Tree-based Ensemble Methods For Individualized Treatment Rules

Kehao Zhu

A thesis

submitted in partial fulfillment of the

requirements for the degree of

Master of Science

University of Washington

2016

Committee:

Xiao-Hua (Andrew) Zhou

Ying Huang

Program Authorized to Offer Degree:

UW Biostatistics

University of Washington

Abstract

Tree-based Ensemble Methods For Individualized Treatment Rules

Kehao Zhu

Co-Chairs of the Supervisory Committee:

Professor Xiao-Hua (Andrew) Zhou
Biostatistics

Affiliate Associate Professor Ying Huang
Biostatistics

There is a growing interest in statistical methods for the personalized medicine or precision medicine, especially for deriving optimal individualized treatment rules (ITRs). An ITR recommends a patient to a treatment based on the patient's characteristics. The common parametric methods for deriving optimal ITR, which model the clinical endpoint as a function of the patient's characteristics in the first step, can have suboptimal performance when the conditional mean model is misspecified. Recent methodology development has cast the problem of deriving optimal ITR under a weighted classification framework. Under this weighted classification framework, we develop a weighted random forests (W-RF) algorithm that derives an optimal ITR nonparametrically. In addition, with the W-RF algorithm, we propose the variable importance measures for quantifying relative relevance of the patient's characteristics to treatment selection, and the out-of-bag estimator for the population average outcome under the estimated optimal ITR. Our proposed methods are evaluated through intensive simulation studies. We apply our methods to data from Clinical Antipsychotic Trials of Intervention Effectiveness Alzheimer's Disease Study (CATIE-AD) as an illustration.

Contents

1 Introduction **1**

2 Review Of Current Research **5**

2.1 Setting And Definition Of The Optimal ITR 5

2.2 Derive Optimal ITR Based On $\hat{\Delta}(\mathbf{X})$ 8

2.3 Derive Optimal ITR Under Weighted Classification Framework 12

3 Random Forests Under The Weighted Classification Framework **17**

3.1 A Single Decision Tree 18

3.2 Weighted Random Forests For Estimating ITR 20

3.3 Evaluation Of The Derived Rule 22

3.4 Variable Importance 24

4 Simulation Study **26**

4.1 Simulation setup 26

4.2 Results 31

4.2.1 Model Performance Comparisons 31

4.2.2 OOB Estimation Of Population Average Outcome 35

4.2.3 Variable Importance Measures 37

5 Data Example **40**

6 Discussion **46**

Appendix A R Functions To Implement W-RF **52**

Appendix B A Demonstration Of The R Functions In Appendix A **61**

List of Tables

4.1 Results for Scenario I summarized from 500 simulated datasets. The sample mean and standard error of the bias ($E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] - E[\tilde{Y}(\hat{g}(\tilde{\mathbf{X}}))]$) and misclassification error rate (MCR) are shown. $E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] = 5.99$; $E[Y(1)] = \hat{E}[Y(0)] = 4.13$. 33

4.2 Results for Scenario II summarized from 500 simulated datasets. The sample mean and standard error of the bias ($E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] - E[\tilde{Y}(\hat{g}(\tilde{\mathbf{X}}))]$) and misclassification error rate (MCR) are shown. $E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] = 7.77$; $E[Y(1)] = 5.13$; $E[Y(0)] = 4.14$. 34

4.3 Results for Scenario III summarized from 500 simulated datasets. The sample mean and standard error of the bias ($E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] - E[\tilde{Y}(\hat{g}(\tilde{\mathbf{X}}))]$) and misclassification error rate (MCR) are shown. $E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] = 6.66$; $E[Y(1)] = 4.17$; $E[Y(0)] = 4.14$. 35

4.4 The sample mean and standard errors of $\hat{E}[Y(\hat{g})]$ of 500 simulations based on estimates from a large testing set (N=10,000), Out-of-Bag (OOB) estimate, 5-fold cross-validated (CV) estimate, and original training set (Naive estimate). 36

4.5 Results for summary statistics U of two types of variable importance measures in different scenarios. The sample mean and standard deviation (in parentheses) from 500 simulated datasets are shown. 37

5.1 Cross-validated estimates of response rate under derived ITR ($\hat{E}[Y(\hat{g})]$) and proportion of the population recommended to treat ($\hat{P}(\hat{g} = 1)$) from different methods. 95% percentile bootstrap confidence intervals are shown in the parentheses 42

List of Figures

4.1 Three different simulation scenarios 27

4.2 The distributions of two types of variable importance measures in Scenario I-A in 500 simulated datasets using AIPWE for $\Delta(\mathbf{X})$. The first two boxplots are distributions of VI of X_1 and X_2 (in red color), and the other boxplots of VIs of the rest of $\mathbf{X}$ are in blue color. 38

4.3 The distributions of two types of variable importance measures in Scenario I-B in 500 simulated datasets using AIPWE for $\Delta(\mathbf{X})$. The first two boxplots are distributions of VIs of X_1 and X_2 (in red color), and the other boxplots of VIs of the rest of $\mathbf{X}$ are in blue color. 39

5.1 Variable importance of 49 baseline covariates 43

5.2 CATE curves of 4 most relevant covariates selected by the variable importance measures of W-RF 44

ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisors, Andrew Zhou and Ying Huang, for their guidance, support, encouragement and patience during my thesis project. I would like to acknowledge CATIE-AD study group led by Lon Schneider and Pierre Tariot for providing the clinical trial dataset used in this thesis. I thank the faculty, staff and students in the Department of Biostatistics for the supportive and friendly environment. Finally, I wish to express my gratitude to my family and friends for their support and belief in me.

DEDICATION

To my family

Chapter 1

Introduction

With recent advancement in basic science and technology (e.g. gene expression profiling), precision medicine or personalized medicine has been advocated in medical community and influenced policymaking (National Research Council, 2011). In 2015, U.S. government launched a $215 million Precision Medicine Initiative (Collins and Varmus, 2015). The basic concept of precision medicine is to make medical decisions, such as prescribing a particular drug or taking preventive health measures, based on individual patients' characteristics, including clinical information, demographics, genetics, environmental factors, etc.

A well-known example of personalized medicine is the preventative management of women who carry BRAC1 or BRAC2 mutations. Linkage studies and molecular biology confirmed that women with the mutations have higher risk of developing breast and ovarian cancer (Ford et al., 1998; Venkitaraman, 2002). Researchers developed individualized preventive management, such as removal of breasts or ovaries for women carrying BRAC1 or BRAC2 mutations (Roukos and Briasoulis, 2007). In biomarker research, individual measurements that predict the risk of developing a diease, like BRAC1 and BRAC2 mutations in this example, are called prognostic biomarkers. A prognostic biomarker is used to identify individuals with higher risk of certain disease.

Another type of biomarkers called prescriptive or predictive biomarkers can inform whether patients will benefit from a treatment. Patients who carry different values of predictive biomarkers would show differential treatment effects. A classic example comes from a clinical trial for women with primary operable breast cancer and positive axillary nodes from National Surgical Adjuvant Breast and Bowel Project (NSABP). There were two treatment arms in the trial, L-phenylalanine mustard and 5-fluorouracil with or without tamoxifen (RF or RFT). Researchers identified patients under 50 years old and with progesterone receptor level smaller than 10 femtomoles to be a subgroup that would benefit from RF, and other patients included in trials would benefit from RFT (Fisher et al., 1983). Patients' progesterone receptor level in this example is a predictive biomarker. An individualized treatment rule (ITR) can be developed based on predictive biomarkers and other predictive factors, including demographic, clinical or environmental information. In the previous example, an ITR can be described as follows: 1) if a patient is under 50 years old and has progesterone receptor level smaller than 10 femtomoles, we recommend to treat her with RF 2) elsewise, we recommend to treat her with RFT.

The above finding of NSABP is a result of subgroup analyses (Fisher et al., 1983). In the standard subgroup analyses, researchers pre-specify subgroups defined by patients' baseline covariates based on a biologically plausible mechanism. The results of RCT are analyzed separately by different subgroups, and the coefficient of the interaction term between treatment and pre-specified baseline covariates is tested (Pocock et al., 2002). The goal of the standard subgroup analyses is to find a subgroup of patients for whom the treatment is particularly effective. The treatment is recommended for those patients in a subgroup who show beneficial outcome in the treatment arm. Subgroup analyses can be viewed as indirect ways for deriving an ITR for the whole population represented by RCT participants, and are inefficient when multiple biomarkers or covariates are examined as potential predictive factors. Subgroup analyses are not the focus of this thesis.

The focus of this thesis is on statistical methods of deriving optimal individual

treatment rules based on data from randomized clinical trials (RCT). An optimal ITR is defined as a rule that maximizes the overall population average outcome under the rule if the outcome is clinically desired, such as CD4 counts in HIV patients or a rule that minimizes the overall population average outcome under the rule if the experimental treatment tries to prevent the outcome, such as tumor sizes. There are two general approaches for deriving an optimal ITR.

The first general approach is to estimate the difference of average outcomes between two treatment arms conditional on predictive biomarkers. This conditional treatment difference is natural for deriving ITR. To estimate the conditional treatment difference, we may use a generalized linear model, including the treatment, biomarkers and the interaction terms between treatment and biomarkers as predictors and clinical endpoint as outcome (Janes et al., 2014). This parametric approach suffers from sensitivity to misspecification of conditional mean model, especially when the biomarkers are high dimensional. When the conditional mean model is misspecified, the estimated conditional treatment difference is biased, and ITR based on the biased estimates will be suboptimal. Several methods have been proposed to reduce the bias in estimating the conditional treatment difference caused by model misspecification (Qian and Murphy, 2011; Matsouaka et al., 2014; Kang et al., 2014).

The second general approach targets at direct optimization of the population average outcome under an ITR. In 2012, three independent groups proposed a new framework of deriving ITR that recasts the problem of maximization of treatment benefit as a weighted classification problem (Zhang et al., 2012a; Rubin and van der Laan, 2012; Zhao et al., 2012). Several methods were proposed under this framework (Zhao et al., 2012; Huang and Fong, 2014; Huang, 2015). In this thesis, we propose to use the techniques of random forests (Breiman, 2001) under this framework to solve the weighted classification problem in order to estimate an optimal ITR.

The reminder of this thesis is organized as follows. In Chapter 2, we review the current research in deriving optimal ITR. We extend the random forests method to solve the weighted classification problem and treatment selection problem in Chapter 3. In Chapter 4, we conduct stimulation studies. In Chapter 5, we use data from Clinical Antipsychotic Trials of Intervention Effectiveness Alzheimer's Disease Study (CATIE-AD) to demonstrate the application of our proposed method. The thesis is concluded in Chapter 6 with discussions on the topic.

Chapter 2

Review Of Current Research

2.1 Setting And Definition Of The Optimal ITR

We consider data collected from a two-arm randomized clinical trial. Let Y be the observed clinical outcome, which can be either a continuous outcome or a binary outcome. Without loss of generality, we assume Y is an undesired binary outcome (e.g. death), which the treatment intends to prevent. Hence, $E(Y)$ is the event rate of the binary outcome in the population. Let $\mathbf{X}$ be p-dimensional baseline covariates, which can be either categorical or continuous. Let A be the treatment assignment, with $A = 1$ for the experimental treatment and $A = 0$ for the control or standard of care. By randomization, A is independent of $\mathbf{X}$. We observe n copies of independent and identically distributed $(Y_i, \mathbf{X_i}, A_i)$ from the RCT, where $i = 1, ..., n$. Let $g(\mathbf{X}) : \mathcal{X} \to (0, 1)$ be the individualized treatment rule (ITR) based on covariates $\mathbf{X}$, which maps from the covariates space to treatment decision: to treat patients with covariates $\mathbf{X}$ if $g(\mathbf{X}) = 1$ or not to treat if $g(\mathbf{X}) = 0$.

It is helpful to describe the ITR under the potential outcome framework. Let Y(a) be the potential outcome under $a \in (0, 1)$. Here, we assume the Stable Unit Treatment Value Assumption (SUTVA); i.e., there is no multiple version of the same treatment, and

one patient's outcome does not depend on any other patient's treatment assignment. Under SUTVA, the observed outcome is connected with the potential outcome by $Y = Y(1)A + Y(0)(1-A)$. By randomization, we have $(Y(0), Y(1)) \perp A$. Hence, we have $E(Y(a)|\mathbf{X}) = E(Y|A = a, \mathbf{X})$; i.e., the potential event rate for treatment option $A = a$ conditional on $\mathbf{X}$ can be estimated by the event rate among patients with baseline covariates $\mathbf{X}$ in the treatment arm $A = a$. Let

$$\Delta(\mathbf{X}) = E(Y(1)|\mathbf{X}) - E(Y(0)|\mathbf{X}) = E(Y|A = 1, \mathbf{X}) - E(Y|A = 0, \mathbf{X})$$

be the difference in the potential event rates between two treatment options, conditional on $\mathbf{X}$. For a subpopulation defined by baseline coveriates $\mathbf{X} = \mathbf{x}$, to minimize the expected event rate, a treatment rule for this subpopulation is to treat them if $\Delta(\mathbf{x}) < 0$ and not to treat them if $\Delta(\mathbf{x}) \geq 0$. As shown in the following paragraph, if we apply this treatment rule to every subpopulation, the expected event rate of the whole population will be minimized.

To develop ITR under the potential outcome framework, let's define the potential outcome if we follow ITR $g(\mathbf{X})$ as follows:

$$Y(g(\mathbf{X})) = Y(1)g(\mathbf{X}) + Y(0)[1 - g(\mathbf{X})].$$

Then, $E_{\mathbf{X},Y}[Y(g(\mathbf{X}))]$ is the average population outcome if we follow ITR. The optimal ITR based on $\mathbf{X}$ is defined as follows:

$$g^{opt}(\mathbf{X}) = argmin_{g(\mathbf{X})} E[Y(g(\mathbf{X}))];$$

i.e., an ITR is considered optimal if we can minimize the population event rate under this

ITR. We can rewrite $E[Y(g(\mathbf{X}))]$ as follows:

$$\begin{aligned} E[Y(g(\mathbf{X}))] =& E[Y(1)g(\mathbf{X}) + Y(0)(1 - g(\mathbf{X}))] = E[\{Y(1) - Y(0)\}g(\mathbf{X})] + E[Y(0)] \\ =& E_{\mathbf{X}}E[\{Y(1) - Y(0)\}g(\mathbf{X})|\mathbf{X}] + E[Y(0)] \\ =& E_{\mathbf{X}}[g(\mathbf{X})\{E(Y|A = 1, \mathbf{X}) - E(Y|A = 0, \mathbf{X})\}] + E[Y(0)] \\ =& E_{\mathbf{X}}[g(\mathbf{X})\Delta(\mathbf{X})] + E[Y(0)]. \end{aligned} \tag{2.1}$$

Hence, $g^{opt}(\mathbf{X}) = argmin_{g(\mathbf{X})}E[Y(g(\mathbf{X}))] = argmin_{g(\mathbf{X})}E_X[g(\mathbf{X})\Delta(\mathbf{X})] = \mathbb{1}(\Delta(\mathbf{X}) < 0)$, where $\mathbb{1}(\cdot)$ is the indicator function (Zhang et al., 2012a).

In addition, we notice that this optimal ITR defined by minimization of the population event rate under the ITR has nice interpretations as average casual effect. $g^{opt}(\mathbf{X}) = \mathbb{1}(\Delta(\mathbf{X}) < 0)$ also maximizes following four quantities:

$$E[Y(1)] - E[Y(g(\mathbf{X}))], \tag{2.2a}$$

$$E[Y(0)] - E[Y(g(\mathbf{X}))], \tag{2.2b}$$

$$min(E[Y(1)], E[Y(0)]) - E[Y(g(\mathbf{X}))], \tag{2.2c}$$

$$E[Y(\bar{g}(\mathbf{X}))] - [Y(g(\mathbf{X}))], \tag{2.2d}$$

where, $\bar{g}(\mathbf{X}) = 0$ if $g(\mathbf{X}) = 1$ and $\bar{g}(\mathbf{X}) = 1$ if $g(\mathbf{X}) = 0$; $Y(\bar{g}(\mathbf{X})) = Y(0)\bar{g}(\mathbf{X}) + Y(1)(1 - \bar{g}(\mathbf{X}))$ is the potential outcome if we defy $g(\mathbf{X})$.

