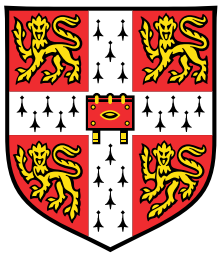
Developing tailored approaches from multi-arm randomised trials with an application to blood donation



Yuejia Xu

MRC Biostatistics Unit
University of Cambridge

This thesis is submitted for the degree of
Doctor of Philosophy

Homerton College July 2020

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this thesis are original and have not been submitted in whole or in part for consideration for any other degree or qualification in University of Cambridge or any other university. This thesis is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This thesis contains fewer than 60000 words.

Yuejia Xu
July 2020

Developing tailored approaches from multi-arm randomised trials with an application to blood donation

Yuejia Xu

Abstract

There is a growing interest in personalised medicine where individual heterogeneity is incorporated into decision-making and treatments are tailored to individual patients or patient subgroups in order to provide better healthcare. The National Health Service Blood and Transplant (NHSBT) in England aims to move towards a more personalised service and the National Institute for Health Research Blood and Transplant Research Unit (NIHR BTRU) call has mandated research to "identify, characterise and exploit biomarkers in personalising donation strategies to maximise donor health and the blood supply". The work presented in this thesis was motivated by a large-scale, UK-based blood donation trial called INTERVAL. In INTERVAL, male donors were randomly assigned to 12-week, 10-week, and 8-week inter-donation intervals, and female donors to 16-week, 14-week, and 12-week inter-donation intervals. The outcomes of this trial include the amount of blood collected (primary), the number of low haemoglobin deferrals, and donor's quality of life. The INTERVAL trial has collected a wealth of information on individual donor characteristics, enabling us to explore (i) whether different inter-donation intervals should be recommended for donors with different characteristics (by age, blood measurements, etc.), and (ii) donor stratification schemes, for example, how to partition donors into those who have the capacity to give blood more frequently than the general donor population and those who tend to be deferred more often due to safety concerns than the average donors.

One of the main statistical challenges arising from the development of personalised donation strategies using the data from the INTERVAL trial is that there are three (ordered) randomised groups for each gender in this trial, while the

majority of existing statistical approaches developed in the personalised medicine context can only handle two randomised groups and thus are not directly applicable to the INTERVAL data. This thesis aims to address issues related to this added methodological complexity. We hope that the methodologies developed in this thesis can not only help us better analyse the INTERVAL data but also facilitate the analysis of other multi-arm trials in a wider range of medical applications in addition to blood donation.

We begin by summarising methods that can be used to estimate the optimal individualised treatment rule (ITR) in multi-arm trials and comparing their performance in large-scale trials via simulation studies in Chapter 2. We also apply these methods to the data from male donors in the INTERVAL trial to estimate the optimal personalised donation strategies under three different objectives: (i) maximise the total units of blood collected by the blood service, (ii) minimise the low haemoglobin deferral rates, and (iii) maximise a utility score that "discounts" the total units of blood collected by the incidences of low haemoglobin deferrals.

The three inter-donation intervals in the INTERVAL trial exhibit a natural ordering, and applying the ITR estimation methods that ignore the ordinality may result in suboptimal decisions. We are thus motivated to propose a method that effectively incorporates information on the ordinality of randomised groups to identify the optimal ITR in the ordinal-arm setting in Chapter 3. We further develop variable selection methods under the proposed framework to handle situations with noise covariates that are irrelevant for decision-making. Through simulation studies and an application to the data from a target donor population ("much-in-demand but vulnerable") in the INTERVAL trial, we demonstrate that the proposed method has superior performance over methods that ignore the ordinality.

In Chapter 4, we switch focus to donor (or "patient" in a more general sense) stratification in multi-arm trials and develop a novel method for stratifying subjects with heterogeneous intervention responses and covariate profiles into more homogeneous subgroups using Bayesian clustering techniques. The "imputed" potential outcomes under different randomised groups are linked to subjects' baseline characteristics nonparametrically through cluster (subgroup) memberships. We examine the performance of our proposed method via simulation studies and we illustrate the utility of the method by applying it to the INTERVAL data to stratify donors based on their capacity to donate.

To my parents.

Acknowledgements

I would first like to express my sincerest gratitude to my supervisor Dr. Brian Tom for his constant encouragement, patient guidance and enormous support in the past three years, without which this thesis would not have been possible. He has been extremely generous with his time and always provides prompt feedback on my work, for which I am deeply grateful.

Besides my supervisor, I would like to thank my advisers, Dr. Jessica Barrett, Dr. Li Su, and Dr. Angela Wood for their helpful advice on my research. Thanks also to my collaborators Dr. Michael Sweeting and Prof. David Roberts for their thoughtful inputs on Chapter 2 in this thesis. In addition, I thank Dr. Stephen Kaptoge and Thomas Bolton for patiently answering all my questions related to the INTERVAL data.

I greatly appreciate the friendly and supportive working environment provided by the MRC Biostatistics Unit, and I thank Cambridge Trust for funding my PhD with the Cambridge International Scholarship.

My gratitude also goes to all my friends in and outside Cambridge for the great times we spent together.

Finally, I would like to thank my family, especially my parents, Duncheng and Tao, to whom this thesis is dedicated, for their unconditional love and unwavering support.

Table of contents

List of figures

List of tables

Nomenclature

Acronyms

ACWL Adaptive Contrast Weighted Learning

AIPWE Augmented Inverse Probability Weighted Estimator

AOL Augmented Outcome Weighted Learning

ARI Adjusted Rand Index

BART Bayesian Additive Regression Trees

BMI Body Mass Index

BTRU Blood and Transplant Research Unit

CART Classification and Regression Trees

COSSO Component Selection and Smoothing Operator

COVID-19 Coronavirus Disease

DP Dirichlet Process

DPMM Dirichlet Process Mixture Model

DTR Dynamic Treatment Regime

GBT Gradient Boosting Trees

GL Group LASSO

Hb Haemoglobin

HGL Hierarchical Group LASSO

IT Interaction Trees

ITR Individualised Treatment Rule

KKT Karush–Kuhn–Tucker

LASSO Least Absolute Shrinkage and Selection Operator

MCH Mean Corpuscular Haemoglobin

MCMC Markov Chain Monte Carlo

MCS Mental Component Score

MCV Mean Corpuscular Volume

MSE Mean Squared Errors

MVN Multivariate Normal

NHSBT National Health Service Blood and Transplant

NIHR National Institute for Health Research

NIW Normal-Inverse-Wishart

NP Non-deterministic Polynomial-time

NUC No Unmeasured Confounder

OSM One-Side Monotonicity

OvO One-versus-One

OvR One-versus-Rest

OWL Outcome Weighted Learning

PAM Partitioning Around Medoids

PC Principal Component

PCS Physical Component Score

PLS Penalised Least Squares

RBC Red Blood Cell

RBF Radial Basis Function

RF Random Forest

RI Rand Index

RKHS Reproducing Kernel Hilbert Space

R-step Re-estimation Step

RWL Residual Weighted Learning

SCAD Smoothly Clipped Absolute Deviation

SD Standard Deviation

SE Standard Error

SF-36v2 Short Form Health Survey Version 2

SMART Sequential Multiple Assignment Randomised Trial

SODA Stepwise Conditional Likelihood Variable Selection for Discriminant Analysis

SR Sequential Re-estimation

S-step Sequential Step

SVM Support Vector Machine

VT Virtual Twin

WBC White Blood Cell

Chapter 1

Introduction

1.1 Overview of personalised medicine

Personalised medicine is an emerging field in the new era of healthcare with the rapid and continuing advancement in technologies for collecting patient-level data and better characterising each individual patient. In clinical practice, it has been well-recognised that responses to treatments can vary substantially due to the heterogeneity in patient populations. Therefore, the treatment that is regarded as the best for a patient with one set of characteristics might not be the best for another, and the traditional "one-size-fits-all" approach does not lead to optimal decisions in most cases. Instead, individualised, evidence-based clinical decision making strategies that account for such heterogeneity are more desirable and have gained much popularity in medical research recently.

The goal of personalised medicine is to improve patient outcomes and provide better healthcare by tailoring treatment to the individual patient based on observed patient characteristics, such as demographic information, clinical measurements, medical history, and genetic information. By administering treatment only if it is needed (reduce the use of ineffective treatments for patients), the cost, treatment burden and treatment side effects can be largely reduced (Ginsburg and Phillips, 2018).

We note that conceptually speaking, the terms "personalised medicine", "stratified medicine", and "precision medicine" refer to an overlapping phenomenon (i.e. give the right treatment to the right patient at the right time), and are often used interchangeably in the literature (Kosorok and Laber, 2019). Erikainen and Chan (2019) and Kennedy (2020) discussed the nuances of these three terms from political and scientific perspectives. In this thesis, we use the word "personalised

medicine" as a cover term, which we think fits better to the context of our motivating example (the motivating example will be introduced later in Section 1.3) and the overall theme of this work.

1.2 Two perspectives on personalised medicine

There are two perspectives on personalised medicine: (i) the estimation of the optimal individualised treatment rule (ITR), and (ii) the subgroup identification/patient stratification. We refer readers to Lipkovich et al. (2017) for a detailed discussion on connections and comparisons between the two perspectives. In either case, there is a need to develop data-driven approaches in order to determine how and in which combination "tailoring variables" can be used for selecting the optimal treatment or stratifying patients into clinically meaningful subgroups.

1.2.1 Estimation of the optimal ITR

The first perspective aims at identifying the best treatment for a given patient based on patient-specific characteristics, that is, the estimation of the optimal ITR. The ITR estimation aspect of personalised medicine commonly reflects the views of patients and policy makers. This is a major component of personalised medicine and is conceptualised as finding the optimal individualised treatment among several treatment options at either a single or multiple decision points. There has been a considerable amount of literature focusing on this matter (Qian and Murphy, 2011; Zhang et al., 2012b; Zhao et al., 2012).

One way to estimate the optimal ITR is to first model conditional mean outcomes or treatment contrasts, and then use the model to derive the optimal ITR (Qian and Murphy, 2011; Tian et al., 2014). These methods are considered as indirect methods of estimation given that the optimal ITRs are inferred from the regression model. The success of these indirect approaches is highly dependent on the correct specification of the posited models (and the precision of model estimates), and a misspecified regression model may result in suboptimal ITRs. In contrast, the other paradigm (i.e. direct method) estimates the optimal ITR by targeting the maximisation (assume a larger outcome is more desirable) of the expected clinical outcome directly without the need for modelling conditional mean outcomes or treatment contrasts as with indirect approaches (Qi and Liu, 2018; Zhang et al., 2012a; Zhao et al., 2012; Zhou et al., 2017). These direct

methods have drawn a lot of attention lately as they offer more protection against model misspecification than indirect approaches. We will discuss indirect and direct ITR estimation methods in more detail in the next chapter.

1.2.2 Subgroup identification/patient stratification

The second perspective on personalised medicine aims at identifying the right patients who are likely to benefit from a particular treatment, which is conceptualised as subgroup identification. Under this perspective, it is of interest to find the subgroup/subgroups of patients (who share similar characteristics) with elevated treatment effects compared to the overall population (i.e. the first type of research question discussed by Sies et al. (2019)), especially when there is substantial treatment effect heterogeneity. In some situations, interests lie in stratifying/partitioning the heterogeneous patient population into smaller subpopulations that are more homogeneous in terms of treatment responses and covariate profiles (i.e. the second type of research question discussed by Sies et al. (2019)) instead of identifying the subgroup/subgroups with enhanced treatment effects. Since this type of analysis (patient stratification) is on the subgroup level, it also belongs to the second perspective on personalised medicine. We do not make further distinctions between subgroup identification and patient stratification, and will use these two terms interchangeably in the following discussion. Interested readers are referred to Sies et al. (2019) for a detailed comparison between them. The subgroup discovery aspect of personalised medicine typically reflects the views of pharmaceutical companies. The identification of clinically relevant subgroups can facilitate the development of targeted (or tailored) therapies and the selection of target population for future investigation, thus improving clinical effectiveness.

Traditional approaches to subgroup analyses involve assessing treatment effects in subgroups defined by patients' baseline characteristics (Wang et al., 2007). Such analyses can be prespecified (analyses are planned before data collection and subgroups being examined are specified in the study protocol) or post-hoc (hypotheses being tested are not specified before the inspection of the data). A well-recognised concern of post-hoc subgroup analyses is that the reported statistically significant post-hoc results may suffer from selective reporting bias since it is usually unclear how many hypotheses have been tested and whether some of them are driven by the examination of the data (i.e. the problem of "data dredging"). Post-hoc analyses can also be problematic due

to insufficient sample sizes, and thus insufficient power to detect differences of treatment effects between subgroups. Prespecified subgroup analyses are regarded as more reliable than post-hoc subgroup analyses (Moher et al., 2010). However, prespecified analyses are not guaranteed to be appropriate, and they can be misleading if subgroups are wrongly defined, especially when true subgroups are determined by multiple baseline characteristics simultaneously.

A second approach which performs subgroup analyses by testing interactions between the treatment and baseline characteristics in regression models (subgroups are identified based on the significance of interactions terms) can also be unreliable since accurate identification of subgroups depends highly on the correct specification of the regression model, and model misspecification is common especially in high-dimensional settings. In addition, multiplicity issues arise when multiple subgroup analyses are performed: an increase in the number of tests for interactions would lead to an increased probability of getting false positive results, thereby reducing the reliability of positive findings (Burke et al., 2015; Lipkovich et al., 2018). Therefore, extra caution should be taken when examining the interactions (e.g. proper adjustments of the statistical significance criterion should be made) and interpreting the results in order to ensure the credibility and replicability of findings.

The aforementioned methods for subgroup analyses may lead to spurious results, and this calls for the development of principled and statistically-justified approaches that result in valid statistical inferences. Recently, a number of data-driven methods have been developed for finding meaningful subgroups (Foster et al., 2011; Su et al., 2009; Ting et al., 2020). Based on the taxonomy proposed by Lipkovich et al. (2017), these methods can be classified into 3 groups: global outcome modelling methods (e.g. virtual twins proposed by Foster et al. (2011)), global treatment effect modelling methods (e.g. interaction trees proposed by Su et al. (2009)), and local modelling methods (e.g. differential effect search method proposed by Lipkovich et al. (2011)). A comprehensive list of methods in each category and more details on each method can be found in Lipkovich et al. (2017).

1.3 The INTERVAL trial

As a result of population ageing and the difficulty in attracting and retaining young donors, the blood supply is likely to decrease in the long term while there

is an increasing demand for blood transfusions in England (Di Angelantonio et al., 2017). One way to maintain the blood supply is to encourage existing donors to donate more often. However, adequate time is required between blood donations to allow the body to replace its iron stores and recover its haemoglobin (Hb) levels. A donor is able to give blood only if he/she passes the routine health screening haemoglobin finger-prick test (minimum thresholds to donate in England are 135g/L for men and 125g/L for women). This ensures that the donor's Hb level is safe for donation, otherwise, the donor will be deferred (temporary suspension of donors from giving blood) for a given period of time. In routine National Health Service Blood and Transplant (NHSBT) practice, males with Hb levels between 125-134g/L and females with Hb levels between 115-124g/L are deferred for a period of 3 months, while males with Hb levels <125g/L and females with Hb levels <115g/L are deferred for a period of 12 months. It is likely that more frequent donations lead to a higher risk of deferrals for low Hb. Such deferrals have multiple adverse consequences: they waste the time of donors and NHSBT staffs, they increase the costs for NHSBT, they demotivate donors to attend future donation sessions, and they may be an indicator of emerging health or quality of life problems for donors. Therefore, there exist some limits on the frequency of whole blood donation in order to reduce the occurrences of low Hb deferrals, safeguard donor health and maintain the quality of blood collected. However, there is substantial variation across blood services in different countries in terms of the maximum frequency of donations allowed due to a lack of evidence on the optimal interval between blood donations. The NHSBT in England currently allows a minimum inter-donation interval of 12 weeks for males and 16 weeks for females, with shorter inter-donation intervals used in other countries (e.g. the USA and France) (Goldman et al., 2016; Karp and King, 2010; Vuk et al., 2017). The UK blood service is interested in investigating whether or not intervals between blood donations in the UK can be decreased to optimise blood supply while maintaining donor health. To this end, the INTERVAL trial was set up by the Universities of Cambridge and Oxford in collaboration with NHSBT (Moore et al., 2014).

INTERVAL was a large, parallel group, pragmatic, and individually-randomised trial conducted at 25 static donor centres of NHSBT over a 2-year period. It was the first randomised trial to evaluate the efficiency and safety of varying the frequency of whole blood donation. 22466 male donors were randomly assigned (1:1:1) to the 12-week (standard) vs. 10-week vs. 8-week inter-donation intervals (the maximum number of donations possible over 2 years was 8, 10, and 12,

respectively), and 22797 female donors were randomly assigned (1:1:1) to the 16-week (standard) vs. 14-week vs. 12-week intervals (the maximum number of donations possible over 2 years was 6, 7, and 8, respectively). The primary outcome was the number of blood donations over 2 years, with the standard practice being to donate 1 unit of blood per session (full donation unit=470 ml). Secondary outcomes such as donors' quality of life, donation-related symptoms, and number of low Hb deferrals were also recorded. Full details of the trial were provided in Moore et al. (2014). Primary analysis of the INTERVAL data concluded that over the 2-year trial period, reducing the inter-donation intervals led to a substantial increase in the amount of blood collected without any detectable effects on donors' overall quality of life, physical activity, or cognitive function. However, increased donation frequency resulted in a greater number of deferrals due to low Hb, lower average Hb and ferritin concentrations, and more self-reported symptoms (Di Angelantonio et al., 2017).

INTERVAL trial participants were well-characterised at baseline (for example, detailed information was available on donors' demographics, blood donation history, full blood count analysis, iron-related biomarkers, and quality of life), providing us an opportunity to harness insights from the INTERVAL study and further explore personalised donation strategies whereby the amount of blood collected is maximised while controlling for the number of deferrals for low Hb (Moore et al., 2016). We expect differential relationships between the amount of blood collected and the number of deferrals for low Hb by donors' individual characteristics. It might be the case that some donors should give blood less frequently to maintain adequate iron store levels, whereas other donors are able to give blood more frequently, depending on a number of factors such as age, body mass index, and donation history. For example, young, female donors with low body mass index may be less able to donate more frequently without a rise in the number of deferrals for low Hb, whereas donors with a long, successful donation history may be able to give blood more frequently with no/minimal increase in the number of deferrals for low Hb (Moore et al., 2014).

In this thesis, we integrate multi-marker data from the INTERVAL trial and develop statistical models which can be used to predict the optimal personalised inter-donation interval that tailors to the individual characteristics of each donor. We also leverage the INTERVAL data to investigate the stratification of donors according to their donation capacity. We hope that the work presented in this thesis will have implications for the future practice of blood donation.

As a remark, throughout this thesis, when we specifically point to the application of methods to the INTERVAL trial in the blood donation context, we refer to subjects/individuals as "donors" and the randomised group/intervention as "inter-donation interval". On the other hand, when we discuss a method in a more generic setup, we switch the use of terms and refer to subjects/individuals as "patients" and the randomised group/intervention as "treatment".

1.4 Tailoring in multi-arm trial settings

The INTERVAL study is one example of multi-arm trials. In clinical contexts other than blood donation, multi-arm trials are also very popular, primarily because they investigate multiple treatments simultaneously and have the potential to deliver results as rapidly as possible across a range of therapeutic options according to the same evaluation criteria (European Medicines Agency, 2020). A review by Baron et al. (2013) found that 17.6% of 1690 randomised trials published in 2009 were multi-arm trials. Recently, a lot of efforts have been made to combat the coronavirus disease (COVID-19) pandemic, including the active investigation of which repurposed medicinal products (i.e. drugs or drug combinations already licensed and used for a different indication, for example, remdesivir, lopinavir/ritonavir, and chloroquine) among a range of potentially useful ones might be effective and safe to treat patients with manifestations of COVID-19. Many trials launched for this purpose, such as the "Solidarity" trial (World Health Organization, 2020) and the "RECOVERY" trial (ISRCTN Registry, 2020; Nuffield Department of Population Health, 2020), are multi-arm and possibly "adaptive" (Stallard et al., 2020). In such trials, multiple potential therapeutic options for COVID-19 can be tested simultaneously and more patients have the chance to receive a potentially active treatment than would be the case if there are many separate two-arm trials each incorporating a control (placebo) arm (Tansey et al., 2020). There is growing evidence that COVID-19 is a heterogeneous disease (Mathew et al., 2020; Parisi and Leosco, 2020; Patel et al., 2020; Shrestha et al., 2020), and thus it is highly likely that different COVID-19 patients may benefit from different treatments and some treatments may be effective only for a particular type of patients (e.g. defined based on disease severity or immunotypes). Therefore, the future investigation of patient stratification and personalised drug therapies based on the data collected from the aforementioned multi-arm COVID-19 treatment trials can be particularly

helpful in guiding the clinical practice and ensuring more successful outcomes (Patel et al., 2020; Shrestha et al., 2020).

Despite the popularity of multi-arm trials in medical research and an increasing interest in finding evidence for personalised interventions and subgroup selections from trial data, most existing statistical methods for personalised medicine were developed for trials with a binary treatment and cannot be extended to incorporate multiple treatment comparisons in multi-arm trials straightforwardly. The thesis aims to address this methodological challenge from both perspectives on personalised medicine (discussed in Section 1.2). We consider methods that can handle more than two treatment options and are directly applicable to multi-arm trials, both at the individual level (for ITR estimation) and at the subgroup level (for patient stratification). We hope that the methods proposed/discussed in this thesis for tailoring in multi-arm trials can not only facilitate the analysis of the INTERVAL trial data in the blood donation context but also help researchers better analyse the data from other multi-arm trials in a wider range of medical applications.

1.5 Thesis structure

In Chapter 2, we first summarise some methodological options to estimate the optimal ITR in multi-arm trials. Then we compare their empirical performance in a variety of simulation settings and present a case study using the data from male donors in the INTERVAL trial.

The three inter-donation interval options in the INTERVAL trial exhibit a natural ordering, and applying the ITR estimation methods developed for nominal treatments (i.e. methods that ignore the ordinality) to the INTERVAL data may result in suboptimal decisions. To this end, in Chapter 3, we propose a sequential re-estimation (SR) learning method that effectively incorporates information on treatment orderings to identify the optimal ITR in the ordinal-arm setting. We also propose variable selection methods under the SR learning framework. The utility of SR learning in guiding personalised treatment selections among ordinal arms is demonstrated via simulation studies and an analysis of the data from a "much-in-demand but vulnerable" donor population (young female donors with O negative blood type) in the INTERVAL trial.

In Chapter 4, we shift our focus from the first to the second perspective on personalised medicine and introduce a new method for stratifying patients with

heterogeneous baseline characteristics and treatment response profiles into more homogeneous subgroups based on the data from multi-arm trials by leveraging Bayesian nonparametric clustering techniques. The performance of the proposed patient stratification method is evaluated via simulations and our analysis of the INTERVAL data (the same donor population as that investigated in Chapter 3) using the proposed method reveals clinically meaningful separation of blood donors.

Finally, in Chapter 5, the work presented in this thesis and our main contributions are summarised. We also outline ideas for future research.

Chapter 2

Learning optimal individualised decision rules from a multi-arm trial: a comparison of methods and an application to tailoring inter-donation intervals among blood donors in the UK

2.1 Introduction

As has been discussed in Chapter 1, one important aspect of personalised medicine is to provide each patient with the right treatment in order to improve patient outcomes. This can be formalised as the estimation of the optimal individualised treatment rule (ITR) by mapping patient information onto the set of treatment options (Murphy, 2003). There has been a considerable amount of literature focusing on this matter (Qian and Murphy, 2011; Zhang et al., 2012a,b; Zhao et al., 2012). We refer readers to Lipkovich et al. (2017) and Kosorok and Laber (2019) for a comprehensive review of the methods.

In the INTERVAL trial described in Section 1.3, there are three randomised groups for each gender, and in general, clinical trials with multiple treatment arms are very common in practice (Section 1.4). However, most existing statistical approaches to estimating the optimal ITR can only handle binary treatments, and directly aggregating multiple pairwise comparison results based on methods

such as one-versus-rest (OvR) or one-versus-one (OvO) can be problematic: there are situations where the two-way preferences may be non-transitive and thus a consensus on the final decision of the original multi-category comparison problem cannot be reached. In addition, OvR and OvO may result in suboptimal decisions (Allwein et al., 2001; Lee et al., 2004).

In this chapter, we describe several methodological options that can be used to identify the optimal ITR in clinical trials with more than two treatment arms and are computationally feasible for large-scale trials, including l_1-penalised least squares (Qian and Murphy, 2011), adaptive contrast weighted learning (Tao and Wang, 2017), direct learning (Qi and Liu, 2018), and a Bayesian approach that is based on Bayesian additive regression trees (Logan et al., 2019). Our motivating example, the INTERVAL trial (Di Angelantonio et al., 2017; Moore et al., 2014) introduced in Section 1.3, had more than 20000 observations on each gender, and we are in particular interested in the performance of these methods on large datasets similar in size to the INTERVAL trial. However, to our knowledge, there is a lack of studies evaluating their performance in datasets with sample sizes larger than 2000. To fill this gap, we compare the aforementioned multi-arm ITR estimation methods in large-scale trials via simulation studies. We then conduct a case study, applying these methods to the data from the INTERVAL trial to estimate the optimal inter-donation interval among three possible options (12-week, 10-week, and 8-week) for male donors. We estimate the proportion of blood donors allocated to each inter-donation interval and quantify the gain (or loss) in outcomes when assigning donors according to the optimal ITRs inferred using these methods under three different targets: (i) maximising the total units of whole blood collected by the blood service, (ii) minimising the risk of deferrals for low haemoglobin (Hb), and (iii) maximising a utility score that seeks to balance the amount of blood collected against the number of deferrals for low Hb. We note from the INTERVAL trial that only considering maximising the blood collection (target (i)) or minimising deferrals for low Hb (target (ii)) results in donor assignment strategies that are not particularly surprising and quite "polar" in nature (Di Angelantonio et al., 2017). The purpose of investigating these two targets by applying personalised medicine-based approaches is to examine in extreme cases where the non-personalised strategy should lead to nearly optimal outcomes, whether ITRs estimated using personalised medicine-based methods are "almost non-personalised" and assign almost all donors to the marginally best inter-donation interval option, or instead "falsely" discover a fair amount of heterogeneity in the optimal inter-donation

interval across different donors (Watson and Holmes, 2020). To our knowledge, this aspect has been rarely explored in the area of personalised medicine as most researchers begin by assuming that different patients would benefit from different treatments. For male blood donors in the INTERVAL trial, results suggest that different statistical methods to estimate the optimal ITR lead to fairly similar recommendations on the optimal donation strategies: optimal ITRs estimated under the first two targets are very close to "one-size-fits-all" strategies, while optimal ITRs estimated under the third target suggest some heterogeneity in different blood donors' optimal inter-donation intervals. In addition, the optimal ITR is highly dependent on how the utility score is constructed, i.e. the trade-off between the blood collection and the deferrals for low Hb.

The remainder of this chapter is organised as follows. In Section 2.2, we discuss the statistical framework and methods for estimating the optimal ITR when the treatment is binary. In Section 2.3, we review a selection of methods for estimating the optimal ITR when more than two treatment options are available. In Section 2.4, we conduct simulation studies to evaluate the performance of the multi-arm ITR estimation methods discussed in Section 2.3 under different scenarios. Section 2.5 presents results for an application of these methods to the data from male donors in the INTERVAL trial. We conclude this chapter with a discussion in Section 2.6.

2.2 Preliminaries

2.2.1 Notations and statistical frameworks

We consider a clinical trial where n subjects are sampled from a population of interest. Let Y be the outcome of interest, $A \in \mathcal{A} = \{1, \ldots, K\}$ denote the treatment assignment, and $\mathbf{X} \in \mathcal{X}$ be a p-dimensional covariate (feature) vector. Without loss of generality, we assume a larger value of Y is preferred. We observe the triplet $(Y_i, \mathbf{X}_i, A_i)$, for $i = 1, \ldots, n$, which are independent and identically distributed across i. The individualised treatment rule (ITR), denoted by $\mathcal{D}$, is a map from the space of feature variables, $\mathcal{X}$, to the domain of treatment assignments, $\mathcal{A}$. The notion of potential outcomes (counterfactuals) is defined as a subject's outcome had he/she followed a regime (can be different from the observed one). Under the counterfactual framework (Imbens and Rubin, 2015), we let $Y^*(a)$ denote the potential outcome had a subject received treatment a.

To connect the counterfactual outcome with the observed data, we introduce the following assumptions:

(A.1) consistency:
the observed outcome is the same as the counterfactual outcome under the treatment a subject actually receives, i.e. $Y = \sum_{a=1}^{K} Y^*(a)I(A = a)$, where $I(\bullet)$ is the indicator function.

(A.2) no unmeasured confounders (NUC):
the treatment assignment A is independent of the potential outcomes conditional on $\mathbf{X}$, i.e. $\{Y^*(1), \ldots, Y^*(K)\} \perp\!\!\!\perp A|\mathbf{X}$.

(A.3) positivity:
with probability 1, $P(A = a|\mathbf{X})$ is bounded strictly away from 0, i.e. $P(A = a|\mathbf{X} = \mathbf{x}) > 0$ for all $a \in \mathcal{A}$ and $\mathbf{x} \in \mathcal{X}$.

Assumption (A.1) is fundamental under the potential outcome framework (Zhang et al., 2012b). Assumption (A.2) is always satisfied in clinical trials (for the intention-to-treat analysis), but typically unverifiable in observational studies (Zhao et al., 2019a). Assumption (A.3) may be violated if the study design excludes certain subjects from receiving a particular treatment. For instance, in the INTERVAL trial, female donors can never be randomised to 8-week or 10-week inter-donation intervals while male donors cannot be assigned to 14-week or 16-week inter-donation intervals. However, this assumption holds when the analysis is stratified by gender.

Under assumptions (A.1)-(A.3), we are able to make inference on $E\{Y^*(a)\}$ ($E(\bullet)$ denotes the expectation) based on the observed data. By assumptions (A.1) and (A.2), we have

$$\begin{aligned} E\{Y^*(a)|\,\mathbf{X}\} &\overset{(A.2)}{=} E\{Y^*(a)|\,\mathbf{X}, A = a\} \\ &\overset{(A.1)}{=} E(Y|\,\mathbf{X}, A = a). \end{aligned} \tag{2.1}$$

Assumption (A.3) ensures the identifiability of $E(Y|\,\mathbf{X}, A = a)$. In the setting with K treatment arms, the counterfactual outcome under regime $\mathcal{D}$ is given by

$$Y^*(\mathcal{D}(\mathbf{X})) = \sum_{a=1}^{K} Y^*(a)I\{\mathcal{D}(\mathbf{X}) = a\}. \tag{2.2}$$

The value function associated with $\mathcal{D}$ (Qian and Murphy, 2011), denoted by $V(\mathcal{D})$, is defined as the expected value of the outcome Y had the regime $\mathcal{D}$ been applied to the given population, i.e. $V(\mathcal{D}) := E\{Y^*(\mathcal{D}(\mathbf{X}))\}$. It follows from (2.1) and (2.2) that

$$\begin{aligned} E\{Y^*(\mathcal{D}(\mathbf{X}))|\,\mathbf{X}\} &\overset{(2.2)}{=} \sum_{a=1}^{K} E\{Y^*(a)|\,\mathbf{X}\} I\{\mathcal{D}(\mathbf{X}) = a\} \\ &\overset{(2.1)}{=} \sum_{a=1}^{K} E(Y|\,\mathbf{X}, A = a) I\{\mathcal{D}(\mathbf{X}) = a\}. \end{aligned} \tag{2.3}$$

Applying the law of iterated expectations, we have

$$\begin{aligned} E\{Y^*(\mathcal{D}(\mathbf{X}))\} &= E\Big[E\{Y^*(\mathcal{D}(\mathbf{X}))|\,\mathbf{X}\}\Big] \\ &= E\Big[\sum_{a=1}^{K} E(Y|\,\mathbf{X}, A = a) I\{\mathcal{D}(\mathbf{X}) = a\}\Big]. \end{aligned} \tag{2.4}$$

Our goal is to find the optimal ITR, $\mathcal{D}^*$, that maximises the value function under $\mathcal{D}$, i.e.

$$\begin{aligned} \mathcal{D}^* &= \arg\max_{\mathcal{D}} V(\mathcal{D}) \\ &= \arg\max_{\mathcal{D}} E\Big[\sum_{a=1}^{K} E(Y|\,\mathbf{X}, A = a) I\{\mathcal{D}(\mathbf{X}) = a\}\Big]. \end{aligned} \tag{2.5}$$

2.2.2 Estimation of the optimal ITR when $K = 2$

Many statistical methods have been proposed to estimate the optimal ITR in the case with two treatments ($K = 2$), including indirect and direct methods.

Indirect methods rely on modelling conditional mean outcomes (e.g. Q-learning) or modelling treatment contrasts (e.g. A-learning). The optimal ITR is then derived (indirectly) by inverting the posited model. In the Q-learning paradigm, a model including treatment-covariate interactions is postulated for $E(Y|\mathbf{X}, A = a)$. It can be shown from (2.5) that $\mathcal{D}^*(\mathbf{x}) = \arg\max_a E(Y|\mathbf{X} = \mathbf{x}, A = a)$, and thus the decision follows naturally by maximising the outcome from the fitted model, i.e. $\widehat{\mathcal{D}^*}(\mathbf{x}) = \arg\max_a \widehat{E}(Y|\mathbf{X} = \mathbf{x}, A = a)$ (Watkins and Dayan, 1992). Alternatively, A-learning methods posit a model on the contrast function $C(\mathbf{x})$ between two treatment options, and the final decision is based on the sign of the estimated contrast $\widehat{C}(\mathbf{x})$ (Murphy, 2003). Compared to Q-learning, A-learning bypasses the need for modelling marginal covariate

effects. Those indirect approaches focus on building a prediction model for either the conditional mean outcome or the treatment contrast. Therefore, their performance is highly dependent on the correct specification of the posited model and model misspecification may lead to suboptimal ITRs. In addition, indirect methods target prediction accuracy instead of the direct maximisation of the value function.

To circumvent these limitations, Zhao et al. (2012) proposed a direct approach, the outcome weighted learning (OWL), to estimate the optimal ITR (directly) by maximising the value function $V(\mathcal{D})$ rather than (indirectly) by inverting estimates from prediction models. We note that $V(\mathcal{D}) = E[Y|A = \mathcal{D}(\mathbf{X})] = E\Big[I\{A = \mathcal{D}(\mathbf{X})\}Y/P(A|\mathbf{X})\Big]$, and thus

$$\begin{aligned} \mathcal{D}^* &= \arg\max_{\mathcal{D}} E\left[\frac{I\{A = \mathcal{D}(\mathbf{X})\}}{P(A|\mathbf{X})}Y\right] \\ &= \arg\min_{\mathcal{D}} E\left[\frac{I\{A \neq \mathcal{D}(\mathbf{X})\}}{P(A|\mathbf{X})}Y\right]. \end{aligned} \tag{2.6}$$

Based on (2.6), Zhao et al. (2012) reformulated the problem of maximising the value function into a weighted classification problem (with the aim of minimising the weighted misclassification error), where they classified A based on $\mathbf{X}$ and weighted the misclassification by $Y/P(A|\mathbf{X})$. This could then be solved using support vector machines (SVM). To further improve the performance of OWL, Zhou et al. (2017) and Liu et al. (2018) introduced residual weighted learning (RWL) and augmented outcome weighted learning (AOL) respectively by using different weights in the objective function. Those direct methods target the optimal decision rules directly and avoid the need for modelling the outcome or contrast. Therefore, they offer protection against model misspecification (outcome model or contrast model).

Most of the aforementioned methods, except for Q-learning (Q-learning only specifies an outcome model, hence it can handle any treatment type), were developed for the case of two treatments, and the extension of these methods to the multi-arm setting is not straightforward.

2.3 Review of a selection of methods for $K > 2$

Recently, some new methods have been proposed to deal with the case of more than two treatments. In this section, we review (in publication order) a selection of approaches that can be used to estimate the optimal ITR in multi-arm trials. In particular, we only consider methods that scale well and do not impose high computational costs for large datasets similar in size to the INTERVAL trial, including l_1-penalised least squares (Qian and Murphy, 2011), adaptive contrast weighted learning (Tao and Wang, 2017), direct learning (Qi and Liu, 2018), and a Bayesian approach that is based on Bayesian additive regression trees (Logan et al., 2019). Although some details on each approach are given, we refer readers to the original publications for more technical discussions.

2.3.1 l_1-penalised least squares (indirect method)

Qian and Murphy (2011) proposed the two-stage model-based l_1-penalised least squares (l_1-PLS) method. The conditional mean response was first estimated using l_1-PLS and then the estimated means under different treatments were used to derive the optimal ITR. Specifically, we fit the model for $E(Y|\mathbf{X}, A)$ using the basis function $(1, \mathbf{X}, A, \mathbf{X}A)$, in which we use $K - 1$ dummy variables to replace A. Following (2.5), the decision can be derived as $\widehat{\mathcal{D}^*}(\mathbf{x}) = \arg\max_{a\in\{1,\ldots,K\}} \widehat{E}(Y|\mathbf{X} = \mathbf{x}, A = a)$.

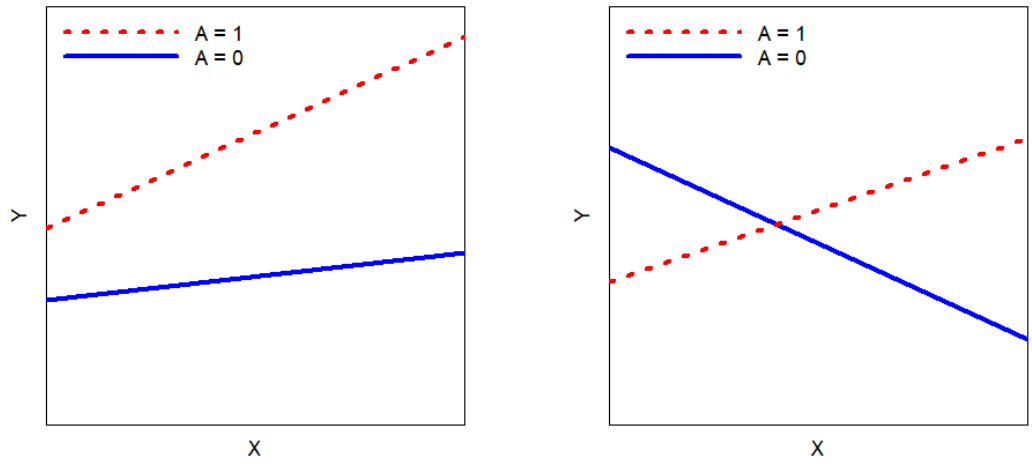
$\mathbf{X}$ in the INTERVAL trial dataset contains several nominal categorical covariates. In order to avoid the over-selection of variables with many categories and ensure that all dummy variables encoding the same categorical covariate are either included or excluded from the model simultaneously, we use group LASSO (GL) which encourages sparsity at the factor level for variable selection (Gunter et al., 2011; Yuan and Lin, 2006).

Another variable selection approach that is of interest in this context is hierarchical group LASSO (HGL). HGL was designed for learning pairwise interactions in regression models that satisfy strong hierarchy, i.e. if the coefficient associated with the interaction term is estimated to be non-zero, then its two associated main effects also have non-zero estimated coefficients. The assumption underlying HGL is that covariates that predict treatment effect heterogeneity (included as interactions) are also prognostic (included in main effects).

As has been pointed out by Gunter et al. (2011), maintaining hierarchy in variable selection avoids finding spurious interactions that may appear due to the

exclusion of important main effect terms. The HGL method proposed by Lim and Hastie (2015) picks out only main effects if the truth has no interactions, but when interactions truly exist, it allows the discovery of important interaction terms despite their weak effects by regularising coefficients of main effects using the "glinternet" penalty. We can embed HGL in l_1-PLS to identify important treatment-covariate interactions. For ease of interpretation, we restrict the search space to be all possible two-way interactions between treatment A and covariates $\mathbf{X}$.

As an aside, there are two types of interactions: quantitative interactions and qualitative interactions (Lipkovich et al., 2017). A quantitative interaction between A and $\mathbf{X}$ refers to the situation where the magnitude of the effect of A on the outcome Y depends on $\mathbf{X}$ but the direction of this effect is consistent for all possible values of $\mathbf{X}$, whereas a qualitative interaction between A and $\mathbf{X}$ indicates that both the direction and the magnitude of A's effect on the outcome Y can depend on $\mathbf{X}$ (Figure 2.1).



(a) *Quantitative interaction* **(b)** *Qualitative interaction*

Fig. 2.1: *Illustration of two types of interactions in the setting with a binary treatment.*

GL and HGL interaction selection procedures are not able to automatically distinguish between qualitative and quantitative interactions. Therefore, further assessment of interactions (e.g. using the Gail-Simon test proposed by Gail and Simon (1985)) is required to determine whether the selected interactions are useful for achieving specific aims (e.g. the aim can either be to detect a qualitative difference in treatment effects, i.e. crossover treatment effect heterogeneity, or to detect a quantitative difference in treatment effects that informs the identification

of subgroups with elevated response to a given treatment, i.e. non-crossover treatment effect heterogeneity) in the context of personalised medicine.

One limitation of l_1-PLS is that the correct inference on the optimal ITR depends on a correctly specified outcome model, while the postulated outcome model is prone to misspecification, especially when $\mathbf{X}$ is high-dimensional. Unlike tree-based methods, which can capture more general forms of interactions, l_1-PLS is restricted to searching for additive-type interactions. However, the true relationship between covariates, treatment assignment, and the outcome is usually complicated in practice, and the underlying interaction structures can be much more flexible than additive. Another issue with this approach is the interference between main effects and treatment-covariate interactions, which may impair the search for the optimal ITR. In most cases, the variability in the outcome is predominantly explained by main effects rather than interactions. Therefore, interaction effects may be overlooked due to their low predictive ability in models that focus primarily on prediction (Tian et al., 2014).

2.3.2 Adaptive contrast weighted learning (direct method)

Adaptive contrast weighted learning (ACWL) was proposed by Tao and Wang (2017) to estimate the optimal ITR in the multi-arm setting. They constructed doubly-robust semi-parametric regression-based contrasts with the adaptation of treatment effect orderings, and the adaptive contrasts simplified the optimisation problem with multiple treatment comparisons into a weighted classification problem.

Denote the conditional mean $E(Y|\mathbf{X}, A = a)$ as $\mu_a(\mathbf{X})$. Let $\mu_{(1)}(\mathbf{X}) \leq \ldots \leq \mu_{(K)}(\mathbf{X})$ be the order statistics of $\mu_1(\mathbf{X}), \ldots, \mu_K(\mathbf{X})$, and l_a be the treatment effect order with $\mu_{(a)}(\mathbf{X}) = \mu_{l_a}(\mathbf{X})$. It follows that

$$\begin{aligned} \mathcal{D}^*(\mathbf{X}) &= \arg\max_{\mathcal{D}} \sum_{a=1}^{K} E\Big[\mu_{(a)}(\mathbf{X}) I\{\mathcal{D}(\mathbf{X}) = l_a(\mathbf{X})\}\Big] \\ &= \arg\min_{\mathcal{D}} \sum_{a=1}^{K-1} E\Big[\{\mu_{(K)}(\mathbf{X}) - \mu_{(a)}(\mathbf{X})\} I\{\mathcal{D}(\mathbf{X}) = l_a(\mathbf{X})\}\Big]. \end{aligned} \tag{2.7}$$

The optimal ITR derived from (2.7) can be interpreted as the rule that minimises the expected loss in the outcome due to suboptimal treatments in the entire population of interest, i.e. the ITR classifies as many subjects as possible to their corresponding optimal treatment l_K and puts more "penalties" on subjects with larger contrasts. In practice, it can be challenging to utilise all $K - 1$ contrasts

as weights to classify treatment. Tao and Wang (2017) addressed this problem by constructing the lower and upper bounds of the expected loss in the outcome due to suboptimal treatments as follows:

$$\begin{aligned}\sum_{a=1}^{K-1} E\Big[\{\mu_{(K)}(\mathbf{X}) - \mu_{(a)}(\mathbf{X})\} I\{\mathcal{D}(\mathbf{X}) = l_a(\mathbf{X})\}\Big] &\geq \sum_{a=1}^{K-1} E\Big[\{\mu_{(K)}(\mathbf{X}) - \mu_{(K-1)}(\mathbf{X})\} I\{\mathcal{D}(\mathbf{X}) = l_a(\mathbf{X})\}\Big] \\ &= E\Big[\{\mu_{(K)}(\mathbf{X}) - \mu_{(K-1)}(\mathbf{X})\} I\{\mathcal{D}(\mathbf{X}) \neq l_K(\mathbf{X})\}\Big] \\ &= E\Big[C_1(\mathbf{X}) I\{\mathcal{D}(\mathbf{X}) \neq \underbrace{l_K(\mathbf{X})}_{\text{optimal}}\}\Big],\end{aligned} \tag{2.8}$$

and

$$\begin{aligned}\sum_{a=1}^{K-1} E\Big[\{\mu_{(K)}(\mathbf{X}) - \mu_{(a)}(\mathbf{X})\} I\{\mathcal{D}(\mathbf{X}) = l_a(\mathbf{X})\}\Big] &\leq \sum_{a=1}^{K-1} E\Big[\{\mu_{(K)}(\mathbf{X}) - \mu_{(1)}(\mathbf{X})\} I\{\mathcal{D}(\mathbf{X}) = l_a(\mathbf{X})\}\Big] \\ &= E\Big[\{\mu_{(K)}(\mathbf{X}) - \mu_{(1)}(\mathbf{X})\} I\{\mathcal{D}(\mathbf{X}) \neq l_K(\mathbf{X})\}\Big] \\ &= E\Big[C_2(\mathbf{X}) I\{\mathcal{D}(\mathbf{X}) \neq \underbrace{l_K(\mathbf{X})}_{\text{optimal}}\}\Big],\end{aligned} \tag{2.9}$$

where $C_1(\mathbf{X}) = \mu_{(K)}(\mathbf{X}) - \mu_{(K-1)}(\mathbf{X})$ and $C_2(\mathbf{X}) = \mu_{(K)}(\mathbf{X}) - \mu_{(1)}(\mathbf{X})$.
In the least conservative case where suboptimal treatments only lead to the minimal expected loss in the outcome,

$$\mathcal{D}^*(\mathbf{X}) = \arg\min_{\mathcal{D}} E\Big[C_1(\mathbf{X}) I\{\mathcal{D}(\mathbf{X}) \neq l_K(\mathbf{X})\}\Big], \tag{2.10}$$

while in the most conservative case where suboptimal treatments lead to the maximal expected loss in the outcome,

$$\mathcal{D}^*(\mathbf{X}) = \arg\min_{\mathcal{D}} E\Big[C_2(\mathbf{X}) I\{\mathcal{D}(\mathbf{X}) \neq l_K(\mathbf{X})\}\Big]. \tag{2.11}$$

The optimal ITR estimated using (2.10) or (2.11) minimises the lower or upper bounds of the expected loss in the outcome due to suboptimal treatments over the entire population, respectively. By optimising these bounds, the optimisation problem with multiple treatment comparisons is simplified to a weighted classification problem, which can be solved by classification techniques, such as classification and regression trees (CART).

In order to get robust estimators for the conditional mean $\mu_a(\mathbf{X})$ and the treatment effect order $l_a(\mathbf{X})$, Tao and Wang (2017), following Zhang et al. (2012b), employed the doubly-robust augmented inverse probability weighted estimator (AIPWE):

$$\widehat{\mu}_a^{\text{AIPWE}}(\mathbf{X}) = \frac{I(A=a)}{\widehat{P}(A=a|\mathbf{X})}Y + \left\{1 - \frac{I(A=a)}{\widehat{P}(A=a|\mathbf{X})}\right\}\widehat{\mu}_a(\mathbf{X}), \tag{2.12}$$

where $\widehat{P}(A=a|\mathbf{X})$ is the estimated propensity score (in randomised trials, the true propensity score is known from the randomisation scheme, while in observational studies, propensity score can be estimated via multinomial logistic regression), and $\widehat{\mu}_a(\mathbf{X})$ can be obtained as the regression-based conditional mean estimator from a parametric model including treatment-covariate interactions. AIPWE is doubly-robust since $\widehat{\mu}_a^{\text{AIPWE}}$ is a consistent estimator for μ_a if either the model for $P(A=a|\mathbf{X})$ (i.e. the propensity score model) or that for $\mu_a(\mathbf{X})$ (i.e. the outcome model) is correctly specified (for randomised trials, the propensity score model is always correctly specified). Working orders $\widehat{l}_a^{\text{AIPWE}}(\mathbf{X})$ can then be obtained by sorting $\widehat{\mu}_1^{\text{AIPWE}}(\mathbf{X}), \ldots, \widehat{\mu}_K^{\text{AIPWE}}(\mathbf{X})$, and adaptive contrasts can be estimated as $\widehat{C}_1^{\text{AIPWE}}(\mathbf{X}) = \widehat{\mu}_{(K)}^{\text{AIPWE}}(\mathbf{X}) - \widehat{\mu}_{(K-1)}^{\text{AIPWE}}(\mathbf{X})$ and $\widehat{C}_2^{\text{AIPWE}}(\mathbf{X}) = \widehat{\mu}_{(K)}^{\text{AIPWE}}(\mathbf{X}) - \widehat{\mu}_{(1)}^{\text{AIPWE}}(\mathbf{X})$.

There are several advantages of ACWL. Firstly, the optimal ITR based on ACWL is efficient and robust against model misspecification (due to nonparametric classification and the doubly-robust property of AIPWE). Secondly, this method is highly flexible: it can be applied to both clinical trials and observational studies; many machine learning classification algorithms can be applied to solve the weighted classification problem; and the method can be applied to data with different outcome types as long as the relationship between the outcome, treatment, and covariates can be properly modelled. However, additional uncertainties may be induced through the introduction of the "ordering labels". Therefore, using adaptive contrasts may not be the most efficient way to get around the multiple treatment comparison problem.

2.3.3 Direct learning (direct method)

Qi and Liu (2018) developed the direct learning (D-learning) approach that uses regression methods to directly estimate the optimal ITR. The advantage of D-learning over Q-learning is that D-learning directly targets the maximisation of the value function, thus avoiding the need to model main effects. As shown by

Qi and Liu (2018), in the binary arm case (treatment A is encoded as -1 or 1),

$$\begin{aligned}\mathcal{D}^*(\mathbf{X}) &= \arg\max_{\mathcal{D}} V(\mathcal{D}) \\ &= \text{sign}\{E(Y|\mathbf{X}, A=1) - E(Y|\mathbf{X}, A=-1)\} \\ &= \text{sign}\left[E\left\{\frac{YA}{P(A|\mathbf{X})}\middle|\mathbf{X}\right\}\right] \\ &= \text{sign}\{f^*(\mathbf{X})\},\end{aligned} \tag{2.13}$$

where $f^*(\mathbf{X})$ is the optimal decision function. According to (2.13), $f^*(\mathbf{X})$ can be estimated using the fact that $f^*(\mathbf{X}) = E\left\{\frac{YA}{P(A|\mathbf{X})}\middle|\mathbf{X}\right\}$. Qi and Liu (2018) showed that under the assumption of exchangeability between differentiation and expectation,

$$\begin{aligned}\frac{\partial E\left[\frac{1}{P(A|\mathbf{X})}\{2YA - f(\mathbf{X})\}^2\right]}{\partial f} &= 4E\left[f(\mathbf{X}) - E\left\{\frac{YA}{P(A|\mathbf{X})}\middle|\mathbf{X}\right\}\right] \\ &= 4E\Big[f(\mathbf{X}) - f^*(\mathbf{X})\Big].\end{aligned} \tag{2.14}$$

Therefore, we have

$$f^*(\mathbf{X}) \in \arg\min_f E\left[\frac{1}{P(A|\mathbf{X})}\{2YA - f(\mathbf{X})\}^2\right], \tag{2.15}$$

based on which we can estimate $f^*(\mathbf{X})$ using regression methods for either linear or nonlinear decision rules. For example, suppose the decision function is linear and $f(\mathbf{X}) = \mathbf{X}^\mathsf{T}\boldsymbol{\beta}$. Then $\boldsymbol{\beta}$ can be estimated using the weighted least squares (the weight is $1/P(A|\,\mathbf{X})$ and the response variable is $2YA$). The estimated optimal linear decision function is $\widehat{f^*}(\mathbf{X}) = \mathbf{X}^\mathsf{T}\widehat{\boldsymbol{\beta}}$, and the estimated optimal ITR $\widehat{\mathcal{D}^*}(\mathbf{X})$ is the sign of $\widehat{f^*}(\mathbf{X})$. In the high-dimensional case, LASSO can be used.

D-learning was extended to the case with more than two treatments (multi-category D-learning). The optimal ITR in the multi-arm setting can be written as:

$$\begin{aligned}\mathcal{D}^*(\mathbf{X}) &= \arg\max_{a\in\{1,\ldots,K\}} \sum_{i\neq a}^{K} E\left\{\frac{YA_{ai}}{P(A_{ai}|\mathbf{X})}\middle|\mathbf{X}, A = a \text{ or } i\right\} \\ &= \arg\max_{a\in\{1,\ldots,K\}} \sum_{i\neq a}^{K} f^*_{ai}(\mathbf{X}),\end{aligned} \tag{2.16}$$

where $A_{ai} \in \{-1, 1\}$ denotes a binary treatment indicator for whether a patient is on treatment i ($A_{ai} = -1$) or on treatment a ($A_{ai} = 1$) for $i \neq a$, and

$f^*_{ai}(\mathbf{X})$ denotes the optimal decision function between treatment a and i. Each pairwise decision function $f^*_{ai}(\mathbf{X})$, for $a, i = 1, \ldots, K, i \neq a$, can be estimated using regression methods as in the binary setting.

We note that even though multi-category D-learning builds on multiple pairwise comparisons, it is different from the one-versus-one (OvO) approach (Bishop, 2007) in that OvO aggregates multiple pairwise decisions based on the majority voting rule, whereas multi-category D-learning measures the effect of each treatment based on the sum of pairwise contrasts and then picks the treatment with the largest effect measure.

2.3.4 Bayesian additive regression trees (indirect method)

Bayesian additive regression trees (BART) is a flexible nonparametric prediction model that was first introduced by Chipman et al. (2010) and it has gained much popularity with numerous applications in recent years (Hill et al., 2020; Tan and Roy, 2019). The BART model is an additive ensemble of many simple regression trees with each tree explaining a small portion of the outcome. Suppose $Y = \mu(\mathbf{X}) + \epsilon$, where μ is unknown and $\epsilon \sim N(0, \sigma^2)$. We model and approximate $\mu(\mathbf{X}) = E(Y|\mathbf{X})$ by the sum of m regression trees:

$$\mu(\mathbf{X}) \approx h(\mathbf{X}) = \sum_{j=1}^{m} g_j(\mathbf{X}), \tag{2.17}$$

where each g_j represents a regression tree, $j = 1, \ldots, m$ (m denotes the number of trees, which is usually fixed at a large number, e.g. 200). Such a model can handle complex nonlinear terms and interaction effects without the need to specify their functional forms. Let T denote the tree structure, and $M = \{\tau_1, \tau_2, \ldots, \tau_b\}$ denote the set of values at each of the b terminal nodes of T. The sum-of-trees model can be expressed more explicitly as:

$$Y = \sum_{j=1}^{m} g(\mathbf{X}; T_j, M_j) + \epsilon. \tag{2.18}$$

For a fixed number of trees, the model is determined by $(T_1, M_1), \ldots, (T_m, M_m)$, and σ. When m is large, $\{g(\mathbf{X}; T_1, M_1), \ldots, g(\mathbf{X}; T_m, M_m)\}$ is highly redundant and forms an "over-complete" basis, since many different choices of $(T_1, M_1), \ldots, (T_m, M_m)$ can lead to the same $\sum_{j=1}^{m} g(\mathbf{X}; T_j, M_j)$. Therefore, strongly influential regularisation priors are imposed on all parameters of this model in favour of weak learners, i.e. trees that are small with leaf parameters that are close to zero.

In Chipman et al. (2010), the authors assumed that (i) $\{(T_1, M_1), \ldots, (T_m, M_m)\}$ and σ are independent, and (ii) $(T_1, M_1), \ldots, (T_m, M_m)$ are independent of each other. They suggested specifying the joint prior as:

$$\begin{aligned} p\{(T_1, M_1), \ldots, (T_m, M_m), \sigma\} &= p(\sigma) \times p\{(T_1, M_1), \ldots, (T_m, M_m)\} \\ &= p(\sigma) \times \prod_{j=1}^{m} p(M_j|T_j)p(T_j) \\ &= p(\sigma) \times \prod_{j=1}^{m} \prod_{l=1}^{b_j} p(\tau_{jl}|T_j)p(T_j), \end{aligned} \tag{2.19}$$

where b_j is the number of terminal nodes for the j^{th} tree (i.e. T_j), and τ_{jl} is the value at the l^{th} terminal node of T_j.

An inverse - χ^2 distribution is recommended for $p(\sigma)$ such that $\sigma^2 \sim \nu\lambda/\chi^2_\nu$, where λ (scale) and ν (degree of freedom) are hyperparameters. A data-driven approach is used for specifying λ and ν. The underlying idea is that if the true model for $E(Y|\,\mathbf{X})$ is nonlinear, a linear regression fit to the observed data is likely to overestimate the residual standard deviation. Specifically, $\widehat{\sigma}$ is first roughly guessed by the residual standard deviation from the least square regression. Then, hyperparameters are picked such that $P(\sigma < \widehat{\sigma}) = q$ (i.e. the q^{th} quantile of the prior on σ is located at $\widehat{\sigma}$), with q being a large number typically set at 0.75, 0.90, or 0.99. In practice, ν is fixed at 3, 5, or 10, and λ is chosen based on $P(\sigma < \widehat{\sigma}) = q$. The prior for $\tau_{jl}|T_j$ is specified as the conjugate normal distribution $N(\mu_\tau, \sigma^2_\tau)$, where μ_τ and σ^2_τ are set on the basis that $N(m\mu_\tau, m\sigma^2_\tau)$ assigns large probability to $(y_{\min}, y_{\max})$. For example, setting $m\mu_\tau - 2\sqrt{m}\sigma_\tau = y_{\min}$ and $m\mu_\tau + 2\sqrt{m}\sigma_\tau = y_{\max}$ indicates that there is a 95% prior probability that $E(Y|\mathbf{X})$ lies within the interval $(y_{\min}, y_{\max})$. In Chipman et al. (2010), the authors suggested rescaling Y into $(-0.5, 0.5)$ and then fixing μ_τ at 0 and σ_τ at $0.25/\sqrt{m}$. The purpose of setting μ_τ to zero is to keep the effect of each tree component on the overall fit small. Specification of $p(T_j)$ involves three components: the first one is the probability that a node at depth d is nonterminal (will further split), which is given by $\alpha/(1+d)^\beta$. Here, $\alpha \in (0, 1)$ controls how likely a terminal node will split (a larger α indicates a higher likelihood of a node being split), and $\beta \in [0, \infty)$ controls the number of terminal nodes (a larger β suggests a smaller number of terminal nodes). Chipman et al. (2010) recommended setting $(\alpha, \beta) = (0.95, 2)$, which enforces small trees. The second and the third components are the distributions of splitting covariates and

splitting cut-off points (conditional on the splitting covariates) at interior nodes, both of which are assumed to have uniform priors.

Logan et al. (2019) incorporated the idea of BART into the estimation of the optimal ITR by using BART to model the dependency structure of the response and covariates. In the first stage, a BART model for $E(Y|\mathbf{X}, A)$ was built using the treatment and all covariates as input variables. In the second stage, the optimal arm was chosen as the one that maximises $E(Y|\mathbf{X} = \mathbf{x}, A = a)$, which can be approximated by the average over posterior draws. Under this framework, the posterior distribution of the value function of an ITR can be obtained using posterior samples straightforwardly, and uncertainties about the value of an ITR can be captured by posterior samples from the prediction model directly. The R package `BART` can be used to fit the BART model and carry out posterior inference.

Thanks to the use of the BART prediction model in the first stage, this ITR estimation method performs well even when the underlying relationship between the outcome, treatment, and covariates is complicated (Logan et al., 2019). However, one potential problem of this method is that the prediction model constructed with BART is "black-box", which impairs the interpretability of the model and the decision rules. In the two-arm setting, Logan et al. (2019) proposed a way to facilitate the interpretation by fitting a single tree to the posterior mean treatment differences. However, the extension to the multi-arm case is not straightforward given that there are multiple pairwise contrasts when more than two treatment options are available. In addition, BART is more computationally demanding than the aforementioned non-Bayesian approaches due to its reliance on Markov chain Monte Carlo (MCMC).

2.4 Simulation studies

To our knowledge, there is a lack of studies comparing the performance of different multi-arm ITR estimation methods under scenarios where the training sample size is larger than 2000. It is reasonable to expect that the relative performance of different methods may vary with the training sample size, and thus findings in previous works may not apply to the INTERVAL trial, where the number of male/female participants is more than 20000. In addition, the BART ITR estimation method (Logan et al., 2019) discussed in Section 2.3.4 has not been compared with other methods in multi-arm trial settings before. In this section,

we conduct simulation studies to compare the performance of the following five methods in large samples:

(i) l_1-PLS-HGL: l_1-penalised least squares with hierarchical group LASSO variable selection using the basis function $(1, \mathbf{X}, A, \mathbf{X}A)$ (Lim and Hastie, 2015; Qian and Murphy, 2011);

(ii) l_1-PLS-GL: l_1-penalised least squares with group LASSO variable selection using the basis function $(1, \mathbf{X}, A, \mathbf{X}A)$ (Qian and Murphy, 2011; Yuan and Lin, 2006);

(iii) ACWL: adaptive contrast weighted learning using (2.11), where suboptimal decisions lead to maximal expected loss in the outcome (i.e. ACWL-C2) (Tao and Wang, 2017). We only present the results from ACWL-C2, since we observe that adaptive contrast weighted learning using (2.10) (i.e. ACWL-C1) gives very similar results in simulation studies, as has also been noted in the original publication (Tao and Wang, 2017);

(iv) D-learning: direct learning with linear decision functions (Qi and Liu, 2018);

(v) BART: Bayesian additive regression trees with default prior parameters as specified in the R package `BART` (Chipman et al., 2010; Logan et al., 2019).

R codes to implement these methods on a simulated example dataset are provided in Appendix A.1.

2.4.1 Base scenarios

We consider the training sample size being $n = 20000$ (similar in size to the number of male/female donors in the INTERVAL trial). We simulate data under 6 settings with different types of covariates and forms of treatment-covariate interactions. In each simulation setting, we generate 5 covariates $X_1, \ldots, X_5$ independently. Treatment A is sampled uniformly from $\{1, 2, 3\}$. We assume that the outcome Y is normally distributed with mean $m(\mathbf{X}) + \Delta(\mathbf{X}, A)$ and variance $\sigma^2 = 1$, where $m(\mathbf{X})$ is the main effect of covariates on the outcome and $\Delta(\mathbf{X}, A)$ denotes the treatment-covariate interaction effect. We consider $m(\mathbf{X}) = 1 + 0.5X_4 + 0.3X_5$ and $\Delta(\mathbf{X}, A) = 0.5\{I(A = 1)\Delta_1(\mathbf{X}) + I(A = 2)\Delta_2(\mathbf{X}) + I(A = 3)\Delta_3(\mathbf{X})\}$, with $\Delta_1(\mathbf{X})$, $\Delta_2(\mathbf{X})$, and $\Delta_3(\mathbf{X})$ taking on different functional forms. Details on distributions of $X_1, \ldots, X_5$ (we use $\mathcal{U}\{a, b\}$ to denote the continuous uniform distribution that takes values in the range $[a, b]$, and

Bern(p) to denote the Bernoulli distribution with the success probability being p) and expressions for $\Delta_1(\mathbf{X})$, $\Delta_2(\mathbf{X})$, and $\Delta_3(\mathbf{X})$ in settings 1-6 are provided in Table 2.1.

Settings 1 and 2 consider tree-type and linear interaction effects, respectively. True underlying decision boundaries are nonlinear in settings 3 and 4, with setting 3 including a between-covariate interaction. Setting 5 contains discrete covariates (one categorical covariate and one binary covariate). We examine the scenario where the true optimal treatment is the same for all individuals in setting 6. This scenario mimics the situation in the INTERVAL trial when we target two outcomes separately, i.e. objectives (i) and (ii) described in Section 2.1: in this case, we would expect the donor assignment strategy to be quite "polar" and the non-personalised rule to yield nearly optimal outcomes based on the primary analysis results presented in Di Angelantonio et al. (2017).

We evaluate the performance of different methods on a large independent testing dataset of size 10000 using two criteria: (i) the misclassification rate of the estimated optimal ITR compared to the true optimal ITR, and (ii) the value function under the estimated ITR. Smaller misclassification rates and larger value functions indicate better performance. Each simulation is repeated 100 times and all tuning parameters are selected via 5-fold cross-validation. We report both the mean and the standard deviation of misclassification rates and value functions across 100 replicates. We note that the BART ITR estimation method is different from the other four non-Bayesian methods by nature, and the uncertainty of the value function can be directly quantified under the Bayesian framework (based on posterior samples), as has been discussed in Section 2.3.4. However, to make different methods comparable, the standard deviation estimates of value functions associated with the BART ITR are calculated as the standard deviation across 100 runs rather than the posterior standard deviation.

Table 2.1: *Description of simulation settings 1-6 and simulation results for $n = 20000$ based on 100 replicates: mean (SD) of misclassification rates and value functions. Methods under comparison include the l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), direct learning (D-learning), and Bayesian additive regression trees (BART). The smallest misclassification rates and the largest value functions for each setting are in bold.*

Setting	Functional form of interaction	True optimal treatment	Covariates type	Method	Misclassification	Value
1	tree $\Delta_1(\mathbf{X}) = 4 \times I(X_1 > 0.5) - 2$ $\Delta_2(\mathbf{X}) = 2 \times I(X_2 \geq 0.5)I(X_3 < 0.25) - 1$ $\Delta_3(\mathbf{X}) = 0$	1 or 2 or 3	continuous $X_1, \ldots, X_5 \sim \mathcal{U}\{-1, 1\}$	l_1-PLS-HGL	0.101 (0.016)	1.226 (0.010)
				l_1-PLS-GL	0.094 (0.015)	1.229 (0.010)
				ACWL	0.028 (0.040)	1.276 (0.023)
				D-learning	0.100 (0.020)	1.226 (0.013)
				BART	**0.010 (0.004)**	**1.286 (0.003)**
2	linear $\Delta_1(\mathbf{X}) = 3X_1 - 2X_2$ $\Delta_2(\mathbf{X}) = 5X_3 - X_4 + X_5 - 1$ $\Delta_3(\mathbf{X}) = 0$	1 or 2 or 3	continuous $X_1, \ldots, X_5 \sim \mathcal{U}\{-1, 1\}$	l_1-PLS-HGL	0.015 (0.004)	1.736 (0.003)
				l_1-PLS-GL	**0.013 (0.004)**	**1.737 (0.003)**
				ACWL	0.171 (0.020)	1.662 (0.016)
				D-learning	0.018 (0.005)	**1.737 (0.004)**
				BART	0.056 (0.004)	1.730 (0.006)
3	nonlinear $\Delta_1(\mathbf{X}) = 3X_1^2 - \exp(X_2)$ $\Delta_2(\mathbf{X}) = X_3X_4$ $\Delta_3(\mathbf{X}) = 0$	1 or 2 or 3	continuous $X_1, \ldots, X_5 \sim \mathcal{U}\{-1, 1\}$	l_1-PLS-HGL	0.566 (0.012)	1.089 (0.008)
				l_1-PLS-GL	0.565 (0.014)	1.088 (0.011)
				ACWL	0.561 (0.016)	1.089 (0.009)
				D-learning	0.572 (0.013)	1.087 (0.008)
				BART	**0.192 (0.038)**	**1.209 (0.010)**
4	nonlinear $\Delta_1(\mathbf{X}) = 3X_1^2 - \exp(X_2)$ $\Delta_2(\mathbf{X}) = X_3^3$ $\Delta_3(\mathbf{X}) = 0$	1 or 2 or 3	continuous $X_1, \ldots, X_5 \sim \mathcal{U}\{-1, 1\}$	l_1-PLS-HGL	0.350 (0.011)	1.129 (0.004)
				l_1-PLS-GL	0.352 (0.010)	1.128 (0.004)
				ACWL	0.362 (0.011)	1.118 (0.004)
				D-learning	0.359 (0.016)	1.129 (0.004)
				BART	**0.163 (0.045)**	**1.220 (0.012)**
5	nonlinear $\Delta_1(\mathbf{X}) = 2\{I(X_1 = 1) + I(X_1 = 2)\}X_2 - 1$ $\Delta_2(\mathbf{X}) = 5I(X_1 = 5)X_3 - 2$ $\Delta_3(\mathbf{X}) = 0$	1 or 2 or 3	continuous + binary + categorical $X_1 \sim$ discrete uniform$\{1, 5\}$ $X_2 \sim$ Bern(0.5) $X_3, X_4, X_5 \sim \mathcal{U}\{-1, 1\}$	l_1-PLS-HGL	0.077 (0.019)	1.101 (0.012)
				l_1-PLS-GL	0.078 (0.018)	1.101 (0.011)
				ACWL	0.029 (0.028)	1.129 (0.016)
				D-learning	0.090 (0.032)	1.094 (0.019)
				BART	**0.007 (0.005)**	**1.142 (0.002)**
6	tree $\Delta_1(\mathbf{X}) = I(X_1 > 0.5) + 2$ $\Delta_2(\mathbf{X}) = 2 \times I(X_2 \geq 0.5)I(X_3 < 0.25) - 3$ $\Delta_3(\mathbf{X}) = 0$	1 for everyone	continuous $X_1, \ldots, X_5 \sim \mathcal{U}\{-1, 1\}$	l_1-PLS-HGL	**0.000 (0.000)**	**2.093 (0.000)**
				l_1-PLS-GL	**0.000 (0.000)**	**2.093 (0.000)**
				ACWL	**0.000 (0.000)**	**2.093 (0.000)**
				D-learning	**0.000 (0.000)**	**2.093 (0.000)**
				BART	**0.000 (0.000)**	**2.093 (0.000)**

Simulation results are presented in Table 2.1. As expected, when the true underlying decision boundaries are tree-type (setting 1), ACWL that builds on decision trees performs better than l_1-PLS-HGL, l_1-PLS-GL, and D-learning. BART leads to a slightly smaller misclassification rate than ACWL in this setting. In contrast, when decision boundaries are linear (setting 2), l_1-PLS-HGL, l_1-PLS-GL, and D-learning perform similarly well and much better than ACWL. BART is superior to ACWL but slightly worse than the other three methods in this case, possibly due to its "over-parameterisation" for linear effects. None of these methods manage to capture the nonlinear structures in settings 3 and 4 properly, and misclassification rates are not as low as in other settings for all methods despite the large training sample size. However, BART outperforms the rest of the methods to a large extent. This is not surprising given the flexibility of BART. All methods achieve good performance when some covariates are discrete (setting 5). Results for setting 6 imply that when the true optimal treatment is the same for all subjects ("trivial" decision rule that assigns all to the marginally best treatment), all methods perform perfectly with no misclassification. This may be due to the fact that the sample size we consider is sufficiently large ($n = 20000$), and the "trivial" decision rule is easier to be "learnt" compared to more complicated ones (where qualitative interactions between A and $\mathbf{X}$ exist). Similar phenomenon has been observed in the application of these methods to the INTERVAL data when our objective is to maximise the total units of blood collected by the blood service (more details will follow in the next section). As an aside, when the sample size is smaller, ACWL results in non-zero misclassification rates in setting 6, while the other methods still lead to almost perfect classification in this setting. For example, when $n = 200$, the average misclassification rate by ACWL is 0.188, while for the remaining methods, the average misclassification rate is less than 0.006.

