DEVELOPING MACHINE LEARNING METHODOLOGY FOR PRECISION HEALTH

Xiaotong Jiang

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Biostatistics in the Gillings School of Global Public Health.

Chapel Hill
2020

Approved by:

Michael R. Kosorok

Donglin Zeng

Jianwen Cai

Todd A. Schwartz

Richard F. Loeser

ABSTRACT

Xiaotong Jiang: Developing Machine Learning Methodology for Precision Health
(Under the direction of Michael R. Kosorok)

Precision health has been an increasingly popular solution to improve health care quality and guide the decision making process. This includes precision medicine (at the individual level) and precision public health (at the population level such as communities and institutions). By learning from the available medical data with advanced analytical tools, precision health recommends the treatments that are individualized to each patient or entity to maximize clinical outcomes for each individual.

We extend and develop three machine learning methods to improve the estimation of optimal individualized treatment regimes in precision health: the jackknife estimator of value function of precision medicine models compared with zero-order models, doubly robust outcome-weighted estimators with deep neural network structures for complex and large data, and risk-adjusted adverse event monitoring for survival data. First, motivated by a knee osteoarthristis trial, we estimate value functions and select the optimal treatment with the jackknife method whose consistency is established under weak assumptions. Next, we implement deep learning architecture in augmented outcome-weighted learning to increase model flexibility and computation efficiency, especially for high-dimensional data such as medical imaging. Lastly, we develop a risk-adjusted survival model to monitor adverse events and estimate its variance for hierarchical, right-censored data with recurrent events. All three methodologies aim to solve practical, health-related challenges and provide data-driven decision support and operations.

To Benjamin.

ACKNOWLEDGEMENTS

This work was partially supported by the NIAMS grant P60 AR064166, P30 AR072580, P01 CA142538, and UL1 TR002489 as well as Cystic Fibrosis Foundation 3rd Year Clinical Fellowship Grant - "Monitoring Risk-Adjusted Incidence Rates of MRSA and P. aeruginosa" STOUDE18A0-D3. Chapter 1 used data from the IDEA trial which was funded by a grant from NIAMS (R01 AR052528). Chapter 2 used NACC database which is funded by NIA/NIH Grant U01 AG016976. NACC data are contributed by the NIA-funded ADCs: P30 AG019610 (PI Eric Reiman, MD), P30 AG013846 (PI Neil Kowall, MD), P30 AG062428-01 (PI James Leverenz, MD) P50 AG008702 (PI Scott Small, MD), P50 AG025688 (PI Allan Levey, MD, PhD), P50 AG047266 (PI Todd Golde, MD, PhD), P30 AG010133 (PI Andrew Saykin, PsyD), P50 AG005146 (PI Marilyn Albert, PhD), P30 AG062421-01 (PI Bradley Hyman, MD, PhD), P30 AG062422-01 (PI Ronald Petersen, MD, PhD), P50 AG005138 (PI Mary Sano, PhD), P30 AG008051 (PI Thomas Wisniewski, MD), P30 AG013854 (PI Robert Vassar, PhD), P30 AG008017 (PI Jeffrey Kaye, MD), P30 AG010161 (PI David Bennett, MD), P50 AG047366 (PI Victor Henderson, MD, MS), P30 AG010129 (PI Charles DeCarli, MD), P50 AG016573 (PI Frank LaFerla, PhD), P30 AG062429-01(PI James Brewer, MD, PhD), P50 AG023501 (PI Bruce Miller, MD), P30 AG035982 (PI Russell Swerdlow, MD), P30 AG028383 (PI Linda Van Eldik, PhD), P30 AG053760 (PI Henry Paulson, MD, PhD), P30 AG010124 (PI John Trojanowski, MD, PhD), P50 AG005133 (PI Oscar Lopez, MD), P50 AG005142 (PI Helena Chui, MD), P30 AG012300 (PI Roger Rosenberg, MD), P30 AG049638 (PI Suzanne Craft, PhD), P50 AG005136 (PI Thomas Grabowski, MD), P30 AG062715-01 (PI Sanjay Asthana, MD, FRCP), P50 AG005681 (PI John Morris, MD), P50 AG047270 (PI Stephen Strittmatter, MD, PhD).

I dedicate this entire work to my academic and dissertation advisor Michael Kosorok. Dr. Kosorok, I always remember what you told me about how I should take on new tasks that would make me a little bit nervous because that's the level of challenge that would keep me learning and growing. You are my research mentor and my life mentor who have given me so much support and confidence than anyone else. I am forever grateful to be your student. I acknowledge the following contributors to this research: Chapter 1 collaborators (Richard Loeser, Amanda Nelson, Rebecca Cleveland, Daniel Beavers, Todd Schwartz, Liubov Arbeeva, Carolina Alvarez, Leigh Callahan, and Stephen Messier), Chapter 2 collaborators (Xin Zhou and Kinh Truong), Chapter 3 collaborators (William Stoudemire, Marianne Muhlebach, Lisa Saiman, and Juyan Zhou), NACC research scientists (Jessica Culhane and Merilee Teylan), my dissertation committee members (Donglin Zeng and Jianwen Cai), and all previous and current members of the Kosorok Lab (especially Sebastian Hidalgo, Jingxiang "Sean" Chen, Jon Hibbard, Daniel Luckett, Michael Lawson, Crystal Nguyen, Owen Leete, Xinyi Li, Arkopal Choudhury, Hunyong Cho, and Teeranan "Ben" Pokaprakarn, Nikki Freeman, and John Sperger). You all helped me in many ways and I thank you for your wisdom, financial and moral support. In addition, I acknowledge other close relationships I made at UNC, NCSU, and the Triangle Curling Club (including but not limited to Richard Zink, Busola Sanusi, Laura Zhou, Sean McCabe, Pedro Baldoni, Shaina Mitchell, Jipcy Amador, Eric Van Buren, Scott Van Buren, Anqi Zhu, Jon Rosen, Jiawei Xu, Yue Wang, Ting Wang, Xingjian Yu, Alex Karsten, Antonio LoPiano, Betsy Seagroves, and Melissa Hopgood). These are people who laughed with me, sighed with me, gave me a thumbs up, and gave me a pat on the back during all the ups and downs of graduate school. Our relationships meant a lot to me and made Chapel Hill my second home. Last but not least, to my parents, the continuous help I received from you was tremendous and I am grateful for your selfless support through my toughest and most glorious times in the past six years. I hope I have made you proud.

Keep smiling and laughing, Phoebe. The world is yet to be explored.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AOL	Augmented Outcome-weighted Learning
CDF	Cumulative Distribution Function
CF	Cystic Fibrosis
CFFPR	Cystic Fibrosis Foundation Patient Registry
CI	Confidence Interval
CV	Cross Validation
DDROWL	Deep Doubly Robust Outcome Weighted Learning
DNN	Deep Neural Network
DKL	Deep Kernel Learning
FFNN	Feedforward Neural Network
IPW	Inverse Probability Weighting
ITR	Individualized Treatment Rule
JK	Jackknife
KOA	Knee Osteroarthritis
LASSO	Least Absolute Shrinkage and Selection Operator
MI	Multiple Imputation
MLE	Maximum Likelihood Estimation
NACC	National Alzheimer's Coordinating Center
OWL	Outcome Weighted Learning
PDF	Probability Density Function
RF	Random Forest
RLT	Reinforcement Learning Trees
RWL	Residual Weighted Learning
SD	Standard Deviation
SVM	Support Vector Machine

CHAPTER 1: INTRODUCTION

With the rapid developments in science and technology, the sheer amount of data generated in healthcare and health-related research has been both empowering and overwhelming. Machine learning tools are able to absorb more and learn better about health policy and behaviors because they are no longer constrained with homogeneity and small sample sizes. Meanwhile, traditionally well-established methods have been challenged because it is difficult to accommodate for the increasing complexity of data, such as source, structure, size, and type. Assigning the same treatment to everybody is often not effective, and there has been a higher and higher demand for new, more powerful analytical methods to target on the well-being of each individual (rather than a population as a whole) and personalize treatments better without strong domain knowledge. This, however, does not mean that we are negating the traditional treatments; instead, we are looking for the treatments (traditional or innovative) that best fit each patient. It is this "big data" era we are living in that offers such opportunities for this precision medicine concept to thrive and sustain. We aim to develop data-driven and generalizable machine learning approaches with weak or minimal assumptions under the framework of precision medicine. More specifically, we will examine the following three research topics.

First, comparison and selection of the best individualized treatment regime for clinical trial patients are key tasks in precision medicine. To improve model selection with well-defined uncertainty estimation and statistical testing, we propose to apply the jackknife method (also known as the leave-one out cross validation) to more than a dozen of machine learning models. The consistency proof of jackknife estimators was established under minimal assumptions. The variance of jackknife estimators is composed of a newly defined, influence-function inspired

value and we show evidence of asymptotic normality through simulations. The usage of jackknife estimators is fully studied in a clinical trial called IDEA for knee osteoarthritis.

The existing machine learning approaches have proven their ability in optimal individualized treatment estimation for small sample sizes of observational studies and clinical trials. There is yet room for collaboration between precision medicine and deep learning, the two increasingly popular areas in public health application. The non-parametric hierarchical architecture in deep learning increases the flexibility of existing learning regimes and expands the influence of precision medicine to large, high-dimensional data. We show how to get the best of both worlds between deep learning and augmented outcome weighted learning, a recent method that thrives on doubly robustness and residuals.

Finally, we arrive at the crossroad of precision medicine and survival analysis, where we are interested in monitoring survival time while taking into account multi-level hierarchical structure and recurrent events in right-censored data. we aim to extend a mixed-effect Andersen-Gill model (also known as a frailty model) with risk adjustment and provide variability estimates of the survival time. This method will be particularly useful for infection prevention and control, where health programs or hospitals want to gain knowledge on their infection rates in advance and be able to take proactive actions.

We organize the remainder of this document as follows. We proceed with the aforementioned three precision medicine research topics in chapters 2, 3, and 4 respectively. Chapter 5 discusses future directions of the three areas, followed by technical details and references of the entire document.

CHAPTER 2: A PRECISION MEDICINE APPROACH TO DEVELOP AND INTERNALLY VALIDATE OPTIMAL TREATMENTS FOR OVERWEIGHT AND OBESE SENIOR ADULTS WITH KNEE OSTEOARTHRITIS

2.1 Introduction

Knee osteoarthritis (KOA) is one of the most common forms of arthritis worldwide accounting for a significant proportion of pain and disability in the adult population (Cross et al., 2014). Known risk factors for KOA include older age (especially 55 years and older), increased body weight, previous joint injury, and genetics (Vina and Kwoh, 2018). Clinical trials in overweight and obese adults with symptomatic KOA have shown weight loss and exercise interventions can improve pain and function, although not all individuals achieve a similar amount of benefit (Messier et al., 2013; Nelson et al., 2014; Messier et al., 2018). Overweight and obese patients with KOA will want to know if they need to diet and exercise or whether exercise or diet alone would be sufficient. Likewise, clinicians still have limited knowledge and will need additional insight into which specific therapies are most likely to benefit particular patients in a given situation. To address these questions, we utilized machine learning tools to develop and internally validate the optimal precision medicine treatment regime from OA clinical trial data and simulations that would maximize expected clinical outcomes.

A precision medicine approach incorporates patient heterogeneity to inform clinical decisions. In many routine clinical settings, it is common for all patients with a given condition to receive the same treatment, despite the fact that treatment effectiveness differs by individual. Precision medicine is able to leverage the abundant patient information collected in the clinical setting (e.g., demographic and social economic characteristics, clinical history and physical exam findings, lab results, and in some cases even medical imaging and genetic traits) in the decision making process of who should receive what treatment at what time. This is done through a

function called a decision rule which maps individual characteristics to a recommended intervention. The decision rules are estimated by machine learning models, which have been recommended to aid clinical decision-making (Jamshidi et al., 2018). Although many decision rules could potentially map patient information to a treatment, an optimal treatment rule (or optimal treatment regime) can be identified that maximizes the expected clinical outcomes of interest, thus serving to provide the optimal treatment recommendation to a patient population of interest (Kosorok and Laber, 2018).

We use the precision medicine approach to develop and internally validate the optimal treatment regimen for making exercise and weight loss recommendations for individuals with symptomatic KOA, utilizing data collected during the Intensive Diet and Exercise for Arthritis (IDEA) trial. The IDEA trial compared 3 interventions over 18 months: 1) E - exercise alone (considered standard of care as a control group), 2) D - diet-induced weight loss with the goal of a 10% reduction in body weight, and 3) D+E - diet plus exercise, in overweight and obese adults with symptomatic radiographic KOA (Messier et al., 2013). IDEA results showed that, compared to exercise alone, participants randomized to D and D+E groups had greater weight loss and greater reductions in interleukin-6 (IL-6) at 18th month. The other primary outcome, knee compressive force, was significantly reduced in the D group but not the D+E group. Self-reported pain and function scores improved more in the D+E group. Not unexpectedly, there was a variable response to each intervention among the study participants and those who lost more weight demonstrated more improvements in function, pain, knee compressive force and IL-6 levels (Messier et al., 2013, 2018), independent of group assignment. We hypothesized that one or more of these variables could be used to determine an optimal treatment regime that would inform which individuals would benefit the most (in terms of specific outcomes) from a given intervention when compared to assigning all individuals to just one of the three interventions.

We emphasize the following contributions: i) Exercise and weight loss, alone or together, can benefit individuals with KOA although the response varies suggesting there may be subgroups who would achieve more benefit from a specific intervention; ii) This study is the first to apply

precision medicine-based machine learning approaches to clinical trial data in KOA; iii) These approaches identified subgroups of patients for whom a precision medicine decision rule would lead to improved outcomes over assignment of all individuals to the combined exercise and weight loss intervention.

2.2 Patients and Methods

2.2.1 Patient Data

IDEA was an assessor-blinded, single-center randomized trial conducted during 2006 - 2011 at Wake Forest University and Wake Forest School of Medicine. Details of the study design and the results for the main outcomes have been previously published (Messier et al., 2009, 2013). In brief, IDEA included 454 individuals with mild or moderate radiographic OA in one or both knees. They were ambulatory, community-dwelling persons aged 55 or older (mean 66 ± 6 (SD) years) with a body mass index (BMI) between 27 and 41 (mean 33.6 ± 3.7 (SD) kg/m^2), a sedentary lifestyle, and pain on most days due to KOA. Measures (76 covariates) relevant to participant demographics, standard sociodemographic factors, physical performance measures, KOA, and its effects on pain and function were collected at baseline with selected outcome measures also obtained at 6 months (not used in this study) and 18 months.

2.2.2 Preprocessing

The initial precision medicine analysis used five of the seven clinical outcomes at 18 months that would be easiest for a clinician to obtain in a practice setting: weight loss since baseline, WOMAC (Western Ontario and McMaster Universities OA index) pain, function and stiffness scores, and the SF-36 physical component score (PCS). Of the 454 participants who entered the trial, 399 completed the 18-month study. Because observed outcomes provide important information that drives the decision rule, we excluded participants missing one or more outcomes at 18 months leaving 343 participants (Input data 1, Table 2.1). Dimension reduction was applied to control overfitting and extract the important features of the original 76 covariates at baseline, from which 15 covariates (Table 2.1) were chosen based on three criteria: 1) <15% missing data, 2) clinically important and potentially measurable in clinical practice, and 3) statistically

important as determined by the variable importance measure from random forests (RF) (Breiman, 2001). Selected covariates were then imputed via a non-parametric random forest method called missForest (Stekhoven and Bühlmann, 2011), which does not require assumptions about the data distribution, utilizes out-of-bag imputation error estimates to avoid cross validation (CV), and can be applied to high-dimensional mixed-type data of unequal scales. Lastly, categorical variables were conformed and dichotomized, and all outcomes were transformed such that higher values represented improvements in the outcomes. All baseline covariates were standardized to the standard normal distribution to avoid artifacts from differences in scaling, due to the potential for varying scales to create misleading values of coefficients in models such as penalized regression. Missing data were investigated in the original IDEA study with multiple imputation analysis, which "revealed minimal differences from [the] original intention-to-treat analysis" (Messier et al., 2013). Further details on data cleaning, dimension reduction, and imputation are provided in the Supplemental Materials.

A second analysis used all seven outcomes including the two mechanistic outcomes (knee compressive force and plasma IL-6) at 18 months. This analysis was considered so as not to overlook any potentially valuable information from the two mechanistic outcomes, although they are not patient reported or as easily obtainable in clinical practice as the other outcomes. We cleaned and imputed the second input dataset (Input Data 2, Table 2.1) and applied the same preprocessing procedure as described above. Values for IL-6 at 18 months were log-transformed in the analyses due to right-skewness and exponentiated back to original values during testing and optimal estimation.

2.2.3 Training Process and Performance

After the input data were cleaned and preprocessed, a total of 24 machine learning models were implemented (Table 2.2). They were selected specifically to suit the IDEA data, which represent a single-decision setting. The candidate models can be summarized in the following categories: penalized linear regression (M1-4), ensemble learning of decision trees (M5-7), tree-based dynamic treatment regime (DTR) (M8-20), support vector machine-based learning

Table 2.1: Description of input datasets used in the analyses

	Input Data 1	Input Data 2
Participants (n)	343	293
Outcomes at 18m (n)	5 Physical component score (PCS), Weight loss since baseline, WOMAC pain score, WOMAC function score, WOMAC stiffness score	7 Compressive force, Plasma IL-6, Physical component score (PCS), Weight loss since baseline, WOMAC pain score, WOMAC function score, WOMAC stiffness score
Baseline Covariates (n)	15 ABC, BMI, walking distance, WOMAC function score, gait, heart attack, hip circumference, WOMAC pain score, PCS baseline, average walking speed, WOMAC stiffness score, waist circumference, whole body lean DXA, whole body fat DXA, weight, randomization group	17 ABC, BMI, walking distance, WOMAC function score, gait, heart attack, hip circumference, IL-6, WOMAC pain score, PCS baseline, average walking speed, WOMAC stiffness score, waist circumference, whole body lean DXA, whole body fat DXA, whole body percentage fat DXA, weight, randomization group

Abbreviations: ABC - Activities-specific balance confidence scale, BMI – Body mass index, DXA – Dual-energy X-ray absorptiometry, IL- Interleukin, PCS – Physical component score, WOMAC - Western Ontario and McMaster Universities OA index

(M21-23), and Bayesian model (M24). Our selection of models covered both conventional and emerging concepts in the statistical literature; the rational for each model choice is included in the Supplemental Materials. In addition to the precision medicine models, we investigated three zero-order models (ZOMs) which assigned just one of the treatments (E, D, D+E) to all participants (M25-27). ZOMs are named after zero-order processes which are fixed decision rules that do not change by individual.

Twenty-four machine learning models and the 3 ZOMs, for a total of 27 models, provided estimated individualized treatment rules (ITRs), which were compared based on estimated value functions separately for each outcome. The value function is a scalar measure of performance for each ITR that evaluates the expectation of an outcome if future patients followed the estimated decision rule that is derived from training input data. A higher value function indicates a higher

Table 2.2: Listing of precision medicine based machine learning models and zero-order models used in the analyses

Model	Parameters	M#
Penalized regression (Tibshirani, 1996) (Zou and Hastie, 2005)	Lasso, $\alpha = 1$	1
	Ridge, $\alpha = 0$	2
	Elastic Net, $\alpha = 0.5$	3
Kernel ridge regression (KRR) (Zhang et al., 2018)	Gaussian kernel	4
Random forests (RF) Number of trees = 500 (Breiman, 2001)	Rules based on each individual outcome	5
	Rules based on a weighted outcome of weight loss, pain, and function	6
Reinforcement learning trees (RLT) (Zhu et al., 2015)	Number of trees = 50	7
List-based dynamic treatment regime (DTR) (Zhang et al., 2018)	Q-functions estimated by KRR Number of nodes = 2, 3, 5, 10	8,9, 10,11
	Q-functions estimated by RF Number of nodes = 2, 3, 5, 10	12,13, 14,15
	Q-functions estimated by Super Learning Number of nodes = 2, 3, 5, 10	16,17, 18,19
	Q-functions estimated by elastic net Number of nodes = 10	20
Residual weighted learning (RWL) (Zhou et al., 2017)	Linear kernel	21
	Polynomial kernel with 2nd order	22
	Polynomial kernel with 3rd order	23
Bayesian additive regression trees (BART) (Chipman et al., 2010)	Number of trees = 500 Number of draws = 5500 (with 500 burn-ins)	24
Zero-order (ZOM)	Always assign to E	25
	Always assign to D	26
	Always assign to $D + E$	27

quality of the estimated ITR and more benefit to future patients in terms of that outcome. Hence, a learning model that maximizes the value estimate with small variation would be preferred. Mathematical definitions of the true and estimated value functions can be found in subsection Value Function of Supplemental Materials. The value estimates are usually derived from model evaluation techniques such as CV. A simple CV procedure that splits the sample data into one training and one testing set would usually generate biased, ungeneralizable results. An alternative is K-fold CV, which refers to training ITRs on $K - 1$ of the randomly divided folds and testing

the performance and generalization of ITRs on the one remaining fold. This is repeated until every fold has been tested.

We used the jackknife method to estimate the bias and standard error of the estimated value function used for model selection. The jackknife is a a leave-one out cross validation (LOOCV) or n-fold CV method where each individual serves as a fold so the training sample leaves one observation out at a time (Efron and Tibshirani, 1994). We chose the jackknife estimator because it requires weak assumptions (i.e., unrestricted shape of the probability distribution as long as the observations are independently and identically distributed) and is approximately unbiased for the true prediction error (Friedman et al., 2001). In addition, stratified 10-fold CVs were also performed to check the stability of jackknife value function estimates and compare test results. Such validation methods (jackknife and CV) as well as simulation experiments accommodate for internal validation to prevent overfitting. More details on the jackknife and CV estimators as well as simulations on their theoretical properties may be found in subsections: The Jackknife and Stratified Cross Validation in Supplemental Materials.