(2.2a) is the difference of average event rate if we follow $g(\mathbf{X})$, compared to a rule that would treat all patients; (2.2b) is the difference of average event rate if we follow $g(\mathbf{X})$, compared to a rule that would treat no patient; (2.2c) is the difference of average event rate if we follow $g(\mathbf{X})$, compared to a rule that assign all patients to a treatment option which minimizes the whole population event rate; (2.2d) is the difference of average event rate comparing $g(\mathbf{X})$ to $\bar{g}(\mathbf{X})$. Therefore, $g^{opt}(\mathbf{X}) = \mathbb{1}(\Delta(\mathbf{X}) < 0)$ maximizes the benefit, the difference in event rate, compared to other possible rules.

Note that the above results can be naturally extended to a continuous outcome. If a lower clinical outcome Y is desired, the optimal ITR is still $g^{opt}(\mathbf{X}) = \mathbb{1}(\Delta(\mathbf{X}) < 0)$. If a higher clinical continuous outcome or higher event rate of binary outcome is desired, the optimal ITR will be defined as $g^{opt}(\mathbf{X}) = argmax_{g(\mathbf{X})}E[Y(g(\mathbf{X}))]$ and, following the same derivation, we will have the optimal ITR to be $\mathbb{1}(\Delta(\mathbf{X}) > 0)$. Because $g^{opt}(\mathbf{X}) = \mathbb{1}(\Delta(\mathbf{X}) < 0)$ is the optimal rule, many research in predictive biomarkers and ITR focuses on estimating $\Delta(\mathbf{X})$, which are reviewed in the next section.

2.2 Derive Optimal ITR Based On $\hat{\Delta}(\mathbf{X})$

A straightforward approach to estimate $\Delta(\mathbf{X})$ is to impose a parametric model for the outcome. For example, we can assume a generalized linear model (GLM) with a mean model: $E(Y|A, \mathbf{X}, \boldsymbol{\beta}, \boldsymbol{\gamma}) = f^{-1}(\beta_0 + \beta_1 A + \mathbf{X}\boldsymbol{\gamma}_0 + A\mathbf{X}\boldsymbol{\gamma}_1)$. $f(\cdot)$ is the link function (e.g. logit function is commonly used for binary outcome), and $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$ are the regression parameters. Under the proposed model, $\Delta(\mathbf{X})$ is estimated by:

$$\Delta(\mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) = E(Y|A = 1, \mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) - E(Y|A = 0, \mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}).$$

However, like all parametric approaches, this approach is prone to model misspecification, especially misspecification of the mean model. In fact, the misspecification of mean model can have huge impact on the ITR estimated based on the parametric model. An estimated ITR based on proposed mean model, $\hat{g}^{opt}(\mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}})$, is restricted to a class of decision rule (Zhang et al., 2012b). Note that

$$\begin{aligned}
\hat{g}^{opt}(\mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) =& \mathbb{1}(\Delta(\mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) < 0) \\
=& \mathbb{1}(E(Y|A = 1, \mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) - E(Y|A = 0, \mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) < 0) \\
=& \mathbb{1}[f^{-1}(\hat{\beta}_0 + \hat{\beta}_1 + \mathbf{X}(\hat{\boldsymbol{\gamma}}_0 + \hat{\boldsymbol{\gamma}}_1)) - f^{-1}(\hat{\beta}_0 + \mathbf{X}\hat{\boldsymbol{\gamma}}_0) < 0].
\end{aligned}$$

The inverse of link function f^{-1} is monotone, hence

$$\hat{g}^{opt}(\mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) = \mathbb{1}[\mathbf{X}\hat{\boldsymbol{\gamma}}_1 + \hat{\beta}_1 > 0] \text{ if } f^{-1} \text{ is descreasing,}$$
$$\text{and } \hat{g}^{opt}(\mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) = \mathbb{1}[\mathbf{X}\hat{\boldsymbol{\gamma}}_1 + \hat{\beta}_1 < 0] \text{ if } f^{-1} \text{ is increasing.}$$

When $E(Y|A, \mathbf{X}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ is correctly specified, $E(Y|A, \mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}})$ is a consistent estimate of $E(Y|A, \mathbf{X})$, and $\Delta(\mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}})$ is a consistent estimate of $\Delta(\mathbf{X})$, and $\hat{g}^{opt}(\mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}})$ is a good approximation of $g^{opt}(\mathbf{X})$. However, when $E(Y|A, \mathbf{X}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ is not correctly specified, and the boundary of optimal ITR is not linear in (1,$\mathbf{X}$), i.e., the optimal ITR is not within the class taking the form: $\mathbb{1}[\mathbf{X}\boldsymbol{\gamma}_1 + \beta_1 > 0]$ or $\mathbb{1}[\mathbf{X}\boldsymbol{\gamma}_1 + \beta_1 < 0]$, $\hat{g}^{opt}(\mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}})$ may not be a good approximation for the optimal ITR. For example, suppose the decision boundary of optimal ITR based on 2 dimensional covariates is a circle in the covariates space: patients with covariates value inside the circle benefit from the treatment, and patients with covariates value outside the circle do not benefit from the treatment. In this case, a GLM including covariates as linear terms is misspecified. A GLM including covariates as linear terms would force the decision boundary to be a straight line, which will never be a good approximation of a circle; thus ITR depending on this parametric model will be far from optimal. Scenario II in simulation studies in Chapter 5 demonstrates this situation.

Several methods have been proposed to reduce the impact of misspecification of the mean model on ITR. When X is just a single covariate / 1-dimensional biomarker, one could estimate $E(Y|X, A = 1)$, $E(Y|X, A = 0)$ via kernel smoothing; hence, $\hat{\Delta}(X)$ is estimated nonparametrically. The optimal ITR can be approximated by $\mathbb{1}[\hat{\Delta}(X) < 0]$ (Matsouaka et al., 2014). In the case of time-to-event outcome, Zhou and Ma (2012), Ma and Zhou (2014) consider a varying-coefficient proportional hazard regression model:

$$\lambda(t|A, X) = \lambda_0(t)exp\{\beta(X)A + g(X)\},$$

where $\lambda(t)$ is the hazard function, and $\beta(X)$ is the coefficient of the treatment indicator which

is a function of X. $\beta(X)$ is estimated nonparametrically by local partial likelihood approach, and $\beta(X)$ can be used to select treatment like $\Delta(X)$ because under the proportional hazard assumption, $\mathbb{1}[\Delta(X) < 0] = \mathbb{1}[\beta(X) < 0]$ and $\mathbb{1}[\Delta(X) > 0] = \mathbb{1}[\beta(X) > 0]$. In the case of binary outcome, Han et al. (2015) consider a varying-coefficient model with the following logit link,

$$logit(\mu(X, A)) = \beta(X)A + g(X),$$

where $\mu(X, A) = P(Y = 1|X, A)$; Here, $\beta(X)$ is estimated using B-spline methods, and again $\beta(X)$ can be used to select treatment like $\Delta(X)$. For example, if we have a desirable outcome, such as treatment response, we should treat patients with $\beta(X) > 0$.

One advantage of studying a single biomarker is that we could plot $\hat{\Delta}(X)$ against X or percentile of X with confidence band around the curve, which is a useful way to evaluate a predictive biomarker graphically(Janes et al., 2014). Similarly, we can plot $\hat{\beta}(X)$ against X, which is termed the covariate-specific treatment effect curve (CTE) in time-to-event outcome scenario (Ma and Zhou, 2014; Zhou and Ma, 2012) or the conditional average treatment effect (CATE) curve in binary outcome scenario (Han et al., 2015).

In the presence of multiple biomarkers, we want to combine information from multiple biomarkers to guide treatment selection. The method that uses multivariate kernel smoothing to estimate the conditional mean model, $E(Y|\mathbf{X}, A)$, suffers from the "curse of dimensionality" (Matsouaka et al., 2014). Another general purpose nonparametric regression method, random forests, has also been used to estimate $E(Y|\mathbf{X}, A)$ (Taylor et al., 2015). In Chapter 5, we compare the performance of random forests that estimate $E(Y|\mathbf{X}, A)$ and derive ITR with our proposed approach (developed in Chapter 3) through simulation studies. Cai et al. (2011) proposed a two-step method to estimate a quantity similar to $\Delta(\mathbf{X})$, and Matsouaka et al. (2014) extended it to treatment selection. In the first step, similar to what we discussed in the beginning of this section, a parametric model is used to create an univariate score $S(\mathbf{X}) = \Delta(\mathbf{X}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}})$ for each observation. In the second step, a non-parametric method is used to estimate the average treatment difference condition-

ing on $S(\mathbf{X})$, $\hat{E}[Y(1) - Y(0)|S(\mathbf{X})]$, which is then used to select treatment. Note that $\hat{E}[Y(1) - Y(0)|S(\mathbf{X})]$ is not the same as the treatment difference conditioning on $\mathbf{X}$, which suggests that this approach depends on specification of $E[Y|\mathbf{X}, A, \boldsymbol{\beta}, \boldsymbol{\gamma}]$, and the ITR derived based on this approach can be suboptimal.

Qian and Murphy (2011) considered a richer mean model by expanding the basis functions $\mathbf{X}$ from p-dimensional baseline covariates space to higher dimensions. For example, supposing that we have $\mathbf{X} = (X_1, X_2)$, we could expand the basis functions to $\mathbf{X}_{expand} = (X_1, X_2, X_1X_2, X_1^2, X_2^2)$. The new model is: $E(Y|A, \mathbf{X}_{expand}, \boldsymbol{\beta}, \boldsymbol{\gamma}) = \beta_0 + \beta_1 A + \mathbf{X}_{expand}\boldsymbol{\gamma}_0 + A\mathbf{X}_{expand}\boldsymbol{\gamma}_1$. A linear decision boundary in the expanded covariates can approximate a non-linear decision boundary in the original space. However, as we are expanding the covariates space, it quickly becomes a high-dimensional problem. Qian and Murphy (2011) proposed to use L_1 penalized least squares ($L_1 - PLS$) method, also known as LASSO. We can rewrite the mean model: $E(Y|A, \mathbf{X}_{expand}, \boldsymbol{\beta}, \boldsymbol{\gamma}) = E(Y|\mathbf{X}^*, \boldsymbol{\beta}^*) = \mathbf{X}^*\boldsymbol{\beta}^*$, where $\mathbf{X}^* = (1, A, \mathbf{X}_{expand}, A\mathbf{X}_{expand})$ and $\boldsymbol{\beta}^* = (\beta_0, \beta_1, \boldsymbol{\gamma}_0, \boldsymbol{\gamma}_1)$. Using the LASSO to estimate $\boldsymbol{\beta}^*$:

$$\hat{\boldsymbol{\beta}}^* = argmin_{\boldsymbol{\beta}^*}\{\sum_{i=1}^{n}(y_i - \mathbf{x}_i^*\boldsymbol{\beta}^*)^2 + \lambda\sum_{j=1}^{p^*}\sigma_j|\beta_j^*|\},$$

where $\sigma_j = [1/n\sum_{i=1}^{n}(\mathbf{x}_i^*\boldsymbol{\beta}_j^*)]^{1/2}$ is the weight to balance the scale of different basis function, and λ is the LASSO penalty parameter, which can be selected by cross-validation. Then, $\hat{\Delta}(\mathbf{X}) = E(Y|A = 1, \mathbf{X}_{expand}, \hat{\boldsymbol{\beta}}^*) - E(Y|A = 0, \mathbf{X}_{expand}, \hat{\boldsymbol{\beta}}^*)$, and the ITR can be constructed based on $\hat{\Delta}(\mathbf{X})$.

Kang et al. (2014) proposed a boosting iterative algorithm to reduce the bias and misclassification caused by misspecification of the working mean model. In iteration m, a working mean model, $E(Y|A, \mathbf{X}, \boldsymbol{\beta}, \boldsymbol{\gamma})$, is fitted with more weights given to observations whose $\hat{\Delta}(X)^{(m-1)}$ estimated in the previous iteration is near zero. The final $\hat{\Delta}(X)$ is the average of $\hat{\Delta}(X)^{(m)}$ from all iterations. Authors argued that observations with $\hat{\Delta}(X)^{(m)}$ near 0 are more likely to be given wrong treatment recommendation and hence should be given

more weights. This method has been criticized by a group of discussants of the paper for its sensitivity to working model specification and lack of theoretical justification (Tian, 2014; Zhao and Kosorok, 2014)

All of above methods for treatment selection firstly estimate the conditional treatment difference, $\Delta(\mathbf{X})$, and then select treatment based on the estimated conditional treatment difference. These methods can be considered as indirect methods compared to the weighted classification framework that we describe in the next section.

2.3 Derive Optimal ITR Under Weighted Classification Framework

In this section, we review the weighted classification framework proposed by Rubin and van der Laan (2012) and Zhang et. al. (2012). Unlike the methods described in Section 2.2, the methods under this framework aim to directly solve the problem of optimization of population average outcome under the ITR, i.e. $E[Y(g)]$, instead of modeling $E(Y|X, A)$ and constructing ITR based on $\hat{\Delta}(X)$.

In Section 2.1, we define the optimal ITR as $g^{opt}(\mathbf{X}) = argmin_{g(\mathbf{X})}E[Y(g(\mathbf{X}))]$ when our treatment goal is to decrease a continuous outcome Y, such as blood pressure among patients with hypertension, or to reduce the probability of an undesired binary outcome, such as mortality rate; $g^{opt}(\mathbf{X}) = argmax_{g(\mathbf{X})}E[Y(g(\mathbf{X}))]$ when our treatment goal is to increase a continuous outcome Y, such as CD4 count in HIV infected patients, or to increase the probability of a desired binary outcome, such as treatment response rate. Without loss of generality, here we follow the derivation of Zhang et. al. (2012) to cast this optimization problem $g^{opt}(\mathbf{X}) = argmin_{g(\mathbf{X})}E[Y(g(\mathbf{X}))]$ as a weighted classification problem for situation where lower Y is desired.

Firstly, recall (2.1), we have $E[Y(g(\mathbf{X}))] = E[g(\mathbf{X})\Delta(\mathbf{X})] + E[Y(0)]$. Thus, $argmin_{g(\mathbf{X})}E[Y(g(\mathbf{X}))] = argmin_{g(\mathbf{X})}E[g(\mathbf{X})\Delta(\mathbf{X})]$. Now, we can rewrite $g(\mathbf{X})\Delta(\mathbf{X})$ as follows:

$$\begin{aligned} g(\mathbf{X})\Delta(\mathbf{X}) &= g(\mathbf{X})\mathbb{1}(\Delta(\mathbf{X}) > 0)|\Delta(\mathbf{X})| - g(\mathbf{X})\mathbb{1}(\Delta(\mathbf{X}) \leq 0)|\Delta(\mathbf{X})| \\ &= |\Delta(\mathbf{X})|[\{1 - g(\mathbf{X})\}\mathbb{1}(\Delta(\mathbf{X}) \leq 0) + g(\mathbf{X})\mathbb{1}(\Delta(\mathbf{X}) > 0)] - \mathbb{1}(\Delta(\mathbf{X}) \leq 0)|\Delta(\mathbf{X})|. \end{aligned}$$

Since the $\mathbb{1}(\Delta(\mathbf{X}) < 0)$ and $g(\mathbf{X})$ take value either 1 and 0,

$$g(\mathbf{X})\Delta(\mathbf{X}) = |\Delta(\mathbf{X})|[\mathbb{1}(\Delta(\mathbf{X}) < 0) - g(\mathbf{X})]^2 - \mathbb{1}(\Delta(\mathbf{X}) \leq 0)|\Delta(\mathbf{X})|.$$

Hence, $argmin_{g(\mathbf{X})}E[Y(g(\mathbf{X}))] = argmin_{g(\mathbf{X})}E(|\Delta(\mathbf{X})|[\mathbb{1}(\Delta(\mathbf{X}) < 0) - g(\mathbf{X})]^2)$. $E(|\Delta(\mathbf{X})|[\mathbb{1}(\Delta(\mathbf{X}) < 0) - g(\mathbf{X})]^2)$ can be viewed as the expected weighted misclassification error rate, where $|\Delta(\mathbf{X})|$ is the weight; $\mathbb{1}(\Delta(\mathbf{X}) < 0)$ is the binary class label, and $g(\mathbf{X})$ is the classifier.