Overall, our simulation studies suggest that unless we have *a priori* information on the type of underlying interaction effects (e.g. linear additive or tree-type), the BART multi-arm ITR estimation method should be a good choice in general since it is robust and performs better than or comparable to the other competing methods regardless of the functional form of interaction terms. We note that BART takes longer to run than other non-Bayesian methods, especially in the large sample size case that we examine in this chapter (on a Windows-based computing system with 1 core, 3.40 GHz Intel processor, BART takes about 200 seconds per run while other methods take less than 20 seconds per run when $n = 20000$). Parallel processing with multi-threading can be used to speed up

the computation of BART (Sparapani et al., 2019). We also note that in settings with nonlinear decision boundaries, the performance of l_1-PLS and D-learning may be improved if we use polynomials of higher degrees as basis functions. For D-learning, we would also expect the nonlinear version which estimates the optimal decision function using the component selection and smoothing operator (COSSO) to perform better than the linear version considered in our simulation studies if the underlying decision boundaries are nonlinear. However, our numerical experiments suggest that nonlinear D-learning with COSSO is much more computationally demanding and performs less well than BART when $n = 20000$. On average, each replicate of BART takes about 200 seconds, while each replicate of nonlinear D-learning (3 arms) takes more than an hour to run. As expected, in settings with nonlinear decision boundaries, we do observe an improvement in the performance by using nonlinear D-learning compared to linear D-learning. For example, for setting 1 (tree-type qualitative interaction), the average misclassification rate and value function for nonlinear D-learning are 0.060 and 1.260, while the average misclassification rate and value function for linear D-learning are 0.100 and 1.226. However, when compared with BART, nonlinear D-learning results in a larger misclassification rate and a smaller value function (for BART, the average misclassification rate and the value function are 0.010 and 1.286). Similarly, for setting 3 (nonlinear qualitative interaction), the average misclassification rates for nonlinear D-learning, linear D-learning, and BART are 0.348, 0.572, and 0.192, respectively. The average value functions for nonlinear D-learning, linear D-learning, and BART are 1.177, 1.087, and 1.209, respectively. Therefore, we may still prefer using BART if we believe the underlying decision boundaries are nonlinear.

2.4.2 Scenarios with correlated covariates

We conduct additional simulation studies under scenarios with correlated covariates when $n = 20000$. To demonstrate the idea, we pick settings 1, 2, 3, and 6 in Section 2.4.1. Five correlated covariates, $X_1, \ldots, X_5$, are generated. We examine the case with pairwise correlation coefficient being 0.3 and 0.6, respectively. Results are presented in Table 2.2.

Table 2.2: *Simulation results under scenarios with correlated covariates based on 100 replicates ($n = 20000$): mean (SD) of misclassification rates and value functions. We examine the case with pairwise correlation between covariates being 0.3 and 0.6. Methods under comparison include the l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), direct learning (D-learning), and Bayesian additive regression trees (BART). The smallest misclassification rates and the largest value functions for each setting are in bold.*

		pairwise correlation = 0.3		pairwise correlation = 0.6	
Setting	Method	Misclassification	Value	Misclassification	Value
1 tree-type qualitative interaction	l_1-PLS-HGL	0.102 (0.009)	1.220 (0.010)	0.074 (0.008)	1.213 (0.012)
	l_1-PLS-GL	0.097 (0.011)	1.223 (0.010)	0.074 (0.008)	1.212 (0.011)
	ACWL	0.080 (0.019)	1.242 (0.010)	0.052 (0.002)	1.243 (0.002)
	D-learning	0.097 (0.012)	1.225 (0.013)	0.075 (0.013)	1.213 (0.017)
	BART	**0.009 (0.004)**	**1.279 (0.003)**	**0.010 (0.005)**	**1.271 (0.004)**
2 linear qualitative interaction	l_1-PLS-HGL	0.020 (0.004)	1.626 (0.003)	0.027 (0.005)	1.534 (0.003)
	l_1-PLS-GL	**0.015 (0.004)**	**1.628 (0.003)**	**0.016 (0.004)**	**1.538 (0.003)**
	ACWL	0.177 (0.017)	1.554 (0.011)	0.170 (0.021)	1.481 (0.009)
	D-learning	0.022 (0.007)	**1.628 (0.003)**	0.023 (0.006)	1.537 (0.004)
	BART	0.063 (0.004)	1.626 (0.007)	0.070 (0.005)	1.532 (0.006)
3 nonlinear qualitative interaction	l_1-PLS-HGL	0.531 (0.016)	1.100 (0.006)	0.445 (0.002)	1.137 (0.003)
	l_1-PLS-GL	0.534 (0.018)	1.097 (0.009)	0.446 (0.005)	1.137 (0.004)
	ACWL	0.529 (0.016)	1.101 (0.006)	0.445 (0.004)	1.133 (0.003)
	D-learning	0.538 (0.024)	1.096 (0.009)	0.453 (0.027)	1.137 (0.006)
	BART	**0.200 (0.045)**	**1.228 (0.012)**	**0.251 (0.030)**	**1.230 (0.010)**
6 tree-type quantitative interaction	l_1-PLS-HGL	**0.000 (0.000)**	**2.114 (0.000)**	**0.000 (0.000)**	**2.114 (0.000)**
	l_1-PLS-GL	**0.000 (0.000)**	**2.114 (0.000)**	**0.000 (0.000)**	**2.114 (0.000)**
	ACWL	**0.000 (0.000)**	**2.114 (0.000)**	**0.000 (0.000)**	**2.114 (0.000)**
	D-learning	**0.000 (0.000)**	**2.114 (0.000)**	**0.000 (0.000)**	**2.114 (0.000)**
	BART	**0.000 (0.000)**	**2.114 (0.000)**	**0.000 (0.000)**	**2.114 (0.000)**

We get similar comparative conclusions across methods as with independent covariates: except for the case with linear decision boundaries (setting 2), BART performs as well or better than the other competing methods in all settings. In addition, when the true optimal is the "one-size-fits-all" rule (setting 6), all methods recover the true optimal ITR with no misclassification.

2.4.3 Scenarios with moderate-dimensional covariates

In this section, we conduct simulation studies under scenarios with more covariates and more prognostic factors than those considered in Section 2.4.1. We fix n at 20000 and pick settings 1, 2, 3 and 6 as representative settings for illustration. In Section 2.4.1, we examine the case with $p = 5$ covariates. Here, we run additional simulations with

(i) $p = 20$ covariates, among which 10 covariates are involved in the main effect, $m(\mathbf{X}) = 1+0.1X_1-0.2X_2-0.3X_3+0.4X_4-0.5X_5+X_6+X_7+X_8-X_9-X_{10}$.

(ii) $p = 50$ covariates, among which 20 covariates are involved in the main effect, $m(\mathbf{X}) = 1 + 0.1X_1 - 0.2X_2 - 0.3X_3 + 0.4X_4 - 0.5X_5 + X_6 + X_7 + X_8 - X_9 - X_{10} + 0.5(X_{11} + X_{12} + X_{13} + X_{14} + X_{15}) - 0.8(X_{16} + X_{17} + X_{18} + X_{19} + X_{20})$.

For both (i) and (ii), the treatment-covariate interaction effects $\Delta(\mathbf{X}, A)$ remain the same as those in the base scenarios (described in Table 2.1). Simulation results are summarised in Table 2.3.

Table 2.3: *Simulation results under scenarios with different numbers of covariates based on 100 replicates ($n = 20000$): mean (SD) of misclassification rates and value functions. We examine the case with $p = 20$ and $p = 50$. Methods under comparison include the l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), direct learning (D-learning), and Bayesian additive regression trees (BART). The smallest misclassification rates and the largest value functions for each setting are in bold.*

		$p = 20$		$p = 50$	
Setting	Method	Misclassification	Value	Misclassification	Value
1 tree-type qualitative interaction	l_1-PLS-HGL	0.108 (0.017)	1.283 (0.015)	0.112 (0.018)	1.267 (0.014)
	l_1-PLS-GL	0.095 (0.014)	1.293 (0.013)	0.101 (0.014)	1.276 (0.013)
	ACWL	0.017 (0.034)	1.348 (0.021)	**0.010 (0.026)**	**1.336 (0.014)**
	D-learning	0.121 (0.019)	1.276 (0.016)	0.134 (0.018)	1.251 (0.017)
	BART	**0.016 (0.007)**	**1.352 (0.005)**	0.030 (0.017)	1.323 (0.011)
2 linear qualitative interaction	l_1-PLS-HGL	**0.015 (0.004)**	1.774 (0.005)	**0.015 (0.003)**	**1.804 (0.005)**
	l_1-PLS-GL	0.020 (0.003)	**1.775 (0.004)**	0.024 (0.004)	1.800 (0.007)
	ACWL	0.175 (0.020)	1.682 (0.022)	0.173 (0.021)	1.712 (0.023)
	D-learning	0.028 (0.007)	1.774 (0.006)	0.036 (0.008)	1.798 (0.007)
	BART	0.066 (0.004)	1.765 (0.009)	0.081 (0.006)	1.777 (0.013)
3 nonlinear qualitative interaction	l_1-PLS-HGL	0.565 (0.006)	1.140 (0.013)	0.567 (0.006)	1.121 (0.020)
	l_1-PLS-GL	0.565 (0.007)	1.136 (0.016)	0.566 (0.005)	1.121 (0.020)
	ACWL	0.559 (0.012)	1.146 (0.011)	0.561 (0.009)	1.120 (0.015)
	D-learning	0.571 (0.009)	1.139 (0.014)	0.574 (0.011)	1.118 (0.020)
	BART	**0.316 (0.053)**	**1.226 (0.017)**	**0.364 (0.027)**	**1.191 (0.022)**
6 tree-type quantitative interaction	l_1-PLS-HGL	**0.000 (0.000)**	**2.151 (0.000)**	**0.000 (0.000)**	**2.123 (0.000)**
	l_1-PLS-GL	**0.000 (0.000)**	**2.151 (0.000)**	**0.000 (0.000)**	**2.123 (0.000)**
	ACWL	**0.000 (0.000)**	**2.151 (0.000)**	**0.000 (0.000)**	**2.123 (0.000)**
	D-learning	**0.000 (0.000)**	**2.151 (0.000)**	**0.000 (0.000)**	**2.123 (0.000)**
	BART	**0.000 (0.000)**	**2.151 (0.000)**	**0.000 (0.000)**	**2.123 (0.000)**

Findings from these moderate-dimensional scenarios are similar to those from low-dimensional scenarios: when decision boundaries are tree-type (setting 1), ACWL and BART perform much better than the other competing methods, while when decision boundaries are linear (setting 2), l_1-PLS-HGL, l_1-PLS-GL and D-learning perform similarly, and significantly better than ACWL as expected. BART outperforms all the other methods under nonlinear decision boundaries (setting 3) due to its flexibility. When the true optimal treatment is the same for everyone (setting 6), all methods perform perfectly with no misclassification. We note that except for BART, all the other methods perform variable selection intrinsically when estimating the optimal ITR (l_1-PLS-HGL,

l_1-PLS-GL and D-learning perform LASSO-type variable selection, while ACWL performs variable selection via the node splitting process). However, BART still performs reasonably well for $p = 20$ and $p = 50$ (when $n = 20000$) since we still have $n \gg p$. The performance of BART can be further improved using the variable selection method proposed by Linero (2018) for large p at a slightly higher computational cost.

2.4.4 Variations of setting 6

In this section, we focus on simulation settings in which the true optimal treatment for everyone is the same (i.e. the "one-size-fits-all" rule) and we examine some variations of setting 6.

We first consider the case where we increase the noise or reduce the signal in setting 6. We simulate data assuming $Y \sim N(m(\mathbf{X})+t\times\{I(A=1)\Delta_1(\mathbf{X})+I(A=2)\Delta_2(\mathbf{X})+I(A=3)\Delta_3(\mathbf{X})\}, \sigma^2)$. The functional forms of $m(\mathbf{X})$, $\Delta_1(\mathbf{X})$, $\Delta_2(\mathbf{X})$, and $\Delta_3(\mathbf{X})$ are the same as those in setting 6, which implies that treatment 1 is the optimal treatment for everyone. However, in setting 6, we set $t = 0.5$ and $\sigma = 1$, while here we vary values of t and σ with the sample size n fixed at 20000. The results are shown in Table 2.4.

We observe that when the noise level increases to $\sigma = 5$, all methods still perform perfectly if the "one-size-fits-all" rule is the optimal treatment regime. When we fix σ and reduce the value of t from 0.5 to 0.1, the perfect classification results remain. These results are not surprising since the sample size we consider is sufficiently large.

In setting 6, the quantitative interactions are tree-type. We also consider other types of quantitative interactions, such as linear additive and nonlinear (Table 2.5). According to Table 2.5, when the true optimal treatment is the same for all subjects ("trivial" decision rule that assigns all to the marginally best treatment), all methods perform perfectly with no misclassification, regardless of whether the functional forms of quantitative interactions are linear, nonlinear, or tree-type.

Table 2.4: *Simulation results based on 100 replicates ($n = 20000$) under the setting where the optimal ITR is the "one-size-fits-all" rule with different levels of signals and noises: mean (SD) of misclassification rates and value functions. Methods under comparison include the l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), direct learning (D-learning), and Bayesian additive regression trees (BART). The smallest misclassification rates and the largest value functions for each setting are in bold.*

Scenario	Method	Misclassification	Value
$t = 0.5$, $\sigma = 1$ (setting 6)	l_1-PLS-HGL	**0.000 (0.000)**	**2.093 (0.000)**
	l_1-PLS-GL	**0.000 (0.000)**	**2.093 (0.000)**
	ACWL	**0.000 (0.000)**	**2.093 (0.000)**
	D-learning	**0.000 (0.000)**	**2.093 (0.000)**
	BART	**0.000 (0.000)**	**2.093 (0.000)**
$t = 0.5$, $\sigma = 5$ (larger noise)	l_1-PLS-HGL	**0.000 (0.000)**	**1.989 (0.000)**
	l_1-PLS-GL	**0.000 (0.000)**	**1.989 (0.000)**
	ACWL	**0.000 (0.000)**	**1.989 (0.000)**
	D-learning	**0.000 (0.000)**	**1.989 (0.000)**
	BART	0.001 (0.003)	1.988 (0.004)
$t = 0.1$, $\sigma = 1$ (smaller signal)	l_1-PLS-HGL	**0.000 (0.000)**	**1.194 (0.000)**
	l_1-PLS-GL	**0.000 (0.000)**	**1.194 (0.000)**
	ACWL	**0.000 (0.000)**	**1.194 (0.000)**
	D-learning	**0.000 (0.000)**	**1.194 (0.000)**
	BART	0.001 (0.003)	**1.194 (0.001)**

Table 2.5: *Simulation results based on 100 replicates ($n = 20000$) under the setting where the optimal ITR is the "one-size-fits-all" rule with different types of quantitative interactions: mean (SD) of misclassification rates and value functions. Methods under comparison include the l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), direct learning (D-learning), and Bayesian additive regression trees (BART). The smallest misclassification rates and the largest value functions for each setting are in bold.*

Forms of quantitative interactions	Method	Misclassification	Value
tree $\Delta_1(\mathbf{X}) = I(X_1 > 0.5) + 2$ $\Delta_2(\mathbf{X}) = 2 \times I(X_2 \geq 0.5)I(X_3 < 0.25) - 3$ $\Delta_3(\mathbf{X}) = 0$	l_1-PLS-HGL	**0.000 (0.000)**	**2.093 (0.000)**
	l_1-PLS-GL	**0.000 (0.000)**	**2.093 (0.000)**
	ACWL	**0.000 (0.000)**	**2.093 (0.000)**
	D-learning	**0.000 (0.000)**	**2.093 (0.000)**
	BART	**0.000 (0.000)**	**2.093 (0.000)**
linear $\Delta_1(\mathbf{X}) = 3X_1 - 2X_2 + 6$ $\Delta_2(\mathbf{X}) = 5X_3 - X_4 + X_5 - 8$ $\Delta_3(\mathbf{X}) = 0$	l_1-PLS-HGL	**0.000 (0.000)**	**3.975 (0.000)**
	l_1-PLS-GL	**0.000 (0.000)**	**3.975 (0.000)**
	ACWL	**0.000 (0.000)**	**3.975 (0.000)**
	D-learning	**0.000 (0.000)**	**3.975 (0.000)**
	BART	**0.000 (0.000)**	**3.975 (0.000)**
nonlinear $\Delta_1(\mathbf{X}) = 3X_1^2 - \exp(X_2) + 3$ $\Delta_2(\mathbf{X}) = X_3^3 - 2$ $\Delta_3(\mathbf{X}) = 0$	l_1-PLS-HGL	**0.000 (0.000)**	**2.386 (0.000)**
	l_1-PLS-GL	**0.000 (0.000)**	**2.386 (0.000)**
	ACWL	**0.000 (0.000)**	**2.386 (0.000)**
	D-learning	**0.000 (0.000)**	**2.386 (0.000)**
	BART	**0.000 (0.000)**	**2.386 (0.000)**

2.5 Application to the INTERVAL trial

To illustrate the use of multi-arm ITR estimation methods on large-scale clinical trials, we apply the five methods compared in Section 2.4 to the data from male donors in the INTERVAL trial and estimate each blood donor's optimal inter-donation interval. We also compare the personalised donation strategies with "one-size-fits-all" donor assignment rules that recommend the same inter-donation intervals of 8, 10 or 12 weeks for all male donors.

The INTERVAL trial achieved 99.5% completeness in the primary outcome (Di Angelantonio et al., 2017). Regarding the handling of missing values in donors' baseline characteristics, we add a new category (the "missing"/"unknown" category) to each categorical covariate with missing data, and use the complete-case analysis to deal with missingness in continuous covariates (missingness proportions are less than 2% for all the continuous covariates that are considered in our analysis). After data processing (exclude donors who had zero attendance over the 2-year trial period and donors with missing continuous covariate values), 20574 male blood donors are included in the analysis following the intention-to-treat principle according to donors' assigned randomised groups. We examine the covariate balance after data cleaning (Appendix A.2) and conclude that the data cleaning process does not distort the balance of baseline covariates across randomised groups.

2.5.1 Outcomes of interest

We consider two outcomes, namely, the total units of blood collected by the blood service per donor over a 2-year period (the standard practice is to donate 1 unit of blood per session, with a full donation unit containing 470 ml of whole blood), denoted by G, and the rate of low Hb deferrals per donor attendance during the same period (calculated as the total number of "at session" deferrals for low Hb divided by the total number of attendances in the 2-year trial period), denoted by R. We note that these two outcomes are not independent: assigning a donor to a more frequent inter-donation interval in principle will lead to an increase in the total units of blood collected. However, this increased frequency may have the opposite effect through increased risks of deferrals for low Hb, which may consequently cause existing donors to come back less often and even to leave the donor register permanently (Di Angelantonio et al., 2017; Grieve et al., 2018; Moore et al., 2014). Potential loss of donors may have a cost impact (Grieve et al.,

2018). The current donor loss rate following a deferral for low Hb is 40-50%, and this would incur substantial costs (approximately £2.3 million in the worst-case scenario) for the blood service to recruit sufficient new donors and stabilise the donor base (Green and Davies, 2018; Hillgrove et al., 2011). Therefore, when recommending the optimal inter-donation interval to a blood donor, there is a trade-off between the benefit and the risk: neither the optimal ITR solely based on the benefit nor that solely based on the risk may be acceptable, and it is generally not possible to find a strategy that optimises both (maximises benefit and minimises risk) simultaneously. The goal of maximising the total units of blood collected needs to be considered in conjunction with controlling for the deferrals for low Hb. This motivates us to construct a utility outcome which "discounts" the units of blood collected by the incidences of deferrals for low Hb as follows:

$$U = G - b \times \widetilde{R}, \tag{2.20}$$

where G is the gain/benefit (total units of blood collected in the 2-year trial period), $\widetilde{R}$ is the risk (number of deferrals for low Hb in the 2-year trial period: $\widetilde{R} = R \times$ total number of attendances over 2 years), and b is the "trade-off" parameter reflecting the equivalent benefit loss for one unit increase in the risk. In the context of the INTERVAL trial, b is interpreted as "the equivalent loss in total units of blood collected by the blood service per donor over 2 years for one extra deferral for low Hb per donor attendance during the same period". We examine different values of b within a range considered to be reasonable by NHSBT and vary b from 1 to 5 at an increment of 1 to see how results change with this parameter. A range of b from 1 to 5 covers the range of the extra costs of deferrals for low Hb considering reduced efficiency of collection, reduced donor retention and increase in recruitment of the many new donors to replace a regular donor who retires from donation.

We assume a larger outcome to be more desirable when we introduce the statistical framework in Section 2.2.1. This holds for the benefit and the utility outcomes, but is not the case for the low Hb deferral rate (R). We address this issue by considering the maximisation of $1 - R$ instead, which is equivalent to the minimisation of R. In addition, the low Hb deferral rate is a proportion and so we use the arcsine square root transformation for variance stabilisation (Tukey, 1977). This transformation is monotonically increasing, and thus rank-preserving.

2.5.2 Baseline covariates

Based on the findings reported in Di Angelantonio et al. (2017), we include the following 19 variables measured at each donor's baseline visit:

- Continuous: age, body mass index, Short Form Health Survey version 2 (SF-36v2) physical component score and mental component score, units of whole blood donations in the 2 years before enrolment into the trial, haemoglobin level, white blood cell count, red blood cell count, mean corpuscular haemoglobin, mean corpuscular volume, and platelet count.

- Categorical: ethnicity (Asian, Black, Mixed, White, Other, Unknown), blood group (A+, A−, AB+, AB−, B+, B−, O+, O−), iron prescription (Yes, No, Unknown), smoke ever (Yes, No, Unknown), smoke currently (Yes, No, Unknown), alcohol ever (Yes, No, Unknown), alcohol currently (Yes, No, Unknown), and new or returning donor status (New donor, Returning donor; "new" and "returning" have been defined according to the classification used by NHSBT – an individual who has not provided a full donation prior to the trial is considered as a new donor (Moore et al., 2016)).

2.5.3 Evaluation criteria

We calculate proportions of donors assigned to each of the three inter-donation intervals according to the optimal ITR estimated using different methods. We are also interested in the quantity "ITR effect" or "benefit function" (Qiu et al., 2018; Xu et al., 2015). The ITR effect, δ, associated with the rule $\mathcal{D}(\mathbf{X})$ is defined as:

$$\delta(\mathcal{D}(\mathbf{X})) = E\{Y|\mathbf{X}, A = \mathcal{D}(\mathbf{X})\} - E\{Y|\mathbf{X}, A \neq \mathcal{D}(\mathbf{X})\}. \tag{2.21}$$

In the context of blood donation, this quantity can be interpreted as the average increase in the outcome when assigning donors according to the rule $\mathcal{D}(\mathbf{X})$, or more precisely, the difference in the value function between the strategy that assigns donors to inter-donation intervals according to $\mathcal{D}(\mathbf{X})$ and that assigns donors to inter-donation intervals different from $\mathcal{D}(\mathbf{X})$. For an ITR $\mathcal{D}(\mathbf{X})$, we can verify that

$$E\left[\frac{\{Y - V(\mathcal{D})\}I\{A = \mathcal{D}(\mathbf{X})\}}{P(A|\mathbf{X})}\right] = 0, \tag{2.22}$$

where $V(\mathcal{D}) = E\Big[YI\{A = \mathcal{D}(\mathbf{X})\}/P(A|\mathbf{X})\Big]$. Therefore, an unbiased estimator of $V(\mathcal{D})$ is

$$\frac{\mathbb{P}_n[YI\{A = \mathcal{D}(\mathbf{X})\}/P(A|\mathbf{X})]}{\mathbb{P}_n[I\{A = \mathcal{D}(\mathbf{X})\}/P(A|\mathbf{X})]}, \tag{2.23}$$

where $\mathbb{P}_n$ denotes the empirical average. The ITR effect of $\mathcal{D}(\mathbf{X})$ can subsequently be estimated empirically as:

$$\frac{\mathbb{P}_n[YI\{A = \mathcal{D}(\mathbf{X})\}/P(A|\mathbf{X})]}{\mathbb{P}_n[I\{A = \mathcal{D}(\mathbf{X})\}/P(A|\mathbf{X})]} - \frac{\mathbb{P}_n[YI\{A \neq \mathcal{D}(\mathbf{X})\}/P(A|\mathbf{X})]}{\mathbb{P}_n[I\{A \neq \mathcal{D}(\mathbf{X})\}/P(A|\mathbf{X})]}. \tag{2.24}$$

In the INTERVAL trial, the donor assignment proportions are equal for three randomised groups (i.e. $P(A = a|\,\mathbf{X}) = P(A = a) = 1/3$ for all a), and (2.24) can be considered as the difference in the average outcome between donors whose assigned inter-donation intervals in the trial are the same as $\mathcal{D}(\mathbf{X})$ (i.e. average across all donors whose $a = \mathcal{D}(\mathbf{x})$) and those whose assigned inter-donation intervals are different from $\mathcal{D}(\mathbf{X})$ (i.e. average across all donors whose $a \neq \mathcal{D}(\mathbf{x})$).

When two outcomes (G and R) are analysed separately, in addition to ITR effects of the estimated optimal assignment rule $\widehat{\mathcal{D}^*}(\mathbf{X})$ on the outcome that we aim to optimise, we also calculate the effect of assigning donors according to $\widehat{\mathcal{D}^*}(\mathbf{X})$ on the other outcome that we do not take into consideration when estimating $\mathcal{D}^*(\mathbf{X})$.

When we consider the combined outcome and aim at maximising the utility score, ITR effects on the units of blood collected, G, the low Hb deferral rates, R, and the utility score, U, are computed by replacing Y in (2.24) with G, R, and U, respectively. A larger ITR effect on donation and utility, and a smaller ITR effect on deferral are more desirable.

2.5.4 Non-Bayesian approaches: l_1-PLS, ACWL, and D-learning

We first apply the l_1-PLS with HGL/GL variable selection, ACWL, and D-learning to the data from male donors in the INTERVAL trial.

In Section 2.5.4.1, we present cross-validation-based results. Specifically, in each of the analyses, we randomly split the data into five roughly equal-sized folds. The optimal decision rules are obtained by applying each method to four folds of the data (training data), and then each decision rule is applied to predict the optimal inter-donation interval for donors in the remaining fold of the data (validation data). This process is iterated until all five folds have been used as the

validation set and all donors have an out-of-sample prediction of their optimal inter-donation interval. We repeat this whole procedure 100 times with different fold partitions. All tuning parameters are selected via 5-fold cross-validation. The means and the standard deviations of empirical assignment proportions and ITR effects evaluated on the validation data across 100 splits are reported.

We note that the standard deviation estimates based on cross-validation capture the randomness of data-splitting ("repeatability") rather than the uncertainty of the observed data. In Section 2.5.4.2, we provide bootstrap-based standard deviation estimates that reflect the "biological variation".

2.5.4.1 Cross-validation-based results

Target two outcomes separately Table 2.6 presents the results for analysing the donation and deferral outcomes separately. As expected from the simulation results, we observe a consistent pattern across different methods in these extreme cases where true optimal decisions should be "almost trivial" and the non-personalised strategy that assigns everyone to the marginally best "treatment" should lead to almost optimal outcomes. As suggested by assignment proportions, almost all donors (ranging from 99.4% to 100.0%) are assigned to the shortest inter-donation interval (8-week) if the goal is to maximise the total units of blood collected by the blood service, and ITRs estimated using these personalised medicine-based methods are indeed very close to "one-size-fits-all" rules. In contrast, if our aim is to minimise the low Hb deferral rates, then the longest inter-donation interval (12-week) should be recommended for most donors (ranging from 94.4% to 99.7%). These findings are consistent with the results from the primary analysis of the INTERVAL trial data (Di Angelantonio et al., 2017).

For comparison, we also calculate the ITR effects on donation and deferral of three non-personalised rules where the same inter-donation interval is recommended for all donors. ITR effects of non-personalised (fixed) rules measure the difference in the average outcome between donors whose assigned inter-donation intervals in the trial are the same as the one specified in the fixed rule and those whose assigned inter-donation intervals are different from that specified in the fixed rule. We still use (2.24) to calculate empirical ITR effects of fixed rules except that $\mathcal{D}(\mathbf{X})$ is replaced with fixed rules that do not depend on $\mathbf{X}$. ITR effects on donation are -1.248, -0.077, and 1.315 for assigning all donors to the 12-, 10-, and 8-week inter-donation intervals, respectively; and ITR effects on deferral are -0.025, -0.002, and 0.027 for assigning all donors to the 12-, 10-,

Table 2.6: *Applications of the l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), and direct learning (D-learning) to the data from male donors in the INTERVAL trial. Means and standard deviations (in parenthesis) of assignment proportions in % and empirical ITR effects on donation and deferral outcomes across 100 repetitions of 5-fold cross-validation are reported. ITR effects measure the difference in the average outcome between donors whose assigned inter-donation intervals in the trial are optimal (with respect to the method used to estimate the ITR) and those whose assigned inter-donation intervals are non-optimal. A larger ITR effect on donation and a smaller ITR effect on deferral are more desirable. The first four and last four rows correspond to the target being maximising total units of blood collected by the blood service, and minimising the low Hb deferral rates, respectively.*

		Assignment Percentages			ITR Effects	
Target Outcome	Method	12 weeks	10 weeks	8 weeks	Donation	Deferral
Donation	l_1-PLS-HGL	0.1 (0.0)	0.3 (0.0)	99.6 (0.0)	1.308 (0.004)	0.026 (0.000)
	l_1-PLS-GL	0.0 (0.0)	0.3 (0.3)	99.7 (0.3)	1.311 (0.005)	0.027 (0.000)
	ACWL	0.0 (0.0)	0.0 (0.0)	100.0 (0.0)	1.315 (0.000)	0.027 (0.000)
	D-learning	0.3 (0.1)	0.3 (0.2)	99.4 (0.2)	1.307 (0.006)	0.027 (0.000)
Deferral	l_1-PLS-HGL	94.4 (0.6)	5.5 (0.6)	0.0 (0.0)	-1.188 (0.010)	-0.024 (0.000)
	l_1-PLS-GL	99.7 (0.6)	0.2 (0.5)	0.0 (0.1)	-1.246 (0.006)	-0.024 (0.000)
	ACWL	99.7 (0.6)	0.3 (0.6)	0.0 (0.0)	-1.244 (0.007)	-0.025 (0.000)
	D-learning	95.7 (0.5)	4.1 (0.5)	0.2 (0.1)	-1.200 (0.011)	-0.024 (0.000)

and 8-week inter-donation intervals, respectively. This suggests that if the sole interest is in collecting more blood, the non-personalised rule that recommends all male donors to donate every 8 weeks leads to the largest increase in the units of blood collected compared to personalised rules estimated using different methods. On the other hand, if we are only concerned with minimising deferrals for low Hb, then the non-personalised rule that assigns all male donors to the 12-week inter-donation interval yields the largest reduction in the rate of low Hb deferrals compared to personalised rules. However, we also observe from Table 2.6 that by following the optimal rule for maximising the total units of blood collected by the blood service, the average increase in blood donations is about 1.31 units (616 ml) per donor over 2 years, but at the same time, there is also an increasing number of deferrals for low Hb at about 2.7 per 100 donor attendances on average. This is consistent with our intuition: an "optimal" rule that maximises clinical benefits also leads to safety concerns (high risks of adverse events), and vice versa. If the aim is to maximise the total units of blood collected by the blood service, then assigning donors to the optimal ITR derived with this target will lead to an increase in units of blood collected with the consequence of increased low Hb deferral rates. Similarly, if donors are assigned according to the optimal

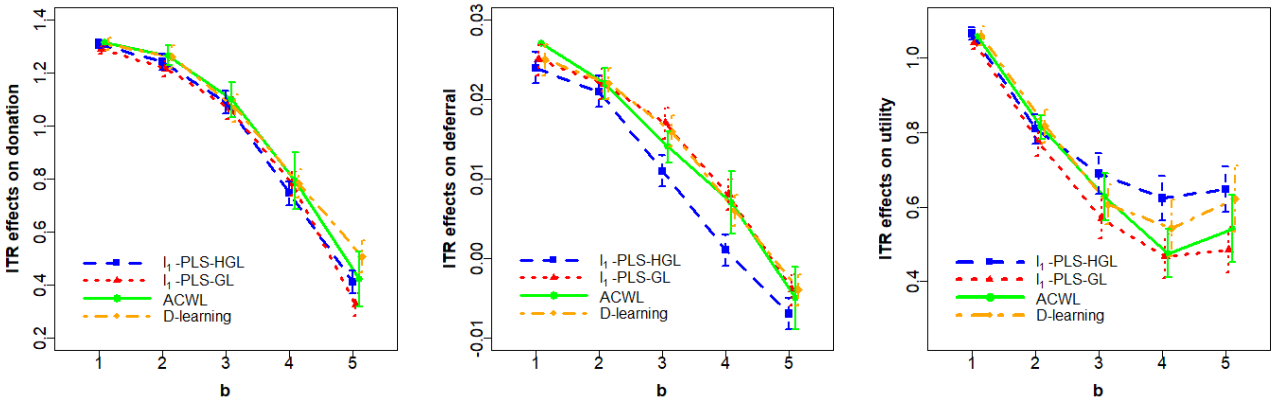
ITR estimated for minimising the low Hb deferral rates, there will be a decrease in the low Hb deferral rates, but we will also see a substantial decrease in the total units of blood collected by the blood service. This clearly illustrates that it is necessary to strike a balance between the two "competing" outcomes by maximising the utility score which incorporates the trade-off between the benefit and the risk.

Target the utility outcome Table 2.7 summarises allocation proportions and ITR effects associated with $\widehat{\mathcal{D}^*}(\mathbf{X})$ obtained by maximising the utility score (with different values of the trade-off parameter b), and Figures 2.2 (a)-(c) plot the relationships between estimated ITR effects (on donation, deferral and utility outcomes, respectively) and b when the target is to maximise the utility score. Again, we observe very similar results using different methods ("similar" in terms of clinical meaningfulness), especially when the value of b is small. As b increases, assignment proportions shift from the more frequent to less frequent inter-donation intervals: less donors are allocated to the shortest inter-donation interval, and more donors are allocated to the longest one. Consequently, both the increase in the total units of blood collected (i.e. benefit) and the increase in the low Hb deferral rates (i.e. risk) become smaller. When $b = 1, 2, 3$, or 4, both the benefit and the risk increase, while in the extreme case where $b = 5$, there is an increase in the benefit and a decrease in the risk.

As a reference, we also calculate the ITR effects of three non-personalised rules on the utility outcome when the trade-off parameter b varies from 1 to 5 (Table 2.8) and compare with those of personalised rules presented in Table 2.7. Unlike the case when we analyse two outcomes separately where the "one-size-fits-all" rule seems to be sufficient for achieving a desirable outcome, when we target the utility outcome, personalised rules that tailor to each donor's capacity to donate lead to a higher gain in utility scores compared to non-personalised rules, and the advantage over non-personalised rules becomes more pronounced as b increases. For example, when $b = 5$, ITR effects on the utility score of the best non-personalised rule (assign all donors to the 10-week inter-donation interval) is 0.183, while ITR effects of personalised rules range from 0.485 to 0.648, depending on which method is used to estimate the optimal ITR.

Table 2.7: *Applications of the l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), and direct learning (D-learning) to the data from male donors in the INTERVAL trial assuming the target is to maximise the utility. The trade-off parameter b in the utility function varies from 1 to 5 at an increment of 1. Means and standard deviations (in parenthesis) of assignment proportions in % and empirical ITR effects on donation, deferral, and utility across 100 repetitions of 5-fold cross-validation are reported. ITR effects measure the difference in the average outcome between donors whose assigned inter-donation intervals in the trial are optimal (with respect to the method used to estimate the ITR) and those whose assigned inter-donation intervals are non-optimal. A larger ITR effect on donation/utility and a smaller ITR effect on deferral are more desirable.*

		Assignment Percentages			ITR Effects		
Trade-off Parameter	Method	12 weeks	10 weeks	8 weeks	Donation	Deferral	Utility
$b = 1$	l_1-PLS-HGL	0.9 (0.1)	1.2 (0.1)	97.9 (0.1)	1.309 (0.008)	0.024 (0.001)	1.064 (0.009)
	l_1-PLS-GL	0.3 (0.1)	2.4 (0.8)	97.2 (1.0)	1.289 (0.014)	0.025 (0.001)	1.040 (0.014)
	ACWL	0.0 (0.0)	0.0 (0.1)	100.0 (0.1)	1.314 (0.003)	0.027 (0.000)	1.055 (0.002)
	D-learning	0.7 (0.2)	1.2 (0.5)	98.1 (0.5)	1.309 (0.012)	0.025 (0.001)	1.058 (0.013)
$b = 2$	l_1-PLS-HGL	3.4 (0.1)	4.2 (0.3)	92.4 (0.4)	1.242 (0.016)	0.021 (0.001)	0.809 (0.020)
	l_1-PLS-GL	1.7 (0.4)	7.2 (1.6)	91.1 (2.0)	1.217 (0.027)	0.022 (0.002)	0.774 (0.024)
	ACWL	2.7 (0.8)	3.4 (1.7)	93.9 (1.3)	1.266 (0.019)	0.022 (0.001)	0.814 (0.016)
	D-learning	1.5 (0.4)	5.8 (1.1)	92.7 (1.0)	1.260 (0.022)	0.022 (0.001)	0.816 (0.023)
$b = 3$	l_1-PLS-HGL	8.6 (0.3)	11.9 (0.5)	79.5 (0.6)	1.091 (0.022)	0.011 (0.001)	0.689 (0.028)
	l_1-PLS-GL	4.8 (1.1)	15.0 (3.3)	80.2 (4.4)	1.069 (0.056)	0.017 (0.003)	0.569 (0.041)
	ACWL	9.3 (1.4)	8.2 (2.7)	82.5 (2.2)	1.100 (0.034)	0.014 (0.001)	0.627 (0.032)
	D-learning	3.8 (0.8)	17.5 (1.6)	78.6 (1.3)	1.067 (0.027)	0.016 (0.001)	0.607 (0.027)
$b = 4$	l_1-PLS-HGL	17.0 (0.4)	23.3 (0.5)	59.7 (0.5)	0.745 (0.023)	0.001 (0.001)	0.623 (0.030)
	l_1-PLS-GL	10.5 (2.3)	27.9 (3.2)	61.6 (5.2)	0.782 (0.070)	0.008 (0.004)	0.468 (0.081)
	ACWL	16.8 (1.9)	16.2 (3.5)	67.0 (3.1)	0.793 (0.055)	0.007 (0.002)	0.475 (0.033)
	D-learning	9.9 (1.6)	30.0 (1.7)	60.2 (0.7)	0.783 (0.027)	0.006 (0.001)	0.543 (0.039)
$b = 5$	l_1-PLS-HGL	26.4 (0.4)	33.4 (0.5)	40.3 (0.3)	0.410 (0.022)	-0.007 (0.001)	0.648 (0.031)
	l_1-PLS-GL	18.2 (3.2)	48.4 (6.2)	33.4 (3.4)	0.324 (0.059)	-0.004 (0.002)	0.485 (0.084)
	ACWL	30.1 (2.5)	22.6 (4.2)	47.3 (3.2)	0.422 (0.053)	-0.005 (0.002)	0.541 (0.046)
	D-learning	19.3 (1.6)	37.4 (1.3)	43.3 (0.7)	0.505 (0.031)	-0.004 (0.001)	0.622 (0.045)



(a) *ITR effects on donation* **(b)** *ITR effects on deferral* **(c)** *ITR effects on utility*

Fig. 2.2: *Plots of the means and 95% confidence intervals for ITR effects (across 100 repetitions of 5-fold cross-validation) of the optimal ITRs estimated using various methods as the trade-off parameter b in the utility function varies from 1 to 5 at an increment of 1. Optimal ITRs are estimated using the data from male donors in the INTERVAL trial assuming the target is to maximise the utility. Methods to estimate the optimal ITR include l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), and direct learning (D-learning). ITR effects on the (a) donation, (b) deferral, and (c) utility outcomes are presented. ITR effects measure the difference in the average outcome between donors whose assigned inter-donation intervals in the trial are optimal (with respect to the method used to estimate the ITR) and those whose assigned inter-donation intervals are non-optimal. A larger ITR effect on donation/utility and a smaller ITR effect on deferral are more desirable.*

Table 2.8: *ITR effects of three non-personalised rules on the utility outcome. The trade-off parameter b in the utility function varies from 1 to 5 at an increment of 1. ITR effects measure the difference in the average outcome between donors whose assigned inter-donation intervals in the trial are the same as the one specified in the non-personalised rule and those whose assigned inter-donation intervals are different from that specified in the non-personalised rule. A larger ITR effect on utility is more desirable.*

	ITR Effects on Utility				
Non-personalised Rule	$b=1$	$b=2$	$b=3$	$b=4$	$b=5$
Recommend all male donors to donate every 12 weeks	-1.308	-0.828	-0.618	-0.408	-0.199
Recommend all male donors to donate every 10 weeks	-0.025	0.027	0.079	0.131	0.183
Recommend all male donors to donate every 8 weeks	1.055	0.795	0.535	0.275	0.015

Variable selection by HGL and GL In addition to estimating the optimal ITR, l_1-PLS with group LASSO (GL) or hierarchical group LASSO (HGL) variable selection also picks important "treatment"-covariate interactions when building the prediction model for $E(Y|\mathbf{X}, A)$. Here, we investigate which "treatment"-covariate interactions are estimated as non-zero and regarded as important by HGL and GL in the prediction model. For demonstration, we focus on the situation where maximising the total units of blood collected by the blood service is our primary goal. The means (standard deviations) of the cross-validated mean squared errors (MSE) for the regression model with HGL and GL are 7.73 (0.03) and 7.76 (0.04), respectively. Table 2.9 summarises selection percentages (across 100 repetitions of 5-fold cross-validation) of different "treatment"-covariate interactions in the prediction model for donation when GL that does not impose strong hierarchy between main effects and interactions, or HGL that enforces such hierarchy is used for variable selection.

Table 2.9: *Selection percentages of treatment-covariate interactions in the prediction model for the donation outcome across 100 repetitions of 5-fold cross-validation when l_1-PLS is used to estimate the optimal ITR. Hierarchical group LASSO (HGL) enforces strong hierarchy between main effects and interactions, and group LASSO (GL) does not impose strong hierarchy between main effects and interactions.*

		Variable Selection	
Baseline Variables	Variable Type	HGL	GL
Age	continuous	63	21
Body mass index	continuous	33	38
SF-36v2 physical component score	continuous	47	11
SF-36v2 mental component score	continuous	100	20
Blood donations in the 2 years before trial enrolment	continuous	100	100
Haemoglobin level	continuous	100	100
White blood cell count	continuous	100	46
Red blood cell count	continuous	0	11
Mean corpuscular haemoglobin	continuous	100	96
Mean corpuscular volume	continuous	62	11
Platelet count	continuous	100	12
Ethnicity	categorical	47	97
Blood group	categorical	99	99
Iron prescription	categorical	1	74
Smoke ever	categorical	13	39
Smoke currently	categorical	98	99
Alcohol ever	categorical	85	32
Alcohol currently	categorical	100	100
New or returning donor status	categorical	100	32

We observe that some interactions between the randomised group (inter-donation interval) and baseline characteristics are selected almost all the time by both variable selection methods, such as blood donations in the 2 years before trial enrolment, baseline haemoglobin level, and blood group. We also notice that for some baseline covariates, selection percentages of their interaction with the randomised group differ substantially between the two variable selection approaches, even though the MSE of the prediction model, donor assignment proportions and ITR effects for l_1-PLS-HGL and l_1-PLS-GL (Table 2.6) are very similar. A possible explanation for an interaction being selected much more often by HGL than by GL (the case for most continuous covariates) is that the effect of the interaction itself is not strong enough and may be dominated by the main effect, but as has been noted by Lim and Hastie (2015), their proposed HGL method can still discover important interaction terms in this case due to the use of the "glinternet" penalty.

It is worth noting that the high selection rates of interactions between randomised groups and baseline covariates do not imply different recommendations on the inter-donation interval for different donors, and the finding that almost all male donors should donate every 8 weeks to maximise the total units of blood collected (Table 2.6) does not contradict the observation that many interaction effects are estimated to be non-zero (Table 2.9). This is because variable selection methods that we use to identify important interactions do not distinguish between quantitative and qualitative interactions, whereas only qualitative interactions can lead to different inter-donation interval recommendations for different subpopulations. To illustrate this point, we pick three categorical baseline characteristics (alcohol currently, smoke currently, and blood group) that are not reported in the prespecified subgroup analysis section of Di Angelantonio et al. (2017). We plot their relationships with the total units of blood collected by randomised groups in Figure 2.3. We note that we are not conducting any post-hoc subgroup analysis here, but rather an exploratory analysis through visualisation. It is clear from Figure 2.3 that even though the interactions between these three baseline characteristics and the randomised groups do exist and are selected almost 100% of times by both HGL and GL according to Table 2.9, these interactions are quantitative and do not alter the recommendation on the optimal inter-donation interval for donors with different covariate patterns, i.e. if the goal is to maximise the total units of blood collected by the blood service, all male donors should be encouraged to give blood every 8 weeks.

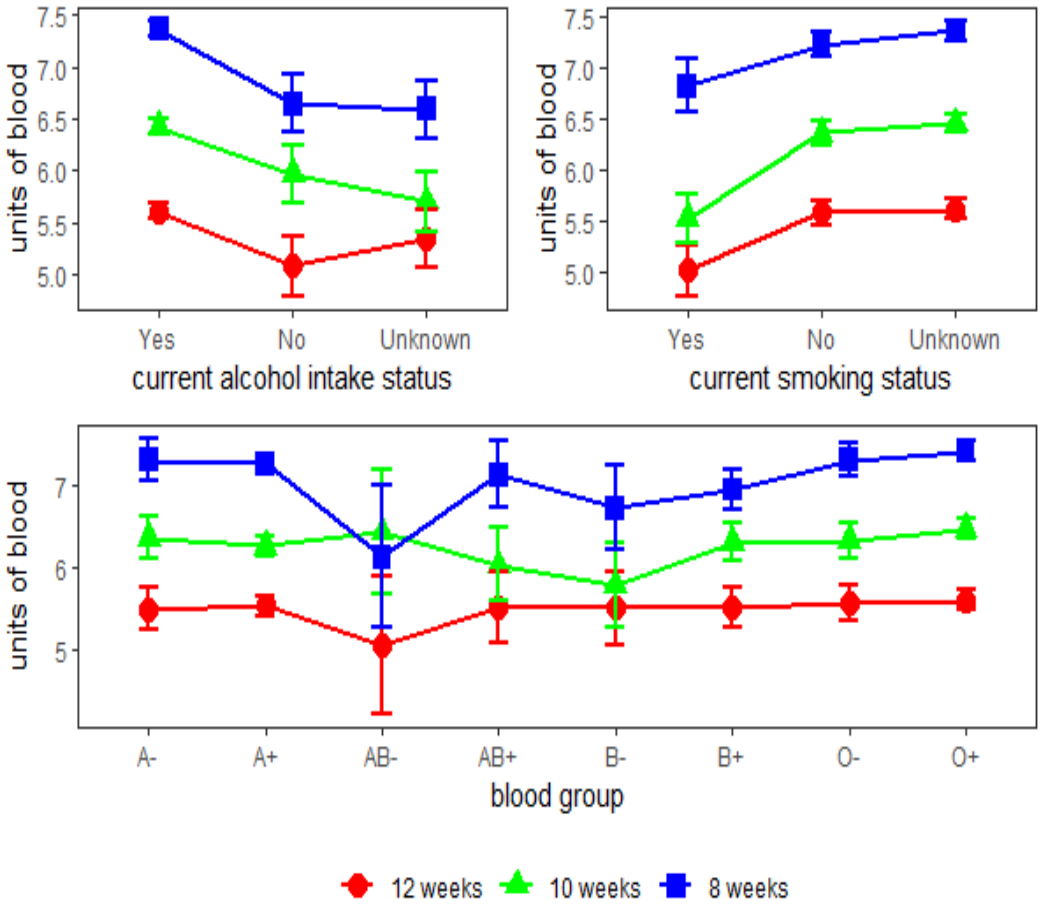


Fig. 2.3: *Visualisation of interactions between baseline characteristics and randomised groups: mean (95% confidence interval) units of blood collected during the 2-year trial period by randomised groups for male donors in the INTERVAL trial. Three baseline characteristics (current alcohol intake status, current smoking status, and blood group) are examined.*

2.5.4.2 Bootstrap-based results

In Tables 2.6 and 2.7, we present means and standard deviations of donor assignment proportions and empirical ITR effects across 100 repetitions of 5-fold cross-validation. The standard deviation estimates based on cross-validation reflect the repeatability of a method, i.e. how much variation would we expect in the answers obtained if we employ the method with cross-validation to the same dataset multiple times. In this section, we present additional analysis results based on 500 bootstrap samples (stratified by randomised groups) from male donors in the INTERVAL trial (Tables 2.10-2.13). Different from the cross-validation approach, the bootstrap approach attempts to capture the confidence we have in the size of the effects estimated by different methods, i.e. how much variation would we expect in the effect estimates if we were to repeat the study multiple times in the same population and calculate an effect estimate using the same method each time.

Table 2.10 presents the results for analysing two outcomes separately. Table 2.11 shows empirical ITR effects of three non-personalised rules on both the donation and the deferral outcomes. Table 2.12 summarises allocation proportions and ITR effects when the aim is to maximise the utility score with different values of the trade-off parameter b, and Table 2.13 presents the ITR effects of three non-personalised rules on the utility score when the trade-off parameter b varies from 1 to 5 at an increment of 1. All point estimates presented in Tables 2.10-2.13 are corresponding sample statistics computed from the original INTERVAL data, and standard deviation estimates (in parenthesis) are calculated based on 500 bootstrap samples (Efron and Tibshirani, 1993).

Results on assignment percentages and ITR effects indicate that different methods perform similarly. Not surprisingly, bootstrap-based standard deviation estimates that reflect the uncertainty of the observed dataset are much larger than cross-validation-based standard deviation estimates reported in Section 2.5.4.1.

Table 2.10: *Applications of the l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), and direct learning (D-learning) to the data from male donors in the INTERVAL trial. Sample estimates based on the INTERVAL data and standard deviation estimates based on 500 bootstrap samples (in parenthesis) of assignment proportions in % and empirical ITR effects on donation and deferral outcomes are reported. ITR effects measure the difference in the average outcome between donors whose assigned inter-donation intervals in the trial are optimal (with respect to the method used to estimate the ITR) and those whose assigned inter-donation intervals are non-optimal. A larger ITR effect on donation and a smaller ITR effect on deferral are more desirable. The first four and last four rows correspond to the target being maximising total units of blood collected by the blood service, and minimising the rate of low Hb deferrals, respectively.*

		Assignment Percentages			ITR Effects	
Target Outcome	Method	12 weeks	10 weeks	8 weeks	Donation	Deferral
Donation	l_1-PLS-HGL	0.1 (0.2)	0.2 (0.7)	99.7 (0.7)	1.313 (0.047)	0.026 (0.002)
	l_1-PLS-GL	0.1 (0.2)	1.9 (1.1)	98.1 (1.2)	1.326 (0.048)	0.025 (0.002)
	ACWL	0.0 (0.0)	0.0 (0.4)	100.0 (0.4)	1.315 (0.048)	0.027 (0.002)
	D-learning	0.5 (1.1)	0.7 (2.9)	98.8 (3.1)	1.310 (0.063)	0.026 (0.002)
Deferral	l_1-PLS-HGL	93.1 (2.6)	6.8 (2.6)	0.0 (0.2)	-1.191 (0.050)	-0.024 (0.002)
	l_1-PLS-GL	100.0 (6.1)	0.0 (5.7)	0.0 (0.5)	-1.248 (0.083)	-0.025 (0.002)
	ACWL	100.0 (3.1)	0.0 (3.1)	0.0 (0.2)	-1.248 (0.053)	-0.025 (0.002)
	D-learning	94.9 (4.0)	4.9 (4.1)	0.2 (0.8)	-1.197 (0.067)	-0.024 (0.002)

Table 2.11: *Sample estimates based on the INTERVAL data and standard deviation estimates based on 500 bootstrap samples (in parenthesis) of empirical ITR effects of three non-personalised rules on donation and deferral outcomes. ITR effects measure the difference in the average outcome between donors whose assigned inter-donation intervals in the trial are the same as the one specified in the non-personalised rule and those whose assigned inter-donation intervals are different from that specified in the non-personalised rule. A larger ITR effect on donation and a smaller ITR effect on deferral are more desirable.*

	ITR Effects	
Non-personalised Rule	Donation	Deferral
Recommend all male donors to donate every 12 weeks	-1.248 (0.040)	-0.025 (0.002)
Recommend all male donors to donate every 10 weeks	-0.077 (0.042)	-0.002 (0.002)
Recommend all male donors to donate every 8 weeks	1.315 (0.048)	0.027 (0.002)

Table 2.12: *Applications of the l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), and direct learning (D-learning) to the data from male donors in the INTERVAL trial assuming the target is to maximise the utility. The trade-off parameter b in the utility function varies from 1 to 5 at an increment of 1. Sample estimates based on the INTERVAL data and standard deviation estimates based on 500 bootstrap samples (in parenthesis) of assignment proportions in % and empirical ITR effects on donation, deferral, and utility are reported. ITR effects measure the difference in the average outcome between donors whose assigned inter-donation intervals in the trial are optimal (with respect to the method used to estimate the ITR) and those whose assigned inter-donation intervals are non-optimal. A larger ITR effect on donation/utility and a smaller ITR effect on deferral are more desirable.*

		Assignment Percentages			ITR Effects		
Trade-off Parameter	Method	12 weeks	10 weeks	8 weeks	Donation	Deferral	Utility
$b = 1$	l_1-PLS-HGL	0.9 (0.6)	1.0 (1.7)	98.1 (1.7)	1.331 (0.050)	0.022 (0.002)	1.089 (0.050)
	l_1-PLS-GL	0.5 (0.4)	3.8 (1.9)	95.7 (2.0)	1.320 (0.050)	0.023 (0.002)	1.079 (0.049)
	ACWL	0.0 (0.1)	0.0 (1.9)	100.0 (1.9)	1.315 (0.048)	0.027 (0.002)	1.055 (0.050)
	D-learning	0.7 (1.5)	2.6 (4.2)	96.7 (4.5)	1.323 (0.076)	0.024 (0.003)	1.079 (0.065)
$b = 2$	l_1-PLS-HGL	3.9 (1.3)	3.8 (3.3)	92.2 (3.1)	1.277 (0.061)	0.019 (0.002)	0.869 (0.055)
	l_1-PLS-GL	0.0 (1.0)	0.8 (3.6)	99.2 (3.8)	1.320 (0.061)	0.026 (0.003)	0.806 (0.055)
	ACWL	3.7 (2.9)	0.0 (5.9)	96.3 (5.0)	1.299 (0.061)	0.022 (0.003)	0.838 (0.054)
	D-learning	0.6 (2.3)	6.8 (5.6)	92.6 (5.9)	1.304 (0.092)	0.022 (0.003)	0.863 (0.064)
$b = 3$	l_1-PLS-HGL	9.8 (2.6)	11.4 (5.3)	78.9 (4.6)	1.122 (0.079)	0.009 (0.003)	0.767 (0.061)
	l_1-PLS-GL	6.6 (2.3)	19.0 (5.7)	74.4 (6.6)	1.104 (0.087)	0.011 (0.004)	0.744 (0.072)
	ACWL	14.8 (4.2)	12.4 (10.6)	72.9 (9.5)	1.009 (0.119)	0.007 (0.004)	0.731 (0.066)
	D-learning	6.2 (3.4)	13.7 (5.9)	80.1 (5.7)	1.087 (0.096)	0.014 (0.004)	0.659 (0.069)
$b = 4$	l_1-PLS-HGL	18.1 (3.7)	22.2 (6.7)	59.7 (5.2)	0.816 (0.092)	-0.002 (0.003)	0.765 (0.067)
	l_1-PLS-GL	14.1 (4.1)	30.7 (6.6)	55.2 (6.7)	0.810 (0.106)	-0.001 (0.004)	0.756 (0.101)
	ACWL	15.7 (7.5)	9.5 (14.1)	74.9 (12.0)	0.931 (0.168)	0.007 (0.006)	0.560 (0.085)
	D-learning	15.7 (4.4)	27.1 (5.9)	57.1 (5.2)	0.740 (0.100)	0.000 (0.004)	0.667 (0.075)
$b = 5$	l_1-PLS-HGL	27.2 (4.5)	32.3 (7.2)	40.5 (5.2)	0.453 (0.098)	-0.009 (0.003)	0.751 (0.073)
	l_1-PLS-GL	23.5 (5.8)	39.3 (8.8)	37.2 (5.6)	0.486 (0.104)	-0.010 (0.004)	0.810 (0.121)
	ACWL	30.3 (8.0)	16.2 (15.1)	53.5 (12.2)	0.522 (0.182)	-0.004 (0.005)	0.579 (0.097)
	D-learning	19.3 (4.7)	36.8 (6.3)	44.0 (5.0)	0.587 (0.102)	-0.006 (0.003)	0.766 (0.078)

Table 2.13: *Sample estimates based on the INTERVAL data and standard deviation estimates based on 500 bootstrap samples (in parenthesis) of empirical ITR effects of three non-personalised rules on the utility outcome. The trade-off parameter b in the utility function varies from 1 to 5 at an increment of 1. ITR effects measure the difference in the average outcome between donors whose assigned inter-donation intervals in the trial are the same as the one specified in the non-personalised rule and those whose assigned inter-donation intervals are different from that specified in the non-personalised rule. A larger ITR effect on utility is more desirable.*

	ITR Effects on Utility				
Non-personalised Rule	$b = 1$	$b = 2$	$b = 3$	$b = 4$	$b = 5$
Recommend all male donors to donate every 12 weeks	-1.038 (0.042)	-0.828 (0.045)	-0.618 (0.049)	-0.408 (0.055)	-0.199 (0.060)
Recommend all male donors to donate every 10 weeks	-0.025 (0.044)	0.027 (0.048)	0.079 (0.053)	0.131 (0.058)	0.183 (0.065)
Recommend all male donors to donate every 8 weeks	1.055 (0.050)	0.795 (0.054)	0.535 (0.059)	0.275 (0.066)	0.015 (0.074)

2.5.5 The Bayesian approach: BART

Following Logan et al. (2019), we first build the conditional mean outcome model using BART, and then we identify the optimal ITR based on posterior predictive distributions of the conditional mean under each randomised group (the resulting optimal ITR is referred to as the "BART ITR"). According to Section 2.3.4, the BART ITR is the one in which the recommended arm for each individual is given by maximising the subject-specific MCMC estimate (e.g. posterior mean) of the posterior predictive distribution of $E(Y|\mathbf{X} = \mathbf{x}, A = a)$. In the INTERVAL trial, the BART ITR is the rule that assigns each donor to the inter-donation interval that leads to the largest posterior mean of utilities/total units of blood collected, or the inter-donation interval that corresponds to the smallest posterior mean of low Hb deferral rates.

Target two outcomes separately Consistent with our findings using frequentist approaches, when we fit the BART model to the INTERVAL data with the target of maximising the total units of blood collected by the blood service, the BART ITR, $\widehat{\mathcal{D}^*}_{\text{BART}}(\mathbf{X})$, assigns all male donors to the most frequent inter-donation interval (8-week) with the posterior mean of the ITR effects on donation being 1.313 (95% credible interval: [1.237,1.392]). On the other hand, when the aim is to minimise low Hb deferral rates, $\widehat{\mathcal{D}^*}_{\text{BART}}(\mathbf{X})$ recommends 91.6% of male donors to donate every 12 weeks, leading to an ITR effect of -0.024 (posterior mean) on the deferral outcome with the 95% credible interval being $[-0.027,-0.022]$.

Target the utility outcome We also fit the BART model to the INTERVAL data with the utility score being the outcome of interest and obtain the posterior predictive distribution of the utility score under each inter-donation interval option. We vary the trade-off parameter b in the utility function from 1 to 5 at an increment of 1 to see how $\widehat{\mathcal{D}^*}_{\text{BART}}(\mathbf{X})$ changes with this parameter. For each b, we compare five donor assignment rules:

(i) all donors donate every 8 weeks;

(ii) all donors donate every 10 weeks;

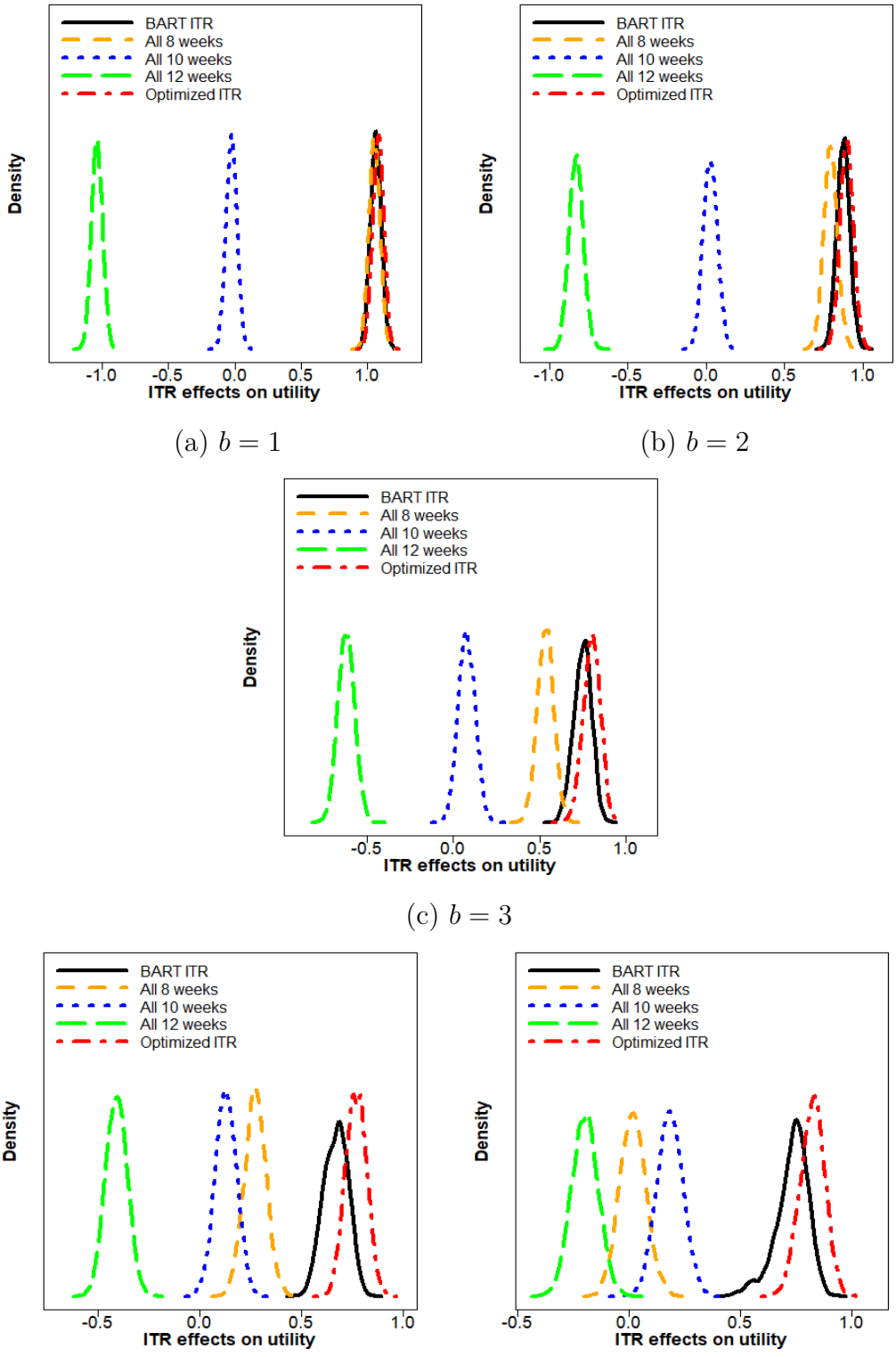
(iii) all donors donate every 12 weeks;

(iv) donors donate according to the BART ITR (each donor is assigned to the inter-donation interval associated with the maximum posterior mean of the utility score);

(v) donors donate according to the optimised ITR based on the BART estimation (an idealised scenario in which for each MCMC draw from the posterior distribution, each donor is assigned to the inter-donation interval associated with the maximum value of the utility score. This is non-achievable in practice, but we use this rule as a reference for the best-case scenario).

Posterior distributions of the ITR effects associated with those five donor assignment rules corresponding to five different trade-off parameters are plotted in Figure 2.4. In Table 2.14, we summarise donor allocation proportions based on the BART ITR, and we also report the posterior mean and the 95% equal tail credible interval of the ITR effect for each donor assignment strategy (we note that standard deviation estimates reported in Table 2.7 for non-Bayesian methods are calculated across 100 repetitions of 5-fold cross-validation, while in Table 2.14, we quantify the uncertainty of the estimates directly based on posterior samples from BART).

We observe that when b is small (e.g. $b = 1$ or 2), the best "one-size-fits-all" strategy is to recommend all donors to donate every 8 weeks. In these cases, both the BART ITR and the optimised ITR are close to the "all 8 weeks" rule in that the ITR effect distribution of the BART ITR and the optimised ITR almost overlaps with that of the "all 8 weeks" rule. When b takes values from 3 to 5, the BART ITR is better than the best non-personalised assignment rule ("all 8 weeks" for $b = 3, 4$, and "all 10 weeks" for $b = 5$) with a high probability, and the advantage of the BART ITR over non-personalised rules becomes more pronounced as b gets larger. Even though the BART ITR is slightly inferior to the optimised ITR (which is generally not achievable in practice) as expected, the differences are minimal.



