 1 (1991), pp. 795–801. ISSN: 0002-8703. DOI: https://doi.org/10.1016/0002-8703(91)90527-O. URL: http://www.sciencedirect.com/science/article/pii/0002870391905270.

[26] Max HENRION. "Propagating Uncertainty in Bayesian Networks by Probabilistic Logic Sampling". In: *Uncertainty in Artificial Intelligence*. Ed. by John F. LEMMER and Laveen N. KANAL. Vol. 5. Machine Intelligence and Pattern Recognition. North-Holland, 1988, pp. 149–163. DOI: https://doi.org/10.1016/B978-0-444-70396-5.50019-4. URL: http://www.sciencedirect.com/science/article/pii/B9780444703965500194.

[27] *Hold-out vs. Cross Validation in Machine Learning*. URL: https://medium.com/@eijaz/holdout-vs-cross-validation-in-machine-learning-7637112d3f8f.

[28] Michiel Hove and Stefan Neubauer. "Evaluating metabolic changes in heart disease by magnetic resonance spectroscopy". In: *Heart and Metabolism* (Jan. 2006), pp. 18–21.

[29] *How to derive the probabilistic interpretation of the AUC?* Aug. 2017. URL: https://stats.stackexchange.com/questions/180638/how-to-derive-the-probabilistic-interpretation-of-the-auc.

[30] Luis Miguel de Campos Ibáñez. *DAGs and Equivalence Classes of DAGs*. May 2003. URL: https://www.cs.cmu.edu/afs/cs/project/jair/pub/volume18/acid03a-html/node2.html.

[31] Satya Jonnalagadda, Robert Skinner, and Leah Moore. "Overweight athlete: Fact or fiction?" In: *Current sports medicine reports* 3 (Sept. 2004), pp. 198–205. DOI: 10.1007/s11932-004-0016-y.

[32] Ron Kohavi. "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection". In: 14 (Mar. 2001).

[33] Timo Koski and John Noble. "A Review of Bayesian Networks and Structure Learning". In: *Mathematica Applicanda*. Annales Societatis Mathematicae Polonae, Series III 40.1 (2012), pp. 51–103. ISSN: 2299-4009. URL: https://wydawnictwa.ptm.org.pl/index.php/matematyka-stosowana/article/view/278.

[34] *L19: Random Forest Math*. URL: http://math.bu.edu/people/mkon/MA751/L19RandomForestMath.pdf.

[35] Steffen L. Lauritzen. *Graphical Models*. Oxford University Press, 1996. ISBN: 0-19-852219-3.

[36] Alexander Lavin. *Healthcare Needs AI, AI Needs Causality*. URL: https://www.forbes.com/sites/alexanderlavin/2019/08/13/healthcare-needs-ai-ai-needs-causality/.

[37] Jin Hyuk Lee and John Huber Jr. *Multiple imputation with large proportions of missing data: How much is too much?* United Kingdom Stata Users' Group Meetings 2011 23. Stata Users Group, Sept. 2011. URL: https://ideas.repec.org/p/boc/usug11/23.html.

[38] Boaz Lerner and Roy Malka. "Investigation of the K2 Algorithm in Learning Bayesian Network Classifiers." In: *Applied Artificial Intelligence* 25 (Jan. 2011), pp. 74–96. DOI: 10.1080/08839514.2011.529265.

[39] Eylem Levelt et al. "Relationship Between Left Ventricular Structural and Metabolic Remodeling in Type 2 Diabetes". In: *Diabetes* 65.1 (2016), pp. 44–52. ISSN: 0012-1797. DOI: 10.2337/db15-0627. eprint: https://diabetes.diabetesjournals.org/content/65/1/44.full.pdf. URL: https://diabetes.diabetesjournals.org/content/65/1/44.

[40] Roderick J A Little and Donald B Rubin. *Statistical Analysis with Missing Data*. USA: John Wiley Sons, Inc., 1986. ISBN: 0471802549.

[41] Gilles Louppe et al. "Understanding variable importances in Forests of randomized trees". In: vol. 26. Dec. 2013.

[42] Scott M Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 4765–4774. URL: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

[43] Masliza Mahmod et al. "Myocardial perfusion and oxygenation are impaired during stress in severe aortic stenosis and correlate with impaired energetics and subclinical left ventricular dysfunction". In: *Journal of cardiovascular magnetic resonance : official journal of the Society for Cardiovascular Magnetic Resonance* 16 (Apr. 2014), p. 29. DOI: 10.1186/1532-429X-16-29.

[44] M R Malinow, A Perley, and P McLaughlin. "Muscular exercise and cholesterol degradation: mechanisms involved." In: *Journal of Applied Physiology* 27.5 (1969). PMID: 5360440, pp. 662–665. DOI: 10.1152/jappl.1969.27.5.662. eprint: https://doi.org/10.1152/jappl.1969.27.5.662. URL: https://doi.org/10.1152/jappl.1969.27.5.662.

[45] Arend Mosterd and Arno W Hoes. "Clinical epidemiology of heart failure". In: *Heart* 93.9 (2007), pp. 1137–1146. ISSN: 1355-6037. DOI: 10.1136/hrt.2003.025270. eprint: https://heart.bmj.com/content/93/9/1137.full.pdf. URL: https://heart.bmj.com/content/93/9/1137.