2.2.4 Testing Process and Model Selection

We applied all 27 candidate models to each outcome for training, recorded their estimated decision rules, and compared the jackknife value function estimators and their standard errors in the testing process. For each outcome separately, the optimal precision medicine model (PMM) was the model with the highest estimated value function with smaller standard error among the 24 machine learning models, i.e., its decision rule would bring the highest reward to future patients with small uncertainty in the value estimate. We found that, in general, standard errors of the value estimators on the same outcome did not differ substantially across candidate models, so we focused on the value estimators. Similarly, the optimal ZOM is the model with highest value estimate and relatively small standard error from the three ZOMs. We performed a two-sample Z-test to compare the optimal PMM with the optimal ZOM (details in Model Selection in Supplemental Materials). Once outcomes with statistically significant results were found, we estimated the decision rule of the optimal PMM trained on the entire dataset (without

jackknife validation), which served as the final data-driven, precision medicine-based treatment recommendation.

2.2.5 Multiple Outcomes

To account for potential correlations among outcomes, we derived optimal treatment rules based on a weighted sum across multiple outcomes. A minimax algorithm was proposed to optimize data-driven weights for the three outcomes of greatest interest: weight loss since baseline, WOMAC pain sub score, and WOMAC function sub score at 18 months. To reduce computational time, we used a coarse-to-fine grid search with RF models to determine the weight combination that maximized the lowest jackknife value function estimates among the three outcomes, hence the name "minimax" (details in subsection Multiple Outcomes in Supplemental Materials). The selected minimax weights were then used to create a composite outcome, i.e., the weighted sum of weight loss, pain, and function score, to train a RF model (M6) and estimate the optimal treatment rule. This contrasts with the other models discussed above where the PM treatment rule was trained on a single outcome but all models were tested on a single outcome.

All analyses were performed with R version 3.4.4 (R Core Team, 2019). In formation on specific packages can be found in subsection Choice of Models in Supplementary Materials. As these analyses were exploratory in nature, the significance level was relaxed to be 0.1 throughout this paper. A complete outline of the entire precision medicine approach is visualized in Figure 2.1.

2.3 Results

Descriptive characteristics for the two input datasets, as well as the full dataset with all 399 participants who finished 18 months of follow-up, are summarized in Table 2.3. In general, baseline characteristics of participants with available data were evenly distributed across the three intervention groups, as would be expected from a randomized clinical trial. There were no differences in selected baseline characteristics for participants with or without missing outcome data.

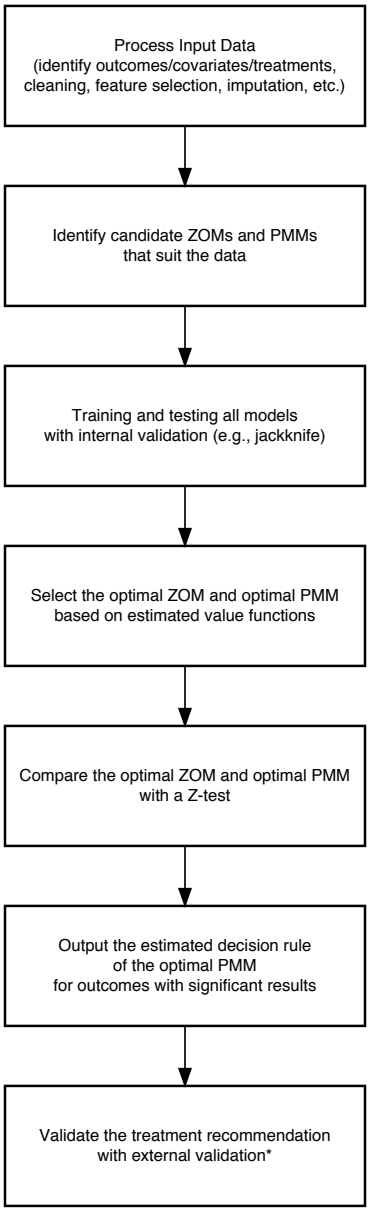

Figure 2.1: Flowchart of the proposed precision medicine approach. An asterisk means the step was not performed in this analysis due to unavailable data but is highly recommended for a more complete analysis.

Table 2.3: Descriptive characteristics of baseline input datasets

	Count (%) or Mean (SD)		
	Input Data 1 (n = 343)	Input Data 2 (n = 293)	Overall (n = 399)
Randomization Group			
Exercise (E)	111 (32%)	99 (34%)	135 (34%)
Diet (D)	116 (34%)	95 (32%)	129 (32%)
Diet and Exercise (D+E)	116 (34%)	99 (34%)	136 (34%)
Age in years	65.5 (6.1)	65.9 (6.2)	65.9 (6.2)
Weight in kg	92.1 (14.5)	92.0 (14.8)	92.4 (14.6)
BMI in kg/m^2	33.4 (3.8)	33.3 (3.8)	33.5 (3.7)
Female	251 (73%)	211 (72%)	291 (73%)
Race			
Black	57 (17%)	47 (16%)	68 (17%)
White	286 (83%)	246 (84%)	332 (83%)
Education			
High School	100 (29%)	84 (29%)	117 (29%)
College	164 (48%)	142 (48%)	194 (49%)
Post College	77 (22%)	65 (22%)	87 (22%)
Missing	2 (1%)	2 (1%)	2 (< 1%)
Smoking			
Never	196 (57%)	169 (58%)	229 (57%)
Former	132 (38%)	112 (38%)	153 (38%)
Current	10 (3%)	8 (3%)	12 (3%)
Missing	5 (1%)	4 (1%)	6 (2%)
Alcohol			
Never	66 (19%)	60 (20%)	77 (19%)
Former	69 (20%)	51 (17%)	83 (21%)
Current	199 (58%)	174 (59%)	229 (57%)
Missing	9 (3%)	8 (3%)	11 (3%)
Marital Status			
Presently married or in a marriage-like relationship	239 (70%)	208 (71%)	276 (69%)
Never married, divorced, separated, widowed	103 (30%)	85 (29%)	123 (31%)
Missing	1 (<0.5%)	-	1 (<0.5%)

Table 2.4: Comparison between the optimal Zero Order Model (ZOM) and the optimal Precision Medicine Model (PMM) for each outcome

Dataset	Outcomes (18 mo.)	Optimal ZOM	Optimal PMM[1]	Estimated Value (Optimal PMM)	Relative Increment[2]	p-value[3]
Input Data 1 (n = 343)	Physical component score	D+E	M10	45.47	0.10	0.88
	Weight loss since baseline	**D+E**	**M10**	**11.21**	**1.45**	**0.01**[4]
	WOMAC pain score	D+E	M15	3.25	0.02	0.38
	WOMAC function score	D+E	M1, M12	12.63	0.00	1.00
	WOMAC stiffness score	D+E	M19	2.12	0.03	0.86
Input Data 2 (n = 293)	Compress force	D	M9	2336.21	21.74	0.73
	Plasma IL-6	**D**	**M12**	**2.29**	**0.26**	**0.09**
	Physical component score	D+E	M7	46.46	0.96	0.24
	Weight loss since baseline	**D+E**	**M7**	**11.76**	**1.31**	**0.06**
	WOMAC pain score	D+E	M9	3.24	0.08	0.59
	WOMAC function score	D+E	M1, M12-15	12.58	0.00	1.00
	WOMAC stiffness score	D+E	M8	2.08	0.04	0.31

[1] This table focuses on modeling on individual outcomes. PM model selection was among 23 models in Table 2.2, excluding M6 whose results are presented in Table 2.5. M1-4 are penalized regression models. M5 and M7 are random forests and reinforcement learning trees. M8-11, M12-15, and M16-19 are respectively kernel ridge regression, random forests, and super learning list-based DTRs with 2, 3, 5, 10 nodes. M20 is elastic net list-based DTR with 10 nodes. M21-23 are residual weighted learning of different kernels, and M24 is Bayesian regression model.

[2] Relative increment is the jackknife estimated value function of the optimal PMM minus the jackknife estimated value function of the optimal ZOM; the increment in future expected outcome based on the optimal PMM relative to the optimal ZOM.

[3] The p-value comes from the Z-test (details found in Supplementary Materials).

[4] Significant results are boldfaced.

Table 2.5: Comparison between the optimal Zero Order Model (ZOM) and the Random Forrest model for weighted sum of selected outcomes (M6)

Dataset	Outcomes (18 mo.)	Optimal ZOM	Optimal PMM	Estimated Value (Optimal PMM)	Relative Increment[1]	p-value[2]
Input Data 1 (n = 343)	Physical component score	D+E	M6	45.43	0.05	0.86
	Weight loss since baseline	**D+E**	**M6**	**10.10**	**0.34**	**0.05**[3]
	WOMAC pain score	D+E	M6	3.24	0.03	0.71
	WOMAC function score	D+E	M6	12.42	0.21	0.54
	WOMAC stiffness score	D+E	M6	2.11	0.03	0.44
Input Data 2 (n = 293)	Compress force	D	M6	2446.46	-88.50	0.41
	Plasma IL-6	D	M6	2.55	0.01	0.98
	Physical component score	D+E	M6	45.70	0.20	0.58
	Weight loss since baseline	D+E	M6	10.76	0.32	0.29
	WOMAC pain score	D+E	M6	3.23	0.10	0.47
	WOMAC function score	D+E	M6	12.26	0.32	0.47
	WOMAC stiffness score	D+E	M6	2.03	0.09	0.13

[1] Relative increment is the jackknife estimated value function of the M6 optimal PMM minus the jackknife estimated value function of the D+E; the increment in future expected outcome based on M6 relative to the optimal ZOM.
[2] The p-value comes from the Z-test (details found in Supplementary Materials).
[3] Significant results are boldfaced.

2.3.1 The optimal zero-order model (ZOM)

Considering the three ZOMs, we found that the optimal ZOM model assigned every individual to D+E for all 5 clinical outcomes: weight loss since baseline, WOMAC pain, function and stiffness scores, and PCS at 18 months (Table 2.4). Treatment D was the optimal ZOM for the two mechanistic outcomes: knee compressive force and plasma IL-6 level at 18 months.

2.3.2 The optimal precision medicine model (PMM)

The RF model with minimax weights (M6) was the optimal PMM for each of the three WOMAC sub scores regardless of input data (Table 2.5). For the rest of the outcomes (Table 2.4), list-based models (M9-13) and RLT (M7) were optimal among the 24 PMMs.

2.3.3 The optimal ZOM vs. the optimal PMM

The relative increments between the estimated value functions of the optimal PMM and those of the optimal ZOM were positive (Table 2.4), indicating that the optimal PMM outperformed the optimal ZOM for all outcomes. According to the Z-test, such improvement of the optimal PMMs compared to the optimal ZOMs was significant both for weight loss since baseline and for IL-6 levels (Table 2.4). We investigated these two outcomes further.

For weight loss between baseline and 18 months, the application of our PM approach showed that future patients are estimated to lose 11.2kg of weight on average between baseline and 18 months, according to the optimal PMM (list-based DTR with at most 5 nodes). This is an average of 1.4kg more weight loss than if all patients had received D+E, the optimal ZOM (significant improvement, p = 0.01). Trained on input data 1 as a whole, the estimated optimal decision regime for weight loss would recommend intervention D+E to patients who meet either of the following two conditions:

1.1) If, at baseline, weight does not exceed 109.35 kg *and* waist circumference is above 90.25 cm,

1.2) If, at baseline, weight is greater than 109.35 kg *or* waist circumference does not exceed 90.25 cm, *and* they have reported a prior heart attack.

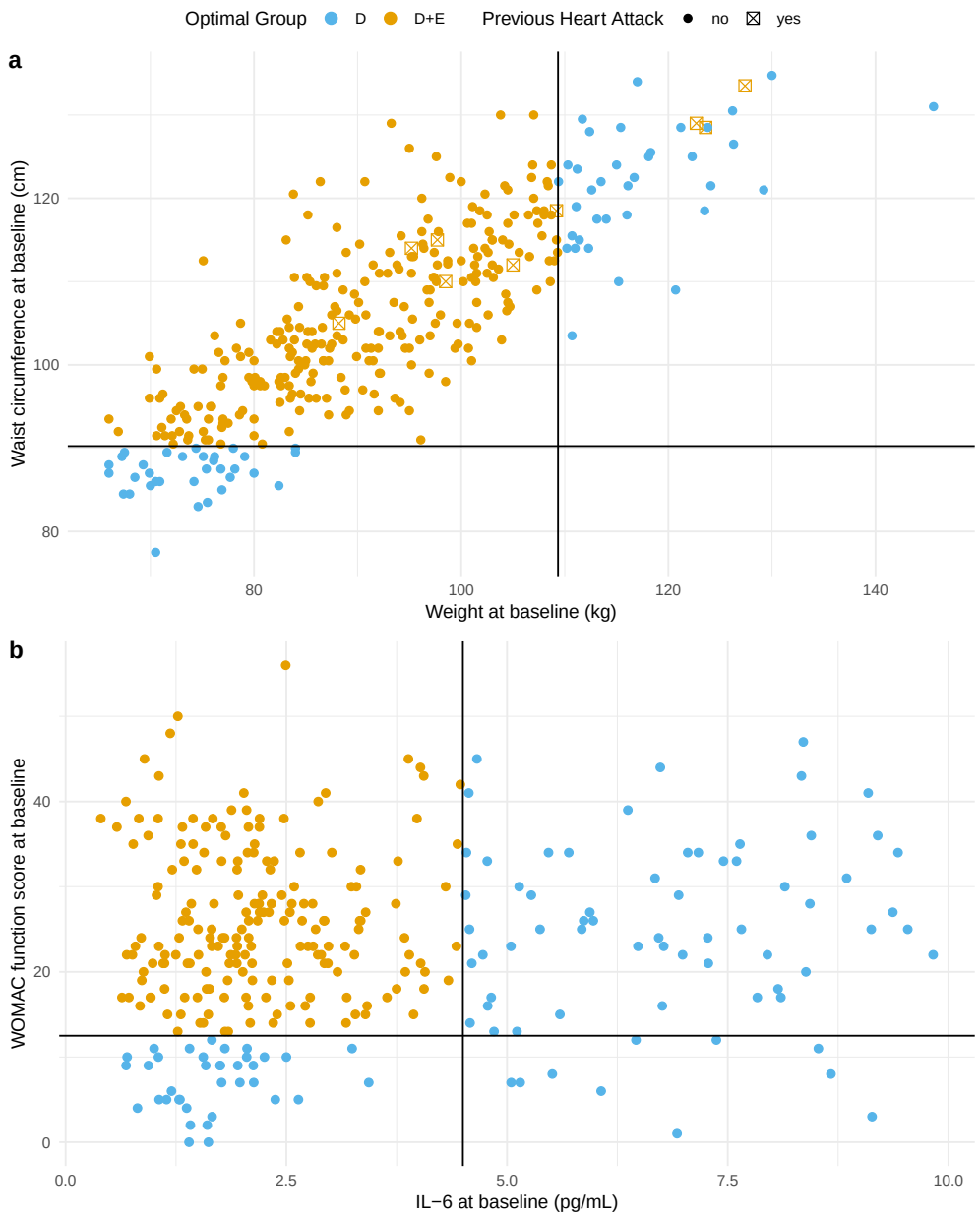

Figure 2.2: Visualization of the estimated optimal decision regimes for outcomes (a) weight loss since baseline and (b) IL-6 at 18 months. Scatter plots of data for each individual are color-coded to indicate the optimal treatment group assignment of all individuals in the input data (input data 1 for outcome weight loss since baseline in panel (a) and input data 2 for outcome IL-6 at 18 months in panel (b)). Blue indicates individuals who would be assigned to diet only (D) and orange to those assigned to diet plus exercise (D+E). For weight loss since baseline (a), previous heart attack (yes or no) also determined group assignment and is shown as a checked box for those individuals who met that criteria. The horizontal and vertical reference lines indicate the cut-off levels for the variables shown on the horizontal and vertical axis, respectively, that determined group assignment.

If neither of these conditions are met, the recommendation is for treatment D. The visualization of this optimal treatment rule can be found in Figure 2.2a.

For IL-6, the application of our PM approach showed that future patients are estimated to decrease IL-6 to 2.29 pg/mL on average at 18 months, according to the optimal PMM (list-based DTR with at most 2 nodes). This is an average of 0.26 pg/mL more reduction than if all patients had received D, the optimal ZOM (significant improvement, p = 0.09). Trained on input data 2, the estimated optimal treatment rule for IL-6 assigned D+E to patients who meet the following condition:

2.1) If, at baseline, IL-6 does not exceed 4.5 pg/mL *and* WOMAC function score is more than 12.5.

If this condition is not met, patients would be assigned to treatment D (Figure 2.2b). As evidence of stability, we found similar patterns and similar conclusions for weight loss and IL-6 using the stratified 10-fold CV method (see Stratified Cross Validation in Supplementary Materials).

2.3.4 Multiple Outcomes

The outcomes were positively correlated to some extent. The highest correlations were found among WOMAC scores (pain, function, stiffness) and PCS scores (Pearson correlation coefficients ranged from 0.52 to 0.87). For input data 1, the minimax rule selected 0.1, 0.6, 0.3 as data-driven weights for the three selected outcomes (weight loss, pain, and function, respectively) and 0, 0.32, and 0.68 for input data 2. We did not scale the outcomes but allowed the weights to adjust for different scales in the outcomes. Similar to the Z-test comparison in the previous subsection (Table 2.4), we compared the optimal ZOM with one PMM: the RF model trained on the weighted composite outcome (M6) (Table 2.5). There was evidence of significant improvements of M6 relative to the optimal ZOM (D+E) for weight loss since baseline for input data 1 (p = 0.05). Although not statistically significant, the remaining outcomes (except compressive force) also expressed positive relative improvement in both input datasets. In particular, M6 outperformed other PMMS in terms of the estimated value function for the

three WOMAC scores, but not for outcomes uncorrelated to the three weighed outcomes, which are compressive force and IL-6.

2.4 Discussion

In this paper, we investigated optimal treatment recommendations for older and overweight or obese individuals with KOA using precision medicine techniques and machine learning tools applied to data obtained from the IDEA trial. The individual treatment decisions obtained from our precision medicine approach are data-driven (requiring no strong assumptions), reproducible (with careful reporting of the analysis process) (Kosorok and Laber, 2018), and generalizable and extendable to other clinical settings (because of rich heterogeneity in the clinical input data).

The results of the optimal ZOM, where everyone would be assigned to a single intervention, match with those from the published IDEA trial (Messier et al., 2009, 2013). The assignment of patients to the D+E intervention would be expected to result in the optimal improvement in the majority of patients in the clinical outcomes of weight loss since baseline, WOMAC pain, function, and stiffness scores, as well as PCS and so should remain the recommendation of choice. In individuals where the primary goal is to reduce systemic inflammation as measured by plasma IL-6 levels and/or reduce the knee compressive force, D alone would be the treatment of choice.

The optimal treatment rules of the optimal PMMs suggested that not everyone benefits from D+E even though patients are expected to be assigned to this group based on the ZOM. Further improvements in weight loss could be obtained in certain patients selected by measures of high baseline weight (over 109.35 kg) or low waist circumference (90.25 cm or less) accompanied by lack of a previous heart attack that would result in assigning them to D rather than D+E. This would only be a consideration if weight loss alone was more important to the patient than the level of improvement in pain and function. We can only speculate why people of higher weight with lower waist circumference and no history of heart attack would benefit more from D than D+E. First, it is likely that following the suggested exercise program may be more difficult for patients with a height weight. Second, higher weight with lower waist circumference could be

seen in individuals who have more peripheral adiposity rather than central adiposity. In these cases, D could be more effective in losing weight. The finding that our results were modified by a history of a heart attack may be that the cardiac status of these individuals encourages optimal compliance and improves more with the combined D+E than D alone and this allows for greater activity levels resulting in greater weight loss.

The finding that the IL-6 outcome improves more with D than D+E in certain individuals is not easily explained. We noted that individuals with high baseline IL-6 levels (i.e. above 4.5 pg/mL) or those with low baseline function scores (12.5 or less in a range of 0 to 68) reduced their IL-6 more from diet only. Individuals whose IL-6 was not high but have poorer function are recommended to receive both diet and exercise. The decrease in IL-6 suggests less systemic inflammation but there is no solid evidence to suggest that exercise would modulate the reduction in IL-6 that occurs with dietary weight loss. Because all three groups received an intervention, the significant differences in outcomes noted among the groups at 18 months would be unlikely to be due to regression to the mean. Our findings that specific subgroups of individuals received more benefits from specific interventions argues against the premise that response was simply due to patient perception rather than to the intervention itself.

As for the multiple outcomes, comparison between Table 2.4 and Table 2.5 suggested that our minimax rule together with the coarse-to-fine grid search for parameter optimization can be a useful way to incorporate multiple outcomes, and combining correlated outcomes has the potential for bringing more benefits to patients than single outcomes. However, uncorrelated outcomes do not benefit from the composite outcome.

2.4.1 Limitations

Potential limitations of this study include the following. First, we were not able to use the information of about 100 of the trial participants due to missing outcome data. Although larger sample size might lead to higher power, our two input datasets remain representative of the overall data as Table 2.3 shows. Secondly, the analyses did not include intermediate follow-up data at 6 months. Although longitudinal analysis methods could be applied to the IDEA data, we

were more interested in the final improvements of each outcome between the start and the end of the trial and less on the intermediate progress. It is also unlikely for adding one more time point shortly after the trial would be influential as we expect it takes time for the interventions to take effect. Thirdly, there were some covariates with a large proportion of missing data excluded from the analysis. The majority of these were measures that would not be routinely collected in the clinical setting such as full-length lower extremity radiographs for alignment, computed tomography for abdominal and thigh fat, knee MRI, and isokinetic strength testing. Finally, our results are from a single clinical trial of patients with mild-to-moderate symptomatic KOA (Kellgren-Lawrence scores of 2-3) (Messier et al., 2013) and may not be generalizable to populations with more severe KOA.