In addition to above abstract mathematical derivation, the minimization of the weighted classification error rate has an intuitive interpretation. The class label is the optimal treatment rule, $g^{opt}(\mathbf{X}) = \mathbb{1}(\Delta(\mathbf{X}) < 0)$, as we have shown in Chapter 2. We would like to have an ITR $g(\mathbf{X})$ matched with the $g^{opt}(\mathbf{X})$. Hence, we would like to minimize the misclassification error rate. If a misclassication error occurs, should we give every misclassification error the same weight? This weighted classification framework says that the weight of this misclassification should be $|\Delta(\mathbf{X})|$. Suppose a subpopluation defined by covariates $\mathbf{x}^*$ is misclassified by $g(\mathbf{x}^*)$; i.e., $g(\mathbf{x}^*) = 1 - g^{opt}(\mathbf{x}^*) \neq g^{opt}(\mathbf{x}^*)$. This misclassification would increase $E[Y(g(\mathbf{X}))]$, the average outcome under the ITR for the whole population, by

$$|E[Y(g^{opt}(\mathbf{x}^*))|\mathbf{x}^*] - E[Y(1 - g^{opt}(\mathbf{x}^*))|\mathbf{x}^*]| = |E_Y[Y(1)|\mathbf{x}^*] - E_Y[Y(0)|\mathbf{x}^*]| = |\Delta(\mathbf{x}^*)|,$$

which is the absolute difference of the average outcomes under the optimal rule and the average outcome defying the optimal rule of this subpopulation. Hence, if our goal is to minimize $E[Y(g(\mathbf{X}))]$ with respect to $g(\mathbf{X})$, we should weight each misclassification error

with $|\Delta(\mathbf{X})|$.

To solve this minimization problem with the observed data, we approximate the objective function, an expectation, by its empirical counterpart, a sample average:

$$1/n \sum_{i=1}^{n} |\Delta(\mathbf{X}_i)|[\mathbb{1}(\Delta(\mathbf{X}_i) < 0) - g(\mathbf{X}_i)]^2.$$

In addition, we need to estimate the unobservable $\Delta(\mathbf{X}_i)$ by information from the sample. Zhang and others (2012) proposed two consistent estimators for $\Delta(\mathbf{X}_i)$ for individual i, namely the inverse probability weighted estimator (IPWE) and the augmented inverse probability weighted estimator (AIPWE). We describe these estimators in details below.

The IPWE of $\Delta(\mathbf{X}_i)$ is given as follows:

$$\hat{\Delta}(\mathbf{X}_i)_{IPWE} = \frac{A_i}{\pi_i} Y_i - \frac{1 - A_i}{1 - \pi_i} Y_i, \tag{2.3}$$

where $\pi_i = P(A_i = 1)$ is the probability of individual i being assigned to treatment group. In randomized clinical trials, π_i, also known as propensity score, is known by the study design, and $\hat{\Delta}(X_i)_{IPWE}$ is a consistent estimator. In observational studies, an extra effort is needed to estimated π_i based on subjects' covariates.

The AIPWE of $\Delta(\mathbf{X}_i)$ is given as follows:

$$\hat{\Delta}(\mathbf{X}_i)_{AIPWE} = \frac{A_i}{\pi_i} Y_i - \frac{1 - A_i}{1 - \pi_i} Y_i - \frac{A_i - \pi_i}{\pi_i} \hat{\mu}(\mathbf{X}_i, 1) - \frac{A_i - \pi_i}{1 - \pi_i} \hat{\mu}(\mathbf{X}_i, 0), \tag{2.4}$$

where $\hat{\mu}(\mathbf{X}_i, a) = \hat{E}[Y|\mathbf{X}_i, A = a]$, based on a working model for the outcome Y. Although the working model for the outcome could be misspecified, $\hat{\Delta}(\mathbf{X}_i)_{AIPWE}$ is again a consistent estimator because π_i is known in randomized clinical trials. In addition, compared to IPWE, AIPWE gains more precision by utilizing the information from baseline covariates.

In summary, under this framework, obtaining an optimal ITR from a randomized

clinical trial can be cast as a weighted classification problem:

$$\hat{g}^{opt}(\mathbf{X}) = argmin_g[1/n \sum_{i=1}^{n} |\hat{\Delta}(\mathbf{X}_i)|[\mathbb{1}(\hat{\Delta}(\mathbf{X}_i) < 0) - g(\mathbf{X}_i)]^2] \tag{2.5}$$

Interestingly, Zhao and other (2012) use a different derivation to reach the same result, i.e. casting optimization of $E[Y(g(\mathbf{X}))]$ as minimization of the sample averaged weighted misclassification error, where $\hat{\Delta}$ is estimated by IPWE (Zhang et al., 2012a). Authors call their method the 'outcome weighted learning (OWL)'.

The advantage of this weighted classification framework is that the ITR, $g(\mathbf{X})$, is no longer restricted by any assumption of the outcome model, $E[Y|X, \boldsymbol{\beta}, \boldsymbol{\gamma}, A]$, like approaches we discussed in Chapter 2. We can choose the class of $g(X)$ as a linear or nonlinear combination of all $\mathbf{X}$ or subset of $\mathbf{X}$ or place no restriction on the form of $g(\mathbf{X})$. Hence, this framework provides a direct way to deriving ITR, which is more flexible and robust than the methods discussed in Chapter 2.

After casting the goal of deriving an optimal ITR as a weighted classification problem, various well developed classification techniques can be applied. Some researchers tackled this problem by using optimization algorithms closely related to support vector machine (SVM). Using the language in optimization, (2.5) is a weighted sum of 0-1 loss, which is difficult to minimize. We could restrict $g(\mathbf{X})$ to the linear combination of $\mathbf{X}$, $\mathbf{X}\boldsymbol{\gamma}$, and replace 0-1 loss by a convex and continuous surrogate loss function, such as the hinge loss (Zhao et al., 2012) and exponential loss (Huang, 2015), or the difference of two convex loss functions, such as the ramp loss (Huang and Fong, 2014) and the smooth ramp loss (Huang, 2015). Various convex optimization procedures can be then deployed to minimize this objective function. However, by restricting $g(\mathbf{X})$ to the linear combination of $\mathbf{X}$, the decision boundary of ITR is restricted to linear boundary. To be more flexible in decision boundary, Zhao et al. (2012), Huang and Fong (2014) and Huang (2015) used non-linear kernels, such as radial basis function (RBF) kernel, to replace the linear kernel, which enables the algorithm

to identify a non-linear decision boundary for ITR.

Another family of methods for classification is tree-based methods. Zhang and others (2012) used classification and regression tree (CART) to minimize the objective function in (2.5). Tree-based models are easy to interpret, but they restrict the decision boundary to a certain class, and CART is known to be unstable and lack of smoothness (Hastie et al., 2009). Rubin and van der Laan (2012) used bagging trees to solve the minimization problem, which alleviates the limitation of CART. Another popular classification and regression method is random forests (Breiman, 2001), which is demonstrated to have greater predictive power in many settings when compared to bagging trees and SVM. In this thesis, to extend the method of bagging trees, we use random forests to minimize the objective function in (2.5). Methods like SVM with RBF kernel, bagging trees and random forests sometimes are called 'black box' models, which are difficult to interpret and provide little clinical insights on different biomarkers. Hence, motivated by the lack of interpretability of random forests, we also discuss the variable importance measures in the context of ITRs. The variable importance measures are also helpful in selecting predictive biomarkers among larger number of potential candidate biomarkers to ensure stability of the derived ITRs. Under the weighted classification framework, Huang (2015) proposed an algorithm to select predictive biomarkers when the decision boundary is restricted to linear combinations of biomarkers. The proposed variable importance measures quantify the relative relevance of the patient's characteristics to treatment selection without restriction on forms of combination of biomarkers.

Chapter 3

Random Forests Under The Weighted Classification Framework

Breiman and others popularized the tree-based models in their monograph, Classification And Regression Tree (CART) (Breiman et al., 1984). CART is a general model for regression and classification. Based on CART, Breiman proposed two new methods called bagging (bootstrap aggregation) trees (Breiman, 1996) and random forests (Breiman, 2001). Random forests have been one of the most popular off-the-shelf methods for classification and regression (Hastie et al., 2009). The excellent performance of random forests is at the cost of interpretability. Breiman et. al. proposed two types of variable importance measures in tree-based models and random forests, which could help to gain insight of relative importance of each covariates on outcome (Breiman et al., 1984; Breiman, 2001). Here, we apply the idea of random forests to solve the weighted classification problem as stated in (2.5). In addition, we develop and compare two types of variable importance measures in the context of deriving the optimal ITR. Next, we first describe a single tree classifier, the building block of random forests, in the context of deriving the optimal ITR.

3.1 A Single Decision Tree

Recall that under the weighted classification framework discussed in the previous chapter, we have:

$$\hat{g}^{opt}(\mathbf{X}) = argmin_g[1/n \sum_{i=1}^{n} |\hat{\Delta}(\mathbf{X}_i)|[\mathbb{1}(\hat{\Delta}(\mathbf{X}_i) < 0) - g(\mathbf{X}_i)]^2].$$

Let us denote $\mathbb{1}(\hat{\Delta}(\mathbf{X}_i) < 0)$ be Z_i, the estimated binary class label of subject i. Let $|\hat{\Delta}(\mathbf{X}_i)|$ be W_i, the estimated weight for misclassification error of subject i. Recall that $\hat{\Delta}(\mathbf{X}_i)$ can be estimated by IPWE or AIPWE. Assume that subjects' baseline covariates $\mathbf{X} \in R^p$. The problem is to find a classifier $g(\mathbf{X}) : R^p \to \{0, 1\}$, such that the weighted misclassification error is minimized.

A tree based classifier, $T(\mathbf{X})$ can be written as follows:

$$g(\mathbf{X}) = T(\mathbf{X}) = \sum_{k=1}^{K} c_k \mathbb{1}(\mathbf{X} \in R_k),$$

where R_k's are the mutually exclusive and collectively exhaustive regions, which partition the covariates space R^p. Here, $c_k \in \{0, 1\}$ is the predicted class label for observations in the region R_k. In other words, patients in the same region defined by their covariates $\mathbf{X}$ is recommended to the same treatment rule, $g(\mathbf{X}) = c_k \mathbb{1}(\mathbf{X} \in R_k)$. A weighted version of CART method is used to construct R_k and c_k based on the covariates $\mathbf{X}$, the binary label Z and the weight W through a recursive binary splitting algorithm as described in the following paragraphs.

At start, all the observations $(\mathbf{x}_i, w_i, z_i)$, where $i = 1, ..., n$, are in the root node, N_0. In the tree terminology, a node is also a region of covariates space. The root node is just the entire covariates space, R^p. We can define the weighted misclassification error for

the root node Q_0 as follows:

$$Q_0(h) = \sum_{i=1}^{n} (|w_i|[z_i - h]^2 \mathbb{1}(\mathbf{x}_i \in N_0)),$$

and

$$Q_0 = min(Q_0(h = 1), Q_0(h = 0)).$$

In the decision tree terminology, Q_0 is called impurity measure (of the root node). For tree-based models, the impurity measure is quantity that measures the heterogeneity in each node and guides the construction of trees. The recursive binary splitting algorithm aims to minimize the impurity measure.

We can split the root node N_0 into two children nodes N_1 and N_2 by j^{th} covariate and cutoff point s: $N_1^{(j,s)} = \{\mathbf{x}|x_j \leq s\}$ and $N_2^{(j,s)} = \{\mathbf{x}|x_j > s\}$. The impurity measures of $N_1^{(j,s)}$ and $N_2^{(j,s)}$ are denoted by $Q_1^{(j,s)}$ and $Q_2^{(j,s)}$ respectively. To achieve the goal of reducing the weighted misclassification error, we want to choose $(j, s) = argmax_{j \in p, s \in R}[Q_0 - Q_1^{(j,s)} - Q_2^{(j,s)}]$. That is to choose (j, s) to achieve maximum decrease in the weighted misclassification error. Computationally, we consider all the covariates and find a best cutoff point that maximizes decrease in the weighted misclassification error for each covariate; then from p candidates, we choose a covariate with its best cutoff point that maximizes decrease in the weighted misclassification error. Henceforth, we drop the superscript (j, s) from regions and impurity measure notations. Let's denote $\delta_{l=1}(Q)^j = (Q_0 - Q_1 - Q_2)$ to be the decrease of weighted misclassification error at the first split ($l = 1$), which is split at j^{th} covariate. Set $\delta_l(Q)^j = 0$ if the l^{th} split is not split by X_j. This quantity will be used to construct one of variable importance measures, which is described in Section 3.4. After the first split, N_0 became an internal node, and N_1 and N_2 became the terminal nodes, which are subject to the next split. The splitting process is recursively repeated to the terminal nodes and produces a tree like structure model. Ultimately, we have a big tree with each terminal node only containing 1 observation. If we use these terminal nodes as regions for a tree model, it is

likely that we will have an overfitting issue. The recommended practice to avoid overfitting in tree model is to grow a big tree first and then use some pruning methods to collapse the terminal nodes, which results in a smaller and less complex tree (Hastie et al., 2009). However, overfitting of individual trees is not a problem in random forests (Breiman, 2001). In a random forest, we grow a complex tree without pruning. Therefore, we skip discussion of pruning details. The final terminal nodes are the regions, R_k. And c_k is the label that minimizes the weighted misclassification in each region. If there is only one observation in a region, c_k will be the observed class label.

Suppose some of covariates are unordered categorical variables. For each categorical variable with finite levels, there are finite possible combinations to split the sample into two nodes. The splitting procedure for the categorical variable is to find the best combination which gives the maximum decrease in weighted misclassification error.

A single tree model has several drawbacks. Because of the binary splitting, the decision boundary is restricted to boundary of some unions of rectangle regions. Because of the hierarchical structure of tree, the earlier splits would impact the later splits, which make the tree model unstable. Few outliers could influence an earlier splits, affect latter splits and change the whole structure of a tree (Hastie et al., 2009). In addition, pruning a tree requires extra effort to choose a tuning parameter. Those drawbacks lead to moderate classification performance for a single tree. In the next section, we describe the method of random forests in deriving optimal ITR under the weighted classification framework, which overcomes these drawbacks of a single tree model.

3.2 Weighted Random Forests For Estimating ITR

The method of random forests is an extension of bagging trees. 'Bootstraping aggregation' (bagging) is a method to minimize the instability of a predictive model, such as CART

(Breiman, 1996). Here, we first describe the bagging trees.

We have a training dataset $D = \{(\mathbf{x}_1, y_1, a_1), ..., (\mathbf{x}_n, y_n, a_n)\}$ of size n from a clinical trial. The first step is bootstrapping the training data D B times, which gives bootstrapped samples D_b, where $b = 1, ..., B$. For each D_b, the binary label z_i and the weight w_i are estimated by IPWE or AIPWE. Then, we can use CART to build a single tree model without pruning, T_b, based on each bootstrapped sample as described in Section 3.1. The final model is a committee of T_b. For new data with covariates $\tilde{\mathbf{x}}$, the predicted value, a binary classification, from the bagging trees is the majority votes from the committee of trees, denoted as $\{T_b(\tilde{\mathbf{x}})\}_1^B$. By bootstrapping sample and aggregating results through the majority voting from the committee of trees, the stability and prediction accuracy of a model is improved. Also, overfitting is not an issue as the aggregated prediction is not made from a single tree (Breiman, 1996). In addition, the decision boundary is not limited to boundaries of unions of rectangle regions which could be produced by a single tree.

Since every individual tree is built on a bootstrapped sample of the same training data, there are correlations between trees. Breiman (2001) proposed the random forests, which randomly selects a subset of covariates as candidate covariates for each split. Through this random selection of covariates, the correlation between trees is reduced. Same as bagging trees, for new data with covariates $\tilde{\mathbf{x}}$, the predicted value is the majority votes from the committee of trees, denoted as $RF(\tilde{\mathbf{x}}) = \{T_b(\tilde{\mathbf{x}})\}_1^B$. We propose to use this tree-based ensemble algorithm with individual weights estimated from each boostrapped sample and random selection of candidate covariates at each split to estimate the ITR. We term the algorithm Weighted Random Forest (W-RF).

In general, random forests have better performance than the bagging trees because of smaller correlation between individual trees in random forests. Hastie et al. (2009) suggested we should treat the number of covariates randomly sampled as candidates at each split, often denoted as $mtry$, as a tuning parameter. Notice that, in our algorithm,

the individual weights and labels are estimated using each bootstrapped sample instead of all training data; hence, the weight or label for an individual subject could be slightly different, which can further reduce the correlation between trees. Based on results of our numerical experiments, this strategy has moderate improvement in performance compared to estimating individual weights or labels from the whole sample before the bootstrap procedure.