(a) $b = 1$ (b) $b = 2$

(c) $b = 3$

(d) $b = 4$ (e) $b = 5$

Fig. 2.4: *Density plots of ITR effects on utility for five donor assignment rules: recommend all male donors to (i) donate every 8 weeks, (ii) donate every 10 weeks, (iii) donate every 12 weeks, (iv) donate according to the BART ITR, and (v) donate according to the optimised ITR (non-achievable in practice). The trade-off parameter b in the utility function varies from 1 to 5 at an increment of 1. A larger ITR effect on utility is more desirable.*

Table 2.14: *Applications of Bayesian additive regression trees (BART) to the data from male donors in the INTERVAL trial assuming the target is to maximise the utility score. The trade-off parameter b in the utility function varies from 1 to 5 at an increment of 1. Assignment proportions of the BART ITR in % and the posterior mean [95% equal tail credible interval] of the ITR effect on the utility outcome for five donor assignment rules are reported. Assignment rules include: recommend all male donors to (i) donate every 8 weeks, (ii) donate every 10 weeks, (iii) donate every 12 weeks, (iv) donate according to the BART ITR, and (v) donate according to the optimised ITR (non-achievable in practice). A larger ITR effect on utility is more desirable.*

		Trade-off Parameter				
Criteria	Assignment Rule	$b = 1$	$b = 2$	$b = 3$	$b = 4$	$b = 5$
BART ITR Assignment Percentages	12 weeks	0.0	3.6	10.7	17.5	26.7
	10 weeks	3.2	9.4	18.8	25.5	28.4
	8 weeks	96.8	87.1	70.5	57.0	44.9
ITR Effects on Utility Posterior Mean [95% Credible Interval]	All 12 weeks	-1.037 [-1.118,-0.956]	-0.827 [-0.915,-0.739]	-0.618 [-0.712,-0.524]	-0.408 [-0.511,-0.305]	-0.199 [-0.317,-0.082]
	All 10 weeks	-0.025 [-0.106,0.055]	0.027 [-0.060,0.114]	0.079 [-0.017,0.174]	0.131 [0.027,0.232]	0.183 [0.068,0.297]
	All 8 weeks	1.054 [0.973,1.134]	0.794 [0.708,0.883]	0.535 [0.437,0.628]	0.275 [0.169,0.379]	0.016 [-0.101,0.135]
	BART ITR	1.064 [0.983,1.144]	0.876 [0.793,0.959]	0.750 [0.643,0.843]	0.671 [0.553,0.780]	0.732 [0.548,0.852]
	Optimised ITR	1.082 [1.000,1.159]	0.898 [0.814,0.981]	0.802 [0.701,0.893]	0.770 [0.671,0.870]	0.823 [0.710,0.928]

2.5.6 Measures of agreement between methods

Results presented in Sections 2.5.4 and 2.5.5 suggest that donor assignment proportions and empirical ITR effects are fairly similar across different methods for each target outcome at the population level. We are also interested in investigating for a given male donor, to what extent the five methods (l_1-PLS-HGL, l_1-PLS-GL, ACWL, D-learning, and BART) "agree" on his optimal inter-donation interval (at the individual level). Given the differences in the decision boundary types assumed by each method, we expect that the forms of estimated regimes based on different methods may be very different and the set of baseline characteristics included in the optimal decision rule may also vary across methods. However, the optimal inter-donation interval estimated by different approaches for a given donor can still "overlap" despite the possible heterogeneity in the forms of estimated regimes (Zhang and Zhang, 2018). To evaluate the degree of agreement between different methods in terms of the recommended optimal inter-donation interval for each male donor in the INTERVAL trial, we present two inter-method agreement measures (overall and pairwise).

2.5.6.1 The first measure of agreement: percent agreement

The first one is a simple "agreement proportion" measure that assesses the amount of "overlap" of the optimal decisions made according to each method (l_1-PLS-HGL, l_1-PLS-GL, ACWL, D-learning, and BART). We calculate the "observed proportion of agreement" (percentage of male donors who receive the same recommendation on his optimal inter-donation interval by all five methods):

$$\frac{\sum_{i=1}^{n} I\Big[\max\big\{\sum_{j=1}^{5} I(\widehat{\mathcal{D}}_j^*(\mathbf{X}_i) = 8 \text{ weeks}), \sum_{j=1}^{5} I(\widehat{\mathcal{D}}_j^*(\mathbf{X}_i) = 10 \text{ weeks}), \sum_{j=1}^{5} I(\widehat{\mathcal{D}}_j^*(\mathbf{X}_i) = 12 \text{ weeks})\big\} = 5\Big]}{n},$$

where $\max(\bullet)$ is the maximum function, $I(\bullet)$ is the indicator function, i is the donor index, $i = 1, \ldots, n$, j is the method index, $j = 1, \ldots, 5$, and $\widehat{\mathcal{D}}_j^*(\mathbf{X}_i)$ denotes the optimal inter-donation interval estimated by method j for the i^{th} donor.

In the case where recommendations are not consistent across all five methods, but three or four out of five methods "agree", the majority voting rule can be used to determine the optimal personalised inter-donation interval. Therefore, we also calculate the percentage of male donors for whom at least four methods "agree" on his optimal inter-donation interval:

$$\frac{\sum_{i=1}^{n} I\Big[\max\big\{\sum_{j=1}^{5} I(\widehat{\mathcal{D}}_j^*(\mathbf{X}_i) = 8 \text{ weeks}), \sum_{j=1}^{5} I(\widehat{\mathcal{D}}_j^*(\mathbf{X}_i) = 10 \text{ weeks}), \sum_{j=1}^{5} I(\widehat{\mathcal{D}}_j^*(\mathbf{X}_i) = 12 \text{ weeks})\big\} \geq 4\Big]}{n},$$

and the percentage of male donors for whom at least three methods "agree" on his optimal inter-donation interval:

$$\frac{\sum_{i=1}^{n} I\Big[\max\big\{\sum_{j=1}^{5} I(\widehat{\mathcal{D}_j^*}(\mathbf{X}_i) = 8 \text{ weeks}), \sum_{j=1}^{5} I(\widehat{\mathcal{D}_j^*}(\mathbf{X}_i) = 10 \text{ weeks}), \sum_{j=1}^{5} I(\widehat{\mathcal{D}_j^*}(\mathbf{X}_i) = 12 \text{ weeks})\big\} \geq 3\Big]}{n}.$$

Results are presented in Table 2.15. We observe that when the target outcome is donation, deferral, or utility with $b = 1$ or 2, the levels of agreement between five methods are high ($> 80\%$). As the trade-off parameter b in the utility function increases, the levels of agreement decrease. For the most stringent agreement criterion (regarded as "agree" only if all five methods lead to the same recommendation), the percentage of donors getting the same recommendation from all five methods is only 33.5% for $b = 5$. On the other hand, according to the least stringent criterion of agreement (regarded as "agree" if at least three methods agree), we can always decide the optimal inter-donation interval for at least 90% of donors using the majority voting rule regardless of the target outcome.

Table 2.15: *Percentages of male donors who receive the same recommendation on the optimal inter-donation interval by all five methods, by at least four methods, or by at least three methods when the target outcome is donation, deferral, and utility, respectively. Methods to estimate the optimal personalised inter-donation interval include the l_1-penalised least squares with hierarchical group LASSO variable selection, l_1-penalised least squares with group LASSO variable selection, adaptive contrast weighted learning, direct learning, and Bayesian additive regression trees.*

Agreement Type	Donation	Deferral	Utility				
			$b = 1$	$b = 2$	$b = 3$	$b = 4$	$b = 5$
All five methods agree	97.5	86.9	92.2	82.6	59.0	45.2	33.5
At least four methods agree	99.1	94.4	96.9	92.5	79.0	71.2	64.3
At least three methods agree	100.0	100.0	99.5	97.8	96.3	95.5	94.4

2.5.6.2 The second measure of agreement: pairwise B statistics

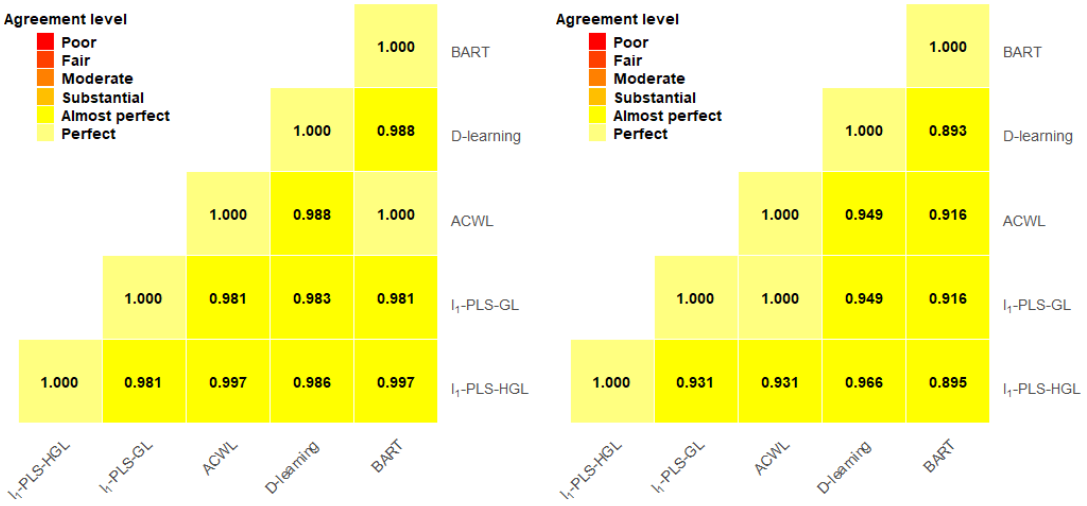
One limitation of the "percent agreement" measure is that this type of statistics does not account and correct for the "agreement by chance". Chance-corrected measures have been proposed for assessing inter-rater agreement, for example, the Cohen's kappa for two raters or Fleiss' kappa for more than two raters (Cohen, 1960; Fleiss, 1971). However, we note that kappa is not particularly suitable in the current setting because the performance of kappa depends on marginal distributions, while in our case, marginal distributions are highly symmetrically-imbalanced, especially when the target outcome is donation, deferral, or utility

with small b. This phenomenon has been well-documented in the literature and is commonly referred to as the "high agreement but low kappa paradox" (Feinstein and Cicchetti, 1990; Gwet, 2008; Shankar and Bangdiwala, 2014). Some improved measures that still account for chance agreement but address the kappa paradox have been proposed. For example, the Bangdiwala's B statistics proposed by Bangdiwala (1985) has been shown to be robust to different marginal distributions (Shankar and Bangdiwala, 2008, 2014). However, B statistics has not been extended to handle the case with more than two raters, and thus we are not able to calculate the overall B statistics that measures the agreement of all five methods. Instead, we report the pairwise B statistics and use the guidelines suggested by Munoz and Bangdiwala (1997) for the interpretation of B statistics (Table 2.16).

Table 2.16: *Guidelines for the interpretation of B statistics (Munoz and Bangdiwala, 1997).*

Range of B statistics	Agreement Level
[0,0.09)	Poor agreement
[0.09,0.25)	Fair agreement
[0.25,0.49)	Moderate agreement
[0.49,0.81)	Substantial agreement
[0.81,1.00)	Almost perfect agreement
1.00	Perfect agreement

Figure 2.5 presents the pairwise B statistics among 5 methods when the donation and deferral outcomes are analysed separately, and Figure 2.6 shows the corresponding results when the utility score is the target outcome. Different colours indicate the extent of agreement according to pairwise B statistics with yellow or light yellow representing the situations where the agreement is "almost perfect" or "perfect". We observe that all pairwise B statistics suggest at least "almost perfect" agreements when the target outcome is donation, deferral, or utility with $b = 1$ or 2. As b gets larger, the degree of pairwise agreements decreases. For $b = 3$ or 4, pairwise agreements never go below "substantial", whereas when the target outcome is the utility score with $b = 5$, several pairwise B statistics indicate only "moderate" agreement. In general, pairwise B statistics between l_1-PLS-HGL and the other methods seem to be slightly higher than the rest of the pairwise B statistics. Even though l_1-PLS-HGL and l_1-PLS-GL are built under the same framework and the only difference between these two methods is the variable selection approach used when constructing the outcome model, the

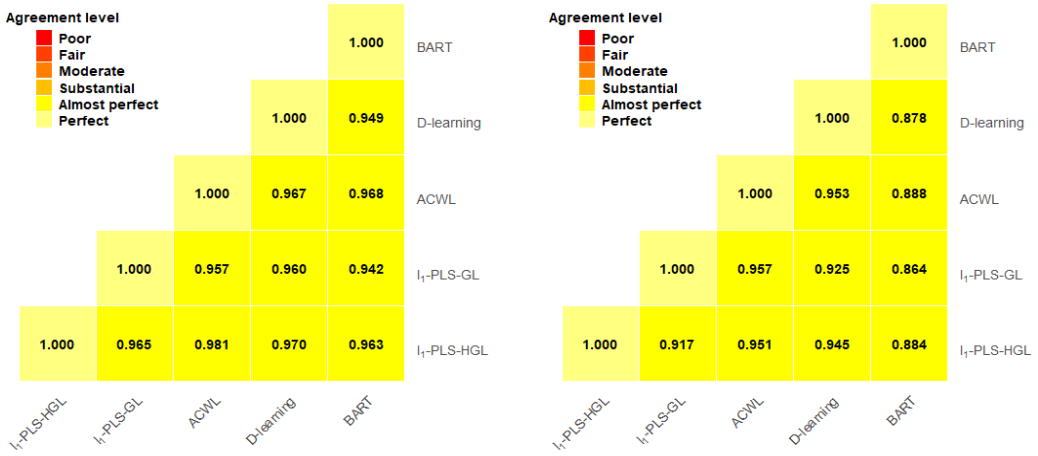


(a) *Pairwise B statistics (donation)* **(b)** *Pairwise B statistics (deferral)*

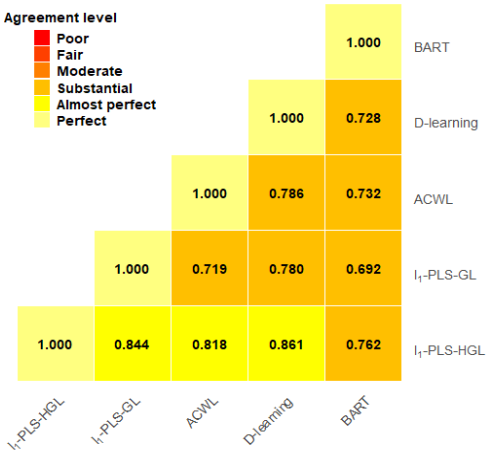
Fig. 2.5: *Pairwise B statistics that measure the agreement of the optimal personalised donation strategies estimated using five different methods: l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), direct learning (D-learning), and Bayesian additive regression trees (BART). Decision rules are obtained with the aim of (a) maximising the units of blood collected by the blood service, and (b) minimising the low Hb deferral rates using the data from male donors in the INTERVAL trial. Different colours indicate different levels of agreement.*

levels of agreement between these two methods do not seem to be consistently high. In cases where estimated optimal ITRs are far from the "one-size-fits-all" regime (e.g. when the target outcome is the utility score with $b = 3$, 4, or 5), pairwise agreements between BART and l_1-PLS-HGL are higher than those between BART and ACWL. This may be due to the fact that both l_1-PLS-HGL and BART use a two-stage procedure to estimate the optimal ITR, where in the first stage, the outcome model is constructed, and in the second stage, the optimal ITR is derived as the one that optimises the expected outcome. On the other hand, even though both ACWL and BART are tree-based methods, BART uses "additive regression trees" to model the dependency structure between the response and covariates, whereas ACWL uses "classification trees" to solve the weighted classification problem and estimate the decision rule.

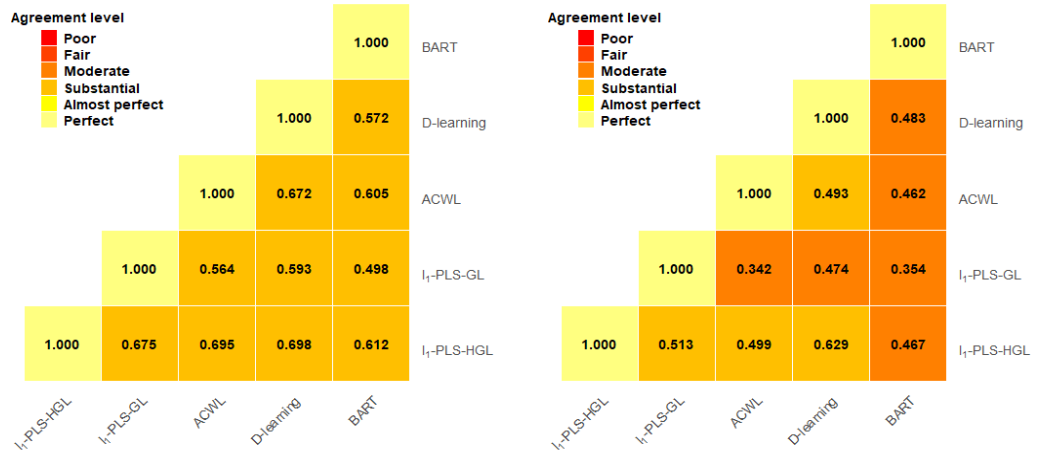
In summary, both agreement measures (percent agreement and pairwise B statistics) suggest that in most cases, ITRs estimated by different methods "agree" to a large degree.



(a) *Pairwise B statistics (utility: $b = 1$)* (b) *Pairwise B statistics (utility: $b = 2$)*



(c) *Pairwise B statistics (utility: $b = 3$)*



(d) *Pairwise B statistics (utility: $b = 4$)* (e) *Pairwise B statistics (utility: $b = 5$)*

Fig. 2.6: *Pairwise B statistics that measure the agreement of the optimal personalised donation strategies estimated using five different methods: l_1-penalised least squares with hierarchical group LASSO variable selection (l_1-PLS-HGL), l_1-penalised least squares with group LASSO variable selection (l_1-PLS-GL), adaptive contrast weighted learning (ACWL), direct learning (D-learning), and Bayesian additive regression trees (BART). Decision rules are obtained with the aim of maximising the utility score using the data from male donors in the INTERVAL trial. The trade-off parameter b in the utility function varies from 1 to 5 at an increment of 1. Different colours indicate different levels of agreement.*

2.6 Discussion

2.6.1 Summary

Most statistical methods for estimating the optimal individualised treatment rule (ITR) are restricted to binary treatment comparisons. However, clinical trials with multiple treatment arms are common in practice. In this chapter, we review several recent approaches that can be used to estimate the optimal ITR in large-scale clinical trials with more than two treatment options. Methods considered include: the l_1-penalised least squares with hierarchical group LASSO or group LASSO variable selection, the adaptive contrast weighted learning method, the direct learning method, and a Bayesian approach that builds on Bayesian additive regression trees.

We conduct simulation studies under various scenarios to evaluate the performance of these methods in clinical trials with large sample sizes. Our simulation results suggest that the BART multi-arm ITR estimation method has better or similar performance compared to other methods across different settings with different types of interaction terms despite that this method is indirect (i.e. it focuses on the construction of the prediction model rather than the direct maximisation of the value function). This may be attributed to the nonparametric nature of BART, which to a great extent alleviates the general concern about indirect ITR estimation methods that "regression model is likely to be misspecified and thus indirect regression-based ITR estimation methods may lead to suboptimal ITRs". We also observe from simulation studies that when the sample size is large (as in the INTERVAL trial), all methods perform equally well under scenarios where baseline characteristics interact with treatment arms only quantitatively (and thus the true optimal treatment is the same for all subjects) in the sense that they all accurately identify the true universal decision rules (assign all to the marginally best one), regardless of the type of underlying quantitative interactions being linear, nonlinear, or tree-type. We note that the conclusions drawn from simulation studies in this chapter only apply to the clinical trial setting, which is the case for our motivating example and the focus of this thesis. For observational studies, we may get different comparative conclusions across methods, but this is beyond the scope of our discussion.

We further illustrate the use of these methods by applying them to the data from male donors in the INTERVAL trial. We conclude that there is no "all-time

best performer" and results are fairly consistent across different approaches in terms of clinical meaningfulness.

When the target is to maximise the total units of blood collected by the blood service, or to minimise the low Hb deferral rates (i.e. the benefit outcome and the risk outcome are not considered in conjunction with each other), all methods detect almost no qualitative heterogeneity of the "inter-donation interval effects". More specifically, if we focus on maximising the total units of blood collected, then the optimal ITR tends to choose shorter inter-donation intervals that are associated with increased benefits at the cost of higher risks of deferrals for low Hb and almost all donors are assigned to the highest frequency of donation. On the other hand, if our aim is to minimise the low Hb deferral rates, then the optimal ITR picks longer inter-donation intervals and almost everyone is assigned to the lowest frequency of donation. These results are not surprising and support the trial's primary findings that interactions between baseline characteristics and the inter-donation interval are not qualitative, but rather quantitative (Di Angelantonio et al., 2017). As an aside, in general, failure to detect qualitative interactions between baseline characteristics and the "treatment" assignment among trial participants does not preclude the existence of such qualitative interactions over the entire eligible population due to restrictive entry criteria of clinical trials. However, this is not of great concern to us in the INTERVAL trial, and we will discuss this point in more detail later in Chapter 5. It is also worth noting that even though results based on the analysis of the INTERVAL trial data indicate that almost all male donors can donate every 8 weeks if the aim is to maximise the total units of blood collected by the blood service, there may be challenges associated with the long-term implementation of the "all 8 weeks" donation strategy. For example, during the 2-year period of the INTERVAL trial, more than 70% of male participants' assigned visits were achieved within 2 weeks of the inter-donation interval allocated (Di Angelantonio et al., 2017). However, it is unclear whether donors will come back every 8 weeks beyond a 2-year period, since we would expect that the general blood donor population may be less motivated and less willing to come back every 8 weeks in the long run due to various reasons.

Maximising the benefit and minimising the risk are two competing goals. While the optimal decision for each goal is obvious, the two decisions may be very different. To deal with this, we create a utility score that balances two outcomes and derive the optimal ITR with the goal of maximising the utility score. Investigation of the utility outcome suggests some heterogeneity in the optimal

inter-donation interval across donors with different baseline characteristics, and such heterogeneity becomes larger as the trade-off parameter b in the utility function (the equivalent benefit loss for one unit increase in risk) gets larger. We note that the optimal donor assignment strategy is highly dependent on b. We examine different values for b within a reasonable range and estimate the optimal ITR under each b. Ideally, b should be specified based on clinicians' domain knowledge. However, there is a lack of accurate information on b in most cases because the right balance between the benefit and the risk is usually not obvious and can be different for different subgroups (Liu et al., 2017).

2.6.2 Model extensions

When deriving the optimal ITR, we only use baseline measurements that are routinely collected at the regular donation session. In the future, we may include additional blood-based biomarkers (e.g. those related to iron stores such as ferritin, transferrin, and hepcidin) and/or information on self-reported symptoms (e.g. fatigue and restless legs syndrome) to estimate the optimal donation strategy. Extra expenses would be incurred if we would like to collect such data, and failure to account for the additional costs can lead to suboptimal decisions from the cost-effectiveness perspective. We will investigate how much additional information on biomarkers/symptoms adds to the reinforcement of decision-making (compared to the donation strategy estimated solely based on routinely collected data) and evaluate whether it is worthwhile for the blood service to collect symptom-related information or measure additional biomarkers at the regular donation session.

We note that in this chapter, we treat the inter-donation interval as a nominal variable with 3 categories. However, there is a natural ordering of the 12-, 10-, and 8-week inter-donation intervals and these three options of inter-donation interval can be considered as three levels of an ordinal "treatment". We would expect to have some information loss by treating an ordinal variable as nominal. Thus, it would be of great interest to explore ways to incorporate information on the ordinality into decision-making. We will investigate this in the following chapter.

Chapter 3

Sequential re-estimation learning of optimal individualised treatment rules among ordinal treatments

3.1 Introduction

In Chapter 2, we discuss some recently-proposed approaches that can be used to estimate the optimal individualised treatment rule (ITR) in clinical trials with more than two treatments. However, those multi-arm ITR estimation methods assume that multiple treatment options are nominal (i.e. have no intrinsic ordering), while in the INTERVAL trial, the three options of the inter-donation interval exhibit a natural order and can be viewed as three levels of an ordinal "treatment". In the blood donation context, treating three inter-donation interval options as ordinal instead of nominal can be particularly beneficial for identifying the optimal donation strategy. For example, for a female donor whose true optimal inter-donation interval is 16-week, incorrectly allocating her to the 12-week inter-donation interval might lead to more severe consequences on donor health than to the 14-week one which is closer to the true optimal.

To our knowledge, there is little research on estimation of the optimal ITR among ordinal treatments. Extensions of the binary methods to the ordinal setting is nontrivial. For example, dichotomising ordinal treatments using a prespecified cutoff and then applying methods developed for binary settings (as has been done in some literature on estimating the causal effects of ordinal exposures) makes the identification of an optimal ordinal treatment level impossible (Lopez and Gutman, 2017). On the other hand, when methods developed for nominal treatments are applied to ordinal treatments, it is highly likely that we will get suboptimal decisions, since one may expect to lose some useful information on treatment orderings that can potentially help improve the prediction performance of the model.

Motivated by the aforementioned methodological gap, in this chapter, we propose a sequential re-estimation (SR) learning approach to identify the optimal ITR among ordinal treatments for both linear and nonlinear decision rules. SR learning exploits and effectively incorporates information on the ordinality of treatment arms and thus avoids unnecessary pairwise comparisons. Specifically, by taking advantage of treatment orderings, we first decompose the optimal ITR estimation problem into a sequence of binary treatment comparison subproblems, including "sequential" ones that determine whether a more "intensive" treatment should be given and "re-estimation" ones that compare two "consecutive" treatment categories. Existing methods for estimating the optimal ITR in the two-arm case can subsequently be applied to solve each binary subproblem. In particular, we employ the augmented outcome weighted learning (AOL) method proposed by Liu et al. (2018) whereby optimal binary decision rules can be estimated under the weighted classification framework. We then ensemble multiple binary decisions obtained from binary classifiers and derive the optimal ITR among ordinal treatments based on a decision tree.

Clinical studies typically collect a large amount of patient information, but some of them may be irrelevant for making treatment decisions and it is usually challenging to acquire *a priori* knowledge on which patient characteristics are truly helpful. Inclusion of unimportant covariates when estimating the optimal ITR may lead to poor model performance and excessively complicated decision rules (Gunter et al., 2011; Song et al., 2015). Therefore, variable selection is vital for deriving the optimal ITR in order to remove covariates that are unnecessary and reduce the complexity of treatment decision rules. In light of this, we further propose variable selection methods for linear and nonlinear decision rules,

respectively, under the SR learning framework to improve the performance of SR learning in the presence of noise covariates.

The rest of this chapter is organised as follows. In Section 3.2, we introduce the statistical framework and the main idea of SR learning for estimating the optimal ITR in the ordinal treatment setting. We also develop variable selection techniques under the proposed framework. In Section 3.3, we conduct extensive simulation studies to evaluate the finite sample performance of SR learning. In Section 3.4, we illustrate our proposed method by applying it to the data from a "much-in-demand but vulnerable" donor population in the INTERVAL trial. This chapter concludes with a discussion in Section 3.5.

3.2 Methodology

3.2.1 Notations and statistical frameworks

The notations used and the standard causal assumptions made in this chapter are almost the same as those in Chapter 2, except that now we assume that treatments are ordered. We reiterate the notations and assumptions in the following for clarity. We assume that the data are collected from a clinical trial with n subjects and $K \geq 3$ ordinal treatments (the definition of ordinal treatments will follow shortly). Let $A \in \mathcal{A} = \{1, \ldots, K\}$ denote the treatment assignment, $Y \in \mathbb{R}$ be the target outcome, and $\mathbf{X} = (X_1, \ldots, X_p)^\mathsf{T} \in \mathcal{X}$ be a p-dimensional covariate (feature) vector. We observe $(Y_i, \mathbf{X}_i, A_i)$, for $i = 1, \ldots, n$, which are independent and identically distributed across i. Given the natural ordering of treatments $1, \ldots, K$, the reference arm can be either treatment 1 or K (1 is the least "intensive" treatment option and K is the most "intensive" one). Without loss of generality, we assume that treatment 1 is the reference arm and a larger Y is more desirable in the following discussion. An individualised treatment rule (ITR), denoted by $\mathcal{D}$, is a map from the feature space, $\mathcal{X}$, to the domain of treatment assignment, $\mathcal{A}$. We make the assumptions of "consistency", "no unmeasured confounders (NUC)", and "positivity" as in Chapter 2. The "value function" (Qian and Murphy, 2011) associated with the treatment rule $\mathcal{D}$ is given as follows:

$$V(\mathcal{D}) = E\left[\frac{I\{A = \mathcal{D}(\mathbf{X})\}}{P(A|\,\mathbf{X})}Y\right], \tag{3.1}$$

where $E(\cdot)$ is the expectation and $I(\cdot)$ is the indicator function. $V(\mathcal{D})$ can be interpreted as the expected outcome had all subjects in the given population

followed the rule $\mathcal{D}$, and we aim to find the optimal ITR, $\mathcal{D}^*$, that maximises $V(\mathcal{D})$, i.e.

$$\mathcal{D}^* = \arg\max_{\mathcal{D}} E\left[\frac{I\{A = \mathcal{D}(\mathbf{X})\}}{P(A|\,\mathbf{X})}Y\right]. \tag{3.2}$$

After introducing all these notations, we can now define "ordinal" treatment in a more rigorous way: given two nonoptimal realisations of the ordinal treatment A, denoted by a_1 and a_2, if $\text{sign}(a_1 - \mathcal{D}^*(\mathbf{x})) = \text{sign}(a_2 - \mathcal{D}^*(\mathbf{x}))$, and $|a_1 - \mathcal{D}^*(\mathbf{x})| > |a_2 - \mathcal{D}^*(\mathbf{x})|$, then we have $E(Y|\,\mathbf{X} = \mathbf{x}, A = a_1) < E(Y|\,\mathbf{X} = \mathbf{x}, A = a_2)$. This implies that for two treatments that are on the same side of the true optimal, the treatment that is further away from the true optimal one will lead to a worse outcome (since we assume a larger Y is preferable, worse outcome means a smaller Y) than the treatment that is closer to the true optimal.

3.2.2 Sequential re-estimation (SR) learning for ordinal treatments

In this section, we propose a method called "sequential re-estimation (SR) learning" that takes advantage of the ordering information on treatment arms and estimates the optimal ITR among ordinal treatments. Specifically, we first decompose the problem of estimating the optimal ITR among ordinal treatments into multiple subproblems of binary treatment comparisons (this stage is referred to as the "learning" stage and will be discussed in Section 3.2.2.1), which subsequently can be solved using existing methods developed for the binary treatment setting (details on methods for solving these binary subproblems will be discussed later in Section 3.2.3). Decisions from multiple binary subproblems are then aggregated for ordinal prediction (this stage is referred to as the "ensembling" stage and will be discussed in Section 3.2.2.2).

3.2.2.1 Learning

The model learning stage consists of $K-1$ "sequential" steps and $K-2$ "re-estimation" steps.

Sequential step (S-step) Each sequential step trains a binary classifier S_k that compares treatment $\{k\}$ with $\{k+1, \ldots, K\}$, $k = 1, \ldots, K-1$. Intuitively, S_k determines whether or not treatment options that are more "intensive" than treatment k can lead to more desirable outcomes than k. All subjects are included in the training model for S_1. On the other hand, following Zhou et al. (2018),

for $k = 2, \ldots, K-1$, only individuals whose observed treatments A_i do not belong to $\{1, \ldots, k-1\}$ and optimal treatments are not estimated as j in the j^{th} sequential step for all $j < k$, i.e. those who satisfy $I\Big(A_i \notin \{1, \ldots, k-1\}, \widehat{S}_1(\mathbf{X}_i) \neq 1, \ldots, \widehat{S}_{k-1}(\mathbf{X}_i) \neq k-1\Big) = 1$, are included in the training model for S_k.

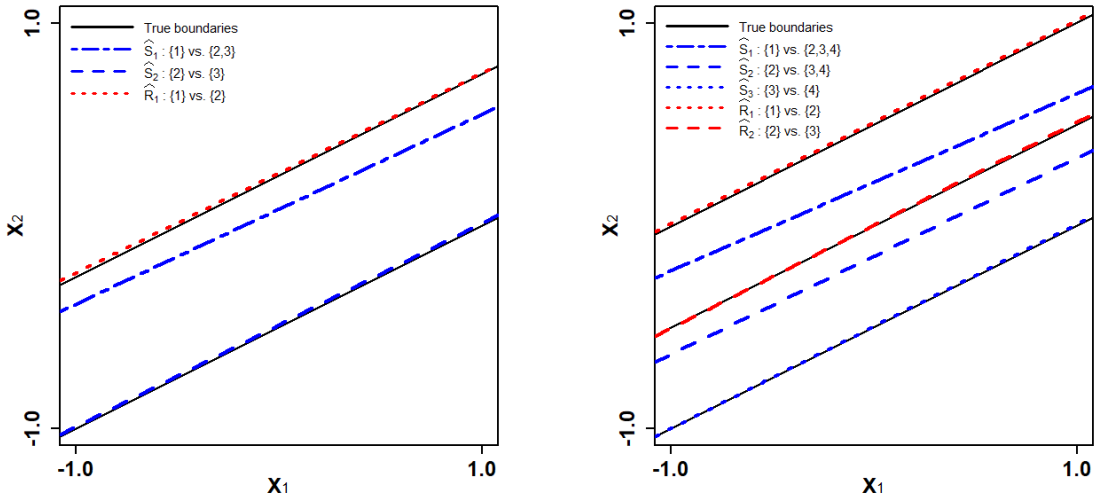
Re-estimation step (R-step) Each re-estimation step trains a binary classifier R_k that compares treatment $\{k\}$ with $\{k+1\}$, $k = 1, \ldots, K-2$, using the data from individuals whose observed treatments A_i are either k or $k+1$ and optimal treatments are estimated as k in the k^{th} sequential step, i.e. those satisfying $I\Big(A_i \in \{k, k+1\}, \widehat{S}_k(\mathbf{X}_i) = k\Big) = 1$.

We note that the number of eligible subjects who are included in training R-step classifiers is in general smaller than that of eligible subjects who are included in training S-step classifiers. For example, information on all n subjects are used when training the first sequential classifier, S_1, which determines whether or not a patient should receive a more "intensive" treatment than the reference arm (treatment 1), whereas when training the first re-estimation classifier, R_1, between treatments 1 and 2, we only use the data from a subpopulation of subjects whose predicted optimal treatments based on S_1 are 1 and observed treatments are either 1 or 2. This is closely aligned with how clinical decisions are made in practice: the decision on whether or not a patient should receive a treatment that is more "intensive" than the control/current clinical practice is made before deciding on the "exact" treatment option to be given. The proposed SR learning trains the S-step classifiers with a larger number of subjects and thus clinicians can be more confident when deciding whether or not a more "intensive" treatment should be administered.

The purpose of training $K-2$ R-step classifiers in addition to S-step classifiers is to estimate decision boundaries between two "consecutive" treatment categories k and $k+1$ more accurately by using the data from a more "refined" population. Moreover, it allows patients who have been assigned to receive "conservative" treatments in S-steps to be reconsidered for whether a treatment "step-up" may benefit them.

We illustrate the advantage of adding R-steps in Figure 3.1 ((a) for $K = 3$ and (b) for $K = 4$) by two simulated examples where true decision boundaries between "consecutive" treatment categories are parallel and depend only on a linear combination of two covariates X_1 and X_2. Black lines represent true underlying decision boundaries, blue lines are boundaries estimated by S-step classifiers, and red lines are boundaries estimated by R-step classifiers. Since all

but the last S-steps compare one vs. multiple treatments, estimated boundaries in S-steps are biased towards the "multiple side". One vs. one comparisons in R-steps correct for this bias and recover true boundaries.



(a) $K = 3$ **(b)** $K = 4$

Fig. 3.1: *Illustration of estimated boundaries in sequential and re-estimation steps.*

We also justify the importance of including R-steps by presenting classification results obtained with (denoted by $\widehat{\mathcal{D}^*}_{\mathrm{SR}}(\mathbf{x})$) and without (denoted by $\widehat{\mathcal{D}^*}_{\mathrm{S}}(\mathbf{x})$) R-steps from a simulated example. We assume that there are 4 ordinal treatments and decision boundaries depend only on a linear combination of two independent covariates X_1 and X_2. Treatment A is sampled from $\{1, 2, 3, 4\}$ with equal probabilities, and the true optimal ITR, $\mathcal{D}^*(\mathbf{X}) = I(X_1 - 2X_2 < -1) + 2I(-1 \leq X_1 - 2X_2 < 0) + 3I(0 \leq X_1 - 2X_2 < 1) + 4I(X_1 - 2X_2 \geq 1)$. The outcome Y is simulated from a normal distribution with mean $X_1 + X_2 - 4|A - \mathcal{D}^*(\mathbf{X})|$ and variance 1. We examine cases with $n = 400$ and $n = 800$. The overall classification accuracies of $\widehat{\mathcal{D}^*}_{\mathrm{SR}}(\mathbf{x})$ and $\widehat{\mathcal{D}^*}_{\mathrm{S}}(\mathbf{x})$ are presented in Table 3.1. In addition, we show classification results by category with respect to true optimal ITRs (Table 3.2) and another classification-by-category measure that uses estimated ITRs as references (Table 3.3). Results presented in Tables 3.1-3.3 are evaluated on a large independent testing dataset of size 10000 based on 100 simulation replicates.

Table 3.1: *Mean (SD) of overall classification accuracy for the simulated example with 4 ordinal treatments based on 100 replicates. S corresponds to the method with only S-steps and no R-steps, and method SR corresponds to the proposed SR learning where both S- and R-steps are included.*

	Classification accuracy	
Method	$n = 400$	$n = 800$
S	0.78 (0.04)	0.78 (0.03)
SR	0.97 (0.01)	0.98 (0.01)

Table 3.2: *Mean (SD) of classification results by category (with respect to true optimal ITRs) for the simulated example with 4 ordinal treatments based on 100 replicates. S corresponds to the method with only S-steps and no R-steps, and method SR corresponds to the proposed SR learning where both S- and R-steps are included. True positive rates by category are in bold.*

		$n = 400$				$n = 800$			
Method	True optimal	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 1$	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 2$	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 3$	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 4$	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 1$	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 2$	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 3$	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 4$
S	$\mathcal{D}^*(\mathbf{x}) = 1$	**1.00 (0.00)**	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	**1.00 (0.00)**	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)
	$\mathcal{D}^*(\mathbf{x}) = 2$	0.57 (0.13)	**0.43 (0.13)**	0.00 (0.00)	0.00 (0.00)	0.56 (0.10)	**0.43 (0.10)**	0.00 (0.00)	0.00 (0.00)
	$\mathcal{D}^*(\mathbf{x}) = 3$	0.00 (0.00)	0.28 (0.07)	**0.70 (0.08)**	0.02 (0.02)	0.00 (0.00)	0.28 (0.06)	**0.70 (0.07)**	0.01 (0.01)
	$\mathcal{D}^*(\mathbf{x}) = 4$	0.00 (0.00)	0.00 (0.00)	0.02 (0.03)	**0.98 (0.03)**	0.00 (0.00)	0.00 (0.00)	0.01 (0.01)	**0.99 (0.01)**
SR	$\mathcal{D}^*(\mathbf{x}) = 1$	**0.98 (0.02)**	0.02 (0.02)	0.00 (0.00)	0.00 (0.00)	**0.99 (0.01)**	0.01 (0.01)	0.00 (0.00)	0.00 (0.00)
	$\mathcal{D}^*(\mathbf{x}) = 2$	0.02 (0.02)	**0.97 (0.03)**	0.02 (0.02)	0.00 (0.00)	0.01 (0.01)	**0.98 (0.02)**	0.01 (0.01)	0.00 (0.00)
	$\mathcal{D}^*(\mathbf{x}) = 3$	0.00 (0.00)	0.03 (0.03)	**0.96 (0.04)**	0.02 (0.02)	0.00 (0.00)	0.01 (0.01)	**0.98 (0.02)**	0.01 (0.01)
	$\mathcal{D}^*(\mathbf{x}) = 4$	0.00 (0.00)	0.00 (0.00)	0.02 (0.03)	**0.98 (0.03)**	0.00 (0.00)	0.00 (0.00)	0.01 (0.01)	**0.99 (0.01)**

Table 3.3: *Mean (SD) of classification results by category (with respect to estimated optimal ITRs) for the simulated example with 4 ordinal treatments based on 100 replicates. S corresponds to the method with only S-steps and no R-steps, and method SR corresponds to the proposed SR learning where both S- and R-steps are included. Positive predictive values by category are in bold.*

		$n = 400$				$n = 800$			
Method	Estimated optimal	$\mathcal{D}^*(\mathbf{x}) = 1$	$\mathcal{D}^*(\mathbf{x}) = 2$	$\mathcal{D}^*(\mathbf{x}) = 3$	$\mathcal{D}^*(\mathbf{x}) = 4$	$\mathcal{D}^*(\mathbf{x}) = 1$	$\mathcal{D}^*(\mathbf{x}) = 2$	$\mathcal{D}^*(\mathbf{x}) = 3$	$\mathcal{D}^*(\mathbf{x}) = 4$
S	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 1$	**0.65 (0.06)**	0.35 (0.05)	0.00 (0.00)	0.00 (0.00)	**0.65 (0.04)**	0.35 (0.04)	0.00 (0.00)	0.00 (0.00)
	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 2$	0.00 (0.00)	**0.59 (0.10)**	0.41 (0.10)	0.00 (0.00)	0.00 (0.00)	**0.60 (0.07)**	0.40 (0.07)	0.00 (0.00)
	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 3$	0.00 (0.00)	0.00 (0.00)	**0.97 (0.03)**	0.03 (0.03)	0.00 (0.00)	0.00 (0.00)	**0.99 (0.02)**	0.01 (0.02)
	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 4$	0.00 (0.00)	0.00 (0.00)	0.02 (0.02)	**0.98 (0.02)**	0.00 (0.00)	0.00 (0.00)	0.01 (0.01)	**0.99 (0.01)**
SR	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 1$	**0.98 (0.02)**	0.02 (0.02)	0.00 (0.00)	0.00 (0.00)	**0.99 (0.01)**	0.01 (0.01)	0.00 (0.00)	0.00 (0.00)
	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 2$	0.02 (0.02)	**0.95 (0.04)**	0.02 (0.03)	0.00 (0.00)	0.01 (0.01)	**0.97 (0.02)**	0.01 (0.01)	0.00 (0.00)
	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 3$	0.00 (0.00)	0.02 (0.02)	**0.96 (0.03)**	0.02 (0.03)	0.00 (0.00)	0.01 (0.01)	**0.98 (0.01)**	0.01 (0.01)
	$\widehat{\mathcal{D}}^*(\mathbf{x}) = 4$	0.00 (0.00)	0.00 (0.00)	0.02 (0.02)	**0.98 (0.02)**	0.00 (0.00)	0.00 (0.00)	0.01 (0.01)	**0.99 (0.01)**

The overall classification accuracy (Table 3.1) highlights a substantial improvement from the inclusion of R-steps. By examining classification results by category with respect to true optimal ITRs (Table 3.2), we observe that when re-estimation steps are ignored, among those whose true optimal treatments are 2 (i.e. $\mathcal{D}^*(\mathbf{x}) = 2$), more than 55% are misallocated to treatment 1 (i.e. $\widehat{\mathcal{D}}^*(\mathbf{x}) = 1$). Similarly, among those whose true optimal treatments are 3 (i.e. $\mathcal{D}^*(\mathbf{x}) = 3$), more than 25% are misallocated to treatment 2 (i.e. $\widehat{\mathcal{D}}^*(\mathbf{x}) = 2$). These problems remain even when the sample size is increased to 800. On the contrary, SR

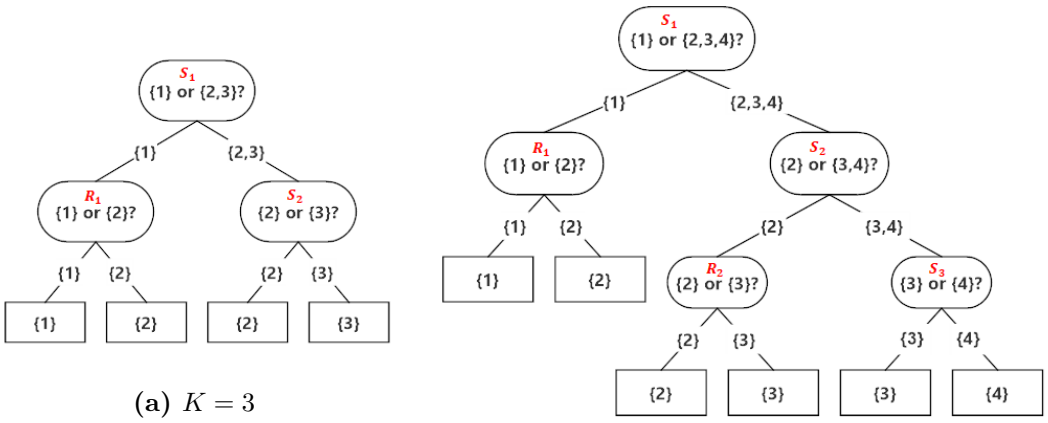
learning (with both S- and R-steps) reduces corresponding misallocation rates to less than 5%. True positive rates (in bold) of method "SR" for "middle" treatment categories are much higher than those of method "S", implying that by re-estimating boundaries between treatments 1 and 2 (with R_1), and between treatments 2 and 3 (with R_2), misclassifications between "consecutive" treatment categories can be corrected to a great extent.

Table 3.3 confers similar information and reassures the importance of R-steps: without R-steps, 35% of those whose optimal treatments are estimated as 1 (i.e. $\widehat{\mathcal{D}^*}(\mathbf{x}) = 1$) actually benefit most from treatment 2 (i.e. $\mathcal{D}^*(\mathbf{x}) = 2$), and about 40% of those whose optimal treatments are estimated as 2 (i.e. $\widehat{\mathcal{D}^*}(\mathbf{x}) = 2$) gain most by undertaking treatment 3 (i.e. $\mathcal{D}^*(\mathbf{x}) = 3$). Treatment decisions made according to both S- and R-steps (SR learning) are much more accurate and are almost perfect in terms of positive predictive values (in bold). As an aside, we note that classification results between treatments 3 and 4 are very similar for method "S" and method "SR". This is because when $K = 4$, the last sequential step S_3 is a one vs. one comparison between treatments 3 and 4 and thus the estimated boundary between them does not need to be corrected by further re-estimation steps, unlike the 1 vs. 2, and the 2 vs. 3 boundaries, which require R-steps to rectify the bias introduced in S-steps.

3.2.2.2 Ensembling

For a given subject with covariates $\mathbf{x}$, we can estimate his/her optimal treatment based on $\widehat{S}_k(\mathbf{x})$, $k = 1, \ldots, K-1$, and $\widehat{R}_k(\mathbf{x})$, $k = 1, \ldots, K-2$. We demonstrate the idea of aggregating multiple binary treatment selection decisions and predicting the optimal treatment among K ordinal options by tree diagrams in Figure 3.2 ((a) for $K = 3$ and (b) for $K = 4$).

For example, when $K = 3$, the optimal treatment for a subject with covariates $\mathbf{x}$ is $I\big(\widehat{S}_1(\mathbf{x}) = 1\big)\widehat{R}_1(\mathbf{x}) + I\big(\widehat{S}_1(\mathbf{x}) \neq 1\big)\widehat{S}_2(\mathbf{x})$, implying that if the first sequential step favours {1} over {2,3}, then the final decision is determined by the re-estimation step where {1} and {2} are compared. Otherwise, the decision depends on the second sequential step. When $K = 4$, the optimal treatment is given by $I\big(\widehat{S}_1(\mathbf{x}) = 1\big)\widehat{R}_1(\mathbf{x}) + I\big(\widehat{S}_1(\mathbf{x}) \neq 1, \widehat{S}_2(\mathbf{x}) = 2\big)\widehat{R}_2(\mathbf{x}) + I\big(\widehat{S}_1(\mathbf{x}) \neq 1, \widehat{S}_2(\mathbf{x}) \neq 2\big)\widehat{S}_3(\mathbf{x})$. The idea follows similarly for $K > 4$, where the optimal



(a) $K = 3$

(b) $K = 4$

Fig. 3.2: *Illustration of how multiple binary decisions are aggregated for ordinal prediction.*

decision is:

$$\begin{aligned}\widehat{\mathcal{D}^*}(\mathbf{x}) =& I\Big(\widehat{S}_1(\mathbf{x}) = 1\Big)\widehat{R}_1(\mathbf{x}) + I\Big(\widehat{S}_1(\mathbf{x}) \neq 1, \widehat{S}_2(\mathbf{x}) = 2\Big)\widehat{R}_2(\mathbf{x}) + \dots \\ &+ I\Big(\widehat{S}_1(\mathbf{x}) \neq 1, \dots, \widehat{S}_{K-3}(\mathbf{x}) \neq K-3, \widehat{S}_{K-2}(\mathbf{x}) = K-2\Big)\widehat{R}_{K-2}(\mathbf{x}) \\ &+ I\Big(\widehat{S}_1(\mathbf{x}) \neq 1, \dots, \widehat{S}_{K-2}(\mathbf{x}) \neq K-2\Big)\widehat{S}_{K-1}(\mathbf{x}).\end{aligned} \tag{3.3}$$

3.2.3 Training binary classifiers S_k and R_k

In principle, all methods developed for estimating the optimal ITR in settings with two treatment options are applicable for learning binary decision rules in S- and R-steps. In practice, due to the reduced sample sizes in intermediate steps, we recommend using methods with high convergence rate and desirable small sample performance for binary comparisons (Liu et al., 2018; Zhou et al., 2017). We follow Zhou et al. (2018) and adopt in our implementation the augmented outcome weighted learning (AOL) method proposed by Liu et al. (2018) for solving binary subproblems and training binary classifiers S_k and R_k in SR learning. In the following discussion, we take S_k as an example and focus on deriving the training process for S_k. R_k can be trained in a similar way.

For notational clarity, we denote the new treatment label for training S_k by A^{S_k} (without loss of generality, we assume $A^{S_k} \in \{-1, 1\}$), the number of eligible subjects included in training S_k by n^{S_k}, the propensity score associated with the binary treatment label by $P^{S_k}(A^{S_k}|\,\mathbf{X})$ (the propensity score can be

estimated via logistic regression), and an ITR for the comparison between $\{k\}$ and $\{k+1,\ldots,K\}$ in S_k by $\mathcal{D}^{S_k}$, for $k=1,\ldots,K-1$. Similar to (3.1), the value function associated with $\mathcal{D}^{S_k}$ for the binary comparison in S_k can be written as:

$$V(\mathcal{D}^{S_k}) = E\left[\frac{I\{A^{S_k}=\mathcal{D}^{S_k}(\mathbf{X})\}}{P^{S_k}(A^{S_k}|\,\mathbf{X})}Y\right]. \tag{3.4}$$

The augmented outcome weighted learning (AOL) method proposed by Liu et al. (2018) improves the efficiency of the outcome weighted learning (OWL) method discussed in 2.2.2 (whereby more stable ITR estimation can be achieved) by using a modified weight to replace the classification weight $Y/P^{S_k}(A^{S_k}|\,\mathbf{X})$ in (3.4). Liu et al. (2018) showed that

$$\begin{aligned}
V(\mathcal{D}^{S_k}) &= E\left[\frac{I\{A^{S_k}=\mathcal{D}^{S_k}(\mathbf{X})\}}{P^{S_k}(A^{S_k}|\,\mathbf{X})}Y\right] \\
&= E\left[\frac{I\{A^{S_k}=\mathcal{D}^{S_k}(\mathbf{X})\}\{Y-m(\mathbf{X})\}}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\right] + E\left[\frac{I\{A^{S_k}=\mathcal{D}^{S_k}(\mathbf{X})\}m(\mathbf{X})}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\right] \\
&= E\left[\frac{I\{A^{S_k}=\mathcal{D}^{S_k}(\mathbf{X})\}\{Y-m(\mathbf{X})\}}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\right] + E\left\{E\left[\frac{I\{A^{S_k}=\mathcal{D}^{S_k}(\mathbf{X})\}m(\mathbf{X})}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\middle|\,\mathbf{X}\right]\right\} \\
&= E\left[\frac{I\{A^{S_k}=\mathcal{D}^{S_k}(\mathbf{X})\}\{Y-m(\mathbf{X})\}}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\right] + E\left\{m(\mathbf{X})E\left[\frac{I\{A^{S_k}=\mathcal{D}^{S_k}(\mathbf{X})\}}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\middle|\,\mathbf{X}\right]\right\},
\end{aligned} \tag{3.5}$$

where $m(\mathbf{X})$ can be any measurable function that depends only on $\mathbf{X}$ (and not on A). We note that

$$\begin{aligned}
&E\left[\frac{I\{A^{S_k}=\mathcal{D}^{S_k}(\mathbf{X})\}}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\middle|\,\mathbf{X}\right] \\
&= E\Big[I\Big\{\mathcal{D}^{S_k}(\mathbf{X})=1\Big\}\Big|\,\mathbf{X}, A^{S_k}=1\Big] + E\Big[I\Big\{\mathcal{D}^{S_k}(\mathbf{X})=-1\Big\}\Big|\,\mathbf{X}, A^{S_k}=-1\Big] \\
&= 1.
\end{aligned} \tag{3.6}$$

Therefore,

$$V(\mathcal{D}^{S_k}) = E\left[\frac{I\{A^{S_k}=\mathcal{D}^{S_k}(\mathbf{X})\}\{Y-m(\mathbf{X})\}}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\right] + E\{m(\mathbf{X})\}. \tag{3.7}$$

The modified weight (i.e. $\{Y - m(\mathbf{X})\}/P^{S_k}(A^{S_k}|\,\mathbf{X})$) in (3.7) may be negative. To address this problem, Liu et al. (2018) rewrote (3.7) as:

$$\begin{aligned} V(\mathcal{D}^{S_k}) &= E\Big[\frac{I\{A^{S_k} = \mathcal{D}^{S_k}(\mathbf{X})\}\{Y - m(\mathbf{X})\}}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\Big] + E\{m(\mathbf{X})\} \\ &= E\Big[\frac{I\{A^{S_k}\text{sign}\{Y - m(\mathbf{X})\} = \mathcal{D}^{S_k}(\mathbf{X})\}|Y - m(\mathbf{X})|}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\Big] - E\Big[\frac{\{Y - m(\mathbf{X})\}^-}{P^{S_k}(A^{S_k}|\,\mathbf{X})} - m(\mathbf{X})\Big], \end{aligned} \tag{3.8}$$

where $(\bullet)^- = -\min(0, \bullet)$. According to (3.8),

$$\arg\max_{\mathcal{D}^{S_k}} V(\mathcal{D}^{S_k}) = \arg\max_{\mathcal{D}^{S_k}} E\left[\frac{I\{A^{S_k}\text{sign}\{Y - m(\mathbf{X})\} = \mathcal{D}^{S_k}(\mathbf{X})\}|Y - m(\mathbf{X})|}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\right]. \tag{3.9}$$

Intuitively, the new objective function in (3.9) is sensible given that

(i) $m(\mathbf{X})$ models the main effect and it does not contain information on treatment-covariate interactions. Therefore, subtracting $m(\mathbf{X})$ from Y should have no influence on the estimated optimal ITR.

(ii) for those with large $|Y - m(\mathbf{X})|$, if $Y - m(\mathbf{X}) > 0$ (i.e. the observed outcome is larger than expected), then their optimal treatments tend to be the same as the observed treatments A^{S_k}, while if $Y - m(\mathbf{X}) < 0$ (i.e. the observed outcome is smaller than expected), then their optimal treatments are likely to be different from A^{S_k}.

As a remark, Liu et al. (2018) proved that AOL is Fisher consistent regardless of the choice of $m(\mathbf{X})$, and if $m(\mathbf{X})$ is chosen properly (i.e. $m(\mathbf{X})$ helps reduce the overall variability of the outcome), the resulting ITR estimator will be less variable compared to that based on OWL. In Liu et al. (2018), $m(\mathbf{X})$ was chosen as $E(Y|\,\mathbf{X})$ and it can be estimated as the fitted value from the model that regresses Y on $\mathbf{X}$.

Solving (3.9) is equivalent to finding the ITR that minimises the weighted misclassification error, which is given as:

$$E\left[\frac{I\{A^{S_k}\text{sign}\{Y - m(\mathbf{X})\} \neq \mathcal{D}^{S_k}(\mathbf{X})\}|Y - m(\mathbf{X})|}{P^{S_k}(A^{S_k}|\,\mathbf{X})}\right]. \tag{3.10}$$

Empirically, the optimal ITR can be obtained by minimising

$$\frac{1}{n^{S_k}} \sum_{i=1}^{n^{S_k}} \frac{I\{A_i^{S_k}\text{sign}\{Y_i - m(\mathbf{X}_i)\} \neq \mathcal{D}^{S_k}(\mathbf{X}_i)\}|Y_i - m(\mathbf{X}_i)|}{P^{S_k}(A_i^{S_k}|\,\mathbf{X}_i)}. \tag{3.11}$$

To simplify notations, we let $e_i = Y_i - m(\mathbf{X}_i)$ denote the residual, and $\pi_i^{S_k} = P^{S_k}(A_i^{S_k} | \mathbf{X}_i)$ denote the propensity score such that (3.11) can be written as:

$$\frac{1}{n^{S_k}} \sum_{i=1}^{n^{S_k}} \frac{I\{A_i^{S_k} \text{sign}(e_i) \neq \mathcal{D}^{S_k}(\mathbf{X}_i)\}|e_i|}{\pi_i^{S_k}}. \tag{3.12}$$

As suggested by Zhao et al. (2012), any ITR $\mathcal{D}^{S_k}(\mathbf{X}_i)$ can be represented as $\mathcal{D}^{S_k}(\mathbf{X}_i) = \text{sign}\{f^{S_k}(\mathbf{X}_i)\}$, for some measurable decision function f^{S_k}. Therefore, (3.12) is equivalent to

$$\frac{1}{n^{S_k}} \sum_{i=1}^{n^{S_k}} \frac{I\{A_i^{S_k} \text{sign}(e_i) f^{S_k}(\mathbf{X}_i) < 0\}|e_i|}{\pi_i^{S_k}}. \tag{3.13}$$

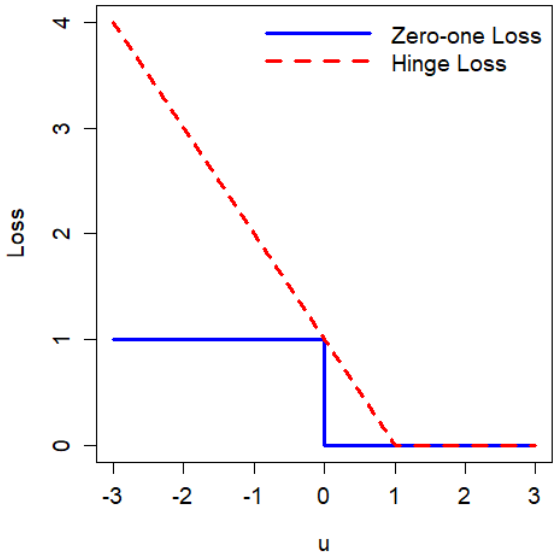


Fig. 3.3: *Zero-one loss and the approximated hinge loss.*

The zero-one loss function, $\Phi_{\text{zero-one}}(u) = I(u < 0)$, is discontinuous and nonconvex (see Figure 3.3) and thus minimising (3.13) is non-deterministic polynomial-time (NP) hard. This problem can be addressed by replacing the zero-one loss with a convex surrogate loss, for example, the hinge loss (Figure 3.3), $\Phi_{\text{hinge}}(u) = (1-u)^+$, where $(\bullet)^+ = \max(0, \bullet)$. Then, we can estimate the optimal decision function by solving a weighted support vector machine (SVM) problem

(Zhao et al., 2012). The estimated optimal decision function $\widehat{f^*}^{S_k}(\mathbf{X})$ minimises

$$\frac{1}{n^{S_k}}\sum_{i=1}^{n^{S_k}}\frac{\{1-A_i^{S_k}\text{sign}(e_i)f^{S_k}(\mathbf{X}_i)\}^+|e_i|}{\pi_i^{S_k}}+\lambda\|f^{S_k}\|^2. \tag{3.14}$$

An l_2 regularisation term $\lambda\|f^{S_k}\|^2$ is included in the objective function (3.14) to avoid over-fitting, where λ is the tuning parameter (Liu et al., 2018; Zhao et al., 2012). We note that minimising (3.14) is a convex optimisation problem. Similar to SVM-type algorithms, we introduce slack variables ξ_i, $i=1,\ldots,n^{S_k}$, and then the minimisation of (3.14) can be reformulated as follows:

$$\begin{aligned}&\min\ \frac{1}{2}\|f^{S_k}\|^2+C\sum_{i=1}^{n^{S_k}}\frac{\xi_i|e_i|}{\pi_i^{S_k}},\\ &\text{s.t.}\quad\begin{cases}A_i^{S_k}\text{sign}(e_i)f^{S_k}(\mathbf{X}_i)\geq 1-\xi_i, & i=1,\ldots,n^{S_k},\\ \xi_i\geq 0, & i=1,\ldots,n^{S_k},\end{cases}\end{aligned}$$

where $C>0$ is the regularisation (cost) parameter that determines the penalty on misclassified cases, and it is typically selected via cross-validation (in our case, we choose the one that maximises the empirical value function on the validation set). By introducing Lagrange multipliers $\boldsymbol{\alpha}$ and $\boldsymbol{\mu}$, the Lagrange function for the primal is:

$$\frac{1}{2}\|f^{S_k}\|^2+C\sum_{i=1}^{n^{S_k}}\frac{\xi_i|e_i|}{\pi_i^{S_k}}-\sum_{i=1}^{n^{S_k}}\alpha_i\{A_i^{S_k}\text{sign}(e_i)f^{S_k}(\mathbf{X}_i)-(1-\xi_i)\}-\sum_{i=1}^{n^{S_k}}\mu_i\xi_i, \tag{3.15}$$

where $\alpha_i\geq 0$, $\mu_i\geq 0$, for $i=1,\ldots,n^{S_k}$. The decision function f^{S_k} can either be linear or nonlinear (with respect to $\mathbf{X}_i$), and we discuss both in the following.

3.2.3.1 Linear decision function

For linear decision functions, $f^{S_k}(\mathbf{X})=\beta_0+\mathbf{X}^\mathsf{T}\boldsymbol{\beta}$, where $\boldsymbol{\beta}=(\beta_1,\ldots,\beta_p)^\mathsf{T}$. The primal (3.15) can be written as:

$$\frac{1}{2}\|\boldsymbol{\beta}\|^2+C\sum_{i=1}^{n^{S_k}}\frac{\xi_i|e_i|}{\pi_i^{S_k}}-\sum_{i=1}^{n^{S_k}}\alpha_i\{A_i^{S_k}\text{sign}(e_i)(\beta_0+\mathbf{X}_i^\mathsf{T}\boldsymbol{\beta})-(1-\xi_i)\}-\sum_{i=1}^{n^{S_k}}\mu_i\xi_i. \tag{3.16}$$

Taking derivatives of (3.16) with respect to β_0, $\boldsymbol{\beta}$, and ξ_i $(i = 1, \ldots, n^{S_k})$, and setting them to zero, we get

$$\begin{cases} 0 = \sum_{i=1}^{n^{S_k}} \alpha_i A_i^{S_k} \text{sign}(e_i) \\ \boldsymbol{\beta} = \sum_{i=1}^{n^{S_k}} \alpha_i A_i^{S_k} \text{sign}(e_i) \mathbf{X}_i \\ \alpha_i = C|e_i|/\pi_i^{S_k} - \mu_i, \text{ for } i = 1, \ldots, n^{S_k}. \end{cases} \tag{3.17}$$

Plugging (3.17) into (3.16), we derive the dual problem as follows:

$$\begin{aligned} &\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n^{S_k}} \alpha_i - \frac{1}{2} \sum_{i=1}^{n^{S_k}} \sum_{j=1}^{n^{S_k}} \alpha_i \alpha_j A_i^{S_k} \text{sign}(e_i) A_j^{S_k} \text{sign}(e_j) \mathbf{X}_i^{\mathsf{T}} \mathbf{X}_j, \\ &\text{s.t.} \quad \begin{cases} 0 \le \alpha_i \le C|e_i|/\pi_i^{S_k}, \text{ for } i = 1, \ldots, n^{S_k}, \\ \sum_{i=1}^{n^{S_k}} \alpha_i A_i^{S_k} \text{sign}(e_i) = 0, \end{cases} \end{aligned} \tag{3.18}$$

which can be solved efficiently using quadratic programming algorithms. The coefficient vector $\boldsymbol{\beta}$ can then be estimated as $\widehat{\boldsymbol{\beta}} = \sum_{i=1}^{n^{S_k}} \widehat{\alpha_i} A_i^{S_k} \text{sign}(e_i) \mathbf{X}_i$. The intercept term β_0 can be obtained based on Karush–Kuhn–Tucker (KKT) conditions, $(C|e_i|/\pi_i^{S_k} - \alpha_i)\xi_i = 0$ and $\alpha_i\{A_i^{S_k} \text{sign}(e_i)(\beta_0 + \mathbf{X}_i^{\mathsf{T}}\boldsymbol{\beta}) - (1 - \xi_i)\} = 0$. When $0 < \alpha_i < C|e_i|/\pi_i^{S_k}$, we have $\xi_i = 0$ and $A_i^{S_k} \text{sign}(e_i)(\beta_0 + \mathbf{X}_i^{\mathsf{T}}\boldsymbol{\beta}) - 1 = 0$. Therefore, $\widehat{\beta_0}$ can be calculated by taking the average of $\{A_i^{S_k} \text{sign}(e_i)\}^{-1} - \mathbf{X}_i^{\mathsf{T}}\widehat{\boldsymbol{\beta}}$ among those satisfying $0 < \alpha_i < C|e_i|/\pi_i^{S_k}$, for $i = 1, \ldots, n^{S_k}$. The optimal decision function is estimated as $\widehat{f^*}^{S_k}(\mathbf{X}) = \widehat{\beta_0} + \mathbf{X}^{\mathsf{T}}\widehat{\boldsymbol{\beta}} = \widehat{\beta_0} + \sum_{i=1}^{n^{S_k}} \widehat{\alpha_i} A_i^{S_k} \text{sign}(e_i)\, \mathbf{X}^{\mathsf{T}}\, \mathbf{X}_i$, and the optimal ITR for S_k is given by $\widehat{\mathcal{D}^*}^{S_k}(\mathbf{X}) = \text{sign}\{\widehat{f^*}^{S_k}(\mathbf{X})\}$.

3.2.3.2 Nonlinear decision function

For nonlinear decision functions, $f^{S_k}(\mathbf{X})$ can be written as $f^{S_k}(\mathbf{X}) = \beta_0 + h(\mathbf{X}) = \beta_0 + \phi(\mathbf{X})^{\mathsf{T}}\boldsymbol{\gamma}$, where $h(\mathbf{X}) \in \mathcal{H}_{\mathcal{K}}$ and $\mathcal{H}_{\mathcal{K}}$ is a reproducing kernel Hilbert space (RKHS) associated with the kernel function $\mathcal{K}(\bullet, \bullet)$ (maps from $\mathcal{X} \times \mathcal{X}$ to $\mathbb{R}$), $\phi(\bullet)$ is a kernel mapping function, and $\boldsymbol{\gamma}$ is the parameter vector associated with $\phi(\mathbf{X})$. For example, the Gaussian radial basis function (RBF) kernel is a universal kernel (has nice theoretical properties, e.g. universal consistency) and it is the most widely used nonlinear kernel in practice (Liu et al., 2018; Zhou et al., 2017) due to its flexibility (i.e. provides good approximations to any continuous function). Gaussian RBF kernel can be written as $\mathcal{K}_\sigma^{\text{RBF}}(u, v) = \exp\Big(- \|u - v\|^2 / 2\sigma^2 \Big)$, where $\sigma > 0$ is the bandwidth parameter that determines how far the influence of a data point reaches, and it is usually selected via cross-validation (in our

case, we select the "best" σ together with the cost parameter C in the SVM formulation and pick the pair that maximises the empirical value function on the validation set).

In the case of nonlinear decision functions, we replace $\mathbf{X}_i^\mathsf{T}\mathbf{X}_j$ in (3.18) with $\phi(\mathbf{X}_i)^\mathsf{T}\phi(\mathbf{X}_j) = \mathcal{K}_\sigma^{\text{RBF}}(\mathbf{X}_i, \mathbf{X}_j)$ and solve the following dual problem via quadratic programming:

$$\begin{aligned} &\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n^{S_k}} \alpha_i - \frac{1}{2}\sum_{i=1}^{n^{S_k}}\sum_{j=1}^{n^{S_k}} \alpha_i\alpha_j A_i^{S_k}\text{sign}(e_i)A_j^{S_k}\text{sign}(e_j)\mathcal{K}_\sigma^{\text{RBF}}(\mathbf{X}_i, \mathbf{X}_j), \\ &\text{s.t.} \quad \begin{cases} 0 \le \alpha_i \le C|e_i|/\pi_i^{S_k}, \text{ for } i = 1, \ldots, n^{S_k}, \\ \sum_{i=1}^{n^{S_k}} \alpha_i A_i^{S_k}\text{sign}(e_i) = 0. \end{cases} \end{aligned} \tag{3.19}$$

$\widehat{\boldsymbol{\alpha}}$ is obtained from (3.19), and β_0 can be estimated in the same way as done for linear decision functions. The estimated optimal nonlinear decision function is then given by $\widehat{f^*}^{S_k}(\mathbf{X}) = \widehat{\beta_0} + \sum_{i=1}^{n^{S_k}} \widehat{\alpha_i} A_i^{S_k}\text{sign}(e_i)\mathcal{K}_\sigma^{\text{RBF}}(\mathbf{X}, \mathbf{X}_i)$, and the optimal ITR for S_k is $\widehat{\mathcal{D}^*}^{S_k}(\mathbf{X}) = \text{sign}\{\widehat{f^*}^{S_k}(\mathbf{X})\}$.

3.2.4 Variable selection

With rapid technological advancement in collecting individual-level information, an increasing number of clinical and biological covariates can be measured and are available in clinical studies. However, some information may be unnecessary and substantial resources can be saved by measuring only relevant covariates. In addition, in the presence of high-dimensional covariates, noise variables may "pollute" and impair model performance (increase computational time, affect convergence, decrease generalisability and prediction accuracy, etc.) due to the limited sample sizes in clinical studies - a phenomenon commonly referred to as "the curse of dimensionality". Interpretability is also a major concern in this case as models that are fitted with many covariates can be hard to interpret. In the context of ITR estimation, only covariates that interact qualitatively with the treatment are of clinical importance for decision-making; and thus it is crucial to identify key covariates that have impact on the optimal treatment decisions through variable selection, and to derive simple and practically implementable decision rules (Fan et al., 2016; Gunter et al., 2011; Song et al., 2015; Zhang and Zhang, 2018; Zhou et al., 2017).

When the dimension of the covariate space is relatively high, we follow the SR learning framework introduced in Section 3.2.2 and incorporate the variable

selection feature into the training of each binary classifier S_k and R_k to select a subset of covariates and improve the discriminative ability of each classifier. Selecting important covariates for each classifier independently also makes sense in practice, since it is reasonable to think that boundaries between "consecutive" treatment categories may depend on different covariates, and performing variable selection independently for each classifier allows for this flexibility. As before, we take S_k as an example. R_k can be trained in a similar manner.