2.5 Supplementary Materials

2.5.1 Data Cleaning

We performed extensive data cleaning for the baseline covariates in the raw IDEA trial data. First, marital status was originally classified into six categories and we combined them into two categories, where 1 stands for presently married and in a married-like relationship and 0 stands for never married, divorced/separated, and widowed. Second, three questions, "How many falls have resulted in injury, minimal medical attention, and hospitalized/bedridden?", had -1 values because the patient answered "no" to a previous question "Over the past 6 months, how many times you have fallen on the ground?". We converted all -1 values to 0's because the patient had not fallen in the past six months so no falls have resulted in the three conditions. Third, we reordered ordinal variables such as education, alcohol, and thinking of falling (i.e. "how often do you think about falling") so they were all ordered from never/low to current/high. Binary variables were converted from values of 1 or 2 to values of 0 or 1. Fourth, two variables, "number of falls resulted in hospitalization" and "pacemaker condition" were excluded from the cleaned data because they have no or only a few cases when almost all other patients answered "no". With too few cases, these would not spread evenly across training and validation sets, which could generate unreliable estimates and predictions. Lastly, we excluded follow up data at

6th and 12th month, kept baseline and 18th month data only, and reshaped the raw data from a long format to a wide format.

2.5.2 Dimension Reduction

The original IDEA analysis included 76 variables, all measured at baseline, including standard sociodemographic factors, anthropomorphic measures, measures related to co-morbidities, self-reported measures of pain and function, quality of life, physical performance measures including gait and strength analysis, and blood levels of inflammatory markers including IL-6 at baseline and C-reactive protein (CRP) at baseline. As part of the preprocessing, we performed dimensionality reduction to remove unimportant or highly correlated covariates to speed up computation and avoid multicollinearity. Among patients with complete outcome measures, we applied a random forests model to each outcome separately and acquired variable importance measures of the covariates. Random forests (RF) are a suitable dimension reduction tool because they can examine both the marginal and multivariate predictive performance of predictors. The categorical variables in our input data have two to six categories, which are comparable scales for the importance scores of RF to be valid. A variable was considered to be "statistically important" if it was among the top 10 most important variables for at least three of the outcomes that RF was modeled on, or the value of its scaled mean squared error (MSE) rate was at least 9. We observed that more predictors have low importance scores compared to those with high scores. Thus, our two criteria filtered out noisy, non-influential variables and maintained influential variables that are either important to many outcomes or extremely important to one single outcome. A variable was considered to be "clinically important" if it is patient-reported or a test result that could be obtained in medical settings (such as blood levels of IL-6 or DXA). Thus, we avoided using the variables generated by gait analysis since this would require patients to be assessed in a gait lab which would not be available in most medical settings. As a result, we included 15 covariates in input Data 1 and 17 baseline covariates in input Data 2 (Table 2.1) because they were missing less than 15% of the observations and were considered to be both clinically and statistically important.

2.5.3 Missing Data and Imputation

In addition to the multiple imputation analysis conducted in the original IDEA study, we looked into the missingness pattern in the preprocessed Input Data 2 (with 7 outcomes) from two perspectives. We first performed logistic regression modeling for raw X covariates on the missingness of each outcome (0/1), and none of the X covariates was found to be significant. We also calculated Spearman correlation coefficients between the original data and the corresponding indicator data of 0/1's (where 0 means the value is not missing, and 1 means it is missing). Mixing the covariates and outcomes together, we randomly sampled five variables at a time (to maintain the size of the square matrix) for the correlation matrix but repeated this random sampling 15 times. Most pairwise correlations between the original data and missingness indicator data were less than 0.2, except around one to six moderate correlations occasionally observed per sample, which was partially due to moderate correlations between the two original variables. After we confirmed that there were no obvious missing data problems, imputation was conducted on covariates missing in less than 15% of the participants because too many missing data points would lead to biases and potentially poorly imputed values. The imputation process was unsupervised (using only X-variables) because more biases would be introduced if outcomes were involved at this stage. The algorithm missForest suit our study because it did not require assumptions about the data distribution, can be applied to high-dimensional mixed-type data of unequal scales, utilized out-of-bag imputation error estimates to avoid CV, and outperformed other popular imputation methods such as multiple imputation by chained equations (MICE) (Stekhoven and Bühlmann, 2011). The R package "missForest" was used to run this imputation (Stekhoven and Bühlmann, 2011).

2.5.4 Choice of Models

Table 2.2 shows all the models we considered in the clinical data analysis. Penalized regression is a linear regression model that penalizes high-dimensional data in the model by shrinking the coefficients. This kind of model reduces variance of coefficients and automatically reduces dimensions while making predictions as a regular linear model. We chose three kinds

of penalty terms (lasso, ridge, and elastic net) that differ in terms of the amount of shrinkage determined by a penalty parameter: lasso (M1) forces coefficients to be exactly zero with an absolute value penalty and ridge (M2) forces the coefficients to be close to zero with a squared term penalty, whereas elastic net (M3) with parameter is an even mix of lasso and ridge. Kernel ridge regression (KRR) (M4) is a ridge regression method that directly computes kernel functions to make predictions and the Gaussian kernel extends the regression model to a more flexible, non-linear space (Paterek, 2007). We chose RF for individual outcomes (M5) and weighted multiple outcomes (M6) because RF is a common prediction method that reduces overfitting and variance significantly by aggregating a group of independent individual classifiers. RF also takes into account variable interactions sequentially with its tree structure without user-specification. RLT (M7) is similar to RF as they are both tree-based methods, but it has two attractive properties: RLTs can eliminate noise variables with a built-in muting procedure and implements reinforcement learning to select variables at each node that improve outcomes in the long run (Zhu et al., 2015).

Although RF usually generates good predictions, it is often considered a "black-box" because it lacks the interpretability when averaging over many individual trees. In contrast, list-based DTRs estimate optimal treatment regimens with a sequence of decision rules which can be easily interpreted as lists, a set of "if-then" statements (Zhang et al., 2018). The list-based models require predictions of the potential outcome under all treatment options first before generating interpretable lists. We embedded four models to make such predictions: KRR, RF, super learning, and elastic net with . Super learning (SL) is a recent semi-parametric ensemble method (Van der Laan et al., 2007; Polley and Van Der Laan, 2010) that learns the optimal decision rules by combining candidate learners with weights instead of selecting only one optimal model, and is applicable to dynamic treatment regimens with multiple time points. The candidate SL base learners are BART, elastic net, KRR, lasso, RF, RLT, and ridge. We implemented a super learning algorithm with simulated annealing as the meta learner that learns the weights of the seven base learners and optimizes the super learning model (Rashid et al.,

2019). Simulated annealing is a numerical optimization method that speeds up optimization by finding the approximate global optima (as opposed to the exact global optima) without getting stuck in local optima (Givens and Hoeting, 2005). Our implementation can be found on our GitHub repository (https://github.com/phoebejiang/pmoa). The default number of list nodes, which is the maximum number of "if-then" statements, is 10. The higher the number of nodes, the more complicated it will be for interpretation of the estimated list of statements, so we also considered simpler lists with at most 2, 3, and 5 nodes for list with KRR, SL, and RF (M8 - M19). The majority of the candidate models are list-based DTRs because of their interpretability and flexibility in accepting multiple treatments and different types of embedded models for outcome prediction. Note that a list with 1 node is equivalent to a ZOM because it always assigns patients to the same group.

Residual weighted learning (RWL) models use outcome residualization to improve on outcome-weighted learning (OWL), a policy learning model that optimizes clinical outcomes directly (Zhou et al., 2017; Zhao et al., 2012). Using residuals instead of the original outcomes allows RWL to handle various types of outcomes and any shifts in the outcomes. We chose three kernels to learn both linear and nonlinear decision rules: linear kernel (M21), 2nd order polynomial kernel (M22), and 3rd order polynomial kernels (M23). Gaussian kernels can be more flexible but were not considered due to the burden of extensive computation times and interpretational difficulties. The current RWL algorithm can only deal with two treatments, and we generalized the model to three treatments by performing a two-stage procedure: 1) First fit RWL to the control group E versus the combined D and D+E group; 2) Then fit RWL again to only patients whose optimal group was predicted to be in the D and D+E combined group to further distinguish the optimal group into D or D+E. Finally, Bayesian Additive Regression Trees (BART) (M24) (Chipman et al., 2010) is an ensemble method similar to the idea of tree-based methods, but each tree in the ensemble is regularized by a prior distribution, and predictions are made from resampling from the posterior distribution. We included this nonparametric model for its flexibility in accepting various data types and its direct inference on estimation precision.

The following R packages were used to run the models mentioned above: "glmnet" (Friedman et al., 2010) for penalized regression models (M1-M3, M20), "listdtr" for KRR and list-based models (M4, M8-20) (Zhang et al., 2018), "randomForest" (Liaw et al., 2002) for RF models (M5-M6), "RLT" (Zhu et al., 2015) for the RLT (M7) model, and "DynTxRegime" (Holloway et al., 2018) for RWL models (M21-23), and "BART" (Sparapani et al., 2016) for the Bayesian model (M24).

2.5.5 Value Function

The true value function is the expected reward of a potential outcome under ITR and is defined as

$$V_0(d) = E[Y^d] = E\left[Y\frac{1\{A = d(\boldsymbol{X})\}}{P(A|\boldsymbol{X})}\right] \tag{2.1}$$

where $\boldsymbol{X} \in \mathcal{X} \subseteq \mathbb{R}^p$ represents patient covariates, $A = \{-1, 1\} \subseteq \mathcal{A}$ is treatment group, $Y \in \mathcal{Y} \subseteq \mathbb{R}$ is the clinical outcome of interest, and d is the decision rule that maps patient information to a treatment (Qian and Murphy, 2011). The true value function is often estimated by

$$\widehat{V}(\hat{d}) = \frac{\sum_{i=1}^n Y_i 1\{A_i = \hat{d}(\boldsymbol{X}_i)\}/\hat{P}(A_i|\boldsymbol{X}_i)}{\sum_{i=1}^n 1\{A_i = \hat{d}(\boldsymbol{X}_i)\}/\hat{P}(A_i|\boldsymbol{X}_i)} \tag{2.2}$$

which can be deemed as a weighted combination of individual outcomes.

2.5.6 The Jackknife

The "leave-one-out" jackknife idea was first brought up by Quenouille (1949) and Tukey (1958) to reduce estimation bias, which later inspired the bootstrap resampling methods. Back in 1995, Kohavi et al. (1995) claimed that jackknife "has smaller (pessimistic) bias but larger variance than leave-more-out CV". This statement is influential, but rather broad and debatable. There has been much literature pertaining to the performance of jackknife estimators that is more specific and substantiated. In least square linear regressions, Burman (1989) showed that jackknife (also referred to as ordinary CV) estimates reached the lowest biases and variances among other k-fold cross-validated estimates for simulated samples of 12 and 24. A recent publication by Zhang and Yang (2015) in 2015 argued that the jackknife is "typically the best or

among the best for a fixed model or a very stable modeling procedures (such as BIC) in both bias and variance, or quite close to the best in mean squared error (MSE) for a more unstable procedure (such as AIC or even high-dimensional LASSO)".

The jackknife method was chosen because it requires weak assumptions, which are typical assumptions such as independent and identically distributed observations (thus can be more representative of future data). Given reasonable sample sizes like that in the main paper, the jackknife is relatively easy to implement because it loops through each patient once and only once without the need of repetitions. As for bias and variance trade-off, we took into account the correlation among the training sets when calculating variance estimates and test statistics of jackknife estimators, as shown below and in the next section. Furthermore, we show that our jackknife estimators retain algorithmic stability by comparing them with estimators obtained from stratified 10-fold CV.

Mathematically, the jackknife value estimator is defined as

$$\widehat{V}^{jk}\left(\hat{d}_n\right) = \frac{\sum_{i=1}^{n} U_i}{\sum_{i=1}^{n} W_i} \tag{2.3}$$

where $U_i = Y_i \frac{1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}}{\hat{P}(A_i|\boldsymbol{X}_i)}$ and $W_i = \frac{1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}}{\hat{P}(A_i|\boldsymbol{X}_i)}$, $\hat{d}_n^{(-i)}$ is the decision rule estimated from a training set of size n with the i-th observation left out, and $\hat{P}(A_i|\boldsymbol{X}_i)$ is the estimated propensity score of the testing set ith observation. Conceptually, $\widehat{V}^{jk}\left(\hat{d}_n\right)$ is similar to $\widehat{V}(\hat{d})$ with the decision rule estimated by the jackknife $\hat{d}_n^{(-i)}$. For our IDEA trial data, the propensity score is known and is simply the proportion of being in each of the three treatments since treatment and covariates are independent for randomized trials. For non-randomized studies, propensity scores need to be estimated by methods such as logistic regression.

Let $R_i^{jk} = \frac{1}{\bar{W}_n} U_i - \frac{\bar{U}_n}{\bar{W}_n^2} W_i$ be a bias-corrected, influence function-inspired form of value function with $\bar{U}_n = \sum_{i=1}^{n} U_i$ and $\bar{W}_n = \sum_{i=1}^{n} W_i$ (see Appendix A for derivation). This is bias-corrected because $\sum_{i=1}^{n} R_i^{jk} = 0$. We defined the estimated variance of the jackknife value

estimator as

$$\widehat{\text{Var}}\left[\hat{V}^{jk}\left(\hat{d}_n\right)\right] = \frac{1}{n(n-1)}\sum_{i=1}^{n} R_i^{jk\,2} \tag{2.4}$$

We adjusted the summation by $n(n-1)$ because of $n-1$ degrees of freedom in the summation and correlation among the n training sets. The standard error of the value estimator is thus $\widehat{\text{SE}} = \sqrt{\widehat{\text{Var}}(\hat{V}^{jk})}$.

2.5.7 Model Selection

We performed a Z-test to compare the optimal PMM with the optimal ZOM. The test results are used to inform us of whether there is a strong precision medicine effect and whether or not ZOMs are always the optimal choice. Let $\widehat{V}^{jk}(\hat{d}_{\text{PMM}})$ be the jackknife estimator of the value function of the selected optimal PMM and $\widehat{V}^{jk}(\hat{d}_{\text{ZOM}})$ be the jackknife estimator of the value function of the optimal zero-order model. The null hypothesis was that there is no difference between the values of the selected optimal PM decision rule $H_o : V_0(\hat{d}_{\text{PMM}}) = V_0(\hat{d}_{\text{ZOM}})$ and the zero-order decision rule and the alternative hypothesis was two-sided $H_a : V_0(\hat{d}_{\text{PMM}}) \neq V_0(\hat{d}_{\text{ZOM}})$. The test statistic for the jackknife was a standardized difference between the two value estimates:

$$T^{jk}(\hat{d}_{\text{PMM}}, \hat{d}_{\text{ZOM}}) = \frac{\widehat{V}^{jk}(\hat{d}_{\text{PMM}}) - \widehat{V}^{jk}(\hat{d}_{\text{ZOM}})}{\sqrt{\frac{\sum_{i=1}^{n}(R_{PMM}^{jk} - R_{ZOM}^{jk})^2}{n(n-1)}}} \tag{2.5}$$

The p-value for this test was defined as $p = 2P(|T| \leq z) = 2\int_{|T|}^{\infty} f(z)dz$ where $z \sim N(0,1)$. Note that the test statistic is nonnegative because the optimal PMM would either outperform the optimal ZOM or assign the same treatment to everyone just like the optimal ZOM. With a significance level chosen to be 0.1, we conclude that there is evidence that the treatment rules derived from the optimal PMM provides statistically significant improvement in the outcome if $p < 0.1$.

2.5.8 Stratified Cross Validation

In addition to the jackknife method (i.e., LOOCV), we applied stratified 10-fold CV with 50 repetitions, with each fold stratified by the randomization group. Let $M = 50$ denote the total

number of repetitions, $K = 10$ denote the number of CV folds, and $j = 1, \ldots, KM$ denote all 500 folds regardless of the repetition. The estimated value function was defined as

$$\widehat{V}^{cv}(\hat{d}_n) = \frac{\sum_{j=1}^{MK} \sum_{i=1}^{n_j} U_{ji}}{\sum_{j=1}^{MK} \sum_{i=1}^{n_j} W_{ji}} \tag{2.6}$$

where $i = 1, \ldots, n_j$ is the ith observations in the j-th overall fold, $W_{ji} = \frac{1\{A_{ji} = \hat{d}_n^{(-j)}(\boldsymbol{X}_{ji})\}}{\hat{P}(A_{ji}|\boldsymbol{X}_{ji})}$, $U_{ji} = Y_{ji}W_{ji}$, $\hat{d}_n^{(-j)}$ is the decision rule estimated from a training set of size n with the jth fold left out, and $\hat{P}(A_{ji}|\boldsymbol{X}_{ji})$ is the estimated propensity score. We defined the estimated variance of the jackknife value estimator as

$$\widehat{\text{Var}}[\ \widehat{V}^{cv}(\hat{d}_n)] = \frac{1}{K(MK-1)} \sum_{j=1}^{MK} \sum_{i=1}^{n_j} R_{ji}^{cv2} \tag{2.7}$$

where $R_{ji}^{cv} = \frac{1}{\bar{W}_j} U_{ji} - \frac{\bar{U}_j}{\bar{W}_j^2} W_{ji}$ is an influence function-inspired form of the value function with $\bar{U}_j = \sum_{i=1}^{n_j} U_{ji}$ and $\bar{W}_j = \sum_{i=1}^{n_j} W_{ji}$, similar to that defined in the jackknife method above and $\sum_{j=1}^{MK} \sum_{i=1}^{n_j} R_{ji}^{cv} = 0$. The variance estimate was adjusted by the degrees of freedom $MK - 1$ for MK overall folds as well as by the correlations among K folds for each repetition. The standard error of the value estimator is then $\widehat{\text{SE}} = \sqrt{\widehat{\text{Var}}(\hat{V}^{cv})}$. Because the jackknife is a special case of CV, the value function estimates of the jackknife are also special cases of those of CV with $M = 1, K = n$. As we can see, the jackknife is much simpler in terms of notation and computation.

We applied the Z-test to stratified CV value estimates defined above with the test statistic adjusted by $K(MK - 1)$:

$$T^{cv}(\hat{d}_{\text{PMM}}, \hat{d}_{\text{ZOM}}) = \frac{\widehat{V}^{cv}(\hat{d}_{\text{PMM}}) - \widehat{V}^{cv}(\hat{d}_{\text{ZOM}})}{\sqrt{\frac{\sum_{j=1}^{MK}(R_{PMM,j}^{cv} - R_{ZOM,j}^{cv})^2}{K(MK-1)}}} \tag{2.8}$$

Similar test results were observed when we compared the optimal PMM with the optimal ZOM: For input data 1, the optimal PMM for weight loss since baseline was a list model with at most

2 nodes. Applied to future patients with the 10-fold CV method, the optimal rules derived from this list-based model is expected to increase average weight loss to 10.8 kg at 18 months, contrasted with 9.8 kg had all patient received D+E. The relative increment of 1.0 kg between the PMM and ZOM was significant (p = 0.06). For input data 2, the optimal PMM for IL-6 (log-transformed in the model) was list-based DTR with at most 2 nodes embedded with RF, which is one of the two optimal PM models the jackknife method detected in the main Results section. Applied to future patients with the 10-fold CV method, the optimal rules derived from this list model is expected to decrease average IL-6 level to 2.34 pg/mL, compared with 2.56 pg/mL had all patients received D+E. The relative increment of 0.22 pg/mL was significant (p = 0.09). The CV estimated optimal rule for weight loss was the same as condition 1.1) in the Results section. The estimated optimal rule based on list DTR with a maximum of 2 nodes was the same as condition 2.1) in the Results section. Stratified 10-fold CVs tend to produce similar results as the jackknife method.

2.5.9 Multiple Outcomes

The coarse-to-fine grid search was defined as follows. First, a coarse grid search of weights between 0 and 1 with increment 0.1 was applied to find the weight combinations (of length 3) that generate the top five lowest jackknife estimators of the value function, $\widehat{V}^{jk}(\hat{d})$. Here, $\hat{d}$ was trained by a RF model and the $\widehat{V}$'s were transformed to percentiles to be compared on the same scale across the three outcomes. Next, a finer grid search of increment 0.02 was conducted to a range within ± 0.06 of the five selected sets of weights. This is to find the one weight combination that maximizes the lowest $\widehat{V}^{jk}(\hat{d})$. This two-step grid search reduced computation time significantly, as opposed to for example one large fine grid search of weights between 0 and 1 with increment 0.02.

2.5.10 Generalizability

Our efforts in preventing overfitting and improving generalizability of our PM approach are represented in three major ways: i) we applied feature selection with RF to reduce the complexity of input data; ii) we used the jackknife resampling technique to evaluate model performance as

internal validation to all models; iii) we proposed a precision medicine-based approach for many of its advantages, one of which is that PM is carried under a causal framework which produces generalizable rules. For instance, basic causal assumptions (e.g., consistency, positivity, no unmeasured confounders) and the definition of value function estimators enable the conclusions to be applied to future population. Nonetheless, we recommend a follow-up to the IDEA trial or a new randomized clinical trial to be performed in the future to confirm our findings.