3.3 Evaluation Of The Derived Rule

In the previous sections, we propose the weighted random forests (W-RF) method to derive an optimal ITR based on data from clinical trials. Before the derived ITR is used in practice, we need to evaluate performance of the derived ITR and compare it to ITRs derived from other methods. There are two performance measures for the evaluation of derived ITRs. The first performance measure is the misclassification error rate (MCR) of an ITR, $Pr(\mathbb{1}[g^{opt}(\mathbf{X}) \neq \hat{g}^{opt}(\mathbf{X})])$. However, only in the simulation studies where $g^{opt}(\mathbf{X})$ is known, MCR can be estimated by the empirical proportion of disagreement between the optimal ITR and the derived ITR in the population. Hence, it can not be used as a performance measure in real data analyses. The second performance measure is the population average outcome under the derived ITR, $E[Y(\hat{g}^{opt})]$, as the optimal ITR is defined as a rule optimizing $E[Y(g)]$. In the following paragraphs, we discuss the possible ways to estimate the population average outcome under the derived rule.

A naive approach is to estimate the population average outcome under the rule using the same data that is used to derive the rule, which is often termed 'naive estimate'. In essence, deriving an optimal ITR is a prediction problem, which is to predict whether a patient with specific characteristics would benefit from a treatment. In all prediction problems, statistical models derived from optimization procedures on training data, including maximum likelihood estimation for the conditional mean model and direct optimization procedures including our method, have the risk of overfitting. Because $E[Y(g)]$ is also the

quantity we use to evaluate the ITR, the direct optimization approach could be even more likely to overfit the training data than indirect methods that depend on $\hat{\Delta}(\mathbf{X})$. For future patients with covariates $\tilde{\mathbf{X}}$, $\hat{g}^{opt}(\tilde{\mathbf{X}})$ might be far from maximizing $E[Y(g(\tilde{\mathbf{X}}))]$. Hence, in the existing literature, in order to evaluate performance of an estimated ITR $\hat{g}(\mathbf{X})$ in simulation studies, a large testing dataset is typically generated to estimate $\hat{E}[Y(\hat{g}(\mathbf{X}))]$ (Huang and Fong, 2014; Huang, 2015; Matsouaka et al., 2014; Zhao et al., 2012). Suppose we derived a treatment rule $\hat{g}(.)$ from the training data, the empirical estimate of $E[Y(\hat{g}(.))]$ using the testing data $(\tilde{\mathbf{X}}_i, \tilde{A}_i, \tilde{Y}_i)$, i from 1 to n^*, is as follows:

$$\begin{aligned}
\hat{E}[\tilde{Y}(\hat{g}(\tilde{\mathbf{X}}))] =& \hat{E}[\tilde{Y}(1) * \hat{g}(\tilde{\mathbf{X}}) + \tilde{Y}(0) * (1 - \hat{g}(\tilde{\mathbf{X}}))] \\
=& \hat{E}[\tilde{Y} * \hat{g}(\tilde{\mathbf{X}}) | \tilde{A} = 1] + \hat{E}[\tilde{Y} * (1 - \hat{g}(\tilde{\mathbf{X}})) | \tilde{A} = 0] \\
=& \frac{\hat{E}[\tilde{Y} * \hat{g}(\tilde{\mathbf{X}}) * \tilde{A}]}{\hat{E}[\tilde{A}]} + \frac{\hat{E}[\tilde{Y} * (1 - \hat{g}(\tilde{\mathbf{X}})) * (1 - \tilde{A})]}{\hat{E}[1 - \tilde{A}]} \\
=& \frac{\sum_i^{n^*} \mathbb{1}[\hat{g}(\tilde{\mathbf{X}}_i) = \tilde{A}_i = 1]\tilde{Y}_i}{\sum_i^{n^*} \mathbb{1}[\tilde{A}_i = 1]} + \frac{\sum_i^{n^*} \mathbb{1}[\hat{g}(\tilde{\mathbf{X}}_i) = \tilde{A}_i = 0]\tilde{Y}_i}{\sum_i^{n^*} \mathbb{1}[\tilde{A}_i = 0]}.
\end{aligned} \tag{3.1}$$

However, in analyses of real data, there is no large testing data available in most cases. The most common method adopted in predictive modeling practice is K-fold cross-validation. The original dataset is randomly split into K equal-sized folds; $K-1$ folds are used as a training dataset to derive ITR, and the rest of one fold are used as a testing dataset to estimate $E[Y(\hat{g}(\mathbf{X}))]$; the procedure is repeated K times, which allows every fold of the original dataset being used as a testing dataset; the average of K $\hat{E}[Y(\hat{g}(\mathbf{X}))]$ is then calculated. One can further repeat the random split process multiple times and take the average of the estimates. Because only $(K-1)/K$ of the original dataset is used to build the model, the model performance is underestimated especially when K is small; on the other hand, a large K, such as leave-one-out cross-validation, can have large variance and also be computationally intractable. Hence, it is recommended to use 5-fold or 10-fold cross-validation (Hastie et al., 2009). Five or 10-fold cross-validation has been used in the data analysis example of evaluating a derived ITR in the existing literature (Huang and Fong,

2014; Huang, 2015; Matsouaka et al., 2014; Zhao et al., 2012). For our proposed algorithm, we consider an alternative and novel method for estimation of population average outcome under the derived ITR, which is described in the following paragraph.

In the classic context of classification, out-of-bag error estimates from random forests have been used to replace the K-fold cross-validation (Breiman, 2001; Hastie et al., 2009). Here, we introduce the out-of-bag (OOB) estimate of $E[Y(\hat{g}(\mathbf{X}))]$ with our proposed method. Recall that the bootstrap procedure resamples the original training dataset, D, with replacement to obtain a bootstrapped sample D_b. By chance, for each bootstrap, some of observations in the original training dataset are not sampled, which are termed OOB sample, $D_b^{oob} = D \setminus D_b$. D_b^{oob} is not used to build b^{th} tree, T_b, hence D_b^{oob} naturally forms a testing dataset for T_b. The prediction result of T_b on OOB sample, $T_b(D_b^{oob})$, is an OOB prediction for T_b. We aggregate $T_b(D_b^{oob})$ and use the majority vote rule to construct OOB estimated ITR of the original training dataset for the random forests model. Let's denoted it as $RF^{oob}(\mathbf{X})$. Similar to the problem of prediction error estimate in the standard classification context, we can use $\hat{g}(\mathbf{X})^{oob} = RF^{oob}(\mathbf{X})$ and the original training dataset $(\mathbf{X}, A, Y)$ to estimate $\hat{E}[Y(\hat{g}(\mathbf{X}))]^{oob}$ using (3.1) to evaluate the performance of $\hat{g}(\mathbf{X})$. In Chapter 4, we numerically assess the quality of $\hat{E}[Y(\hat{g}(\mathbf{X}))]^{oob}$ by comparing it to the estimate from the large testing set, 5-fold CV estimate and naive estimate in simulation studies.

3.4 Variable Importance

Although random forests have better prediction performance than a single decision tree, the aggregation of trees loses a simple interpretation. The variable importance measures in random forests are helpful in exploratory analysis to rank the covariates according to their predictiveness (Hastie et al., 2009). Here, we extend the concept of variable importance measures in the context of deriving optimal ITRs. Variable importance measures for each covariate can help us select the predictive markers associated with treatment decision from

multiple candidate markers in an exploratory analysis. We propose two different variable importance measures for p covariates, which are described in the following paragraphs.

The first type of variable importance measure for covariates, denoted as $VI_1(X_j)$, is sum of decreases in weighted misclassification error for all splits attributed to X_j, averaged over all trees. To elaborate this definition, recall the decrease of weighted classification error $\delta_l(Q)^j$ described in Section 3.1. Suppose there are L splits in tree b; the first type of variable importance measure of X_j in tree b is defined as $VI_1(X_j)_b = \sum_l^L \delta_l(Q)^j$. Taking the average over B trees, we have $VI_1(X_j) = \frac{1}{B}\sum_{b=1}^B VI_1(X_j)_b$. The higher value of $VI_1(X_j)$ suggests X_j is more relevant to minimize the weighted classification error in (2.5) hence more relevant to treatment selection.

The second type of variable importance measures is related to the concept of out-of-bag (OOB) estimation of $E[Y(\hat{g}(\mathbf{X}))]$, which is described in Section 3.3. For b^{th} OOB sample, D_b^{oob}, we can permute j^{th} covariates, x_j, which gives $D_b^{oob}(\bar{j})$. We repeat this permutation procedure for all OOB samples, and an OOB prediction for a random forest with x_j permutated, denoted as $RF^{oob}(\mathbf{x}^{(\bar{j})})$, is obtained by majority votes from all OOB samples. The second type of variable importance measure, $VI_2(X_j)$, is defined as $\hat{E}[Y(\hat{g}(\mathbf{X}^{(\bar{j})}))]^{oob}$, which is the OOB estimate of $E[Y(\hat{g})]$ with X_j permuted. If we want to minimize $E[Y(g)]$, permuting a variable relevant to the treatment selection will increase $E[Y(g)]$. Hence, the higher $VI_2(X_j)$ suggests greater importance of X_j in minimizing $E[Y(g)]$. Similarly, when our goal is to maximize $E[Y(g)]$, the lower $VI_2(X_j)$ suggests greater importance of X_j in maximizing $E[Y(g)]$.

Chapter 4

Simulation Study

In this chapter, we conduct simulation studies to numerically evaluate the performance of the proposed method: random forests under the Weighted Classification Framework (W-RF). The model performance of W-RF are compared with some existing methods in three scenarios, which represent different types of decision boundaries. Under these scenarios, the performance of the out-of-bag estimation of the population average outcome under the derived rule and variable importance measures are also evaluated.

4.1 Simulation setup

We consider data generated from clinical trials with sample size $n = 100$ and 1:1 randomization with treatment assignment $A \sim Bern(0.5)$. We consider a continuous clinical outcome, Y, of which higher value is desired. Therefore $g^{opt}(\mathbf{X}) = argmax_{g(\mathbf{X})} E[Y(g(\mathbf{X}))]$. There are 3 scenarios in our simulation study. In all three scenarios, there are 2 predictive biomarkers, (X_1, X_2), and 5 prognostic biomarkers, $(X_3, X_4, X_5, X_6, X_7)$. The outcome Y is generated by

$$Y = C(\mathbf{X}) + \delta(\mathbf{X}) \times A + \epsilon.$$

$C(\mathbf{X})$ is the prognostic effect of biomarkers $\mathbf{X}$, and $\delta(\mathbf{X})\times A = \delta(X_1, X_2)\times A$ is the interaction term between predictive biomarkers and treatment. $\epsilon \sim N(0,1)$ is the error term. All three scenarios share the same $C(\mathbf{X})$,

$$C(\mathbf{X}) = 2 + X_1 + X_1^2 + X_2 + X_2^2 + X_3 + log(|X_4|) + 0.1e^{X_5} + sin(X_6) + cos(X_7).$$

Here, $C(\mathbf{X})$ represents a complicated generative model, which is hard to correctly specify. For scenario I, we have $\delta(\mathbf{X}) = 3(X_1 + X_2)$, which represents a linear optimal decision boundary. For scenario II, we have $\delta(\mathbf{X}) = 9 - 4X_1^2 - 4X_2^2$, which represents an optimal decision boundary made of a circle. For scenario III, we have $\delta(\mathbf{X}) = 5[2\mathbb{1}(X_1 > -0.5\ \&\ X_2 > -0.5) - 1]$, which represents a decision boundary that can be represented as 'and' / 'or' rule based on $\mathbf{X}$. Figure 4.1 shows the optimal decision boundaries and best linear boundaries that approximate the corresponding optimal decision boundaries. The plots are overlaid with 5000 simulated observations from these three scenarios.

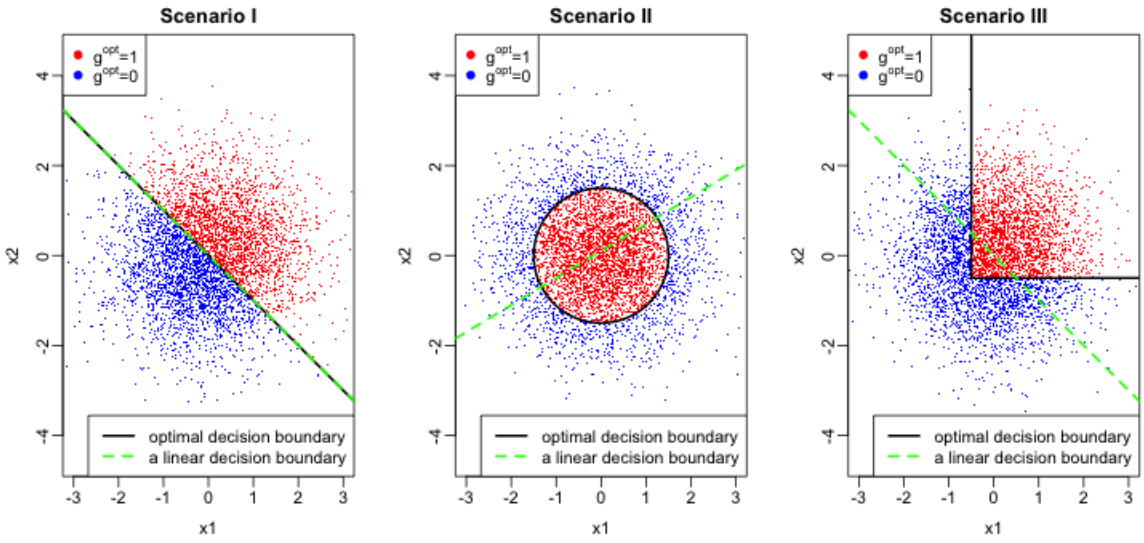

Figure 4.1: Three different simulation scenarios

Each scenario has two subsettings. For subsetting A, in addition to 7 covariates

associated with outcomes, there are 8 additional covariates providing no information on the outcome. Hence, the dimension of $\mathbf{X}$, p, is 15 for subsetting A. For subetting B, there are 43 additional irrelevant covariates ($p = 50$), which represent a relative high dimensions of covariates.

Predictive biomarkers (X_1, X_2) are generated from bivariate normal distribution with mean 0, variance 1 each and correlation 0.2. Other covariates X_j, where $j = 3, ..., p$, is generated independently from a standard normal distribution. In this study, there are 500 simulated Monte Carlo datasets for each subsetting of every scenario.

We compare W-RF with the following existing methods:

- The $L_1 - PLS$ method (PLS) (Qian and Murphy, 2011)
- Random forests method that directly estimates $E(Y|\mathbf{X}, A)$ and select treatment based on $\hat{\Delta}(\mathbf{X})$ (Direct-RF)
- The robust kernel methods proposed by Huang and Fong (2014) (HF).
- Weighted logistic regression with L1-norm penalty (Huang, 2015) ($W - logistic - l_1$)

PLS and Direct-RF first estimate $E(Y|\mathbf{X}, A)$ and then select treatment based on $\hat{\Delta}(\mathbf{X})$; i.e., $\hat{g}(\mathbf{X}) = \mathbb{1}(\hat{\Delta}(\mathbf{X}) > 0)$. In this simulation, PLS uses $(1, \mathbf{X}, A, \mathbf{X}A)$ as the basis function set, and the LASSO procedure is implemented with the tuning parameter selected by *cv.glmnet* function with default settings from *glmnet* R package. Hence, the decision boundary of PLS is linear in $\mathbf{X}$. In this simulation, Direct-RF uses $(\mathbf{X}, A)$ as input covariates. It is implemented by *randomForest* function from *randomForest* R package with default settings except that the number of covariates randomly sampled as candidates at each split, *mtry*, is selected among $([(p + 1)/3], [2(p + 1)/3], p + 1)$ based on OOB estimates of mean square error of $\hat{Y}$, where p is the dimension of $\mathbf{X}$, and [.] is the function rounding a real number to the nearest integer.