In general, variable selection methods can be classified into three categories: filter methods, wrapper methods, and embedded methods (Saeys et al., 2007). Filter methods evaluate the feature importance and select variables without considering the modelling process (i.e. purely based on the intrinsic property of the data using some predefined metrics such as the information gain). The selected best subset of variables will then be used as the model input. Filter methods are computationally efficient in general and are less prone to over-fitting. However, such methods ignore the interaction between the variable and the model of interest, and thus may fail to select the most useful features. Wrapper methods search the space of all variable subsets, and then different feature subsets are evaluated based on a given learning algorithm. Compared to filter methods, wrapper methods select the best subset of variables on the basis of a specific learning algorithm (e.g. choose the subset that maximises the classification accuracy) and account for variable dependencies, but they are typically more computationally intensive and are more susceptible to over-fitting. The third class of variable selection methods is commonly referred to as "embedded methods". For embedded methods, the search for an optimal variable subset is embedded/integrated into the learning algorithm and the variable search is guided by the learning process (e.g. built-in feature selection methods such as LASSO and ridge). These methods are similar to wrapper methods in the sense that they select variables that optimise the performance of a particular learning algorithm, but they can be less computationally expensive and less prone to over-fitting (combine the efficiency of filters with the accuracy of wrappers). In particular, for SR learning, we propose to use an embedded variable selection method when the decision function is linear, and a wrapper method for variable selection when the decision function is nonlinear.

3.2.4.1 Linear decision function

In Section 3.2.3, the objective function (3.14) for training binary classifiers S_k includes the l_2 penalty term $\lambda\|f^{S_k}\|^2$. It is well-known that l_2 penalty shrinks coefficients towards zero, but does not perform variable selection nor lead to sparse solutions. In contrast, l_1 penalty allows some coefficients to be exactly zero when λ is sufficiently large and inherently performs variable selection (Tibshirani, 1996). Analogous to the 1-norm SVM formulation (Bradley and Mangasarian, 1998; Fung and Mangasarian, 2004; Mangasarian, 2006; Zhu et al., 2003), we replace the l_2 penalty term $\lambda\|f^{S_k}\|^2$ in (3.14) with the l_1 penalty $\lambda\|f^{S_k}\|$ to incorporate the variable selection feature into the AOL framework. The optimal decision function can be estimated by minimising

$$\frac{1}{n^{S_k}}\sum_{i=1}^{n^{S_k}}\frac{\{1-A_i^{S_k}\text{sign}(e_i)f^{S_k}(\mathbf{X}_i)\}^+|e_i|}{\pi_i^{S_k}}+\lambda\|f^{S_k}\|, \tag{3.20}$$

where λ is the tuning parameter that controls the trade-off between the misclassification error and the complexity of the estimated decision function. For linear decision functions, $f^{S_k}(\mathbf{X})=\beta_0+\mathbf{X}^\mathsf{T}\boldsymbol{\beta}$, where $\boldsymbol{\beta}=(\beta_1,\ldots,\beta_p)^\mathsf{T}$. By introducing slack variables ξ_i, minimising (3.20) is equivalent to solving the following problem:

$$\begin{aligned}&\min\ \sum_{j=1}^{p}|\beta_j|+C\sum_{i=1}^{n^{S_k}}\frac{\xi_i|e_i|}{\pi_i^{S_k}},\\ &\text{s.t.}\ \begin{cases}A_i^{S_k}\text{sign}(e_i)(\beta_0+\mathbf{X}_i^\mathsf{T}\boldsymbol{\beta})\geq 1-\xi_i, & i=1,\ldots,n^{S_k},\\ \xi_i\geq 0, & i=1,\ldots,n^{S_k},\end{cases}\end{aligned} \tag{3.21}$$

where C is the cost parameter. We rewrite β_j as $\beta_j=r_j-s_j$ and $|\beta_j|$ as $|\beta_j|=r_j+s_j$, where $r_j\geq 0$ and $s_j\geq 0$ for $j=1,\ldots,p$. We note that there is a unique correspondence between β_j and (r_j,s_j), and indeed there are only three possible solutions for (r_j,s_j), namely, $(0,0)$, $(r_j,0)$ with $r_j>0$, or $(0,s_j)$ with $s_j>0$ (Bi et al., 2003; Fung and Mangasarian, 2004; Kecman and Hadzic, 2000). Covariate X_j is not selected when $(r_j,s_j)=(0,0)$, while in the other two cases, X_j is regarded as important and kept in the decision function. Based on the

aforementioned reparameterisation of β_j and $|\beta_j|$, (3.21) can be reformulated as:

$$\begin{aligned}
&\min_{(\boldsymbol{r},\boldsymbol{s},\beta_0,\boldsymbol{\xi})} \ \sum_{j=1}^{p}(r_j+s_j)+C\sum_{i=1}^{n^{S_k}}\frac{\xi_i|e_i|}{\pi_i^{S_k}},\\
&\text{s.t.} \begin{cases}
\xi_i \geq 1-A_i^{S_k}\text{sign}(e_i)\{\beta_0+\mathbf{X}_i^{\mathsf{T}}(\boldsymbol{r}-\boldsymbol{s})\}, & i=1,\ldots,n^{S_k},\\
\xi_i \geq 0, & i=1,\ldots,n^{S_k},\\
r_j \geq 0, & j=1,\ldots,p,\\
s_j \geq 0, & j=1,\ldots,p.
\end{cases}
\end{aligned} \tag{3.22}$$

This optimisation problem can be solved efficiently using standard linear programming solvers, for example, `linprog` in Matlab solves the following problem:

$$\begin{aligned}
&\min_{\boldsymbol{u}} \ f^{\mathsf{T}}\boldsymbol{u},\\
&\text{s.t.} \begin{cases}
\boldsymbol{g}_l \leq \boldsymbol{u} \leq \boldsymbol{g}_u,\\
\mathbf{E}\cdot\boldsymbol{u}=\boldsymbol{d},\\
\mathbf{C}\cdot\boldsymbol{u}\leq\boldsymbol{b}.
\end{cases}
\end{aligned} \tag{3.23}$$

In our case, the primal variable to solve is $\boldsymbol{u}=\begin{bmatrix}\boldsymbol{r}\\ \boldsymbol{s}\\ \beta_0\\ \boldsymbol{\xi}\end{bmatrix}=[r_1,\ldots,r_p,s_1,\ldots,s_p,\beta_0,\xi_1,\ldots,\xi_{n^{S_k}}]^{\mathsf{T}}$, where $\boldsymbol{r}$, $\boldsymbol{s}$ and $\boldsymbol{\xi}$ are column vectors. The coefficient vector associated with $\boldsymbol{u}$ is $f=[\,\mathbf{1}_p,\mathbf{1}_p,0,C|\boldsymbol{e}|/\boldsymbol{\pi}^{S_k}]^{\mathsf{T}}$. There is no upper bound for $\boldsymbol{u}$, so $\boldsymbol{g}_u=[\,]$. The lower bound $\boldsymbol{g}_l=[\mathbf{0}_p,\mathbf{0}_p,-\infty,\mathbf{0}_{n^{S_k}}]^{\mathsf{T}}$. In addition, there are no equality constraints, therefore, $\mathbf{E}=[\,]$ and $\boldsymbol{d}=[\,]$. For this problem, matrix $\mathbf{C}$ is of dimension $n^{S_k}\times(2p+n^{S_k}+1)$, and each row of $\mathbf{C}$ corresponds to the coefficient vector of a linear inequality constraint. According to (3.22), $\mathbf{C}$ can be expressed as $\mathbf{C}=[\underbrace{-\text{diag}\{\boldsymbol{A}^{S_k}\text{sign}(\boldsymbol{e})\}\mathbf{X}^{\mathsf{T}}}_{n^{S_k}\times p},\underbrace{\text{diag}\{\boldsymbol{A}^{S_k}\text{sign}(\boldsymbol{e})\}\mathbf{X}^{\mathsf{T}}}_{n^{S_k}\times p},\underbrace{-\boldsymbol{A}^{S_k}\text{sign}(\boldsymbol{e})}_{n^{S_k}\times 1},\underbrace{-\mathbf{I}_{n^{S_k}}}_{n^{S_k}\times n^{S_k}}]$, where $\text{diag}(\bullet)$ is a diagonal matrix with main diagonal entries being $\bullet$, and $\mathbf{I}$ is the identity matrix. The upper bound vector of linear constraints is $\boldsymbol{b}=-\mathbf{1}_{n^{S_k}}^{\mathsf{T}}$. The solution to this optimisation problem is $\widehat{\boldsymbol{u}}=\begin{bmatrix}\widehat{\boldsymbol{r}}\\ \widehat{\boldsymbol{s}}\\ \widehat{\beta_0}\\ \widehat{\boldsymbol{\xi}}\end{bmatrix}$. The optimal decision function is estimated as $\widehat{f^*}^{S_k}(\mathbf{X})=\widehat{\beta_0}+\mathbf{X}^{\mathsf{T}}(\widehat{\boldsymbol{r}}-\widehat{\boldsymbol{s}})$, and the optimal ITR for S_k is given by $\widehat{\mathcal{D}^*}^{S_k}(\mathbf{X})=\text{sign}\{\widehat{f^*}^{S_k}(\mathbf{X})\}$.

3.2.4.2 Nonlinear decision function

We can apply similar techniques discussed in Section 3.2.4.1 to nonlinear decision functions. However, this will result in a penalised kernel space (a higher-dimensional transformed space) rather than a penalised input feature space, i.e. instead of reducing the number of input space features as in the linear case, imposing l_1 penalty in the nonlinear case reduces the number of kernel functions used to generate the nonlinear classifier (dimensionality of the transformed space) (Adeli et al., 2017; Dasgupta and Huang, 2020; Fung and Mangasarian, 2004; Mangasarian and Wild, 2007). In the SVM literature, Mangasarian and Wild (2007) addressed this problem by introducing a diagonal matrix $\mathbf{D}$ with diagonal entries being either zero or one, where zeros correspond to eliminated features and ones correspond to features utilised for constructing nonlinear decision rules. They then solved a mixed-integer nonlinear programming problem by alternating between solving a linear programming problem and updating the diagonal elements of $\mathbf{D}$. One major issue with this approach is that the mixed-integer programming problem is nonconvex (Nguyen and de la Torre, 2010). Therefore, the optimisation procedure under the framework proposed by Mangasarian and Wild (2007) can be computationally demanding, very sensitive to the choice of starting values, and may get stuck at local optima instead of converging to a solution that is globally optimal (Boyd and Vandenberghe, 2004).

In practice, picking a decent starting value for a moderately high-dimensional vector can be challenging. In our case, it might be tempting to use the variable selection result from linear kernel as a starting point for nonlinear kernel. However, numerical experiments suggest that when true decision boundaries are nonlinear, variable selection with linear kernel cannot distinguish the real signals from the noise variables, and thus it does not provide a good "initial guess" of which variables are important in constructing the nonlinear decision boundaries. For this reason, instead of seeking a good starting value for the nonconvex optimisation problem, we propose a two-stage procedure to select important covariates and learn nonlinear classification rules when the number of covariates collected in clinical studies is relatively large. The basic idea is to first identify a subset of informative covariates that predict treatment effect heterogeneity with first-order effects X_j or second-order effects X_jX_k ($1 \leq j \leq k \leq p$; quadratic terms when $j = k$ and two-way interactions when $j \neq k$), and then train binary classifiers with nonlinear decision functions using selected covariates. This two-stage procedure is similar to the variable selection approach proposed by Bi et al. (2003), but

the methods that we employ for covariate screening in the first stage and for nonlinear classification in the second stage are different from those used by Bi et al. (2003).

Specifically, in the first stage, we "prescreen" covariates using the "stepwise conditional likelihood variable selection for discriminant analysis (SODA)" method proposed by Li and Liu (2019). SODA was developed for solving high-dimensional classification problems under the logistic regression framework. It performs variable selection for both first-order and second-order terms in a robust and efficient manner through a stepwise procedure and has been shown to enjoy superior performance in terms of variable selection accuracy, classification accuracy, and robustness to non-normality of covariate distributions. We refer readers to the original publication for details (Li and Liu, 2019). In our context, we apply SODA to train a logistic regression classifier with the class label $A^{S_k}\text{sign}(e)/2 + 1/2$ and covariates $(\mathbf{X}, \mathbf{X} \otimes \mathbf{X})$, where $\mathbf{X} \otimes \mathbf{X}$ represents all second-order terms of $\mathbf{X}$ (quadratic terms and two-way interaction terms). We retain covariates that are part of any selected monomials and we denote the set of selected covariates by $\mathbf{X}^{\mathbf{S}}$.

In the second stage, we estimate the nonlinear decision function under a framework similar to that proposed by Mangasarian and Wild (2007), except that the diagonal matrix $\mathbf{D}$ is fixed (based on variable selection results from the first stage) to be $\mathbf{D}_{(\mathbf{X}^{\mathbf{S}})}$, which is a $p \times p$ diagonal matrix with zeros or ones on its diagonal, where ones correspond to covariates in $\mathbf{X}^{\mathbf{S}}$ and zeros correspond to covariates not in $\mathbf{X}^{\mathbf{S}}$. In this way, the optimisation problem becomes a linear programming one. Specifically, we solve the following problem:

$$\begin{aligned} &\min \ \sum_{i=1}^{n^{S_k}} |\nu_i| + C \sum_{i=1}^{n^{S_k}} \frac{\xi_i |e_i|}{\pi_i^{S_k}}, \\ &\text{s.t.} \ \begin{cases} A_i^{S_k} \text{sign}(e_i)\{\beta_0 + \mathcal{K}_\sigma^{\text{RBF}}(\mathbf{D}_{(\mathbf{X}^{\mathbf{S}})}\mathbf{X}_i, \mathbf{D}_{(\mathbf{X}^{\mathbf{S}})}\mathbf{X})\boldsymbol{\nu}\} \geq 1 - \xi_i, & i = 1, \ldots, n^{S_k}, \\ \xi_i \geq 0, & i = 1, \ldots, n^{S_k}, \end{cases} \end{aligned} \tag{3.24}$$

where ξ_i are slack variables, C is the cost parameter, $\mathcal{K}_\sigma^{\text{RBF}}(\bullet, \bullet)$ denotes the Gaussian RBF kernel function, and $\boldsymbol{\nu}$ is the coefficient vector associated with the kernel. Similar to Section 3.2.4.1, we replace ν_i and $|\nu_i|$ with $r_i - s_i$ and $r_i + s_i$,

respectively, where $r_i, s_i \geq 0$ for $i = 1, \ldots, n^{S_k}$, and solve

$$\min_{(\boldsymbol{r},\boldsymbol{s},\beta_0,\boldsymbol{\xi})} \sum_{i=1}^{n^{S_k}}(r_i + s_i) + C\sum_{i=1}^{n^{S_k}} \frac{\xi_i|e_i|}{\pi_i^{S_k}},$$

$$\text{s.t.} \begin{cases} \xi_i \geq 1 - A_i^{S_k}\text{sign}(e_i)\{\beta_0 + \mathcal{K}_\sigma^{\text{RBF}}(\mathbf{D}_{(\mathbf{X}^{\mathbf{s}})}\mathbf{X}_i, \mathbf{D}_{(\mathbf{X}^{\mathbf{s}})}\mathbf{X})(\boldsymbol{r} - \boldsymbol{s})\}, & i = 1, \ldots, n^{S_k}, \\ \xi_i \geq 0, & i = 1, \ldots, n^{S_k}, \\ r_i \geq 0, & i = 1, \ldots, n^{S_k}, \\ s_i \geq 0, & i = 1, \ldots, n^{S_k}. \end{cases}$$

We can reformulate this linear programming problem according to (3.23) and then solve it using `linprog` in Matlab. The variable that we aim to solve is $\boldsymbol{u} = \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{s} \\ \beta_0 \\ \boldsymbol{\xi} \end{bmatrix} = [r_1, \ldots, r_{n^{S_k}}, s_1, \ldots, s_{n^{S_k}}, \beta_0, \xi_1, \ldots, \xi_{n^{S_k}}]^\intercal$. The coefficient vector associated with $\boldsymbol{u}$ is $f = [\mathbf{1}_{n^{S_k}}, \mathbf{1}_{n^{S_k}}, 0, C|\boldsymbol{e}|/\boldsymbol{\pi}^{S_k}]^\intercal$. There is no upper bound for $\boldsymbol{u}$, so $\boldsymbol{g}_u = [\,]$. The lower bound $\boldsymbol{g}_l = [\mathbf{0}_{n^{S_k}}, \mathbf{0}_{n^{S_k}}, -\infty, \mathbf{0}_{n^{S_k}}]^\intercal$. In addition, there are no equality constraints, so $\mathbf{E} = [\,]$ and $\boldsymbol{d} = [\,]$. For this problem, the dimension of matrix $\mathbf{C}$ is $n^{S_k} \times (3n^{S_k} + 1)$ and $\mathbf{C}$ can be written as $\mathbf{C} = [\underbrace{-\text{diag}\{\boldsymbol{A}^{S_k}\text{sign}(\boldsymbol{e})\}\mathcal{K}_\sigma^{\text{RBF}}(\mathbf{D}_{(\mathbf{X}^{\mathbf{s}})}\mathbf{X}, \mathbf{D}_{(\mathbf{X}^{\mathbf{s}})}\mathbf{X})}_{n^{S_k}\times n^{S_k}}, \underbrace{\text{diag}\{\boldsymbol{A}^{S_k}\text{sign}(\boldsymbol{e})\}\mathcal{K}_\sigma^{\text{RBF}}(\mathbf{D}_{(\mathbf{X}^{\mathbf{s}})}\mathbf{X}, \mathbf{D}_{(\mathbf{X}^{\mathbf{s}})}\mathbf{X})}_{n^{S_k}\times n^{S_k}}, \underbrace{-\boldsymbol{A}^{S_k}\text{sign}(\boldsymbol{e})}_{n^{S_k}\times 1}, \underbrace{-\mathbf{I}_{n^{S_k}}}_{n^{S_k}\times n^{S_k}}]$. The upper bound vector of linear constraints is $\boldsymbol{b} = -\mathbf{1}_{n^{S_k}}^\intercal$. The solution to this optimisation problem is $\widehat{\boldsymbol{u}} = \begin{bmatrix} \widehat{\boldsymbol{r}} \\ \widehat{\boldsymbol{s}} \\ \widehat{\beta_0} \\ \widehat{\boldsymbol{\xi}} \end{bmatrix}$. We estimate the optimal decision function as $\widehat{f^*}^{S_k}(\mathbf{X}) = \widehat{\beta_0} + \sum_{i=1}^{n^{S_k}} \mathcal{K}_\sigma^{\text{RBF}}(\mathbf{D}_{(\mathbf{X}^{\mathbf{s}})}\mathbf{X}, \mathbf{D}_{(\mathbf{X}^{\mathbf{s}})}\mathbf{X}_i)(\widehat{r_i} - \widehat{s_i})$, and the optimal ITR for S_k is given by $\widehat{\mathcal{D}^*}^{S_k}(\mathbf{X}) = \text{sign}\{\widehat{f^*}^{S_k}(\mathbf{X})\}$.

We note that even though the prescreening stage focuses only on first-order and second-order terms and may fail to capture more complicated nonlinear structures, in most medical applications, covariates that interact with the treatment decision in a more complex fashion lead to harder-to-interpret rules and are less useful. In addition, the relationship between covariates selected in the first stage and the final optimal decision rule is reevaluated without the second-order restriction by training the nonlinear classifier using the Gaussian RBF kernel in the second stage. Therefore, SODA should be sufficient for the purpose of covariate screening. We also demonstrate through simulation studies in the next section that in most cases, this prescreening method suffices for identifying covariates that inform decision-making, and it performs well even when true underlying decision boundaries involve nonlinear terms other than the second-order ones.

We also comment that although the variable selection in the first stage and the training of nonlinear classifiers in the second stage seem to be performed independently, we consider the proposed two-stage procedure as a wrapper method rather than a filter method for variable selection (Guyon and Elisseeff, 2003; Jović et al., 2015; Kohavi and John, 1997; Saeys et al., 2007). This is because we take into consideration the effects of selected covariates on classification performance by wrapping variable selection around the logistic regression framework. Indeed, the two stages are tightly coupled since they both seek to optimise classification performance, and the inclusion of the first stage adds robustness to boundary estimations in the second stage despite the fact that different classification algorithms are implemented for each stage. As pointed out by Bi et al. (2003), another major advantage of this type of two-stage procedure for variable selection is that the relevance of covariates in classification can be assessed in a computationally cheaper way compared to the method that directly wraps variable selection around a nonlinear SVM classifier.

3.3 Simulation studies

We conduct extensive simulation studies to assess the finite sample performance of the proposed sequential re-estimation learning method in both low-dimensional (without noise covariates and all covariates inform optimal treatment decisions) and moderate-dimensional (with noise covariates) settings. The sample Matlab codes are provided in Appendix B.2.

3.3.1 Simulation design and evaluation criteria

In each simulation setting, covariates $X_1, X_2, \ldots, X_p$ are independently generated from the uniform distribution $\mathcal{U}\{-1, 1\}$, and treatment A is sampled uniformly from $\{1, 2, \ldots, K\}$ such that $P(A = a|\,\mathbf{X} = \mathbf{x}) = 1/K$ for all $\mathbf{x} \in \mathcal{X}$ and $a \in \mathcal{A}$. Similar to Chen et al. (2016), we assume that the outcome Y is normally distributed with mean $\mu(\mathbf{X}) - \varphi\{A, \mathcal{D}^*(\mathbf{X})\}$ and variance 1, where $\mu(\mathbf{X})$ is the main effect of covariates on the outcome and $\varphi\{A, \mathcal{D}^*(\mathbf{X})\}$ represents the loss in outcome when the assigned treatment is A and the true optimal treatment is $\mathcal{D}^*(\mathbf{X})$.

We examine 10 settings, which cover scenarios with different numbers of ordinal treatment options, different loss functions for receiving nonoptimal treatments, and a broad set of decision boundaries, some of which have rarely been explored previously in the literature. For each setting, we consider two training sample sizes: $n = 400$ and $n = 800$, and we repeat the simulation 500 times.

Settings 1-6 mimic situations where decision boundaries between two "consecutive" treatment categories are parallel to each other and are determined by a combination of X_1, X_2, X_3, X_4, and X_5. Decision boundaries are linear in settings 1-5 and nonlinear in setting 6. Similar to dose-finding problems (Chen et al., 2016), when treatments are ordinal, we would expect that incorrectly allocating a patient to a treatment that is closer to his/her true optimal leads to a smaller loss in the outcome. We consider two types of losses with such property: the absolute loss, $\varphi\{A, \mathcal{D}^*(\mathbf{X})\} = |A - \mathcal{D}^*(\mathbf{X})|$, in settings 1-5, and the quadratic loss, $\varphi\{A, \mathcal{D}^*(\mathbf{X})\} = \{A - \mathcal{D}^*(\mathbf{X})\}^2$, in setting 6. In addition, we vary intercepts of decision functions in settings 1-4 (this will affect the distribution of the true optimal ITR. e.g. the proportion of subjects with $\mathcal{D}^*(\mathbf{X}) = 1$ is about 10% in setting 2, but greater than 60% in setting 3) to examine the robustness of the proposed method to different distributions of the true optimal ITR, $\mathcal{D}^*(\mathbf{X})$, across treatments $1, \ldots, K$. Details on simulation designs of settings 1-6 are provided in Table 3.4.

Table 3.4: *Descriptions of simulation settings 1-6. K denotes the total number of treatment options, $\mu(\mathbf{X})$ denotes main covariates effects, $s(\mathbf{X})$ denotes the functional forms of decision boundaries, $\mathcal{D}^*(\mathbf{X})$ denotes the true optimal ITR, and $\varphi\{A, \mathcal{D}^*(\mathbf{X})\}$ denotes the loss in outcome when the assigned treatment is A and the true optimal treatment is $\mathcal{D}^*(\mathbf{X})$.*

Setting	K	$\mu(\mathbf{X})$	$s(\mathbf{X})$	$\mathcal{D}^*(\mathbf{X})$	$\varphi\{A, \mathcal{D}^*(\mathbf{X})\}$
1	3	$5 + X_1 + X_2 + 2X_3 + 0.5X_4$	$0.5X_1 + 2X_2 + X_3 + X_4 + X_5$	$I\{s(\mathbf{X}) < -1\} + 2I\{-1 \leq s(\mathbf{X}) < 1\} + 3I\{s(\mathbf{X}) \geq 1\}$	$4\lvert A - \mathcal{D}^*(\mathbf{X})\rvert$
2	3	$5 + X_1 + X_2 + 2X_3 + 0.5X_4$	$0.5X_1 + 2X_2 + X_3 + X_4 + X_5$	$I\{s(\mathbf{X}) < -2\} + 2I\{-2 \leq s(\mathbf{X}) < -0.5\} + 3I\{s(\mathbf{X}) \geq -0.5\}$	$4\lvert A - \mathcal{D}^*(\mathbf{X})\rvert$
3	3	$5 + X_1 + X_2 + 2X_3 + 0.5X_4$	$0.5X_1 + 2X_2 + X_3 + X_4 + X_5$	$I\{s(\mathbf{X}) < 0.5\} + 2I\{0.5 \leq s(\mathbf{X}) < 2\} + 3I\{s(\mathbf{X}) \geq 2\}$	$4\lvert A - \mathcal{D}^*(\mathbf{X})\rvert$
4	3	$5 + X_1 + X_2 + 2X_3 + 0.5X_4$	$0.5X_1 + 2X_2 + X_3 + X_4 + X_5$	$I\{s(\mathbf{X}) < -0.5\} + 2I\{-0.5 \leq s(\mathbf{X}) < 0.5\} + 3I\{s(\mathbf{X}) \geq 0.5\}$	$4\lvert A - \mathcal{D}^*(\mathbf{X})\rvert$
5	4	$5 + X_1 + X_2 + 2X_3 + 0.5X_4$	$0.5X_1 + 2X_2 + X_3 + X_4 + X_5$	$I\{s(\mathbf{X}) < -1\} + 2I\{-1 \leq s(\mathbf{X}) < 0\} + 3I\{0 \leq s(\mathbf{X}) < 1\} + 4I\{s(\mathbf{X}) \geq 1\}$	$4\lvert A - \mathcal{D}^*(\mathbf{X})\rvert$
6	3	$5 + X_1^2 + X_2^2 + 2X_3^2 + 0.5X_4^2$	$-(0.5X_1 + 2X_2)^2 + \exp(X_3) + \sin^{-1}(X_4) + X_5^3$	$I\{s(\mathbf{X}) < -1\} + 2I\{-1 \leq s(\mathbf{X}) < 1\} + 3I\{s(\mathbf{X}) \geq 1\}$	$4\{A - \mathcal{D}^*(\mathbf{X})\}^2$

Settings 7-10 correspond to situations where decision boundaries between "consecutive" treatment categories are no longer shifts of each other and thus nonparallel. To facilitate the visualisation of these settings, we take $p = 2$ and assume that only X_1 and X_2 inform the optimal treatment choice. The main effect $\mu(\mathbf{X})$ is fixed to be $5 + X_1^2 + X_2^2$, and the loss is $\varphi\{A, \mathcal{D}^*(\mathbf{X})\} = 4\{A - \mathcal{D}^*(\mathbf{X})\}^2$. In the following, we list the number of treatment options, K, and the true optimal ITR, $\mathcal{D}^*(\mathbf{X})$, that are considered in each setting:

- Setting 7:
  - $K = 3$
  - $s_1(\mathbf{X}) = (X_1 + 1)^2 + (X_2 - 1)^2$
  - $s_2(\mathbf{X}) = X_1^2 + X_2 + X_1 X_2$
  - $\mathcal{D}^*(\mathbf{X}) = \begin{cases} 1, & \text{if } s_1(\mathbf{X}) < 1, \\ 2, & \text{if } s_1(\mathbf{X}) \geq 1 \text{ and } s_2(\mathbf{X}) > 0, \\ 3, & \text{otherwise.} \end{cases}$
- Setting 8:
  - $K = 3$
  - $s(\mathbf{X}) = X_1^2 + X_2^2$
  - $\mathcal{D}^*(\mathbf{X}) = \begin{cases} 1, & \text{if } s(\mathbf{X}) < \frac{1}{4}, \\ 2, & \text{if } \frac{1}{4} \leq s(\mathbf{X}) < \frac{9}{16}, \\ 3, & \text{otherwise.} \end{cases}$
- Setting 9:
  - $K = 3$
  - $s_1(\mathbf{X}) = I(|X_1| < \frac{1}{2}) I(|X_2| < \frac{1}{2})$
  - $s_2(\mathbf{X}) = I\left\{ \frac{X_1^2}{(7/8)^2} + \frac{X_2^2}{(21/32)^2} < 1 \right\}$
  - $\mathcal{D}^*(\mathbf{X}) = \begin{cases} 1, & \text{if } s_1(\mathbf{X}) = 1, \\ 2, & \text{if } s_1(\mathbf{X}) \neq 1 \text{ and } s_2(\mathbf{X}) = 1, \\ 3, & \text{otherwise.} \end{cases}$

- Setting 10:
  - $K = 4$
  - $s(\mathbf{X}) = X_1^2 + X_2^2$
  - $\mathcal{D}^*(\mathbf{X}) = \begin{cases} 1, & \text{if } s(\mathbf{X}) < \frac{1}{4}, \\ 2, & \text{if } \frac{1}{4} \leq s(\mathbf{X}) < \frac{9}{16}, \\ 3, & \text{if } \frac{9}{16} \leq s(\mathbf{X}) < \frac{121}{144}, \\ 4, & \text{otherwise.} \end{cases}$

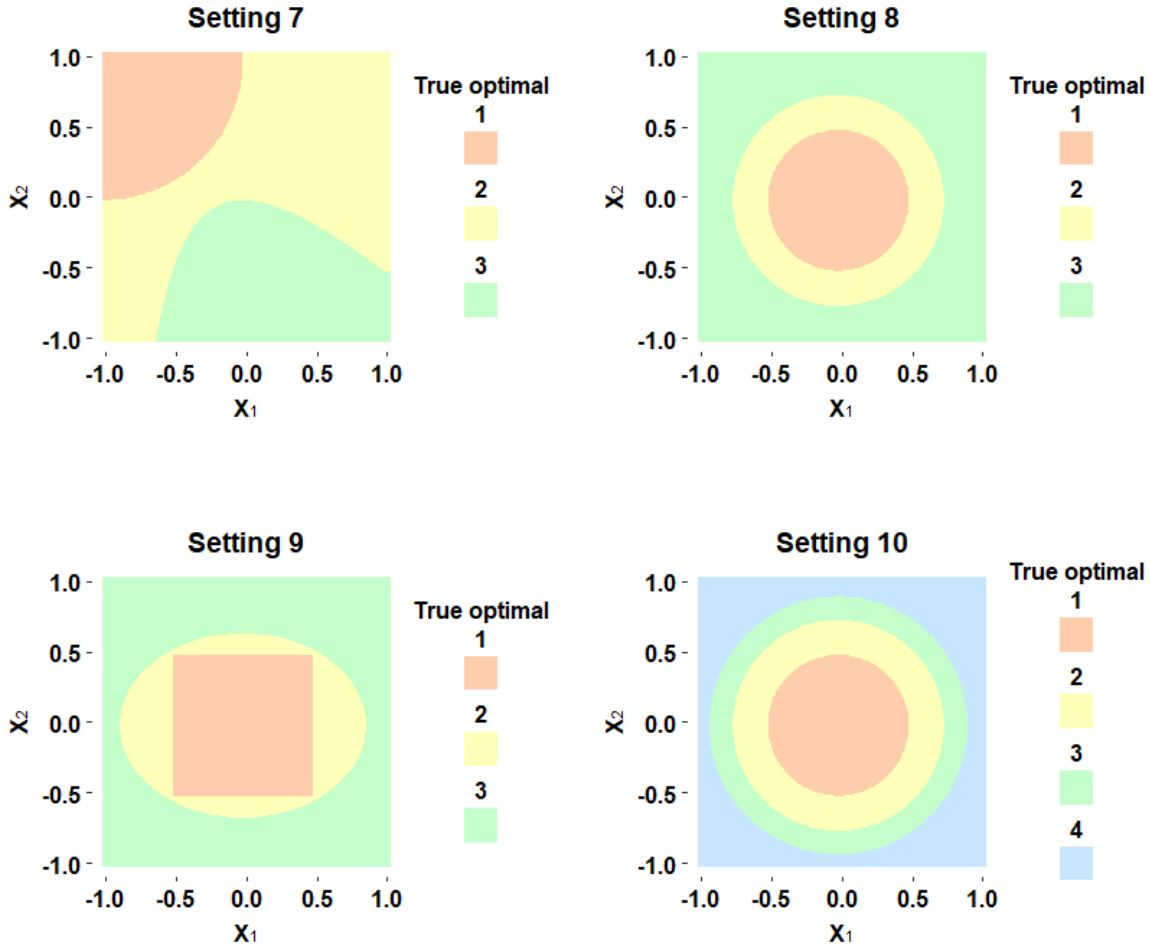


Fig. 3.4: *Visualisation of true underlying decision boundaries and optimal ITRs in simulation settings with nonparallel boundaries (7-10).*

True optimal ITRs and decision boundaries in settings 7-10 are illustrated in Figure 3.4. Specifically, in setting 7, the two decision boundaries are quarter circle and parabola. For settings 8 and 10, we consider circle boundaries with expanding radius ($K = 3$ in setting 8 and $K = 4$ in setting 10). Setting 9 examines the case where one decision boundary is smooth (ellipse) and the other is nonsmooth (square).

For settings with parallel boundaries (1-6), we apply SR learning using both linear (SR-Linear) and Gaussian (SR-Gaussian) kernels to train binary classifiers S_k and R_k, while for settings with nonparallel boundaries (7-10), we use the more flexible Gaussian kernel to better train classifiers. In all settings, we compare SR learning with existing methods that can be used to estimate the optimal ITR in multi-arm trials and incorporate variable selection features, including the multi-category direct learning method (D-Learn) proposed by Qi and Liu (2018), the l_1-penalised least squares method (PLS) developed by Qian and Murphy (2011), the adaptive contrast weighted learning method introduced by Tao and Wang (2017) with the minimum contrasts (ACWL-C1) and the maximum contrasts (ACWL-C2). In particular, for D-Learn, we estimate linear decision rules with LASSO when true boundaries are linear; and estimate nonlinear decision rules with component selection and smoothing operator (COSSO) when true boundaries are nonlinear (Qi and Liu, 2018). For PLS, we use the basis function set $(1, \mathbf{X}, A, \mathbf{X}A)$ when true underlying boundaries are linear and basis $(1, \mathbf{X}, \mathbf{X}^2, A, \mathbf{X}A, \mathbf{X}^2A)$ when true boundaries are nonlinear. All tuning parameters are selected via 5-fold cross-validation.

To assess the performance of our proposed variable selection methods in situations with noise covariates, we employ the same data-generating process as before and repeat simulations in all settings with p now set to be 50. That means, there are 45 covariates that are unrelated to decision functions for settings 1-6, and 48 irrelevant covariates for settings 7-10. In these moderate-dimensional scenarios, we apply SR learning both with and without variable selection using the kernel that reflects the true boundary types (linear or nonlinear). We still compare our proposed method with D-Learn, PLS, ACWL-C1, and ACWL-C2, since all these methods inherently perform variable selection and thus are directly applicable to cases with moderate-dimensional covariates.

We evaluate the performance of each method on a large independent testing dataset of size 10000 using two criteria: (i) the misclassification rate of the estimated optimal ITR compared to the true optimal ITR, i.e. $\mathbb{P}_{n_{\text{test}}} I\{\widehat{\mathcal{D}^*}(\mathbf{X}) \neq \mathcal{D}^*(\mathbf{X})\}$, where $\mathbb{P}_{n_{\text{test}}}$ denotes the empirical average over the testing dataset,

and (ii) the estimated value under $\widehat{\mathcal{D}^*}(\mathbf{X})$ on the testing dataset, i.e. $\widehat{V}(\widehat{\mathcal{D}^*}) = \mathbb{P}_{n_{\text{test}}}[YI\{A = \widehat{\mathcal{D}^*}(\mathbf{X})\}/P(A|\,\mathbf{X})]/\mathbb{P}_{n_{\text{test}}}[I\{A = \widehat{\mathcal{D}^*}(\mathbf{X})\}/P(A|\,\mathbf{X})]$. In the simulation settings that we examine, $P(A|\,\mathbf{X})$ is constant. Therefore, estimated value can be interpreted as the average outcome of testing samples whose observed treatments are the same as the estimated optimal ones.

3.3.2 Simulation results

3.3.2.1 Low-dimensional X (without noise covariates)

In this section, we show results from simulation studies when all components of $\mathbf{X}$ play a role in determining the optimal ITR.

Parallel settings Empirical misclassification rates and values corresponding to parallel settings 1-6 in the low-dimensional case are presented in Figure 3.5 and Table 3.5. For all these settings, our method performs the best in that it leads to the smallest misclassification rate and the largest value. Among the other competing methods, PLS seems to be slightly better than D-Learn, and much better than tree-based methods ACWL-C1 and ACWL-C2. In settings 1-5, decision boundaries are linear, and SR-Gaussian performs slightly worse than SR-Linear due to the flexibility of the Gaussian RBF kernel (i.e. "over-parameterised" for linear decision rules). In contrast, when true decision boundaries are nonlinear (setting 6), SR-Gaussian outperforms SR-Linear to a large extent as SR-Linear is misspecified in this case. Since we optimise the performance of PLS and D-Learn by modelling the nonlinearity in setting 6, they both enjoy some advantages over the misspecified SR-Linear. In addition, even though our method involves sequential steps and we expect the "effective" sample size for training each binary classifier to vary when the distribution of the true optimal ITR changes, results from settings 1-4 demonstrate that our method is robust to such variation and performs similarly well across scenarios where proportions of subjects whose $\mathcal{D}^*(\mathbf{X}) = 1$ are very different. Expectedly, as the sample size increases, misclassification rates decrease, value function estimates get closer to true optimal values (blue dashed lines in Figure 3.5), and standard deviation estimates of both criteria get smaller.

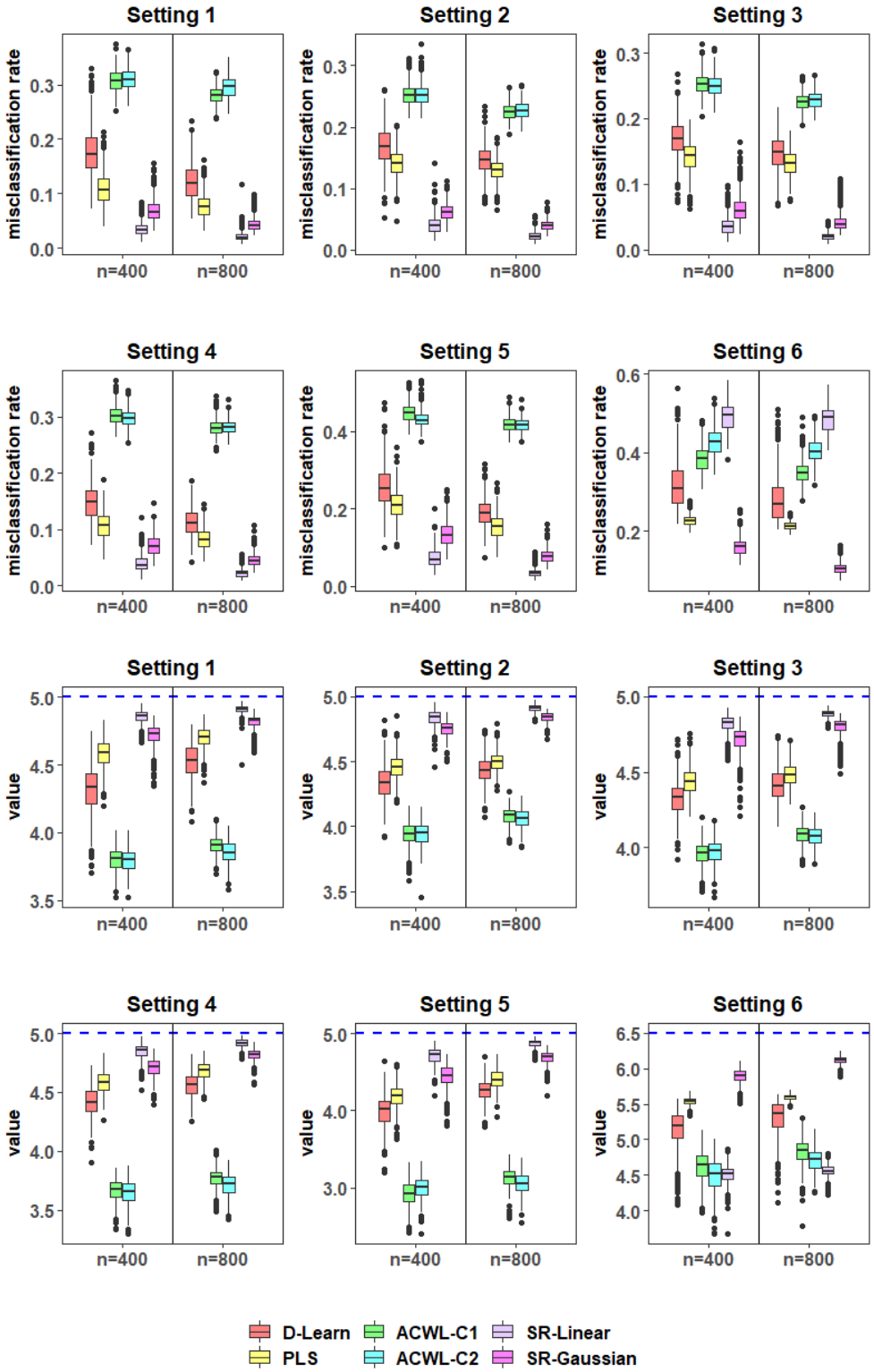


Fig. 3.5: *Boxplots of empirical misclassification rates (top 2 rows) and value functions (bottom 2 rows) based on 500 replicates for settings with parallel boundaries (1-6) and no noise covariates. Blue dashed lines represent the true optimal value corresponding to each setting.*

Table 3.5: *Simulation results based on 500 replicates: mean (SD) of misclassification rates and value functions for settings with parallel boundaries (1-6) and no noise covariates. The smallest misclassification rate and largest value function for each setting are in bold.*

		$n = 400$		$n = 800$	
Setting	Method	Misclassification	Value	Misclassification	Value
1	D-Learn	0.18 (0.04)	4.32 (0.17)	0.12 (0.03)	4.53 (0.13)
	PLS	0.11 (0.03)	4.58 (0.11)	0.08 (0.02)	4.70 (0.08)
	ACWL-C1	0.31 (0.02)	3.80 (0.09)	0.28 (0.02)	3.91 (0.06)
	ACWL-C2	0.31 (0.02)	3.79 (0.08)	0.30 (0.02)	3.85 (0.08)
	SR-Linear	**0.03 (0.01)**	**4.85 (0.04)**	**0.02 (0.01)**	**4.91 (0.03)**
	SR-Gaussian	0.07 (0.02)	4.72 (0.08)	0.04 (0.01)	4.82 (0.05)
2	D-Learn	0.17 (0.03)	4.34 (0.13)	0.15 (0.02)	4.44 (0.10)
	PLS	0.14 (0.02)	4.46 (0.09)	0.13 (0.02)	4.50 (0.07)
	ACWL-C1	0.25 (0.02)	3.94 (0.09)	0.22 (0.01)	4.08 (0.06)
	ACWL-C2	0.25 (0.02)	3.94 (0.09)	0.23 (0.01)	4.06 (0.07)
	SR-Linear	**0.04 (0.01)**	**4.83 (0.06)**	**0.02 (0.01)**	**4.91 (0.03)**
	SR-Gaussian	0.06 (0.01)	4.75 (0.06)	0.04 (0.01)	4.84 (0.04)
3	D-Learn	0.17 (0.03)	4.33 (0.12)	0.15 (0.03)	4.41 (0.11)
	PLS	0.14 (0.02)	4.44 (0.10)	0.13 (0.02)	4.48 (0.08)
	ACWL-C1	0.25 (0.02)	3.96 (0.08)	0.23 (0.01)	4.09 (0.06)
	ACWL-C2	0.25 (0.02)	3.97 (0.08)	0.23 (0.01)	4.08 (0.06)
	SR-Linear	**0.04 (0.01)**	**4.82 (0.06)**	**0.02 (0.01)**	**4.89 (0.02)**
	SR-Gaussian	0.06 (0.02)	4.72 (0.08)	0.04 (0.01)	4.80 (0.05)
4	D-Learn	0.15 (0.03)	4.42 (0.13)	0.11 (0.02)	4.56 (0.10)
	PLS	0.11 (0.02)	4.58 (0.09)	0.08 (0.02)	4.68 (0.07)
	ACWL-C1	0.30 (0.02)	3.67 (0.09)	0.28 (0.01)	3.77 (0.08)
	ACWL-C2	0.30 (0.02)	3.65 (0.11)	0.28 (0.01)	3.71 (0.09)
	SR-Linear	**0.04 (0.01)**	**4.85 (0.06)**	**0.02 (0.01)**	**4.92 (0.03)**
	SR-Gaussian	0.07 (0.02)	4.71 (0.08)	0.05 (0.01)	4.82 (0.05)
5	D-Learn	0.26 (0.05)	3.99 (0.20)	0.19 (0.04)	4.27 (0.14)
	PLS	0.21 (0.04)	4.18 (0.15)	0.15 (0.03)	4.40 (0.12)
	ACWL-C1	0.45 (0.02)	2.92 (0.16)	0.42 (0.02)	3.13 (0.12)
	ACWL-C2	0.43 (0.02)	3.00 (0.14)	0.42 (0.02)	3.05 (0.14)
	SR-Linear	**0.07 (0.02)**	**4.71 (0.10)**	**0.03 (0.01)**	**4.87 (0.04)**
	SR-Gaussian	0.13 (0.03)	4.45 (0.15)	0.08 (0.02)	4.68 (0.08)
6	D-Learn	0.32 (0.06)	5.14 (0.27)	0.28 (0.06)	5.30 (0.25)
	PLS	0.23 (0.01)	5.54 (0.06)	0.21 (0.01)	5.60 (0.04)
	ACWL-C1	0.38 (0.03)	4.63 (0.22)	0.35 (0.03)	4.83 (0.18)
	ACWL-C2	0.43 (0.03)	4.50 (0.23)	0.41 (0.03)	4.72 (0.16)
	SR-Linear	0.49 (0.04)	4.52 (0.12)	0.49 (0.03)	4.56 (0.09)
	SR-Gaussian	**0.16 (0.02)**	**5.89 (0.10)**	**0.11 (0.01)**	**6.11 (0.06)**

Nonparallel settings Table 3.6 and Figure 3.6 summarise results corresponding to nonparallel settings 7-10 in the low-dimensional case based on 500 replications.

Table 3.6: *Simulation results based on 500 replicates: mean (SD) of misclassification rates and value functions for settings with nonparallel boundaries (7-10) and no noise covariates. The smallest misclassification rate and largest value function for each setting are in bold.*

		$n = 400$		$n = 800$	
Setting	Method	Misclassification	Value	Misclassification	Value
7	D-Learn	0.14 (0.05)	5.12 (0.21)	0.11 (0.03)	5.23 (0.12)
	PLS	0.14 (0.03)	5.11 (0.10)	0.14 (0.02)	5.13 (0.07)
	ACWL-C1	0.16 (0.04)	4.92 (0.21)	0.13 (0.03)	5.11 (0.16)
	ACWL-C2	0.14 (0.04)	5.09 (0.15)	0.12 (0.03)	5.19 (0.11)
	SR-Gaussian	**0.03 (0.01)**	**5.51 (0.05)**	**0.02 (0.01)**	**5.56 (0.03)**
8	D-Learn	0.25 (0.07)	4.64 (0.30)	0.24 (0.06)	4.70 (0.24)
	PLS	0.30 (0.05)	4.44 (0.22)	0.29 (0.04)	4.48 (0.17)
	ACWL-C1	0.24 (0.04)	4.53 (0.28)	0.21 (0.03)	4.68 (0.24)
	ACWL-C2	0.52 (0.07)	3.52 (0.32)	0.61 (0.07)	3.19 (0.28)
	SR-Gaussian	**0.05 (0.01)**	**5.47 (0.06)**	**0.03 (0.01)**	**5.55 (0.03)**
9	D-Learn	0.26 (0.08)	4.64 (0.34)	0.24 (0.07)	4.73 (0.28)
	PLS	0.33 (0.06)	4.37 (0.24)	0.31 (0.04)	4.46 (0.14)
	ACWL-C1	0.22 (0.05)	4.60 (0.29)	0.18 (0.03)	4.81 (0.26)
	ACWL-C2	0.57 (0.07)	3.34 (0.29)	0.69 (0.07)	2.93 (0.28)
	SR-Gaussian	**0.08 (0.01)**	**5.35 (0.05)**	**0.06 (0.01)**	**5.40 (0.03)**
10	D-Learn	0.40 (0.09)	3.83 (0.58)	0.32 (0.08)	4.28 (0.45)
	PLS	0.26 (0.07)	4.63 (0.28)	0.23 (0.05)	4.76 (0.20)
	ACWL-C1	0.44 (0.04)	3.30 (0.37)	0.43 (0.03)	3.18 (0.34)
	ACWL-C2	0.70 (0.05)	0.90 (0.74)	0.77 (0.03)	0.09 (0.50)
	SR-Gaussian	**0.09 (0.02)**	**5.25 (0.09)**	**0.05 (0.01)**	**5.40 (0.06)**

Our proposed SR learning produces the smallest misclassification rates and largest value functions compared to other methods in all settings. Even in the fairly complicated case where the nonsmooth boundary between treatments 1 and 2, and the smooth boundary between treatments 2 and 3 are almost "connected" at the square corner (setting 9), the proposed method still has superior performance and manages to correctly estimate the optimal ITR for more than 90% of subjects in the testing dataset on average. SR learning also shows the greatest stability in the sense of yielding much smaller standard deviation estimates of misclassification rates and value functions than other methods. Interestingly, we observe that ACWL-C1 and ACWL-C2 perform substantially different in settings 8-10 where decision boundaries "expand" and are nested by nature. Under these scenarios, ACWL-C1 that relies on the minimum contrasts performs much better than ACWL-C2 that uses the maximum contrasts (Tao and Wang, 2017).

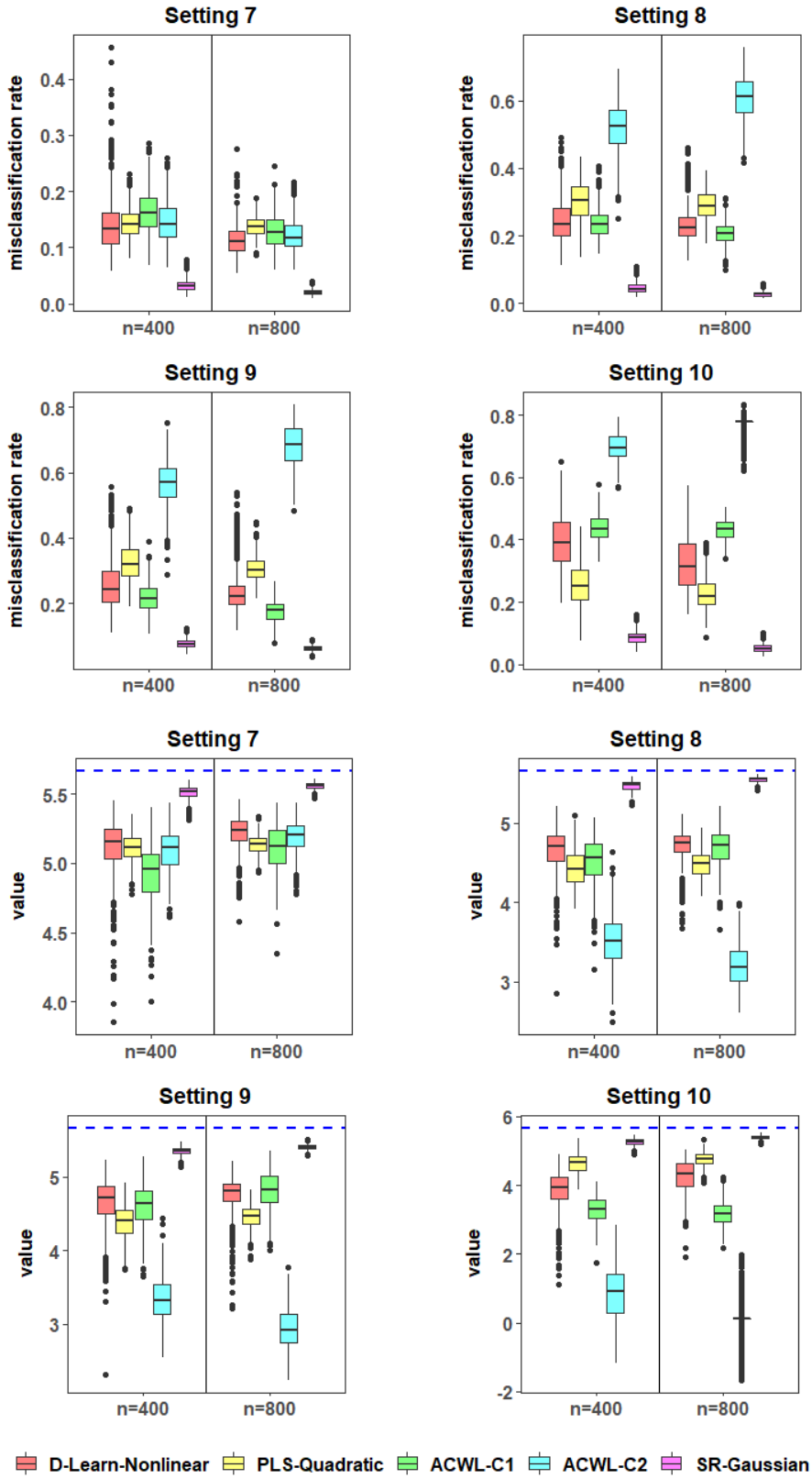


Fig. 3.6: *Boxplots of empirical misclassification rates (top 2 rows) and value functions (bottom 2 rows) based on 500 replicates for settings with nonparallel boundaries (7-10) and no noise covariates. Blue dashed lines represent the true optimal value corresponding to each setting.*

Remark on the computational time of SR learning The running time of SR learning with different sample sizes and kernels is measured on a Windows-based computing system with 1 core, 3.40 GHz Intel processor. For example, in setting 1 (3 ordinal arms), the computational time per run for $n = 400$ linear kernel, $n = 400$ Gaussian kernel, $n = 800$ linear kernel, and $n = 800$ Gaussian kernel is 7 seconds, 13 seconds, 16 seconds, and 35 seconds, respectively. In setting 5 (4 ordinal arms), the computational time per run for $n = 400$ linear kernel, $n = 400$ Gaussian kernel, $n = 800$ linear kernel, and $n = 800$ Gaussian kernel is 12 seconds, 22 seconds, 26 seconds, and 50 seconds, respectively.

Remark on misclassification results We measure classification performance through the misclassification rate, which treats all misclassified cases equally. In the ordinal setting, we are also concerned about the mean absolute error (Baccianella et al., 2009), or a more direct measure, the proportion of individuals whose optimal ITRs are misclassified by more than one treatment category, i.e. $\mathbb{P}_{n_{\text{test}}} I\{|\widehat{\mathcal{D}^*}(\mathbf{X}) - \mathcal{D}^*(\mathbf{X})| > 1\}$. A smaller value of this quantity indicates better performance: if the true optimal treatment for a patient is 1, then treatment 3 would be less preferable than treatment 2, given that treatment 3 is further away from the true optimal, and thus we would like the number of patients who are misclassified by more than one category to be as small as possible. Simulation results (not presented here) imply that by applying the proposed method, the average proportion (over 500 replicates) of subjects whose optimal ITRs are misclassified by more than one treatment category is less than 0.1% for all parallel and nonparallel settings. Both PLS and D-Learn perform comparably well with regard to this criterion, suggesting that even though these methods lead to larger misclassification rates than SR learning, almost all misclassifications "miss" by just one category when the number of ordinal treatment options is 3 or 4.

Remark on the choice of the reference treatment As has been discussed in Section 3.2.1, we assume that treatment 1 is the reference/control arm in SR learning. For ordinal treatments, either the least "intensive" one (treatment 1) or the most "intensive" one (treatment K) can be used as the reference. Simulation results presented above are produced by assuming that the least "intensive" treatment is the control. We also examine the performance of SR learning when the most "intensive" treatment is regarded as the control. In this case, SR learning starts with the comparison between $\{K\}$ and $\{K-1, \ldots, 1\}$ (S_1). A more detailed discussion on how SR learning works when treatment K is

chosen as the control and the corresponding simulation results under settings 1-10 are provided in Appendix B.1.1. As expected, in all settings, SR learning with treatment 1 being the control and SR learning with treatment K being the control lead to almost identical misclassification rates and value functions, which implies that SR learning is robust to whether the least or the most "intensive" treatment is considered as the reference.

3.3.2.2 Moderate-dimensional X (with noise covariates)

In this section, we present simulation results for the case where there are many noise covariates that are irrelevant to estimating the optimal ITR. Results for SR learning with variable selection are obtained by applying variable selection methods proposed in Section 3.2.4.1 and Section 3.2.4.2 for linear and nonlinear decision boundaries, respectively.

Parallel settings Figure 3.7 and Table 3.7 show simulation results of settings 1-6 in the presence of noise covariates ($p = 50$) for $n = 400$ and 800. For reference, we also present the average (across 500 replicates) of misclassification rates and value functions corresponding to the "oracle" case for SR learning (red solid lines in Figure 3.7), where we know exactly which covariates are noise covariates that have no impact on decision boundaries, and we apply SR learning to the simulated data with these noise covariates being excluded (i.e. only signal covariates are included when estimating decision functions).

Our proposed method with variable selection (SR-Select) has competitive performance in settings with noise covariates. As expected, misclassification rates and value functions get closer to those from the "oracle" case as sample size gets larger. When $n = 400$, SR-Select only has marginal gains over PLS and D-Learn in general (and yields slightly worse value functions than D-Learn in setting 5). When the training sample size increases to 800, the advantage of SR-Select becomes more pronounced. Since both D-Learn and PLS perform variable selection while estimating the optimal ITR, they do better than SR learning without variable selection in most settings, especially when $n = 400$. However, under the larger sample size scenario ($n = 800$) for linear settings (1-5), SR learning without variable selection performs similarly well as these two methods which incorporate the variable selection feature. We note that even though tree-based methods ACWL-C1 and ACWL-C2 perform variable selection intrinsically via the node-splitting process, it seems that they may not be able to capture true decision boundaries, nor to select relevant covariates for estimating

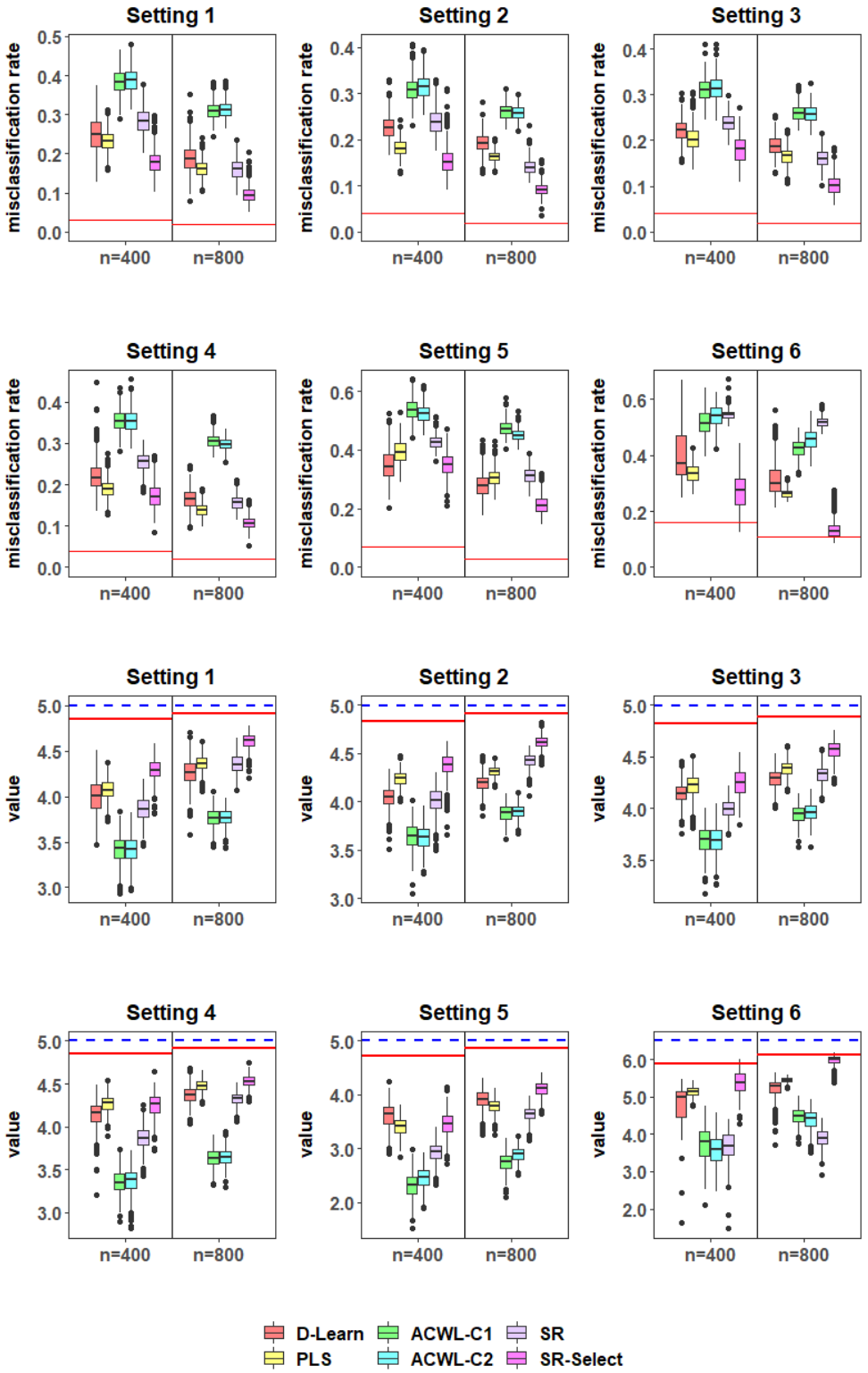


Fig. 3.7: *Boxplots of empirical misclassification rates (top 2 rows) and value functions (bottom 2 rows) based on 500 replicates for parallel settings (1-6) with noise covariates ($p = 50$). Red solid lines represent results corresponding to the "oracle" case for SR learning where only true signal covariates are used to estimate decision boundaries. Blue dashed lines represent the true optimal value of each setting.*

Table 3.7: *Simulation results based on 500 replicates: mean (SD) of misclassification rates and value functions for parallel settings (1-6) with noise covariates ($p = 50$). The smallest misclassification rate and largest value function for each setting are in bold.*

		$n = 400$		$n = 800$	
Setting	Method	Misclassification	Value	Misclassification	Value
1	D-Learn	0.25 (0.05)	4.00 (0.19)	0.19 (0.04)	4.27 (0.15)
	PLS	0.23 (0.03)	4.07 (0.11)	0.16 (0.02)	4.37 (0.09)
	ACWL-C1	0.38 (0.03)	3.42 (0.16)	0.31 (0.02)	3.76 (0.10)
	ACWL-C2	0.39 (0.03)	3.41 (0.15)	0.31 (0.02)	3.76 (0.09)
	SR-Linear	0.28 (0.03)	3.86 (0.13)	0.16 (0.03)	4.36 (0.10)
	SR-Linear-Select	**0.18 (0.03)**	**4.30 (0.12)**	**0.10 (0.02)**	**4.61 (0.08)**
2	D-Learn	0.23 (0.03)	4.05 (0.12)	0.19 (0.02)	4.20 (0.09)
	PLS	0.18 (0.02)	4.24 (0.07)	0.16 (0.01)	4.32 (0.05)
	ACWL-C1	0.31 (0.03)	3.64 (0.14)	0.26 (0.02)	3.88 (0.09)
	ACWL-C2	0.31 (0.03)	3.63 (0.13)	0.26 (0.02)	3.90 (0.08)
	SR-Linear	0.24 (0.03)	4.01 (0.14)	0.14 (0.02)	4.43 (0.07)
	SR-Linear-Select	**0.15 (0.03)**	**4.38 (0.13)**	**0.09 (0.02)**	**4.62 (0.06)**
3	D-Learn	0.22 (0.02)	4.15 (0.10)	0.19 (0.02)	4.29 (0.09)
	PLS	0.20 (0.03)	4.22 (0.11)	0.17 (0.02)	4.39 (0.08)
	ACWL-C1	0.31 (0.02)	3.69 (0.13)	0.26 (0.02)	3.94 (0.08)
	ACWL-C2	0.31 (0.03)	3.69 (0.14)	0.26 (0.02)	3.96 (0.09)
	SR-Linear	0.24 (0.02)	3.99 (0.09)	0.16 (0.02)	4.33 (0.08)
	SR-Linear-Select	**0.18 (0.03)**	**4.25 (0.12)**	**0.10 (0.02)**	**4.57 (0.09)**
4	D-Learn	0.22 (0.04)	4.14 (0.15)	0.17 (0.02)	4.37 (0.10)
	PLS	0.19 (0.02)	**4.27 (0.10)**	0.14 (0.02)	4.48 (0.07)
	ACWL-C1	0.36 (0.02)	3.36 (0.14)	0.31 (0.02)	3.63 (0.11)
	ACWL-C2	0.36 (0.03)	3.37 (0.15)	0.30 (0.01)	3.64 (0.10)
	SR-Linear	0.25 (0.02)	3.87 (0.13)	0.16 (0.02)	4.32 (0.08)
	SR-Linear-Select	**0.17 (0.03)**	4.25 (0.13)	**0.11 (0.02)**	**4.52 (0.07)**
5	D-Learn	**0.35 (0.06)**	**3.61 (0.24)**	0.28 (0.04)	3.91 (0.17)
	PLS	0.39 (0.04)	3.41 (0.16)	0.31 (0.03)	3.78 (0.13)
	ACWL-C1	0.54 (0.03)	2.31 (0.23)	0.48 (0.03)	2.74 (0.17)
	ACWL-C2	0.53 (0.03)	2.45 (0.19)	0.45 (0.02)	2.89 (0.13)
	SR-Linear	0.43 (0.02)	2.92 (0.18)	0.31 (0.03)	3.63 (0.14)
	SR-Linear-Select	**0.35 (0.04)**	3.45 (0.22)	**0.21 (0.03)**	**4.09 (0.14)**
6	D-Learn	0.39 (0.08)	4.83 (0.42)	0.32 (0.07)	5.18 (0.33)
	PLS	0.34 (0.03)	5.14 (0.13)	0.26 (0.01)	5.44 (0.06)
	ACWL-C1	0.52 (0.05)	3.71 (0.47)	0.43 (0.03)	4.48 (0.21)
	ACWL-C2	0.54 (0.04)	3.59 (0.39)	0.46 (0.04)	4.39 (0.26)
	SR-Gaussian	0.54 (0.02)	3.68 (0.39)	0.52 (0.02)	3.91 (0.24)
	SR-Gaussian-Select	**0.27 (0.06)**	**5.38 (0.30)**	**0.14 (0.04)**	**5.96 (0.15)**

the boundaries in these settings. In addition, SR-Select results for setting 6 further confirm that SODA appears to be sufficient for covariate screening in nonlinear settings although it only considers first-order and second-order terms: in this setting, true underlying decision boundaries are nonlinear and involve multiple nonlinear functions of covariates besides second-order polynomials, but SODA still helps reduce the misclassification rate substantially, especially when $n = 800$, in which situation SR-Select approaches the "oracle" case. A possible reason is that the Taylor series expansion of a function up to the second-order terms gives sufficiently good approximations in many cases.

Nonparallel settings In Table 3.8 and Figure 3.8, we report simulation results corresponding to moderate-dimensional $\mathbf{X}$ ($p = 50$) for nonparallel settings. Similar to parallel settings, we also present mean misclassification rates and value functions of the "oracle" case as a reference (red solid lines in Figure 3.8).