2.6 Simulation Analyses

2.6.1 Simulation Settings

In addition to the clinical data analyses on the IDEA data, we carried out extensive simulations to assess the performance of the jackknife estimators of value functions in various settings. We set up the simulations to be as close to the IDEA data as possible. For each observation, the simulation data can be written as a triplet $(\boldsymbol{X}, A, Y)$ that consists of three clinical covariates $\boldsymbol{X} = \{X_1, X_2, X_3\}$ i.i.d. from uniform distribution $U(-2, 2)$, a treatment variable A of values $\{0, 1, 2\}$ generated from multinomial distribution Multinom $\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$ independently of $\boldsymbol{X}$, and a response variable Y normally distributed with mean $Q_0 = X_1 + X_2 + \delta_0(\boldsymbol{X}, A)$ and standard deviation 1. We considered four scenarios with the following different choices of Q_0:

$$
\begin{aligned}
(1) \quad \delta_0(\boldsymbol{X}, A) &= 1\{A > 0\}(1 - X_1^2 - X_2^2)(X_1^2 + X_2^2 - 3)^{1\{A=1\}} \\
(2) \quad \delta_0(\boldsymbol{X}, A) &= 1\{A > 0\}(1\{X_2 \leq \lceil X_1 - 2 \cdot 1\{A = 2\}\rceil\} \\
&\qquad\qquad -1\{X_2 > \lceil X_1 - 2 \cdot 1\{A = 2\}\rceil\}) \\
(3) \quad \delta_0(\boldsymbol{X}, A) &= 1\{A > 0\}(X_1 + X_2 - 1)(-X_1 - X_2 - 1)^{1\{A=1\}} \\
(4) \quad \delta_0(\boldsymbol{X}, A) &= 1\{A > 0\}(X_2 - X_1^2)(X_1^2 - X_2^2 - 2)^{1\{A=1\}}
\end{aligned}
$$

Scenarios (1)-(4) were determined by X_1 and X_2 only, with X_3 as a nuisance variable. The true decision boundaries of these scenarios can be described as (1) two concentric circles, (2) two parallel sets of steps of length 1, (3) two parallel lines with slope , and (4) two nested sets of parabolas (Figure 2.3). The purpose of including various scenarios in this simulation study is to

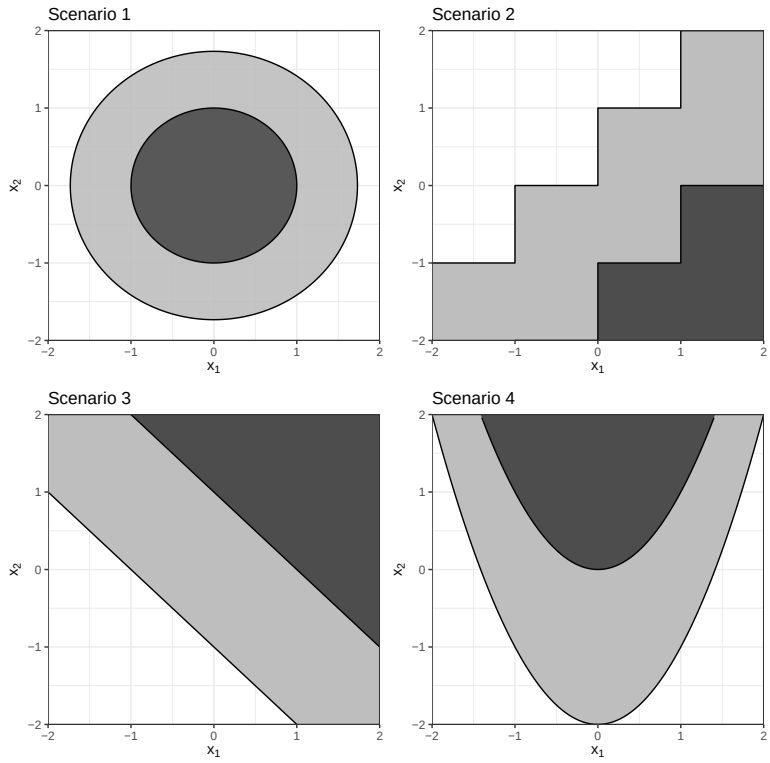

Figure 2.3: True decision boundaries of simulation scenarios (where white, light gray, and dark gray areas represent true optimal treatment 0, 1, and 2 respectively)

explore if candidate models can work well with difficult boundary structures. Samples sizes were chosen to be 50, 100, 200, 400, which cover our clinical data sample sizes, and 100 simulations were performed. In Jiang et al. (2020a), a higher sample size of $n = 800$ was explored. The estimated decision rule was denoted as $\hat{d}_n$ where n is the training size, and an independent sample of the same size as the simulation data $(\boldsymbol{X}, A, Y)$ was denoted as $(\tilde{\boldsymbol{X}}, \tilde{A}, \tilde{Y})$. We derived four estimators on simulation data:

1) Empirical estimator, where the same n observations were used for training and testing the decision rule.

$$\widehat{V}_1(\hat{d}_n) = \frac{\sum_{i=1}^{n} Y_i 1\{A_i = \hat{d}_n(\boldsymbol{X}_i)\}/\hat{P}(A_i|\boldsymbol{X}_i)}{\sum_{i=1}^{n} 1\{A_i = \hat{d}_n(\boldsymbol{X}_i)\}/\hat{P}(A_i|\boldsymbol{X}_i)}$$

2) Jackknife estimator, where $n-1$ observations were used for training the decision rule and the n-th observation for testing.

$$\widehat{V}_2(\hat{d}_n) = \frac{\sum_{i=1}^{n} Y_i 1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}/\hat{P}(A_i|\boldsymbol{X}_i)}{\sum_{i=1}^{n} 1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}/\hat{P}(A_i|\boldsymbol{X}_i)}$$

3) Jackknife + test estimator, where $n-1$ observations were used for training and an independent copy of one observation was used for testing.

$$\widehat{V}_3(\hat{d}_n) = \frac{\sum_{i=1}^{n} \tilde{Y}_i 1\{\tilde{A}_i = \hat{d}_n^{(-i)}(\tilde{\boldsymbol{X}}_i)\}/\hat{P}(\tilde{A}_i|\tilde{\boldsymbol{X}}_i)}{\sum_{i=1}^{n} 1\{\tilde{A}_i = \hat{d}_n^{(-i)}(\tilde{\boldsymbol{X}}_i)\}/\hat{P}(\tilde{A}_i|\tilde{\boldsymbol{X}}_i)}$$

4) Empirical + test estimator, where all n observations were used for training and an independent copy of the training data with the same size n was used for testing.

$$\widehat{V}_4(\hat{d}_n) = \frac{\sum_{i=1}^{n} \tilde{Y}_i 1\{\tilde{A}_i = \hat{d}_n(\tilde{\boldsymbol{X}}_i)\}/\hat{P}(\tilde{A}_i|\tilde{\boldsymbol{X}}_i)}{\sum_{i=1}^{n} 1\{\tilde{A}_i = \hat{d}_n(\tilde{\boldsymbol{X}}_i)\}/\hat{P}(\tilde{A}_i|\tilde{\boldsymbol{X}}_i)}$$

5) Empirically true estimator, where all n observations were used for training and an independent copy of the training data with size $n_{pop} = 1,000,000$ was used for testing.

$$V_0(\hat{d}_n) = \frac{\sum_{i=1}^{n_{pop}} \tilde{Y}_i 1\{\tilde{A}_i = \hat{d}_n(\tilde{\boldsymbol{X}}_i)\}/\hat{P}(\tilde{A}_i|\tilde{\boldsymbol{X}}_i)}{\sum_{i=1}^{n_{pop}} 1\{\tilde{A}_i = \hat{d}_n(\tilde{\boldsymbol{X}}_i)\}/\hat{P}(\tilde{A}_i|\tilde{\boldsymbol{X}}_i)}$$

The jackknife estimator $\widehat{V}_2$ was applied in the KOA study described in the main text. Here in the simulations, we focused on the other three estimators for comparison. Estimator $\widehat{V}_1$ was not honest (i.e. 'honesty' means that data are used for training or testing but not both) and was expected to overfit the training set. Estimator $\widehat{V}_3$ was considered as a bridge between $\widehat{V}_2$ and $\widehat{V}_4$ because it was tested on independent copies like $\widehat{V}_4$ but trained based on the jackknife method like $\widehat{V}_2$. We hoped to find that $\widehat{V}_2$ has similar performances and statistical inferences as $\widehat{V}_4$, which is the honest estimator with the largest training set. In comparison, estimator V_0 was trained on the same training set of size n as the empirical estimator but tested on a simulated data set of a much larger test size that generated the best possible estimates of the truth one can obtain empirically.

Similar to the IDEA trial analysis, we fit the three ZOMs and 23 PMMs (excluding M6) to the simulation data, with two main changes to the precision medicine models: i) multiple outcome RF model (M6 in Table 2.2) was not possible for data with one outcome hence we applied only 23 PMMs for simulations, and ii) for RLT and RF models, the default number of variables randomly sampled at each split as candidates was one for four covariates (three $\boldsymbol{X}$'s and one A) and we forced it to be two so the model does not split on the only one candidate variable. We also examined the distribution of four estimators of value functions together with the empirically true estimates and compared the optimal PMM and optimal ZOM with a Z-test. The test statistic for each simulation is

$$T^{sim}(\hat{d}_{\text{PMM}}, \hat{d}_{\text{ZOM}}) = \frac{\widehat{V}(\hat{d}_{\text{PMM}}) - \widehat{V}(\hat{d}_{\text{ZOM}})}{\sqrt{\frac{\sum_{i=1}^{n}(R_{\text{PMM,i}} - R_{\text{ZOM, i}})^2}{n(n-1)}}} \tag{2.9}$$

where $\hat{d}_{\text{PMM}}$ and $\hat{d}_{\text{ZOM}}$ are estimated decision rules for the optimal PMM and the optimal ZOM respectively, $\widehat{V}$ stands for estimated value function respectively, and $R_{\text{PMM,i}}$ and $R_{\text{ZOM,i}}$ represent the bias-corrected, influence function-inspired value function of the ith individual under the rule for the optimal PMM and ZOM respectively. The null hypothesis was that the expected future reward of the optimal PMM is the same as that of the optimal ZOM.

2.6.2 Simulation Results

First, we looked at the accuracy of the precision medicine models for reproducing the true decision boundaries of the four scenarios (Figure 2.4).KRR was selected because it represented a precision medicine model with good performance in different scenarios. The estimated decision boundaries were based on fitting the model once for a simulation sample size of n = 500. It is clear that KRR was able to estimate the decision boundary of scenarios 1, 3, and 4, with scenario 3 being the best. It had a difficulty finding the second set of steps for scenario 2 due to the small decision area of treatment $A = 2$ but was able to detect the longer set of steps.

Second, we compared the distribution of our jackknife estimators $\widehat{V}_2$ with the distribution of empirical + test estimators $\widehat{V}_4$, which are the best possible estimators but not feasible in reality

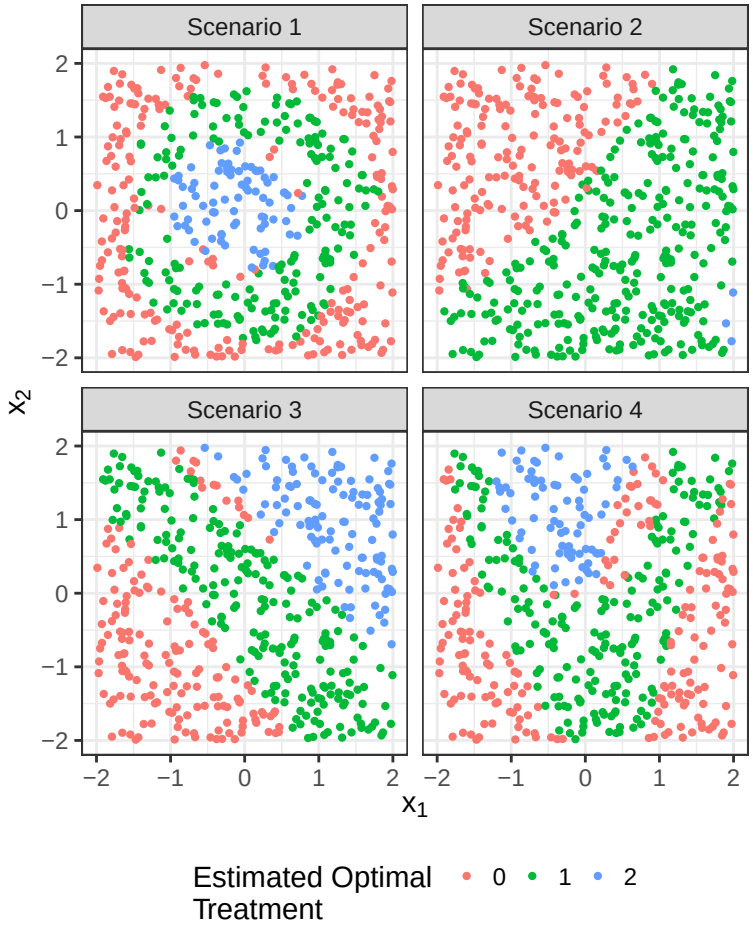

Figure 2.4: Estimated decision boundaries for a simulated dataset of size n = 500 trained by KRR models (scenario 1 - circles, scenario 2 - steps, scenario 3 – lines, scenario 4 – quadratic curves)

because we often do not have an independent copy of the sample data with the same size (or we cannot afford 50/50 random splitting of our available data). Figure 2.5 contains Q-Q plots that compare four estimators $\widehat{V}_2$ to $\widehat{V}_4$ with the empirical + test estimator $\widehat{V}_4$ based on the KRR model, which had better performance compared with other models. The purple curve is a straight line because it is a comparison between $\widehat{V}_4$ and itself. The green curve, our jackknife estimator, mostly follows the straight line except at the left tail (i.e. when the estimators are less than -2) for difficult scenarios such as 1 and 4 when n=50. As sample size increases the green curve becomes much straighter for all scenarios. This indicates that our jackknife estimators have a similar distribution as the empirical + test estimators, especially for higher sample sizes.

Next, we explored the coverage of the jackknife estimators $\widehat{V}_2$ over $\widehat{V}_4$. Given $\widehat{V}_2$ and its standard error, a 95% confidence interval (CI) was calculated for each simulation, $\widehat{V}_2 \pm z_{0.975} \cdot \text{SE}(\hat{V}_2)$, where $z_{0.975} \approx 1.96$ is the standard normal quantile of 97.5%. The coverage was defined as the proportion of simulations whose 95% CI contains V_0, which has a Monte Carlo error (the maximum standard error of the estimated proportion) of 5%. We continued to use KRR as an example of a good performing PMM and summarized the percentage of coverage with 95% CIs in Table 2.6. Overall, coverage generally increases as sample size goes up. Scenarios 2-4 have at least or almost 95% coverage for higher sample sizes. Scenario 1 did not reach as high a coverage as the other scenarios across sample sizes; we believe this is because concentric circles are complex, non-linear decision boundaries. After increasing the sample size to 800 (Jiang et al., 2020a), we saw a 96% coverage for scenario 1 and concluded that the 92%-to-88% dip at n = 400 was due to Monte Carlo errors. With the additional simulations, we were able to see more uniform patterns.

Table 2.6: Coverage of the empirically true estimator V_0 with 95% CI of $\hat{V}_2$ based on 100 simulations

Sample Size	50	100	200	400
Scenario 1	84%	87%	92%	88%
Scenario 2	91%	91%	96%	96%
Scenario 3	93%	97%	96%	96%
Scenario 4	89%	88%	90%	94%

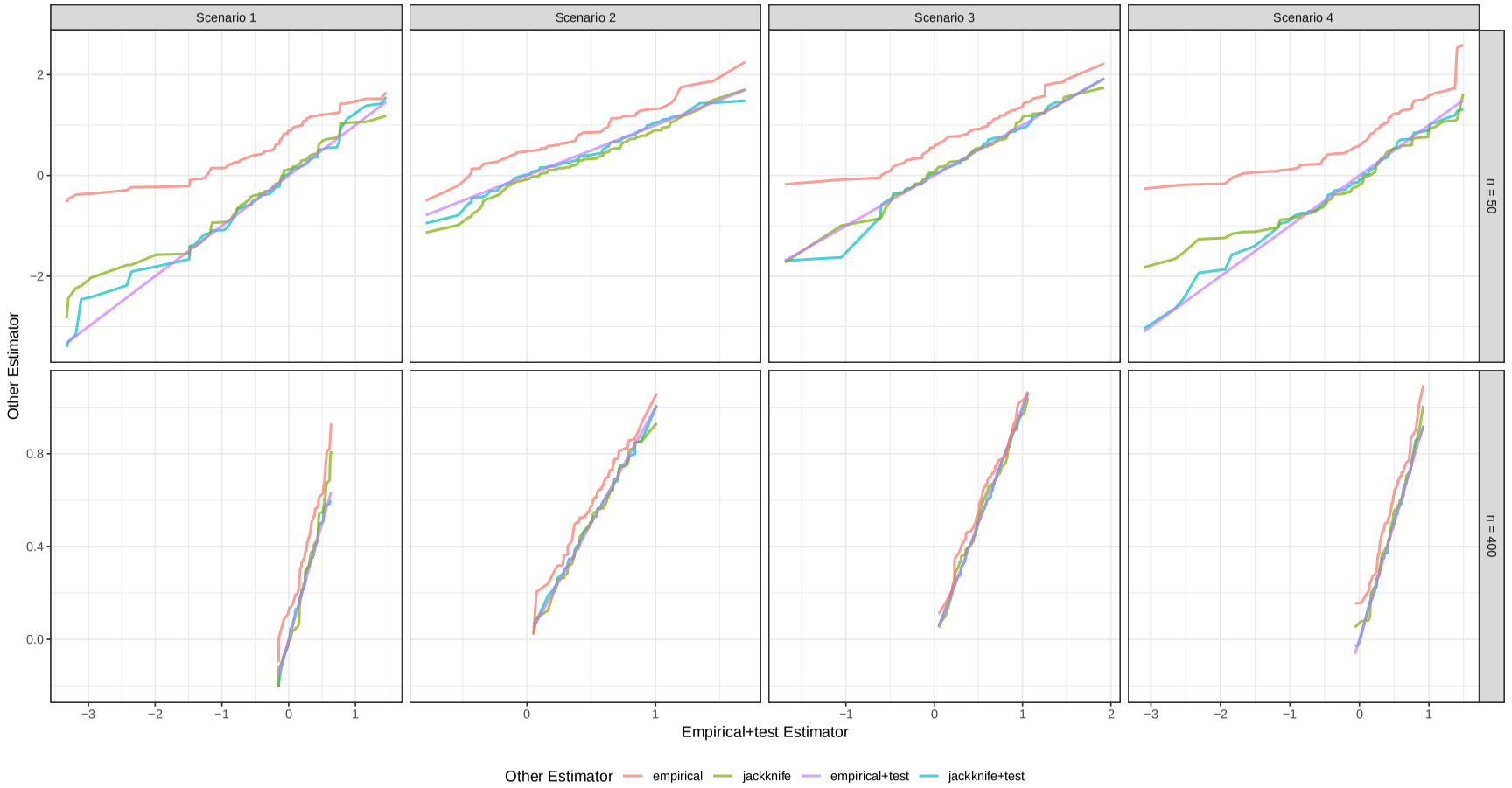

Figure 2.5: Q-Q plots of the distribution of estimators $\hat{V}_1$ to $\hat{V}_4$ versus the distribution of $\hat{V}_4$ based on the KRR model across 100 simulations for $n = 50$ and $n = 400$ over 4 scenarios (empirical is red, jackknife is green, empirical+test is purple, jackknife+test is blue)

Lastly, we estimated the power of the jackknife test statistic T^{sim}. Power was estimated by the proportion of simulations whose p-values were under 0.05 out of 100 simulations, which also has a Monte Carlo error of 5%. This reflects how often the test will detect a significant effect of PMM over ZOM when there is an effect. We found that T_0^{sim} experienced difficulty at small sample sizes and at large sample sizes for complex boundaries such as scenarios 1 and 2 (Table 2.7). Yet, the estimated power almost always increased as sample size increased for each scenario. Scenario 3, the decision boundary that KRR had good accuracy in predicting, reached the highest power (81%) at n = 400 compared with other scenarios. We saw higher power for the jackknife test statistic with larger sample size n = 800 and confirmed that power generally increases with larger sample sizes, a similar trend to CI coverage (Jiang et al., 2020a).

Table 2.7: Estimated power of jackknife T^{sim} based on 100 simulations

Sample Size	50	100	200	400
Scenario 1	13%	7%	18%	38%
Scenario 2	16%	13%	23%	34%
Scenario 3	15%	24%	41%	81%
Scenario 4	11%	15%	35%	56%

2.6.3 Consistency of Jackknife Estimators

Statistical inference properties of the jackknife estimators were evaluated through a mathematical proof and simulations. It is known that the estimate of expected prediction error calculated from cross validation is conditionally unbiased but its variance can be very large (Breiman et al., 1996). Moreover, there have been theoretical arguments that claimed that it depends on how correlated the training data are and there are no universally unbiased estimator of such variance of expected prediction error under all distributions of observations (Bengio and Grandvalet, 2004). We argued that for our case jackknife estimates of value functions are asymptotically unbiased and their variances converge to zero as sample size increases. We summarized the consistency statement of a jackknife estimator summarized in Theorem 2.1 as well as its assumptions.

Assumption 2.1.

$$E[P_{\boldsymbol{X}}(\hat{d}_n(\boldsymbol{X}) \neq \hat{d}_{n-1}(\boldsymbol{X}))] \to 0$$

Assumption 2.2.

$$E\left[\frac{Y^2}{\hat{P}(A|\boldsymbol{X})} + \frac{1}{\hat{P}(A|\boldsymbol{X})}\right] < \infty$$

Assumption 2.1 is reasonable because the training sets of size n and $n-1$ are asymptotically equal, which implies that the decision functions estimated from these two training sets eventually converge as sample size grows to infinity. Assumption 2.2 requires a finite second moment of the outcome adjusted by the propensity score and thus a finite variance of the adjusted outcome, which is easily satisfied for clinical data where the outcome itself is finite and is usually contained in a range. The second term in Assumption 2.2 is automatically satisfied because propensity scores are bounded between 0 and 1 and it is used in the analogous proof of W_n.