HF is one of the existing methods under the weighted classification framework as we mentioned in Chapter 3. HF is implemented by the R code which can be found in the supplemental material of Huang and Fong (2014). When a linear combination of $\mathbf{X}$ is considered (HF-linear), there is one tuning parameter, λ, and when the non-linear kernel (the radial basis function) is considered (HF-RBF), there are two tuning parameters, λ and γ. The tuning parameters are selected by 5-fold cross-validation.

In addition to using HF-linear, HF-RBF and W-RF to solve the weighted classification problem with potentially many irrelevant covariates, we also included weighted logistic regression with L1-norm penalty ($W-logistic-l_1$) in the context of deriving ITRs (Huang, 2015). It is implemented with $(1, \mathbf{X})$ as the design matrix, estimated individual weights and the tuning parameter selected by *cv.glmnet* function.

W-RF is implemented by our R functions, which can be found in the Appendix A. These functions have been tested against the $randomForest$ function in an usual unweighted classification setting. Similar to Direct-RF, $mtry$ is selected among $([p/3], [2p/3], p)$ based on OOB estimates of $E[Y(g(\mathbf{X}))]$. The total number of bootstrapped trees is set to 500, which is the same as the number in the default setting of $randomForest$ function.

HF-linear, HF-RBF, $W-logistic-l_1$ and W-RF require estimation of the contrast function, $\Delta(\mathbf{X}_i)$, for each observation. We use IPWE (2.3) or AIPWE (2.4) as described in Chapter 3. In this simulation, the conditional mean function, $\hat{\mu}(X_i, a) = \hat{E}[Y|X_i, A = a]$, is estimated from Direct-RF to construct AIPWE.

To compare the $\hat{g}(\mathbf{X})$ derived from different methods, a large testing dataset of $N = 100,000$, $(\tilde{\mathbf{X}}_i, \tilde{A}_i, \tilde{Y}_i)$, is generated. $E[\tilde{Y}(\hat{g}(\tilde{\mathbf{X}}))]$ is estimated from the testing dataset using the estimator described in (3.1). Since we know the true data generating mechanism, $g^{opt}(\tilde{\mathbf{X}}_i) = \mathbb{1}(\Delta(\tilde{\mathbf{X}}_i) > 0)$ is known for each individual in the testing dataset, and $E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))]$ thus can be estimated using (3.1) as well. We can calculate the bias of the population average outcome under the estimated ITR compared to the population average

outcome under the true optimal ITR; that is, $E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] - E[\tilde{Y}(\hat{g}(\tilde{\mathbf{X}}))]$. Smaller biases indicate the better performance of methods in deriving the optimal ITR. Since $g^{opt}(\tilde{\mathbf{X}}_i)$ is known, we can also estimate the misclassification error rate (MCR) of the estimated ITR, $Pr(\mathbb{1}[g^{opt}(\tilde{\mathbf{X}}_i) \neq \hat{g}(\tilde{\mathbf{X}}_i)])$, by the empirical proportion of disagreement between the optimal rule and estimated ITR among the observations in the testing dataset. Since the above estimations of average population outcome and MCR under a rule are based on a large testing dataset, we consider them as fixed parameters of a treatment rule. That is, 'hat' notation is not used for these estimates based on the large datasets.

Note that in a real data analysis, $g^{opt}(\mathbf{X}_i)$ is not known, and thus MCR cannot be computed to evaluate the performance of a ITR. In contrast, $E[Y(\hat{g}(\mathbf{X})]$ can still be estimated to evaluate the performance of an ITR. In Section 3.3, we discussed the K-fold cross-validation (CV) estimates and OOB estimates for $E[Y(\hat{g}(\mathbf{X})]$. In this simulation, we also compare $\hat{E}[Y(\hat{g}(\mathbf{X}))]$ from the large testing dataset to OOB estimates, naive estimates and CV estimates to numerically assess the usage of OOB estimates in the context of using W-RF to derive ITR.

In this simulation study, we also assess the variable importance measures (VIs) defined in Section 3.4. Recall that we discussed 2 types of VIs, which are numeric value for each biomarker. The ranking of VI of each biomarker provides the relative relevance of the biomarker to treatment selection. In our simulation setting, $VI_1(X_1)$ and $VI_1(X_2)$ should have higher rankings than all other $VI_1(X_j)$. Among $VI_2(X_j)$, $VI_2(X_1)$ and $VI_2(X_2)$ should have lowest rankings since larger value of Y is desired. W-RF results in a set of VIs based on every Monte Carlo dataset with sample size of 100. In addition to side-by-side boxplots of VIs to summarize the result of VIs from 500 Monte Carlo datasets, we use a summary statistic for ranking of VIs. Let us denote $U(VI)$ as follows:

$$U(VI_1) = Pr[VI_1(X_j) > VI_1(X_{j'})] \text{ for } j \in \{1,2\} \text{ and } j' \in \{3,...,p\},$$
$$U(VI_2) = Pr[VI_2(X_j) < VI_2(X_{j'})] \text{ for } j \in \{1,2\} \text{ and } j' \in \{3,...,p\}.$$

$U(VI_k)$ can be estimated by sample proportion. If $U(VI_k)$ is near 1, it suggests that VI_k performs well in selecting informative biomarkers among multiple irrelevant baseline covariates. $U(VI_k)$ can be thought as the area under the receiver operating characteristic (ROC) curve, where X_1, X_2 are 'cases', and the rest of $\mathbf{X}$ are 'controls' like in diagnostic setting. A similar ROC curve analysis has been used to summarize the ranking of variable importance measures in a tree-based model in a previous study (Huang and Wang, 2012).

4.2 Results

4.2.1 Model Performance Comparisons

The results of the bias (i.e., $E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] - E[\tilde{Y}(\hat{g}(\tilde{\mathbf{X}}))]$) and MCR of Scenario I, II and III are summarized in Table 4.1, Table 4.2 and Table 4.3 respectively. Smaller values of the bias and MCR indicate better performance of the models. Firstly, we notice that for all three methods under the weighted classification framework, results from methods using AIPWE for $\Delta(\mathbf{X}_i)$ are uniformly better than results from methods using IPWE for $\Delta(\mathbf{X}_i)$. This observation that IPWE is less efficient and stable agrees with previous results (Zhang et al., 2012a). Hereafter, the results from the methods under the weighted classification framework refer to results using AIPWE for $\Delta(\mathbf{X}_i)$. In general, the performance of the models from subsetting B of each scenario is worse than that in subsetting A, the subsetting with fewer noisy covariates.

In Scenario I (I-A and I-B), where the decision boundary is linear in X_1 and X_2, PLS has the best performance with respect to minimization of the bias and MCR. This is because that the optimal ITR is within the same class of PLS, and PLS has the variable selection features. $W - logistic - l_1$ has the second best performance in Scenario I-A and I-B for the same reasons. Direct-RF has the third and the fourth best performance in Scenario I-A and I-B. The comparison between PLS and Direct-RF in Scenario I suggests that flexibility of

Direct-RF is at cost of performance when the optimal decision boundary is coincident with the decision boundary of PLS. In Scenario I, except for $W-logistic-l_1$, the methods under the weighted classification framework have moderate performance with MCR around 0.20. In scenarios I-A, W-RF does not outperform the other methods. However, as we increase the number of irrelevant covariates in Scenario I-B, our proposed method, W-RF, suffers less from the 'curse of dimensionality' and outperforms HF-Linear, HF-RBF and Direct-RF. Similar pattern is observed in Scenario III, where Direct-RF and W-RF have almost the same performance in lower dimensional subsetting (Scenario III-A) but W-RF outperforms Direct-RF in higher dimensional subsetting (Scenario III-B).

The decision boundary of the optimal ITR in Scenario II is a circle, which cannot be approximated well by a linear decision boundary (see Figure 4.1). Therefore, PLS has worst performance (MCR is about 0.65), and Direct-RF has reasonable performance due to its flexibility. For the same reason, HF-RBF performs better than HF-linear and $W-logistic-l_1$. In this scenario, W-RF outperforms every other method by a large margin both in low and high dimension subsettings.

W-RF also has best performance in Scenario III-A and Scenario III-B. Although the decision boundary of the optimal ITR in Scenario III is not linear in X_1 and X_2, it can be approximated by a linear boundary to some extent (see Figure 4.1). Therefore, PLS has reasonable performance, which is not affected by the increasing dimensionality because of its variable selection feature.

Scenario I								
	Subsetting A (p=15)				Subsetting B (p=50)			
	Bias	SE	MCR	SE	Bias	SE	MCR	SE
PLS	0.04	0.005	0.05	0.002	0.04	0.005	0.05	0.002
Direct-RF	0.34	0.009	0.20	0.003	0.56	0.015	0.26	0.004
				AIPWE for $\Delta(\mathbf{X})$				
W-RF	0.46	0.011	0.20	0.003	0.52	0.013	0.23	0.003
HF-RBF	0.41	0.015	0.21	0.004	0.78	0.015	0.30	0.003
HF-linear	0.35	0.013	0.19	0.003	0.78	0.012	0.30	0.003
$W - logistic - l_1$	0.18	0.015	0.12	0.004	0.28	0.018	0.15	0.005
				IPWE $\Delta(\mathbf{X})$				
W-RF	1.23	0.012	0.38	0.003	1.37	0.015	0.41	0.003
HF-RBF	0.88	0.018	0.32	0.004	1.21	0.014	0.39	0.002
HF-linear	0.98	0.019	0.34	0.004	1.25	0.011	0.39	0.002
$W - logistic - l_1$	1.12	0.030	0.36	0.006	1.24	0.029	0.38	0.006

Table 4.1: Results for Scenario I summarized from 500 simulated datasets. The sample mean and standard error of the bias ($E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] - E[\tilde{Y}(\hat{g}(\tilde{\mathbf{X}}))]$) and misclassification error rate (MCR) are shown. $E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] = 5.99$; $E[Y(1)] = \hat{E}[Y(0)] = 4.13$.

Scenario II								
	Subsetting A (p=15)				Subsetting B (p=50)			
	Bias	SE	MCR	SE	Bias	SE	MCR	SE
PLS	3.53	0.013	0.65	0.004	3.54	0.010	0.65	0.003
Direct-RF	1.71	0.019	0.25	0.002	2.35	0.014	0.31	0.001
				AIPWE for $\Delta(\mathbf{X})$				
W-RF	0.47	0.012	0.13	0.002	0.83	0.016	0.17	0.002
HF-RBF	1.93	0.015	0.30	0.001	2.66	0.009	0.36	0.003
HF-linear	2.40	0.017	0.34	0.003	2.80	0.009	0.40	0.003
$W - logistic - l_1$	2.64	0.015	0.34	0.004	2.71	0.014	0.36	0.004
				IPWE $\Delta(\mathbf{X})$				
W-RF	2.17	0.022	0.43	0.004	2.62	0.023	0.49	0.004
HF-RBF	2.39	0.014	0.37	0.003	2.82	0.012	0.41	0.004
HF-linear	2.77	0.018	0.41	0.004	2.98	0.012	0.45	0.004
$W - logistic - l_1$	2.86	0.018	0.41	0.006	2.89	0.018	0.42	0.006

Table 4.2: Results for Scenario II summarized from 500 simulated datasets. The sample mean and standard error of the bias ($E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] - E[\tilde{Y}(\hat{g}(\tilde{\mathbf{X}}))]$) and misclassification error rate (MCR) are shown. $E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] = 7.77$; $E[Y(1)] = 5.13$; $E[Y(0)] = 4.14$.

Scenario III								
	Subsetting A (p=15)				Subsetting B (p=50)			
	Bias	SE	MCR	SE	Bias	SE	MCR	SE
PLS	0.88	0.009	0.17	0.002	0.90	0.009	0.18	0.002
Direct-RF	0.68	0.017	0.13	0.003	0.93	0.022	0.18	0.004
				AIPWE for $\Delta(\mathbf{X})$				
W-RF	0.66	0.016	0.13	0.003	0.85	0.023	0.17	0.005
HF-RBF	1.27	0.014	0.25	0.003	1.75	0.013	0.35	0.003
HF-linear	1.19	0.011	0.24	0.002	1.70	0.011	0.34	0.002
$W-logistic-l_1$	1.11	0.021	0.22	0.004	1.29	0.024	0.26	0.005
				IPWE $\Delta(\mathbf{X})$				
W-RF	1.79	0.018	0.36	0.004	2.04	0.018	0.41	0.004
HF-RBF	1.78	0.017	0.36	0.003	2.12	0.011	0.42	0.002
HF-linear	1.83	0.016	0.36	0.003	2.09	0.009	0.42	0.002
$W-logistic-l_1$	2.13	0.024	0.42	0.005	2.24	0.020	0.45	0.004

Table 4.3: Results for Scenario III summarized from 500 simulated datasets. The sample mean and standard error of the bias ($E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] - E[\tilde{Y}(\hat{g}(\tilde{\mathbf{X}}))]$) and misclassification error rate (MCR) are shown. $E[\tilde{Y}(g^{opt}(\tilde{\mathbf{X}}))] = 6.66$; $E[Y(1)] = 4.17$; $E[Y(0)] = 4.14$.

4.2.2 OOB Estimation Of Population Average Outcome

Above results of model assessment are based on the large testing set ($N = 10,000$) and the knowledge of optimal treatment rule, which give the most objective and reliable assessment of different methods. However, this approach of assessment is only applicable in simulation studies. To investigate model assessment methods applicable in real data analysis, we further compare the estimate of $E[Y(\hat{g}(\mathbf{X}))]$ based on the large testing set, the original training set, i,e. the naive estimate, and estimates of $E[Y(\hat{g}(\mathbf{X}))]$ from OOB and 5-fold CV procedure. The results of the comparison are summarized in Table 4.4. It is obvious that the naive estimates are overoptimistic due to the overfitting bias. The average of estimates from 5-fold CV procedure are uniformly lower than the average of estimates from the large testing set and the average of OOB estimates because only 80% of training data are used to derive the

ITR.

Compared to 5-fold CV estimates, the averages of OOB estimates of population average outcome under the estimated ITR from 500 simulations in most of the scenarios are very close to the estimates based on the the large testing sets (4.4). Among W-RFs with AIPWE used for $\Delta(\mathbf{X}_i)$, only in III-B, the average of OOB estimates (5.70) is significantly different from the average of estimates based on the large testing sets (5.81) based on paired T-test. In this scenario, the OOB estimates tend to be conservative. Notice that in our simulation study, there is a tuning parameter, *mtry*, which is selected by OOB estimate as well. But its impact on OOB estimate appears to be ignorable. In summary, based on the results from our simulation study, the OOB estimate can be used to evaluate the derived ITR even in the presence of some parameter tuning procedures.

$\hat{E}[Y(\hat{g})]$					
Scenario		Large Testing Set	OOB	5-Fold CV	Naive
I-A	AIPWE	5.53 (0.01)	5.47 (0.04)	5.33 (0.05)	7.01 (0.03)
	IPWE	4.76 (0.01)	4.68 (0.04)	4.73 (0.04)	9.19 (0.04)
I-B	AIPWE	5.47 (0.01)	5.53 (0.05)	5.42 (0.05)	7.35 (0.03)
	IPWE	4.62 (0.01)	4.58 (0.04)	4.59 (0.04)	9.21 (0.04)
II-A	AIPWE	7.30 (0.01)	7.29 (0.04)	7.09 (0.04)	7.79 (0.03)
	IPWE	5.60 (0.02)	5.51 (0.04)	5.49 (0.05)	10.43 (0.03)
II-B	AIPWE	6.94 (0.02)	6.94 (0.04)	6.52 (0.04)	7.99 (0.03)
	IPWE	5.15 (0.02)	5.02 (0.05)	4.98 (0.04)	10.35 (0.03)
III-A	AIPWE	6.00 (0.02)	5.97 (0.04)	5.72 (0.04)	7.53 (0.03)
	IPWE	4.87 (0.02)	4.81 (0.04)	4.82 (0.04)	9.39 (0.04)
III-B	AIPWE	5.81 (0.02)	5.70 (0.05)	5.48 (0.05)	7.79 (0.03)
	IPWE	4.62 (0.02)	4.57 (0.04)	4.54 (0.04)	9.40 (0.04)

Table 4.4: The sample mean and standard errors of $\hat{E}[Y(\hat{g})]$ of 500 simulations based on estimates from a large testing set (N=10,000), Out-of-Bag (OOB) estimate, 5-fold cross-validated (CV) estimate, and original training set (Naive estimate).