Table 3.8: *Simulation results based on 500 replicates: mean (SD) of misclassification rates and value functions for nonparallel settings (7-10) with noise covariates ($p = 50$). The smallest misclassification rate and largest value function for each setting are in bold.*

		$n = 400$		$n = 800$	
Setting	Method	Misclassification	Value	Misclassification	Value
7	D-learn	0.14 (0.04)	5.09 (0.16)	0.11 (0.02)	5.21 (0.08)
	PLS	0.24 (0.03)	4.73 (0.12)	0.19 (0.02)	4.94 (0.07)
	ACWL-C1	0.24 (0.05)	4.57 (0.25)	0.15 (0.03)	4.97 (0.15)
	ACWL-C2	0.25 (0.05)	4.62 (0.22)	0.15 (0.03)	5.04 (0.12)
	SR-Gaussian	0.32 (0.03)	4.30 (0.17)	0.25 (0.01)	4.66 (0.06)
	SR-Gaussian-Select	**0.08 (0.03)**	**5.37 (0.14)**	**0.03 (0.01)**	**5.55 (0.05)**
8	D-Learn	0.48 (0.10)	3.46 (0.73)	0.38 (0.10)	4.08 (0.57)
	PLS	0.36 (0.02)	4.23 (0.10)	0.34 (0.02)	4.30 (0.07)
	ACWL-C1	0.45 (0.07)	3.10 (0.75)	0.31 (0.04)	4.18 (0.24)
	ACWL-C2	0.52 (0.06)	2.58 (0.73)	0.44 (0.04)	3.79 (0.20)
	SR-Gaussian	0.59 (0.14)	2.01 (0.54)	0.59 (0.14)	2.07 (0.54)
	SR-Gaussian-Select	**0.06 (0.03)**	**5.39 (0.12)**	**0.03 (0.01)**	**5.53 (0.04)**
9	D-Learn	0.53 (0.12)	3.18 (0.84)	0.43 (0.11)	3.87 (0.62)
	PLS	0.43 (0.02)	3.92 (0.10)	0.42 (0.02)	3.96 (0.07)
	ACWL-C1	0.45 (0.06)	3.07 (0.74)	0.30 (0.05)	4.21 (0.29)
	ACWL-C2	0.53 (0.07)	2.52 (0.83)	0.47 (0.04)	3.64 (0.22)
	SR-Gaussian	0.64 (0.16)	1.69 (0.79)	0.62 (0.17)	1.66 (0.79)
	SR-Gaussian-Select	**0.11 (0.03)**	**5.20 (0.18)**	**0.07 (0.01)**	**5.36 (0.06)**
10	D-Learn	0.62 (0.07)	1.60 (1.23)	0.52 (0.09)	3.04 (0.88)
	PLS	0.45 (0.03)	3.80 (0.17)	0.40 (0.02)	4.06 (0.09)
	ACWL-C1	0.63 (0.06)	-0.86 (2.05)	0.50 (0.03)	2.76 (0.39)
	ACWL-C2	0.70 (0.04)	-2.40 (1.36)	0.63 (0.04)	1.39 (0.85)
	SR-Gaussian	0.71 (0.04)	-2.57 (1.82)	0.69 (0.04)	-2.82 (1.45)
	SR-Gaussian-Select	**0.22 (0.09)**	**4.61 (0.67)**	**0.07 (0.03)**	**5.34 (0.12)**

Unsurprisingly, the performance of the proposed method without variable selection (SR-Gaussian) is seriously affected by the presence of noise covariates and is worse than the other methods that inherently select important variables when making treatment decisions. However, the two-stage variable selection method introduced in Section 3.2.4.2 (SR-Gaussian-Select) improves SR-Gaussian substantially and achieves much better performance than D-Learn, PLS, ACWL-C1 and ACWL-C2 in all settings: it leads to much smaller misclassification rates and much larger value functions for both $n = 400$ and $n = 800$. In particular, it gets very close to the "oracle" case when $n = 800$.

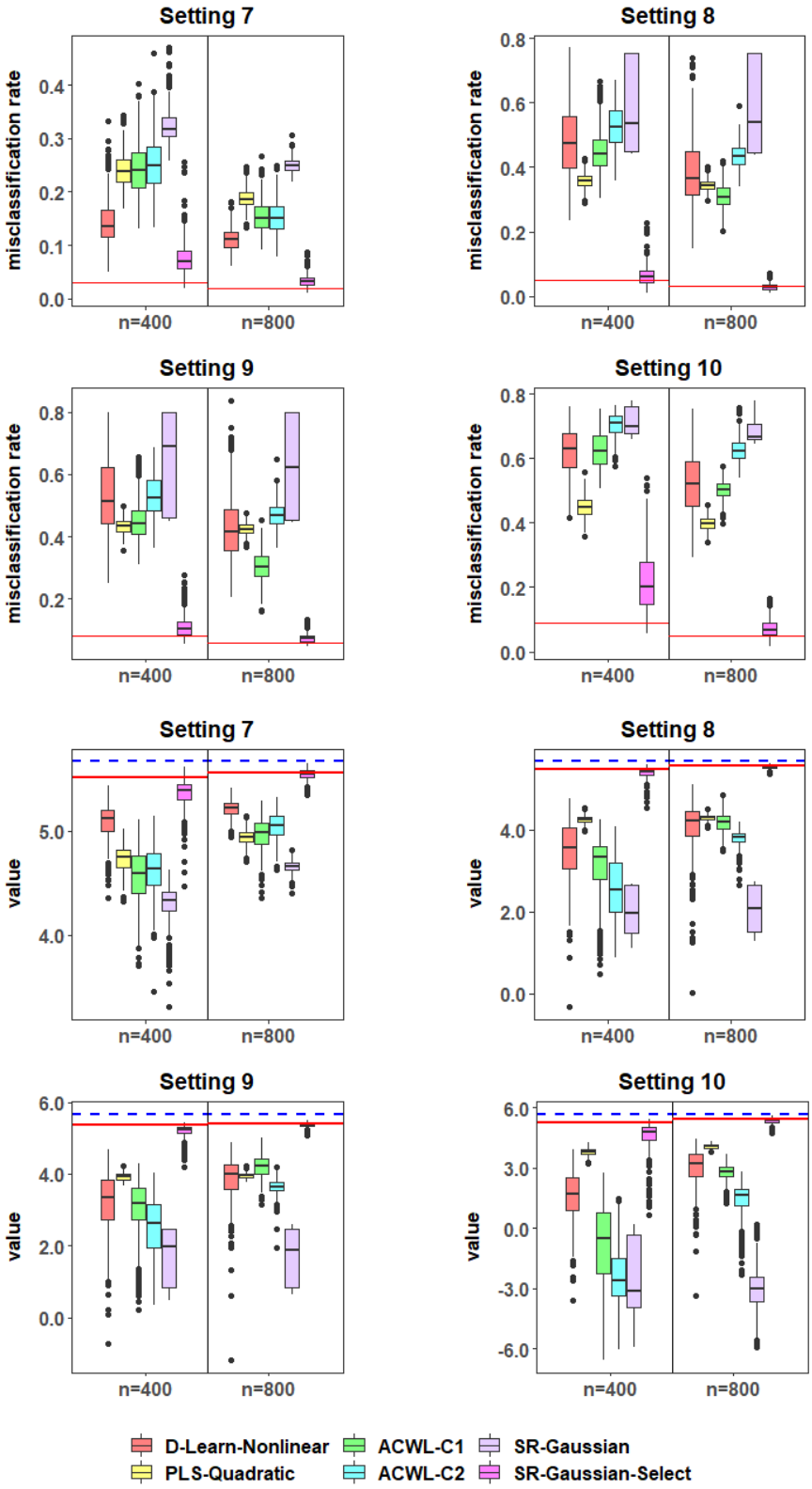


Fig. 3.8: *Boxplots of empirical misclassification rates (top 2 rows) and value functions (bottom 2 rows) based on 500 replicates for nonparallel settings (7-10) with noise covariates ($p = 50$). Red solid lines represent results corresponding to the "oracle" case for SR learning where only true signal covariates are used to estimate decision boundaries. Blue dashed lines represent the true optimal value of each setting.*

Remark on variable selection Additional simulations are performed by selecting important covariates in a "nested" way. For example, when $K = 3$, we still use all components of $\mathbf{X}$ as the inputs for S_1 (compares $\{1\}$ vs. $\{2,3\}$) and S_2 (compares $\{2\}$ vs. $\{3\}$) and perform variable selection separately for S_1 and S_2, since the boundary between treatments 1 and 2 and that between treatments 2 and 3 may depend on different sets of covariates. However, we use the set of covariates selected by S_1 (compares $\{1\}$ vs. $\{2,3\}$) as the inputs for R_1 (compares $\{1\}$ vs. $\{2\}$), given that S_1 and R_1 try to estimate the same boundary. In this way, R_1 might be trained more accurately because it starts with a smaller number of noise covariates (S_1 "prescreens" covariates for R_1). Similar results as in the case of "non-nested" variable selection (Tables 3.7 and 3.8) are observed and thus omitted.

3.3.3 Additional simulation settings

In this section, we consider additional simulation settings to evaluate the validity and robustness of SR learning under some variations of the base settings described in Section 3.3.1. Here, we only provide a brief description of these modified settings. We refer readers to Appendix B.1 for details on simulation designs and corresponding results. To demonstrate the idea, all these additional simulation studies are conducted under the low-dimensional scenario with no noise covariates. We would expect similar conclusions to apply under the moderate-dimensional scenario.

3.3.3.1 "Permuted" orderings of true optimal ITRs

As has been noted in Section 3.2.1, the ordinality of treatments suggests that A has an intrinsic ordering, and if two treatment options are on the same side of the true optimal, $\mathcal{D}^*(\mathbf{X})$, then receiving a treatment that is further away from $\mathcal{D}^*(\mathbf{X})$ will incur a larger loss in the outcome. Simulations in Section 3.3.1 are designed in a way that the relationships between true optimal ITRs and covariates (or covariate combinations) are "ordered"/monotonic. However, this is not a necessary condition for A being ordinal, and treatment orderings do not imply any particular relationship between true optimal ITRs and covariates. To demonstrate this point and evaluate the robustness of SR learning to various underlying orderings of true optimal ITRs with respect to covariates, we conduct additional simulation studies under scenarios where orderings of $\mathcal{D}^*(\mathbf{X})$ are "permuted". For example, we examine settings where the decision boundaries are

the same as those in setting 8 (Figure 3.4), but the true optimal ITRs from the innermost to the outermost no longer follow the order of $1 \to 2 \to 3$. Details on simulation designs (Figure B.1) and simulation results (Table B.2) are given in Appendix B.1.2. We observe that SR learning substantially outperforms all the other methods that ignore the ordinality of treatment and is robust to different ordering relationships between $\mathcal{D}^*(\mathbf{X})$ and covariates.

3.3.3.2 Intersecting decision boundaries

In addition, in Section 3.3.1, we only consider settings where true optimal treatment boundaries do not intersect. For example, from Figure 3.4, we can see that the decision boundary between $\mathcal{D}^*(\mathbf{X}) = 1$ and $\mathcal{D}^*(\mathbf{X}) = 2$, and that between $\mathcal{D}^*(\mathbf{X}) = 2$ and $\mathcal{D}^*(\mathbf{X}) = 3$ do not cross. However, the fact that treatments are ordered does not preclude the possibility of having crossed decision boundaries within the covariate space $\mathcal{X}$ (e.g. tree-type decision rules). In Appendix B.1.3, we present two simulated examples in which true optimal treatment boundaries intersect and we examine the performance of SR learning under these situations. Table B.3 in the Appendix demonstrates the superior performance of SR learning over competing methods in the case with crossed decision boundaries. This is not surprising since the constructions of S- and R-steps in SR learning do not depend on non-crossing boundaries and thus SR learning does not restrict to the situation where decision boundaries do not cross.

3.3.3.3 Violation of the ordinality assumption

According to Section 3.2.1, treatment orderings imply "one-side monotonicity", i.e. $E(Y|\,\mathbf{X} = \mathbf{x}, A = a_1) < E(Y|\,\mathbf{X} = \mathbf{x}, A = a_2)$ if a_1 and a_2 are on the same side of the true optimal treatment with a_1 being further away from the true optimal than a_2. Intuitively, since SR learning takes advantage of the treatment ordering information, we would expect its performance to get worse if "one-side monotonicity" does not hold for some subjects (this can be considered as the "noise" in ordinal settings). Under this circumstance, the advantage of SR learning over multi-arm ITR estimation methods that ignore the ordinality will diminish and may even disappear when the violation percentage ("noise" level) is high. For settings 1-10 considered in Section 3.3.1, all subjects satisfy the "one-side monotonicity" rule. It is of interest to perform a sensitivity analysis by changing the violation percentages and assess the sensitivity of SR learning's performance to a range of "noise" levels. Details about the simulation designs and the results

(Tables B.4 and B.5) of the sensitivity analysis can be found in Appendix B.1.4. To summarise our findings, whether and how much the violation of the "one-side monotonicity" rule affects the performance of SR learning depends on several factors, including the type of violation, the percentage of "rule violators", and the sample size.

3.4 Application to the INTERVAL trial

As has been discussed in Chapter 1, the National Health Service Blood and Transplant (NHSBT) in England is investigating making blood donation safer and more sustainable by developing personalised approaches to donation and tailoring the inter-donation interval to donor's donation capacity. In particular, blood donors with some specific types of characteristics are of great interest to NHSBT. For example, donors with O Rhesus D (RhD) negative blood type are in high demand, since O negative is the "universal" blood type that can be transfused to any patient in need and used in medical emergencies. However, only about 13% of the donor population has O negative blood (NHS Blood and Transplant, 2018). In addition, young female donors are of major concern to the blood service since these donors are more at risk of iron deficiency and anaemia than others in general (Patel et al., 2019).

3.4.1 SR learning for estimating the optimal ITR among ordinal inter-donation intervals

In this section, we apply SR learning to the data from 884 female donors in the INTERVAL trial who were younger than 40 and had O negative blood type (a "much-in-demand but vulnerable" donor population). In INTERVAL, female donors were randomly assigned (1:1:1) to the 16-week (standard), 14-week, and 12-week inter-donation intervals. These three options of the inter-donation interval can be considered as 3 levels of an ordinal "treatment" and the reference arm is the 16-week inter-donation interval since it is the standard clinical practice for female donors and also the safest one among the three options.

Same as in the previous chapter, we consider the total units of blood (a full donation unit contains 470 ml of whole blood) collected by the blood service per donor over the 2-year trial period as the benefit outcome (denoted by G), and the number of deferrals for low Hb per donor during the same period as the risk outcome (denoted by $\widetilde{R}$). When recommending the optimal inter-donation

interval to a blood donor, we should account for the trade-off between the benefit and the risk. Therefore, as before, we construct a utility outcome which "discounts" the total units of blood collected by the incidences of low Hb deferrals as $U = G - b \times \widetilde{R}$, where b is the trade-off parameter reflecting the equivalent benefit loss for one unit increase in the risk, and we seek the personalised donation strategy that maximises the expected value of the utility score U. We note that the trade-off parameter b should be specified based on clinicians' subject-matter knowledge. In the following analysis, we examine two values for b, namely $b = 2$ and $b = 3$, both of which are considered reasonable for the donor population we study (young female donors with universal blood type) by our medical colleagues in the blood service as these values reflect the potential costs of low Hb deferrals incurred by reduced efficiency of collection and reduced donor retention (Hillgrove et al., 2011; Willis et al., 2019). For example, $b = 2$ implies that for one extra low Hb deferral per donor attendance, the equivalent loss in the amount of blood collected by the blood service per donor (due to, for example, potential loss of donors following deferrals) is 2 units (over 2 years).

Fourteen baseline donor characteristics are used for ITR estimation, including age, body mass index (BMI, calculated as weight (kg)/height^2 (m^2)), Short Form Health Survey version 2 (SF-36v2) summary scores (physical component score, mental component score), new or returning donor status, 2-year donation history (i.e. units of whole blood donations in the 2 years before enrolment into the trial), six routine blood measurements (white blood cell count, red blood cell count, haemoglobin level, platelet count, mean corpuscular haemoglobin, mean corpuscular volume), and two blood-based biomarkers (ferritin, transferrin). Following the recommendation by Hsu et al. (2016), we linearly scale each covariate in the training dataset to the range $[-1, 1]$ before applying SR learning and use the same method to scale covariates in the testing dataset. Table 3.9 summarises means and standard deviations of continuous baseline covariates (the binary covariate "new or returning donor status" is excluded) that are used in the analysis of the INTERVAL trial data before and after scaling to $[-1, 1]$ for 884 female donors in the INTERVAL trial who were younger than 40 and had O negative blood type, and Figure 3.9 displays pairwise correlations among these continuous baseline covariates. Not surprisingly, some routinely-collected blood measurements are highly-correlated. For example, the correlation between mean corpuscular volume (MCV) and mean corpuscular haemoglobin (MCH) is greater than 0.8, and both of them have large negative correlations (< -0.5) with the red blood cell count (RBC).

Table 3.9: *Descriptive statistics for 884 female donors in the INTERVAL trial who were younger than 40 and had O negative blood type: means and standard deviations (SD) of continuous baseline covariates that are used in the analysis of the INTERVAL trial data before and after scaling to* $[-1, 1]$. *Values before scaling are rounded to 3 significant figures and values after scaling are rounded to 3 decimal places.*

		Before Scaling		After Scaling	
Baseline Covariates	Short Name	Mean	SD	Mean	SD
Age (years)	`age`	29.3	5.78	0.029	0.525
Physical component score	`pcs`	57.8	4.26	0.397	0.201
Mental component score	`mcs`	52.3	7.19	0.386	0.258
2-year donation history	`donhist`	2.55	1.54	-0.271	0.440
Ferritin (μg/L)	`ferr`	30.7	24.7	-0.688	0.251
Transferrin (mg/L)	`transf`	35.5	7.58	-0.273	0.297
White blood cell count (10^9/L)	`wbc`	7.04	1.71	-0.224	0.299
Red blood cell count (10^{12}/L)	`rbc`	4.58	0.356	0.210	0.223
Haemoglobin level (g/dL)	`hgb`	13.3	0.941	-0.035	0.273
Platelet count (10^9/L)	`plt`	256	59.4	0.057	0.248
Mean corpuscular volume (fL)	`mcv`	92.6	5.30	-0.033	0.295
Mean corpuscular haemoglobin (pg)	`mch`	29.2	2.09	-0.558	0.138
Body mass index (kg/m^2)	`bmi`	25.3	5.29	-0.555	0.248

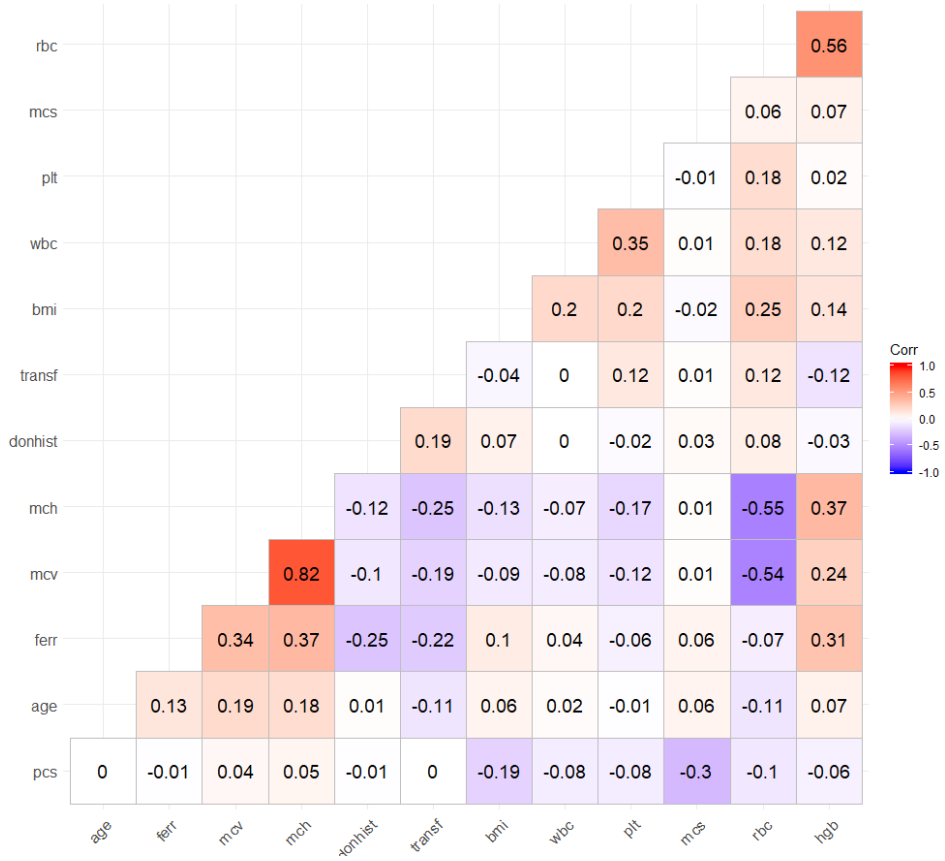


Fig. 3.9: *Pairwise correlation plot of continuous baseline covariates that are used in the analysis of the INTERVAL data.*

Similar to simulation studies, we compare the proposed SR learning (SR and SR-Select) with D-learn, PLS, ACWL-C1, and ACWL-C2, and all tuning parameters are chosen by 5-fold cross-validation. We focus on linear decision rules for ease of interpretation (we also notice that analogous to simulation settings 1-5, for this dataset, using the Gaussian kernel does not improve the performance over the linear kernel, which may suggest that true decision boundaries do not deviate much from linear ones). Same as in Chapter 2, to evaluate the performance of each method, we randomly split the data into five almost equal-sized parts and repeat this procedure 100 times. Within each split, we use four parts as the training dataset to learn the optimal rule and predict the optimal inter-donation intervals for donors in the remaining part (testing dataset) based on the estimated rule. We calculate proportions of donors assigned to each inter-donation interval option and the empirical value function on the testing set. In addition, we also estimate the "ITR effect", δ, of the estimated rule on the testing set as follows:

$$\widehat{\delta}(\widehat{\mathcal{D}^*}) = \frac{\mathbb{P}_{n_{\text{test}}}[YI\{A = \widehat{\mathcal{D}^*}(\mathbf{X})\}/P(A|\,\mathbf{X})]}{\mathbb{P}_{n_{\text{test}}}[I\{A = \widehat{\mathcal{D}^*}(\mathbf{X})\}/P(A|\,\mathbf{X})]} - \frac{\mathbb{P}_{n_{\text{test}}}[YI\{A \neq \widehat{\mathcal{D}^*}(\mathbf{X})\}/P(A|\,\mathbf{X})]}{\mathbb{P}_{n_{\text{test}}}[I\{A \neq \widehat{\mathcal{D}^*}(\mathbf{X})\}/P(A|\,\mathbf{X})]},$$

where $\mathbb{P}_{n_{\text{test}}}$ denotes the empirical average over the testing dataset. The estimated ITR effect, $\widehat{\delta}(\widehat{\mathcal{D}^*})$, measures the difference in the empirical value function between the strategy that assigns donors according to the estimated rule and that assigns donors to inter-donation intervals different from the estimated rule (Qiu et al., 2018; Xu et al., 2015).

Figure 3.10 plots distributions of estimated value functions, ITR effects and proportions of donors assigned to the 16-week inter-donation interval (safest option) based on 100 repetitions of 5-fold cross-validation, and Table 3.10 summarises means and standard deviations of cross-validated value functions, ITR effects and donor assignment proportions. For comparison, we also present in Table 3.10 the results corresponding to three non-personalised rules (fixed donation strategies) where all donors are assigned to the same inter-donation interval (16-week, 14-week, or 12-week).

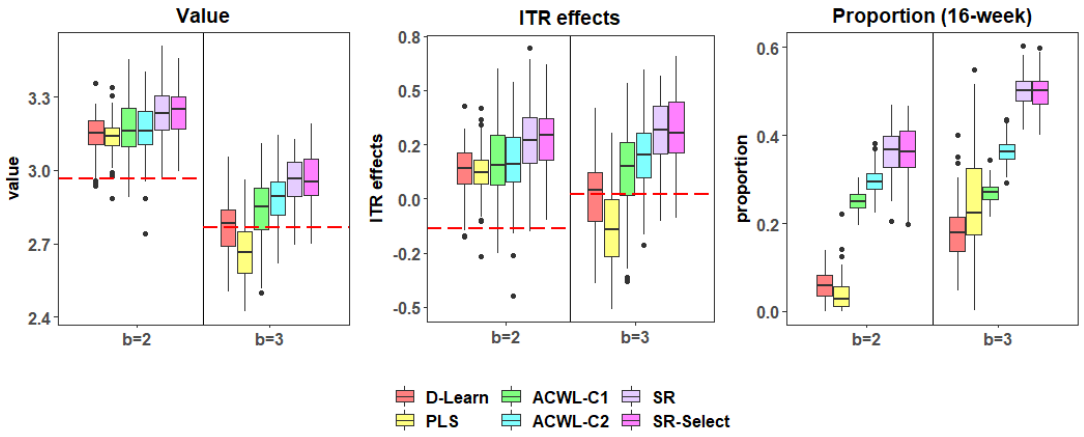


Fig. 3.10: *Analysis results of the INTERVAL data when the trade-off parameter b in the utility function takes values of 2 and 3: boxplots of estimated value functions, ITR effects, and proportions of donors assigned to the 16-week inter-donation interval (safest option) evaluated on testing datasets based on 100 repetitions of 5-fold cross-validation. Red dashed lines represent value functions/ITR effects under the current clinical practice (assign all female donors to the 16-week inter-donation INTERVAL).*

Table 3.10: *Analysis results of the INTERVAL data when the trade-off parameter b in the utility function takes values of 2 and 3: mean (SD) of donor assignment proportions, value functions, and ITR effects evaluated on testing datasets over 100 repetitions of 5-fold cross-validation for personalised and non-personalised rules. The largest value functions and ITR effects are in bold. Value functions and ITR effects corresponding to the current clinical practice are underlined.*

$b = 2$					
	Assignment Proportion (%)			Value	ITR Effects
	16 weeks	14 weeks	12 weeks		
D-Learn	5.9 (3.4)	7.6 (4.2)	86.6 (5.3)	3.144 (0.081)	0.131 (0.121)
PLS	3.8 (3.7)	9.0 (5.5)	87.2 (6.7)	3.137 (0.072)	0.125 (0.110)
ACWL-C1	25.1 (2.1)	29.4 (2.3)	45.6 (2.4)	3.167 (0.111)	0.168 (0.169)
ACWL-C2	29.6 (3.0)	27.3 (2.9)	43.1 (3.2)	3.167 (0.107)	0.169 (0.159)
SR	36.2 (5.3)	9.2 (4.5)	54.6 (4.7)	3.233 (0.101)	0.269 (0.154)
SR-Select	35.9 (6.3)	8.8 (5.2)	55.3 (5.5)	**3.236 (0.096)**	**0.272 (0.145)**
Fixed-16	100.0 (0.0)	0.0 (0.0)	0.0 (0.0)	2.964 (0.014)	-0.135 (0.015)
Fixed-14	0.0 (0.0)	100.0 (0.0)	0.0 (0.0)	3.028 (0.014)	-0.041 (0.015)
Fixed-12	0.0 (0.0)	0.0 (0.0)	100.0 (0.0)	3.162 (0.015)	0.166 (0.017)

$b = 3$					
	Assignment Proportion (%)			Value	ITR Effects
	16 weeks	14 weeks	12 weeks		
D-Learn	18.0 (6.0)	13.3 (5.4)	68.7 (7.5)	2.770 (0.114)	0.020 (0.169)
PLS	24.4 (11.1)	9.5 (8.1)	66.1 (11.3)	2.667 (0.114)	-0.136 (0.172)
ACWL-C1	27.0 (2.3)	28.1 (2.3)	44.9 (2.7)	2.843 (0.136)	0.134 (0.206)
ACWL-C2	36.3 (3.1)	25.4 (3.1)	38.3 (3.3)	2.890 (0.111)	0.205 (0.167)
SR	50.1 (3.4)	8.5 (4.1)	41.4 (3.8)	2.957 (0.095)	0.307 (0.144)
SR-Select	49.9 (3.6)	7.5 (3.8)	42.6 (4.1)	**2.966 (0.101)**	**0.321 (0.152)**
Fixed-16	100.0 (0.0)	0.0 (0.0)	0.0 (0.0)	2.768 (0.014)	0.020 (0.016)
Fixed-14	0.0 (0.0)	100.0 (0.0)	0.0 (0.0)	2.752 (0.016)	-0.003 (0.017)
Fixed-12	0.0 (0.0)	0.0 (0.0)	100.0 (0.0)	2.745 (0.016)	-0.016 (0.016)

We observe that for both $b = 2$ and $b = 3$, SR and SR-Select outperform other ITR estimation methods by achieving larger values and larger ITR effects on the utility outcome. In addition, by comparing donor assignment proportions across personalised rules, we find that ITRs estimated by our proposed methods are less "aggressive" than others in the sense that both SR and SR-Select assign more donors to the safest and currently-implemented inter-donation interval (16-week), especially when $b = 3$, in which case low Hb deferral is considered to incur a larger equivalent loss in blood collection. This feature is highly desirable for the donor population under investigation (young female donors with O negative blood type) since these donors are more vulnerable to iron deficiencies and low Hb levels. Due to the growing demand for universal blood groups, the blood service is keen to collect more blood from these donors, but in the meantime, to safeguard donor health and retain such donors, there is a limit on the donation frequency such that deferrals due to low Hb can be well-controlled. If we follow personalised donation strategies suggested by our proposed methods and assign more donors to the longest and safest inter-donation interval, we would anticipate a reduction in the average risk (low Hb deferral) compared to using the personalised donation strategies estimated by other alternative methods.

We also compare personalised donation strategies resulting from SR/SR-Select with the current clinical practice (Fixed-16). Comparisons in terms of donor assignment proportions suggest that our proposed methods encourage more than half of the donors to donate more frequently than the current practice in order to achieve a larger overall utility score which accounts for both the benefit and the risk. ITR effects associated with proposed methods are much larger than those of Fixed-16. When $b = 2$, the empirical values of SR-Select and Fixed-16 are 3.236 units and 2.964 units, respectively, suggesting that the personalised donation strategy estimated by SR-Select leads to an average increase of 127.84 ml (0.272 units) blood collected ("discounted" by the low Hb deferral) by the blood service *per donor* over 2 years compared to the current practice. The benefit of using the donation strategy estimated by SR-Select when $b = 3$ can be calculated in a similar way, which yields 93.06 ml (0.198 units) additional blood collected ("discounted" by the low Hb deferral) *per donor* over 2 years. It is likely that the estimated personalised donation strategies based on the INTERVAL data are generalisable to the general UK blood donor pool, since Moore et al. (2016) showed that INTERVAL trial participants were broadly representative of the national donor population of England. According to the NHSBT blood donation database, there are about 23600 female donors under 40 with O negative blood

group in the UK general donor population, implying an increase of approximately 3000 litres of blood collected from this donor population when $b = 2$ (and roughly 2200 litres when $b = 3$) in a 2-year period by implementing this personalised donation strategy.

Since linear decision boundaries are assumed in our analysis, we can assess the influences of different covariates (i.e. measure the "covariate importance") on linear decision functions by calculating and comparing the "standardised" absolute effects of covariates, i.e. $|\widehat{\boldsymbol{\beta}}|/\widehat{\mathrm{SE}}(\widehat{\boldsymbol{\beta}})$, where $\widehat{\boldsymbol{\beta}}$ is the vector of coefficient estimates (for scaled covariates) in linear decision functions and $\widehat{\mathrm{SE}}(\widehat{\boldsymbol{\beta}})$ is the vector of standard error estimates for $\widehat{\boldsymbol{\beta}}$. For illustration purposes, in Table 3.11, we present $\widehat{\boldsymbol{\beta}}$, $\widehat{\mathrm{SE}}(\widehat{\boldsymbol{\beta}})$ (estimated by the stratified bootstrap method based on 500 bootstrap samples), $|\widehat{\boldsymbol{\beta}}|/\widehat{\mathrm{SE}}(\widehat{\boldsymbol{\beta}})$, and the ranks of $|\widehat{\boldsymbol{\beta}}|/\widehat{\mathrm{SE}}(\widehat{\boldsymbol{\beta}})$ (in descending order such that the largest one ranks first) for the linear boundary that determines whether or not a young female donor with O negative blood group should donate more frequently than the current clinical practice (every 16 weeks).

Table 3.11: *Data analysis results from the proposed SR learning for $b = 2$ and $b = 3$: coefficient estimates for scaled baseline covariates (i.e. $\widehat{\boldsymbol{\beta}}$) of the decision boundary for whether or not a young female donor with O negative blood group can donate more frequently than the current practice. Standard error estimates of $\widehat{\boldsymbol{\beta}}$ (i.e. $\widehat{SE}(\widehat{\boldsymbol{\beta}})$) are obtained using the bootstrap method based on 500 bootstrap samples. The "standardised" absolute effects (i.e. $|\widehat{\boldsymbol{\beta}}|/\widehat{SE}(\widehat{\boldsymbol{\beta}})$) and their corresponding ranks (in descending order such that the largest one ranks first) are also provided. The two largest "standardised" absolute effects are in bold.*

	$b = 2$				$b = 3$			
Baseline Covariates	$\widehat{\boldsymbol{\beta}}$	$\widehat{\mathrm{SE}}(\widehat{\boldsymbol{\beta}})$	$\lvert\widehat{\boldsymbol{\beta}}\rvert/\widehat{\mathrm{SE}}(\widehat{\boldsymbol{\beta}})$	Rank	$\widehat{\boldsymbol{\beta}}$	$\widehat{\mathrm{SE}}(\widehat{\boldsymbol{\beta}})$	$\lvert\widehat{\boldsymbol{\beta}}\rvert/\widehat{\mathrm{SE}}(\widehat{\boldsymbol{\beta}})$	Rank
Age (years)	0.258	0.304	0.846	7	0.270	0.299	0.900	6
Physical component score	-0.908	0.552	1.646	4	-1.022	0.547	1.868	3
Mental component score	0.870	0.447	1.945	3	0.752	0.422	1.783	4
New or returning donor status	0.176	0.274	0.644	8	0.232	0.272	0.850	7
2-year donation history	0.519	0.342	1.516	5	0.447	0.339	1.318	5
Ferritin (μg/L)	1.244	0.547	**2.275**	2	1.121	0.577	**1.942**	2
Transferrin (mg/L)	-0.169	0.471	0.359	10	-0.313	0.465	0.673	9
White blood cell count (10^9/L)	-0.262	0.460	0.569	9	-0.129	0.445	0.289	10
Red blood cell count (10^{12}/L)	0.049	8.062	0.006	14	-0.109	8.407	0.013	14
Haemoglobin level (g/dL)	0.532	6.288	0.085	13	0.698	6.538	0.107	12
Platelet count (10^9/L)	-0.450	0.505	0.892	6	-0.415	0.506	0.819	8
Mean corpuscular volume (fL)	-1.800	0.727	**2.476**	1	-1.570	0.735	**2.136**	1
Mean corpuscular haemoglobin (pg)	1.681	12.638	0.133	11	1.168	13.585	0.086	13
Body mass index (kg/m^2)	-0.048	0.536	0.089	12	0.150	0.531	0.283	11

Results in Table 3.11 suggest that for both $b = 2$ and $b = 3$, the two most important baseline covariates (with the largest "standardised" absolute effects) in determining whether or not a young female donor with O negative blood group can be assigned to a shorter inter-donation interval than the 16-week one are mean corpuscular volume and ferritin.

3.4.2 Differences in donors' covariate profiles between those who can donate more frequently than the current clinical practice and those who cannot

In this section, we consider another way to examine which covariates "drive" the decision of whether or not a female donor younger than 40 with O negative blood group is capable of giving blood more frequently than every 16 weeks and to better understand the differences in donors' covariate profiles between those who are recommended to give blood more frequently than every 16 weeks by SR learning and those recommended to stick with the current clinical practice. The method that we employ is called "profile regression", which is a nonparametric outcome-guided clustering approach under the Bayesian mixture modelling framework (Molitor et al., 2010). Profile regression partitions subjects into groups according to their covariate and outcome profiles, and it enables the investigation of the joint effects of multiple individual characteristics. One major advantage of profile regression is that it can handle correlated covariates (i.e. address the multicollinearity issue which may cause instability in model estimation). We leave the detailed discussion on the components of the profile regression model to the next chapter, in which we propose a new framework for patient stratification based on the profile regression model.

Here, we apply profile regression to the personalised donation strategies recommended by SR learning when $b = 3$ to demonstrate that profile regression can help us gain a better understanding of the ITR estimation results. We define the outcome of the profile regression model as a binary variable which takes the value of 1 if SR learning recommends the donor to give blood more often than every 16 weeks, and takes the value of 0 if SR learning recommends the donor to donate every 16 weeks. All 14 baseline covariates used for the estimation of the optimal personalised donation strategies are included in the profile regression model as profile variables. We run the algorithm for 40000 iterations with the initial 20000 iterations discarded as burn-in. The inspection of the posterior samples suggests no evidence against convergence.

The heatmap of the posterior similarity matrix (Liverani et al., 2015; Molitor et al., 2010) for 884 female donors is shown in Figure 3.11. Observations have been reordered such that the pairs of donors who tend to have the same cluster memberships appear together. We observe a clear separation across these donors (i.e. strong indication of clustering into 2 separate groups).

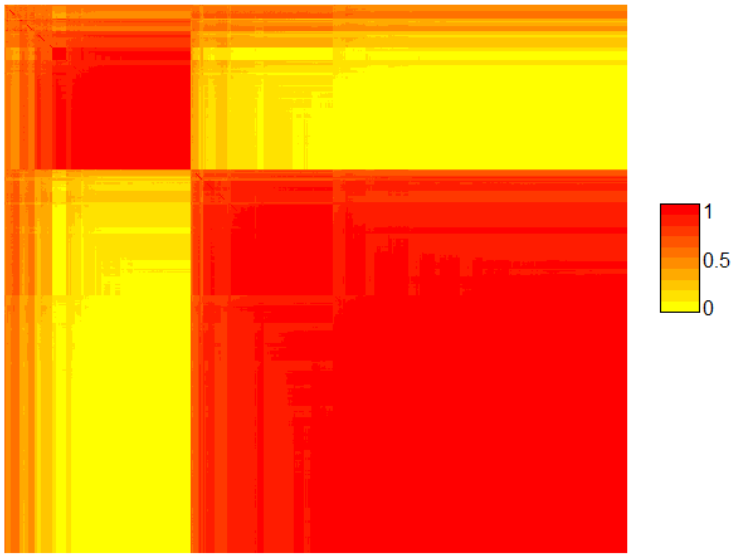
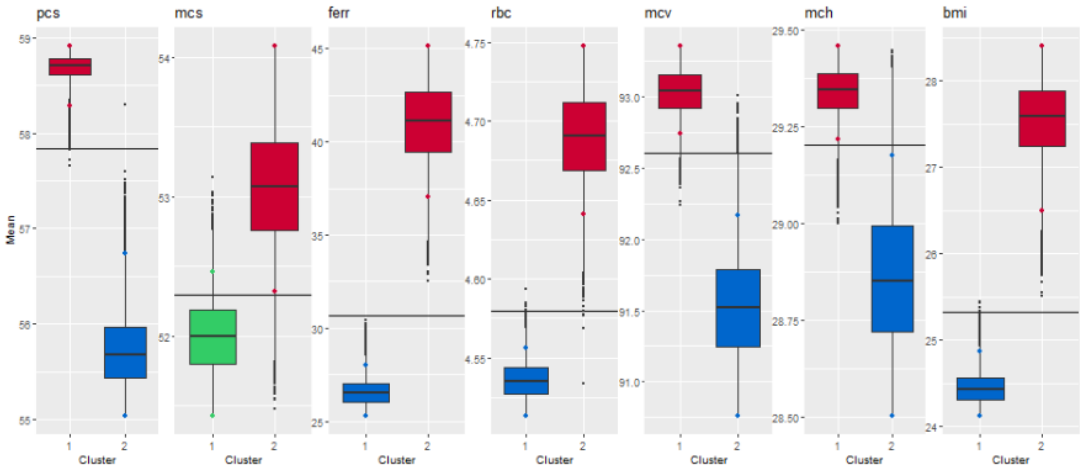
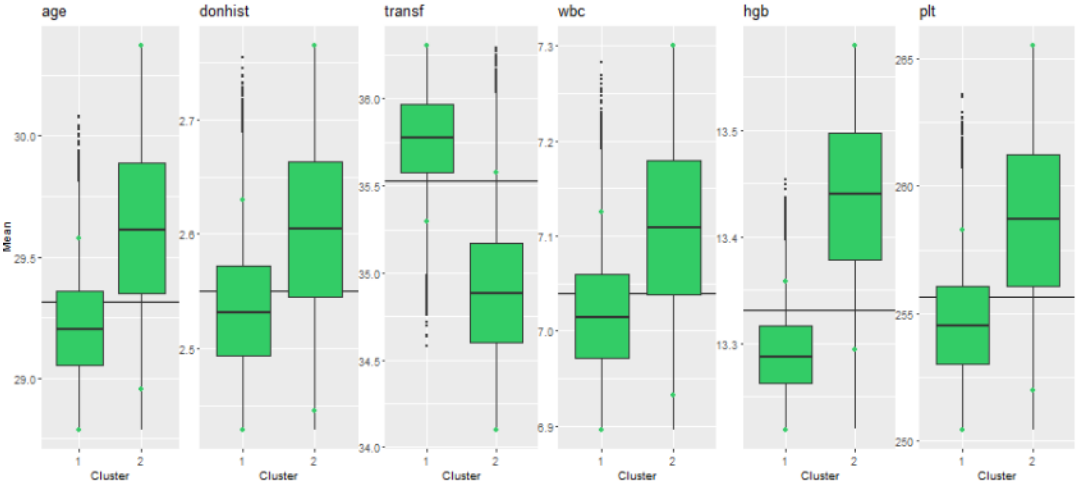


Fig. 3.11: *Heatmap of 884 female donors' posterior similarity matrix based on profile regression. Different colours correspond to different degrees of individuals' similarity to each other (yellow - low similarity; red - high similarity).*

Covariate profiles (posterior distributions for the means) of donors in each cluster are plotted in Figure 3.12. The red and the blue colours of the boxes (the lower and upper hinges of the boxes represent the 25^{th} and 75^{th} percentiles, respectively) indicate that the 90% credible intervals for the cluster-specific mean are above and below the average values across two clusters (black solid lines), respectively. The green colour implies that the 90% credible intervals for the cluster-specific mean include the average. Figure 3.12 (a) corresponds to important covariates that "drive" the clustering structure (for these covariates, the profiles in clusters 1 and 2 are very different), and Figure 3.12 (b) corresponds to covariates that are less important for the clustering of the donors into those who can give blood more frequently than the current clinical practice and those who cannot (the only discrete covariate in our analysis, "new or returning donor status", also belongs to this category. We exclude the subfigure for this covariate due to the inconsistency in the y-axis for discrete and continuous covariates).



(a) *Important covariates that have a major impact on the clustering structure.*



(b) *Covariates that are less relevant to the clustering structure.*

Fig. 3.12: *Covariate profiles from the profile regression model: posterior distributions of the mean parameter for profile variables in clusters 1 and 2. Black solid lines represent the average of the cluster-specific mean in clusters 1 and 2 for this covariate. The red-coloured and the blue-coloured boxes indicate that the 90% credible intervals for the cluster-specific mean are above and below the average values across clusters 1 and 2, respectively. The green-coloured boxes imply that the 90% credible intervals for the cluster-specific mean include the average. We note that the boxes cover the interquartile range (lower hinge: the 25^{th} percentile; upper hinge: the 75^{th} percentile).*

In cluster 1 and cluster 2, the posterior means for the probability of a donor being capable of giving blood more frequently than every 16 weeks are 0.32 and 0.79, respectively, i.e. the second cluster represents "super donors": on average, donors in this cluster have a higher probability of being able to donate more frequently (compared to donors in the first cluster). Figure 3.12 suggests that "super donors" (donors in cluster 2) are characterised by having higher than

average ferritin level, mental component score, red blood cell count, and body mass index. In addition, these donors tend to have lower than average physical component score, mean corpuscular volume, and mean corpuscular haemoglobin.

We note that some covariates that are regarded as important by profile regression (Figure 3.12 (a)) are considered to be "not very important" in the linear decision function (for the comparison of {16-week} vs. {14-week, 12-week}) of SR learning (Table 3.11) according to the variable importance measure (i.e. "standardised" absolute covariate effects). For example, the covariate profile figure suggests that mean corpuscular haemoglobin plays an important role in distinguishing between the two clusters, but in Table 3.11, mean corpuscular haemoglobin is the second least important covariate in the decision function according to the "standardised" absolute covariate effects (when $b = 3$). This may be due to the fact that mean corpuscular haemoglobin is highly correlated with mean corpuscular volume (correlation is 0.82, mean corpuscular volume ranks first, while mean corpuscular haemoglobin has a rank of 13 in the linear decision boundary between {16-week} and {14-week, 12-week}). The $\widehat{\mathrm{SE}}(\widehat{\boldsymbol{\beta}})$ column in Table 3.11 suggests that the standard error estimate corresponding to the estimated coefficient of the mean corpuscular haemoglobin is "inflated" (possibly due to multicollinearity), and thus the "standardised" effect of mean corpuscular haemoglobin may be "masked" in linear decision rules. On the other hand, profile regression does not suffer from the multicollinearity problem and it can handle correlated covariates properly: the covariate profile plots clearly illustrate that both mean corpuscular volume and mean corpuscular haemoglobin are important. A similar argument can be made to explain why the red blood cell count ranks last in Table 3.11 but is considered as important in profile regression, given that it is also moderately correlated with mean corpuscular volume. As a remark, similar to the case of multiple linear regression, multicollinearity should not undermine SR learning's prediction performance, its reliability, and its usefulness as a whole, but it may cause problems if we aim to understand the effect of each individual covariate on decision-making. In this case, a follow-up profile regression analysis may offer a more reliable way of assessing the covariate importance.

3.5 Discussion

This chapter presents a sequential re-estimation (SR) learning approach to estimating the optimal ITR among ordinal treatments. By exploiting information on treatment orderings, we decompose the ordinal treatment prediction problem into two sets (sequential and re-estimation) of binary treatment comparison subproblems that can be solved using existing methods designed for situations with two treatment options. In particular, we solve each subproblem via a weighted SVM (Liu et al., 2018), which is computationally efficient and guarantees optimal solutions given the convexity of the underlying optimisation problem (Boyd and Vandenberghe, 2004). Multiple binary decisions can then be aggregated based on a decision tree that again takes into consideration the ordering information on treatments. Empirical results demonstrate that the proposed SR learning significantly improves classification accuracy and value functions on unseen data compared to methods that do not account for the ordinality of treatments. We note that even though the BART ITR estimation method discussed in Section 2.3.4 has competitive performance in multi-arm (unordered) trial settings examined in Chapter 2, it is not included as one of the comparison methods in this chapter (Sections 3.3 and 3.4) due to its Bayesian nature and the fact that the current implementation of this method does not incorporate the variable selection feature (while all the comparison methods considered in Sections 3.3 and 3.4 are frequentist methods and perform variable selection intrinsically when estimating the optimal ITR). Preliminary simulation results based on 100 replicates in low-dimensional scenarios suggest that SR learning outperforms BART in all ordinal simulation settings that we examine, and the advantage becomes more pronounced as the number of treatments gets larger. For example, when $n = 400$, for settings 1-4 (linear parallel boundaries), the misclassification rates by SR learning are always smaller than 0.05, whereas the misclassification rates by BART are about 0.20. For nonparallel and nonlinear settings (7-10), BART performs better than D-learn, PLS, and ACWL, but its performance is still consistently worse than SR learning. An extreme example of this is setting 10 (with 4 ordinal arms). In this case, the misclassification rates by SR learning and BART are 0.09 and 0.31, respectively.

SR learning can be applied to situations with any types of underlying decision boundaries, given that it allows the estimation of both linear and nonlinear decision rules. Linear decision rules derived based on the linear kernel are easy to

interpret, but they are prone to misspecification. On the other hand, Gaussian kernel is much more flexible and robust, but the final decision rules based on the Gaussian kernel are "black-box" and hard to interpret, which may limit their use by practitioners in clinical practice. We note that the Gaussian kernel is the recommended kernel when no additional knowledge about the data is available (Smola et al., 1998). Other simpler but less flexible nonlinear kernels, such as the polynomial kernel, can also be used if we have *a priori* knowledge on the possible type/shape of decision boundaries.

We also develop variable selection methods for ITR estimation under the proposed framework in order to identify covariates that inform treatment decisions and mitigate "contamination" from noise covariates on decision-making. We propose an "embedded" variable selection method for linear decision functions and a two-stage "wrapper" method for nonlinear decision functions (Guyon and Elisseeff, 2003; Jović et al., 2015; Saeys et al., 2007). We note that a two-stage procedure similar to the one used for the nonlinear case can be applied to select covariates and estimate optimal ITRs in the linear case by excluding second-order terms from SODA in the first stage and replacing the Gaussian kernel with linear kernel in the second stage. Much like the justification we provide for the nonlinear case, this two-stage method should work well in the linear case too.

We focus on randomised clinical trials in this thesis, so when we introduce the statistical framework and design the simulation studies in this chapter, we only consider the trial setting. In general, SR learning should also be applicable to observational studies if the consistency, positivity, and no unmeasured confounder (typically unverifiable in observational studies) assumptions hold true and the propensity score model is correctly specified.

Chen et al. (2018) recently proposed the generalised outcome weighted learning (GOWL) method to estimate the optimal ITR in the ordinal-arm setting using data duplication techniques and they assumed that the decision boundaries do not cross with each other, while SR learning does not make this assumption and can handle a wider range of situations (simulated examples are provided in Appendix B.1.3). Whereas, we propose variable selection methods for both linear and nonlinear decision boundaries under the SR learning framework, variable selection has not been incorporated into GOWL. GOWL has not yet been included as a comparison method in our simulation studies, since our attempts to implement this method (using the Python codes provided in the supplementary materials of Chen et al. (2018)) show some inconsistencies in the computation that we are not able to reconcile at the moment, and this requires further investigation. After

these issues are fixed, we will compare SR learning with GOWL in settings 1-10 considered in Section 3.3.1 (only for the low-dimensional scenario without noise covariates, given that GOWL has not been extended to handle the situation with many noise covariates).

The work in Chapters 2 and 3 mainly focus on the estimation of the optimal ITR. The other perspective on personalised medicine, i.e. subgroup identification/patient stratification, can also be very useful in practice in terms of facilitating the understanding of patient heterogeneity, and informing what covariate information should be collected and which target population should be selected for future studies. In Section 3.4.2, we present a follow-up analysis on the optimal personalised donation strategy estimated by SR learning to demonstrate how we can take advantage of Bayesian clustering techniques to cluster individuals into subgroups, characterise resulting subgroups, and identify the covariates that actively "drive" the clustering structures (i.e. covariates that are important in differentiating between various subgroups). We further explore this idea in the next chapter.

Chapter 4

Patient stratification in multi-arm trials: a two-stage procedure with Bayesian profile regression

4.1 Introduction

In the previous two chapters, we mainly consider the mapping of patient-specific characteristics to the treatment option that will potentially optimise patients' clinical outcomes, and our ultimate goal is to identify the best personalised treatment assignment strategy. In this regard, most methods (for ITR estimation) discussed in Chapters 2 and 3 do not directly target the patient stratification and the understanding and characterisation of the underlying patient/treatment effect heterogeneity, which can be of great interest and importance for a variety of medical applications. For example, in the blood donation context, it would be useful to stratify donors into those who have the capacity to donate more often than the current clinical practice (they are highly-motivated donors with minimal number of low Hb deferrals, and are referred to as "super donors") and those who tend to be deferred more frequently due to safety concerns than the average donors (these donors' iron stores drain at a greater rate than the average donors after blood donation and they struggle more than the average donors to replace iron stores. They are commonly referred to as "brittle donors"): "super donors" may be encouraged to give blood more frequently when there is a blood shortage or if their blood type is rare or universal, while "brittle donors" may be asked to

donate less often than the general donor population in order to allow enough time for post-donation iron stores replenishment (Blood and Transplant Research Unit, 2019). The availability of the extensive data from the INTERVAL trial creates an opportunity to investigate donor stratification (based, for example, on donors' covariate profiles and potential responses to different inter-donation intervals), and such an investigation will facilitate the characterisation of different types of donors, which can be important for achieving more effective targeted recruitment of donors and increased efficiency of blood collection in the future.

A number of data-driven methods for subgroup identification/patient stratification have been proposed. For example, Su et al. (2009) developed the "interaction tree" (IT) method to capture treatment effect heterogeneity by leveraging the idea of recursive partitioning and extending the "classification and regression trees" (CART) framework: a big initial tree is first grown by selecting the best split based on the significance of the interaction terms, and then this initial tree is pruned based on an interaction-complexity criterion. Foster et al. (2011) proposed an approach called "virtual twins" (VT) for finding subgroups with elevated treatment effects in clinical trials with a binary treatment. VT is a two-stage process that builds on the idea of counterfactuals: in the first stage, potential outcomes under both levels of a binary treatment are predicted by random forest models, and in the second stage, a regression tree model is fit with the outcome being the estimated treatment contrasts (i.e. difference between two predicted potential outcomes) and the covariates being patients' baseline characteristics. We refer readers to Lipkovich et al. (2017), Sies et al. (2019), and a recent book by Ting et al. (2020) for a comprehensive review of the subgroup analysis methods. To our knowledge, most of the existing subgroup analysis methods have not been extended to the situation with multi-level treatments, and thus they cannot be directly applied to the INTERVAL data for donor stratification.

In this chapter, we propose a method for exploring treatment effect heterogeneity and uncovering subgroups with differential treatment responses in the multi-arm trial setting. We approach the heterogeneity detection problem from a clustering perspective, which to our knowledge, has rarely been explored previously. Our aim is to stratify the heterogeneous population and group individuals into more homogeneous subgroups according to patients' treatment response profiles and covariate profiles. In practice, the functional form of the relationship between treatment responses and covariates is typically unknown *a priori*. To this end, we leverage Bayesian nonparametric approaches. In particular, our

proposed method is a two-stage procedure. In the first stage, we use Bayesian additive regression trees (BART) to "predict" each patient's potential outcomes under different treatment options. In the second stage, we employ a Bayesian mixture model called "profile regression" (Molitor et al., 2010) to link the predicted potential treatment responses obtained in the first stage (multivariate continuous outcome variable in profile regression) to a set of possibly correlated patient characteristics (covariates in profile regression) nonparametrically through cluster/subgroup memberships. The Bayesian framework allows the quantification of the uncertainty associated with the cluster assignments, which is important for interpreting clustering results (Molitor et al., 2010; Rouanet et al., 2020).

We note that in profile regression, the covariates and the outcome are modelled jointly, and thus both the covariate data and the outcome information influence cluster allocations. In the context of personalised medicine, interest typically lies in identifying subgroups that are associated with the outcome on which treatment effects are measured (this outcome is the target outcome of clinical interest). For example, by leveraging the INTERVAL data, we aim to stratify blood donors based on the utility score (our target outcome) which balances the amount of blood collected and the low Hb deferrals. In clinical studies, there are typically many covariates that can give rise to many different clustering structures. The inclusion of the target outcome information in the clustering algorithm can mitigate the concerns about identifying clusters that are irrelevant for the specific outcome of clinical interest and will lead to more clinically meaningful clustering results (Bair, 2013; Rouanet et al., 2020+).

For classical clustering approaches (e.g. K-means, Gaussian mixture models), the number of clusters generally needs to be set *a priori*. Even though multiple criteria have been proposed to determine the number of clusters, there is no consensus on the optimal criterion, and thus specifying the number of clusters can be challenging in practice (Rousseeuw, 1987; Tibshirani et al., 2001; Wang, 2010). Our method sidesteps this difficulty by using the Dirichlet process prior, in which case the number of subgroups/clusters can be directly inferred from the data.

Another important advantage of our proposed approach is that it can handle correlated covariates. This feature is appealing since variables collected in clinical studies are likely to be correlated, and it would be highly desirable if correlations among covariates do not undermine the model performance and the interpretative value of each individual covariate (Hennig and Liao, 2013).

In practice, some covariates available in clinical studies may not play an important role in "driving" the clustering structure that is clinically relevant. In this case, it would be useful to identify the covariates that actively "drive" the clustering components, and to this end, a variable selection procedure is incorporated into our proposed approach.

This chapter is structured as follows. In Section 4.2, we introduce the statistical framework of our proposed method for patient stratification. The numerical performance of the method is evaluated by simulation studies in Section 4.3. An application of the proposed method to the INTERVAL data is presented in Section 4.4. This chapter is concluded with a discussion in Section 4.5.

4.2 Methodology

We propose a two-stage procedure for patient stratification in multi-arm trials. Specifically, in the first stage, the Bayesian additive regression tree (BART) is used to predict each individual's potential outcomes under different treatment options. In the second stage, a Bayesian clustering model called "profile regression" is considered for clustering patients into subgroups according to their covariate profiles and treatment response profiles.

As in Chapters 2 and 3, we consider a clinical trial with K treatments (randomised groups) and n subjects. We let $Y \in \mathbb{R}$ denote the observed outcome of interest, $A \in \mathcal{A} = \{1, \ldots, K\}$ denote the treatment assignment (A can be nominal or ordinal), and $\mathbf{X} \in \mathcal{X}$ denote the covariate data. The potential outcome under treatment a is denoted by $Y^*(a)$.

4.2.1 BART - predict potential outcomes

In the first stage, we fit a flexible regression model for Y given A and $\mathbf{X}$ in order to get the potential outcome predictions for subjects. We denote the predicted potential outcomes under treatments $1, \ldots, K$ by $\widehat{Y}^*(1), \ldots, \widehat{Y}^*(K)$, respectively. In theory, $\widehat{Y}^*(1), \ldots, \widehat{Y}^*(K)$ can be obtained from the observed data using any supervised machine learning or regression algorithms (Künzel et al., 2019), including random forest (RF) (Breiman, 2001), gradient boosting trees (GBT) (Friedman, 2001), Bayesian additive regression trees (BART) (Hill, 2011), neural networks (Haykin, 1998), and super learner (van der Laan et al., 2007). In particular, we use BART in our implementation because it is highly flexible in the sense that it can capture nonlinear and interaction effects. BART's

excellent predictive performance has been reported in various applications to causal inference (Hill et al., 2020) and also been demonstrated in Chapter 2 of this thesis. Another practical advantage of BART over other statistical learning approaches is that its performance is fairly robust to the choice of tuning parameters (Green and Kern, 2012; Hill, 2011). The formulation of the BART model and the choice of priors can be found in Section 2.3.4, and so we do not repeat them here.

We note that $\widehat{Y}^*(1), \ldots, \widehat{Y}^*(K)$ can also be obtained by fitting a separate model for $Y|\,\mathbf{X}$ under each treatment option ("separate-learner") as opposed to the method described above where we fit one single model for $Y|A, \mathbf{X}$ using the data from all subjects ("single-learner"). These two ways of predicting potential outcomes yield similar results in our numerical studies, and we only present the results based on the "single-learner" in Sections 4.3 and 4.4. However, "single-learner" and "separate-learner" might perform differently in some other situations, and this has been discussed extensively in Künzel et al. (2019).

4.2.2 Bayesian profile regression - clustering

Molitor et al. (2010) proposed a Bayesian infinite mixture model called "profile regression", which links the outcome vector to a set of possibly correlated covariates nonparametrically through cluster memberships. As has been discussed in Section 4.1, the outcome also informs the cluster assignments in profile regression due to the joint modelling of the covariates and the outcome, and this ensures that the clustering result has clinical utility with regard to our aim. Profile regression partitions subjects into clusters by leveraging the Dirichlet process mixture model (DPMM). Liverani et al. (2015) discussed the applications of profile regression in settings with binary, categorical, count, or univariate continuous outcomes. Rouanet et al. (2020+) extended the profile regression model to the case with a multivariate normal outcome, and we follow their proposed framework when developing our model. We use $\mathbf{Y}^* = (\widehat{Y}^*(1), \ldots, \widehat{Y}^*(K))$ to denote the potential outcome vector, which is the multivariate continuous response variable in the profile regression model. We assume that $\mathbf{Y}^*$ and $\mathbf{X}$ are independent given cluster membership (conditional independence assumption). The likelihood function is then given by

$$p(\mathbf{Y}^*, \mathbf{X}\,|\boldsymbol{\pi}, \boldsymbol{\Theta}, \boldsymbol{\Phi}) = \sum_{c=1}^{\infty} \pi_c f(\mathbf{Y}^*\,|\Theta_c) f(\mathbf{X}\,|\Phi_c), \tag{4.1}$$

where c is the index of mixture components, $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots)$ is a vector of mixture weights, $\boldsymbol{\Theta} = (\Theta_1, \Theta_2, \ldots)$ denotes cluster-specific parameters in the density function for $\mathbf{Y}^*$, and $\boldsymbol{\Phi} = (\Phi_1, \Phi_2, \ldots)$ denotes cluster-specific parameters in the density function for $\mathbf{X}$. We note that by construction, DPMM allows infinite components. Therefore, the summation in (4.1) goes from 1 to infinity. As a remark, in practice, the conditional independence assumption can be hard to verify and how realistic this assumption is (whether or not this assumption provides a good approximation to reality) may depend on the specific application. However, conditional independence can always be achieved by increasing the number of clusters, and if some of the resulting clusters are considered as redundant by practitioners, they can be merged with other clusters based on subject-matter knowledge.

4.2.2.1 Specification of model components

In the following, we describe the model for each component of (4.1), but we do not provide further details on the Bayesian computation aspect in this chapter. We refer the interested readers to Liverani et al. (2015) and Rouanet et al. (2020+) for computational details.

Mixture weights $\boldsymbol{\pi_c}$ The mixture weights (π_c) are modelled according to the stick-breaking construction of the Dirichlet process (DP) as follows (Sethuraman, 1994):

$$\begin{aligned} V_c &\sim \text{Beta}(1, \alpha) \quad \alpha > 0, \\ \pi_1 &= V_1, \\ \pi_c &= V_c \prod_{r=1}^{c-1} (1 - V_r) \quad \text{for } c \geq 2, \end{aligned} \tag{4.2}$$

where $V_1, V_2, \ldots$ are independent random variables. Under this representation, α is the concentration parameter of DP, which reflects the dispersion level and controls the number of non-empty clusters implicitly (Frühwirth-Schnatter and Malsiner-Walli, 2019; Hastie et al., 2015; Teh, 2010). We follow Rouanet et al. (2020+) and adopt a Gamma prior for α.

The model for covariates X The profile regression model can handle both continuous and discrete covariates. In the following, we consider the case where $\mathbf{X}$ consists of p_1 continuous covariates and p_2 discrete covariates. In order to describe

the full covariate model, we assume that $p_1, p_2 \geq 1$ for now. The situations with $p_1 = 0$ or $p_2 = 0$ will be discussed later. We let $\mathbf{X}^{\text{cont}}$ and $\mathbf{X}^{\text{disc}}$ denote the subset of continuous and discrete covariates in $\mathbf{X}$, respectively. Without loss of generality, and for notational convenience, we assume that the first p_1 covariates in $\mathbf{X}$ are continuous, i.e. X_j (the j^{th} covariate in $\mathbf{X}$) is continuous for $j = 1, \ldots, p_1$, and X_j is discrete for $j = p_1 + 1, \ldots, p_1 + p_2$.

We assume that $\mathbf{X}^{\text{cont}}$ and $\mathbf{X}^{\text{disc}}$ are independent conditional on the cluster allocations, and then the "density" for covariates $\mathbf{X}$ can be written as:

$$f(\mathbf{X}\,|\Phi_c) = f(\mathbf{X}^{\text{cont}}|\boldsymbol{\mu}_{(\mathbf{X})\,c}, \Sigma_{(\mathbf{X})\,c}) f(\mathbf{X}^{\text{disc}}|\boldsymbol{\Psi}_c), \tag{4.3}$$

where $\Phi_c = (\boldsymbol{\mu}_{(\mathbf{X})\,c}, \Sigma_{(\mathbf{X})\,c}, \boldsymbol{\Psi}_c)$ represents the cluster-specific parameter set for covariates. In particular, $\boldsymbol{\mu}_{(\mathbf{X})\,c}$ and $\Sigma_{(\mathbf{X})\,c}$ are the mean vector and the covariance matrix for continuous covariates in cluster c, and $\boldsymbol{\Psi}_c$ denotes the parameter for discrete covariates in cluster c. Note that we add "$(\mathbf{X})$" in the subscripts of the parameters for continuous covariates (i.e. $\boldsymbol{\mu}_{(\mathbf{X})\,c}$ and $\Sigma_{(\mathbf{X})\,c}$) in order to distinguish them from the parameters for the outcome model (which will be introduced shortly). In addition, as mentioned earlier, Equation (4.3) corresponds to the case where $\mathbf{X}$ consists of both continuous and discrete covariates (i.e. $p_1, p_2 \geq 1$). If all the covariates in $\mathbf{X}$ are continuous (i.e. p_2=0), the right-hand side of Equation (4.3) will be replaced by $f(\mathbf{X}\,|\boldsymbol{\mu}_{(\mathbf{X})\,c}, \Sigma_{(\mathbf{X})\,c})$. On the other hand, if all the covariates in $\mathbf{X}$ are discrete (i.e. p_1=0), the right-hand side of Equation (4.3) will be replaced by $f(\mathbf{X}\,|\boldsymbol{\Psi}_c)$.

For $\mathbf{X}^{\text{cont}}$, we assume the following probability density function:

$$f(\mathbf{X}^{\text{cont}}|\,\boldsymbol{\mu}_{(\mathbf{X})\,c}, \Sigma_{(\mathbf{X})\,c}) = \Big\{(2\pi)^{p_1}|\Sigma_{(\mathbf{X})\,c}|\Big\}^{-\frac{1}{2}} \exp\Big\{-\frac{1}{2}(\mathbf{X}^{\text{cont}} - \boldsymbol{\mu}_{(\mathbf{X})\,c})^{\mathsf{T}}\Sigma^{-1}_{(\mathbf{X})\,c}(\mathbf{X}^{\text{cont}} - \boldsymbol{\mu}_{(\mathbf{X})\,c})\Big\}. \tag{4.4}$$

The conjugate normal-inverse-Wishart (NIW) prior is used for inference, i.e.

$$\begin{aligned} \boldsymbol{\mu}_{(\mathbf{X})\,c}\,|\,\Sigma_{(\mathbf{X})\,c} &\sim \mathcal{N}(\boldsymbol{\mu}_{(\mathbf{X})}, \Sigma_{(\mathbf{X})\,c}/\kappa_{(\mathbf{X})}), \\ \Sigma_{(\mathbf{X})\,c} &\sim \mathcal{W}^{-1}(\Lambda_{(\mathbf{X})}, \nu_{(\mathbf{X})}), \\ \Big(\boldsymbol{\mu}_{(\mathbf{X})\,c}, \Sigma_{(\mathbf{X})\,c}\Big) &\sim NIW(\boldsymbol{\mu}_{(\mathbf{X})}, \kappa_{(\mathbf{X})}, \Lambda_{(\mathbf{X})}, \nu_{(\mathbf{X})}), \end{aligned} \tag{4.5}$$

where $\mathcal{N}$ denotes the multivariate normal distribution, $\mathcal{W}^{-1}$ denotes the inverse-Wishart distribution, NIW denotes the normal-inverse-Wishart distribution, and $\boldsymbol{\mu}_{(\mathbf{X})}$, $\kappa_{(\mathbf{X})}$, $\nu_{(\mathbf{X})}$, and $\Lambda_{(\mathbf{X})}$ (matrix) are hyperparameters.

For discrete covariates, we assume that they are locally independent (independent conditional on the cluster assignments). Then the probability mass function of $\mathbf{X}^{\text{disc}}$ is given by

$$f(\mathbf{X}^{\text{disc}}|\mathbf{\Psi}_c) = \prod_{j=p_1+1}^{p_1+p_2} \psi_{c,j,X_j}\ , \tag{4.6}$$

where $\psi_{c,j,k}$ denotes the probability that covariate j takes the value k in cluster c, $j = p_1+1, \ldots, p_1+p_2$, $k = 1, \ldots, K_j$ (K_j denotes the number of categories for covariate j). Following Liverani et al. (2015), we adopt the conjugate Dirichlet prior, i.e.

$$\Psi_{c,j} = (\psi_{c,j,1}, \psi_{c,j,2}, \ldots, \psi_{c,j,K_j}) \sim \text{Dirichlet}(\boldsymbol{a}_j), \tag{4.7}$$

where $\boldsymbol{a}_j = (a_{j,1}, a_{j,2}, \ldots, a_{j,K_j})$, $j = p_1+1, \ldots, p_1+p_2$.

The model for the outcome $\mathbf{Y}^*$ We let $\Theta_c = (\boldsymbol{\mu}_{(\mathbf{Y}^*)\,c}, \Sigma_{(\mathbf{Y}^*)\,c})$ denote the cluster-specific parameters for the multivariate continuous outcome $\mathbf{Y}^*$ and we consider a multivariate normal (MVN) model for $\mathbf{Y}^*$, i.e.

$$f(\mathbf{Y}^* \,|\, \boldsymbol{\mu}_{(\mathbf{Y}^*)\,c}, \Sigma_{(\mathbf{Y}^*)\,c}) = \Big\{(2\pi)^K |\Sigma_{(\mathbf{Y}^*)\,c}|\Big\}^{-\frac{1}{2}} \exp\Big\{ -\frac{1}{2}(\,\mathbf{Y}^* - \boldsymbol{\mu}_{(\mathbf{Y}^*)\,c})^{\mathsf{T}} \Sigma^{-1}_{(\mathbf{Y}^*)\,c} (\,\mathbf{Y}^* - \boldsymbol{\mu}_{(\mathbf{Y}^*)\,c})\Big\}. \tag{4.8}$$

Similar to $\mathbf{X}^{\text{cont}}$, we specify the conjugate normal-inverse-Wishart prior for inference. Specifically,

$$\begin{aligned} \boldsymbol{\mu}_{(\mathbf{Y}^*)\,c} \,|\, \Sigma_{(\mathbf{Y}^*)\,c} &\sim \mathcal{N}(\boldsymbol{\mu}_{(\mathbf{Y}^*)}, \Sigma_{(\mathbf{Y}^*)\,c}/\kappa_{(\mathbf{Y}^*)}), \\ \Sigma_{(\mathbf{Y}^*)\,c} &\sim \mathcal{W}^{-1}(\Lambda_{(\mathbf{Y}^*)}, \nu_{(\mathbf{Y}^*)}), \\ \Big(\boldsymbol{\mu}_{(\mathbf{Y}^*)\,c}, \Sigma_{(\mathbf{Y}^*)\,c}\Big) &\sim NIW(\boldsymbol{\mu}_{(\mathbf{Y}^*)}, \kappa_{(\mathbf{Y}^*)}, \Lambda_{(\mathbf{Y}^*)}, \nu_{(\mathbf{Y}^*)}), \end{aligned} \tag{4.9}$$

where $\boldsymbol{\mu}_{(\mathbf{Y}^*)}$, $\kappa_{(\mathbf{Y}^*)}$, $\nu_{(\mathbf{Y}^*)}$ and $\Lambda_{(\mathbf{Y}^*)}$ are hyperparameters. We note that the dependence structure of $Y^*(1), \ldots, Y^*(K)$ is hard to track (Du et al., 2020), and thus the specification of the covariance structure of $\mathbf{Y}^*$ can be challenging. In the profile regression model, we make no assumptions about the covariance structure of $\mathbf{Y}^*$, and the posterior inference about the covariance matrix is based on the prior and the predicted potential outcomes from BART (Rouanet et al., 2020+).