Theorem 2.1. *Given Assumptions 2.1 and 2.2,*

$$\frac{\sum_{i=1}^{n} \frac{Y_i 1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}}{\hat{P}(A_i|\boldsymbol{X}_i)}}{\sum_{i=1}^{n} \frac{1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}}{\hat{P}(A_i|\boldsymbol{X}_i)}} - E[Y|A = \hat{d}_n(\boldsymbol{X})] \underset{p}{\to} 0$$

The proof can be found in Appendix A.

2.6.4 Asymptotic Normality of Jackknife Estimators

We examined the asymptotic normality property of jackknife estimators via simulations. For each estimator, sample size, and data scenario across 100 simulations, we calculated the following shifted test statistic:

$$T_0^{sim} = \frac{[\hat{V}^{jk}(\hat{d}_{\text{PMM}}) - \hat{V}^{jk}(\hat{d}_{\text{ZOM}})] - [V_0(\hat{d}_{\text{PMM}}) - V_0(\hat{d}_{\text{ZOM}})]}{\sqrt{\frac{\sum_{i=1}^{n}(R_{\text{PMM},i}^{jk} - R_{\text{ZOM},i}^{jk})^2}{n(n-1)}}} \tag{2.10}$$

which is a function of jackknife estimators and a "shifted" version of T^{sim}. The test statistic measures the difference between the estimated value function of optimal PMM and that of optimal ZOM, shifted by the corresponding difference in the true value function. The distribution of

T_0^{sim} over 100 simulations was compared with the standard normal distribution and visualized via Q-Q plots for all four scenarios (Figure 2.6). Only two sample sizes 50 and 400 were shown for cleaner plots. We can see that the distribution of test statistic is mostly standard normal in the middle and the scattering of points in scenarios 2 and 3 is particularly close to a straight line. In addition to visual inspections, we also tested the normality on T_0^{sim} using the Shapiro-Wilk test. Simulation studies have shown that Shapiro-Wilk (SW) test has good power properties over symmetric distributions as well as a wide range of skewed distributions (Yap and Sim, 2011). Based on both the Q-Q plots and SW test results (Table 2.8), T_0^{sim} has fewer outliers and is more normally distributed (especially in the middle) when sample size increases. We were able to reject the null hypothesis that T_0^{sim} follows a standard normal distribution for scenarios 3 and 4 when n = 50, which we believe was due to a few outliers at the positive end. In summary, we conclude that there is not enough evidence to reject the hypothesis that T_0^{sim} is standard normally distributed for moderate to high sample sizes ($n \geq 100$) in all four decision boundary scenarios, and the Q-Q plots indicated evidence for asymptotic normality. In addition to the consistency and asymptotic normality properties inspected here, there are more complex and technically rigorous approaches to value function inference than what we propose here: for a review of such methods, see (Laber et al., 2014).

Table 2.8: P-values of Shapiro-Wilk test of normality on jackknife T_0^{sim} based on 100 simulations

Sample Size	50	100	200	400
Scenario 1	0.20	0.37	0.15	0.18
Scenario 2	0.85	0.92	0.41	0.79
Scenario 3	<0.01	0.99	0.13	0.61
Scenario 4	<0.01	0.67	0.81	0.84

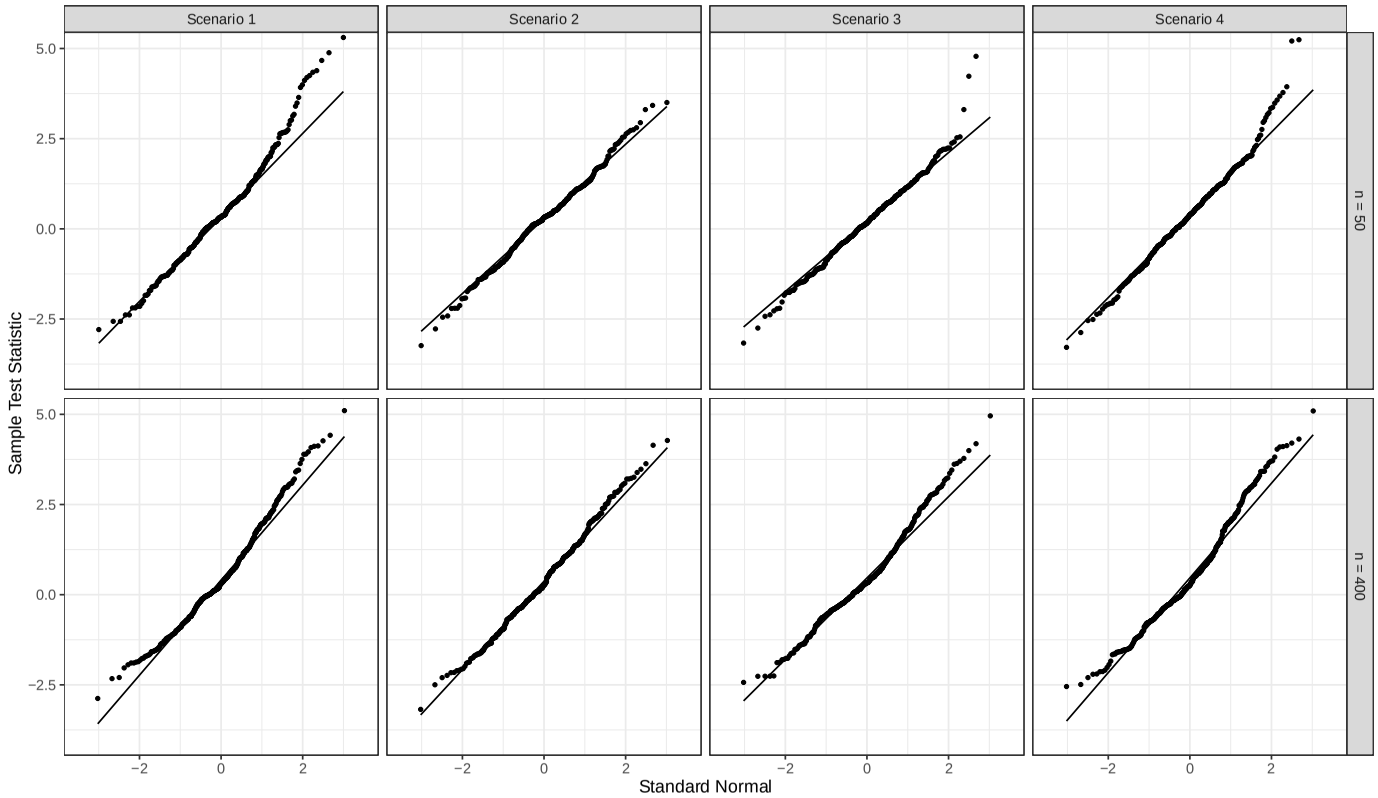

Figure 2.6: Q-Q plots of the distribution of jackknife T_0^{sim} versus the standard normal distribution across 100 simulations for $n = 50$ and $n = 400$ over 4 scenarios

CHAPTER 3: DEEP DOUBLY ROBUST OUTCOME-WEIGHTED LEARNING

3.1 Introduction

Clinical practices have been gradually undergoing a slow-but-steady transformation, from prescribing the same treatment to all patients of one disease (which tends to work well "on average") to personalizing treatment options that target precisely on smaller groups of patients. This is often accomplished through a data-driven diagram called precision medicine (Kosorok and Laber, 2018), which takes into account patient heterogeneity into the decision making process of disease treatment and does not require prespecified values or strong assumptions. Patient heterogeneity can be acquired in the richer medical history of routine visits, lab exams, medication, surveys, as well as larger and more complex data in medical images and genetic traits. The goal of precision medicine is to identify the combination of treatment and patient groups that achieves optimal clinical outcomes, which can be described by individualized treatment regimes (ITRs).

Assume the input data is a sample of n i.i.d. triplets $(\boldsymbol{X}_i, A_i, Y_i)$ for $i = 1, \ldots, n$, where $\boldsymbol{X}_i \in \mathcal{X} \subseteq \mathbb{R}^p$ represents p-dimensional patient information, $A_i = \{-1, 1\} \in \mathcal{A}$ represents treatment, and $Y_i \in \mathbb{R}$ clinical outcome. The underlying data-generating model does not change regardless of the chosen treatment and the observed input data are used to determine the optimal treatment rule to be applied to future patients (Kosorok and Laber, 2018). Let $P_a(\boldsymbol{x}) = P(A = a|\boldsymbol{X} = \boldsymbol{x})$ denote the propensity score, which is known for randomized trials and needs to be estimated for observational studies or non-randomized trials. An ITR is a function $d : \mathcal{X} \to \mathcal{A}$ that maps from the patient characteristic space $\mathcal{X}$ to the treatment space $\mathcal{A}$. The optimal ITR is the one function d^{opt}, within the class of all available functions $\mathcal{D}$, that gives the best expected outcome results. Throughout this paper, we make the following

three standard causal assumptions: consistency $Y = Y^*(A)$, no unmeasured confounders $Y^*(a) \perp A|\boldsymbol{X}$ for all $a \in \mathcal{A}$, and positivity $P_a(\boldsymbol{x}) > 0$ for all $a \in \mathcal{A}$ and $\boldsymbol{x} \in \mathcal{X}$ in addition to the SUTVA assumptions of no interference and no multiple versions of treatments. We also assume higher outcomes are more desirable. Under these assumptions, the optimal decision rule is $d^{opt}(\boldsymbol{x}) = \arg\max_{d\in\mathcal{D}} E(Y^d)$ where $E(Y^d)$ is the expectated outcome under treatment d, which is known as the "value" of an ITR as $V(d) = E(Y^d) = E\left[\frac{Y1\{A=d(\boldsymbol{X})\}}{P(A|\boldsymbol{X})}\right]$. Because $V(d) = E[Q(\boldsymbol{X}, d(\boldsymbol{X}))]$, the optimal ITR also satisfies $d^{opt}(\boldsymbol{x}) = \arg\max_{d\in\mathcal{D}} Q(x, d)$ a.s. for all $\boldsymbol{x} \in \mathcal{X}$, where $Q(x, a) = E[Y|\boldsymbol{X} = \boldsymbol{x}, A = a]$ is the "quality" of treatment a applied at patient observation $\boldsymbol{x}$ and can be estimated by traditional regression models (Qian and Murphy, 2011). The goal of precision medicine boils down to finding $\hat{d}_n$, the estimator of d^{opt}, given n triplets of observed data $\{(\boldsymbol{X}_i, A_i, Y_i)\}_{i=1}^n$.

Many machine learning methods that solve for the ITR estimation problem are indirect and regression-based like described above. For example, Qian and Murphy (2011) used a two-step procedure with a ℓ_1-penalized least squares model. Zhang et al. (2012) went on another path with counterfactuals, where the optimal ITR was obtained by the augmented inverse probability weighting estimator (AIPWE, Robins et al. (1994); Rotnitzky et al. (1998)). The outcome regression model was thus protected by double robustness from misspcification while gaining estimation precision (Zhang et al., 2012). An alternative to completely avoid the potential misspecification of regression models is a direct, classification-based approach. A leading example is outcome weighted learning (OWL), which estimated the optimal treatment by converting the maximization problem to a weighted classification problem with support vector machine (SVM) (Zhao et al., 2012). A lot of research has been done to generalize this classification approach to a broader range of problems and datasets. To name a few: residual weighted learning (RWL) improved finite sample performance of OWL by replacing the outcome with residuals which leave only the heterogeneous part of the $X - A - Y$ relationship and are more invariant to different types and scales of data (Zhou et al., 2017), and augmented

outcome-weighted learning (AOL) extended RWL further by combining RWL and AIPWE to achieve even better performance and computational efficiency (Zhou and Kosorok, 2017).

We propose a new method called deep, doubly robust outcome weighted learning (DDROWL) that applies deep learning techniques to solve the AOL problem with greater flexibility, especially for large, complex data sizes. AOL benefits from desirable properties of both residuals and doubly robustness while outperforming either model. Deep learning has recently gained much popularity due to its supreme prediction accuracy trained from high-dimensional data. Living under the umbrella of machine learning, deep neural networks (DNN) are flexible, data-driven tools that effectively extract important representations layer-by-layer from a large amount of input data. We propose to apply DNNs directly in the optimization problem to find the optimal decision function d^{opt}. If well-trained, DNNs avoid the need to posit regression models on outcomes or residuals like the regression-based methods and thus the possibility of model misspecification. Instead of using kernel mapping to generalize the decision rule to non-linear problems, the nonlinear hierarchial architecture of DNN can accommodate for any form of decision rules while maintaining good performance and computation speed. In addition, DNN enables DDROWL to take in high-dimensional and complex patient data such as medical imaging and time series data (e.g., audio and speech). It is known that DNN can easily overfit, therefore training and tuning the network well is an important task in our methodology development.

The rest of the paper is organized as follows. In Sections 3.2.1 and 3.2.2, we review existing methods (i.e., OWL, RWL, and AOL) and present the proposed DDROWL for estimating optimal ITRs in more general settings. Background information of DNN structures we used can be found in Sections 3.2.3 and 3.2.4. Theoretical properties and intuitions are explored in Section 3.2.5. In comparison with other competing methods, the performance of our proposed method in terms of estimated value functions is demonstrated through simulations in Section 3.3 and clinical application using clinical data and brain images in Section 3.4. Section 3.5 concludes the article with discussions. Future research directions and technical details can be found in Chapter 5.

3.2 Methods

Before introducing DDROWL and its theoretical properties, we first review several current methods on which our proposed idea was based.

3.2.1 Existing Work

Recall that the objective function of outcome weighted learning (OWL) can be turned into a weighted misclassification error as each misclassified event $1\{A \neq d(\boldsymbol{X})\}$, a 0-1 loss, is weighted by $Y/P(A|\boldsymbol{X})$,

$$d^{opt} = \arg\min_{d \in \mathcal{D}} \ E\left[\frac{Y1\{A \neq d(\boldsymbol{X})\}}{P(A|\boldsymbol{X})}\right].$$

This misclassification error is approximated using observed data and the optimal ITR can be obtained through some function f in a class of all possible decision functions $\mathcal{F}$, such that

$$\widehat{f}^{opt} = \arg\min_{f \in \mathcal{F}} \ n^{-1} \sum_{i=1}^{n} \left[\frac{Y_i 1\{A_i \neq \operatorname{sign} f(\boldsymbol{X}_i)\}}{\hat{P}(A_i|\boldsymbol{X}_i)}\right]$$

where $\hat{P}(A_i|\boldsymbol{X}_i)$ is the estimated propensity score for the ith subject and $d^{opt}(\boldsymbol{x}) = \operatorname{sign}(f^{opt}(\boldsymbol{x}))$. To alleviate the non-convexity of the 0-1 loss function in this optimization problem, Zhao et al. (2012) proposed a nonparametric approach to adopt hinge loss as a surrogate hinge loss because it is a tight and convex upper bound of 0-1 loss. Thus, OWL aims to solve the following objective function with an added term to penalize the complexity of f and avoid overfitting:

$$\widehat{f}^{opt} = \arg\min_{f \in \mathcal{F}} \ n^{-1} \sum_{i=1}^{n} \left\{\left[\frac{Y_i}{\hat{P}(A_i|\boldsymbol{X}_i)} \ell_{\mathrm{H}}(A_i f(\boldsymbol{X}_i))\right] + \lambda_n ||f||^2\right\},$$

where $\ell_{\mathrm{H}}(t) = \max(1-t, 0)$ is the hinge loss function, $||\cdot||$ is some appropriate norm in the $\mathcal{F}$ space, and λ_n is a penalization parameter. This optimization problem can then be solved by support vector machine (SVM). Zhao et al. (2012) derived the estimators for both linear and non-linear decision rules and established consistency proofs of optimal ITRs estimated by OWL as well as the risk bounds.

As it is indicated in Section 3.1, OWL has potential weaknesses in that: i) it tries to minimize misclassification rates and tends to keep the same treatments that patients already received, ii) the estimated ITR can be affected by a shift in the outcome which leads to unstable estimates especially when sample size is small, and iii) it lacks of variable selection (VS) features (Zhou et al., 2017). Residual weighted learning (RWL), proposed by Zhou et al. (2017), mitigates these problems with the following optimization

$$\widehat{f}^{opt}_{RWL} = \arg\min_{f\in\mathcal{F}} \; n^{-1}\sum_{i=1}^{n} \frac{Y_i - \hat{g}(\boldsymbol{X}_i)}{\hat{P}(A_i|\boldsymbol{X}_i)}\ell_{\text{SR}}(A_i f(\boldsymbol{X}_i)) + \frac{\lambda_n}{2}||f||^2 \tag{3.11}$$

where ℓ_{SR} is a smoothed ramp loss function and λ_n and $||\cdot||$ are defined similarly as those in (3.11). In (3.11), the outcome Y is replaced by the residuals of a model $g(\boldsymbol{X})$ that only depends on $\boldsymbol{X}$ thus the optimal ITR is invariant to many kinds of outcomes (binary, continuous, count). A good choice of function g should not depend on d because d is unknown but g should reduce the variance of $\frac{Y-g(\boldsymbol{X})}{P(A|\boldsymbol{X})}1\{A \neq d(\boldsymbol{X})\}$. The authors recommend a reasonable choice of g to be

$$g^*(\boldsymbol{X}) = E\left(\frac{Y}{2P(A,\boldsymbol{X})}\middle|\boldsymbol{X}\right),$$

and the estimation of g^* can be achieved by either the main effects model $g^*(\boldsymbol{X}) = \beta_0 + \boldsymbol{X}_i^T\boldsymbol{\beta}$ or the null model $g^*(\boldsymbol{X}) = \beta_0$. The finite sample performance has demonstrated to be improved because residuals could stabilize the variance of the value function while controlling for the treatment matching factor (Zhou et al., 2017).

RWL requires computationally intensive optimization due to the non-convex loss function ℓ_{SR} which does not always promise a global solution or possess fully semi-parametrical efficiency. Zhou and Kosorok (2017) proposed augmented outcome-weighted learning (AOL) which optimizes the following objective function with weights derived from augmented outcomes:

$$\widehat{f}^{opt}_{AOL}(\boldsymbol{x}) = \arg\min_{f\in\mathcal{F}} \; n^{-1}\sum_{i=1}^{n}\left\{\frac{|y_i - \hat{\tilde{g}}(\boldsymbol{x}_i)|}{\hat{P}(a_i|\boldsymbol{x}_i)}\ell_{\text{HH}}(a_i\,\text{sign}(y_i - \hat{\tilde{g}}(\boldsymbol{x}_i))f(\boldsymbol{x}_i)) + \frac{\lambda_n}{2}||f||^2\right\} \tag{3.12}$$

where $\tilde{g}(\boldsymbol{x}) = P(-1, \boldsymbol{x})E[Y|\boldsymbol{X} = \boldsymbol{x}, A = +1] + P(+1, \boldsymbol{x})E[Y|\boldsymbol{X} = \boldsymbol{x}, A = -1]$ is derived from the AIPWE and ℓ_{HH} is the Huberized hinge loss function that is smooth everywhere. AOL inherits good properties of RWL(e.g. location-scale invariant, universally consistent, and stable variability) and builds on the doubly robust estimator to construct semi-parametrically efficient regimes. AIPWEs provide double protection against mismodeling because they require either the propensity score model or the regression model to be correctly specified but not both. Zhao et al. (2019) showed theoretical results that the AIPWE estimator of value function is consistent for the true value function for a fixed d. In addition to the benefits of using a residual (as discussed in the previous paragraph), AOL is location-scale invariant to the outcomes and recommends optimal treatments based on the sign of residuals (Zhou and Kosorok, 2017).

3.2.2 Deep Doubly Robust Outcome Weighted Learning (DDROWL)

Let $\mu_a(\boldsymbol{x}) = E[Y|A = a, \boldsymbol{X} = \boldsymbol{x}]$ be the conditional expectation of outcome given covariates $\boldsymbol{X} = \boldsymbol{x}$ and treatment $A = a$. We define $Y - \hat{r}(\boldsymbol{X})$ as residuals where

$$\hat{r}(\boldsymbol{x}) = \sum_{a \in \{-1,1\}} \hat{P}_a(\boldsymbol{x})\hat{\mu}_a(\boldsymbol{x}) \tag{3.13}$$

is the estimated weighted average of conditional outcomes μ_a for $a \in \{-1, 1\}$. Note that $\hat{r}(\boldsymbol{x})$ is equivalent to $\hat{E}[Y|\boldsymbol{X} = \boldsymbol{x}]$, the estimated conditional expectation of outcomes.