[46] Judea Pearl. *Causality: Models, Reasoning and Inference*. 2nd. USA: Cambridge University Press, 2009. ISBN: 052189560X.

[47] Judea Pearl and Dana Mackenzie. *The Book of Why: The New Science of Cause and Effect*. 1st. USA: Basic Books, Inc., 2018. ISBN: 046509760X.

[48] *Permutation Importance vs Random Forest Feature Importance (MDI)*¶. URL: https://scikit-learn.org/stable/auto_examples/inspection/plot_permutation_importance.html.

[49] *Post pruning decision trees with cost complexity pruning*¶. URL: https://scikit-learn.org/stable/auto_examples/tree/plot_cost_complexity_pruning.html#sphx-glr-auto-examples-tree-plot-cost-complexity-pruning-py.

[50] QuantumBlack. *CausalNex*. URL: https://github.com/quantumblacklabs/causalnex.

[51] Laura Elena Raileanu and Kilian Stoffel. "Theoretical Comparison between the Gini Index and Information Gain Criteria". In: *Annals of Mathematics and Artificial Intelligence* 41.1 (May 2004), pp. 77–93. ISSN: 1012-2443. DOI: 10.1023/B:AMAI.0000018580.96245.c6. URL: https://doi.org/10.1023/B:AMAI.0000018580.96245.c6.

[52] Munir Ahmad Rashid et al. "Impact of body mass index on left ventricular mass." In: *Journal of Ayub Medical College, Abbottabad : JAMC* 26 2 (2014), pp. 167–9.

[53] Oliver Rider et al. "Gender-specific differences in left ventricular remodelling in obesity: Insights from cardiovascular magnetic resonance imaging". In: *European heart journal* 34 (Oct. 2012). DOI: 10.1093/eurheartj/ehs341.

[54] Donald B. Rubin. "Inference and Missing Data". In: *Biometrika* 63.3 (1976), pp. 581–592. ISSN: 00063444. URL: http://www.jstor.org/stable/2335739.

[55] Mihaela van der Schaar. *Classification and Regression Trees*. URL: http://www.stats.ox.ac.uk/~flaxman/HT17_lecture13.pdf.

[56] Joseph Schafer and John Graham. "Missing Data: Our View of the State of the Art". In: *Psychological Methods* 7 (June 2002), pp. 147–177. DOI: 10.1037/1082-989X.7.2.147.

[57] P. Christian Schulze, Konstantinos Drosatos, and Ira J. Goldberg. "Lipid Use and Misuse by the Heart". In: *Circulation Research* 118.11 (2016), pp. 1736–1751. DOI: 10.1161/CIRCRESAHA.116.306842. eprint: https://www.ahajournals.org/doi/pdf/10.1161/CIRCRESAHA.116.306842. URL: https://www.ahajournals.org/doi/abs/10.1161/CIRCRESAHA.116.306842.

[58] Marco Scutari. *Bayesian Network Modelling*. Nov. 2016. URL: https://www.bnlearn.com/about/slides/slides-ibm16.pdf.

[59] Piia P. Simonen, Helena K. Gylling, and Tatu A. Miettinen. "Diabetes Contributes to Cholesterol Metabolism Regardless of Obesity". In: *Diabetes Care* 25.9 (2002), pp. 1511–1515. ISSN: 0149-5992. DOI: 10.2337/diacare.25.9.1511. eprint: https://care.diabetesjournals.org/content/25/9/1511.full.pdf. URL: https://care.diabetesjournals.org/content/25/9/1511.

[60] Devin Soni. *Introduction to Bayesian Networks*. URL: https://towardsdatascience.com/introduction-to-bayesian-networks-81031eeed94e.

[61] Daniele Soria et al. "A 'non-parametric' version of the naive Bayes classifier". In: *Knowl.-Based Syst.* 24 (Aug. 2011), pp. 775–784. DOI: 10.1016/j.knosys.2011.02.014.

[62] William Stanley and Margaret Chandler. "Energy Metabolism in the Normal and Failing Heart: Potential for Therapeutic Interventions". In: *Heart failure reviews* 7 (May 2002), pp. 115–30. DOI: 10.1023/A:1015320423577.

[63] Rebecca C. Steorts. *STA 325, Chapter 5 ISL*. 2017. URL: http://www2.stat.duke.edu/~rcs46/lectures_2017/05-resample/05-cv.pdf.

[64] Philipp Tschandl et al. "Comparison of the accuracy of human readers versus machine-learning algorithms for pigmented skin lesion classification: an open, web-based, international, diagnostic study". In: *The Lancet Oncology* 20.7 (2019), pp. 938–947. ISSN: 1470-2045. DOI: https://doi.org/10.1016/S1470-2045(19)30333-X. URL: http://www.sciencedirect.com/science/article/pii/S147020451930333X.

[65] Xunzheng. *xunzheng/notears*. Apr. 2020. URL: https://github.com/xunzheng/notears.

[66] Xun Zheng et al. *DAGs with NO TEARS: Continuous Optimization for Structure Learning*. 2018. arXiv: 1803.01422 [stat.ML].