4.2.2.2 Post-processing of the clustering output

Identify the "representative" clustering We note that the proposed Bayesian mixture modelling framework (stochastic) takes into account the uncertainty

associated with the number of clusters and cluster assignments, and the clustering output can vary across iterations of the MCMC sampler. To facilitate the interpretation of the clustering results, we "summarise" the clustering output across iterations and identify a "representative" clustering structure. In each MCMC iteration, we can construct an $n \times n$ score matrix, where the element (i_1, i_2) equals to 1 if subjects i_1 and i_2 are allocated to the same cluster in this iteration, and equals to 0 if i_1 and i_2 are assigned to different clusters. We can then average the score matrices over all MCMC iterations to get a posterior similarity matrix, $\mathbf{S}$, which records the probability that two subjects are assigned to the same cluster (i.e. $\mathbf{S}$ records the posterior co-clustering probabilities of all pairs of subjects). The "representative" clustering can be identified as the partition of the data that best represents $\mathbf{S}$. Specifically, we use the partitioning around medoids (PAM) algorithm (Kaufman and Rousseeuw, 1990): PAM is directly applied to the posterior dissimilarity matrix $1 - \mathbf{S}$, which allocates subjects to clusters in a way consistent with $\mathbf{S}$. For each fixed number of clusters up to a prespecified maximum, we select the best PAM partition that minimises the sum of dissimilarities between the centre of each cluster and all other members of the same cluster. The final "representative" clustering is then chosen by maximising the average silhouette width (Rousseeuw, 1987) across these best PAM partitions (Liverani et al., 2015; Molitor et al., 2010; Rouanet et al., 2020).

Quantify the uncertainty associated with the "representative" clustering We can evaluate how confident we are about the "representative" clustering by examining whether or not across different iterations the second-stage clustering model consistently clusters subjects in a way similar to the "representative" clustering, and we would expect the credible intervals associated with cluster parameter estimates to be narrower for more consistent clustering (stronger clustering signal). To this end, the model-averaging approach discussed in Molitor et al. (2010) can be employed under our proposed framework.

4.2.2.3 Variable selection in profile regression

In many clinical applications, the number of covariates may be quite large and the full covariate profiles can be hard to interpret. It is likely that some of the covariates collected in clinical studies have similar profiles across all clusters and it would be of interest to determine which covariates actively "drive" the clustering structure. Variable selection methods can be embedded into profile

regression in order to identify the covariates that contribute significantly to the formation of patient subgroups (clusters). In our implementation, we follow the variable selection approach taken by Liverani et al. (2015).

Continuous covariates We let $\boldsymbol{\gamma}_c^{\text{cont}} = (\gamma_{c,1}, \gamma_{c,2}, \ldots, \gamma_{c,\,p_1})$, where $\gamma_{c,j}$ is a binary random variable that determines whether or not covariate j, $j = 1, \ldots, p_1$, is important for allocating subjects to cluster c ($\gamma_{c,j} = 1$ if the answer is yes and 0 otherwise). Let $\bar{x}_j$ denote the average value of covariate j (sample average), for $j = 1, \ldots, p_1$, and we define $\boldsymbol{\mu}_c^* = (\mu_{c,1}^*, \mu_{c,2}^*, \ldots, \mu_{c,\,p_1}^*)$, where

$$\begin{aligned} \mu_{c,j}^* &= \gamma_{c,j}\,\mu_{c,j} + (1 - \gamma_{c,j})\,\bar{x}_j \\ &= (\mu_{c,j})^{\gamma_{c,j}} \times (\bar{x}_j)^{(1-\gamma_{c,j})}, \end{aligned} \tag{4.10}$$

where $\mu_{c,j}$ is the $j^{\,\text{th}}$ element of $\boldsymbol{\mu}_{(\mathbf{X})\,c}$, $j = 1, \ldots, p_1$. We then replace $\boldsymbol{\mu}_{(\mathbf{X})\,c}$ in (4.4) with $\boldsymbol{\mu}_c^*$. We assume that $\gamma_{c,j} \sim \text{Bernoulli}(\rho_j)$, $j = 1, \ldots, p_1$. A sparsity inducing prior is used for ρ_j (Liverani et al., 2015; Papathomas et al., 2012).

Discrete covariates We can perform variable selection on discrete covariates in a similar manner. We let $\boldsymbol{\gamma}_c^{\text{disc}} = (\gamma_{c,\,p_1+1}, \gamma_{c,\,p_1+2}, \ldots, \gamma_{c,\,p_1+p_2})$, where $\gamma_{c,j}$ denotes a binary variable indicating whether or not covariate j, $j = p_1 + 1, \ldots, p_1 + p_2$, is important for assigning subjects to cluster c. Let $\psi_{c,j,k}$ denote the probability that covariate j takes the value k in cluster c, and $\psi_{0,j,k}$ be the observed proportion of covariate j taking the value k, $j = p_1 + 1, \ldots, p_1 + p_2$, $k = 1, \ldots, K_j$.

To incorporate the variable selection feature, the discrete covariate model (4.6) is modified accordingly to

$$\begin{aligned} f(\mathbf{X}^{\text{disc}} | \boldsymbol{\Psi}_c, \boldsymbol{\gamma}_c^{\text{disc}}) &= \prod_{j=p_1+1}^{p_1+p_2} \gamma_{c,j}\,\psi_{c,j,X_j} + (1 - \gamma_{c,j})\,\psi_{0,j,X_j} \\ &= \prod_{j=p_1+1}^{p_1+p_2} (\psi_{c,j,X_j})^{\gamma_{c,j}} \times (\psi_{0,j,X_j})^{(1-\gamma_{c,j})}. \end{aligned} \tag{4.11}$$

Similar to the continuous covariate case, we assume that $\gamma_{c,j} \sim \text{Bernoulli}(\rho_j)$, and each ρ_j is assigned a sparsity inducing prior, $j = p_1 + 1, \ldots, p_1 + p_2$.

4.2.2.4 Predicting the cluster membership for a new subject

The cluster memberships of new subjects can be predicted from the fitted model. Specifically, in each MCMC iteration, each new subject is assigned to one of the current clusters (we note that these new subjects do not affect the likelihood, and thus they have no impact on the clustering or the cluster-specific parameters at each iteration). In the post-processing step, we compute a similarity matrix $\mathbf{S}^{\text{pred}}$ between these new subjects and the subjects whose information has been used for model fitting in a way similar to how we compute $\mathbf{S}$ in Section 4.2.2.2. $\mathbf{S}^{\text{pred}}$ is then used to predict each new subject's cluster membership in the "representative" clustering – each new subject is allocated to the cluster whose medoid exhibits the greatest similarity to the new subject.

4.3 Simulation studies

In this section, we evaluate the performance of our proposed method via simulation studies. The sample R codes are provided in Appendix C.1.

4.3.1 Simulation design

We consider two simulation scenarios. The first scenario corresponds to the case where all covariates are continuous, and we evaluate the performance of the proposed method under different levels of correlations between covariates. In the second scenario, covariates include a mix of continuous and discrete ones, and we illustrate the utility of the proposed method in identifying noise covariates that do not inform the clustering structure. We examine two sample sizes: $n = 450$ and $n = 900$, and we repeat the simulation 100 times for each scenario. Details on the simulation design are given in the following.

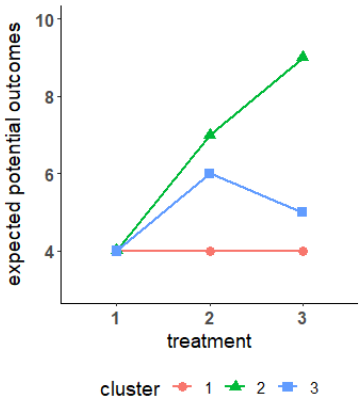
4.3.1.1 Scenario 1

In this setting, we consider a three-arm trial ($K = 3$) where treatment A is sampled from $\{1, 2, 3\}$ with equal probabilities. $\mathbf{X}$ consists of 3 normally-distributed continuous covariates: X_1, X_2, and X_3. There are 3 underlying subgroups/clusters of equal size under this scenario. The mean values of covariates and expected potential outcomes under each treatment option for 3 clusters are summarised in Table 4.1. We also plot the treatment response profiles of subjects in each cluster in Figure 4.1 for better visualisation.

Table 4.1: *Covariate profiles and potential outcome profiles in each cluster under scenario 1.*

	cluster 1	cluster 2	cluster 3
$E(X_1)$	2	4	6
$E(X_2)$	4	6	1
$E(X_3)$	5	1	3
$E\{Y^*(1)\}$	4	4	4
$E\{Y^*(2)\}$	4	7	6
$E\{Y^*(3)\}$	4	9	5

Fig. 4.1: *Plot of the expected potential outcome values (by cluster) under different treatments in scenario 1.*



In particular, we can think of the outcome in our simulation studies as a utility score and without loss of generality, we assume that the treatment becomes more "intensive" as A increases (we note that this assumption is made to facilitate the description of the potential outcome profiles in this setting, but A does not have to be "ordered" in terms of treatment "intensity" and our proposed method can still be used when A is nominal). Cluster 1 corresponds to treatment "non-responders", i.e. the potential utility scores remain unchanged regardless of the treatment "intensity". Cluster 2 contains subjects whose utility scores get larger as the treatment becomes more "intensive", and cluster 3 represents subjects whose optimal treatment is 2 and further increasing the treatment "intensity" leads to a lower utility score. The observed outcome Y is simulated from a normal distribution with mean $\sum_{a=1}^{3} E\{Y^*(a)\}I(A=a)$ and variance σ_Y^2. We consider different noise levels in covariates ($\sigma_{\mathbf{X}} = 0.5, 1$) and noise levels in the outcome ($\sigma_Y = 0.5, 1$), which reflect different degrees of cluster separability. In addition, we also vary the correlation between X_1 and X_2 ($\rho = 0$, 0.5, 0.8 conditional on the cluster assignment) to examine the influence of correlated covariates on the performance of our proposed method.

4.3.1.2 Scenario 2

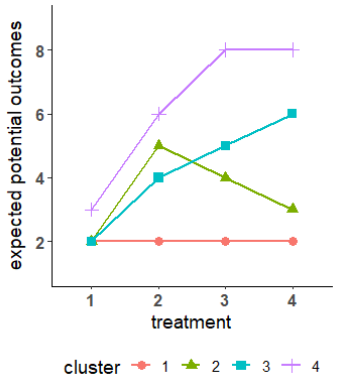
In the second setting, we consider a four-arm trial ($K = 4$) where treatment A is sampled uniformly from $\{1, 2, 3, 4\}$. There exist 4 subgroups/clusters, and the sizes of clusters 1-4 are $n/9$, $2n/9$, $2n/9$, and $4n/9$, respectively.

With no noise covariates We first consider the situation where there are four covariates, all of which inform the clustering structure (i.e. signal covariates). In particular, X_1 and X_2 are binary variables that take on the value of 0 or 1, X_3 is a categorical variable taking on values 0, 1, or 2, and X_4 is a normally-distributed continuous variable. In Table 4.2, we summarise the covariate profiles and potential treatment response profiles for 4 clusters. The treatment response profiles (by cluster) are also plotted in Figure 4.2.

Table 4.2: *Covariate profiles and potential outcome profiles in each cluster under scenario 2.*

	cluster 1	cluster 2	cluster 3	cluster 4
$P(X_1 = 1)$	0.2	0.4	0.6	0.8
$P(X_2 = 1)$	0.4	0.4	0.6	0.6
$P(X_3 = 0)$	0.1	0.2	0.3	0.4
$P(X_3 = 1)$	0.15	0.3	0.15	0.3
$E(X_4)$	2	4	8	6
$E\{Y^*(1)\}$	2	2	2	3
$E\{Y^*(2)\}$	2	5	4	6
$E\{Y^*(3)\}$	2	4	5	8
$E\{Y^*(4)\}$	2	3	6	8

Fig. 4.2: *Plot of the expected potential outcome values (by cluster) under different treatments in scenario 2.*



Similar to the first scenario, to facilitate the description of the potential outcome patterns in this setting, the outcome Y can be considered as a utility score and we can assume that the treatment becomes more "intensive" as A increases: cluster 1 represents treatment "non-responders"; cluster 2 contains subjects whose "maximum tolerance" is treatment 2, and the utility gradually decreases as the treatment becomes more "intensive", possibly due to the fact that the increased side-effects outweigh the added benefits by receiving a more "intensive" treatment; cluster 3 corresponds to subjects whose response to treatment gets better as the treatment level increases, and cluster 4 represents the "exceptional-responder" subgroup given that its members' average responses to all treatment levels are always better than other subgroups. The observed outcome Y follows a normal distribution with mean $\sum_{a=1}^{4} E\{Y^*(a)\}I(A = a)$ and variance σ_Y^2.

With noise covariates To demonstrate the variable selection feature of our proposed method, we modify the simulation setting described in the "with no noise covariates" paragraph and include two additional binary covariates, X_5 and X_6. The profiles of X_5 and X_6 are the same in all four clusters and thus neither

of them contributes to the formation of clusters (noise covariates). Specifically, $P(X_5 = 1) = 0.8$ and $P(X_6 = 1) = 0.15$ for all clusters. The profiles of other covariates and treatment responses are the same as those presented in Table 4.2.

We assess the performance of our proposed method under scenario 2 both when there are no noise covariates and when there are noise covariates. Four different combinations of noise levels in X_4 (the continuous covariate) and noise levels in the outcome Y are considered, namely, $(\sigma_Y, \sigma_{\mathbf{X}}) = (0.2, 0.2)$, $(0.5, 0.5)$, $(1, 0.5)$, $(1, 1)$.

4.3.2 Evaluation metrics

The performance of our proposed method in simulation studies is evaluated based on the following metrics:

(i) Estimated number of clusters, $\widehat{N}_{\text{cluster}}$.

(ii) Adjusted Rand index (ARI):
ARI (Hubert and Arabie, 1985) is a pair counting-based measure. It is the "corrected-for-chance" version of the Rand index (RI). We use C to denote the clustering found by our proposed method and G to denote the ground truth in simulation studies (i.e. ideal clustering). Let a be the number of data pairs that are clustered together in the same set in both C and G, and b be the number of data pairs that are in different sets in both C and G. RI is defined as:

$$\text{RI} = \frac{a + b}{n(n-1)/2}, \tag{4.12}$$

where the denominator is the total number of data pairs. One problem of RI is that the expected value of (4.12) for two random partitions is not a constant (e.g. 0). To correct this problem, Hubert and Arabie (1985) proposed an adjusted measure, ARI, which is given by

$$\text{ARI} = \frac{\text{RI} - E(\text{RI})}{\max(\text{RI}) - E(\text{RI})}. \tag{4.13}$$

The expected value of ARI is 0 under random clustering, and ARI is bounded above by 1. A larger value indicates a higher level of agreement between two partitions.

(iii) Conditional entropy-based external clustering evaluation metrics (Rosenberg and Hirschberg, 2007):

(a) Homogeneity:
Same as before, we denote the computed clustering by C and the ideal clustering (i.e. ground truth in simulation studies) by G. We will refer to G as the "class structure" to distinguish it from the computed clustering structure. The homogeneity is defined as:

$$\text{homogeneity} = \begin{cases} 1 & \text{if } H(G,C) = 0, \\ 1 - \frac{H(G|C)}{H(G,C)} & \text{otherwise,} \end{cases} \tag{4.14}$$

where $H(\bullet)$ is the entropy. In particular, $H(G|C)$ denotes the conditional entropy of G given C and quantifies the amount of additional information contained in G with respect to C. It is calculated as:

$$H(G|C) = -\sum_{c=1}^{|C|}\sum_{g=1}^{|G|}\frac{n_{gc}}{n}\log\left(\frac{n_{gc}}{\sum_{g=1}^{|G|} n_{gc}}\right), \tag{4.15}$$

where n_{gc} denotes the number of subjects that are from class g and clustered into cluster c, and $|G|$ and $|C|$ denote the cardinality of G (the number of classes according to the ground truth in simulation studies) and the cardinality of C (the estimated number of clusters), respectively. $H(G,C)$ is the joint entropy that is used for normalisation, and it is calculated as:

$$H(G,C) = -\sum_{c=1}^{|C|}\sum_{g=1}^{|G|}\frac{n_{gc}}{n}\log\left(\frac{n_{gc}}{n}\right). \tag{4.16}$$

The clustering is more homogeneous if less classes are included in one cluster. A perfectly homogeneous clustering corresponds to the case where each cluster contains only members of a single class (i.e. $H(G|C) = 0$). Homogeneity takes values between 0 and 1, and a larger value is more desirable.

(b) Completeness:
Completeness can be defined in a symmetric manner as follows:

$$\text{completeness} = \begin{cases} 1 & \text{if } H(C,G) = 0, \\ 1 - \frac{H(C|G)}{H(C,G)} & \text{otherwise,} \end{cases} \tag{4.17}$$

where

$$H(C|G) = -\sum_{g=1}^{|G|}\sum_{c=1}^{|C|}\frac{n_{gc}}{n}\log\left(\frac{n_{gc}}{\sum_{c=1}^{|C|}n_{gc}}\right), \tag{4.18}$$

and

$$H(C,G) = H(G,C) = -\sum_{g=1}^{|G|}\sum_{c=1}^{|C|}\frac{n_{gc}}{n}\log\left(\frac{n_{gc}}{n}\right). \tag{4.19}$$

The clustering is more complete if the members of a given class are allocated to less number of clusters. A perfectly complete clustering refers to the situation where all members of a given class are assigned to the same cluster (i.e. $H(C|G) = 0$). Completeness takes values between 0 and 1, and a larger value is preferred.

We note that sometimes the harmonic mean of homogeneity and completeness (referred to as "v-measure") is used to measure the clustering performance (Rosenberg and Hirschberg, 2007). However, we prefer using two separate measures (homogeneity and completeness) rather than the aggregated v-measure in order to better understand what (if any) type of mistake is made by the clustering algorithm.

4.3.3 Simulation results

In each simulation replicate, BART is run for 6000 iterations to predict the potential outcomes and profile regression is run for 2000 iterations to cluster subjects into subgroups. For both BART and profile regression, the first 1000 iterations are discarded as burn-in. We use the default values specified in the `BART` package and the `PReMiuMar` package (https://github.com/anarouanet/PReMiuMar) for all hyperparameters. Investigation of posterior samples suggests no evidence against convergence in our simulation studies. For both scenarios, the computational time per run is about 230 seconds when $n = 450$ and 440 seconds when $n = 900$ on a Windows-based computing system with 1 core, 3.40 GHz Intel processor.

4.3.3.1 Scenario 1

Simulation results for scenario 1 are presented in Table 4.3.

Table 4.3: *Simulation results based on 100 replicates: mean (SD) of adjusted Rand index (ARI), completeness, homogeneity, and the estimated number of clusters ($\widehat{N}_{cluster}$) under scenario 1 for different noise levels in the outcome (σ_Y), noise levels in covariates ($\sigma_{\mathbf{X}}$), and levels of correlations between covariates X_1 and X_2 (ρ) conditional on the cluster allocation.*

			$n = 450$				$n = 900$			
σ_Y	$\sigma_{\mathbf{X}}$	ρ	ARI	Completeness	Homogeneity	$\widehat{N}_{\text{cluster}}$	ARI	Completeness	Homogeneity	$\widehat{N}_{\text{cluster}}$
0.5	0.5	0	0.99 (0.01)	0.96 (0.03)	1.00 (0.00)	4.11 (0.96)	0.97 (0.02)	0.92 (0.04)	1.00 (0.00)	5.11 (0.87)
		0.5	0.98 (0.02)	0.95 (0.04)	1.00 (0.01)	4.17 (1.02)	0.97 (0.02)	0.93 (0.04)	1.00 (0.00)	5.00 (0.96)
		0.8	0.98 (0.02)	0.95 (0.04)	1.00 (0.01)	4.17 (1.01)	0.97 (0.02)	0.93 (0.04)	1.00 (0.00)	5.07 (0.98)
1	1	0	0.88 (0.07)	0.81 (0.08)	0.95 (0.02)	4.43 (0.88)	0.77 (0.09)	0.69 (0.07)	0.95 (0.01)	5.86 (0.94)
		0.5	0.88 (0.08)	0.82 (0.09)	0.97 (0.02)	4.44 (1.03)	0.80 (0.09)	0.73 (0.08)	0.96 (0.01)	5.76 (1.00)
		0.8	0.90 (0.08)	0.84 (0.09)	0.97 (0.02)	4.26 (0.77)	0.83 (0.08)	0.75 (0.07)	0.97 (0.01)	5.60 (0.95)

We observe that the results are fairly robust across different levels of correlations (ρ) among covariates. This is a highly desirable property of our proposed approach compared to many other standard approaches, whose performance can be very sensitive to the degree of correlations among predictors due to the multicollinearity problem (Dormann et al., 2013; Molitor et al., 2010; Sambandam, 2003).

The $\widehat{N}_{\text{cluster}}$ column shows that our proposed method over-estimates the true number of clusters ($N_{\text{cluster}} = 3$ in this scenario), especially when the sample size is large ($n = 900$). This is not surprising given that the profile regression adopts a Dirichlet process mixture model (DPMM): it has been demonstrated that when the true number of clusters is finite and small, the posterior inference on the number of clusters by using the DPMM may be inconsistent, and DPMM tends to over-estimate the true number of clusters and produce some small extraneous clusters around the true components (Lu et al., 2018; Miller and Harrison, 2013; Onogi et al., 2011; Yang et al., 2019). This phenomenon is commonly referred to as "over-clustering" in the literature (Lu et al., 2018), and we speculate that the "over-clustering" might be due to the sensitivity of DPMM to even minor deviations that exist among groups. We note that in order to demonstrate this point in our simulation studies, we do not set an upper bound on the number of clusters or place a constraint on the minimum cluster size. However, in practice, if the examination of the output suggests that some resulting clusters contain only a small number of subjects, or if we have *a priori* knowledge on the maximum number of subgroups that we are aiming for, such information can be incorporated into the post-processing step straightforwardly in the implementation (e.g. by specifying a value for "`maxNClusters`" in the "`calcOptimalClustering`" function in package `PReMiuMar`).

Not surprisingly, homogeneity is higher (better) than completeness in most cases, given that N_{cluster} is typically over-estimated by our method. In this sense, it is more likely that subjects who are clustered together are from the same class (the ground truth in simulation studies), and thus leading to "more homogeneous" clusters. On the other hand, small extraneous clusters (centred around true clusters) that are produced by DPMM may explain the "less complete" clustering results.

As expected, higher noise levels in covariates ($\sigma_{\mathbf{X}}$) and outcomes (σ_Y) result in worse clustering performance (i.e. lower ARI, completeness, and homogeneity, and a larger upward bias in the estimation of N_{cluster}), given that we would expect

the underlying clustering structures to be less clear (lower cluster separability) with larger values of $\sigma_{\mathbf{X}}$ and σ_Y.

A comparison of the results obtained when $n = 450$ with those obtained when $n = 900$ suggests that larger sample sizes do not seem to improve the performance of our method in this setting. When the noises in covariates and the outcome are large ($\sigma_Y = \sigma_{\mathbf{X}} = 1$), ARI and completeness even get worse as the sample size increases. One possible reason for this observation is that the over-estimation of N_{cluster} by DPMM is more pronounced when $n = 900$, and this has a subsequent (negative) effect on ARI and completeness.

In addition to measuring the overall clustering accuracy of our proposed method based on the clustering performance metrics described in Section 4.3.2, we also examine how well our method performs in terms of recovering the true underlying cluster-specific mean parameters. This is a useful and important criterion for evaluating patient stratification methods since a method that performs well in this respect can provide more informative insights into the underlying heterogeneity and a better characterisation of the resulting clusters. To this end, in each simulation replicate, we first obtain the posterior means of mean parameters for X_1, X_2, X_3, $Y^*(1)$, $Y^*(2)$, and $Y^*(3)$ in each resulting cluster, and then for each of X_1, X_2, X_3, $Y^*(1)$, $Y^*(2)$, and $Y^*(3)$, we construct a "pooled parameter sample" (across clusters) for its corresponding mean parameter. Specifically, the "pooled parameter sample" is obtained by "re-weighting" the estimated mean for each cluster by the corresponding cluster size (i.e. in the "pooled parameter sample", the number of copies of the estimated mean for a cluster equals the cluster size). We note that the "re-weighting" is important: estimated means for larger clusters should be assigned more weight since larger clusters carry more information, and "re-weighting" corrects for the "over-representation" or "under-representation" of clusters in the "pooled parameter sample". This is analogous to the use of the inverse probability weighting method to correct for the selection bias in a sample that is not representative of the target population.

The densities of the re-weighted results (i.e. the "pooled parameter sample") over 100 simulation replicates in low ($\sigma_Y = \sigma_{\mathbf{X}} = 0.5$) and high ($\sigma_Y = \sigma_{\mathbf{X}} = 1$) noise settings are plotted in Figures 4.3 (for covariates X_1, X_2, and X_3) and 4.4 (for potential outcomes under each treatment option). These plots correspond to the case with $n = 900$ and $\rho = 0.5$. Density plots corresponding to other n and ρ values look similar (do not alter our conclusions) and are thus omitted.

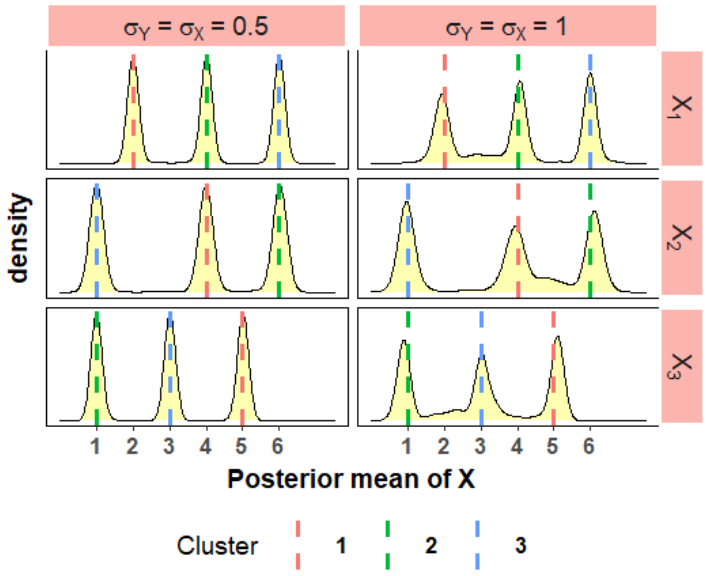


Fig. 4.3: *Density plots of cluster-specific mean parameters for* $\mathbf{X}$ *in scenario 1 with* $n = 900$ *and* $\rho = 0.5$*. Dashed lines refer to true values of* $E(X_1)$*,* $E(X_2)$*, and* $E(X_3)$ *in each cluster. The left and right panels correspond to the low and high noise settings, respectively.*

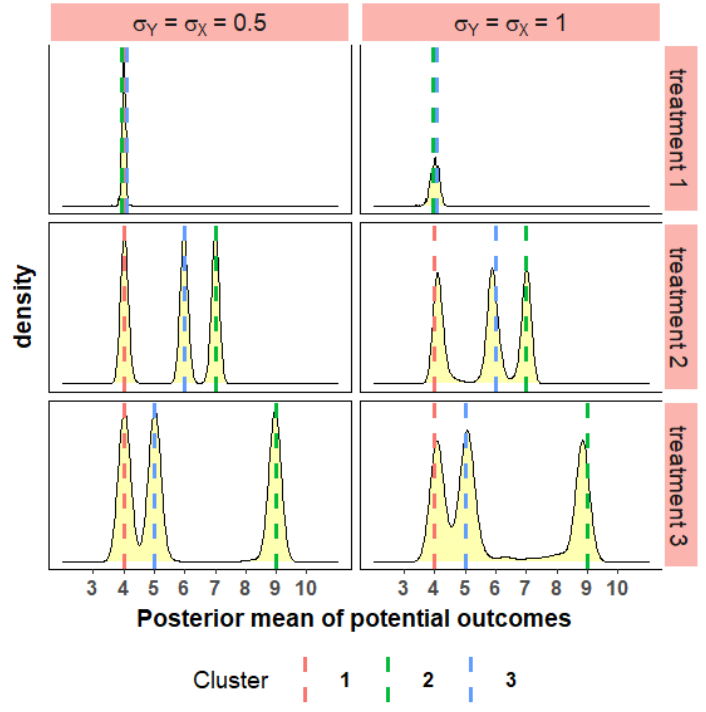


Fig. 4.4: *Density plots of cluster-specific mean parameters for potential outcomes under treatments 1, 2, and 3 in scenario 1 with* $n = 900$ *and* $\rho = 0.5$*. Dashed lines refer to true values of* $E\{Y^*(1)\}$*,* $E\{Y^*(2)\}$*, and* $E\{Y^*(3)\}$ *in each cluster. The left and right panels correspond to the low and high noise settings, respectively.*

Figures 4.3 and 4.4 imply that the true cluster-specific means (dashed lines) for all covariates and potential outcomes can be recovered by our proposed method. This does not contradict the observation that the estimated number of clusters is greater than the truth, given that superfluous clusters are typically small (and thus have small weights) and centred around the truth (as Miller and Harrison (2013) noted, "a good estimate of the density might include superfluous components having vanishingly small weight"). When the noise level is high ($\sigma_Y = \sigma_{\mathbf{X}} = 1$), the densities are flatter (nosier) compared to the case with low noise levels ($\sigma_Y = \sigma_{\mathbf{X}} = 0.5$), but our method still manages to recover the truth.

4.3.3.2 Scenario 2

With no noise covariates Simulation results under scenario 2 with no noise covariates are summarised in Table 4.4. As in the first scenario, the homogeneity score is almost always higher than the completeness score due to "over-clustering". When the noise level is low ($\sigma_Y = \sigma_{\mathbf{X}} = 0.2$), the clusters are well-separated and our proposed method performs almost perfectly in terms of ARI, completeness and homogeneity, despite the over-estimation of N_{cluster} remaining a problem (for this scenario, $N_{\text{cluster}} = 4$). As the noise level increases, the clustering performance gets worse, and it seems to be more sensitive to the change in $\sigma_{\mathbf{X}}$ than to the change in σ_Y (when we compare the results corresponding to $(\sigma_Y, \sigma_{\mathbf{X}}) = (1, 0.5)$ with those corresponding to $(\sigma_Y, \sigma_{\mathbf{X}}) = (1, 1)$). We also observe that $\widehat{N}_{\text{cluster}}$ first increases and then decreases as the noise gets larger, possibly because our proposed algorithm merges some clusters when the true underlying clustering structure has very low separability (i.e. very large noise).

With noise covariates We present the simulation results for scenario 2 with noise covariates in Table 4.5. The variable selection method discussed in Section 4.2.2.3 is employed and we fix all variable selection-related hyperparameters at their default values specified in the `PReMiuMar` package.

Table 4.4: *Simulation results based on 100 replicates: mean (SD) of adjusted Rand index (ARI), completeness, homogeneity, and the estimated number of clusters ($\widehat{N}_{cluster}$) under scenario 2 with no noise covariates for different noise levels in the outcome (σ_Y) and noise levels in the continuous covariate X_4 ($\sigma_{\mathbf{X}}$).*

		$n = 450$				$n = 900$			
σ_Y	$\sigma_{\mathbf{X}}$	ARI	Completeness	Homogeneity	$\widehat{N}_{\text{cluster}}$	ARI	Completeness	Homogeneity	$\widehat{N}_{\text{cluster}}$
0.2	0.2	0.99 (0.02)	0.99 (0.02)	0.99 (0.03)	4.67 (0.83)	0.99 (0.01)	0.98 (0.02)	1.00 (0.00)	5.59 (0.91)
0.5	0.5	0.84 (0.06)	0.79 (0.05)	0.89 (0.06)	5.97 (1.11)	0.84 (0.05)	0.75 (0.04)	0.91 (0.02)	7.28 (1.20)
1	0.5	0.81 (0.09)	0.78 (0.07)	0.86 (0.06)	5.69 (1.24)	0.82 (0.07)	0.75 (0.06)	0.91 (0.02)	6.92 (1.13)
1	1	0.40 (0.07)	0.45 (0.06)	0.50 (0.06)	5.04 (1.50)	0.37 (0.07)	0.42 (0.05)	0.53 (0.03)	6.08 (1.34)

Table 4.5: *Simulation results based on 100 replicates: mean (SD) of adjusted Rand index (ARI), completeness, homogeneity, and the estimated number of clusters ($\widehat{N}_{cluster}$) under scenario 2 with two additional noise covariates that do not define clustering structures for different noise levels in the outcome (σ_Y) and noise levels in the continuous covariate X_4 ($\sigma_{\mathbf{X}}$).*

		$n = 450$				$n = 900$			
σ_Y	$\sigma_{\mathbf{X}}$	ARI	Completeness	Homogeneity	$\widehat{N}_{\text{cluster}}$	ARI	Completeness	Homogeneity	$\widehat{N}_{\text{cluster}}$
0.2	0.2	1.00 (0.00)	1.00 (0.01)	1.00 (0.00)	4.08 (0.31)	1.00 (0.00)	1.00 (0.01)	1.00 (0.00)	4.46 (0.66)
0.5	0.5	0.68 (0.19)	0.87 (0.09)	0.67 (0.18)	3.43 (1.06)	0.77 (0.17)	0.86 (0.04)	0.76 (0.15)	3.88 (0.96)
1	0.5	0.54 (0.09)	0.88 (0.07)	0.54 (0.11)	2.84 (1.12)	0.66 (0.16)	0.87 (0.05)	0.68 (0.14)	3.57 (1.13)
1	1	0.29 (0.13)	0.47 (0.18)	0.30 (0.15)	2.94 (1.35)	0.32 (0.06)	0.50 (0.09)	0.35 (0.09)	3.21 (1.40)

Unlike in the case with no noise covariates where N_{cluster} is over-estimated for all $(\sigma_Y, \sigma_{\mathbf{X}})$ pairs, when two additional noise covariates (X_5 and X_6) that do not "drive" the clustering structure are included, $\widehat{N}_{\text{cluster}}$ is smaller than the truth ($N_{\text{cluster}} = 4$) except in the low noise setting ($\sigma_Y{=}\sigma_{\mathbf{X}} = 0.2$). This is not very surprising since the profiles of noise covariates are the same in all clusters, and if the noise level in signal covariates (e.g. X_4) is also fairly large (e.g. $\sigma_{\mathbf{X}} = 0.5$ or 1), the underlying clustering structure can be unclear and thus the clustering algorithm may merge clusters, which leads to the under-estimation of the actual number of clusters in the end.

We also find that when noise covariates are included, completeness is higher than homogeneity in most cases. These results are opposite to the findings from the case with no noise covariates (Table 4.4), and may be linked to the fact that the true number of clusters is under-estimated in the presence of noise covariates: subjects from different classes (ground truth) are incorrectly merged. Therefore, resulting clusters are "less homogeneous" but "more complete".

Another observation is that even though we conclude that increasing the sample size does not seem to noticeably improve the performance of our method under scenario 1 (Table 4.3) and scenario 2 with no noise covariates (Table 4.4), when noise covariates are present and the variable selection method is embedded into our proposed algorithm, we do see an improvement in the clustering performance (especially in terms of ARI and homogeneity) of our method with increasing the sample size (Table 4.5).

We also examine whether or not our proposed method with variable selection can distinguish the signal covariates from the noise covariates that do not influence the formation of clusters.

In Table 4.6, we summarise the mean and the standard deviation (across 100 simulation replicates) of posterior means of the variable selection parameter, ρ_j (the Bernoulli probability parameter discussed in Section 4.2.2.3), for $j = 1, \ldots, 6$. In addition, we plot the densities of posterior means of ρ_j, $j = 1, \ldots, 6$, in 100 simulation replicates with different sample sizes and different noise levels in Figure 4.5 (we omit the figures corresponding to $(\sigma_Y, \sigma_{\mathbf{X}}) = (1, 0.5)$, which look very similar to the plots for the case with $(\sigma_Y, \sigma_{\mathbf{X}}) = (0.5, 0.5)$).

Table 4.6: *Simulation results based on 100 replicates: mean (SD) of the posterior means of variable selection parameters (ρ_j) under scenario 2 with two additional noise covariates that do not define clustering structures. Different noise levels in the outcome (σ_Y) and noise levels in the continuous covariate X_4 ($\sigma_{\mathbf{X}}$) are examined.*

		$n = 450$						$n = 900$					
σ_Y	$\sigma_{\mathbf{X}}$	ρ_1	ρ_2	ρ_3	ρ_4	ρ_5	ρ_6	ρ_1	ρ_2	ρ_3	ρ_4	ρ_5	ρ_6
0.2	0.2	0.81 (0.04)	0.45 (0.23)	0.74 (0.12)	0.86 (0.03)	0.12 (0.10)	0.11 (0.10)	0.79 (0.04)	0.71 (0.17)	0.84 (0.08)	0.86 (0.03)	0.11 (0.09)	0.09 (0.05)
0.5	0.5	0.82 (0.10)	0.46 (0.25)	0.61 (0.22)	0.84 (0.10)	0.10 (0.05)	0.09 (0.03)	0.83 (0.05)	0.66 (0.20)	0.79 (0.11)	0.87 (0.04)	0.09 (0.05)	0.08 (0.05)
1	0.5	0.85 (0.11)	0.49 (0.25)	0.57 (0.21)	0.85 (0.10)	0.11 (0.08)	0.09 (0.04)	0.86 (0.05)	0.73 (0.18)	0.77 (0.12)	0.87 (0.03)	0.09 (0.05)	0.09 (0.04)
1	1	0.73 (0.28)	0.41 (0.26)	0.40 (0.26)	0.71 (0.29)	0.11 (0.04)	0.12 (0.08)	0.86 (0.08)	0.63 (0.22)	0.63 (0.21)	0.86 (0.08)	0.09 (0.05)	0.09 (0.06)

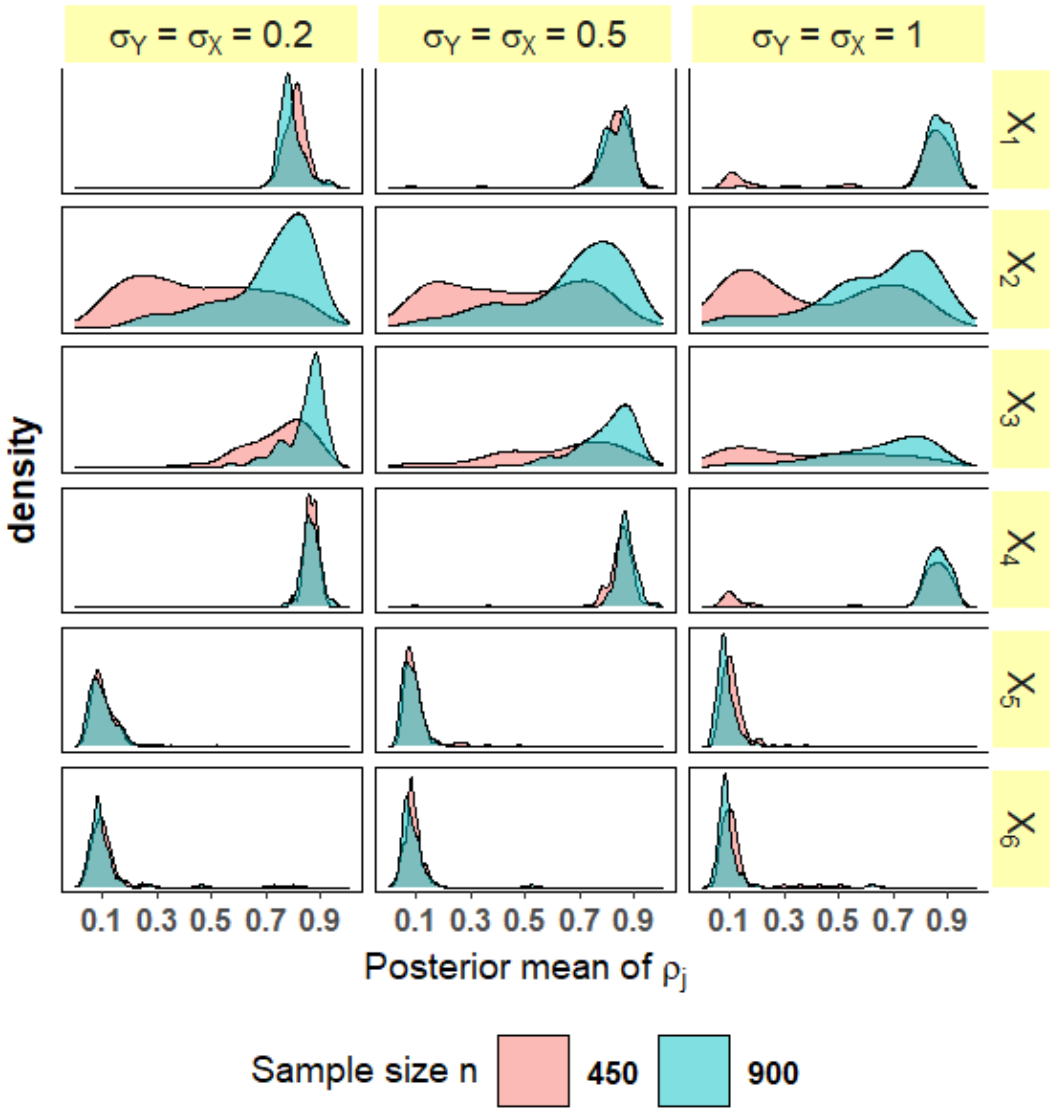


Fig. 4.5: *Density plots of the posterior means of variable selection parameters ρ_j, $j = 1, \ldots, 6$, under scenario 2 with two noise covariates (X_5 and X_6). The left, middle, and right panels correspond to the low, moderate, and high noise settings, respectively.*

Based on summary statistics (Table 4.6) and density plots (Figure 4.5), we conclude that in general, all variable selection results are consistent with the underlying simulation mechanism. The densities of ρ_5 and ρ_6 concentrate around 0.1 regardless of the sample sizes and noise levels, implying that our method is effective for detecting non-informative (noise) covariates (X_5 and X_6) whose profiles are the same in all four clusters and do not support the clustering structure. On the other hand, the densities of ρ_1 and ρ_4 concentrate around 0.9, suggesting that our method correctly identifies the highly-informative covariates (X_1 and X_4) that are important in "driving" the clustering components (the

covariate profiles of X_1 and X_4 are fairly different across four clusters under scenario 2, as shown in Table 4.2). The posterior means of ρ_2 and ρ_3 are not as high as those of ρ_1 and ρ_4, and the density plots corresponding to ρ_2 and ρ_3 also reveal more uncertainties within the range [0,1] compared to ρ_1 and ρ_4. This can be explained by the fact that the clustering information contained in X_2 and X_3 is less than that contained in X_1 and X_4 by design (Table 4.2). In other words, X_2 and X_3 play a less important role in defining the clustering structures compared to X_1 and X_4. For example, X_2 does not distinguish cluster 1 from cluster 2 ($P(X_2 = 1) = 0.4$ for clusters 1 and 2), nor does it differentiate between cluster 3 and cluster 4 ($P(X_2 = 1) = 0.6$ for clusters 3 and 4). Similarly, the between-cluster variation of X_3 is not as significant as that of X_1 or X_4 (X_3 is more informative than X_2 though).

In general, the variable selection results are fairly robust to different noise levels. This is especially true for two non-informative (noise) covariates, X_5 and X_6. For signal covariates, the posterior mean of ρ_j becomes slightly less stable (standard deviation estimates presented in Table 4.6 increase and the density plots in Figure 4.5 become flatter) as the data get noisier.

With regard to how the posterior mean of ρ_j is affected by the sample size n, we find that the conclusion depends on how informative the corresponding covariate (X_j) is on the clustering structure. For parameters corresponding to non-informative (X_5 and X_6) and strongly-informative (X_1 and X_4) covariates, the influence of the sample size on the inference is minimal and the densities for $n = 450$ and $n = 900$ almost overlap (the only exception is when $\sigma_Y = \sigma_{\mathbf{X}} = 1$, in which case we see some improvements in the selection results for X_1 and X_4 when $n = 900$, but our conclusion on the importance of X_1 and X_4 remains unchanged). In contrast, for parameters that correspond to weakly-informative covariates, X_2 and X_3, the improvement obtained by using a larger sample size is fairly significant: the posterior means of ρ_2 and ρ_3 are much larger (still smaller than those of ρ_1 and ρ_4 though) when $n = 900$ compared to their counterparts when $n = 450$ (Table 4.6), and we also observe that the concentration of the density shifts towards 1 in Figure 4.5 as the sample size increases from 450 (pink) to 900 (blue). These findings imply that with a larger sample size, our proposed method becomes more certain (i.e. does better in the recovery of the truth) that X_2 and X_3 also "drive" the clustering although their effects are relatively small compared to X_1 and X_4.

As an aside, we note that ρ_j takes values on a continuous scale between 0 and 1, and describing variable selection results based on ρ_j can be more advantageous

than using classical variable selection methods that only output a binary decision of "select" or "not select". The relative magnitude of posterior summary statistics (e.g. posterior mean) of ρ_j reflects to a great extent the relative importance of different covariates, as demonstrated by this simulated example. The final decision regarding whether a variable should be kept or removed can be made by specifying a cut-off value (for posterior summary statistics of ρ_j) that may depend on how sparse the model should be for the specific application. In addition, the comparison of the results in Table 4.5 and those in Table 4.4 demonstrates that even with the variable selection procedure embedded, our method's clustering performance gets worse in the presence of noise covariates (except when $\sigma_Y = \sigma_{\mathbf{X}} = 0.2$). In practice, in order to achieve a better (more accurate) clustering of the data when there are noise covariates, we can take a two-stage approach where in the first stage, we identify covariates that are important for distinguishing different clusters based on posterior summary statistics of ρ_j and then in the second stage, we re-fit the model (without variable selection) using only those important covariates selected in the first stage (Zhou et al., 2017).

4.4 Application to the INTERVAL trial

We demonstrate the utility of our proposed patient stratification approach by applying it to the INTERVAL data. The purpose of this analysis is to uncover donor subgroups with different baseline characteristics and potential "treatment" (inter-donation interval in the blood donation context) response profiles.

Same as in the previous chapter, we use the data from 884 female donors in the INTERVAL trial who were younger than 40 and had O negative blood type (a "much-in-demand but vulnerable" donor population). The three randomised groups in this trial for female donors are 16-week, 14-week, and 12-week inter-donation intervals. We recall that in Section 3.4.2, we performed a profile regression analysis in order to better understand which donor characteristics "drive" the decision of whether or not a donor is capable of giving blood more frequently than the standard practice. In that analysis, the outcome is a binary variable that is defined according to the ITR estimated by SR learning (binary outcome = 1 if SR learning recommends the donor to give blood more often that every 16 weeks, and = 0 if SR learning recommends the donor to donate every 16 weeks). For the analysis presented in this section, the utility score ($b = 3$) that

"discounts" the total units of blood collected by the increased incidences of low Hb deferrals is used as the target outcome. In Section 3.4.2, we have identified seven baseline donor characteristics (Figure 3.12 (a)) that "drive" the clustering structure (when the target outcome is binary) for this "much-in-demand but vulnerable" donor population, including the Short Form Health Survey version 2 (SF-36v2) physical component score (PCS), mental component score (MCS), ferritin level, red blood cell count (RBC), mean corpuscular volume (MCV), mean corpuscular haemoglobin (MCH), and body mass index (BMI). In this section, we use these seven "important" donor characteristics as profile variables in the profile regression model given that they are potentially informative of the formation of donor subgroups with different donation capacities. Restricting to this "pre-selected" subset of donor characteristics also enhances the interpretability of the clustering result.

In our analysis, we use the default setups that are specified in the `BART` package and the `PReMiuMar` package for priors and hyperparameters. We run BART for 6000 MCMC iterations with an initial burn-in of 1000 iterations to predict potential outcomes, and run profile regression for 40000 iterations with a burn-in of 10000 iterations in the clustering step.

In real data applications, it is particularly important to assess the convergence of Bayesian models, since in general, the signal in real data is not as strong as that in simulation studies (Hastie et al., 2015). We note that due to the well-recognised "label-switching" issue in Bayesian mixture models (Jasra et al., 2005), assessing the convergence based on posterior samples of cluster-specific parameters is not feasible. To our knowledge, finding a reliable indicator of the overall convergence remains a challenging problem and there is no perfect way of demonstrating convergence (Hastie et al., 2015; Liverani et al., 2015). However, monitoring the distribution of the concentration parameter α across multiple chains that are initialised with different numbers of clusters may serve as a preliminary diagnostic tool for assessing whether or not convergence can be reasonably expected (Hastie et al., 2015).

In Figure 4.6, we present the boxplot of the posterior distributions of α for five different initial numbers of clusters (10, 15, 20, 25, and 30). Following Hastie et al. (2015), for each initial number of clusters, we run profile regression three times (with random initialisations of other parameters). We observe that the posterior distributions of α from different initialisations are virtually identical, which implies that there is no strong evidence against convergence. The visual inspection of the traceplot (Figure 4.7) of α (posterior) from five independent

chains also suggests that multiple chains initialised with different conditions mix well and convergence is not an issue of major concern in our data analysis.

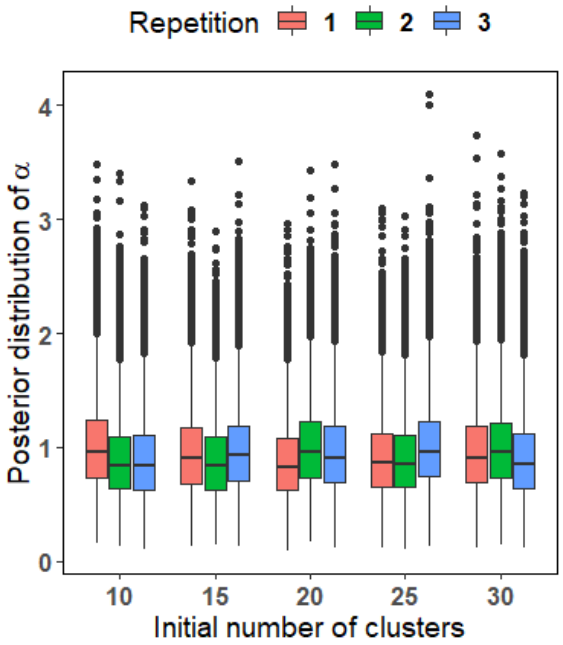


Fig. 4.6: *Convergence diagnostics: boxplot of the posterior distribution of α (30000 sweeps after a burn-in of 10000 samples) in the real data application for five different numbers of initial clusters with three repetitions per initialisation.*

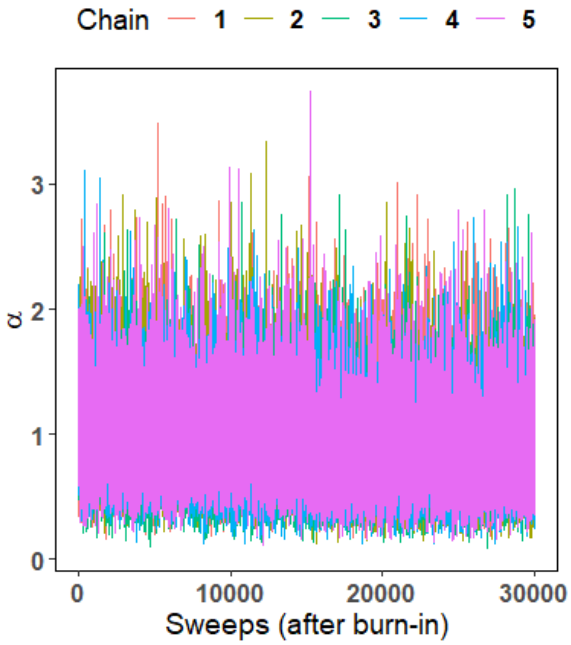


Fig. 4.7: *Convergence diagnostics: traceplot of the posterior of α in the real data application for five different chains (the numbers of initial clusters in these five chains are 10, 15, 20, 25, and 30, respectively).*

Simulation results presented in Section 4.3.3 suggest that the noise levels in covariates and the outcome can have a notable impact on the performance of our proposed patient stratification method. Unlike in the simulation studies where we can evaluate the performance of our method by calculating external metrics such as ARI, homogeneity, and completeness, in the real data application, the ground truth is not known and thus these external validation metrics cannot be used for assessing how well the method performs. However, we can still get some sense of "how strong the clustering signal is in this dataset" and "how reliable the clustering results are" by examining the MCMC output based on the idea discussed in Section 4.2.2.2. In addition, we assess whether or not our proposed approach can give insights into donor heterogeneity and stratify donors into clinically meaningful subgroups based on the final "representative" clustering identified by our method.

Inspection of the raw output from the profile regression model (before applying the post-processing method) suggests that the clustering is fairly consistent across MCMC iterations (implying that the clustering signal in this dataset is fairly strong and the MCMC output reveals high confidence in the clustering result).

We apply the post-processing method discussed in Section 4.2.2.2 to identify the "representative" clustering based on posterior samples. Figure 4.8 presents the heatmap of the posterior similarity matrix for 884 female donors included in our analysis. We have reordered the observations such that the pairs of donors who tend to share the same cluster memberships appear together.

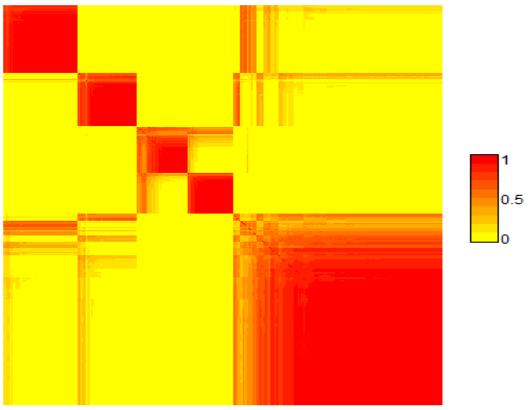


Fig. 4.8: *Heatmap of 884 female donors' posterior similarity matrix (across 30000 posterior draws) based on the profile regression model when the utility score with $b = 3$ is the target outcome. Different colours correspond to different degrees of subjects' similarity to each other (yellow - low similarity; red - high similarity).*

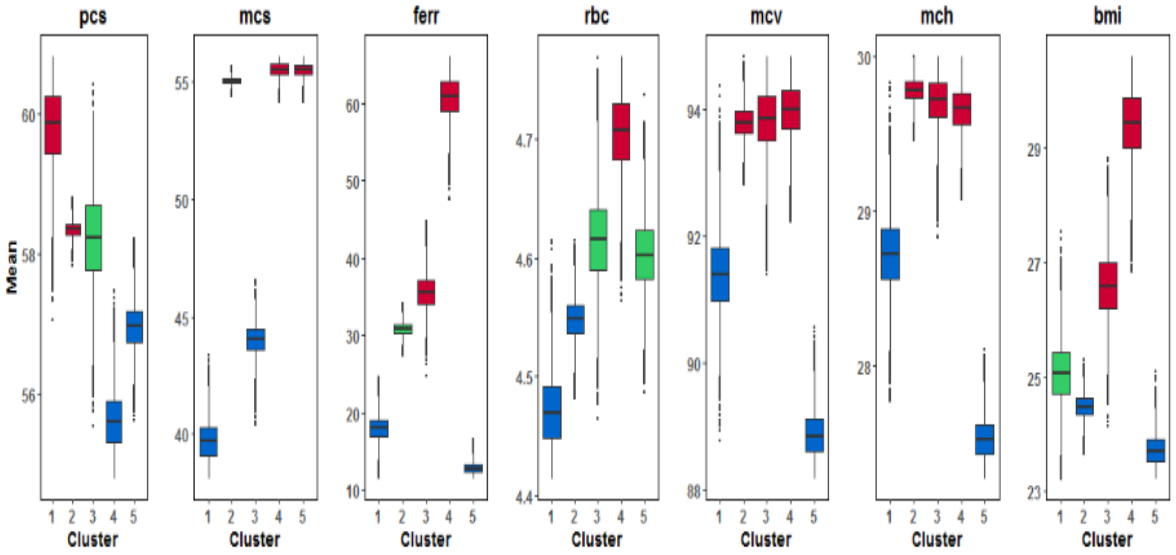
The heatmap reveals that the best partition of the data identified by our proposed approach consists of 5 donor subgroups (clusters), and the sizes of these subgroups are 171, 126, 101, 93, and 393, respectively.

Figure 4.9 shows the covariate profiles (posterior distributions of the mean parameters for donors' baseline characteristics) and the potential outcome profiles (posterior distributions of the mean parameters for potential outcomes under 16-, 14-, and 12-week inter-donation intervals) corresponding to each of the five clusters. These results reveal considerable evidence for the presence of heterogeneity within the donor population under investigation.

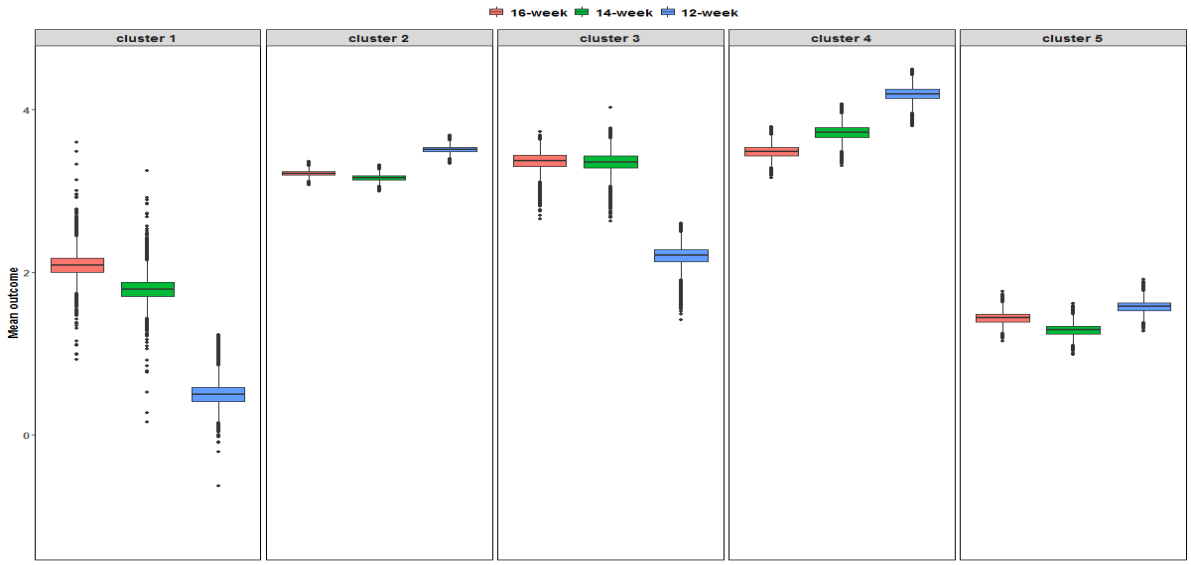
Cluster 4 (with 126 donors) represents the "super donor" subgroup: donors in this cluster are able to give blood more frequently than the current practice and for these donors, more frequent donations lead to larger utilities (Figure 4.9 (b)). Comparing across clusters, we observe that this group of donors has higher donation capacity since the utility scores of donors in cluster 4 are on average higher than those of donors in the other four clusters, especially under the 14-week and 12-week inter-donation intervals. These "super donors" are characterised by having high levels of ferritin, RBC, BMI, MCS, MCV and MCH and low levels of PCS. In particular, the most distinguishing feature that differentiates cluster 4 from the remaining clusters is the ferritin level (significantly higher in cluster 4). This is consistent with *a priori* expectation that donors with higher ferritin levels are in general more capable of donating blood more often (especially for female donors).

Cluster 2 is the largest subgroup with 393 donors. Figure 4.9 (b) suggests that donors in this subgroup are capable of donating blood every 12 weeks, even though the gain in the utility score by switching from the 16-week inter-donation interval to the 12-week inter-donation interval is only moderate and not as significant compared to cluster 4 (the "super donor" subgroup). If there is a shortage of blood, donors in cluster 4 may be encouraged to donate more frequently first, followed by donors in cluster 2. Consistent with cluster 4, donors in cluster 2 on average have high MCS (this may imply that they are highly-motivated donors), MCV, and MCH. However, their ferritin levels, RBC and BMI are lower than those of donors in cluster 4 (Figure 4.9 (a)).

Cluster 1 (with 101 donors) corresponds to the "brittle donor" subgroup. Donors in this subgroup cannot donate more frequently than every 16 weeks and the utility score gets smaller as the inter-donation interval gets shorter (Figure 4.9 (b)). The low ferritin levels, RBC and MCS of these donors (Figure 4.9 (a)) may explain why they are vulnerable.



(a) *Covariate profiles from the profile regression model: posterior distributions of the mean parameters for clusters 1-5. The red-coloured and the blue-coloured boxes indicate that the 90% credible intervals for the cluster-specific mean are above and below the average values across clusters 1-5, respectively. The green-coloured boxes imply that the 90% credible intervals for the cluster-specific mean include the average. We note that the boxes cover the interquartile range (lower hinge: the 25th percentile; upper hinge: the 75th percentile).*



(b) *Potential outcome profiles from the profile regression model: posterior distributions of the mean parameters for clusters 1-5 under the 16-week (pink), 14-week (green) and 12-week (blue) inter-donation intervals.*

Fig. 4.9: *Profile regression plots obtained by applying our proposed patient stratification method to the data from 884 female donors who were younger than 40 and had O negative blood type in the INTERVAL trial: posterior distributions of the parameters associated with the response (utility score) and covariates (baseline characteristics) for the "representative" clustering.*

Cluster 5 (with 171 donors) also represents a subgroup of donors with low donation capacity (a "willing but not able" subgroup). The utility scores of donors in this subgroup are similar under three inter-donation interval options and they are lower than the utility scores of donors in clusters 2, 3, and 4 (Figure 4.9 (b)). Donors in cluster 5 (on average) have a high MCS similar to that of donors in cluster 4 (i.e. the "super donor" subgroup), suggesting that they may be highly-motivated donors. However, low values of donation capacity-related characteristics such as ferritin levels, MCV, MCH, and BMI (Figure 4.9 (a)) may lead to higher than average deferral rates and lower than average utility scores in this subgroup (cluster 5).

The (potential) utility scores of donors in cluster 3 (with 93 donors) are almost identical under 16-week and 14-week inter-donation intervals but decrease significantly when the donation frequency is every 12 weeks (Figure 4.9 (b)). Therefore, we would assign these donors to the 16-week inter-donation interval, given that if the utility scores under two inter-donation interval options are similar, the less "intensive" inter-donation interval would be more desirable in reducing the risk of low Hb deferrals. The ferritin levels, MCV, MCH, and BMI are relatively high (Figure 4.9 (a)) for these donors, and the drop in the utility score when the inter-donation interval is 12-week may be attributed to the low MCS (it is likely that frequent donations demotivate donors with lower MCS from returning to future donation sessions to a higher degree).

To summarise, our analysis of the INTERVAL data using the proposed approach reveals clinically meaningful partition of the "much-in-demand but vulnerable" donor population under investigation. These results also demonstrate the gain of investigating patient stratification (subgroup level) in addition to estimating the optimal ITR (individual level). Of course, based on the estimated optimal ITRs, $\widehat{\mathcal{D}^*}(\mathbf{x})$, we can stratify the donor population into three subgroups where subgroups 1, 2, and 3 represent donors with $\widehat{\mathcal{D}^*}(\mathbf{x}) =$ 16-week, $\widehat{\mathcal{D}^*}(\mathbf{x}) =$ 14-week, and $\widehat{\mathcal{D}^*}(\mathbf{x}) =$ 12-week, respectively. However, such a stratification does not incorporate all the covariate information (even if covariates are included in the ITR estimation model) and the complete patterns of potential outcomes under all inter-donation interval options, and thus the granularity will be low. For example, the stratification may fail to capture the heterogeneity in the baseline characteristics of donors whose estimated optimal inter-donation intervals are the same. In addition, even though all donors in subgroup 3 achieve the highest utility scores when assigned to the 12-week inter-donation interval, their complete potential outcome patterns might differ (e.g. outcome patterns of some donors

may be consistent with those of cluster 2 in Figure 4.9 (b) and outcome patterns of other donors may be consistent with those of cluster 4 in Figure 4.9 (b)). In these situations, a further split of the three subgroups obtained solely based on $\widehat{\mathcal{D}^*}(\mathbf{x})$ may offer additional insights into the underlying donor heterogeneity. On the other hand, our proposed method has a higher "resolution" and enables us to get a more refined stratification of the donor population directly by the joint modelling of the predicted multivariate outcome (a vector of the potential outcome under each inter-donation interval) and the covariates. We note that if some of the resulting subgroups are fairly similar from the clinical perspective (in terms of clinical meaningfulness), they can be merged *a posteriori*. From our point of view, this would be preferable to starting with a method of low "resolution", in which case some useful information on the population heterogeneity may be overlooked.

4.5 Discussion

The uncovering of homogeneous subgroups from a heterogeneous population plays an important role in personalised medicine applications. In this chapter, we propose a two-stage patient stratification approach that leverages Bayesian nonparametric clustering techniques. Our proposed method captures the heterogeneity in the underlying population and partitions the population into subgroups of subjects who share similar covariate profiles and display similar treatment responses. Specifically, in the first stage, we predict the potential outcome under each treatment arm, and in the second stage, we apply profile regression to link the multivariate potential outcome vector to a set of covariates (can be continuous, discrete, or a mix of continuous and discrete ones) through cluster memberships (Molitor et al., 2010). Based on the posterior samples, the resulting subgroups/clusters can be characterised in terms of covariate profiles and potential treatment response profiles.

Our method offers several advantages. Firstly, while most existing methods for subgroup identification only cover two-treatment cases, our proposed approach is applicable to multi-arm trials. Secondly, the use of the Dirichlet process prior allows the number of clusters to be estimated from the data, thus bypassing the need for pre-specifying it. Thirdly, our method can properly handle correlated covariates (avoiding well-known problems caused by multicollinearity), which are common in clinical studies. Fourthly, a variable selection procedure is

embedded into our model for identifying important covariates that actively "drive" the clustering structure (i.e. contribute significantly to the cluster patterns). Fourthly, a variable selection procedure is embedded into our model for identifying important covariates that actively "drive" the clustering components (i.e. contribute significantly to the cluster patterns). Lastly, the proposed approach is built under the Bayesian framework and takes into account model uncertainties (Molitor et al., 2010).

The application of our proposed method to a subset of the INTERVAL data (a "much-in-demand but vulnerable" donor population) identifies 5 clinically meaningful donor subgroups with different donation capacities and covariate (donors' baseline characteristics) profiles. These results provide insight into the underlying donor heterogeneity by highlighting the differences between donors in terms of both baseline characteristics and potential response (to three different inter-donation intervals) patterns, and can be leveraged to inform and guide targeted donor recruitment and donor management strategy. For example, donors who are identified as belonging to the "super donor" subgroup may be asked to give blood more frequently if there is a blood shortage or if their blood group is rare or universal. In contrast, donors who belong to the "brittle donor" subgroup will be allowed longer time between donations to ensure donor health and safety (Blood and Transplant Research Unit, 2019).

As has been discussed in Section 4.2.1, although we use BART in the first stage of our method to predict the potential outcome under each treatment option, the validity of our approach does not rely on any particular choice of the prediction model, and BART can be replaced by other sufficiently flexible models (e.g. super learner proposed by van der Laan et al. (2007)).

The use of the Dirichlet process prior in the second-stage profile regression model allows the number of clusters/subgroups to be discovered in a data-driven way. However, as demonstrated by our simulation studies, Dirichlet process mixture model (DPMM) tends to over-estimate the number of clusters (produce some superfluous small-sized clusters). Indeed, the inconsistent inference (over-estimation) on the number of components by DPMM is a well-known problem (referred to as "over-clustering" in Lu et al. (2018)) that has been empirically observed and reported in the literature before, and it appears that in most cases, the extra clusters/subgroups are centred around the true components and only include a very small number of subjects (Miller and Harrison, 2013; Onogi et al., 2011). To our knowledge, how to correct for such inconsistency remains an open question in the field of Bayesian mixture modelling (Miller and Harrison, 2013;

Yang et al., 2019). Guha et al. (2019) recently proposed a truncation method to resolve the issue, but the success of their approach requires the provision of certain information that is typically unknown in practice (Yang et al., 2019). Another proposal is to reduce "over-clustering" by penalising small clusters (Lu et al., 2018). However, as can be expected, this approach requires careful tuning of the parameter that controls how much small clusters are penalised. In general, we think that a slight over-estimation of the true number of clusters is not an issue of major concern in the context of patient stratification, since in this case, the primary interest typically lies in the characterisation and the interpretation of subgroups rather than the inference on the exact number of underlying subgroups. As has been noted by Onogi et al. (2011), even though the extra clusters produced by DPMM are considered as redundant and interpreted as over-estimation in simulation studies, they may provide useful information in real data applications since they reflect some relatively subtle heterogeneity that might be clinically interesting. In practice, in order to achieve better interpretability of the clustering results, the model-based best partition should be coupled with practitioners' subject-matter knowledge when determining the optimal number of subgroups. If the inspection of the output indicates that the small clusters produced by DPMM are not of much clinical interest because they do not reflect a general pattern, the other larger and more representative clusters will be given more emphasis in the interpretation. In addition, if the profiles of some small clusters are very similar to those of some much larger clusters in terms of clinical meaningfulness, we can merge them *a posteriori*.