The weighted classification problem associated with RWL consists of finding the empirical optimal rule $d(x)$ in (3.14). By (3.15), a convex optimization result from Liu et al. (2016), the optimal rule d^{opt} is equivalent to the weighted classification problem associated with doubly robust outcome weighted learning, which leads to finding the optimal rule $d^*(x)$ in (3.16).

$$\begin{aligned}
\widehat{d}^{opt} &= \arg\min_{d \in \mathcal{D}} \left(\frac{1}{n} \sum_{i=1}^{n} \frac{y_i - \hat{r}(\boldsymbol{x}_i)}{\hat{P}(a_i|\boldsymbol{x}_i)} 1\{a_i \neq d(\boldsymbol{x}_i)\} \right) && (3.14) \\
&= \arg\min_{d \in \mathcal{D}} \left(\frac{1}{n} \sum_{i=1}^{n} \frac{y_i - \hat{r}(\boldsymbol{x}_i)}{\hat{P}(a_i|\boldsymbol{x}_i)} 1\{a_i \neq d(\boldsymbol{x}_i)\} + \frac{1}{n} \sum_{i=1}^{n} \frac{\max(\hat{r}(\boldsymbol{x}_i) - y_i, 0)}{\hat{P}(a_i|\boldsymbol{x}_i)} \right) && (3.15) \\
&= \arg\min_{d \in \mathcal{D}} \left(\frac{1}{n} \sum_{i=1}^{n} \frac{|y_i - \hat{r}(\boldsymbol{x}_i)|}{\hat{P}(a_i|\boldsymbol{x}_i)} 1\{a_i \cdot \text{sign}(y_i - \hat{r}(\boldsymbol{x}_i)) \neq d(\boldsymbol{x}_i))\} \right) && (3.16)
\end{aligned}$$

Here the 0-1 loss is not convex and also NP-hard, which makes it computationally hard to find global minima. Despite choosing the commonly used hinge loss and adding penalty terms to regularize parameters, we propose to apply the Cauchy-Schwarz divergence loss instead. An recent article that looked into many loss functions for deep neural network in classification found that "Cauchy-Schwarz divergence as an optimisation criterion seems to be a consistently better choice than log loss" and recommended further investigation (Janocha and Czarnecki, 2017). For our classification problem, this translates to replacing the 0-1 loss $1\{a_i \cdot \text{sign}(y_i - \hat{r}(\boldsymbol{x}_i)) \neq d(\boldsymbol{x}_i))\}$ in (3.16) with

$$\ell_{CS}(\boldsymbol{x}, y) = -1\{a \cdot \text{sign}(y - \hat{r}(\boldsymbol{x})) = 1\} f(\boldsymbol{x}) + \frac{1}{2} \log\left(1 + e^{2f(\boldsymbol{x})}\right) \tag{3.17}$$

This new surrogate is smooth and convex because all derivatives with respective to f exist and the second derivative is positive. The derivation of the Cauchy-Schwarz divergence loss (3.17) can be found in Appendix B. We express the proposed deep doubly robust outcome weighted learning (DDROWL) as the following optimization over the function $f(x)$:

$$\widehat{f}^{opt}_{DDROWL} = \underset{f \in \mathcal{F}}{\arg\min} \left(\frac{1}{n} \sum_{i=1}^{n} \frac{|y_i - \hat{r}(\boldsymbol{x}_i)|}{\hat{P}(a_i|\boldsymbol{x}_i)} \ell_{CS}(\boldsymbol{x}_i, y_i) \right) \tag{3.18}$$

where $\mathcal{F}$ denotes a class of all possible functions that we choose from. The optimal decision rule is obtained by $\widehat{d}^{opt}_{DDROWL}(\boldsymbol{x}) = \text{sign}(\widehat{f}^{opt}_{DDROWL}(\boldsymbol{x}))$. As the name suggests, DDROWL utilizes deep neural networks to estimate f, whose implementation is laid out with more detail in the next section.

3.2.3 Feedforward Neural Networks (FFNN)

Consider a L-layer feedforward deep neural network (with one hidden layer) where $l = \{0, \ldots, L\}$ represents the layer with input layer $l = 0$, the output layer $l = L$, and everything in the middle as hidden layers. Let $Z^{[l]} \in \mathbb{R}^{n_l}$ be the output of the lth layer, then $Z^{[0]} = \boldsymbol{X}$ and $Z^{[L]} = Y$. Denote the matrix of weights connecting previous layer $l-1$ and the current layer l as $W_l \in \mathbb{R}^{n_l \times n_{l-1}}$ with $n_0 = p$ and $n_L = 1$, and $b^{[l]} \in \mathbb{R}^{n_l}$ as the bias vector. For the ith

observation, the output of the l-th later of a neural network is a $n_l \times 1$ vector of the form

$$Z_i^{[l]} = a^{[l]}({W^{[l]}}^T Z_i^{[l-1]} + b_i^{[l]}) \tag{3.19}$$

where $a^{[l]}$ is a non-linear function called activation function at the lth layer. Often continuously differentiable and convex, activation functions map the linear output from the current layer to a desired range of values as inputs of the next layer. Putting together all layers defined in (3.19), and we can denote the decision function as

$$\begin{aligned} f(\boldsymbol{x}_i) &= a^{[L]}({W^{[L]}}^T \boldsymbol{z}_i^{[L]} + b^{[L]}) \\ &= a^{[L]}\left\{{W^{[L]}}^T\left[\cdots a^{[1]}({W^{[1]}}^T(\boldsymbol{x}_i) + b^{[1]})\cdots\right] + b^{[L]}\right\} \\ &= f(\boldsymbol{x}_i; \boldsymbol{W}, \boldsymbol{b}) \end{aligned}$$

Let $\theta = \{\boldsymbol{W}, \boldsymbol{b}\} = \{(W^{[1]}, b^{[l]}) \ldots, (W^{[L]}, b^{[L]})\}$. The objective function in (3.18) can be further transformed as a minimization over θ, which is automatically implemented in deep neural network algorithms such as forward and backward propagation with (3.18) as the loss function. After we get $\hat{\boldsymbol{W}}_n^{opt}$ and $\hat{\boldsymbol{b}}_n^{opt}$ which are the estimated paramters from carefully trained and tuned DNNs, the optimal decision function follows as $\widehat{f}_{DDROWL}^{opt}(\boldsymbol{x}) = f(\boldsymbol{x}; \hat{\boldsymbol{W}}_n^{opt}, \hat{\boldsymbol{b}}_n^{opt})$.

The deep learning part of DDROWL comes in handy and exhibits flexibility, especially when the covariates are too complex or large in size to be modeled parametrically or model well-specification is unknown or crucial to the question. Later in simulations (Section 3.3), we apply such a feedforward neural network designed from scratch as one of the two DDROWL methods. Although not demonstrated in this article, $\hat{\mu}_a(\boldsymbol{x})$ and $\hat{P}_a(\boldsymbol{x})$ in (3.13) can be obtained from appropriately-chosen deep learning models rather than the commonly used (generalized) linear models.

3.2.4 Deep Kernel Learning (DKL)

We incorporate deep kernel learning to estimate $\hat{f}$ as the second DDROWL model. Deep kernel learning (Wilson et al., 2016) uses neural networks to derive scalable closed-form kernels

from inputs of a spectral mixture base kernel, applies Gaussian processes (GP) to learn these kernels, and produces probabilistic mapping from the infinite non-parametric layer to the outputs. Simply put, DKL can be pictured as a GP mounted on top with a deep learning architecture and it jointly learns both the neural network parameters and base kernel hyperparameters. It is able to lower the computation complexity down to linear order compared to conventional scalable GP approaches. The authors demonstrated scalability and accuracy in learning expressive respresentations in experiments of 16 UCI regression datasets as well as image extractions such face orientation and digit magnitude, all of which have larger sample sizes in the thousands. Although the results show that larger datasets are more useful for extracting representations, we observe the performance of DKL in smaller dataset scales in Sections 3.3 and Sections 3.4.

DKL can be used for both regression and classification, but it is not straightforward to directly apply DKL in our classification problem described in Eq (3.18). We propose to transform the input data with weighted bootstrapping first before applying DKL. First, rewrite Eq (3.18) with $\hat{w}(\boldsymbol{x}_i) = \frac{|y_i - \hat{r}(\boldsymbol{x}_i)|}{\hat{P}_{a_i}(\boldsymbol{x}_i)}$ and $u(\boldsymbol{x}_i, a_i) = 1\{a_i \operatorname{sign}(y_i - \hat{r}(\boldsymbol{x}_i)) = 1\}$, and

$$\begin{aligned}
\widehat{f}_n^{opt} &= \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \left\{ \hat{w}(\boldsymbol{x}_i) \left[-u(\boldsymbol{x}_i, a_i) f(\boldsymbol{x}_i) + \frac{1}{2} \log\left(1 + e^{2f(\boldsymbol{x}_i)}\right) \right] \right\} \\
&= \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \left\{ \frac{\hat{w}(\boldsymbol{x}_i)}{\sum_{i=1}^{n} \hat{w}(\boldsymbol{x}_i)} \left[-u(\boldsymbol{x}_i, a_i) f(\boldsymbol{x}_i) + \frac{1}{2} \log\left(1 + e^{2f(\boldsymbol{x}_i)}\right) \right] \right\} \\
&= \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \left\{ \hat{p}(\boldsymbol{x}_i) \left[-u(\boldsymbol{x}_i, a_i) f(\boldsymbol{x}_i) + \frac{1}{2} \log\left(1 + e^{2f(\boldsymbol{x}_i)}\right) \right] \right\}
\end{aligned}$$

The objective function of a bootstrapped sample of $\boldsymbol{x}$ (resampling with replacement) with weights $\hat{p}_i = \hat{p}(\boldsymbol{x}_i) = \frac{\hat{w}(\boldsymbol{x}_i)}{\sum_{i=1}^{n} \hat{w}(\boldsymbol{x}_i)}$ for $i = 1, \ldots, n$ can be expressed as

$$\arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \left\{ \frac{\omega_i}{n} \left[-u(\boldsymbol{x}_i, a_i) f(\boldsymbol{x}_i) + \frac{1}{2} \log\left(1 + e^{2f(\boldsymbol{x}_i)}\right) \right] \right\}$$

where $(\omega_1, \ldots, \omega_n)$ are the bootstrap weight samples drawn from a multinomial distribution $Multinom(n; \hat{p}_1, \ldots, \hat{p}_n)$ such that $\omega_1 + \omega_2 + \ldots + \omega_n = n$. In a more general case, we can consider m-out-of-n bootstrap. Instead of a bootstrapped sample of the same size as the original

sample size n, m-out-of-n bootstrap draws a sample of size m where $m \geq n$. The bootstrap weights are $(\omega_1, \ldots, \omega_n) \sim Multinom(m; \hat{p}_1, \ldots, \hat{p}_n)$ such that $\omega_1 + \omega_2 + \ldots + \omega_n = m$.

3.2.5 Theoretical Properties

We first draw connections of our proposed objective function with the new surrogate loss to logistic regression and then justify the use of weighted bootstrap described in Section 3.2.4 with a consistency theorem.

3.2.5.1 Connection to Logistic Regression

The use of Cauchy-Schwarz divergence loss has a connection with the objective function of a logistic regression. To see this, we simplify the objective function (3.18) with $\hat{w}(\boldsymbol{x}) = \frac{|y_i - \hat{r}(\boldsymbol{x}_i)|}{\hat{P}_{a_i}(\boldsymbol{x}_i)}$ and $u(\boldsymbol{x}_i, a_i) = 1\{a_i \text{sign}(y_i - \hat{r}(\boldsymbol{x}_i)) = 1\}$ like we did in Section 3.2.4. The optimization problem can be further rewritten as

$$\begin{aligned} & \arg\min_f \mathbb{P}_n \left\{ \hat{w}(\boldsymbol{x})[-u(\boldsymbol{x}, a) f(\boldsymbol{x}) + \frac{1}{2}\log(1 + e^{2f(\boldsymbol{x})})] \right\} \\ = & \arg\min_f \frac{1}{2}\mathbb{P}_n \left\{ \hat{w}(\boldsymbol{x})[-u(\boldsymbol{x}, a) \cdot 2f(\boldsymbol{x}) + \frac{1}{2}\log(1 + e^{2f(\boldsymbol{x})})] \right\} \\ = & \arg\min_f \mathbb{P}_n \left\{ \hat{w}(\boldsymbol{x})[-u(\boldsymbol{x}, a) f(\boldsymbol{x}) + \log(1 + e^{f(\boldsymbol{x})})] \right\} \end{aligned} \tag{3.20}$$

For logistic regression, define the sigmoid function to be $\sigma(z) = \frac{1}{1+e^{-z}} = \frac{e^z}{1+e^z}$. Thus $\log(\sigma(z)) = z - \log(1 + e^z)$ and $\log(1 - \sigma(z)) = -\log(1 + e^z)$. Under the independent, identically distributed assumption (i.i.d.), the log likelihood of all data is

$$\begin{aligned} \ell(\theta) &= \sum_{i=1}^{n} \left\{ y_i \log\left[\sigma(\theta^T \boldsymbol{x}_i)\right] + (1 - y_i) \log\left[1 - \sigma(\theta^T \boldsymbol{x}_i)\right] \right\} \\ &= \sum_{i=1}^{n} \left\{ y_i(\theta^T \boldsymbol{x}_i) - y_i \log\left(1 + e^{\theta^T \boldsymbol{x}_i}\right) - (1 - y_i)\log(1 + e^{\theta^T \boldsymbol{x}_i}) \right\} \\ &= \sum_{i=1}^{n} \left\{ y_i(\theta^T \boldsymbol{x}_i) - \log(1 + e^{\theta^T \boldsymbol{x}_i}) \right\} \end{aligned}$$

The optimization of logistic regression is $\theta^{opt} = \arg\max_\theta \sum_{i=1}^{n} \left\{ y_i(\theta^T \boldsymbol{x}_i) - \log(1 + e^{\theta^T \boldsymbol{x}_i}) \right\}$. The linear predictor $\theta^T \boldsymbol{x}$ can be relaxed to $g(\boldsymbol{x})$ for any function of interest f. Thus, for logistic

regression the generalized objective is maximizing $\ell(f) = \sum_{i=1}^{n} \left\{ y_i f(\boldsymbol{x}_i) - \log(1 + e^{f(\boldsymbol{x}_i)}) \right\}$, which is equivalent to minimizing $-\ell(f) = \sum_{i=1}^{n} \left\{ -y_i f(\boldsymbol{x}_i) + \log(1 + e^{f(\boldsymbol{x}_i)}) \right\}$. We now have converted the optimization problem to

$$f^{opt}_{\text{logistic}}(\boldsymbol{x}) = \arg\min_{f} \mathbb{P}_n \left\{ -y_i f(\boldsymbol{x}_i) + \log(1 + e^{f(\boldsymbol{x}_i)}) \right\}.$$

This means (3.20) can be treated as a weighted classification problem solved by a weighted "generalized" logistic regression with binary outcome $u(\boldsymbol{x})$ and weights $\hat{w}(\boldsymbol{x})$. This aligns with proposition 3 of Janocha and Czarnecki (2017), where Cauchy-Schwarz divergence loss is shown to be equivalent to regularized cross entropy loss.

3.2.5.2 Consistency of Weighted Bootstrap

Lemma 3.1. *Let Θ_n be the space containing all possible θ for a sample size of n. Assume there exists a sequence of partitions in Θ_n such that $\Theta_n = \{\cup_{1 \le j \le K_n} \Theta_{jn}\}$ where Θ_{jn}'s are the finite and disjoint partitions that become smaller in size as $n \to \infty$ and K_n is the number of partitions for n. If $h(f_\theta, \boldsymbol{X}, A) = -1\{A \cdot \text{sign}(Y - \hat{r}(\boldsymbol{X})) = 1\} f(\boldsymbol{X}) + \log(1 + e^{f(\boldsymbol{X})})$, then h is smooth in f_θ and bounded since $\boldsymbol{X}$ and A are bounded and f_θ the deep learning structure is bounded. Thus,*

$$E\left[\max_{1 \le j \le K_n} \sup_{\theta_1, \theta_2 \in \Theta_{jn}} |h(f_{\theta 1}, \boldsymbol{X}, A) - h(f_{\theta 2}, \boldsymbol{X}, A)| \right] \to 0 \tag{3.21}$$

as $n \to \infty$.

Theorem 3.2. *Let Θ be the space of all θ parameters associated with a fixed function f. Let $h(f_\theta, \boldsymbol{X}_i, A_i) = \left[-1\{u(\boldsymbol{X}_i, A_i)\} f(\boldsymbol{X}_i) + \log(1 + e^{f(\boldsymbol{X}_i)}) \right]$. Assume the objective function of DDROWL (Eq* (3.18)*) is uniformly smooth over Θ, and the envelope of $\{h(f_\theta, \boldsymbol{X}, A) : \theta \in \Theta\}$, $H(\boldsymbol{X})$ satisfies $PH^2 < \infty$ or $P^*H^2 < \infty$ if H is not measurable. The objective function of the weighted m-out-of-n bootstrap sample is consistent for the objective function of the original*

sample for all $\theta \in \Theta$:

$$\sup_{\theta\in\Theta}\left|n^{-1}\sum_{i=1}^{n}\left(\frac{\omega_i}{m}-\hat{p}_i\right)h(f_\theta,\boldsymbol{X}_i,A_i)\right|\to 0$$

for $m,n\to\infty, m\geq n$, where $\hat{p}_i$ is the true multinomial weight (derived from data) and ω_i/m is the sampled bootstrap weight.

Proof. By Lemma 3.1,

$$\begin{aligned}
&\sup_{\theta\in\Theta}\left|n^{-1}\sum_{i=1}^{n}\left(\frac{\omega_i}{m}-\hat{p}_i\right)h(f_\theta,\boldsymbol{X}_i,A_i)\right| \\
\leq\ &\max_{1\leq j\leq K_n}\left|n^{-1}\sum_{i=1}^{n}\left(\frac{\omega_i}{m}-\hat{p}_i\right)h(f_{\theta_{jn}},\boldsymbol{X}_i,A_i)\right| \\
&\quad+\max_{1\leq j\leq K_n}\sup_{\theta\in\Theta_{jn}}\left|n^{-1}\sum_{i=1}^{n}\left(\frac{\omega_i}{m}-\hat{p}_i\right)\left(h(f_{\theta_{jn}},\boldsymbol{X}_i,A_i)-h(f_\theta,\boldsymbol{X}_i,A_i)\right)\right| \\
\leq\ &o_p(1)+n^{-1}\sum_{i=1}^{n}\max_{1\leq j\leq K_n}\sup_{\theta\in\Theta_{jn}}\left|h(f_{\theta_{jn}},\boldsymbol{X}_i,A_i)-h(f_\theta,\boldsymbol{X}_i,A_i)\right| \\
=\ &o_p(1)+o_p(1)=o_p(1)
\end{aligned}$$

The $o_p(1)$ in the third line above is based on the proof of such consistency for a fixed $\theta_{jn}\in\Theta_{jn}$ presented in Appendix B, and the inequality in the third line above is based on $\hat{p}_i, \frac{\omega_i}{m}\in[0,1] \implies \left|\frac{\omega_i}{m}-\hat{p}_i\right|\leq 1$ and the fact that the $\max\sup$ does not depend on i. The last equality (last line above) is based on Eq.(3.21). ■

3.3 Numerical Experiments

Various simulations have been conducted to study the performance of the proposed precision medicine DL approach. We examine three main aspects: low versus high dimensions, sparse versus abundant data, and linear versus non-linear boundaries. The sample size was fixed to be $n = 800$ and the dimension of covariate space varies $p = 5, 25, 100, 800$. DDROWL is compared with existing methods such as penalized linear regression with L1 penalty (ℓ_1-PLS), Q-learning with random forests (Q-RF), RWL, and AOL. We look at both linear and Gaussian

kernels for RWL and AOL and utilize the variable selection feature in AOL for high dimensions. The simulations for DDROWL models are performed in Python 3.6.8 and ran on Tesla M10 and V100-SXM2 gpu nodes. Pytorch 1.4.0 (Paszke et al., 2017) and GPytorch 0.3.5 (Gardner et al., 2018) are used for the two DDROWL models, FFNN and DKL respectively. R version 3.5.2 and R packages glmnet 2.0.18 (Friedman et al., 2010), randomForest 4.6.14 (Liaw et al., 2002), DynTxRegime 4.0 (Holloway et al., 2018) are used to run competing methods ℓ_1-PLS, Q-RF, and RWL. MATLAB 9.2 is used to run competing method AOL as in Zhou and Kosorok (2017).

All simulations are run on three sets of data: a training set used for parameter estimation, a tuning set used for hyperparameter tuning, and a test set used for overall model peformance. The training and tuning sets are split based on repeated 5-fold cross-validation (CV) on the sample size of $n_{tr} = 800$. We train the network for each CV fold with out reinitializing the weights for the neural network. We discuss and justify this decision in Section 3.5. The tuning process is a random grid search with HyperBand in Python package Ray (Liaw et al., 2018), a configuration evaluation approach. The testing set contains $n_{te} = 100,000$ samples that are set aside until networks are trained and tuned. For DKL, the weighted bootstrap only applies to the training samples, not the tuning or testing sets. The performance of learning model in both tuning and testing are determined by higher estimate value functions and lower standard deviations. The definitions of value function and its standard deviation can be found in Appendix B. Hyperparameters are chosen based on a combination of higher tuning value functions with lower standard deviations and lower tuning cost.

Next we provide specific configurations of the networks used in simulations. The activation function is chosen to be the rectified linear unit (ReLU) $a(t) = \max(0, t)$ and Adam (Kingma and Ba, 2014) with L2 regularization is the optimization algorithm. As a trending gradient-based algorithm with adaptive learning rates, Adam is efficient, easy-to-implemented, and suitable for non-stationary objectives as well as noisy and sparse gradients. Early stopping was initially applied when the tuning loss has been increasing for a consecutive number of times (i.e., patience) per frequency of recorded loss. We find that early stopping, together with random dropouts on

the input and hidden layers, put a lot of constraints on the networks so we do not use them in the final model. Weight decay in Adam and learning rate decay on plateau are applied to prevent overfitting in the training phase and we set the number of epochs to be relatively small, 500, as an indirect way of overfitting prevention. For testing phase, however, the number of epochs is set to be 1000. The number of simulations vary by the method and dimension p with faster methods and/or smaller p's ran on 500 simulations and slower methods and/or larger p's ran on 50 or 100 simulations. Tables 3.9 and 3.10 summarize the constants and hyperparameters involved in DDROWL. The computation time of all methods we consider are presented and discussed in Appendix B.