4.2.3 Variable Importance Measures

Figure 4.2 and Figure 4.3 display the distributions of two types of VIs in Scenario I-A and Scenario I-B when AIPWE is used. Figures for the other scenarios convey similar messages and thus are omitted. In general, VIs behave as we expected in Section 4.1, and on average VIs can identify the predictive biomarkers, (X_1, X_2). We notice that VI_1 works extremely well in separating (X_1, X_2) from the rest of $\mathbf{X}$, and we plot the logarithm of VI_1 for better visual display. Table 4.5 presents the summary statistics $U(VI)$. The summary statistics $U(VI)$ also show the superiority of VI_1. The relatively moderate discriminative performance of VI_2 might be due to the extra variability induced by the variability of OOB estimates $\hat{E}[Y(\hat{g}(\mathbf{X}^{(\bar{j})}))]^{oob}$. We notice that the VIs do not pick up prognostic biomarkers, which are purely for predicting outcome without providing information in selecting treatment. We also notice that the performance of VIs is positively correlated with the performance of W-RF in terms of maximizing $E[Y(g)]$. For example, $U(VI)$ is higher in Subsetting A compared to Subsetting B, and $U(VI)$ is higher when we use AIPWE for $\Delta(\mathbf{X})$; we already know that W-RF performs better when there are fewer irrelevant covariates (Subsetting A), and when AIPWE is used.

	AIPWE for $\Delta(\mathbf{X})$		IPWE for $\Delta(\mathbf{X})$	
Scenario	$U(VI_1)$	$U(VI_2)$	$U(VI_1)$	$U(VI_2)$
I-A	0.998 (0.02)	0.757 (0.25)	0.858 (0.17)	0.647 (0.25)
I-B	0.997 (0.01)	0.734 (0.28)	0.879 (0.15)	0.593 (0.29)
II-A	0.999 (0.01)	0.947 (0.13)	0.918 (0.12)	0.826 (0.18)
II-B	0.997 (0.01)	0.912 (0.18)	0.891 (0.14)	0.717 (0.24)
III-A	0.998 (0.02)	0.912 (0.19)	0.906 (0.13)	0.725 (0.28)
III-B	0.996 (0.02)	0.864 (0.24)	0.921 (0.11)	0.628 (0.33)

Table 4.5: Results for summary statistics U of two types of variable importance measures in different scenarios. The sample mean and standard deviation (in parentheses) from 500 simulated datasets are shown.

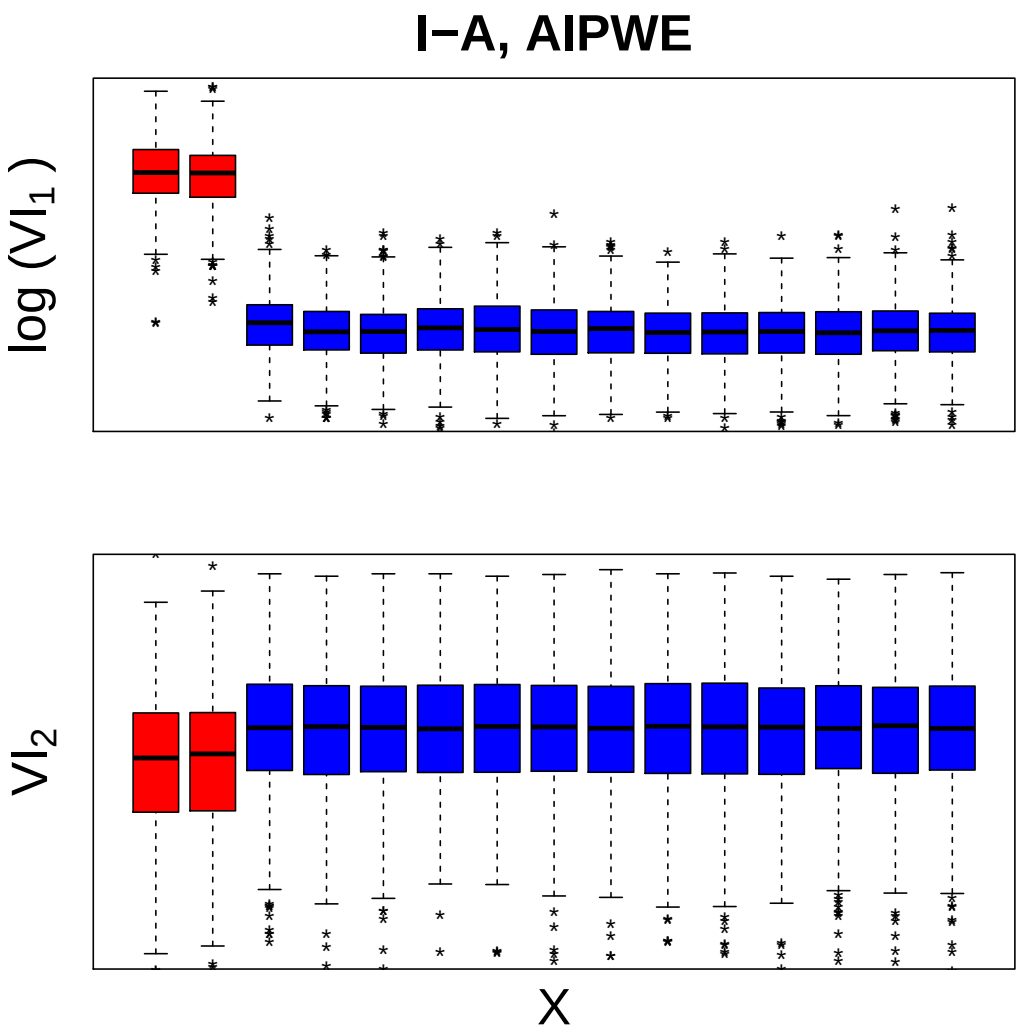

Figure 4.2: The distributions of two types of variable importance measures in Scenario I-A in 500 simulated datasets using AIPWE for $\Delta(\mathbf{X})$. The first two boxplots are distributions of VI of X_1 and X_2 (in red color), and the other boxplots of VIs of the rest of $\mathbf{X}$ are in blue color.

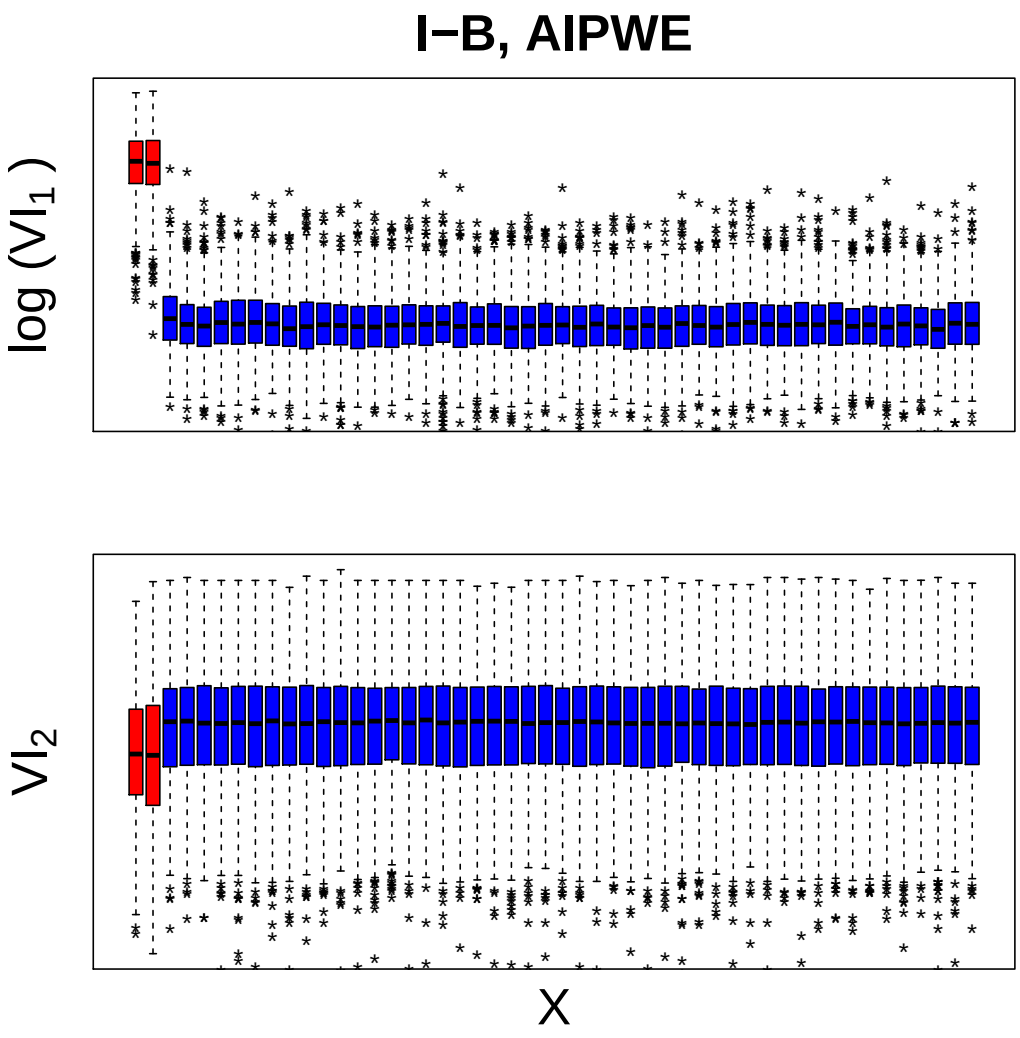

Figure 4.3: The distributions of two types of variable importance measures in Scenario I-B in 500 simulated datasets using AIPWE for $\Delta(\mathbf{X})$. The first two boxplots are distributions of VIs of X_1 and X_2 (in red color), and the other boxplots of VIs of the rest of $\mathbf{X}$ are in blue color.

Chapter 5

Data Example

In this Chapter, we analyze data from Clinical Antipsychotic Trials of Intervention Effectiveness Alzheimer's Disease Study (CATIE-AD) as an illustration of our proposed method: W-RF. CATIE-AD is a double-blinded randomized clinical study to assess the effectiveness of atypical antipsychotic drugs in patients with Alzheimer's disease (Schneider et al., 2006). CATIE-AD has four phases. In Phase 1, subjects were randomly assigned to receive olanzapine, quetiapine, resperdone, or placebo in a 2:2:2:3 ratio. After the first two weeks, physicians could discontinue the treatment and move a patient to the next phase if the patient's response was not adequate. One of the primary outcomes in Phase 1 is response rate at 12 weeks. Patients who were still in Phase 1 and with at least minimal improvement on the Clinical Global Impression of Change (CGIC) scale at 12 weeks were considered responding to the treatment.

In this data analysis example, we apply W-RF to analyze data from patients in olanzapine arm and placebo arm to derive an ITR to maximize patients' response rate based on baseline covariates. We consider a wide range of baseline characteristics recorded in CATIE-AD, including demographic characteristics, such as gender, age and martial status, clinical characteristics, such as Mini–Mental State Examination total score, and many com-

mon laboratory tests. Covariates are excluded if the measurements were not taken at baseline for about 10% or higher proportion of patients (e.g., Alzheimer's Disease Assessment Scale total score is excluded from the analysis).

In total, there were 49 baseline covariates recorded for most of the patients in the study. After excluding the 25 subjects with missing values in any of the 49 covariates (12 subjects in olanzapine arm and 13 subjects in placebo arm), there are 87 subjects in olanzapine arm and 126 subjects in placebo arm. In total, 213 subjects are included in the analysis. Among these 213 subjects, the response rate is 0.29 in olanzapine arm and 0.21 in placebo arm.

We consider all 49 covariates as potential predictive markers. We apply our method, W-RF, to derive an ITR based on the 213 subjects from CATIE-AD. The tuning parameters of W-RF, $mtry$, is chosen among $\{16, 32, 49\}$, which corresponds to $\{[1/3P], [2/3P], P\}$, based on OOB estimate as we did in the simulation study. The total number of bootstrapped trees is set to be 1000. AIPWE is used for $\Delta(\mathbf{X}_i)$ with the conditional mean function estimated by random forests. Alternative methods used in the simulation study in Chapter 4 are also applied to derive ITRs for comparison. The derived ITRs are evaluated by 5-fold CV estimates of response rate under the derived rule. In addition, we evaluate derived ITR by the estimated proportion of population recommended to the treatment. To reduce the variability induced by the random splitting during the 5-fold cross-validation process, the random sample splitting is performed 10 times, and the average of resulting 50 cross-validated estimates is calculated. For W-RF, OOB estimate of the response rate under the derived ITR is also calculated.

To address the sampling variability in estimators, percentile bootstrap confidence intervals are obtained. The procedures described in the previous paragraph are performed on 100 bootstrapped sample of the 213 subjects, and cross-validated estimates (and OOB estimate for W-RF) of the response rate and the proportion recommended to treat under the

derived ITR are calculated. The 2.5 % and 97.5 % sample percentile of these 100 estimates are provided as the lower and upper bounds of percentile bootstrap confidence intervals.

The cross-validated estimate of response rate under the ITR derived from W-RF is 0.303 (95% CI: 0.289 - 0.314), and the OOB estimate of response rate under the ITR derived from W-RF is 0.312 (95% CI: 0.291 - 0.330). Both estimates suggest that the response rate under ITR derived from W-RF are at least comparable to the response rate in olanzapine arm, which corresponds to a rule that recommends olanzapine to all patients. However, the ITR derived from W-RF only requires to treat about 60% of population with olanzapine, represented by the subjects in CATIE-AD. In this data analysis, because the outcome is binary, instead of penalized least square, the direct regression method on outcome is logistic regression with L-1 penalty, which is implemented with the tuning parameter selected by the *cv.glmnet* with default settings. The logistic regression with L-1 penalty fails to select any predictor for the response outcome, hence the corresponding derived ITR is to recommend the treatment to all patients. The estimates from the other methods are summarized in Table 5.1. The estimated response rates derived from the other methods are also comparable to the response rate in olanzapine arm, but W-RF has larger estimate of response rate than all other methods.

Methods	Response Rate Under $\hat{g}$	Proportion Treated
W-RF	0.303 (0.289-0.314)	0.625 (0.615-0.636)
Direct-RF	0.291 (0.253-0.322)	0.717 (0.690-0.752)
HF-linear	0.282 (0.267-0.299)	0.599 (0.580-0.624)
HF-RBF	0.290 (0.276-0.306)	0.854 (0.830-0.882)
$W - logistic - l_1$	0.288 (0.263-0.307)	0.941 (0.890-0.982)

Table 5.1: Cross-validated estimates of response rate under derived ITR ($\hat{E}[Y(\hat{g})]$) and proportion of the population recommended to treat ($\hat{P}(\hat{g} = 1)$) from different methods. 95% percentile bootstrap confidence intervals are shown in the parentheses

We also assess each covariate by the first type variable importance measure (VI_1). Based on VI_1, baseline chloride lab test (CHLORIDE), baseline Brief Psychiatric Rating Scale (BPRS), baseline lactate dehydrogenase lab test (LDH) and baseline Neuropsychiatric

Inventory Score (NPIS) are shown to be more relevant in deriving optimal ITR than other covariates (See Figure 5.1).