Chapter 5

Discussion

5.1 Summary

Personalised medicine is a paradigm that leverages patient heterogeneity and tailors medical intervention according to individual characteristics in order to improve health outcomes through data-driven, evidence-based, and scientifically-rigorous approaches. In this thesis, we focus on statistical methods that can be used to estimate the optimal ITR or stratify the patient population into subgroups in trials with more than two treatment arms, which are common in clinical practice (Baron et al., 2013).

In Chapter 2, we review methods that can be applied for estimating the optimal ITR in multi-arm trials and are computationally affordable in large-scale trials, including l_1-penalised least squares, adaptive contrast weighted learning, direct learning, and Bayesian additive regression trees. To evaluate their empirical performance in the context of large-scale trials with more than two treatments, we conduct extensive simulation studies under a variety of scenarios (different types of decision boundaries, different signal and noise levels, different levels of correlation among covariates, and varying numbers of baseline covariates). We also apply these methods to the data from male donors in the INTERVAL trial to estimate each donor's optimal inter-donation interval based on their individual baseline characteristics under the objectives of (i) maximising the total units of blood collection, (ii) minimising the low Hb deferral rates, and (iii) maximising a utility score which strikes a balance between the amount of blood collected by the blood service and the low Hb deferrals.

We treat the inter-donation interval (randomised group) in the INTERVAL trial as a nominal variable in Chapter 2, i.e. we assume that the three options

of the inter-donation interval (e.g. 12-week, 10-week, 8-week for male donors) have no intrinsic orderings. However, they are indeed "ordered" in terms of the donation intensity, since the 8-week inter-donation interval is shorter (i.e. more frequent and more "intensive") than the 10-week option, which is shorter than the 12-week option. Most existing statistical methods to estimate the optimal ITR cannot handle ordinal arms, and we would expect that ignoring the ordinality information may result in suboptimal decisions. Motivated by this limitation, in Chapter 3, we develop a method that accounts for treatment orderings and estimates the optimal ITR in trials with ordinal treatments. The proposed method, sequential re-estimation (SR) learning, decomposes the ordinal problem into a sequence of binary subproblems, each of which can be solved using existing methods designed for the estimation of the optimal ITR in two-arm trials. Then, multiple binary decisions are aggregated into the ordinal prediction based on decision trees. We also consider the situation where a large number of baseline covariates are available, but some of them may not be relevant for making treatment decisions (noise covariates). In this case, noise covariates may impair the model performance. We therefore extend the proposed SR learning framework and incorporate the variable selection feature in order to identify key covariates that inform decision-making. Both simulation studies and an application to the data from a "much-in-demand but vulnerable" donor population in the INTERVAL trial demonstrate the superior performance of SR learning over existing methods that assume multiple nominal treatments.

We switch our focus from the individual level to the subgroup level in Chapter 4 and propose a novel two-stage approach for patient stratification based on the data from multi-arm trials. In particular, our proposed method builds on a Bayesian framework for clustering with a Dirichlet process prior, which partitions subjects into subgroups according to treatment response profiles and covariate profiles without the need for prespecifying the number of subgroups. Our model also enables us to differentiate the covariates that are key "drivers" of the clustering results (i.e. signal covariates) from the covariates that do not inform the clustering structures (i.e. noise covariates) through variable selection. The utility of the proposed patient stratification method is demonstrated via simulation studies and an application to the data from the same donor population as that considered in Chapter 3 (i.e. a "much-in-demand but vulnerable" donor population).

5.2 Main contributions and the significance of this work

It is worth highlighting that even though this thesis was motivated by the INTERVAL blood donation trial and we use the INTERVAL data to demonstrate the utility of different approaches throughout this thesis, the multi-arm ITR estimation or patient stratification methods discussed in Chapters 2-4 can be applied more broadly for tailoring in multi-arm trial settings.

In the following, we summarise the contributions and implications of this thesis in two aspects: the methodological innovations that are generally applicable to many different disease and clinical areas (after being tailored to the specific substantive problem), and the practical significance of this work that is specific to the blood donation context.

From a methodological point of view that is relevant to the general implementation of personalised medicine-based approaches in practice, hopefully the work presented in Chapter 2 can provide general guidance on the choice of methods to practitioners who are interested in implementing these methods for ITR estimation in multi-arm trials (especially when the sample size of the trial is very large, as in the INTERVAL trial). In addition, Chapters 3 and 4 offer novel ways for ITR estimation among ordinal arms and for patient stratification in multi-arm trial settings, respectively. We hope that these proposed methods will further stimulate the applications of personalised medicine in situations where some added complexities of the dataset (e.g. multiple nominal or ordinal arms) limit the applicability of most existing methods that were primarily designed for two-arm trials. We have made the R/Matlab codes for implementing all the methods discussed in this thesis fully available to promote and facilitate their use in practice.

On the other hand, from the perspective of what impact this work may have on future NHSBT policies and the potential deployment of personalised donation strategies within NHSBT's routine service provision, we hope that the findings from our applications of personalised medicine-based approaches to the INTERVAL data can provide some initial evidence that personalised donation strategies may potentially lead to more sustainable blood donations compared to the current clinical practice. Hopefully, this work will support, advance, and provide further guidance on the implementation of more personalised approaches to blood donation (including personalising inter-donation intervals based on

donor-specific characteristics, stratifying donors into subgroups based on their capacity to give blood, and tailoring inter-donation intervals to different donor subgroups), especially after external validation is performed and practicable algorithms and implementation tools are developed (we will discuss the validation and the implementation in Sections 5.3.1 and 5.4.1.5, respectively). We note that there are also other aspects of donation policies that can be enhanced by personalised approaches and are worth exploring. For example, NHSBT currently uses uniform gender-specific thresholds for pre-donation Hb levels to defer donors (135g/L for males and 125g/L for females), the investigation of donor-specific/subgroup-specific pre-donation Hb cutoffs for deferral would be of interest. However, such an investigation is beyond the scope of this thesis.

5.3 Further considerations for ITR estimation

5.3.1 Validation of the estimated ITRs

Our proposed personalised donation strategies are intended for assigning future donors to their optimal inter-donation intervals. Despite the empirical evidence that such personalised rules are likely to be beneficial to blood donors, validation is essential before applying estimated rules to future donors. We note that the INTERVAL data cannot be used to validate the estimated optimal donation strategies, since we will suffer from optimism bias by doing this. The validation can be challenging partly because we can only observe a donor's outcome under one inter-donation interval, unless it is a cross-over design. One way to validate the estimated regime is by conducting a confirmatory trial (proof-of-concept) that compares estimated personalised donation strategies with the current clinical practice. Based on the data collected from this follow-up trial, we can evaluate whether or not our proposed personalised donation strategies lead to improved outcomes (i.e. increased units of blood collected by the blood service and/or reduced low Hb deferrals rates) over the fixed donation strategy that is currently adopted by the NHSBT.

5.3.2 Generalisability of the estimated ITRs

Another issue that warrants highlighting is the generalisability of estimated ITRs to a broader population. In general, the entry criteria for clinical trials are restrictive, and trial participants may not be representative of the more general

population. Therefore, extra caution should be taken when generalising ITRs estimated using trial data to a broader population (Gunter et al., 2011; Zhao et al., 2019b). However, this may not be a major problem for the INTERVAL trial. As suggested by Moore et al. (2016), participants in the INTERVAL trial were broadly representative of the national donor population of England, and thus it is likely that estimated ITRs based on the INTERVAL data are generalisable to the general donor population.

We also comment that the validation and generalisability issues discussed in this section regarding the estimated optimal ITRs are relevant not only to the INTERVAL trial but also more generally to other trials (Gunter et al., 2011; Zhao et al., 2019b; Zhou et al., 2017).

5.4 Future research directions

The work in this thesis has raised new research questions that are worthy of further investigation. In the following, we discuss some potential directions for future work.

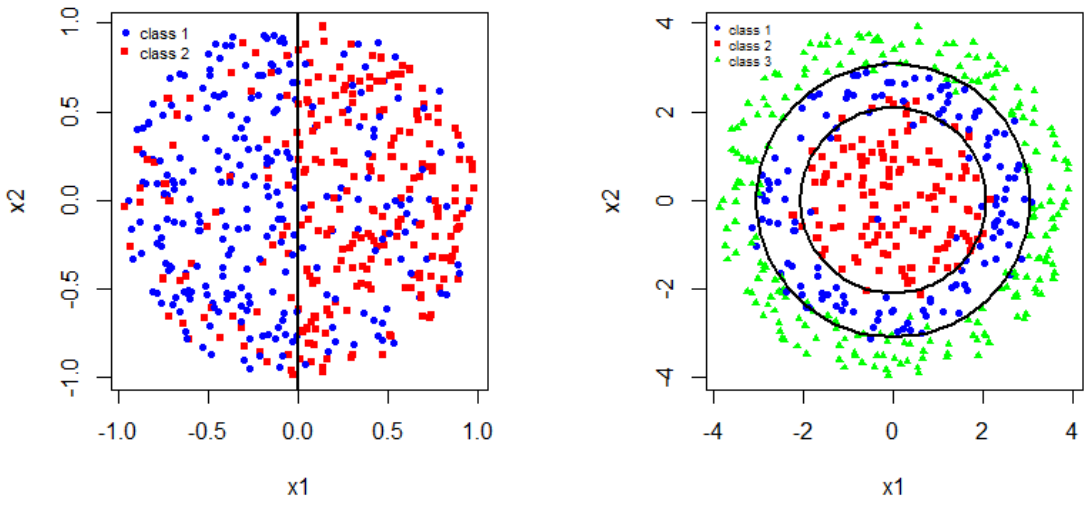
5.4.1 Chapters 2 and 3 - ITR estimation (individual level)

5.4.1.1 Uncertainty quantification of the estimated ITRs

Except for the Bayesian ITR estimation method (BART) discussed in Chapter 2, the rest of the methods used for ITR estimation in this thesis (including our proposed SR learning) only provide a point estimate (decision label) of $\mathcal{D}^*(\mathbf{X})$ and do not come with its uncertainty measure, since the models from which ITR estimators are derived are typically complex and thus it is statistically challenging to quantify the uncertainty directly (Laber and Qian, 2017). However, in medical applications, communicating the uncertainty associated with treatment recommendations prior to treatment delivery can be crucial, since uncertainty measures could help clinicians understand the strength of the evidence associated with the optimal treatment recommended by statistical models, and they can combine this information with their clinical expertise to make a final treatment decision that they believe will benefit the patient most (Kruppa et al., 2014a,b; Laber and Qian, 2017; Malley et al., 2012).

We can provide an informative and meaningful uncertainty quantification by estimating the conditional probability of different treatment options being the

best given individual-specific baseline characteristics. Loosely speaking, such probability estimates deliver the confidence we have in a specific treatment recommendation. To better demonstrate why this is important, we present two simulated examples that mimic the possible relationships between patients' true optimal treatments (classes) and their baseline characteristics in Figure 5.1.



(a) *Two classes* **(b)** *Three classes*

Fig. 5.1: *Simulated examples demonstrating the possible relationships between baseline covariates and the true optimal treatment (class) labels in practice. Black solid lines are Bayes classification boundaries. Different colours and symbols reflect the different classes to which data points belong.*

We can assume that X_1 and X_2 represent a patient's age and BMI, respectively (both are rescaled to the range $[-1, 1]$). For the case in Figure 5.1 (a), the true decision boundary is $X_1 = 0$, and the Bayes classification rule is: if $X_1 < 0$, then recommend treatment (class) 1, otherwise, recommend treatment (class) 2. However, it is obvious that there are uncertainties associated with the treatment (class) labels predicted according to the Bayes classification rule (30% in this simulated example). Therefore, when we report the prediction result for a patient with $X_1 = x_1 > 0$, it would be preferable to say that "since this patient is older than c (threshold in the original age scale), there is a 70% probability that his/her optimal treatment would be treatment 2, and a 30% probability that his/her optimal treatment is treatment 1" rather than only telling this patient that he/she should take treatment 2. The advantage of providing this additional information can be even more pronounced when the probability estimates for

two classes are indeed very close to each other, for example, if the probability of treatments 1 and 2 being optimal for a given patient is 45% and 55%, respectively. Based on the Bayes classification rule, we will recommend treatment 2 to this patient, but if treatment 2 is much more costly or toxic than treatment 1, the patient may go for treatment 1 in the end given that the probability of treatment 2 being better than treatment 1 is not very different from 50%.

In general, it can be challenging to obtain the probability in the context of ITR estimation, but some advancements in the field of machine learning may enable us to address the problem. For example, for ITR estimation methods that rely on SVM to predict the optimal treatment (e.g. augmented outcome weighted learning used in Chapter 3), even though SVM does not produce class probabilities directly, we can borrow ideas from Wang et al. (2008a) and extend their method (by "re-weighting the weights") to estimate the class prediction probabilities nonparametrically through sequential classifications. In the case with two treatments, if we follow the notations introduced in Section 3.2.3 (except that we remove the superscript S_k to simplify the notations), the probability of treatment (class) 1 being the optimal based on the augmented outcome weighted learning (Liu et al., 2018) can be estimated by solving

$$\min_f \frac{1}{n}\Bigg[(1-q) \times \sum_{i:A_i\text{sign}(e_i)=1} \frac{\{1-A_i\text{sign}(e_i)f(\mathbf{X}_i)\}^+|e_i|}{\pi_i} + q \times \sum_{i:A_i\text{sign}(e_i)=-1} \frac{\{1-A_i\text{sign}(e_i)f(\mathbf{X}_i)\}^+|e_i|}{\pi_i}\Bigg] + \lambda\|f\|^2 \quad (5.1)$$

repeatedly for a series of q values $0 < q_1 < \ldots < q_M < 1$ ($M > 0$ determines the estimation precision). A combination of the optimal decision functions estimated from (5.1), $\widehat{f}^*_{q_1}, \ldots, \widehat{f}^*_{q_M}$, based on the monotonicity property will lead to a consistent probability estimate (Wang et al., 2008a). Similarly, for multi-arm trials, we can adapt the multi-class probability estimation method proposed by Wang et al. (2019) to the ITR estimation context to produce an estimate for the probability of different treatments being the optimal for a given patient.

Alternative ways of quantifying the uncertainty in the ITR estimation context include constructing prediction intervals for the outcomes under different treatment assignments (Laber and Qian, 2017) or calculating the ratio (or pairwise ratio when more than two treatment options are available) of predicted patient outcomes under different treatments. Such information can help clinicians and

patients better understand to what extent treatment options differ, and hence it is worthwhile to look into this in the future.

5.4.1.2 Construction of the utility score and other methods for the benefit-risk trade-off

When defining the utility function, we assume that the trade-off parameter b is the same for all donors. In a more practical sense, it would be desirable to have the trade-off parameter b incorporate the perceived importance of two outcomes to donors and the views of health professionals. Outcome preferences can vary across each individual donor or donor subgroups (defined in terms of the blood group, ethnicity, age, donor reliability, donor's geographical location, etc.) in practice. For example, there is a growing demand for universal blood groups and minor blood groups (Grieve et al., 2018), and NHSBT may be particularly reluctant to lose such donors. Therefore, donation strategies which take these factors into account and maximise a utility score where b is tailored to donor subgroups (or even tailored on an individual case basis) may better address the need of the blood service.

In addition, we can easily incorporate other aspects that might be important to the decision-making process (e.g. cost) into the utility score. Extra costs incurred by potential loss of donors due to low Hb deferrals (which can be caused by more frequent donations) are indirectly captured by the trade-off parameter b in the utility function that balances the units of blood collected and the deferrals for low Hb. There can be other types of costs associated with shortening the inter-donation interval. For example, we would expect that more frequent donations may induce higher costs of data collection and processing. A more "cost-effective" personalised donation strategy can be estimated by maximising a properly-defined cost-utility function.

In this thesis, we consider the benefit outcome in conjunction with the risk outcome and deal with the trade-off between two outcomes by constructing a utility function and eliciting the trade-off parameter based on clinician's domain knowledge. Kosorok and Laber (2019) summarised recent work on addressing multiple outcomes in personalised medicine. For example, an alternative approach to balancing competing outcomes is through the constrained optimisation framework, i.e. consider maximising the benefit under the constraint of controlling the average risk under a prespecified and clinically meaningful threshold (Wang et al., 2018). Similar to the utility-based approach, the choice of the

threshold value is important for making the right decision and thus the cut-off needs to be appropriately calibrated.

5.4.1.3 Other types of penalties for variable selection

We adopt l_1 regularisation (i.e. LASSO-type penalty: $\lambda \sum_{j=1}^{p} |\beta_j|$ and variations of LASSO, such as group LASSO and hierarchical group LASSO) for variable selection in this thesis (Chapters 2 and 3). The l_1 penalty can be replaced by other types of penalties. For example, an alternative option is the elastic net penalty (i.e. a convex combination of l_1 and l_2 penalties: $\lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2$), which enjoys some nice properties (e.g. highly-correlated covariates are kept or removed together and the number of selected covariates is not upper-bounded by the sample size), and is especially appropriate for the situation where covariates are highly-correlated (Wang et al., 2008b; Zou and Hastie, 2005). Other options for further investigation include the l_0 penalty (i.e. $\lambda \sum_{j=1}^{p} I(\beta_j \neq 0)$) and the smoothly clipped absolute deviation (SCAD) penalty (Dasgupta and Huang, 2020; Fan and Li, 2001; Song et al., 2015).

5.4.1.4 Dynamic treatment regimes/donation strategies

Dynamic treatment regime (DTR) refers to sequential decision rules that adapt over time to the changing status of each subject to maximise the expected long-term target outcome (Chakraborty and Moodie, 2013; Moodie et al., 2007; Murphy, 2003; Zhao et al., 2015). DTRs are different from single-stage ITRs in the sense that DTRs reflect how each patient is progressing at each stage and better mimic how clinicians treat patients in practice: only the first treatment is decided at the patient's initial visit, and subsequent treatments are determined at each follow-up visit based on the progression (including updated patient information and the response to previously-administered treatments).

DTRs can be estimated from observational longitudinal studies, but we should be cautious about unmeasured confounders. Alternatively, the data collected from a special type of clinical trial, which is commonly referred to as the "sequential multiple assignment randomised trial (SMART)", can facilitate the estimation of DTRs (Murphy, 2005). In SMART, each patient is randomised multiple times at different decision points, and the randomisation probability may depend on observed information up to the decision point. As opposed to single-stage randomised trials, SMART has the ability to capture delayed effects of treatment (Wallace et al., 2016). Another advantage of SMART over the

traditional randomised trial design is that patients are more likely to remain in the study since they know that their treatment will be switched if they do not respond to the previously-assigned treatment.

From the methodological perspective, we develop SR learning under a single-stage setup in Chapter 3. A natural extension is to generalise SR learning to handle multi-stage decision problems and estimate the optimal DTR that maximises the expected long-term outcome in the ordinal treatment setting.

In the blood donation context, dynamic (and personalised) donation strategies that adapt over time to donors' changing characteristics can be more efficient in maintaining the blood supply and donor health than personalised but static ones. However, in the INTERVAL trial, donors were only randomised once at the baseline donation session (and then fixed at the initial randomised group), and thus INTERVAL is not a sequential multiple assignment randomised trial. In addition, we do not have the data on updated donor characteristics (e.g. Hb levels and iron stores) at subsequent donation sessions. Therefore, based on the data currently available from the INTERVAL trial, in this thesis, we only consider the estimation of single-stage ITRs (fixed over time) using baseline measurements of donor characteristics. It would be useful to incorporate dynamic donor stratification and estimate the optimal personalised donation strategies that reflect both heterogeneity across donors and heterogeneity over time within each donor when other data sources become available.

5.4.1.5 Development of implementation tools

Personalised donation strategies proposed in this thesis can be more efficient and sustainable than the current clinical practice, but they are also more complicated to implement in practice. Therefore, the development of a practical and easy-to-use tool is essential to ensuring uptake and translating our proposed personalised approaches to blood donation into NHSBT's routine donation practice. For example, we can build an interactive web application, which takes each donor's baseline measurements as inputs, and then outputs his/her optimal inter-donation interval based on our algorithm. Of course, to best translate the up-stream work into tools that are accessible to the blood service, we need to first discuss with the NHSBT operational and IT teams to determine what types of tools can serve our purpose and require only minimal adaptations to routine NHSBT systems. In this way, we can ensure that we do not over-burden donation centre staffs with too much administration.

5.4.2 Chapter 4 - patient stratification (subgroup level)

5.4.2.1 Uncertainty propagation

As has been discussed in Chapter 4, the uncertainty associated with the clustering results produced by profile regression can be directly quantified under the Bayesian framework. However, since we use the posterior mean of the potential outcome predictions (from BART) rather than the full posterior samples as the outcome in the profile regression model, the uncertainties associated with the predicted potential outcomes (the first stage) are not carried forward to the second stage and thus not reflected in the final output. We may employ the idea of Markov melding (a generic Bayesian computational method for evidence synthesis) to allow uncertainty propagation (Goudie et al., 2019).

5.4.2.2 Scalability

Numerical experiments suggest that our proposed patient stratification method does not scale well to large datasets, mainly due to the computational inefficiency of the second stage where profile regression is implemented for a multivariate continuous outcome. For example, in a simulated dataset with 20000 subjects, the runtime of our proposed patient stratification method with 2000 MCMC iterations for profile regression is about 3 hours. In general, for real data, a much larger number of iterations would be needed to achieve convergence compared to simulated datasets (Hastie et al., 2015), and thus the application of our method to a large-scale real dataset (e.g. the data from all male/female donors in the INTERVAL trial) can pose computational challenges. MCMC sampling can be prohibitively slow, and a variational inference algorithm for Dirichlet process mixture models may be considered in order to speed up the computation (Blei and Jordan, 2006).

5.4.2.3 Extension to settings with multiple continuous outcomes

Even though our patient stratification method is primarily developed for multi-arm trials with a univariate continuous outcome, it can be extended to the setting with multiple and potentially correlated continuous outcomes. The current implementation of BART in R (the `BART` package) cannot handle multivariate outcomes directly (and correlation might be an issue if we fit BART models separately for each outcome), but we can first "decorrelate" multiple outcomes by deriving uncorrelated principal components (PC) from the observed outcome

data and then run BART separately on each PC. For example, we consider a simple case with two outcomes Y_1 and Y_2, and let s_1^2 and s_2^2 be the sample variance of Y_1 and Y_2, respectively. It follows that

$$\mathrm{PC}_1 = \frac{Y_1}{s_1} + \frac{Y_2}{s_2},$$

and

$$\mathrm{PC}_2 = \frac{Y_1}{s_1} - \frac{Y_2}{s_2}.$$

We can verify that $\mathrm{Cov}(\mathrm{PC}_1, \mathrm{PC}_2) = \mathrm{Var}(Y_1/s_1) - \mathrm{Var}(Y_2/s_2) = 0$, and thus PC_1 and PC_2 are uncorrelated.

In the first stage, we fit a BART model for $\mathrm{PC}_1|A, \mathbf{X}$ and a separate BART model for $\mathrm{PC}_2|A, \mathbf{X}$. Let $\widehat{\mathrm{PC}}_1(a)$ and $\widehat{\mathrm{PC}}_2(a)$ denote the posterior means of the predicted values for PC_1 and PC_2 when $A = a$, $a = 1, \ldots, K$. The predicted potential outcome values for Y_1 and Y_2 under treatment a, $a = 1, \ldots, K$, can be derived from $\widehat{\mathrm{PC}}_1(a)$ and $\widehat{\mathrm{PC}}_2(a)$ as:

$$\widehat{Y}_1^*(a) = \frac{s_1 \times (\widehat{\mathrm{PC}}_1(a) + \widehat{\mathrm{PC}}_2(a))}{2}, \quad a = 1, \ldots, K,$$

and

$$\widehat{Y}_2^*(a) = \frac{s_2 \times (\widehat{\mathrm{PC}}_1(a) - \widehat{\mathrm{PC}}_2(a))}{2}, \quad a = 1, \ldots, K.$$

In the second stage, $\mathbf{Y}^* = (\widehat{Y}_1^*(1), \widehat{Y}_1^*(2), \ldots, \widehat{Y}_1^*(K), \widehat{Y}_2^*(1), \widehat{Y}_2^*(2), \ldots, \widehat{Y}_2^*(K))$ is fed into the profile regression model as the outcome and $\mathbf{Y}^*$ is linked to $\mathbf{X}$ nonparametrically through cluster memberships.

5.4.2.4 Influence of the outcome and covariates on the likelihood

In the profile regression model, both covariates ($\mathbf{X}$) and outcome ($\mathbf{Y}^*$) inform the clustering structure. If the dimension of $\mathbf{X}$ is much larger than that of $\mathbf{Y}^*$, the contribution of $\mathbf{Y}^*$ to the likelihood is likely to be overwhelmed by that of $\mathbf{X}$ (i.e. covariates might dominate the likelihood and the relative contribution of the outcome may be undermined). Consequently, the impact of the response data on the cluster allocation will be small and the resulting clusters will be formed mainly based on the similarity in the covariate space. Bigelow and Dunson (2009) argued that this might be a desirable property for some epidemiological studies. However, in some other cases where we expect the outcome to play a more important role in the clustering, we might need to upweight the outcome

likelihood. In practice, the weight may be subjective and may depend heavily on the contexts and research aims. Wade et al. (2014) also commented on the same phenomenon, i.e. the likelihood for $\mathbf{X}$ tends to dominate the clustering structure as the dimension of $\mathbf{X}$ increases. To address this issue, they proposed to use the enriched Dirichlet process (Wade et al., 2011), which induces nested partition of the data into $\mathbf{Y}^*$- clusters and $\mathbf{X}$ - clusters within each $\mathbf{Y}^*$- cluster. We may adopt a similar approach and this warrants further investigation.

References

E. Adeli, G. Wu, B. Saghafi, L. An, F. Shi, and D. Shen. Kernel-based joint feature selection and max-margin classification for early diagnosis of Parkinson's disease. *Scientific Reports*, 7:Article 41069, 2017.

E. L. Allwein, R. E. Schapire, and Y. Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *Journal of Machine Learning Research*, 1:113–141, 2001.

S. Baccianella, A. Esuli, and F. Sebastiani. Evaluation measures for ordinal regression. In *Proceedings of the 2009 Ninth International Conference on Intelligent Systems Design and Applications*, pages 283–287, 2009.

E. Bair. Semi-supervised clustering methods. *Wiley Interdisciplinary Reviews: Computational Statistics*, 5:349–361, 2013.

S. I. Bangdiwala. A graphical test for observer agreement. In *Proceedings of the 45th International Statistical Institute Meeting*, pages 307–308, 1985.

G. Baron, E. Perrodeau, I. Boutron, and P. Ravaud. Reporting of analyses from randomized controlled trials with multiple arms: a systematic review. *BMC Medicine*, 11:Article 84, 2013.

J. Bi, K. Bennett, M. Embrechts, C. Breneman, and M. Song. Dimensionality reduction via sparse support vector machines. *Journal of Machine Learning Research*, 3:1229–1243, 2003.

J. L. Bigelow and D. B. Dunson. Bayesian semiparametric joint models for functional predictors. *Journal of the American Statistical Association*, 104: 26–36, 2009.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.

D. M. Blei and M. I. Jordan. Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, 1:121–143, 2006.

Blood and Transplant Research Unit. Understanding donor characteristics, 2019. URL http://www.donorhealth-btru.nihr.ac.uk/wp-content/uploads/2019/08/Understanding-Donor-Characteristics-Theme-3-06.08.19.pdf.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

P. S. Bradley and O. L. Mangasarian. Feature selection via concave minimization and support vector machines. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 82–90, 1998.

L. Breiman. Random Forests. *Machine Learning*, 45:5–32, 2001.

J. F. Burke, J. B. Sussman, D. M. Kent, and R. A. Hayward. Three simple rules to ensure reasonably credible subgroup analyses. *BMJ*, 351:h5651, 2015.

B. Chakraborty and E. E. Moodie. *Statistical Methods for Dynamic Treatment Regimes: Reinforcement Learning, Causal Inference, and Personalized Medicine*. Springer, 2013.

G. Chen, D. Zeng, and M. R. Kosorok. Personalized dose finding using outcome weighted learning. *Journal of the American Statistical Association*, 111:1509–1521, 2016.

J. Chen, H. Fu, X. He, M. R. Kosorok, and Y. Liu. Estimating individualized treatment rules for ordinal treatments. *Biometrics*, 74:924–933, 2018.

H. A. Chipman, E. I. George, and R. E. McCulloch. BART: Bayesian additive regression trees. *Annals of Applied Statistics*, 4:266–298, 2010.

J. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46, 1960.

S. Dasgupta and Y. Huang. Selecting biomarkers for building optimal treatment selection rules by using kernel machines. *Journal of the Royal Statistical Society: Series C*, 69:69–88, 2020.

E. Di Angelantonio, S. G. Thompson, S. Kaptoge, C. Moore, M. Walker, J. Armitage, et al. Efficiency and safety of varying the frequency of whole blood donation (INTERVAL): a randomised trial of 45000 donors. *Lancet*, 390: 2360–2371, 2017.

C. F. Dormann, J. Elith, S. Bacher, C. Buchmann, G. Carl, G. Carré, et al. Collinearity: a review of methods to deal with it and a simulation study evaluating their performance. *Ecography*, 36:27–46, 2013.

Q. Du, G. Biau, F. Petit, and R. Porcher. Wasserstein random forests and applications in heterogeneous treatment effects. *arXiv e-prints*, 2020.

B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, 1993.

S. Erikainen and S. Chan. Contested futures: envisioning "personalized," "stratified," and "precision" medicine. *New Genetics and Society*, 38:308–330, 2019.

European Medicines Agency. A call to pool EU research resources into large-scale, multi-centre, multi-arm clinical trials against COVID-19, 2020. URL https://www.ema.europa.eu/en/documents/other/call-pool-eu-research-resources-large-scale-multi-centre-multi-arm-clinical-trials-against-covid-19_en.pdf.

A. Fan, W. Lu, and R. Song. Sequential advantage selection for optimal treatment regime. *Annals of Applied Statistics*, 10:32–53, 2016.

J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96: 1348–1360, 2001.

A. R. Feinstein and D. V. Cicchetti. High agreement but low kappa: I. the problems of two paradoxes. *Journal of Clinical Epidemiology*, 43:543 – 549, 1990.

J. Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76:378–382, 1971.

J. C. Foster, J. M. Taylor, and S. J. Ruberg. Subgroup identification from randomized clinical trial data. *Statistics in Medicine*, 30:2867–2880, 2011.

J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 2001.

S. Frühwirth-Schnatter and G. Malsiner-Walli. From here to infinity: sparse finite versus Dirichlet process mixtures in model-based clustering. *Advances in Data Analysis and Classification*, 13:33–64, 2019.

G. M. Fung and O. Mangasarian. A feature selection Newton method for support vector machine classification. *Computational Optimization and Applications*, 28:185–202, 2004.

M. Gail and R. Simon. Testing for qualitative interactions between treatment effects and patient subsets. *Biometrics*, 41:361–372, 1985.

G. S. Ginsburg and K. A. Phillips. Precision medicine: from science to value. *Health Affairs*, 37:694–701, 2018.

M. Goldman, K. Magnussen, J. Gorlin, M. Lozano, J. Speedy, A. Keller, et al. International forum regarding practices related to donor haemoglobin and iron. *Vox Sanguinis*, 111:449–455, 2016.

R. J. B. Goudie, A. M. Presanis, D. Lunn, D. De Angelis, and L. Wernisch. Joining and splitting models with Markov melding. *Bayesian Analysis*, 14: 81–109, 2019.

D. P. Green and H. L. Kern. Modeling heterogeneous treatment effects in survey experiments with Bayesian additive regression trees. *Public Opinion Quarterly*, 76:491–511, 2012.

J. Green and K. Davies. Understanding the implications of the INTERVAL and COMPARE studies, 2018. URL https://nhsbtdbe.blob.core.windows.net/umbraco-assets-corp/7847/interval-and-compare-implications-for-haemoglobin-screening-board-jan-18.pdf.

R. Grieve, S. Willis, K. De Corte, M. Zia Sadique, N. Hawkins, S. Perra, et al. *Options for possible changes to the blood donation service: health economics modelling*. NIHR Journals Library, 2018.

A. Guha, N. Ho, and X. Nguyen. On posterior contraction of parameters and interpretability in Bayesian mixture modeling. *arXiv e-prints*, 2019.

L. Gunter, J. Zhu, and S. Murphy. Variable selection for qualitative interactions. *Statistical Methodology*, 8:42 – 55, 2011.

I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

K. L. Gwet. Computing inter-rater reliability and its variance in the presence of high agreement. *British Journal of Mathematical and Statistical Psychology*, 61:29–48, 2008.

D. I. Hastie, S. Liverani, and S. Richardson. Sampling from Dirichlet process mixture models with unknown concentration parameter: mixing issues in large data implementations. *Statistics and Computing*, 25:1023–1037, 2015.

S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 1998.

C. Hennig and T. F. Liao. How to find an appropriate clustering for mixed-type variables with application to socio-economic stratification. *Journal of the Royal Statistical Society: Series C*, 62:309–369, 2013.

J. Hill, A. Linero, and J. Murray. Bayesian additive regression trees: a review and look forward. *Annual Review of Statistics and Its Application*, 7:251–278, 2020.

J. L. Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20:217–240, 2011.

T. Hillgrove, V. Moore, K. Doherty, and P. Ryan. The impact of temporary deferral due to low hemoglobin: future return, time to return, and frequency of subsequent donation. *Transfusion*, 51:539–547, 2011.

C. W. Hsu, C. C. Chang, and C. J. Lin. A practical guide to support vector classification. Technical reports, Department of Computer Science, National Taiwan University, 2016.

L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2: 193–218, 1985.

G. W. Imbens and D. B. Rubin. *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*. Cambridge University Press, 2015.

ISRCTN Registry. A randomised trial of treatments to prevent death in patients hospitalised with COVID-19 (coronavirus), 2020. URL http://www.isrctn.com/ISRCTN50189673.

A. Jasra, C. C. Holmes, and D. A. Stephens. Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. *Statistical Science*, 20:50–67, 2005.

A. Jović, K. Brkić, and N. Bogunović. A review of feature selection methods with applications. In *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics*, pages 1200–1205, 2015.

J. K. Karp and K. E. King. International variation in volunteer whole blood donor eligibility criteria. *Transfusion*, 50:507–513, 2010.

L. Kaufman and P. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, 1990.

V. Kecman and I. Hadzic. Support vectors selection by linear programming. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks*, pages 193–198, 2000.

R. Kennedy. An introductory guide to biomarkers, precision, personalised and stratified medicine., 2020. URL https://www.almacgroup.com/wp-content/uploads/2017/12/21358-ALMAC-Diagnostics-RK-Blog.pdf.

R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97:273–324, 1997.

M. R. Kosorok and E. B. Laber. Precision medicine. *Annual Review of Statistics and Its Application*, 6:263–286, 2019.

J. Kruppa, Y. Liu, G. Biau, M. Kohler, I. R. König, J. D. Malley, et al. Probability estimation with machine learning methods for dichotomous and multicategory outcome: Theory. *Biometrical Journal*, 56:534–563, 2014a.

J. Kruppa, Y. Liu, H. C. Diener, T. Holste, C. Weimar, I. R. König, et al. Probability estimation with machine learning methods for dichotomous and multicategory outcome: Applications. *Biometrical Journal*, 56:564–583, 2014b.

S. R. Künzel, J. S. Sekhon, P. J. Bickel, and B. Yu. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the National Academy of Sciences*, 116:4156–4165, 2019.

E. Laber and M. Qian. Evaluating personalized treatment regimes. In *Methods in Comparative Effectiveness Research*, chapter 15, pages 503–518. Taylor & Francis, 2017.

Y. Lee, Y. Lin, and G. Wahba. Multicategory support vector machines. *Journal of the American Statistical Association*, 99:67–81, 2004.

Y. Li and J. S. Liu. Robust variable and interaction selection for logistic regression and general index models. *Journal of the American Statistical Association*, 114:271–286, 2019.

M. Lim and T. Hastie. Learning interactions via hierarchical group-lasso regularization. *Journal of Computational and Graphical Statistics*, 24:627–654, 2015.

A. R. Linero. Bayesian regression trees for high-dimensional prediction and variable selection. *Journal of the American Statistical Association*, 113:626–636, 2018.

I. Lipkovich, A. Dmitrienko, J. Denne, and G. Enas. Subgroup identification based on differential effect search - a recursive partitioning method for establishing response to treatment in patient subpopulations. *Statistics in Medicine*, 30: 2601–2621, 2011.

I. Lipkovich, A. Dmitrienko, and R. B.D'Agostino Sr. Tutorial in biostatistics: data-driven subgroup identification and analysis in clinical trials. *Statistics in Medicine*, 36:136–196, 2017.

I. Lipkovich, A. Dmitrienko, C. Muysers, and B. Ratitch. Multiplicity issues in exploratory subgroup analysis. *Journal of Biopharmaceutical Statistics*, 28: 63–81, 2018.

W. Liu, Z. Zhang, L. Nie, and G. Soon. A case study in personalized medicine: Rilpivirine versus Efavirenz for treatment-naive HIV patients. *Journal of the American Statistical Association*, 112:1381–1392, 2017.

Y. Liu, Y. Wang, M. R. Kosorok, Y. Zhao, and D. Zeng. Augmented outcome-weighted learning for estimating optimal dynamic treatment regimens. *Statistics in Medicine*, 37:3776–3788, 2018.

S. Liverani, D. Hastie, L. Azizi, M. Papathomas, and S. Richardson. PReMiuM: An R package for profile regression mixture models using Dirichlet processes. *Journal of Statistical Software*, 64:1–30, 2015.

B. R. Logan, R. Sparapani, R. E. McCulloch, and P. W. Laud. Decision making and uncertainty quantification for individualized treatments using Bayesian additive regression trees. *Statistical Methods in Medical Research*, 28:1079–1093, 2019.

M. J. Lopez and R. Gutman. Estimation of causal effects with multiple treatments: A review and new ideas. *Statistical Science*, 32:432–454, 2017.

J. Lu, M. Li, and D. Dunson. Reducing over-clustering via the powered Chinese restaurant process. *arXiv e-prints*, 2018.

J. D. Malley, J. Kruppa, A. Dasgupta, K. G. Malley, and A. Ziegler. Probability machines. *Methods of Information in Medicine*, 51:74–81, 2012.

O. L. Mangasarian. Exact 1-norm support vector machines via unconstrained convex differentiable minimization. *Journal of Machine Learning Research*, 7: 1517–1530, 2006.

O. L. Mangasarian and E. W. Wild. Feature selection for nonlinear kernel support vector machines. In *Seventh IEEE International Conference on Data Mining Workshops*, pages 231–236, 2007.

D. Mathew, J. R. Giles, A. E. Baxter, A. R. Greenplate, J. E. Wu, C. Alanio, et al. Deep immune profiling of COVID-19 patients reveals patient heterogeneity and distinct immunotypes with implications for therapeutic interventions. *bioRxiv*, 2020.

J. W. Miller and M. T. Harrison. A simple example of Dirichlet process mixture inconsistency for the number of components. In *Advances in Neural Information Processing Systems*, pages 199–206, 2013.

D. Moher, S. Hopewell, K. F. Schulz, V. Montori, P. C. Gøtzsche, P. J. Devereaux, et al. CONSORT 2010 explanation and elaboration: updated guidelines for reporting parallel group randomised trials. *BMJ*, 340:c869, 2010.

J. Molitor, M. Papathomas, M. Jerrett, and S. Richardson. Bayesian profile regression with an application to the National survey of children's health. *Biostatistics*, 11:484–498, 2010.

E. E. M. Moodie, T. S. Richardson, and D. A. Stephens. Demystifying optimal dynamic treatment regimes. *Biometrics*, 63:447–455, 2007.

C. Moore, J. Sambrook, M. Walker, Z. Tolkien, S. Kaptoge, D. Allen, et al. The INTERVAL trial to determine whether intervals between blood donations can be safely and acceptably decreased to optimise blood supply: study protocol for a randomised controlled trial. *Trials*, 15:Article 363, 2014.

C. Moore, T. Bolton, M. Walker, S. Kaptoge, D. Allen, M. Daynes, et al. Recruitment and representativeness of blood donors in the INTERVAL randomised trial assessing varying inter-donation intervals. *Trials*, 17:Article 458, 2016.

S. R. Munoz and S. I. Bangdiwala. Interpretation of kappa and B statistics measures of agreement. *Journal of Applied Statistics*, 24:105–112, 1997.

S. A. Murphy. Optimal dynamic treatment regimes. *Journal of the Royal Statistical Society: Series B*, 65:331–355, 2003.

S. A. Murphy. An experimental design for the development of adaptive treatment strategies. *Statistics in Medicine*, 24:1455–1481, 2005.

M. H. Nguyen and F. de la Torre. Optimal feature selection for support vector machines. *Pattern Recognition*, 43:584 – 591, 2010.

NHS Blood and Transplant. O negative blood type, 2018. URL https://www.blood.co.uk/why-give-blood/blood-types/o-negative-blood-type/.

Nuffield Department of Population Health. RECOVERY: randomised evaluation of COVID-19 therapy, 2020. URL https://www.recoverytrial.net/.

A. Onogi, M. Nurimoto, and M. Morita. Characterization of a Bayesian genetic clustering algorithm based on a Dirichlet process prior and comparison among Bayesian clustering methods. *BMC Bioinformatics*, 12:Article 263, 2011.

M. Papathomas, J. Molitor, C. Hoggart, D. Hastie, and S. Richardson. Exploring data from genetic association studies using Bayesian variable selection and the Dirichlet process: Application to searching for gene $\times$ gene patterns. *Genetic Epidemiology*, 36:663–674, 2012.

V. Parisi and D. Leosco. Precision medicine in COVID-19: IL-1β a potential target. *JACC: Basic to Translational Science*, 5:543–544, 2020.

E. U. Patel, J. L. White, E. M. Bloch, M. K. Grabowski, E. A. Gehrie, P. M. Lokhandwala, et al. Association of blood donation with iron deficiency among adolescent and adult females in the United States: a nationally representative study. *Transfusion*, 59:1723–1733, 2019.

K. Patel, N. Clinton, H. Mukhi, and M. Patel. COVID-19 presents an opportunity for precision medicine to play expanded role in care, 2020. URL https://www.targetedonc.com/view/covid-19-presents-an-opportunity-for-precision-medicine-to-play-expanded-role-in-care.

Z. Qi and Y. Liu. D-learning to estimate optimal individual treatment rules. *Electronic Journal of Statistics*, 12:3601–3638, 2018.

M. Qian and S. A. Murphy. Performance guarantees for individualized treatment rules. *Annals of Statistics*, 39:1180–1210, 2011.

X. Qiu, D. Zeng, and Y. Wang. Estimation and evaluation of linear individualized treatment rules to guarantee performance. *Biometrics*, 74:517–528, 2018.

A. Rosenberg and J. Hirschberg. V-measure: a conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 410—420, 2007.

A. Rouanet, S. R. Richardson, and B. D. Tom. Benefit of Bayesian clustering of longitudinal data: study of cognitive decline for precision medicine. In *Bayesian methods in pharmaceutical research*, chapter 11, pages 223–242. CRC Press, 2020.

A. Rouanet, R. Johnson, M. Strauss, S. Richardson, B. D. Tom, S. R. White, and P. D. W. Kirk. Bayesian profile regression for clustering analysis involving a longitudinal response and explanatory variables, 2020+. Under review.

P. J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20: 53–65, 1987.

Y. Saeys, I. Inza, and P. Larrañaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23:2507–2517, 2007.

R. Sambandam. Cluster analysis gets complicated. *Marketing Research*, 15: 16–21, 2003.

J. Sethuraman. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4: 639–650, 1994.

V. Shankar and S. I. Bangdiwala. Behavior of agreement measures in the presence of zero cells and biased marginal distributions. *Journal of Applied Statistics*, 35:445–464, 2008.

V. Shankar and S. I. Bangdiwala. Observer agreement paradoxes in 2x2 tables: comparison of agreement measures. *BMC Medical Research Methodology*, 14: Article 100, 2014.

G. S. Shrestha, H. R. Paneru, and J. L. Vincent. Precision medicine for COVID-19: a call for better clinical trials. *Critical Care*, 24:1–3, 2020.

A. Sies, K. Demyttenaere, and I. V. Mechelen. Studying treatment-effect heterogeneity in precision medicine through induced subgroups. *Journal of Biopharmaceutical Statistics*, 29:491–507, 2019.

A. J. Smola, B. Schölkopf, and K.-R. Müller. The connection between regularization operators and support vector kernels. *Neural Networks*, 11:637 – 649, 1998.

R. Song, M. Kosorok, D. Zeng, Y. Zhao, E. Laber, and M. Yuan. On sparse representation for optimal individualized treatment selection with penalized outcome weighted learning. *Stat*, 4:59–68, 2015.

R. Sparapani, C. Spanbauer, and R. McCulloch. Nonparametric machine learning and efficient computation with Bayesian additive regression trees: the BART R package. *Journal of Statistical Software*, 2019. In press.

N. Stallard, L. Hampson, N. Benda, W. Brannath, T. Burnett, T. Friede, et al. Efficient adaptive designs for clinical trials of interventions for COVID-19. *Statistics in Biopharmaceutical Research*, 2020. In press.

X. Su, C. L. Tsai, H. Wang, D. M. Nickerson, and B. Li. Subgroup analysis via recursive partitioning. *Journal of Machine Learning Research*, 10:141–158, 2009.

Y. V. Tan and J. Roy. Bayesian additive regression trees and the general BART model. *Statistics in Medicine*, 38:5048–5069, 2019.

S. Tansey, B. Cottam, S. Dollow, A. Lockett, I. Mills, and I. Vranic. The ethics of conducting clinical trials in the search for treatments and vaccines against COVID-19, 2020. URL https://www.fpm.org.uk/blog/the-ethics-of-conducting-clinical-trials-in-the-search-for-treatments-and-vaccines-against-covid-19/.

Y. Tao and L. Wang. Adaptive contrast weighted learning for multi-stage multi-treatment decision-making. *Biometrics*, 73:145–155, 2017.

Y. W. Teh. Dirichlet process. In *Encyclopedia of Machine Learning*, pages 280–287. Springer, 2010.

L. Tian, A. A. Alizadeh, A. J. Gentles, and R. Tibshirani. A simple method for estimating interactions between a treatment and a large number of covariates. *Journal of the American Statistical Association*, 109:1517–1532, 2014.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B*, 58:267–288, 1996.

R. Tibshirani, G. Walther, and T. Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B*, 63:411–423, 2001.

N. Ting, J. C. Cappelleri, S. Ho, and D. D. G. Chen. *Design and Analysis of Subgroups with Biopharmaceutical Applications*. Springer, 2020.

J. W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, 1977.

M. J. van der Laan, E. C. Polley, and A. E. Hubbard. Super Learner. *Statistical Applications in Genetics and Molecular Biology*, 6:Article 25, 2007.

T. Vuk, K. Magnussen, W. de Kort, G. Folléa, G. M. Liumbruno, H. Schennach, et al. International forum: an investigation of iron status in blood donors. *Blood Transfusion*, 15:20–41, 2017.

S. Wade, S. Mongelluzzo, and S. Petrone. An enriched conjugate prior for Bayesian nonparametric inference. *Bayesian Analysis*, 6:359–385, 2011.

S. Wade, D. B. Dunson, S. Petrone, and L. Trippa. Improving prediction from Dirichlet process mixtures via enrichment. *Journal of Machine Learning Research*, 15:1041–1071, 2014.

M. P. Wallace, E. E. M. Moodie, and D. A. Stephens. SMART thinking: a review of recent developments in sequential multiple assignment randomized trials. *Current Epidemiology Reports*, 3:225–232, 2016.

J. Wang. Consistent selection of the number of clusters via cross-validation. *Biometrika*, 97:893–904, 2010.

J. Wang, X. Shen, and Y. Liu. Probability estimation for large-margin classifiers. *Biometrika*, 95:149–167, 2008a.

L. Wang, J. Zhu, and H. Zou. Hybrid huberized support vector machines for microarray classification and gene selection. *Bioinformatics*, 24:412–419, 2008b.

R. Wang, S. W. Lagakos, J. H. Ware, D. J. Hunter, and J. M. Drazen. Statistics in medicine – reporting of subgroup analyses in clinical trials. *New England Journal of Medicine*, 357:2189–2194, 2007.

X. Wang, H. H. Zhang, and Y. Wu. Multiclass probability estimation with support vector machines. *Journal of Computational and Graphical Statistics*, 28:586–595, 2019.

Y. Wang, H. Fu, and D. Zeng. Learning optimal personalized treatment rules in consideration of benefit and risk: with an application to treating type 2 diabetes patients with insulin therapies. *Journal of the American Statistical Association*, 113:1–13, 2018.

C. J. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8:279–292, 1992.

J. A. Watson and C. C. Holmes. Machine learning analysis plans for randomised controlled trials: detecting treatment effect heterogeneity with strict control of type I error. *Trials*, 21:Article 156, 2020.

S. Willis, K. De Corte, J. A. Cairns, M. Zia Sadique, N. Hawkins, M. Pennington, et al. Cost-effectiveness of alternative changes to a national blood collection service. *Transfusion Medicine*, 29:42–51, 2019.

World Health Organization. "Solidarity" clinical trial for COVID-19 treatments, 2020. URL https://www.who.int/emergencies/diseases/novel-coronavirus-2019/global-research-on-novel-coronavirus-2019-ncov/solidarity-clinical-trial-for-covid-19-treatments.

Y. Xu, M. Yu, Y. Zhao, Q. Li, S. Wang, and J. Shao. Regularized outcome weighted subgroup identification for differential treatment effects. *Biometrics*, 71:645–653, 2015.

C. Yang, N. Ho, and M. I. Jordan. Posterior distribution for the number of clusters in Dirichlet process mixture models. *arXiv e-prints*, 2019.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B*, 68:49–67, 2006.

B. Zhang and M. Zhang. Variable selection for estimating the optimal treatment regimes in the presence of a large number of covariates. *Annals of Applied Statistics*, 12:2335–2358, 2018.

B. Zhang, A. A. Tsiatis, M. Davidian, M. Zhang, and E. Laber. Estimating optimal treatment regimes from a classification perspective. *Stat*, 1:103–114, 2012a.

B. Zhang, A. A. Tsiatis, E. B. Laber, and M. Davidian. A robust method for estimating optimal treatment regimes. *Biometrics*, 68:1010–1018, 2012b.

Y. Zhao, D. Zeng, A. J. Rush, and M. R. Kosorok. Estimating individualized treatment rules using outcome weighted learning. *Journal of the American Statistical Association*, 107:1106–1118, 2012.

Y. Zhao, D. Zeng, E. B. Laber, and M. R. Kosorok. New statistical learning methods for estimating optimal dynamic treatment regimes. *Journal of the American Statistical Association*, 110:583–598, 2015.

Y. Zhao, E. B. Laber, Y. Ning, S. Saha, and B. E. Sands. Efficient augmentation and relaxation learning for individualized treatment rules using observational data. *Journal of Machine Learning Research*, 20:1–23, 2019a.

Y. Zhao, D. Zeng, C. M. Tangen, and M. L. Leblanc. Robustifying trial-derived optimal treatment rules for a target population. *Electronic Journal of Statistics*, 13:1717–1743, 2019b.

X. Zhou, N. Mayer-Hamblett, U. Khan, and M. R. Kosorok. Residual weighted learning for estimating individualized treatment rules. *Journal of the American Statistical Association*, 112:169–187, 2017.

X. Zhou, Y. Wang, and D. Zeng. Outcome-weighted learning for personalized medicine with multiple treatment options. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics*, pages 565–574, 2018.

J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani. 1-norm support vector machines. In *Proceedings of the 16th International Conference on Neural Information Processing Systems*, pages 49–56, 2003.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B*, 67:301–320, 2005.

Appendix A

Supplements for Chapter 2

A.1 R codes for implementing multi-arm ITR estimation methods

A.1.1 l_1-PLS-HGL

```
library(glinternet)
library(dplyr)

set.seed(284)

data <- read.csv("data.csv")
n <- dim(data)[1]
# number of covariates
p <- 5
# number of treatment options
K <- 3

# A-1 because has to be coded from 0 in glinternet package
X0 <- data%>%dplyr::select(-Y)%>%mutate(A=A-1)

# vector for number of levels needed for glinternet.cv
lv <- c(K,rep(1,p))
model.cv <- glinternet.cv(X = X0,Y = data$Y,numLevels = lv,
            interactionCandidates = 1, nFolds = 5)
```

```
X0.1 <- X0.2 <- X0.3 <- X0
X0.1[,1] <- rep(0,n)
X0.2[,1] <- rep(1,n)
X0.3[,1] <- rep(2,n)

p1 <- predict(model.cv,X0.1,type="response")
p2 <- predict(model.cv,X0.2,type="response")
p3 <- predict(model.cv,X0.3,type="response")

predict.comb <- cbind(p1,p2,p3)

# estimated optimal ITR by l1-pls-hgl
estopt.pls.hgl <- apply(predict.comb,1,which.max)
```

A.1.2 l_1-PLS-GL

```
library(oem)
library(dplyr)

set.seed(284)

data <- read.csv("data.csv")
n <- dim(data)[1]
X0 <- data%>%dplyr::select(-Y)
X0[,1] <- factor(X0[,1])
covnames <- paste(colnames(X0)[-1],collapse = "+")
fmla <- as.formula(paste("~",covnames,"+A+(",covnames,")*A"))
base <- model.matrix(fmla,data=X0)[,-1]

# number of covariates
p <- 5
# number of treatment options
K <- 3
# group vector needed in cv.oem
grp <- c(1:p,rep(p+1,K-1),rep((p+2):(2*p+1),each=K-1))
model.cv <- cv.oem(base,data$Y,penalty = "grp.lasso",
                   groups=grp, nfolds=5)
```

```
X0.1 <- X0.2 <- X0.3 <- X0
X0.1[,1] <- factor(rep(1,n),levels=c(1,2,3))
X0.2[,1] <- factor(rep(2,n),levels=c(1,2,3))
X0.3[,1] <- factor(rep(3,n),levels=c(1,2,3))

base.1 <- model.matrix(fmla,data=X0.1)[,-1]
base.2 <- model.matrix(fmla,data=X0.2)[,-1]
base.3 <- model.matrix(fmla,data=X0.3)[,-1]

p1 <- predict(model.cv, newx=base.1, which.model = "best.model")
p2 <- predict(model.cv, newx=base.2, which.model = "best.model")
p3 <- predict(model.cv, newx=base.3, which.model = "best.model")

predict.comb <- cbind(p1,p2,p3)

# estimated optimal ITR by l1-pls-gl
estopt.pls.gl <- apply(predict.comb,1,which.max)
```

A.1.3 ACWL

The following R codes are adapted from the codes provided in the supplement of Tao and Wang (2017).

```
# source functions in the supplement of Tao and Wang (2017)
# functions: M.propen, Reg.mu, and CL.AIPW
source("functions_for_simulations.R")

library(rpart)
library(MASS)
library(dplyr)

set.seed(284)

data <- read.csv("data.csv")
n <- dim(data)[1]

X0 <- data%>%dplyr::select(x1,x2,x3,x4,x5)
A <- data$A
```

```
Y <- data$Y
pis.hat <- M.propen(A=A,Xs=rep(1,n))
REG <- Reg.mu(Y=Y,As=A,H=X0)
mus.reg <- REG$mus.reg

# AIPWE adaptive contrasts and working orders
CLs.a <- CL.AIPW(Y,A,pis.hat,mus.reg)

# AIPWE contrasts
C.a1 <- CLs.a$C.a1
C.a2 <- CLs.a$C.a2

# AIPWE working order
l.a <- CLs.a$l.a

dat_ACWL <- data.frame(cbind(l.a,data))

# rpart fit model
fit.a1 <- rpart(l.a ~ x1+x2+x3+x4+x5, data=dat_ACWL, weights=C.a1,
                method="class")
fit.a2 <- rpart(l.a ~ x1+x2+x3+x4+x5, data=dat_ACWL, weights=C.a2,
                method="class")

# estimated optimal ITR
estopt.acwl1 <- as.numeric(predict(fit.a1,dat_ACWL,type="class"))
estopt.acwl2 <- as.numeric(predict(fit.a2,dat_ACWL,type="class"))
```

A.1.4 D-learning

```
library(glmnet)
library(dplyr)

set.seed(284)

data <- read.csv("data.csv")
n <- dim(data)[1]
```

```
# number of covariates
p <- 5
# total number of treatment options
K <- 3
X <- as.matrix(data%>%dplyr::select(-A,-Y))
A <- data$A
Y <- data$Y

# train 3 models
A.1.idx<-which(A==1)
A.2.idx<-which(A==2)
A.3.idx<-which(A==3)

A.32 <- A[-A.1.idx]
A.32 <- ifelse (A.32==3,1,-1)
Y.new.32 <- 2*Y[-A.1.idx]*A.32
model.32 <- cv.glmnet(X[-A.1.idx,], Y.new.32, nfolds = 5)

A.31 <- A[-A.2.idx]
A.31 <- ifelse (A.31==3,1,-1)
Y.new.31 <- 2*Y[-A.2.idx]*A.31
model.31 <- cv.glmnet(X[-A.2.idx,], Y.new.31, nfolds = 5)

A.21 <- A[-A.3.idx]
A.21 <- ifelse(A.21==2,1,-1)
Y.new.21 <- 2*Y[-A.3.idx]*A.21
model.21 <- cv.glmnet(X[-A.3.idx,], Y.new.21, nfolds = 5)

pred.32 <- predict(model.32, newx=X)
pred.31 <- predict(model.31, newx=X)
pred.21 <- predict(model.21, newx=X)
f.1 <- (-pred.21-pred.31)
f.2 <- (pred.21-pred.32)
f.3 <- (pred.31+pred.32)
f.all <- cbind(f.1,f.2,f.3)

# optimal ITR by dlearn
estopt.dlearn <- apply(f.all,1, which.max)
```

A.1.5 BART

```
library(BART)
library(dplyr)

set.seed(284)

data <- read.csv("data.csv")
n <- dim(data)[1]
# number of covariates
p <- 5
# number of treatment options
K <- 3

dat <- as.matrix(data%>%dplyr::select(-Y))
X <- as.matrix(data%>%dplyr::select(-A,-Y))
A <- data$A
Y <- data$Y

# data for prediction
test <- data.matrix(rbind(cbind(rep(1,n),X),
                          cbind(rep(2,n),X),
                          cbind(rep(3,n),X)))
colnames(test) <- colnames(dat)

# fit BART
post <- wbart(dat, Y, test)

itr <- cbind(post$yhat.test.mean[(1:n)],
             post$yhat.test.mean[n+(1:n)],
             post$yhat.test.mean[2*n+(1:n)])

# estimated optimal ITR by BART
estopt.bart <- rep(NA,n)
for(j in 1:n) estopt.bart[j] <- which(itr[j, ]==max(itr[j, ]))
```

A.2 Covariate balance after data cleaning

In this section, we examine the covariate balance after data cleaning, and we present donors' baseline characteristics by randomised groups (inter-donation intervals) after data cleaning in Table A.1 for continuous covariates and Table A.2 for categorical covariates, respectively. Based on these results, we conclude that the data cleaning process does not distort the balance of baseline covariates across randomised groups.

Table A.1: *Mean (SD) of continuous baseline characteristics by randomised groups after data cleaning (rounded to 3 significant digits).*

Baseline covariates	8-week	10-week	12-week
Age (years)	45.7 (14.0)	45.7 (14.2)	45.8 (14.0)
Body mass index (kg/m^2)	26.8 (6.74)	26.7 (5.42)	26.7 (5.65)
SF-36v2 physical component score	56.8 (4.57)	56.9 (4.49)	56.8 (4.49)
SF-36v2 mental component score	54.7 (5.90)	54.6 (6.19)	54.6 (5.99)
Blood donations in the 2 years before trial enrollment	3.64 (1.85)	3.66 (1.84)	3.65 (1.83)
Haemoglobin level (g/dL)	15.0 (1.01)	15.0 (0.992)	15.0 (0.987)
White blood cell count (10^9/L)	6.15 (1.50)	6.15 (1.52)	6.15 (1.51)
Red blood cell count (10^{12}/L)	5.05 (0.383)	5.04 (0.381)	5.04 (0.381)
Mean corpuscular haemoglobin (pg)	29.7 (1.72)	29.7 (1.67)	29.7 (1.66)
Mean corpuscular volume (fL)	92.2 (4.64)	92.3 (4.64)	92.3 (4.67)
Platelet count (10^9/L)	229 (49.7)	229 (50.2)	228 (50.7)

Table A.2: *Number (percentages) of participants in each category of categorical baseline characteristics by randomised groups after data cleaning.*

Baseline covariates	8-week	10-week	12-week
Ethnicity			
White	5790 (83.8)	5751 (83.8)	5675 (83.4)
Black	48 (0.7)	50 (0.7)	59 (0.9)
Asian	161 (2.3)	163 (2.4)	180 (2.6)
Mixed	84 (1.2)	64 (0.9)	71 (1.0)
Other	32 (0.5)	16 (0.2)	27 (0.4)
Unknown	796 (11.5)	815 (11.9)	792 (11.6)
Blood group			
A-	518 (7.5)	523 (7.6)	544 (8.0)
A+	2137 (30.9)	2112 (30.8)	2122 (31.2)
AB-	46 (0.7)	59 (0.9)	48 (0.7)
AB+	208 (3.0)	171 (2.5)	187 (2.7)
B-	130 (1.9)	128 (1.9)	166 (2.4)
B+	597 (8.6)	590 (8.6)	566 (8.3)
O-	789 (11.4)	734 (10.7)	729 (10.7)
O+	2486 (36.0)	2542 (37.1)	2442 (35.9)
Iron prescription			
Yes	8 (0.1)	13 (0.2)	26 (0.4)
No	6833 (98.9)	6767 (98.7)	6692 (98.4)
Unknown	70 (1.0)	79 (1.2)	86 (1.3)
Smoke ever			
Yes	2867 (41.5)	2824 (41.2)	2870 (42.2)
No	3989 (57.7)	3980 (58.0)	3869 (56.9)
Unknown	55 (0.8)	55 (0.8)	65 (1.0)
Smoke currently			
Yes	483 (2.3)	566 (2.8)	520 (2.5)
No	2368 (11.5)	2239 (10.9)	2319 (11.3)
Unknown	4060 (19.7)	4054 (19.7)	3965 (19.3)
Alcohol ever			
Yes	6716 (97.2)	6676 (97.3)	6608 (97.1)
No	176 (2.5)	167 (2.4)	161 (2.4)
Unknown	19 (0.3)	16 (0.2)	35 (0.5)
Alcohol currently			
Yes	6060 (29.5)	6026 (29.3)	5979 (29.1)
No	421 (2.0)	421 (2.0)	380 (1.)
Unknown	430 (2.1)	412 (2.0)	445 (2.2)
New or returning donor status			
New	509 (7.4)	493 (7.2)	484 (7.1)
Returning	6402 (92.6)	6366 (92.8)	6320 (92.9)

Appendix B

Supplements for Chapter 3

B.1 Additional simulation results

B.1.1 Choice of the reference treatment

We note that in the SR learning algorithm, we assume that the least "intensive" treatment (treatment 1) is the reference group. For example, when $K = 3$, SR learning starts by comparing {1} and {2,3} in the first sequential step (S_1) to see whether a treatment that is more "intensive" than treatment 1 should be administered. As has been discussed in Section 3.2.1, in the ordinal arm setting, either the least "intensive" treatment (treatment 1) or the most "intensive" treatment (treatment K) can be used as the reference/control. In the situation where clinicians think that treatment K should be regarded as the reference treatment, and interest lies in investigating whether treatments that are less "intensive" than K should be given, SR learning can still be used by reversing the order of comparisons: in this case (if $K = 3$), we first compare {3} with {2,1} in S_1, then we compare {3} vs. {2} in the first re-estimation step, R_1, and {2} vs. {1} in the second sequential step, S_2. All the simulation results presented in Section 3.3.2.1 are run under the assumption that treatment 1 is the reference (and in S_1, {1} and {2,3} are compared). In this section, we rerun all simulations under settings 1-10 (low-dimensional case without noise covariates) by assuming that treatment K is the reference (and in S_1, {3} and {2,1} are compared). Simulation results are presented in Table B.1, which suggest that SR learning is robust to whether the least or the most "intensive" treatment is considered as the reference.

Table B.1: *Simulation results based on 500 replicates: mean (SD) of misclassification rates and value functions for settings 1-10 when the least "intensive" treatment (treatment 1) is considered as the reference and when the most "intensive" treatment (treatment 3 or 4) is considered as the reference.*

		$n = 400$		$n = 800$	
Setting	Method-Reference	Misclassification	Value	Misclassification	Value
1	SR-Linear-1	0.03 (0.01)	4.85 (0.04)	0.02 (0.01)	4.91 (0.03)
	SR-Linear-3	0.04 (0.01)	4.85 (0.05)	0.02 (0.01)	4.91 (0.03)
	SR-Gaussian-1	0.07 (0.02)	4.72 (0.08)	0.04 (0.01)	4.82 (0.05)
	SR-Gaussian-3	0.07 (0.02)	4.72 (0.08)	0.04 (0.01)	4.82 (0.05)
2	SR-Linear-1	0.04 (0.01)	4.83 (0.06)	0.02 (0.01)	4.91 (0.03)
	SR-Linear-3	0.04 (0.01)	4.86 (0.06)	0.02 (0.01)	4.92 (0.03)
	SR-Gaussian-1	0.06 (0.01)	4.75 (0.06)	0.04 (0.01)	4.84 (0.04)
	SR-Gaussian-3	0.06 (0.02)	4.75 (0.08)	0.04 (0.01)	4.83 (0.06)
3	SR-Linear-1	0.04 (0.01)	4.82 (0.06)	0.02 (0.01)	4.89 (0.02)
	SR-Linear-3	0.04 (0.02)	4.80 (0.06)	0.02 (0.01)	4.87 (0.03)
	SR-Gaussian-1	0.06 (0.02)	4.72 (0.08)	0.04 (0.01)	4.80 (0.05)
	SR-Gaussian-3	0.06 (0.01)	4.73 (0.06)	0.04 (0.01)	4.81 (0.04)
4	SR-Linear-1	0.04 (0.01)	4.85 (0.06)	0.02 (0.01)	4.92 (0.03)
	SR-Linear-3	0.04 (0.01)	4.84 (0.06)	0.02 (0.01)	4.91 (0.03)
	SR-Gaussian-1	0.07 (0.02)	4.71 (0.08)	0.05 (0.01)	4.82 (0.05)
	SR-Gaussian-3	0.07 (0.02)	4.71 (0.07)	0.05 (0.01)	4.81 (0.04)
5	SR-Linear-1	0.07 (0.02)	4.71 (0.10)	0.03 (0.01)	4.87 (0.04)
	SR-Linear-4	0.08 (0.03)	4.67 (0.11)	0.04 (0.01)	4.82 (0.05)
	SR-Gaussian-1	0.13 (0.03)	4.45 (0.15)	0.08 (0.02)	4.68 (0.08)
	SR-Gaussian-4	0.14 (0.03)	4.44 (0.14)	0.08 (0.02)	4.65 (0.08)
6	SR-Linear-1	0.49 (0.04)	4.52 (0.12)	0.49 (0.03)	4.56 (0.09)
	SR-Linear-3	0.47 (0.03)	4.49 (0.13)	0.47 (0.03)	4.52 (0.10)
	SR-Gaussian-1	0.16 (0.02)	5.89 (0.10)	0.11 (0.01)	6.11 (0.06)
	SR-Gaussian-3	0.16 (0.02)	5.92 (0.09)	0.10 (0.01)	6.13 (0.06)
7	SR-Gaussian-1	0.03 (0.01)	5.51 (0.05)	0.02 (0.01)	5.56 (0.03)
	SR-Gaussian-3	0.03 (0.01)	5.51 (0.05)	0.02 (0.01)	5.56 (0.03)
8	SR-Gaussian-1	0.05 (0.01)	5.47 (0.06)	0.03 (0.01)	5.55 (0.03)
	SR-Gaussian-3	0.05 (0.01)	5.47 (0.06)	0.03 (0.01)	5.53 (0.05)
9	SR-Gaussian-1	0.08 (0.01)	5.35 (0.05)	0.06 (0.01)	5.40 (0.03)
	SR-Gaussian-3	0.09 (0.02)	5.28 (0.08)	0.08 (0.01)	5.35 (0.05)
10	SR-Gaussian-1	0.09 (0.02)	5.25 (0.09)	0.05 (0.01)	5.40 (0.06)
	SR-Gaussian-4	0.09 (0.03)	5.25 (0.11)	0.05 (0.01)	5.41 (0.05)

B.1.2 "Permuted" orderings of true optimal ITRs

In this section, we present designs and results of additional simulation studies that evaluate the robustness of SR learning to the underlying orderings of true optimal ITRs with respect to covariates. We examine variations of settings 7 and 8 in Section 3.3.1 (we take these two settings as examples for illustration purposes, but the conclusion drawn here should be applicable to other settings as well). Specifically, decision boundaries are kept the same as those in settings 7 and 8, but orderings of true optimal ITRs with respect to covariate values are "permuted". In Figure B.1, we plot the true underlying decision boundaries and optimal ITRs in these supplementary settings.