Table 3.9: Listing of constants in DDROWL simulations

Constant Name	Notation	Values
Number of Simulations	M	[50, 100, 500]
Training size (before CV split)	n_{tr}	800
Covariate dimension	p	[5, 25, 100, 800]
Testing size	n_{te}	100,000
Number of Epochs		500 (train), 1000 (test)
Activation function	$a^{[l]}$	ReLU
Optimizer		Adam with L2 regularization
Weight decay		$1e^{-5}$
CV fold	K	[5, 10]
Number of samples drawn from a latent Gaussian Process (DKL)		101
Independent treatment	$A \perp \boldsymbol{X}$	True
Treatment probability	$P(A\|\boldsymbol{X})$	$(\frac{1}{2}, \frac{1}{2})$

Table 3.10: Listing of hyperparameters in DDROWL simulations

Hyperparameter Name	Notation	Tuning Values
Learning rate	α	$[0.00001, 0.0001, 0.001]$
Number of hidden layers	L	[1, 2]
Number of hidden units	n_l	if $p = 5$: [4, 8, 16, 32, 64, 128, 256, 512] if $p = 25$: [4, 8, 16, 32, 64, 128, 256, 512, 1024] if $p = 100, 800$: [8, 16, 32, 64, 128, 256, 512, 1024, 2048]
Bootstrap sample size (DKL)	m	$[n, 5n]$

Our simulations settings are defined as follows. Vectors of clinical covariates $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_p$ are generated from independent uniform random variables $U(1,1)$. Treatments $A = \{-1, 1\}$ are independently generated from Binomial distribution with $p = 0.5$ as we assume a randomized trial setting. The outcome variable Y is normally distributed with mean $Q_0(\boldsymbol{x}, a)$ and standard deviation one, where Q_0 is the true decision boundary. We look into a total of 8 decision boundary scenarios.

3.3.1 Low-Dimensional Examples

Four low-dimensional scenarios with both linear and non-linear treatment are considered, which are based on simulation settings in Zhou and Kosorok (2017). The true decision boundaries are defined as

$$
\begin{aligned}
Q_{01}(\boldsymbol{x}, a) &= (0.5 + 0.5x_1 + 0.8x_2 + 0.3x_3 - 0.5x_4 + 0.7x_5) + a(0.2 - 0.6x_1 - 0.8x_2) \\
Q_{02}(\boldsymbol{x}, a) &= \exp[Q_{01}(\boldsymbol{x}, a)] \\
Q_{03}(\boldsymbol{x}, a) &= (0.5 + 0.6x_1 + 0.8x_2 + 0.3x_3 - 0.5x_4 + 0.7x_5) + a(0.6 - x_1^2 - x_2^2) \\
Q_{04}(\boldsymbol{x}, a) &= \exp[Q_{03}(\boldsymbol{x}, a)]
\end{aligned}
$$

The coefficients were chosen so that the Cohen's d index (the standardized treatment difference in mean outcomes) would be near 0.5 which implies medium effect size. We consider two dimensions of $\boldsymbol{X}$ in the low-dimensional setting: $p = 5$ and $p = 25$. When $p = 5$, we observe low-dimensional, abundant scenarios since the outcome is determined by two of the five covariates. When $p = 25$, the four scenarios represent low-dimensional, sparse situations where most covariates are nuisance variables and do not determine the true decision boundary. We tuned number of hidden layers and learning rate, for each scenario and sample size pair. The simulation results of low-dimensional covariates ($p = 5$ and $p = 25$) are presented in Tables 3.11 and 3.12, respectively. The highest estimated values are marked in bold for each scenario. The proposed DDROWL methods are FFNN and DKL. RWL-G does not have results when $p = 5, 25$ because they fail to finish within reasonable time limit (e.g., 11 days), and so does

AOL-VSG for $p = 25$. Note that the VS feature of AOL is turned on when there are many nuisance covariates $p = 25$. For both $p = 5, 25$, values functions of the FFNN model have lower estimated standard errors than those of AOL regardless of scenario or kernel, and FFNN has shorter computation time than AOL (see Appendix B). FFNN is outperformed by the PLS model for simple linear scenario (scenario 1) and AOL-G for complex linear (scenario 2). When $p = 5$, AOL-G proves its performance in scenario 3. For the most complicated non-linear case, FFNN has the best performance (scenario 4) compared with other existing methods. FFNN also has the second highest performance for scenario 3, which implies that neural networks have potential in nonlinear situations. The performance of DKL gets close to competing methods but does not stand out much compared with FFNN. When $p = 25$, all models perform worse than when $p = 5$ because the addition of 20 new nuisance covariates makes it a harder problem than before the addition. In sparse situations, Q-RF has the best performance for the simple and complex non-linear scenarios (scenarios 3 & 4) which could be explained by the fact that random forests are designed to detect interactions and other non-linear relationships among many nuisance variables. Our DDROWL models outperform other linear competing methods but do not outperform Q-RF for nonlinear, sparse scenarios when $p = 25$.

Table 3.11: Mean (sd) of estimated value functions for 5 covariates and 4 simulation scenarios with sample size 800

$p = 5$	Scenario 1 (Optimal 1.000)	Scenario 2 (Optimal 3.660)	Scenario 3 (Optimal 0.850)	Scenario 4 (Optimal 3.314)
ℓ_1-PLS	**0.995 (0.003)**	3.530 (0.026)	0.522 (0.035)	2.662 (0.018)
Q-RF	0.971 (0.008)	3.600 (0.011)	0.771 (0.017)	3.209 (0.019)
RWL-L	0.985 (0.011)	3.636 (0.012)	0.560 (0.022)	2.765 (0.075)
AOL-L	0.991 (0.008)	3.629 (0.022)	0.560 (0.030)	2.666 (0.032)
AOL-G	0.987 (0.012)	**3.640 (0.025)**	**0.824 (0.020)**	3.192 (0.033)
FFNN	0.986 (0.005)	3.609 (0.012)	0.793 (0.015)	**3.244 (0.018)**
DKL	0.978 (0.006)	3.608 (0.017)	0.738 (0.024)	3.121 (0.050)

Table 3.12: Mean (sd) of estimated value functions for 25 covariates and 4 simulation scenarios with sample size 800

$p = 25$	Scenario 1 (Optimal 1.000)	Scenario 2 (Optimal 3.660)	Scenario 3 (Optimal 0.850)	Scenario 4 (Optimal 3.314)
ℓ_1-PLS	**0.989 (0.004)**	3.553 (0.025)	0.515 (0.023)	2.648 (0.032)
Q-RF	0.958 (0.011)	3.582 (0.022)	**0.701 (0.032)**	**3.123 (0.038)**
RWL-L	0.957 (0.016)	3.616 (0.022)	0.535 (0.028)	2.689 (0.042)
AOL-VSL	0.988 (0.010)	**3.617 (0.024)**	0.554 (0.030)	2.655 (0.031)
FFNN	0.959 (0.008)	3.565 (0.021)	0.615 (0.015)	2.994 (0.034)
DKL	0.896 (0.016)	3.479 (0.031)	0.570 (0.021)	2.757 (0.037)

3.3.2 High-Dimensional Examples

We study four non-linear high-dimensional scenarios with $p = 100$ and $p = 800$, the first two of which are sparse and the last two of which are abundant:

$$
\begin{aligned}
Q_{05}(\boldsymbol{x}, a) &= 1 + 0.6x_1 + 0.8x_2 + 0.3x_3 - 0.5x_4 + 0.7x_5 \\
&\quad + a[0.45 - 0.1x_1^2 - 0.2x_2^2 + 0.3x_3^2 + 0.2x_4^2 - 0.9x_5^2] \\
Q_{06}(\boldsymbol{x}, a) &= \exp[Q_{05}(\boldsymbol{x}, a)] \\
Q_{07}(\boldsymbol{x}, a) &= [0.5 + 0.6(x_1 : x_{10}) + 0.8(x_{11} : x_{20}) + 0.3(x_{21} : x_{30}) - 0.5(x_{31} : x_{40}) \\
&\quad + 0.7(x_{41} : x_{50}) + 0.5(x_{51} : x_{60})) + 0.4(x_{61} : x_{70}) - 0.4(x_{71} : x_{80}) \\
&\quad + 0.2x_{81} - 0.9x_{82}] \\
&\quad + a[0.6 - 0.1(x_1^2 : x_{15}^2) - 0.2(x_{16}^2 : x_{30}^2) + 0.3(x_{31}^2 : x_{45}^2)] \\
Q_{08}(\boldsymbol{x}, a) &= \exp[Q_{07}(\boldsymbol{x}, a) - 4]
\end{aligned}
$$

Here, $\gamma(x_a : x_b) = \gamma x_a + \gamma x_{a+1} + \ldots + \gamma x_b$ for cleaner notation. The six unique coefficient values of $\boldsymbol{X}$ main effects are kept the same as those in scenario 4. For scenarios 7 and 8, we added extra coefficients to make the boundary more complicated with more $\boldsymbol{X}_{ji}$ variables involved. The same training/tuning/testing process is applied to the high-dimensional scenarios.

Simulation results of high-dimensional covariates are summarized in Tables 3.13 and 3.14. For $p = 100, 800$, RWL-G and AOL-VSG continue to be slow and fail to give estimates within

Table 3.13: Mean (sd) of estimated value functions for 100 covariates and 4 simulation scenarios with sample size 800

$p = 100$	Scenario 5 (Optimal 1.317)	Scenario 6 (Optimal 5.084)	Scenario 7 (Optimal 1.131)	Scenario 8 (Optimal 3.714)
ℓ_1-PLS	1.210 (0.012)	4.609 (0.027)	1.091 (0.017)	2.614 (1.265)
Q-RF	**1.224 (0.011)**	**4.731 (0.040)**	1.081 (0.033)	3.165 (0.451)
RWL-L	1.207 (0.022)	4.605 (0.066)	1.103 (0.028)	3.095 (0.463)
FFNN	1.109 (0.018)	4.415 (0.044)	**1.111 (0.001)**	**3.523 (0.486)**
DKL	1.138 (0.027)	4.578 (0.031)	1.102 (0.007)	3.070 (0.394)

Table 3.14: Mean (sd) of estimated value functions for 800 covariates and 4 simulation scenarios with sample size 800

$p = 800$	Scenario 5 (Optimal 1.317)	Scenario 6 (Optimal 5.084)	Scenario 7 (Optimal 1.131)	Scenario 8 (Optimal 3.714)
ℓ_1-PLS	**1.208 (0.016)**	4.628 (0.027)	**1.072 (0.020)**	1.687 (0.748)
Q-RF	1.202 (0.039)	**4.716 (0.043)**	1.020 (0.115)	1.795 (0.488)
FFNN	1.078 (0.200)	4.290 (0.635)	0.892 (0.452)	**2.175 (1.111)**
DKL	1.078 (0.199)	4.291 (0.636)	0.892 (0.452)	**2.175 (1.111)**

reasonable time limit; AOL-VSL is able to complete but produces negative values which means that it is not applicable to high-dimensional data either. When p reaches 800, RWL-L becomes slow as well, which results in only two remaining competing methods. In the sparse situation (scenarios 5 & 6), Q-RF and ℓ_1-PLS perform well for $p = 100, 800$ given their ability to reduce feature space efficiently even with higher dimensions and/or larger sample sizes. Our FFNN method stands out with the highest value estimates for both $p = 100$ scenario 8 and $p = 800$ scenario 8 by a large margin compared with other methods. FFNN also has the highest value estimate for $p = 100$ scenario 7. DKL is able to catch up quickly with FFNN in the high-dimensional setting and match the value estimate with FFNN for $p = 800$ which means that they both find similar decision rules. We examine more of DKL's performance in complex imaging data in Section 3.4 the clinical application. Both DDROWL methods have relatively large standard errors for $p = 800$, and we speculate that this is due to our relatively small sample size and large amount of nuisance variables for deep neural networks to generate more stable results.

3.4 Application to Medical Data and Imaging

This is a clinical application of DDROWL using the National Alzheimer's Coordinating Center (NACC) database (Beekly et al., 2004). We acquire three sets of observational data from the NACC database: i) the Uniform Data Set (UDS), a longitudinal data set of detailed clinical records of subjects from 19 Alzheimer's Disease Centers (ADC) funded by the National Institute on Aging (NIA) during the time period 2005-2019, ii) MRI, structural MRI scans in NIFTI and DICOM formats taken within one year of a UDS visit , and iii) Imaging Data (ID), the summarized data that contains descriptive statistics of the MRI scans and information that links the MRIs with the UDS data. The following sections of UDS data are requested: A1 demographics, A4 medication, A5 health history, B1 physical, B4 CDR® (Clinical Dementia Rating), B9 clinician judgement of symptoms, C1/C2 neuropsychological battery, D1 clinician diagnosis, and genetic data such as APOE (Apolipoprotein E) genotype and status of the e4 allele. The raw UDS data have 19,889 observations and 495 variables of 4,036 unique individuals. More information about UDS can be found in Besser et al. (2018). The first three sections of ID data are requested: MRI scan date data, MRI scan type and series-associated data, as well as MRI calculated summary data. The raw ID dataset has 5,644 observations and 177 variables of 4,036 unique individuals.

Preprocessing. The NACC database has been constantly updated and modified throughout the past 15 years. Different centers have different data collection methods and policies during different time periods which might not conform to each other. Such conformity issue could create noisy heterogeneity in a bad way, and we believe data quality trumps data quantity. We apply some stringent inclusion and exclusion criteria to keep the multi-center, multi-stage data relatively clean. We find that there are more missing data in earlier form versions and later visits, and the most information was collected at the initial visits. Under the assumption that we are interested in baseline information, only observations of the first visit of each subject with form version 3 (the latest version) are included. For categorical variables, categories such as unknown (e.g., 9, 99, 999, 8888, 9999 values), not applicable (form submitted did not collect such data or

a skip pattern precludes such responses), or left blank are considered missing. For continuous variables, indicators of unknown, not assessed, and not available (e.g., -4, 888.8) and extreme values outside of the normal range (e.g., height more than 80 inches) are considered missing.

Our outcome Y is change in cognitive status. Cognitive status at UDS visit is a variable in UDS which classifies the cognitive status into normal cognition, dementia, MCI (mild cognitive impairment), and impaired-not-MCI. We dichotomize this variable by combining the last three categories into not normal to mitigate the unbalanced distribution. We look at cognitive status at the initial visit and the closest visit to one year after the initial visit. Subjects who do not have the latter visit are excluded. The analysis outcome Y is defined as 1 if the subject stays normal or was not normal at the initial visit but becomes normal at the visit a year later, and 0 if the subject stays not normal or was normal at the initial visit but becomes not normal at the a year later. This definition is more balanced than a three-category definition of worse, same, and better. The treatment A is chosen among binary modifiable variables at the initial visit, the majority of which are medications and habitual variables such as smoking and alcohol usage. We choose A as current use of any type of antihypertensive or blood pressure medication because it is a reasonable risk factor and is well-distributed between the two categories. For example, a 5/95 distribution would be considered imbalanced, whereas a 30/70 or 40/60 distribution would be considered balanced. This is also the reason that categorical covariates with low proportion ($< 5\%$) in any subcategories are excluded. Because the UDS dataset has many forms/variables that contribute to the assessment of the subject's cognitive status, covariates that have moderate to high estimated Pearson correlations (> 0.5) with the outcome variable are excluded to avoid multicollinearity. Covariates with high estimated Pearson correlation (> 0.8) with other non-outcome covariates are excluded as well (e.g., height/weight and body mass index, and various CDR® scores). The ID data, data containing information that links with UDS and MRI, are processed similarly as the UDS data, such as removing severe missing data and multicollinearity. Subjects could have multiple MRI scans and the MRI scan closest to the initial visit of each subject is used. If there are multiple MRIs on the same day, we choose the latest

one. We force one MRI per subject because we are only interested in baseline covariates and to keep input dimension the same.

The preprocessed ID and preprocessed UDS are merged by unique NACC subject ID, their ADC center, and visit year. Only complete cases are used because imputation on such multi-center observation is often unreliable or needs extremely careful manipulation. More details about the preprocessing of certain variables can be found in Appendix B. In the preprocessed data (before merging with MRI scans but after merging UDS with ID), there are 424 observations and 50 variables collected from 12 ADCs spanning from 2015 to 2019. Among the 424 subjects, 48% have better or maintain normal cognitive status and 48% currently use antihypertensive or blood pressure medication at the initial visit.

After UDS and ID are merged, we preprocess the MRI scans before matching with the merged UDS-ID data. There are a total of 5,616 MRI sessions available, where one subject could have multiple sessions at different times. Each session contains multiple DICOM files, with one file representing one MRI slice. The DICOM format is preferred because it contains image information such as slice position and sequence type in the headers. We extract the MRI slices from each subject's MRI session in a compressed folder, remove slices without series description or image position because series description informs the sequence type of the MRI scan and image position helps sort the slices in the right order, and select every 5th slice among the middle 150 T1 slices. T1 sequence is determined by keyword, not by imageology. We discard end slices because they contain less useful information about the brain. Since the consecutive slices differ by a matter of milliseconds, we select every 5th slice from the 150 middle slices to save space and maintain the same reasonable image dimension for every subject. Each slice is resized to the dimension of 64×64 and standardized to mean 0 and standard deviation 1 for comparable values. The preprossed MRI data are merged to the UDS-ID data by file locator information so only subjects who have qualified UDS, ID, and MRI data are included, resulting in a sample size of 186. The dimension for the preprocessed MRI data is $186 \times 30 \times 4096$ where $30 = 150/5$ and $4096 = 64 \times 64$. The dimension for the preprocessed UDS-ID data is 186×48.

As mentioned above, we apply a strict inclusion criteria to make sure the input data for the DL models are relatively clean and conformative. The MRI data are only lightly processed to preserve the original values but could be piped through more systematic image processing tools; we discuss this more in future research.

Models and Methods. Because of the large dimension difference, it is not appropriate to directly apply FFNN and DKL models used in Section 3.3 to the UDS-ID-MRI data. We apply the pretrained ResNet34 model (He et al., 2016) to the preprocessed brain images first; only the convolutional and pooling layers. ResNet34 is a 34-layer convolution neural network (CNN) based on the deep residual learning framework, which increases the ease of learning by approximating the residual functions instead of the original functions. The residual learning is achieved by shortcut connections and is adopted to every few stacked convolutional layers with fixed dimensions (He et al., 2016). It is not always better to have deeper models because gradients could shrink to zero very quickly and stop updating the weights. ResNet34 is able to address the vanishing gradient problem by letting gradients take shortcuts when transmitting input data thus stopping the information loss (Talo et al., 2019). The ResNet34 model is pretrained on the renowned ImageNet database which has resources of over 10 million images over 20,000 subcategories (or synsets) by 2019 and is commonly used as the first step to train deep architectures. We apply a pretrained model instead of training our own structure because the lower-level representations extracted from the earlier layers of existing models are generally transferrable across images. ResNet34 is chosen as the pretrained model because it has lower model complexity and relatively low top-1 and top-5 errors on the ImageNet data compared wth other famous deep learning architectures such as AlexNet or VGG. Top-1 error means the proportion of test images whose true label does not match with the prediction class with the highest estimated probability. Top-5 error means the proportion fo test images whose true label is not among the 5 prediction classes with the top 5 highest estimated probabilities. ResNet34 has also been shown to work well with MRI. Talo et al. (2019) used it to detect brain abnormality and reached a 5-fold classification accuracy of 100% over 600 MRIs. This shows that ResNet34's

learning from ImageNet can be transferred to learn MRI. This transfer learning technique enables us to apply DL to smaller image datasets while standing on the shoulder of a giant.

Transfer learning can also be regarded as a feature selection tool because images often contain a large amount of nuisance pixels. The outputs of ResNet34 prior to the dense layers have a lowered dimension of 1000, much smaller and more extracted than the original dimension. The MRI data would dominate the dimension if we fed them together with UDS-ID directly into a DL architecture. An alternative to transfer learning is applying unsupervised learning such as autoencoder (AE) to the MRI data. AE is a good dimension reduction method but the encoded outputs are sometimes not necessarily good representations of the original input. Evidence shows that introducing labels earlier helps with dimension reduction as well as prediction, which is why we choose to use transferring learning on the brain imaging. After ResNet34 is applied, we combine the extracted MRI features with preprocessed UDS-ID data and apply a FFNN and a DKL model to estimate the optimal decision rule for antihypertensive medication. The rest is similar to the FFNN and DKL in simulations. Using symbols from ENNUI (an element neural network user interface) (Michel, 2020), Figure 3.7, illustrates the deep learning structure of our DDROWL models designed for the NACC data. We denote our proposed DDROWL methods as ResNet34 + FFNN and ResNet34 + DKL, respectively.

We freeze the weights of ResNet34 and only tune the dense layers. Since it is impossible generate a separate data set to test performance in clinical application, nested 10-fold cross validation is applied. We apply the first 10-fold CV to the combined NACC data to set aside test sets and apply the second 10-fold CV to split each of the nine folds into training and tuning sets. The mean and standard deviation of estimated value functions are calculated from the 10 tuning sets (10 nested folds) to tune three hyperparameters: number of hidden layers, number of hidden units, and learning rate. A summary of constants and hyperparameters used in this clinical application is presented in Table 3.15. The numbers of hidden layers and hidden units refer to the second part of the DDROWL architecture on the combined NACC data. After the hyperparameters have been tuned, we train the two DRROWL models on the 10 combined

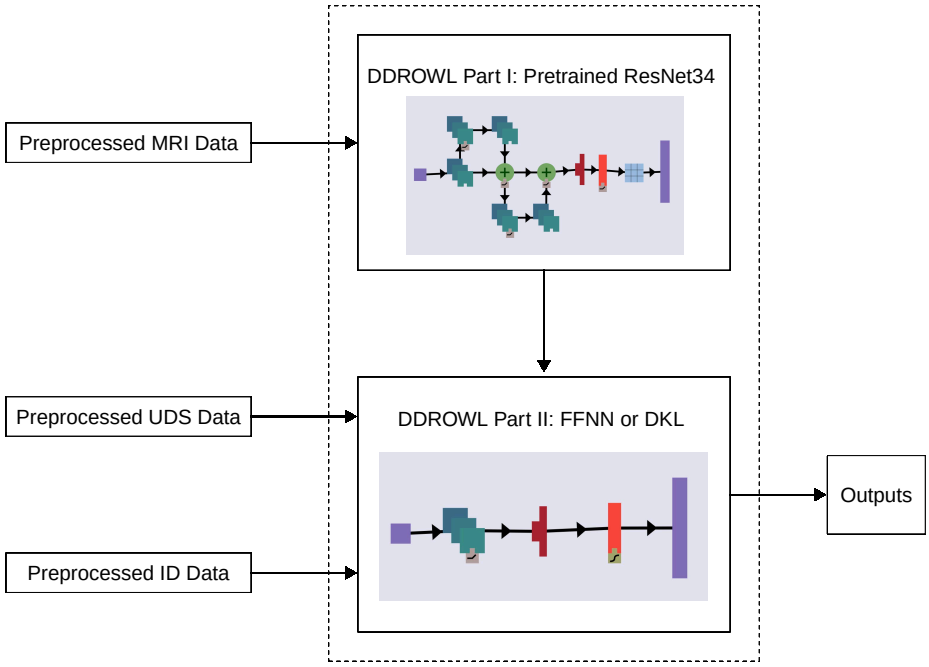

Figure 3.7: Diagram of the DDROWL architecture for the NACC data application

training and tuning sets and calculate the mean and standard deviation of estimated value functions from the 10 test sets.