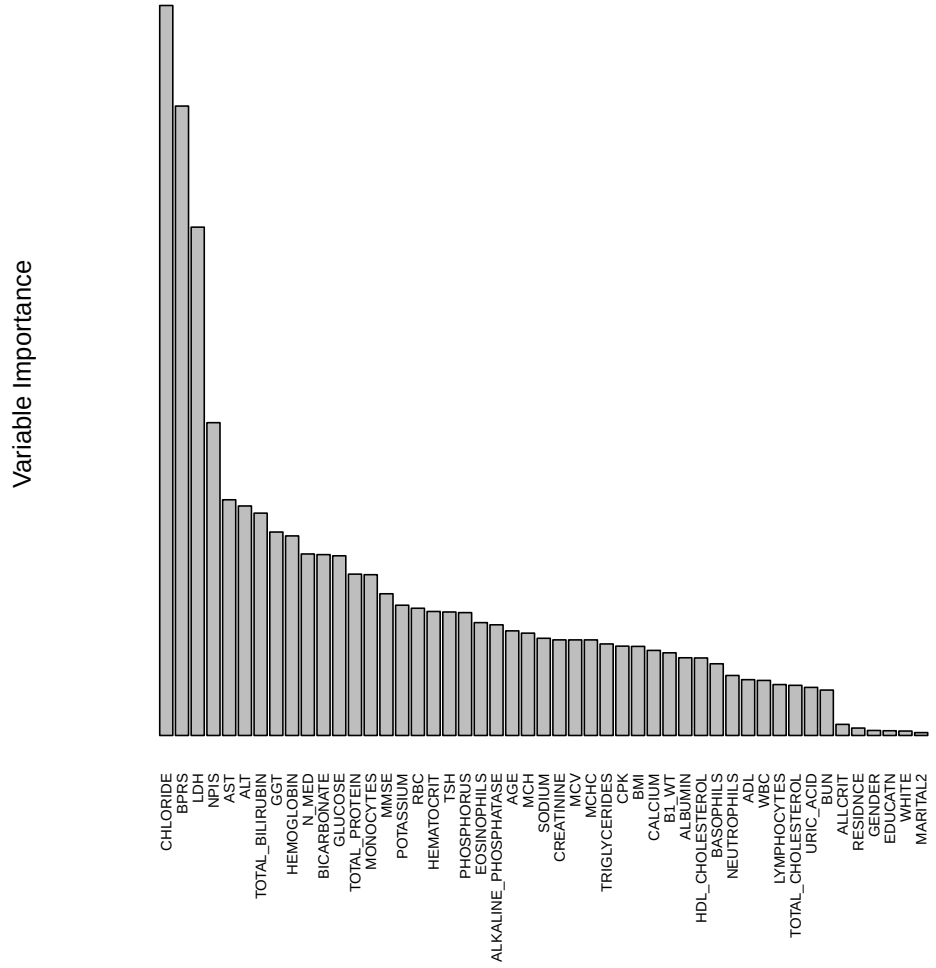

Figure 5.1: Variable importance of 49 baseline covariates

In addition, we also construct the CATE curves (Han et al., 2015) to marginally assess these four covariates for treatment selection. $\beta(X)$ is the covariate-specific coefficient of treatment indicator in the varying-coefficient model with the logit link. As we discussed in Section 2.2, we should recommend to treat patients with CATE curve above horizontal line $\beta(X) = 0$ and not to treat patients with CATE curve below horizontal line $\beta(X) = 0$.

Based on the CATE curves (Figure 5.2), we might not recommend to treat patients with serum chloride and lactate dehydrogenase (LHD) outside the normal range (the normal range for chloride: 96 to 106 mEq/L; the normal range for LDH 105 - 333 IU/L). Also, patients with severe symptoms might be more likely to be responsive to the treatment because CATE curves are above horizontal line $\beta(X) = 0$ when BPRS scores and NPIS scores are higher, which indicate severe symptoms. However, we should note that all 95% simultaneous confidence bands of CATE curves contain horizontal line $\beta(X) = 0$, which suggest a 'statistically non-significant result', and we should be cautious about these patterns without further investigations and more evidence.

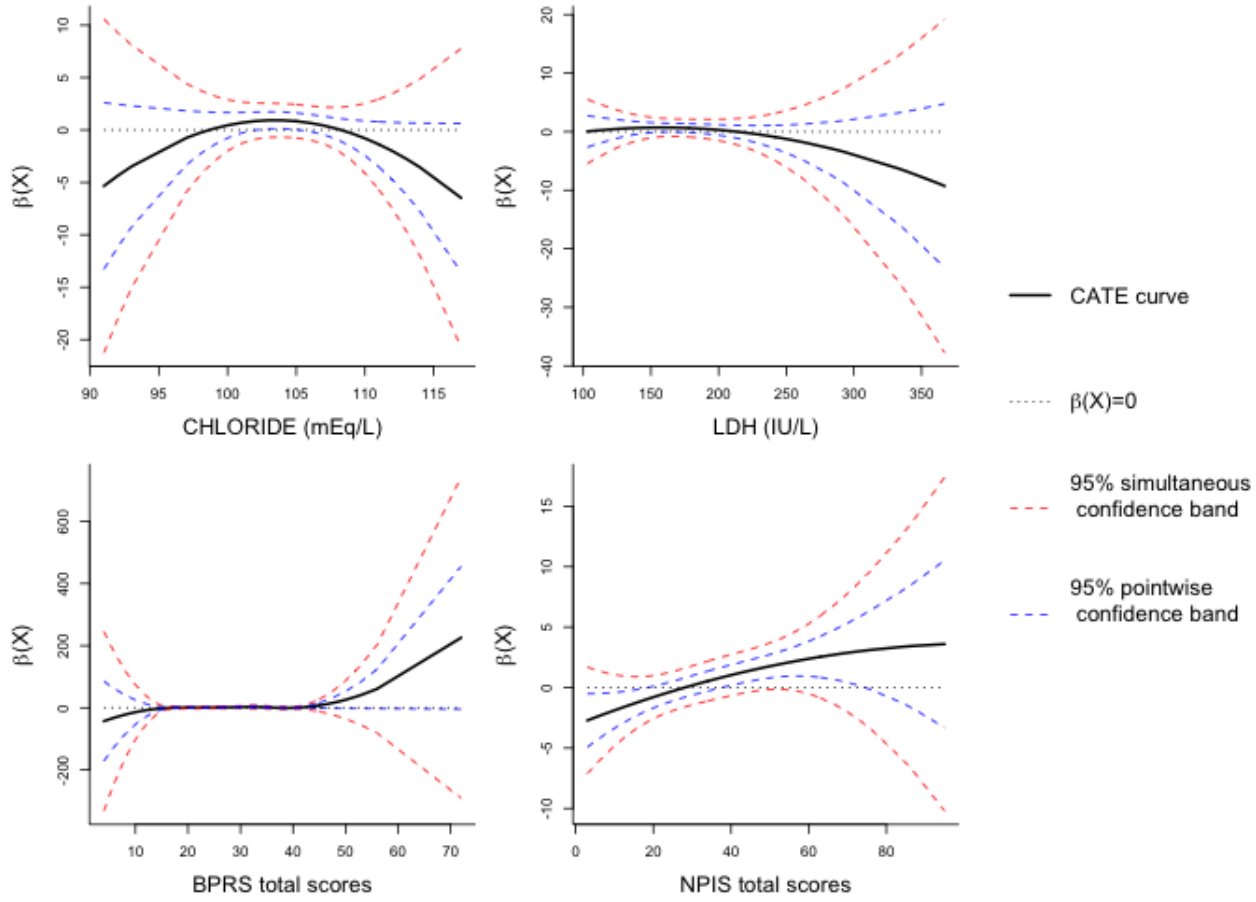

Figure 5.2: CATE curves of 4 most relevant covariates selected by the variable importance measures of W-RF

In summary, W-RF could be used to derive an ITR, which achieves comparable response rate to the response rate of olanzapine arm; but the derived ITR only recommend about 60% of patients to take olanzapine. This can alleviate the treatment burden of olanzapine such as side-effects on the patients as well as financial cost to the health care system. The variable importance measures of baseline covariates could be helpful in inspiring further clinical research in treatment heterogeneity.

Chapter 6

Discussion

In this thesis, we review the current statistical researches on individualized treatment rules. We review causal interpretations of the optimal treatment rule based on $\Delta(\mathbf{X})$ and an intuitive interpretation of the weighted classification framework. Some statistical methods focus on estimation of the conditional mean model $E[Y|\mathbf{X}, A]$ in the first step, which often involve modeling interactions between treatment and baseline covariates; and an ITR is constructed based on $\hat{\Delta}(\mathbf{X}) = \hat{E}[Y|\mathbf{X}, 1] - \hat{E}[Y|\mathbf{X}, 0]$ in the second step. As a more direct approach, the weighted classification framework casts the problem of optimization of clinical outcome as a weighted classification problem. Under this framework, we propose to apply the flexible classification method, random forests, to derive an optimal ITR.

In our simulation study, our proposed method, W-RF, shows better performance in terms of minimizing the misclassification rate and optimizing the desired clinical outcome in the presence of non-linear decision boundary and high dimension covariates. Comparison between our purposed W-RF and the method using random forests modeling $E[Y|\mathbf{X}, A]$ (Direct-RF) shows the advantage of using the weighted classification framework.

Moreover, in our simulation study, our proposed variable importance measures, especially VI_1, show extraordinary performance in ranking candidate biomarkers according

to their relevance to optimizing the estimated average clinical outcome under the treatment rule. Our VIs provide a ranking without an explicit cut-off point to select the relevant baseline covariates. Further research is warranted to construct a systematic variable selection algorithm in W-RF.

It should be clear that the application of either our proposed method or many other methods discussed in this thesis to derive an ITR is an exploratory data analysis. Further confirmatory studies are needed for the estimated ITR before the ITR is used in practice. In addition to derive an ITR, one major goal of this exploratory data analysis is to motivate further clinical research in treatment heterogeneity. As in all other applications of random forests, the superior performance of random forests is at a cost of interpretability. The ITR from 'black box' models, including W-RF and models with non-linear kernels, will not provide any etiological knowledge or biological mechanism on the treatment heterogeneity. The lack of interpretability makes the variable importance measure in W-RF particularly useful in identifying the potential predictive biomarkers and generating scientific hypothesis regarding their associations with treatment heterogeneity.

Bibliography

Breiman, L. (1996). Bagging Predictors. *Machine Learning*, 24(2):123–140.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1):5–32.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA.

Cai, T., Tian, L., Wong, P. H., and Wei, L. J. (2011). Analysis of randomized comparative clinical trial data for personalized treatment selections. *Biostatistics (Oxford, England)*, 12(2):270–82.

Collins, F. S. and Varmus, H. (2015). A New Initiative on Precision Medicine. *The New England journal of medicine*, 372(9):793–5.

Fisher, B., Redmond, C., Brown, A., Wickerham, D. L., Wolmark, N., Allegra, J., Escher, G., Lippman, M., Savlov, E., and Wittliff, J. (1983). Influence of tumor estrogen and progesterone receptor levels on the response to tamoxifen and chemotherapy in primary breast cancer. *Journal of clinical oncology : official journal of the American Society of Clinical Oncology*, 1(4):227–41.

Ford, D., Easton, D., Stratton, M., Narod, S., Goldgar, D., Devilee, P., Bishop, D., Weber, B., Lenoir, G., Chang-Claude, J., Sobol, H., Teare, M., Struewing, J., Arason, A., Scherneck, S., Peto, J., Rebbeck, T., Tonin, P., Neuhausen, S., Barkardottir, R., Eyfjord, J., Lynch, H., Ponder, B., Gayther, S., Birch, J., Lindblom, A., Stoppa-Lyonnet, D., Bignon,

Y., Borg, A., Hamann, U., Haites, N., Scott, R., Maugard, C., Vasen, H., Seitz, S., Cannon-Albright, L., Schofield, A., and Zelada-Hedman, M. (1998). Genetic Heterogeneity and Penetrance Analysis of the BRCA1 and BRCA2 Genes in Breast Cancer Families. *The American Journal of Human Genetics*, 62(3):676–689.

Han, K., Zhou, X.-H., and Liu, B. (2015). CSTE Curve for Selection the Optimal Treatment When Outcome Is Binary. *Science China Mathematics*.

Hastie, T. J., Tibshirani, R. J., and Friedman, J. H. (2009). *The elements of statistical learning : data mining, inference, and prediction*. Springer series in statistics. Springer, New York.

Huang, Y. (2015). Identifying optimal biomarker combinations for treatment selection through randomized controlled trials. *Clinical trials (London, England)*, 12(4):348–56.

Huang, Y. and Fong, Y. (2014). Identifying optimal biomarker combinations for treatment selection via a robust kernel method. *Biometrics*, 70(4):891–901.

Huang, Y. and Wang, P. (2012). Network Based Prediction Model for Genomics Data Analysis. *Statistics in biosciences*, 4(1).

Janes, H., Brown, M. D., Huang, Y., and Pepe, M. S. (2014). An approach to evaluating and comparing biomarkers for patient treatment selection. *The international journal of biostatistics*, 10(1):99–121.

Kang, C., Janes, H., and Huang, Y. (2014). Combining biomarkers to optimize patient treatment recommendations. *Biometrics*, 70(3):695–707.

Ma, Y. and Zhou, X.-H. (2014). Treatment selection in a randomized clinical trial via covariate-specific treatment effect curves. *Statistical methods in medical research*.

Matsouaka, R. A., Li, J., and Cai, T. (2014). Evaluating marker-guided treatment selection strategies. *Biometrics*, 70(3):489–99.

National Research Council (2011). *Toward Precision Medicine: Building a Knowledge Network for Biomedical Research and a New Taxonomy of Disease*. National Academies Press, Washington, D.C.

Pocock, S. J., Assmann, S. E., Enos, L. E., and Kasten, L. E. (2002). Subgroup analysis, covariate adjustment and baseline comparisons in clinical trial reporting: current practice and problems. *Statistics in medicine*, 21(19):2917–30.

Qian, M. and Murphy, S. A. (2011). Performance Guarantees for Individualized Treatment Rules. *Annals of statistics*, 39(2):1180–1210.

Roukos, D. H. and Briasoulis, E. (2007). Individualized preventive and therapeutic management of hereditary breast ovarian cancer syndrome. *Nature clinical practice. Oncology*, 4(10):578–90.

Rubin, D. B. and van der Laan, M. J. (2012). Statistical issues and limitations in personalized medicine research with clinical trials. *The international journal of biostatistics*, 8(1):18.

Schneider, L. S., Tariot, P. N., Dagerman, K. S., Davis, S. M., Hsiao, J. K., Ismail, M. S., Lebowitz, B. D., Lyketsos, C. G., Ryan, J. M., Stroup, T. S., Sultzer, D. L., Weintraub, D., and Lieberman, J. A. (2006). Effectiveness of Atypical Antipsychotic Drugs in Patients with Alzheimer's Disease. *New England Journal of Medicine*, 355(15):1525–1538.

Taylor, J. M. G., Cheng, W., and Foster, J. C. (2015). Reader reaction to "a robust method for estimating optimal treatment regimes" by Zhang et al. (2012). *Biometrics*, 71(1):267–71.

Tian, L. (2014). Discussion of "Combining biomarkers to optimize patient treatment recommendations". *Biometrics*, 70(3):710–3.

Venkitaraman, A. R. (2002). Cancer Susceptibility and the Functions of BRCA1 and BRCA2. *Cell*, 108(2):171–182.

Zhang, B., Tsiatis, A. A., Davidian, M., Zhang, M., and Laber, E. (2012a). Estimating optimal treatment regimes from a classification perspective. *Stat*, 1(1):103–114.

Zhang, B., Tsiatis, A. A., Laber, E. B., and Davidian, M. (2012b). A robust method for estimating optimal treatment regimes. *Biometrics*, 68(4):1010–8.

Zhao, Y., Zeng, D., Rush, A. J., and Kosorok, M. R. (2012). Estimating Individualized Treatment Rules Using Outcome Weighted Learning. *Journal of the American Statistical Association*, 107(449):1106–1118.

Zhao, Y.-Q. and Kosorok, M. R. (2014). Discussion of "Combining biomarkers to optimize patient treatment recommendations". *Biometrics*, 70(3):713–6.

Zhou, X. and Ma, Y. (2012). BATE curve in assessment of clinical utility of predictive biomarkers. *Science China Mathematics*, 55(8):1529–1552.