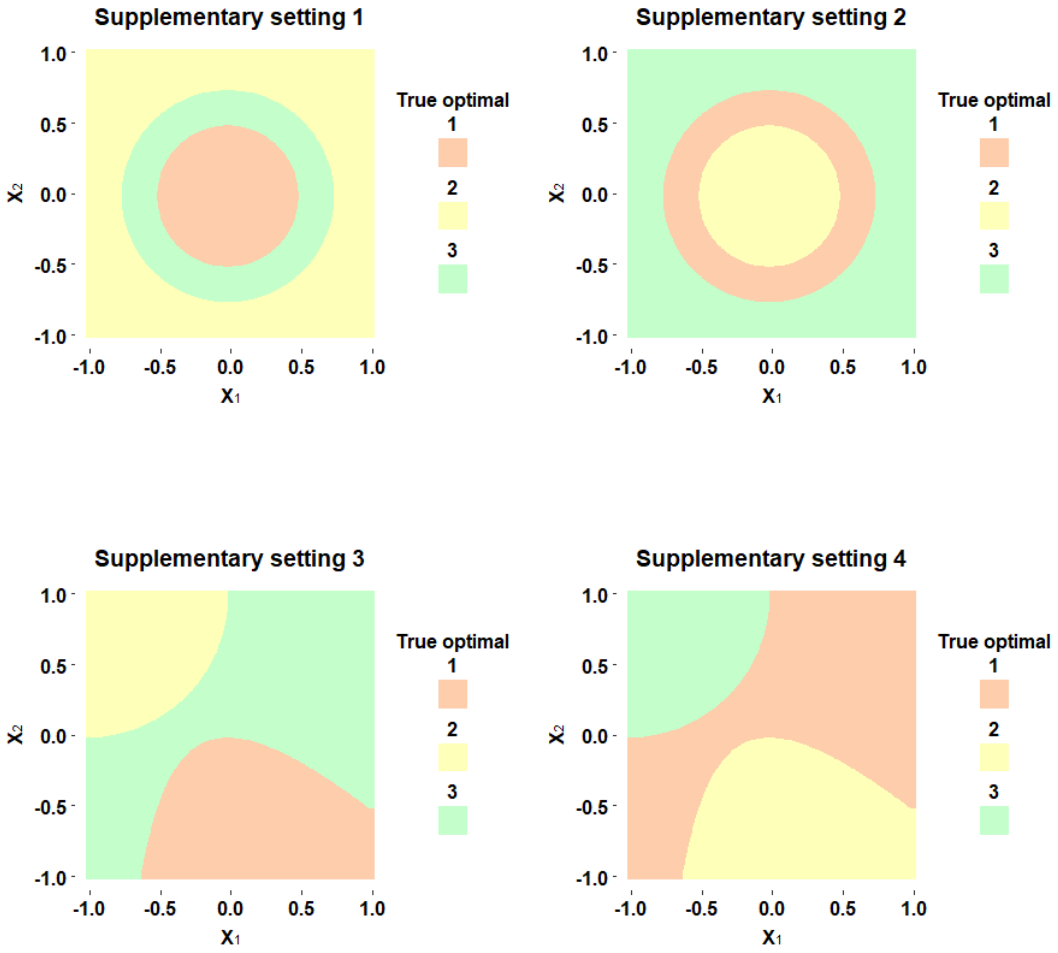


Fig. B.1: *Visualisation of true underlying decision boundaries and optimal ITRs in supplementary settings 1-4.*

Supplementary settings 1 and 2 are adapted from and share the same decision boundaries as setting 8. However, the true optimal ITRs from the innermost to the outermost are not in the order of $1 \to 2 \to 3$, but instead $1 \to 3 \to 2$ or $2 \to 1 \to 3$. Supplementary settings 3 and 4 have the same decision boundaries as setting 7, but the orders of the true optimal ITRs from the top left to the bottom right are $2 \to 3 \to 1$ or $3 \to 1 \to 2$. Same as settings 7 and 8, in supplementary settings 1-4, the main effect $\mu(\mathbf{X})$ is fixed to be $5 + X_1^2 + X_2^2$, and the loss function $\varphi\{A, \mathcal{D}^*(\mathbf{X})\}$ is set to be $4\{A - \mathcal{D}^*(\mathbf{X})\}^2$. Simulation results for supplementary settings 1-4 are shown in Table B.2. We observe that the performance of SR learning remains superior to the other competing methods when the orders of the true optimal ITRs with respect to covariates are "permutated".

Table B.2: *Simulation results based on 500 replicates: mean (SD) of misclassification rates and value functions for supplementary settings 1-4. The smallest misclassification rate and largest value function for each setting are in bold.*

		$n = 400$		$n = 800$	
Setting	Method	Misclassification	Value	Misclassification	Value
supplementary setting 1	D-Learn	0.46 (0.05)	3.80 (0.20)	0.46 (0.04)	3.85 (0.18)
	PLS	0.44 (0.01)	3.92 (0.02)	0.44 (0.00)	3.93 (0.01)
	ACWL-C1	0.25 (0.07)	4.13 (0.36)	0.16 (0.04)	4.53 (0.20)
	ACWL-C2	0.49 (0.07)	3.49 (0.30)	0.45 (0.03)	3.86 (0.15)
	SR-Gaussian	**0.07 (0.03)**	**5.07 (0.22)**	**0.05 (0.02)**	**5.30 (0.12)**
supplementary setting 2	D-Learn	0.46 (0.06)	3.77 (0.27)	0.44 (0.05)	3.90 (0.23)
	PLS	0.45 (0.05)	3.88 (0.19)	0.45 (0.04)	3.89 (0.15)
	ACWL-C1	0.25 (0.06)	3.65 (0.27)	0.21 (0.06)	3.88 (0.33)
	ACWL-C2	0.57 (0.08)	3.08 (0.27)	0.67 (0.07)	2.92 (0.26)
	SR-Gaussian	**0.12 (0.06)**	**4.93 (0.27)**	**0.11 (0.05)**	**5.14 (0.18)**
supplementary setting 3	D-Learn	0.55 (0.08)	3.21 (0.34)	0.53 (0.06)	3.36 (0.26)
	PLS	0.65 (0.06)	2.91 (0.22)	0.65 (0.03)	2.92 (0.14)
	ACWL-C1	0.21 (0.05)	4.12 (0.34)	0.15 (0.05)	4.42 (0.31)
	ACWL-C2	0.45 (0.08)	3.68 (0.31)	0.50 (0.08)	3.59 (0.30)
	SR-Gaussian	**0.06 (0.03)**	**5.14 (0.16)**	**0.04 (0.02)**	**5.26 (0.10)**
supplementary setting 4	D-Learn	0.60 (0.06)	3.18 (0.22)	0.62 (0.05)	3.17 (0.19)
	PLS	0.60 (0.04)	3.23 (0.17)	0.61 (0.03)	3.23 (0.11)
	ACWL-C1	0.20 (0.06)	4.44 (0.27)	0.13 (0.04)	4.73 (0.23)
	ACWL-C2	0.40 (0.06)	3.93 (0.26)	0.44 (0.06)	3.86 (0.24)
	SR-Gaussian	**0.04 (0.02)**	**5.39 (0.10)**	**0.03 (0.01)**	**5.47 (0.06)**

B.1.3 Intersecting decision boundaries

In this section, we consider simulation settings (supplementary settings 5 and 6) where true optimal treatment boundaries intersect and we assess the performance of SR learning in these settings. Same as before, $\mu(\mathbf{X}) = 5 + X_1^2 + X_2^2$ and $\varphi\{A, \mathcal{D}^*(\mathbf{X})\} = 4\{A - \mathcal{D}^*(\mathbf{X})\}^2$. The true optimal decision rules for these settings are plotted in Figure B.2 and listed in the following:

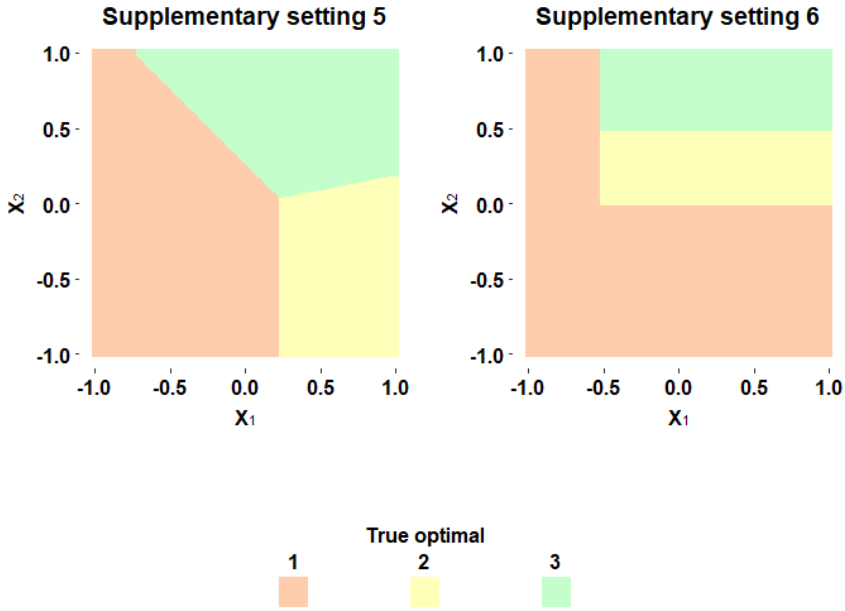


Fig. B.2: *Visualisation of true underlying decision boundaries and optimal ITRs in supplementary settings 5 and 6.*

Supplementary setting 5:

$$\mathcal{D}^*(\mathbf{X}) = \begin{cases} 1, & \text{if } I(X_1 \leq 0.25)I(X_1 + X_2 \leq 0.3) = 1, \\ 2, & \text{if } I(X_1 > 0.25)I(X_2 < 0.2X_1) = 1, \\ 3, & \text{if } I(X_1 + X_2 > 0.3)I(X_2 \geq 0.2X_1) = 1. \end{cases}$$

Supplementary setting 6:

$$\mathcal{D}^*(\mathbf{X}) = \begin{cases} 1, & \text{if } I(X_1 \geq -0.5)I(X_2 \geq 0) = 0, \\ 2, & \text{if } I(X_1 \geq -0.5)I(0 < X_2 < 0.5) = 1, \\ 3, & \text{if } I(X_1 \geq -0.5)I(X_2 \geq 0.5) = 1. \end{cases}$$

Simulation results for supplementary settings 5 and 6 are presented in Table B.3.

Table B.3: *Simulation results based on 500 replicates: mean (SD) of misclassification rates and value functions for supplementary settings 5 and 6. The smallest misclassification rate and largest value function for each setting are in bold.*

		$n = 400$		$n = 800$	
Setting	Method	Misclassification	Value	Misclassification	Value
supplementary setting 5	D-Learn	0.29 (0.09)	4.39 (0.40)	0.26 (0.06)	4.54 (0.31)
	PLS	0.28 (0.03)	4.54 (0.13)	0.27 (0.02)	4.55 (0.09)
	ACWL-C1	0.11 (0.04)	4.78 (0.23)	0.08 (0.03)	4.99 (0.17)
	ACWL-C2	0.19 (0.05)	4.80 (0.18)	0.18 (0.04)	4.86 (0.13)
	SR-Gaussian	**0.04 (0.01)**	**5.43 (0.07)**	**0.02 (0.01)**	**5.49 (0.05)**
supplementary setting 6	D-Learn	0.21 (0.10)	4.70 (0.46)	0.17 (0.09)	4.90 (0.43)
	PLS	0.20 (0.03)	4.81 (0.10)	0.19 (0.02)	4.84 (0.07)
	ACWL-C1	0.08 (0.04)	5.26 (0.19)	0.05 (0.03)	5.42 (0.14)
	ACWL-C2	0.19 (0.05)	4.89 (0.22)	0.17 (0.05)	4.96 (0.19)
	SR-Gaussian	**0.05 (0.02)**	**5.37 (0.10)**	**0.04 (0.01)**	**5.44 (0.07)**

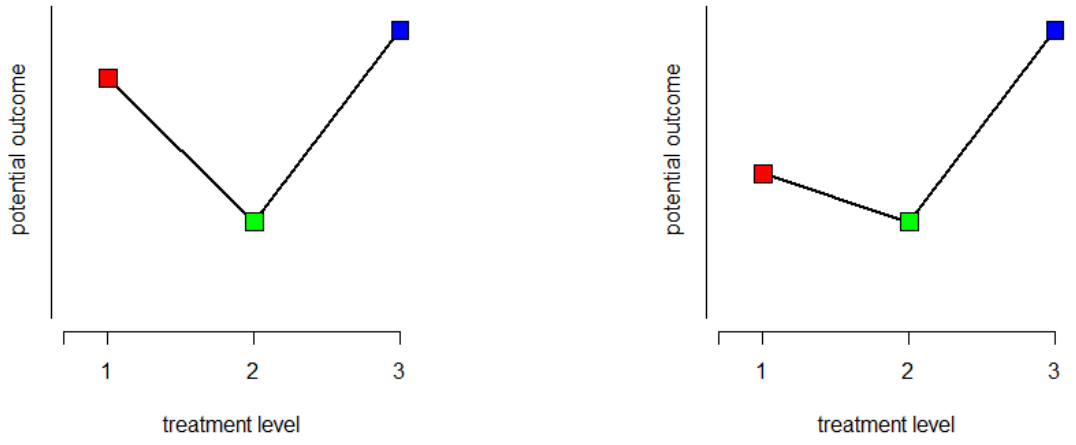
These results suggest that SR learning outperforms the other ITR estimation methods that ignore the treatment orderings in settings where decision boundaries cross.

B.1.4 Violation of the ordinality assumption

In this section, we conduct sensitivity analysis to examine how and to what extent the violation of the "one-side monotonicity (OSM)" rule (i.e. $E(Y|\mathbf{X}=\mathbf{x}, A=a_1) < E(Y|\mathbf{X}=\mathbf{x}, A=a_2)$ if a_1 and a_2 are on the same side of the true optimal treatment with a_1 being further away from the true optimal than a_2) affects the relative performance of SR learning and multi-arm ITR estimation methods that ignore the ordinality.

We design a simulation study where we randomly select $q\%$ of patients whose $\mathcal{D}^*(\mathbf{X}) = 3$ to violate OSM while the remaining $(100-q)\%$ satisfy OSM, i.e. among those whose best treatment is 3, for $(100-q)\%$ of them, treatment 2 is the second best option, which is better than treatment 1, while for the rest $q\%$ of the patients (OSM violators), the second best treatment option is 1 (for these patients, treatment 1 is better than treatment 2, even though 1 is further away from the true optimal). We examine the case with q being 0, 40 and 60.

We consider two possible scenarios which reflect different relationships between $E(Y|\mathbf{X}, A)$ and A among those whose optimal treatment $\mathcal{D}^*(\mathbf{X}) = 3$ but treatment 1 is better than treatment 2. These two scenarios are plotted in Figure B.3.



(a) *Scenario 1* **(b)** *Scenario 2*

Fig. B.3: *Scenarios for sensitivity analysis when the ordinality assumption is violated.*

Specifically, Figures B.3 (a) and (b) reflect the situation with $E(Y|\mathbf{X}, A=1) > \{E(Y|\mathbf{X}, A=2) + E(Y|\mathbf{X}, A=3)\}/2$, and $E(Y|\mathbf{X}, A=1) < \{E(Y|\mathbf{X}, A=2) + E(Y|\mathbf{X}, A=3)\}/2$, respectively. The conclusions under these two scenarios might differ since in the first sequential step of SR learning, we compare $\{1\}$ (coded as -1) vs. $\{2,3\}$ (coded as 1), and according to Zhou et al. (2018), the optimal decision rule for S_1 is given by

$$\text{sign}\left[\frac{E(Y|\mathbf{X}, A=2) + E(Y|\mathbf{X}, A=3)}{2} - E(Y|\mathbf{X}, A=1)\right]. \tag{B.1}$$

B.1.4.1 Scenario 1

For demonstration, we adapt settings 7-9 from Section 3.3.1. The decision boundaries are kept unchanged, but we modify $\varphi\{A, \mathcal{D}^*(\mathbf{X})\}$, i.e. the loss in the outcome when the assigned treatment is A and the true optimal treatment is $\mathcal{D}^*(\mathbf{X})$, as follows:

- For those whose $\mathcal{D}^*(\mathbf{X}) \neq 3$, $\varphi = 4\{A - \mathcal{D}^*(\mathbf{X})\}^2$ (as the original settings).

- For those whose $\mathcal{D}^*(\mathbf{X}) = 3$, we randomly select $q\%$ to be OSM violators, and the rest $(100-q)\%$ are considered as OSM followers:
  - For OSM followers, $\varphi = 4\{A - \mathcal{D}^*(\mathbf{X})\}^2 = 4(A-3)^2$
  - For OSM violators, $\varphi = \begin{cases} 0 & \text{if } A = 3, \\ 4A^2 & \text{if } A \neq 3. \end{cases}$
    For these patients (OSM violators), the relationship between $E(Y|\mathbf{X}, A)$ and A looks similar to Figure B.3 (a), and $E(Y|\mathbf{X}, A=1) > \{E(Y|\mathbf{X}, A=2) + E(Y|\mathbf{X}, A=3)\}/2$.

Simulation results corresponding to this scenario are presented in Table B.4. Not surprisingly, as the violation percentage gets larger, the noise becomes larger and the signal becomes weaker (i.e. less ordinal information). Therefore, SR learning performs worse (and its advantage over the competing methods that ignore the ordinality gets smaller) in that the misclassification rate gets larger and the estimate also becomes less stable. When the violation percentage is fixed, a larger sample size is required for SR learning to have a decent performance. This is also reasonable, since if we consider the total number of OSM followers as the "effective sample size" for training the SR learning algorithm, then we would need a larger overall sample size (n) in order to get enough "effective" samples.

Table B.4: *Sensitivity analysis (scenario 1) of the violation of assumptions based on 100 replicates: mean (SD) of misclassification rates and value functions for settings 7-9 with no violation of the assumption, 40% violation, and 60% violation (among those whose $\mathcal{D}^*(\mathbf{X}) = 3$). The smallest misclassification rate and largest value function for each setting are in bold.*

		$n = 400$		$n = 800$	
Setting	Method	Misclassification	Value	Misclassification	Value
0 % violation (setting 7)	D-Learn	0.14 (0.05)	5.12 (0.21)	0.11 (0.03)	5.23 (0.12)
	PLS	0.14 (0.03)	5.11 (0.10)	0.14 (0.02)	5.13 (0.07)
	ACWL-C1	0.16 (0.04)	4.92 (0.21)	0.13 (0.03)	5.11 (0.16)
	ACWL-C2	0.14 (0.04)	5.09 (0.15)	0.12 (0.03)	5.19 (0.11)
	SR-Gaussian	**0.03 (0.01)**	**5.51 (0.05)**	**0.02 (0.01)**	**5.56 (0.03)**
40 % violation (setting 7)	D-Learn	0.26 (0.06)	4.58 (0.25)	0.25 (0.05)	4.66 (0.20)
	PLS	0.28 (0.06)	4.54 (0.21)	0.28 (0.05)	4.57 (0.18)
	ACWL-C1	0.13 (0.03)	4.90 (0.24)	0.10 (0.02)	5.09 (0.16)
	ACWL-C2	0.11 (0.02)	5.10 (0.13)	0.09 (0.01)	5.23 (0.07)
	SR-Gaussian	**0.06 (0.02)**	**5.39 (0.08)**	**0.05 (0.01)**	**5.46 (0.05)**
60 % violation (setting 7)	D-Learn	0.34 (0.06)	4.30 (0.28)	0.33 (0.05)	4.37 (0.18)
	PLS	0.37 (0.05)	4.25 (0.17)	0.37 (0.03)	4.25 (0.12)
	ACWL-C1	0.11 (0.02)	4.92 (0.22)	0.09 (0.02)	5.07 (0.17)
	ACWL-C2	0.11 (0.02)	5.06 (0.13)	0.10 (0.02)	5.18 (0.08)
	SR-Gaussian	**0.09 (0.05)**	**5.21 (0.40)**	**0.07 (0.03)**	**5.37 (0.15)**
0 % violation (setting 8)	D-Learn	0.25 (0.07)	4.64 (0.30)	0.24 (0.06)	4.70 (0.24)
	PLS	0.30 (0.05)	4.44 (0.22)	0.29 (0.04)	4.48 (0.17)
	ACWL-C1	0.24 (0.04)	4.53 (0.28)	0.21 (0.03)	4.68 (0.24)
	ACWL-C2	0.52 (0.07)	3.52 (0.32)	0.61 (0.07)	3.19 (0.28)
	SR-Gaussian	**0.05 (0.01)**	**5.47 (0.06)**	**0.03 (0.01)**	**5.55 (0.03)**
40 % violation (setting 8)	D-Learn	0.23 (0.06)	4.54 (0.48)	0.21 (0.04)	4.73 (0.33)
	PLS	0.24 (0.04)	4.59 (0.43)	0.23 (0.02)	4.73 (0.16)
	ACWL-C1	0.19 (0.04)	4.43 (0.29)	0.14 (0.02)	4.74 (0.18)
	ACWL-C2	0.27 (0.04)	3.75 (0.43)	0.26 (0.04)	3.76 (0.41)
	SR-Gaussian	**0.11 (0.05)**	**5.18 (0.23)**	**0.08 (0.04)**	**5.32 (0.17)**
60 % violation (setting 8)	D-Learn	0.25 (0.03)	4.59 (0.24)	0.24 (0.02)	4.67 (0.17)
	PLS	0.25 (0.03)	**4.66 (0.24)**	0.24 (0.00)	4.71 (0.01)
	ACWL-C1	**0.18 (0.03)**	4.26 (0.32)	**0.14 (0.02)**	4.54 (0.21)
	ACWL-C2	0.22 (0.04)	4.23 (0.29)	0.17 (0.03)	4.57 (0.23)
	SR-Gaussian	0.27 (0.12)	4.04 (0.87)	0.16 (0.08)	**4.91 (0.48)**
0 % violation (setting 9)	D-Learn	0.26 (0.08)	4.64 (0.34)	0.24 (0.07)	4.73 (0.28)
	PLS	0.33 (0.06)	4.37 (0.24)	0.31 (0.04)	4.46 (0.14)
	ACWL-C1	0.22 (0.05)	4.60 (0.29)	0.18 (0.03)	4.81 (0.26)
	ACWL-C2	0.57 (0.07)	3.34 (0.29)	0.69 (0.07)	2.93 (0.28)
	SR-Gaussian	**0.08 (0.01)**	**5.35 (0.05)**	**0.06 (0.01)**	**5.40 (0.03)**
40 % violation (setting 9)	D-Learn	0.19 (0.04)	4.70 (0.37)	0.18 (0.04)	4.87 (0.30)
	PLS	0.19 (0.02)	**4.91 (0.09)**	0.20 (0.01)	4.93 (0.05)
	ACWL-C1	0.17 (0.04)	4.53 (0.34)	**0.12 (0.02)**	4.94 (0.19)
	ACWL-C2	0.25 (0.06)	3.92 (0.61)	0.20 (0.07)	4.35 (0.63)
	SR-Gaussian	**0.16 (0.04)**	4.84 (0.23)	0.15 (0.04)	**4.99 (0.16)**
60 % violation (setting 9)	D-Learn	0.22 (0.04)	4.65 (0.32)	0.20 (0.02)	4.83 (0.18)
	PLS	0.20 (0.01)	**4.86 (0.08)**	0.20 (0.01)	**4.90 (0.04)**
	ACWL-C1	**0.16 (0.04)**	4.37 (0.39)	**0.11 (0.02)**	4.82 (0.22)
	ACWL-C2	0.20 (0.04)	4.36 (0.31)	0.13 (0.03)	4.85 (0.21)
	SR-Gaussian	0.29 (0.09)	3.82 (0.70)	0.22 (0.07)	4.48 (0.50)

When the violation proportion is greater than 50%, SR learning generally has no advantage over the other methods that ignore the ordinality, especially if the sample size is small. Another interesting finding is that when the sample size increases, D-learn and PLS do not seem to improve much, whereas SR learning and ACWL improve.

B.1.4.2 Scenario 2

Same as the first scenario, we adapt from settings 7-9 by keeping the decision boundaries unchanged, and varying the loss, $\varphi\{A, \mathcal{D}^*(\mathbf{X})\}$. The modified loss is given in the following:

- For those whose $\mathcal{D}^*(\mathbf{X}) \neq 3$, $\varphi = 4\{A - \mathcal{D}^*(\mathbf{X})\}^{1/2}$ (note that here we use the square root loss instead of the squared loss).
- For those whose $\mathcal{D}^*(\mathbf{X}) = 3$, we randomly select $q\%$ to be OSM violators, and the rest $(100 - q)\%$ are considered as OSM followers:
  - For OSM followers, $\varphi = 4\{A - \mathcal{D}^*(\mathbf{X})\}^{1/2} = 4(A - 3)^{1/2}$
  - For OSM violators, $\varphi = \begin{cases} 0 & \text{if } A = 3, \\ 4A^{1/2} & \text{if } A \neq 3. \end{cases}$
    For these patients (OSM violators), the relationship between $E(Y|\,\mathbf{X}, A)$ and A looks similar to Figure B.3 (b), and $E(Y|\,\mathbf{X}, A = 1) < \{E(Y|\,\mathbf{X}, A = 2) + E(Y|\,\mathbf{X}, A = 3)\}/2$.

Simulation results corresponding to scenario 2 are shown in Table B.5. We observe that under this scenario, the impact of the violation of OSM on SR learning is much smaller compared to scenario 1 and SR learning is fairly robust (especially when $n = 800$): even when the violation proportion is as high as 60%, SR learning still performs reasonably well in general, despite that in the last setting (variation of setting 9), we can see a trend that SR learning is getting worse as the violation percentage increases.

It is clear that under scenario 1, where $E(Y|\,\mathbf{X}, A = 1) > \{E(Y|\,\mathbf{X}, A = 2) + E(Y|\,\mathbf{X}, A = 3)\}/2$ for those whose $\mathcal{D}^*(\mathbf{X}) = 3$ and $E(Y|\,\mathbf{X}, A = 1) > E(Y|\,\mathbf{X}, A = 2)$ (i.e. OSM violators), and under scenario 2, where $E(Y|\,\mathbf{X}, A = 1) < \{E(Y|\,\mathbf{X}, A = 2) + E(Y|\,\mathbf{X}, A = 3)\}/2$ for those whose $\mathcal{D}^*(\mathbf{X}) = 3$ and $E(Y|\,\mathbf{X}, A = 1) > E(Y|\,\mathbf{X}, A = 2)$, the extent to which SR learning is affected by OSM violation differs: the influence of OSM violation on SR learning is much smaller under scenario 2. A possible explanation for this

Table B.5: *Sensitivity analysis (scenario 2) of the violation of assumptions based on 100 replicates: mean (SD) of misclassification rates and value functions for settings 7-9 with no violation of the assumption, 40% violation, and 60% violation (among those whose $\mathcal{D}^*(\mathbf{X}) = 3$). The smallest misclassification rate and largest value function are in bold.*

		$n = 400$		$n = 800$	
Setting	Method	Misclassification	Value	Misclassification	Value
0 % violation (setting 7)	D-Learn	0.23 (0.05)	4.78 (0.21)	0.18 (0.04)	4.97 (0.17)
	PLS	0.18 (0.03)	4.95 (0.12)	0.18 (0.02)	4.96 (0.09)
	ACWL-C1	0.09 (0.02)	5.31 (0.07)	0.08 (0.01)	5.38 (0.06)
	ACWL-C2	0.10 (0.02)	5.30 (0.07)	0.08 (0.02)	5.35 (0.06)
	SR-Gaussian	**0.03 (0.01)**	**5.52 (0.04)**	**0.02 (0.01)**	**5.56 (0.02)**
40 % violation (setting 7)	D-Learn	0.24 (0.05)	4.67 (0.23)	0.20 (0.04)	4.85 (0.17)
	PLS	0.18 (0.03)	4.89 (0.13)	0.17 (0.02)	4.93 (0.09)
	ACWL-C1	0.09 (0.01)	5.30 (0.06)	0.08 (0.01)	5.36 (0.06)
	ACWL-C2	0.10 (0.02)	5.28 (0.09)	0.08 (0.01)	5.35 (0.06)
	SR-Gaussian	**0.03 (0.01)**	**5.54 (0.05)**	**0.02 (0.01)**	**5.59 (0.02)**
60 % violation (setting 7)	D-Learn	0.24 (0.05)	4.63 (0.20)	0.20 (0.04)	4.80 (0.17)
	PLS	0.20 (0.03)	4.80 (0.14)	0.18 (0.02)	4.89 (0.11)
	ACWL-C1	0.10 (0.02)	5.24 (0.08)	0.08 (0.01)	5.31 (0.07)
	ACWL-C2	0.10 (0.03)	5.22 (0.11)	0.08 (0.02)	5.32 (0.07)
	SR-Gaussian	**0.03 (0.01)**	**5.55 (0.04)**	**0.02 (0.01)**	**5.59 (0.03)**
0 % violation (setting 8)	D-Learn	0.23 (0.06)	4.70 (0.28)	0.17 (0.06)	4.94 (0.23)
	PLS	0.19 (0.06)	4.88 (0.26)	0.15 (0.07)	5.04 (0.28)
	ACWL-C1	0.16 (0.03)	4.97 (0.15)	0.12 (0.02)	5.15 (0.07)
	ACWL-C2	0.25 (0.04)	4.61 (0.18)	0.23 (0.04)	4.70 (0.18)
	SR-Gaussian	**0.04 (0.01)**	**5.51 (0.04)**	**0.02 (0.01)**	**5.56 (0.03)**
40 % violation (setting 8)	D-Learn	0.23 (0.05)	4.70 (0.23)	0.20 (0.04)	4.86 (0.20)
	PLS	0.24 (0.04)	4.67 (0.22)	0.23 (0.04)	4.73 (0.18)
	ACWL-C1	0.17 (0.03)	4.94 (0.16)	0.13 (0.02)	5.11 (0.08)
	ACWL-C2	0.24 (0.04)	4.60 (0.19)	0.22 (0.04)	4.69 (0.20)
	SR-Gaussian	**0.04 (0.01)**	**5.51 (0.06)**	**0.03 (0.01)**	**5.57 (0.04)**
60 % violation (setting 8)	D-Learn	0.23 (0.05)	4.68 (0.22)	0.21 (0.04)	4.79 (0.16)
	PLS	0.25 (0.03)	4.62 (0.20)	0.24 (0.01)	4.69 (0.09)
	ACWL-C1	0.16 (0.03)	4.92 (0.13)	0.13 (0.02)	5.09 (0.08)
	ACWL-C2	0.21 (0.04)	4.74 (0.17)	0.18 (0.03)	4.91 (0.13)
	SR-Gaussian	**0.05 (0.03)**	**5.44 (0.12)**	**0.03 (0.01)**	**5.54 (0.06)**
0 % violation (setting 9)	D-Learn	0.21 (0.06)	4.83 (0.25)	0.16 (0.04)	5.03 (0.18)
	PLS	0.14 (0.04)	5.12 (0.17)	0.12 (0.02)	5.19 (0.09)
	ACWL-C1	0.14 (0.03)	5.11 (0.16)	0.09 (0.02)	5.31 (0.08)
	ACWL-C2	0.27 (0.04)	4.56 (0.18)	0.26 (0.06)	4.61 (0.23)
	SR-Gaussian	**0.08 (0.01)**	**5.32 (0.06)**	**0.07 (0.01)**	**5.39 (0.03)**
40 % violation (setting 9)	D-Learn	0.21 (0.06)	4.80 (0.27)	0.16 (0.04)	5.02 (0.18)
	PLS	0.18 (0.03)	4.97 (0.13)	0.19 (0.01)	4.96 (0.05)
	ACWL-C1	0.13 (0.03)	5.11 (0.12)	0.09 (0.02)	5.28 (0.08)
	ACWL-C2	0.19 (0.05)	4.86 (0.23)	0.15 (0.05)	5.04 (0.23)
	SR-Gaussian	**0.10 (0.02)**	**5.24 (0.08)**	**0.08 (0.01)**	**5.32 (0.06)**
60 % violation (setting 9)	D-Learn	0.22 (0.05)	4.76 (0.25)	0.18 (0.04)	4.94 (0.18)
	PLS	0.19 (0.01)	4.91 (0.05)	0.20 (0.01)	4.91 (0.03)
	ACWL-C1	0.13 (0.02)	5.12 (0.11)	**0.09 (0.02)**	**5.30 (0.07)**
	ACWL-C2	0.17 (0.03)	4.95 (0.16)	0.13 (0.03)	5.15 (0.13)
	SR-Gaussian	**0.12 (0.04)**	**5.15 (0.15)**	**0.09 (0.02)**	5.29 (0.07)

phenomenon is that in the first sequential step, where {1} and {2,3} are compared, theoretically (according to (B.1)), the optimal decision is {1} if $E(Y|\mathbf{X}, A = 1) > \{E(Y|\mathbf{X}, A = 2) + E(Y|\mathbf{X}, A = 3)\}/2$, and {2,3}, if $E(Y|\mathbf{X}, A = 1) < \{E(Y|\mathbf{X}, A = 2) + E(Y|\mathbf{X}, A = 3)\}/2$. Therefore, under scenario 1, some patients whose true optimal $\mathcal{D}^*(\mathbf{X}) = 3$ may not be able to "pass" the first sequential step (S_1) (not "pass" S_1 means that {1} is recommended when compared with {2,3}), and the estimated optimal treatment for them might be 1 or 2 (depending on the result from the first re-estimation step, R_1), but not their true optimal, treatment 3. In addition, we may not have sufficient "eligible" samples to train the second sequential step classifier (S_2) if many patients are "stuck" at S_1, which also makes it harder for SR learning to estimate the optimal ITR correctly. However, OSM violation may not be very problematic under scenario 2, since even though OSM is violated (i.e. $E(Y|\mathbf{X}, A = 1) > E(Y|\mathbf{X}, A = 2)$) by some patients with $\mathcal{D}^*(\mathbf{X}) = 3$, $E(Y|\mathbf{X}, A = 1) < \{E(Y|\mathbf{X}, A = 2) + E(Y|\mathbf{X}, A = 3)\}/2$ holds, so these patients can still "pass" S_1 ("pass" S_1 means that {2,3} is recommended when compared with {1}) and be included in the training model for S_2 where treatments 2 and 3 are compared. In this case, SR learning may still be able to correctly identify their true optimal treatment (treatment 3), even under OSM violation.

B.2 Matlab codes for implementing the proposed SR learning

B.2.1 SR_main_3.m

This function implements the proposed SR learning method for linear and Gaussian kernels (with or without variable selection) and returns the optimal ITR among 3 ordinal treatments ($K = 3$). It calls functions `AOL`, `predict`, and `decision_3`.

- Input of the function:
  - `H`: covariate matrix (training data)
  - `A`: treatment assignment coded as 1, 2, 3
  - `Y`: observed outcome
  - `Htest`: covariate matrix (testing data)
  - `clinear`: a grid of tuning parameter for cost (used for cross-validation)
  - `sigmatune`: a grid of tuning parameter for bandwidth in the Gaussian kernel (used for cross-validation)
  - `m`: the number of folds of cross-validation for choosing tuning parameters for cost and bandwidth
  - `e`: the rounding error when computing the bias
  - `kernel`: the kernel function: 1 for linear kernel and 2 for Gaussian kernel
  - `select`: 0 for without variable selection and 1 for with variable selection

- Output of the function:
  - the optimal ITR among 3 ordinal treatments for the testing data

Code:

```
function model = SR_main_3(H,A,Y,Htest,clinear,sigmatune,m,e,kernel,select)
    % reset flagunique to 0
    flagunique=0;
    % Training
    % first sequential step S1
    % recode treatment
    A1=-1*(A==1)+1*(A~=1);
    if kernel == 2 && select == 1 % Gaussian kernel with variable selection
```

```
        % write file for R to read in
        csvwrite('covariates.csv',H);
        csvwrite('treatment.csv',A1);
        csvwrite('outcome.csv',Y);
        % call R script for SODA variable selection and to estimate ps
        !R CMD BATCH ps_glmnet_soda.R
        % read R output of ps estimates into Matlab
        ps1=csvread('ps_est.csv',1,1);
        % read R output of SODA variable selection results into Matlab
        select1=csvread('select.csv',1,1);
    else % linear kernel (with or without variable selection)
        % or Gaussian kernel without variable selection
        % write file for R to read in
        csvwrite('covariates.csv',H);
        csvwrite('treatment.csv',A1);
        % call R script to estimate ps
        !R CMD BATCH ps_glmnet.R
        % read R output of ps estimates into Matlab
        ps1=csvread('ps_est.csv',1,1);
        select1=[];
    end
    % call AOL function in S1
    aol_1=AOL(H,A1,Y,ps1,select1,clinear,sigmatune,m,e,kernel,select);
    % predict the optimal one between {1} and {2,3} in S1
    estopt1=predict(aol_1,H,kernel,select);
    % check whether estopt1 value is unique. Extreme case: only need R1
    if length(unique(estopt1))==1 && unique(estopt1)==-1
        flagunique=-1; % flagunique=-1 indicates only need R1
        % re-estimation R1
        % index of individuals eligible for training R1
        idx12=find(estopt1==-1&A~=3);
        Hsub12=H(idx12,:);
        Asub12=A(idx12);
        A12=-1*(Asub12==1)+1*(Asub12~=1);
        Y12=Y(idx12);
        % re-estimation step, no need to rerun SODA for both select=0 and 1
        csvwrite('covariates.csv',Hsub12);
        csvwrite('treatment.csv',A12);
        !R CMD BATCH ps_glmnet.R
        ps12=csvread('ps_est.csv',1,1);
        % call AOL function in R1
        aol_12=AOL(Hsub12,A12,Y12,ps12,select1,clinear,sigmatune,m,e,...
                   kernel,select);
    % extreme case: only need S2: 2 vs. 3
```

```
elseif length(unique(estopt1))==1 && unique(estopt1)==1
    flagunique=1; % flagunique=1 indicates only need S2
    % second sequential step S2
    idx2=find(estopt1==1&A~=1);
    Hsub2=H(idx2,:);
    Asub2=A(idx2);
    A2=-1*(Asub2==2)+1*(Asub2~=2);
    Y2=Y(idx2);
    if kernel == 2 && select == 1 % Gaussian kernel with vs
        csvwrite('covariates.csv',Hsub2);
        csvwrite('treatment.csv',A2);
        csvwrite('outcome.csv',Y2);
        !R CMD BATCH ps_glmnet_soda.R
        ps2=csvread('ps_est.csv',1,1);
        select2=csvread('select.csv',1,1);
    else % linear kernel (w/ or w/o vs) or Gaussian kernel w/o vs
        csvwrite('covariates.csv',Hsub2);
        csvwrite('treatment.csv',A2);
        !R CMD BATCH ps_glmnet.R
        ps2=csvread('ps_est.csv',1,1);
        select2=[];
    end
    % call AOL function in S2
    aol_2=AOL(Hsub2,A2,Y2,ps2,select2,clinear,sigmatune,m,e,...
              kernel,select);
else % normal case: both R1 and S2
    % re-estimation R1
    idx12=find(estopt1==-1&A~=3);
    if isempty(idx12)
        flagunique=1; % no eligible individual for R1
    else
        Hsub12=H(idx12,:);
        Asub12=A(idx12);
        A12=-1*(Asub12==1)+1*(Asub12~=1);
        Y12=Y(idx12);
        csvwrite('covariates.csv',Hsub12);
        csvwrite('treatment.csv',A12);
        !R CMD BATCH ps_glmnet.R
        ps12=csvread('ps_est.csv',1,1);
        aol_12=AOL(Hsub12,A12,Y12,ps12,select1,clinear,sigmatune,m,e,...
                   kernel,select);
    end
    % second sequential step S2
    idx2=find(estopt1==1&A~=1);
```

```
        if isempty(idx2)
            flagunique=-1; % no eligible individual for S2
        else
            Hsub2=H(idx2,:);
            Asub2=A(idx2);
            A2=-1*(Asub2==2)+1*(Asub2~=2);
            Y2=Y(idx2);
            if kernel == 2 && select == 1 % Gaussian kernel with vs
                csvwrite('covariates.csv',Hsub2);
                csvwrite('treatment.csv',A2);
                csvwrite('outcome.csv',Y2);
                !R CMD BATCH ps_glmnet_soda.R
                ps2=csvread('ps_est.csv',1,1);
                select2=csvread('select.csv',1,1);
            else % linear kernel (w/ or w/o vs) or Gaussian kernel w/o vs
                csvwrite('covariates.csv',Hsub2);
                csvwrite('treatment.csv',A2);
                !R CMD BATCH ps_glmnet.R
                ps2=csvread('ps_est.csv',1,1);
                select2=[];
            end
            aol_2=AOL(Hsub2,A2,Y2,ps2,select2,clinear,sigmatune,m,e,...
                      kernel,select);
        end
    end
    % Testing
    % S1
    estopttest1 = predict(aol_1,Htest,kernel,select);
    % R1
    if flagunique == 1 % extreme case: only S2, no R1
        estopttest12 = [];
    else % normal case: both R1 and S2
        estopttest12 = predict(aol_12,Htest,kernel,select);
    end
    % S2
    if flagunique == -1 % extreme case: only R1, no S2
        estopttest2 = [];
    else % normal case: both R1 and S2
        estopttest2 = predict(aol_2,Htest,kernel,select);
    end
    % return a vector of optimal treatments for testing data
    model=decision_3(estopttest1,estopttest12,estopttest2);
end
```

B.2.2 SR_main_4.m

This function implements the proposed SR learning method for linear and Gaussian kernels (with or without variable selection) and returns the optimal ITR among 4 ordinal treatments ($K = 4$). It calls functions `AOL`, `predict`, and `decision_4`.

- Input of the function:
  - `H`: covariate matrix (training data)
  - `A`: treatment assignment coded as 1, 2, 3, 4
  - `Y`: observed outcome
  - `Htest`: covariate matrix (testing data)
  - `clinear`: a grid of tuning parameter for cost (used for cross-validation)
  - `sigmatune`: a grid of tuning parameter for bandwidth in the Gaussian kernel (used for cross-validation)
  - `m`: the number of folds of cross-validation for choosing tuning parameters for cost and bandwidth
  - `e`: the rounding error when computing the bias
  - `kernel`: the kernel function: 1 for linear kernel and 2 for Gaussian kernel
  - `select`: 0 for without variable selection and 1 for with variable selection

- Output of the function:
  - the optimal ITR among 4 ordinal treatments for the testing data

Code:

```
function model = SR_main_4(H,A,Y,Htest,clinear,sigmatune,m,e,kernel,select)
    % reset flag
    % flag for whether there is any problem with re-estimation step R1
    flagproblemR1 = 0;
    % flag for whether there is any problem with re-estimation step R2
    flagproblemR2 = 0;
    % flag for whether there is any problem with sequential step S3
    flagproblemS3 = 0;

    % Training
    % first sequential step S1
    % recode treatment
    A1=-1*(A==1)+1*(A~=1);
```

```
if kernel == 2 && select == 1 % Gaussian kernel with vs
    % write file for R to read in
    csvwrite('covariates.csv',H);
    csvwrite('treatment.csv',A1);
    csvwrite('outcome.csv',Y);
    % call R script for SODA variable selection and to estimate ps
    !R CMD BATCH ps_glmnet_soda.R
    % read R output of ps estimates into Matlab
    ps1=csvread('ps_est.csv',1,1);
    % read R output of SODA variable selection results into MATLAB
    select1=csvread('select.csv',1,1);
else % linear kernel (with or without vs)
    % or Gaussian kernel without variable selection
    % write file for R to read in
    csvwrite('covariates.csv',H);
    csvwrite('treatment.csv',A1);
    % call R script to estimate ps
    !R CMD BATCH ps_glmnet.R
    % read R output of ps estimates into Matlab
    ps1=csvread('ps_est.csv',1,1);
    select1=[];
end
% call AOL function in S1
aol_1=AOL(H,A1,Y,ps1,select1,clinear,sigmatune,m,e,kernel,select);
% predict the optimal one between {1} and {2,3,4} in S1
estopt1=predict(aol_1,H,kernel,select);

% first re-estimation step R1
% index of individuals eligible for training R1
idx12=find(estopt1==-1&A~=3&A~=4);
if isempty(idx12)
    flagproblemR1 = 1;
else
    Hsub12=H(idx12,:);
    Asub12=A(idx12);
    A12=-1*(Asub12==1)+1*(Asub12~=1);
    Y12=Y(idx12);
    % re-estimation step, do not have to rerun SODA no matter select=0/1
    csvwrite('covariates.csv',Hsub12);
    csvwrite('treatment.csv',A12);
    !R CMD BATCH ps_glmnet.R
    ps12=csvread('ps_est.csv',1,1);
    % call AOL function in R1
    try
```

```
        aol_12=AOL(Hsub12,A12,Y12,ps12,select1,clinear,sigmatune,m,e,...
                   kernel,select);
    catch
        flagproblemR1 = 1;
    end
end

% second sequential step S2
% index of individuals eligible for training S2
idx2=find(estopt1==1&A~=1);
Hsub2=H(idx2,:);
Asub2=A(idx2);
A2=-1*(Asub2==2)+1*(Asub2~=2);
Y2=Y(idx2);
if kernel == 2 && select == 1 % Gaussian kernel with vs
    csvwrite('covariates.csv',Hsub2);
    csvwrite('treatment.csv',A2);
    csvwrite('outcome.csv',Y2);
    !R CMD BATCH ps_glmnet_soda.R
    ps2=csvread('ps_est.csv',1,1);
    select2=csvread('select.csv',1,1);
else % linear kernel (with or without vs)
    % or Gaussian kernel without variable selection
    csvwrite('covariates.csv',Hsub2);
    csvwrite('treatment.csv',A2);
    !R CMD BATCH ps_glmnet.R
    ps2=csvread('ps_est.csv',1,1);
    select2=[];
end
% call AOL function in S2
aol_2=AOL(Hsub2,A2,Y2,ps2,select2,clinear,sigmatune,m,e,...
          kernel,select);
estopt2=predict(aol_2,Hsub2,kernel,select);

% second re-estimation R2
% index of individuals eligible for training R2
idx23=find(estopt2==-1&Asub2~=4);
if isempty(idx23)
    flagproblemR2 = 1; % no eligible individual for R2
else
    Hsub23=Hsub2(idx23,:);
    Asub23=Asub2(idx23);
    A23=-1*(Asub23==2)+1*(Asub23~=2);
    Y23=Y2(idx23);
```

```
        % re-estimation step, do not have to rerun SODA no matter select=0/1
        csvwrite('covariates.csv',Hsub23);
        csvwrite('treatment.csv',A23);
        !R CMD BATCH ps_glmnet.R
        ps23=csvread('ps_est.csv',1,1);
        % call AOL function in R2
        try
            aol_23=AOL(Hsub23,A23,Y23,ps23,select2,clinear,sigmatune,m,e,...
                       kernel,select);
        catch
            flagproblemR2 = 1;
        end
    end

    % third sequential step S3
    % index of individuals eligible for training S3
    idx3=find(estopt2==1&Asub2~=2);
    if isempty(idx3)
         flagproblemS3 = 1; % no eligible individual for S3
    else
        Hsub3=Hsub2(idx3,:);
        Asub3=Asub2(idx3);
        A3=-1*(Asub3==3)+1*(Asub3~=3);
        Y3=Y2(idx3);
        if kernel == 2 && select == 1 % Gaussian kernel with vs
            csvwrite('covariates.csv',Hsub3);
            csvwrite('treatment.csv',A3);
            csvwrite('outcome.csv',Y3);
            !R CMD BATCH ps_glmnet_soda.R
            ps3=csvread('ps_est.csv',1,1);
            select3=csvread('select.csv',1,1);
        else % linear kernel (with or without vs)
            % or Gaussian kernel without vs
            csvwrite('covariates.csv',Hsub3);
            csvwrite('treatment.csv',A3);
            !R CMD BATCH ps_glmnet.R
            ps3=csvread('ps_est.csv',1,1);
            select3=[];
        end
        % call AOL function in S3
        try
            aol_3=AOL(Hsub3,A3,Y3,ps3,select3,clinear,sigmatune,m,e,...
                      kernel,select);
        catch
```

```
            flagproblemS3 = 1;
        end
    end

    % Testing
    % S1
    estopttest1 = predict(aol_1,Htest,kernel,select);
    % R1
    if flagproblemR1 == 1 % when there are problems with fitting the R1 model
        estopttest12 = [];
    else
        estopttest12 = predict(aol_12,Htest,kernel,select);
    end
    % S2
    estopttest2 = predict(aol_2,Htest,kernel,select);
    % R2
    if flagproblemR2 == 1 % when there are problems with fitting the R2 model
        estopttest23 = [];
    else
        estopttest23 = predict(aol_23,Htest,kernel,select);
    end
    % S3
    if flagproblemS3 == 1 % when there are problems with fitting the S3 model
        estopttest3 = [];
    else
        estopttest3 = predict(aol_3,Htest,kernel,select);
    end

    % return a vector of optimal treatments for testing data
    model = decision_4(estopttest1,estopttest12,estopttest2,...
                       estopttest23,estopttest3);
end
```

B.2.3 AOL.m

This function implements the augmented outcome weighted learning (AOL) method (Liu et al., 2018) with and without variable selection and returns the S-step and R-step binary classifiers trained with the best tuning parameters selected by cross-validation for linear or Gaussian kernel. It calls the function `hinge_svm`. It is adapted from the R package `DTRlearn` and the Matlab codes provided in Qiu et al. (2018).

- Input of the function:
  - `H`: covariate matrix
  - `A`: treatment assignment coded as 1 or -1
  - `R2`: outcome vector
  - `pi`: propensity score vector
  - `selectedvar`: indices of selected variables by SODA ("[ ]" if use linear kernel/Gaussian kernel without variable selection)
  - `clinear`: a grid of tuning parameter for cost (used for cross-validation)
  - `sigmatune`: a grid of tuning parameter for bandwidth in the Gaussian kernel (used for cross-validation)
  - `m`: the number of folds of cross-validation for choosing tuning parameters for cost and bandwidth
  - `e`: the rounding error when computing the bias
  - `kernel`: the kernel function: 1 for linear kernel and 2 for Gaussian kernel
  - `select`: 0 for without variable selection and 1 for with variable selection
- Output of the function:
  - a model estimated from the `hinge_svm` function with the best tuning parameters picked by cross-validation

Code:

```
function model = AOL(H,A,R2,pi,selectedvar,clinear,sigmatune,m,e,kernel,select)
    npar=length(clinear);
    nsig=length(sigmatune);
    n=length(A);
    % fit LASSO and calculate the residual
    if max(R2)~=min(R2)
        [fit,fi]=lasso(H,R2,'NumLambda',10,'CV',4);
        co=[fi.Intercept(fi.IndexMinMSE);fit(:,fi.IndexMinMSE)];
        r=R2-[ones(n,1),H]*co;
    else
        r=R2;
    end
    % calculate the weight vector
    r=r./pi;

    % splitting for cross-validation
    rand=mod(1:n,m)+1;
```

```
if kernel == 1 % linear Kernel
    V=zeros(m,npar);
    for i = 1:m
        this=(rand~=i);
        X=H(this,:);
        Y=A(this);
        R=r(this);
        Xt=H(~this,:);
        Yt=A(~this);
        Rt=r(~this);
        for j = 1:npar
            c=clinear(j);
            model=hinge_svm(X,Y,R,[],c,e,[],kernel,select);
            intercept=model{2};
            beta=model{3};
            YP=sign(intercept+Xt*beta);
            V(i,j)=sum(Rt.*(YP==Yt))/length(Yt);
        end
    end
    mimi=mean(V);
    [~,best]=max(mimi);
    % best cost parameter based on cv
    cbest=clinear(best);
    % train the model with the best cost parameter selected by cv
    model=hinge_svm(H,A,r,[],cbest,e,[],kernel,select);

elseif kernel == 2 % Gaussian kernel
    V=zeros(m,npar,nsig);
    for i = 1:m
        this=(rand~=i);
        X=H(this,:);
        Y=A(this);
        R=r(this);
        Xt=H(~this,:);
        Yt=A(~this);
        Rt=r(~this);
        for j = 1:npar
            for k = 1:nsig
                % cost parameter
                c=clinear(j);
                % sigma parameter in the Gaussian kernel
                sigma=sigmatune(k);
                model=hinge_svm(X,Y,R,selectedvar,c,e,sigma,kernel,select);
                alpha1 = model{1};
```

```
                    bias = model{2};
                    if select == 0 % without vs (Gaussian kernel)
                        Kt = exp(-sigma* pdist2(Xt,X,'squaredeuclidean'));
                    elseif select == 1 % with vs (Gaussian kernel)
                        Eselect = diag(model{4});
                        Kt = exp(-sigma* pdist2(Xt*Eselect,X*Eselect,...
                                'squaredeuclidean'));
                    end
                    YP=sign(bias+Kt*alpha1);
                    V(i,j,k)=sum(Rt.*(YP==Yt))/length(Yt);
                end
            end
        end
        mimi=reshape(mean(V,1),npar,nsig);
        [~,I]=max(mimi(:));
        [I_row,I_col]=ind2sub(size(mimi),I);
        % best cost and bandwidth parameters based on cv
        cbest=clinear(I_row);
        sigbest=sigmatune(I_col);
        % train the model with the best cost and bandwidth parameters
        model=hinge_svm(H,A,r,selectedvar,cbest,e,sigbest,kernel,select);
    else
        disp('Incorrect Kernel specification');
    end
end
```

B.2.4 hinge_svm.m

This function solves the dual formulation of the weighted SVM problem in AOL (Liu et al., 2018) for linear and Gaussian kernels without variable selection (a linear programming problem) and with variable selection (a quadratic programming problem). The codes with no variable selection is adapted from the R package `DTRlearn` and the Matlab codes provided in Qiu et al. (2018).

- Input of the function:
  - `X`: covariate matrix
  - `A`: treatment assignment coded by 1 and -1
  - `weight`: residual weighted by inverse propensity score
  - `selectedvar`: indices of selected variables by SODA ("[ ]" if use linear kernel/Gaussian kernel without variable selection)
  - `C`: cost parameter in the SVM formulation

  - `e`: rounding error when computing the bias
  - `sigma`: the tuning parameter for Gaussian kernel
  - `kernel`: the kernel function for weighted SVM: 1 for linear kernel and 2 for Gaussian kernel
  - `select`: 0 for without variable selection and 1 for with variable selection

- Output of the function:

  - linear kernel without variable selection, return a cell array with 4 elements:

    (i) the scaled (by weight) solution for the dual problem, $\boldsymbol{\alpha}_1$
    (ii) the intercept term, β_0
    (iii) coefficients for linear SVM, $\boldsymbol{\beta}$
    (iv) predicted treatment label (-1 or 1)

  - linear kernel with variable selection, return a cell array with 3 elements:

    (i) predicted treatment label (-1 or 1)
    (ii) the intercept term, β_0
    (iii) coefficients for linear SVM, $\boldsymbol{\beta}$

  - Gaussian kernel without variable selection, return a cell array with 6 elements:

    (i) the scaled (by weight) solution for the dual problem, $\boldsymbol{\alpha}_1$
    (ii) the intercept term, β_0
    (iii) the bandwidth parameter for the Gaussian kernel, σ
    (iv) fitted value for the nonlinear decision function in the training data
    (v) the covariate matrix of the training data
    (vi) predicted treatment label (-1 or 1)

  - Gaussian kernel with variable selection, return a cell array with 6 elements:

    (i) penalised coefficient vector associated with the kernel
    (ii) the intercept term, β_0
    (iii) the bandwidth parameter for the Gaussian kernel, σ
    (iv) variable selection diagonal matrix: the j^{th} diagonal element equals to 1 if variable j is selected, and equals to 0 if variable j is not selected
    (v) the covariate matrix of the training data
    (vi) predicted treatment label (-1 or 1)

Code:

```
function model = hinge_svm(X,A,weight,selectedvar,C,e,sigma,kernel,select)
    % lower bound on the size a step and the change on the
    % objective function during a step in QP optimisation
```

```
tol = 1e-9;

if kernel == 1 % linear kernel
    if select == 0
        e=e*C;
        n=size(A,1);
        AW=A.*weight;
        % calculate linear kernel function
        K=X*X';
        % calculate quadratic objective term
        H=K.*(AW*AW');
        % calculate linear objective term
        f=-abs(weight);
        % calculate lower bounds and upper bounds
        lb=zeros(n,1);
        ub=ones(n,1)*C;
        % linear equality constraints
        Aeq=AW';
        beq=0;
        % set optimisation options
        opts=optimset('Algorithm','interior-point-convex','TolX',tol,...
                      'TolFun',tol,'display','off');
        % solve the quadratic programming problem
        alpha=quadprog(H,f,[],[],Aeq,beq,lb,ub,[],opts);
        % calculate the scaled solution for the dual problem
        alpha1=alpha.*weight;
        % calculate beta coefficients
        beta=X'*(alpha1.*A);
        % calculate bias based on KKT conditions
        Imid =(alpha < C-e) & (alpha > e);
        rm=sign(weight).*A-X*beta;
        rmid=rm(Imid);
        if sum(Imid)>0
            bias=mean(rmid);
        else
            Iup=((alpha<e)&(A==-sign(weight)))|...
                ((alpha>C-e)&(A==sign(weight)));
            Ilow=((alpha<e)&(A==sign(weight)))|...
                 ((alpha>C-e)&(A==-sign(weight)));
            rup=rm(Iup);
            rlow=rm(Ilow);
            bias=(min(rup)+max(rlow))/2;
        end
```

```
        % calculate fitted value of linear decision function
        fit=bias+X*beta;
        % save results in a cell array with 4 elements
        model=cell(1,4);
        model{1}=alpha1;
        model{2}=bias; % beta0
        model{3}=beta; % beta
        model{4}=2*(fit>0)-1; % predicted treatment

    elseif select == 1 % with variable selection (linear kernel)
        [n,p]=size(X);
        % coefficient vector for LP
        f=[ones(p,1);ones(p,1);0;reshape(C.*abs(weight),[n,1])];
        % lower bound
        lb=[zeros(p,1);zeros(p,1);-Inf;zeros(n,1)];
        % linear inequality constraints
        Aineq=[-diag(A.*sign(weight))*X diag(A.*sign(weight))*X...
              reshape(-A.*sign(weight),[n,1]) -eye(n)];
        b=-ones(n,1);
        % optimisation options
        opts=optimset('Algorithm','dual-simplex','TolFun',tol,...
                      'display','off');
        % solve the linear programming problem
        result=linprog(f,Aineq,b,[],[],lb,[],opts);
        % beta0
        bias=result(2*p+1);
        % beta
        beta=result(1:p)-result((p+1):2*p);
        % fitted decision function
        fit=bias+X*beta;
        % save results in a cell array with 3 elements
        model=cell(1,3);
        model{1}=2*(fit>0)-1; % predicted treatment
        model{2}=bias; % beta0
        model{3}=beta; % beta
    end

elseif kernel == 2 % Gaussian kernel
    if select == 0 % without variable selection
        e=e*C;
        n=size(A,1);
        AW=A.*weight;
        % calculate Gaussian kernel function
        K=exp(-sigma* pdist2(X,X,'squaredeuclidean'));
```

```
        % calculate quadratic objective term
        H=K.*(AW*AW');
        % calculate linear objective term
        f=-abs(weight);
        % calculate lower bounds and upper bounds
        lb=zeros(n,1);
        ub=ones(n,1)*C;
        % calculate linear equality constraints
        Aeq=AW';
        beq=0;
        % set optimisation options
        opts=optimset('Algorithm','interior-point-convex','TolX',tol,...
                      'TolFun',tol,'display','off');
        % solve the quadratic programming problem
        alpha=quadprog(H,f,[],[],Aeq,beq,lb,ub,[],opts);
        % calculate the scaled solution for the dual problem
        alpha1=alpha.*weight.*A;
        % calculate bias based on KKT conditions
        Imid =(alpha < C-e) & (alpha > e);
        rm = sign(weight).*A-K*alpha1;
        rmid = rm(Imid);
        if sum(Imid)>0
            bias=mean(rmid);
        else
            Iup=((alpha<e)&(A==-sign(weight)))|...
                ((alpha>C-e)&(A==sign(weight)));
            Ilow=((alpha<e)&(A==sign(weight)))|...
                 ((alpha>C-e)&(A==-sign(weight)));
            rup=rm(Iup);
            rlow=rm(Ilow);
            bias=(min(rup)+max(rlow))/2;
        end
        % calculate fitted value of nonlinear decision function (Gaussian)
        fit=bias+K*alpha1;
        % save results in a cell array with 6 elements
        model=cell(1,6);
        model{1}=alpha1;
        model{2}=bias; % beta0
        model{3}=sigma; % sigma tuning parameter
        model{4}=fit; % fit
        model{5}=X; % X matrix
        model{6}=2*(fit>0)-1; % predicted treatment;
    elseif select == 1 % with variable selection (Gaussian kernel)
        [n,p]=size(X);
```

```
            % coefficient vector for LP
            f=[ones(n,1);ones(n,1);0;reshape(C.*abs(weight),[n,1])];
            % lower bound
            lb=[zeros(n,1);zeros(n,1);-Inf;zeros(n,1)];
            % initialise vector of 0 (length p)
            main=zeros(p,1);
            % 1 for selected covariates and 0 for unselected covariates
            main(selectedvar)=1;
            % create diagonal matrix E based on the vector "main"
            E=diag(main);
            % calculate Gaussian kernel function
            K=exp(-sigma* pdist2(X*E,X*E,'squaredeuclidean'));
            % linear inequality constraints
            Aineq=[-diag(A.*sign(weight))*K diag(A.*sign(weight))*K...
                  reshape(-A.*sign(weight),[n,1]) -eye(n)];
            b=-ones(n,1);
            % optimisation options
            opts=optimset('Algorithm','dual-simplex','TolFun',tol,...
                          'display','off');
            % solve the linear programming problem
            result=linprog(f,Aineq,b,[],[],lb,[],opts);
            % beta0
            bias=result(2*n+1);
            % penalised coefficient vector associated with kernel
            v=result(1:n)-result((n+1):2*n);
            alpha=v;
            % calculate fitted value of nonlinear decision function (Gaussian)
            fit=K*alpha+bias;
            % save results in a cell array with 6 elements
            model=cell(1,6);
            model{1}=alpha; % coefficient vector associated with kernel
            model{2}=bias; % intercept term
            model{3}=sigma; % sigma tuning parameter
            model{4}=diag(E); % variable selection diagonal matrix
            model{5}=X; % X matrix
            model{6}=2*(fit>0)-1; % predicted treatment
        end
    else
        disp('Incorrect Kernel specification');
    end
end
```

B.2.5 decision_3.m

This function derives the optimal ITR among 3 ordinal treatments based on binary treatment selection decisions in S- and R-steps.

- Input of the function:
  - `a1`: a vector representing binary decisions in S_1, i.e. {1} vs. {2,3}
  - `a12`: a vector representing binary decisions in R_1, i.e. {1} vs. {2}
  - `a2`: a vector representing binary decisions in S_2, i.e. {2} vs. {3}
- Output of the function:
  - `itr_predict`: a vector of best treatments among 3 ordinal treatments

Code:

```
function itr_predict = decision_3(a1,a12,a2)
    if isempty(a12) % extreme case 1, empty R1
        itr_predict = 2*((a1==-1)+(a1==1&a2==-1))+3*(a1==1&a2==1);
    elseif isempty(a2) % extreme case 2, empty S2
        itr_predict = (a1==-1&a12==-1)+2*((a1==-1&a12==1)+(a1==1));
    else % normal case
        itr_predict = (a1==-1&a12==-1)+2*((a1==-1&a12==1)+(a1==1&a2==-1))+...
                      3*(a1==1&a2==1);
    end
end
```

B.2.6 decision_4.m

This function derives the optimal ITR among 4 ordinal treatments based on binary treatment selection decisions in S- and R-steps.

- Input of the function:
  - `a1`: a vector representing binary decisions in S_1, i.e. {1} vs. {2,3,4}
  - `a12`: a vector representing binary decisions in R_1, i.e. {1} vs. {2}
  - `a2`: a vector representing binary decisions in S_2, i.e. {2} vs. {3,4}
  - `a23`: a vector representing binary decisions in R_2, i.e. {2} vs. {3}
  - `a3`: a vector representing binary decisions in S_2, i.e. {3} vs. {4}
- Output of the function:
  - `itr_predict`: a vector of best treatments among 4 ordinal treatments

Code:

```
function itr_predict = decision_4(a1,a12,a2,a23,a3)
    %extreme case 1, empty R1
    if isempty(a12) && ~isempty(a23) && ~isempty(a3)
        itr_predict = 2*((a1==-1)+(a1==1&a2==-1&a23==-1))+...
                      3*((a1==1&a2==-1&a23==1)+(a1==1&a2==1&a3==-1))+...
                      4*(a1==1&a2==1&a3==1);
    %extreme case 2, empty R2
    elseif ~isempty(a12) && isempty(a23) && ~isempty(a3)
        itr_predict = 1*(a1==-1&a12==-1)+...
                      2*(a1==-1&a12==1)+...
                      3*((a1==1&a2==-1)+(a1==1&a2==1&a3==-1))+...
                      4*(a1==1&a2==1&a3==1);
    % extreme case 3, empty R1 and R2
    elseif isempty(a12) && isempty(a23) && ~isempty(a3)
        itr_predict = 2*(a1==-1)+...
                      3*((a1==1&a2==-1)+(a1==1&a2==1&a3==-1))+...
                      4*(a1==1&a2==1&a3==1);
    % extreme case 4, empty R1 and S3
    elseif isempty(a12) && ~isempty(a23) && isempty(a3)
        itr_predict = 2*((a1==-1)+(a1==1&a2==-1&a23==-1))+...
                      3*((a1==1&a2==-1&a23==1)+(a1==1&a2==1));
    % extreme case 5, empty R2 and S3
    elseif ~isempty(a12) && isempty(a23) && isempty(a3)
        itr_predict = 1*(a1==-1&a12==-1)+...
                      2*(a1==-1&a12==1)+...
                      3*((a1==1&a2==-1)+(a1==1&a2==1));
    % extreme case 6, empty R1, R2, and S3
    elseif isempty(a12) && isempty(a23) && isempty(a3)
        itr_predict = 2*(a1==-1)+3*(a1==1);
    % extreme case 7, empty S3
    elseif ~isempty(a12) && ~isempty(a23) && isempty(a3)
        itr_predict = 1*(a1==-1&a12==-1)+...
                      2*((a1==-1&a12==1)+(a1==1&a2==-1&a23==-1))+...
                      3*((a1==1&a2==-1&a23==1)+(a1==1&a2==1));
    % normal case
    else
        itr_predict = 1*(a1==-1&a12==-1)+...
                      2*((a1==-1&a12==1)+(a1==1&a2==-1&a23==-1))+...
                      3*((a1==1&a2==-1&a23==1)+(a1==1&a2==1&a3==-1))+...
                      4*(a1==1&a2==1&a3==1);
    end
end
```

Appendix C

Supplements for Chapter 4

C.1 R codes for implementing the patient stratification method

In this section, we provide the sample R codes for implementing the patient stratification method proposed in Chapter 4.

```
library(BART)
library(PReMiuMar)
library(mclust)
library(mcclust)
library(infotheo)
library(dplyr)
library(aricode)

# data preparation

# A: treatment which takes the value of 1, 2, or 3
# Y: continuous outcome
# cov.all: continuous covariates x1, x2, and x3
data <- data.frame(cbind(A,Y,cov.all))
n <- dim(data)[1]

# step 1: predict potential outcomes using BART

# set the number of iterations in BART
```

```
nburnin <- 1000
npost <- 5000

# prediction dataset in BART (single-learner)
test.1 <- cbind(rep(1,n),cov.all)
test.2 <- cbind(rep(2,n),cov.all)
test.3 <- cbind(rep(3,n),cov.all)
data.test <- rbind(test.1,test.2,test.3)

# BART model fitting and get predictions
model <- wbart(data[,-2],data[,2],data.test,nskip=nburnin,
               ndpost=npost)

# predicted potential outcomes (posterior mean)
yhat.1 <- model$yhat.test.mean[1:n]
yhat.2 <- model$yhat.test.mean[n+(1:n)]
yhat.3 <- model$yhat.test.mean[2*n+(1:n)]

# dataset that will be used in step 2 (for clustering)
dat <- data.frame(cbind(cov.all,yhat.1,yhat.2,yhat.3))

# step 2: profile regression for clustering

# names of covariates
cov.names <- c("x1","x2","x3")

# run profile regression model
runInfoObj <- profRegr(yModel="MVN",xModel="Normal",nSweeps=1000,
                       nBurn=1000,data=dat,covNames=cov.names,
                       outcome=c("yhat.1","yhat.2","yhat.3"))
# note: we can add a "predict" argument in the profRegr function
# if we need to predict the cluster membership for a new subject.
# Details can be found in the R documentation for profRegr.

# calculate the dissimilarity matrix
dissimObj <- calcDissimilarityMatrix(runInfoObj)
# note: if "predict" argument is included in profRegr, the output
# of the calcDissimilarityMatrix also includes a dissimilarity
# matrix between predicted subjects and training subjects.
```

```
# calculate the "representative" clustering
clusObj <- calcOptimalClustering(dissimObj)

# clustering allocation of the "representative" clustering
clust.nolim <- clusObj$clustering
# note: if the "predict" argument is included in profRegr, we can
# get the predicted cluster memberships for new subjects using
# clusObj$clusteringPred.

# number of clusters in the "representative" clustering
numclust <- clusObj$nClusters

# calculate adjusted Rand index
# Z is the true clustering (ground truth)
ari <- adjustedRandIndex(clust.nolim,Z)

# calculate homogeneity and completeness
homo <- v.measure(Z,clust.nolim)[1]
complete <- v.measure(Z,clust.nolim)[2]

# calculate average risks and profiles
riskProfileObj <- calcAvgRiskAndProfile(clusObj)

# plot covariate and outcome profiles
clusterOrderObj <- plotRiskProfile(riskProfileObj,"profile.png")
```