Appendix A

R Functions To Implement W-RF

```
library(compiler);library(Rcpp) #load required packages.

cppFunction('NumericVector pred_tree_cpp(NumericMatrix tree,
            NumericMatrix data){
            // Function to predict the classification
            // label from a single tree
            // Args:
            //   tree: a matrix representing a tree
            //   data: a matrix of (testing) data
            // Returns:
            //   yhat: the predicted class label

            int obs = data.nrow();
            NumericVector out(obs);
            int node=0;

            for(int i = 0; i< obs; ++i){
            node = 0;
            while(TRUE){
            if(tree(node,6) == 1){
            out[i] = tree(node,5);
            break;
            }
            else if ( data(i,(tree(node,1)-1)) > tree(node,2) )
            {
            node = (tree(node,4)-1);
            } else
            {
            node = (tree(node,3)-1);
            }
```

```
            }
            }
            return out;
            }')

impurity.dat<-function(dat)
{
  # Function to calculate impurity / weighted
  # misclassifcate errors of a node
  # Args:
  #   dat: a matrix of data consisting Xs (covariates),
  #        y (estimate class label) and wt (weight)
  # Returns:
  #   weighted misclassifcate errors

  wt<-dat[,"wt"]
  y<-dat[,"y"]
  min(sum(wt*(y-1)^2),sum(wt*(y-0)^2))
}

for.scan<-function(i,yy,wt,N){

  # Function to calculate the impurity of particular split point
  # Args:
  #   i: index to split
  #   yy: class label
  #   wt: weight
  #   N: total number of observation in this node
  # Returns:
  #   impurity of particular split point

  y.l<-yy[1:i]
  wt.l<-wt[1:i]
  y.r<-yy[(i+1):N]
  wt.r<-wt[(i+1):N]
  l.imp<-min(sum(wt.l*(y.l-1)^2),sum(wt.l*(y.l-0)^2))
  r.imp<-min(sum(wt.r*(y.r-1)^2),sum(wt.r*(y.r-0)^2))
  return(c(l.imp,r.imp))
}

c.for.scan<-cmpfun(for.scan)

scan<-function(mydat,feature){

  # Function to scan a covariate and find the best cutoff point
  # Args:
  #   mydat: data in the node
  #   feature: the covariate considered to split
```

```
  # Returns:
  #   The best cutoff point and corresponding impurity measure

  mydat<-mydat[order(mydat[,feature]),]
  N<-nrow(mydat)
  impurity.l<-numeric(N-1)
  impurity.r<-numeric(N-1)
  sequence<-seq(1:(N-1))
  yy<-mydat[,"y"]
  wt<-mydat[,"wt"]

  tmp<-sapply(sequence,c.for.scan,yy=yy,wt=wt,N=N)

  impurity.l<-tmp[1,]
  impurity.r<-tmp[2,]
  index.multi<-which((impurity.l+impurity.r) == min(impurity.l+impurity.r))
  index<-ifelse(length(index.multi)>1,sample(index.multi,size=1),index.multi)
  l.cutoff<-mydat[index, feature]
  c.impurity<-min(impurity.l+impurity.r)
  return(c(l.cutoff,c.impurity))
}

c.scan<-cmpfun(scan)

compare<-function(x.ind,mydat)
{
  # Function to find the best covariate and cutoff point combination
  # Args:
  #   x.ind: covariates randomly selected to be the candidate
  #   mydat: data in the node
  # Returns:
  #   the best covariate and cutoff point

  all.cut<-matrix(ncol=2,nrow=length(x.ind))
  for (index in 1:length(x.ind))
  {
    all.cut[index,c(1:2)]<-c.scan(mydat=mydat,feature=x.ind[index])
  }

  index.many<-which(all.cut[,2]==min(all.cut[,2],na.rm=T))
  #randomly select one node if equal
  index<-ifelse(length(index.many)>1,sample(x=index.many,size=1),index.many)
  s.feature<-x.ind[index]
  return(c(s.feature,all.cut[index,]))
}

c.compare<-cmpfun(compare)
```

```
mytree<-function(dat,min.node.size,x.try)
{
  # Function to build a single tree
  # Args:
  #   dat: a matrix of data consisting Xs (covariates),
  #        y (estimate class label) and wt (weight)
  #   x.try:  (aka mtry) number of Xs considered to split
  #   min.node.size: minimal node size
  # Returns: a list of elements:
  # 1. Matrix represent a single tree-based model
  # 2. OOB prediction without / with X_j permuted
  # 3. Type 1 Variable importance

  #maximal node size
  node<-seq(1:(2*nrow(dat)-min.node.size+2))

  #predictor index selected to be splitted
  x.index<-seq(1:(ncol(dat)-2))

  # a matrix representing a tree model
  tree.matrix<-cbind(node,feature=NA,
                     cutoff=NA,impurity=NA,
                     children.impurity=NA,
                     left.node=NA,
                     right.node=NA,a=NA,terminal=-1,Stop=0)
  # list of data in each tree node
  list.dat<-list(dat)

  #initiate the loop
  tree.matrix[1,"terminal"]<-1
  tree.matrix[1,"impurity"]<-impurity.dat(list.dat[[1]])
  new.node<-1

  #loop to implement recursive binary splitting algorithm
  while(sum(tree.matrix[,"terminal"]==1 & tree.matrix[,"Stop"]==0)!=0)
  {
    terminal.node<-tree.matrix[tree.matrix[,"terminal"]==1 &
                                 tree.matrix[,"Stop"]==0,"node"]
    split.matrix<-cbind(node,x.ind=NA,cutoff=NA,children.impurity=NA)

    for(j in terminal.node)
    {
      if(nrow(list.dat[[j]])>min.node.size &
         impurity.dat(dat=list.dat[[j]])>0)
      {
        split.matrix[j,c("x.ind","cutoff","children.impurity")]<-
          c.compare(x.ind=sort(sample(x.index,size=x.try,replace=F)),
                    mydat=list.dat[[j]])
```

```
  } else
  {

    tree.matrix[j,"Stop"]<-1
    terminal.data<-list.dat[[j]]
    tree.matrix[j,c("a")] <- ifelse(
      sum((terminal.data[,"y"]-1)^2*terminal.data[,"wt"])<
        sum((terminal.data[,"y"]-0)^2*terminal.data[,"wt"]),1,0)

  }
}

if (prod(tree.matrix[tree.matrix[,"terminal"]==1,"Stop"])){break}
#the largest decrease
impurity.descrease<-tree.matrix[,"impurity"]-
  split.matrix[,"children.impurity"]
split.index.multi<-as.vector(which(
  impurity.descrease==max(impurity.descrease,na.rm=T)))
split.index<-ifelse(length(split.index.multi)>1,
                    sample(x=split.index.multi,size=1),split.index.multi)
split.node<-split.matrix[split.index,]

tree.matrix[tree.matrix[,"node"]==split.node["node"],
            c("feature","cutoff","children.impurity","terminal")]<-
  c(split.node[-1],1)

split.data<-list.dat[[split.node["node"]]]
list.dat[[new.node+1]]<-split.data[split.data[,split.node["x.ind"]]<=
                                     split.node["cutoff"],]
list.dat[[new.node+2]]<-split.data[split.data[,split.node["x.ind"]]>
                                     split.node["cutoff"],]
if(class(list.dat[[new.node+1]])=='numeric')
{
  list.dat[[new.node+1]]<-t(as.matrix(list.dat[[new.node+1]]))
}
if(class(list.dat[[new.node+2]])=='numeric')
{
  list.dat[[new.node+2]]<-t(as.matrix(list.dat[[new.node+2]]))
}



tree.matrix[tree.matrix[,"node"]==split.node["node"],
            c("left.node","right.node","terminal")]<-
  c(new.node+1,new.node+2,0)
tree.matrix[c(new.node+1,new.node+2),"terminal"]<-c(1,1)
tree.matrix[new.node+1,"impurity"]<-impurity.dat(list.dat[[new.node+1]])
tree.matrix[new.node+2,"impurity"]<-impurity.dat(list.dat[[new.node+2]])
```

```
    new.node<-new.node+2

  }


  #Type 1 VI
  tmp<-tree.matrix[tree.matrix[,"terminal"]!=-1,]
  tmp<-tmp[is.na(tmp[,"feature"])==F,
           c("feature","impurity","children.impurity")]
  if(class(tmp)=="matrix")
  {
    reduced<-tmp[,"impurity"]-tmp[,"children.impurity"]
    red.im<-as.matrix(xtabs(reduced~tmp[,"feature"]))
  } else if (class(tmp)=="numeric")
  {
    reduced<-tmp["impurity"]-tmp["children.impurity"]
    red.im<-as.matrix(xtabs(reduced~tmp["feature"]))
  }

  vi<-rbind(x.index,0)
  for(i in x.index)
  {
    if(sum(rownames(red.im)==i)!=0)
    {
      vi[2,i]<-red.im[rownames(red.im)==i]
    } else next
  }


  #formating the final tree model in the matrix format
  tree.matrix<-tree.matrix[,c("node","feature","cutoff",
                              "left.node","right.node","a","terminal")]
  tree.matrix<-tree.matrix[tree.matrix[,"terminal"]!=-1,]
  tree.matrix[is.na(tree.matrix)]<- -1
  if(class(tree.matrix)=="numeric"){tree.matrix = t(as.matrix(tree.matrix))}

  return(list(tree.matrix,vi))
}
c.mytree<-cmpfun(mytree)

one.forest<-function(dat,x.try,min.node.size)
{
  # Function to bootstrap samples and build a single tree
  # Args:
  #   dat: a matrix of data consisting Xs (covariates),
  #        y (estimate class label) and wt (weight)
  #   x.try:  (aka mtry) number of Xs considered to split
  #   min.node.size: minimal node size
```

```
  # Returns: a list of elements:
  # 1. Matrix represent a single tree-based model
  # 2. OOB prediction without / with X_j permuted
  # 3. Type 1 Variable importance

  N<-nrow(dat)
  p<-ncol(dat)-2 #number of predictors
  dat.ind<-seq(1:N)
  # sample index with replacement (boostrap)
  boot.ind<-sample(dat.ind,N,replace=T)
  # out-of-bag (OOB) index
  oob.ind<-setdiff(dat.ind,boot.ind)

  # estimate individual weight and label
  datBoot<-data.prepare(dat=dat[boot.ind,],type='AIPWE.rf',ntree=500)

  # call c.mytree to build a single tree
  tmp<-c.mytree(dat=datBoot,
                min.node.size=min.node.size,x.try=x.try)
  one.tree<-tmp[[1]]
  oob.dat<-dat[oob.ind,]

  # OOB predictions with and without X_j permuted
  oob.prediction<-matrix(nrow=N,ncol=p+1)
  oob.prediction[oob.ind,1]<-pred_tree_cpp(as.matrix(one.tree),oob.dat)
  for(i in 1:p)
  {
    permute.oob<-oob.dat
    permute.oob[,i]<-sample(oob.dat[,i],replace=F)
    oob.prediction[oob.ind,i+1]<- pred_tree_cpp(one.tree,permute.oob)
  }

  return(list(one.tree,oob.prediction,tmp[[2]]))
}

c.one.forest<-cmpfun(one.forest)

w.rf<-function(dat,ntree=100,x.try,min.node.size){
  # Function to implement W-RF
  # Args:
  #   dat: a matrix of data consisting Xs (covariates),
  #        y (estimate class label) and wt (weight)
  #   x.try:  (aka mtry) number of Xs considered to split
  #   min.node.size: minimal node size
  # Returns: a list of 3 elements:
  # 1. Matrix represent tree-based models
  # 2. OOB prediction without / with X_j permuted
  # 3. Type 1 Variable importance
```

```
  #repetitvely call c.one.forest to build a single boostrapped tree
  results<-replicate(ntree,
          c.one.forest(dat=dat,x.try=x.try,min.node.size=min.node.size))

  oob.pred<-apply(simplify2array(results[2,]),
                  c(1,2), FUN=function(x)(mean(x,na.rm=T)>0.5))

  impurity.reduce<-apply(simplify2array(results[3,]),
                         c(1,2),FUN=function(x)(mean(x,na.rm=T)))

  trees<-results[1,]
  return(list(trees,oob.pred,impurity.reduce))
}

c.w.rf<-cmpfun(w.rf) #complied the function for better performance



library(glmnet);library(randomForest)
data.prepare<-function(dat,type,ntree)
{
  # Function to estimate individual weights and
  # labels for bootstrapped data
  # Args:
  #   dat: a dataframe consisting Xs (covariates),
  #        Y (outcome) and A (trt assignment)
  #   type: type of weights
  #   min.node.size: minimal node size
  # Returns: a matrix of data consisting Xs (covariates),
  #        y* (estimate class label) and wt (weight)
  dat<-as.data.frame(dat)
  p<-ncol(dat)-2
  if(type=="IPWE"){
    wt<- 2*(dat$y*dat$t-dat$y*(1-dat$t))
  }
  if(type=="AIPWE.lm")
  {
    y<-dat[,"y"]
    xx<-dat[,1:p]
    xx.f<-cbind(1,xx,xx*dat[,"t"],t=dat[,"t"])
    fit<-glm.fit(x=xx.f,y=y)
    xx1<-cbind(1,xx,xx*1,t=1);xx0<-cbind(1,xx,xx*0,t=0)
    fit.coeff<-t(as.matrix(coef(fit)))
    u1<-fit.coeff %*% t(xx1)
    u0<-fit.coeff %*% t(xx0)
    wt<- as.numeric (2*(dat$y*dat$t-dat$y*(1-dat$t)-
```

```
                        (dat$t-0.5)*u1-(dat$t-0.5)*u0))
  }

  if(type=='AIPWE.rf')
  {
    u<-rf.mu(dat=dat,ntree=ntree)
    wt<- 2*(dat$y*dat$t-dat$y*(1-dat$t)-(dat$t-0.5)*u[,1]-(dat$t-0.5)*u[,2])
  }

  dat.out<-as.data.frame(dat[,c(1:p)])
  dat.out$wt<-abs(wt)
  dat.out$y<-as.numeric(wt>0)
  return(as.matrix(dat.out))
}

rf.mu<-function(dat,ntree)
{

  dat<-as.data.frame(dat)
  p<-ncol(dat)-1

  model.rf<-randomForest(y~.,ntree=ntree,data=dat)
  data.rf0<-dat;data.rf0$t<-0
  data.rf1<-dat;data.rf1$t<-1
  u1<-predict(model.rf,newdata=data.rf1)
  u0<-predict(model.rf,newdata=data.rf0)
  return(cbind(u1,u0))
}


pred.forest<-function(dat,forest.obj)
{
  # Function to predict the classification label from a forest
  # Args:
  #   dat: a matrix of data consisting of covariates / predictors
  #   forest.obj: a list return from w.rf
  # Returns:
  #    the predicted class label / treatment rule

  ntree<-length(forest.obj[[1]])
  as.numeric(t(rowMeans(sapply(seq(1:ntree),
                        FUN=function(x){
                          pred_tree_cpp(as.matrix(forest.obj[[1]][[x]]),dat)})
  )>0.5))
}

c.pred.forest<-cmpfun(pred.forest)
```

Appendix B

A Demonstration Of The R Functions In Appendix A

```
# A demonstration of the functions which implement W-RF
library(MASS)
data.gen<-function(N=100,nx=15,case="one")
{
  # a function to generate data from
  # different simulation scenarios
  xx<-mvrnorm(n=N,mu=c(0,0),
              Sigma = matrix(c(1,0.2,0.2,1),ncol=2))
  colnames(xx)<-c("x1","x2")
  x1<-xx[,1]; x2<-xx[,2]
  for (j in 3:nx)
  {
    assign(paste('x',j,sep=''),rnorm(n=N))
    xx<-eval(parse(text=paste0('cbind(xx,','x',j,')')))
  }

  t<-rbinom(n=N, size=1,prob=0.5)

  mu<-2+x1+x1^2+x2+x2^2+x3+log(abs(x4))+
    0.1*exp(x5)+sin(x6)+cos(x7)

  if (case=="one")
  {
    delta<-3*(x1 +x2)
  } else if (case=="two")
  {
```

```
    delta<-9 - 4*x1^2 - 4*x2^2
  } else if (case=="three")
  {
    delta<-5*(2*as.numeric(x1> -0.5 & x2> -0.5)-1)
  }
  q<-mu+delta * t
  y<-sapply(q, FUN=function(x){rnorm(n=1,mean=x,sd=1)})
  out<-cbind(xx,t,y)
  return(out)
}

theta<-function(t,y,rule)
{
  # function to estimate
  # the population average outcome under
  # a treatment rule
  sum(y*rule*t)/sum(t)+
    sum(y*(1-rule)*(1-t))/sum((1-t))
}

# generate training dataset and testing dataset
dat.train<-data.gen(N=100,nx=15)
dat.test<-data.gen(N=100000,nx=15)

# using the training dataset to build the model
model<-c.w.rf(dat=dat.train,ntree=500,x.try=10,min.node.size = 1)
# 1st type Variable importance measure
VI.1<-model[[3]]

# using the testing dataset to predict the rule and
rule<-c.pred.forest(dat=as.matrix(dat.test),forest.obj=model)
# estimate the population average outcome under the derived rule
theta.test<-theta(t=dat.test[,"t"],y=dat.test[,"y"],rule=rule)
# oob estimate of the population average outcome under the derived rule
theta.oob<-theta(t=dat.train[,"t"],
                 y=dat.train[,"y"],rule=as.numeric(model[[2]][,1]))
```