Competing methods are chosen to be ℓ_1-PLS, RF, RWL with linear kernel, and RWL with Gaussian kernel. For RWL models, a nested 10-fold CV is applied to tune two hyperparameters, bandwith parameter and penalty tuning parameter. Four values are given as candidates for both: e^{-1}, e^0, e^1, e^2. Due to the large dimension of the MRI data, existing methods such as ℓ_1-PLS, Q-RF, and RWL use UDS-ID data as inputs whereas the two DDROWL models use merged NACC data as inputs as well as the UDS-ID data for comparison. All analyses in this section are performed in the similar language, packages, and platform as those in simulations Section 3.3: Python 3.6.8, Tesla M10 and V100-SXM2 gpu nodes, Pytorch 1.4.0, GPytorch 0.3.5, R 3.6.1, glmnet 2.0.18, randomForest 4.6.14, and DynTxRegime 4.2. In addition, the pretrained ResNet34 model comes from the torchvision subpackage of Pytorch.

Table 3.15: Listing of constants and hyperparameters in DDROWL clinical application

Constant Name	Notation	Values
Total Sample Size	n	186
Dimension of MRI		30×4096
Dimension of UDS-ID		48
CV fold	K	10
Training size	n_{tr}	150, 151, 152 (depending on rounding)
Tuning size	n_{tu}	16 or 17 (depending on rounding)
Test size	n_{te}	18 or 19 (depending on rounding)
Number of Epochs		1000 (DKL) or 5000 (FFNN)
Batch size		64
Activation function	$a^{[l]}$	ReLU and tanh
Optimizer		Adam with L2 regularization
Bootstrap sample size (DKL)	m	Same as n
Number of samples drawn from a latent Gaussian Process (DKL)		101
Hyperparameter Name	**Notation**	**Tuning Values**
Learning rate	α	$[0.001, 0.01]$
Number of hidden layers	L	[1, 2]

Number of hidden units	n_l	if 1 layer:
		[8, 16, 32, 64, 128, 256, 512, 1024]
		if 2 layers:
		[[1024, 512], [512, 256], [256, 128], [128, 64], [64, 32], [32, 16], [16, 8]]

Results. Results of the clinical application of DDROWL to the NACC data are contained in Table 3.16. During the tuning phase, we find that learning rates do not affect tuning results as much as number of hidden units or layers. Unlike simulations, we cannot derive the optimal value for the NACC data and can only compare performance relative to the models selected. Despite the high standard deviations (SDs), ℓ_1-PLS has the highest estimate value among the methods that use UDS-ID as inputs. This is not surprising because of its penalization feature and versatility in small to medium data, as demonstrated in simulations. For both DDROWL methods, the UDS-ID-MRI inputs give higher estimated value functions than the UDS-ID only inputs, which indicates the additional MRI data are beneficial to the search for decision support. Increment is higher for ResNet34+DKL than ResNet34+FFNN. The ResNet34+DKL model with both UDS-ID and MRI as input data have the highest estimate value function by far, with

Table 3.16: Estimated value function of change in cognitive status between initial visit and the next visit at least a year later and computation time

Model	Learning Rate	Hidden Units	Mean (SD)	Input	Computation Time (active)
ℓ_1-PLS			0.709 (0.213)	UDS-ID	4s
RF			0.690 (0.206)	UDS-ID	4s
RWL-L			0.642 (0.193)	UDS-ID	100s
RWL-G			0.644 (0.194)	UDS-ID	2,763s
ResNet34 + FFNN	0.001	[1024]	0.654 (0.187)	UDS-ID	969s
ResNet34 + FFNN	0.001	[128, 64]	0.664 (0.161)	UDS-ID-MRI	27,724s
ResNet34 + DKL	0.001	[256]	0.659 (0.234)	UDS-ID	1,350s
ResNet34 + DKL	0.001	[64, 32]	**0.876 (0.115)**	UDS-ID-MRI	21,620s

more than one SD higher than the next highest value function. All SDs are relatively large compared to the scale of value functions, and we speculate this is due to the small sample size and the underlying distribution of the data. Three out of four DDROWL methods have lower SDs than those of existing methods. ResNet34+DKL has the smallest SD using all data but the highest SD using only UDS-ID data, suggesting that it has less variation in estimation when inputs are larger and more complex. DKL model fits the NACC data more than FFNN, whereas FFNN fits the simulation data more than DKL. We conclude that the structure of DDROWL is a key performance factor. We recommend choosing an appropriate DL model based on domain knowledge and a good understanding of the input data. The time listed in Table 3.16 is active computing time assuming all parallel programs work at the same time with no waiting. The time spent in reality is longer than those listed due to resource and priority queues. ℓ_1-PLS and Q-RF are still the fastest methods, followed by RWL and DDROWL models with UDS-ID as inputs. Both DDROWL models consider the same amount ot tuning parameter candidates and the ResNet34+DKL model has shorter computation time. RWL models are slow due to the search for hyperparameters and DDROWL models with UDS-ID-MRI data are slow due to the larger sample size. In general, computation is much faster in this application than simulation because of no repetition and small sample size but the general orders of magnitude stay the same for all chosen models.

3.5 Discussion

Three major contributions of the proposed DDROWL method can be identified: 1) We are able to combine the best of three worlds: deep learning, doubly robustness to misspecification and efficiency, and the proven performance and asymptotic properties of residual weighted learning, to develop a robust, efficient, and flexible machine learning tool; 2) With the implementation of deep neural networks, DDROWL is able to expand the influence of precision medicine to high-dimensional data with great flexibility and computation power; 3) We confirm that it is possible to use deep learning models to develop decision support from abundant high-dimensional data and the classical PLS and Q-learning with RF are eminent, computationally fast methods for small to medium, or large and sparse data.

Overall, ℓ_1-PLS dominates the simple, linear decision boundary (scenarios 1 & 5 and UDS-ID data) for various dimensions. AOL-G performs well in scenarios 2 & 3 when dimension is low ($p = 5$), which can be deemed as medium decision boundaries (as opposed to simple boundaries such as scenarios 1 & 5 or complex boundaries such as scenarios 4, 7, & 8). RWL and AOL have trouble with higher dimensions ($p \geq 25$), especially with Gaussian kernels which can be very slow in computation. Q-RF shines with its built-in variable selection feature for difficult, low-dimensional, non-linear data (scenario 4) and medium, high-dimensional, non-linear (scenarios 5 & 6) decision boundaries as well as UDS-ID data. In simulations, FFNN outperforms AOL and Q-RF when data are abundant and the decision boundary is complex and nonlinear (scenarios 4 & 8) regardless of dimension, whereas DKL slowly catches up with FFNN for high-dimensional data. In clinical applications, DKL outperforms FFNN and climbs to the top when input is a mix of extracted images and clinical data. DDROWL methods are able to handle abundant data with complex, non-linear relationships, and depending on the model choice, we even see them perform better than ℓ_1-PLS and Q-RF. The intuition behind this finding is that deep learning has proven performance in areas such as computer vision and natural language processing, both of which have non-linear complexity and abundance in their input data. Images, in particular, do not contain key information in a few pixels but a group of pixels and relative values of pixels

may contain useful information as well as the actual pixel values. Although less DL research has been conducted using the typical sample-by-feature data (such as our simulated data and clinical data), our results confirm this advantage of deep learning methods and show that we can use it to improve the flexibility and complexity of precision medicine models. Despite its promising performance, there is a lot of room to improve DDROWL. The distance between the highest empirical value (2.175) and the theoretical optimal value (3.714) in simulations is large relative to other scenarios, and ResNet34+FFNN fails to outperform ℓ_1-PLS using combined NACC data in clinical applications. The time it takes DDROWL to find the optimal decision rule could also be improved (see Appendix B) by running different simulation copies in parallel. One could argue whether the improvement in value function for the DDROWL is worth the time spent on training and tuning. Additionally, there is value in visualizing the estimated decision rules, yet it is not always approachable for machine learning models. For random forests, the variable importance would be informative of important factors in the decision rule. For deep learning, it is possible to inspect nodes in the networks, identify region of interest (ROI), or create heatmaps from the MRI scans to help interpret the decision support.

In addition to the results, our work provides valuable insights about deep learning architecture. In our experiments, adding hidden units tend to affect performance (in terms of higher value function) more than adding hidden layers or changing learning rates, and thus we consider many hidden unit values but only focus on one or two layers and a few learning rates. Mini-batch gradient descent speeds up computation. There is a trade-off between batch size and model performance because smaller batch size is easier to compute but does not carry enough information for the model to learn useful representations. It is reassuring to see that hyperparameters with higher estimated value functions usually have lower tuning costs. We observe mild to strong negative correlation between them with Pearson correlation as high as around -0.95 to as low as -0.20 for the simulated datasets. During cross validation in the training and tuning phase, recycling the tuning data into training data is expected to cause overfitting because the model has seen the tuning data in the previous CV fold. Based on the test results provided in the previous

section, however, we discover that there is a net benefit even given the potential of overfitting. We think it is because it is similar to data augmentation, where we use more data replicates as we tune through each CV fold. Data recyling elongates the training/tuning period and increases the training sample size and provides better tuned hyperparameters for the testing stage. Future research is given in Chapter 5.

CHAPTER 4: RISK-ADJUSTED INCIDENCE MODELING ON HIERARCHICAL SURVIVAL DATA WITH RECURRENT EVENTS

4.1 Introduction

Right-censored data such as patient encounters in healthcare settings often have a multilevel structure where the assumption of independent observations is violated. Moreover, there is a need for many health institutions to monitor survival events which could occur repeatedly for patients. We expand on existing statistical approaches in mixed effect survival models and resampling methods to provide program specific predictions with confidence intervals (CIs) and detect excessive or fewer-than-expected events at the highest level of hierarchy.

In the world of infection control, there is no perfect measure of risk-adjusted rates of healthcare-associated infection (HAI) (Gustafson, 2006). The primary summary statistic used by the National Healthcare Safety Network (NHSN) to track HAI is standardized infection ratio (SIR). The SIR is a ratio of the observed number of infections divided by the expected number of infections, the latter of which is a summation of the number of patients weighted by the national standard stratum-specific rates (Center for Disease Control, 2018; Gustafson, 2006). In addition to providing absolute numbers, this rate is adjusted for known risk factors associated with infections, such as patient characteristics or geography. SIRs have demonstrated that risk adjustment methods are promising, but the ratio is not ideal for comparisons between hospitals or across time (Delgado-Rodríguez and Llorca, 2005). Infection is one of the examples of lifetime data where time and censoring play an important role in risk adjustment that needs appropriate methodology. Inspired by this clinical background, we aim to develop a more powerful risk-adjusted model that takes into account time-varying covariates and complex right-censoring data.

We aim to provide better, more appropriate methodology to predict and monitor survival events in complex situations. This paper has two main goals and contributions: i) To develop a risk-adjusted model for predicting events for right-censored survival data and incorporating complex but common situations such as multilevel hierarchical grouping and recurrent events; ii) To test the validity and practicality of this model on bacterial infection incidence using U.S. nation-wide data from the cystic fibrosis (CF) foundation. The rest of the paper is organized as follows: We generalize the problem and lay out modeling methods in Section 4.2. In particular, we review existing survival models and explain why frailty model is selected in subsection 4.2.1, set up our problem of interest in subsection 4.2.2, and provide explanation of the parameter estimation as well as its variability estimation in subsections 4.2.3 & 4.2.4. Numerical experiments are explored in Section 4.3 to learn the performance of our risk-adjusted models. In Section 4.4, we introduce the clinical background and implement our methods in the CF data, illustrating each step of the analysis process: preprocessing, variable selection, multiple imputation, survival modeling and results. Finally, strengths, limitations, and future research are discussed in Section 4.5.

4.2 Methods

4.2.1 The Frailty Model

When the survival data contain multilevel hierarchies and recurrent events, the standard Cox proportional hazards model is no longer suitable because it is designed to assess time to first event. Alternative models need to be considered to accommodate for these special features. The Andersen and Gill (AG) model (Andersen and Gill, 1982) is a popular choice that extends the common Cox model to repeated time-to-event data. However, Andersen-Gill assumes that the recurrent event times are independent conditioning on time-varying covariates (Amorim and Cai, 2015) and the baseline intensity is the same across all recurrent events (Yang et al., 2017). These assumptions do not necessarily hold in the hierarchical data setting we are interested in. Other models that also address multiple failure times include Prentice-William-Peterson (PWP) and Wei-Lin-Weissfeld (WLW) are robust and well-developed but they do not explore

the relationships between failures (Wei et al., 1997). There are two leading ways to model both recurrent events and hierarchical structure: 1) A marginal approach: Fit a generalized estimating equation (GEE) to estimate the parameters of the marginal cumulative incidence function where the correlated observations are taken into account by the variance with a sandwich estimator (Logan et al., 2011). This approach focuses more on the marginal covariate effect on failure risks by adjusting the variance but not necessarily the coefficients; 2) A conditional approach: Fit a mixed effect survival model where random effects can be incorporated as a frailty model and the baseline hazard varies by the group variable(s), thus a multiplicative effect on the hazard. This approach adjusts for both coefficients and variance. Conditioning on the random effects, it is assumed that the intensity function of each subject follows the Andersen-Gill model (Wei et al., 1997); thus this approach can be deemed as a mixed effect AG model.

We choose the second approach because we believe the inclusion of random effect improves the model fit and CIs. The random effects describe the term 'frailty', which is the excessive risk for distinct grouping variables (Therneau et al., 2003). In general, the hazard of a frailty model for subject i in group j is

$$\lambda_{ij}(t) = \lambda_0(t)e^{\boldsymbol{U}_{ij}\beta}\omega_{ij} = \lambda_0(t)e^{\boldsymbol{U}_i\beta+\boldsymbol{W}_{ij}b}$$

where $\omega_{ij} = \exp(\boldsymbol{W}_{ij}b)$ is the unmeasured frailty, $\boldsymbol{U}_i$ denotes covariates for the ith subject, $\boldsymbol{W}_{ij} = 1$ if subject i belongs to group j, and β, b are the fixed and random effects (Pickles and Crouchley, 1995). The frailty term accounts for variation in the risk that are not captured solely by the covariates, and the frailty model assumes that the time increments are uncorrelated once we adjust for both the covariates *and* random effects (Pickles and Crouchley, 1995; Amorim and Cai, 2015). Our setup meets this assumption. Intuitively speaking, we are willing to assume that there are unobserved information (i.e., the random effects) that explains the heterogeneity in the data which cannot be explained solely by the observed covariates. The linear mixed effect approach follows the partial likelihood approach of Cox (1975) whose key advantage is that the

the maximum partial likelihood estimation of the covariates does not need the baseline hazard function to be specified and is unbiased and asymptotically normally distributed under mild conditions (Yau, 2001).

4.2.2 The Setup and Overview

Assume we have two hierarchical levels in the survival data. This makes it a little bit more complicated but more generalizable than one-level hierarchy, thus more room to explore and discover. Let $j = 1, \ldots, N$ represent the unique level-1 grouping variable (the highest hierarchical level), $i = 1, \ldots, c_j$ represent the unique level-2 grouping variable within the jth level-1 grouping variable, and $k = 1, \ldots, m_{ij}$ denote the observations in the ith level-2 group and jth level-1 group. To make it easier to comprehend, we set up notations within the framework of the clinical cystic fibrosis (CF) data that we will officially introduce in Section 4.4. For CF data, an event is bacterial infection incidence, which could be recurrent given a long enough washout period (e.g., 2 years). The level-1 grouping variable is a CF program j, level-2 is the CF patient i, and number of encounters for a CF patient at a CF program is k. CF program is the highest level because they accept CF patients who have different numbers of events. Our model allows nested and crossed random effects. An example of the crossed random effects would be the case where CF patients do not necessarily go to only one program given that they could relocate so we specify two random effects instead of two nested random effects. Because there might be recurrent events, m_{ij} can be greater than 1. Covariates (time-varying and/or time-invariant) for the jth level-1 group, ith level-2 group, and kth encounter are denoted by $\boldsymbol{Z}_{jik}(t) \in \mathbb{R}^p$ at time t where p is the total number of covariates of interest. The at-risk time interval is denoted by T_{jik}. For CF, the start date is the first date that the patient has not had any bacterial infection for 2 years during the time period of interest and end date is date of event, lost to follow-up date, or the cutoff date (last day of time period of interest). We propose the true intensity process of the counting process to be

$$R_{0j} = \sum_{i=1}^{c_j} \sum_{k=1}^{m_{ij}} \int_0^{T_{jik}} e^{\boldsymbol{Z}_{jik}(s)\beta_0 + b_{ij}} d\Lambda_0(s) \tag{4.22}$$

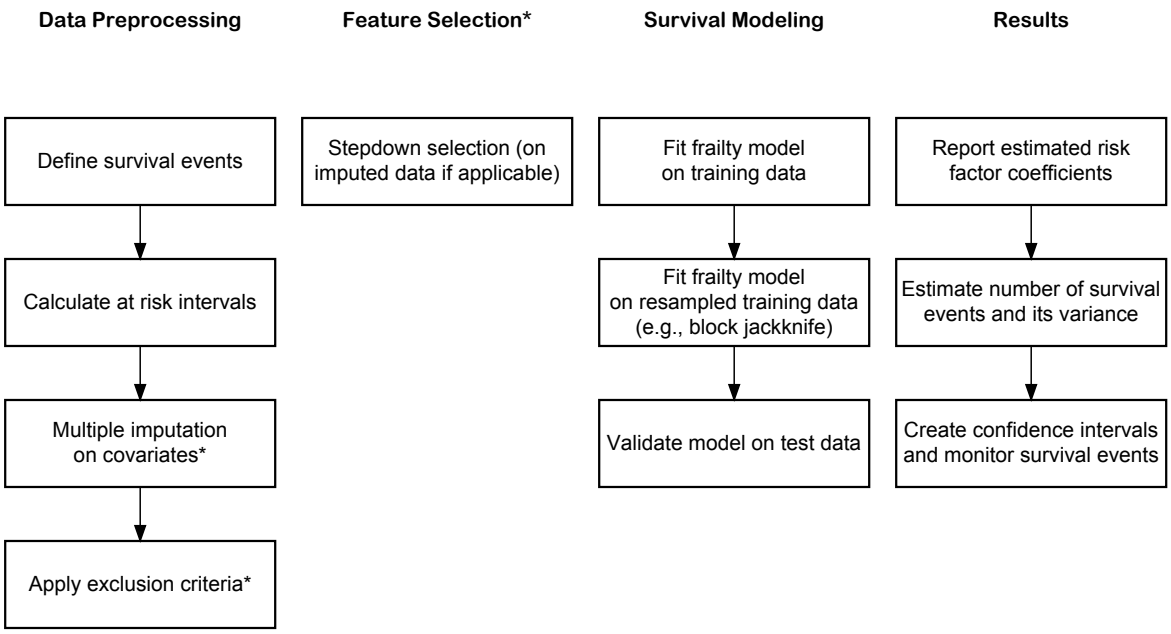

Figure 4.8: Diagram of the risk-adjusted survival analysis (from left to right). Steps marked with asterisks are optional but recommended if applicable.

for each program j (Lawless, 1987). The random effects b_{ij} are i.i.d. zero-mean Gaussian random variables. The covariate effects β_0 will be estimated by partial likelihood on the full n dataset, i.e. all events for all patients at all programs. Hence, $n = \sum_{j=1}^{N} \sum_{i=1}^{c_j} m_{ij}$. With the covariate effect estimator, $\widehat{\beta}_n$, the hazard will be estimated using an extension of the Breslow estimator (Lin, 2007). Point estimates by themselves are not useful. The next section will elaborate more on estimating the variability of the estimators with the block jackknife method (Ma et al., 2005). We will also provide a two-sided CI for various significance levels for the predicted survival events at each program. This interval is calculated based on our estimated intensity process and a Poisson process that describes the survival events. At the end, we validate the model estimated in the training period with a separate test data. Note that we use the words “test” and ”validate” interchangeably throughout the paper. Later, we compare the estimated, risk-adjusted CI with the true, observed survival cases to evaluate the accuracy of the model in predicting survival events. Figure 4.8 describes the pipeline of our methods and data analysis presented in the next couple of sections.

4.2.3 Estimation of Parameters and Their Variability

Missing data have become a universal issue in data analysis due to nonresponses and data collection mistakes or simply because the information we want is not available. Since it could cause problems if left untended or removed completely, many researchers resort to imputation and more particularly, multiple imputation (MI), which is used in our survival analysis before modeling.

Recall that β is a p-dimensional vector where p is the number of covariates in the data. Assume there are M copies of MI and let $\hat{\beta}_{nl}$ be the estimated coefficient from the fraily model for the lth MI dataset where $l = 1, \ldots, M$ trained on a data size of n. Pooling the multiple imputed $\hat{\beta}_{nl}$ together to determine important risk factors, we apply Rubin's rule (Rubin, 2004) as follows. The pooled beta estimate is the average over $\hat{\beta}_{nl}$'s across M multiple imputations

$$\bar{\hat{\beta}}_M = \frac{1}{M} \sum_{l=1}^{M} \hat{\beta}_{nl}.$$

The variance of this pooled estimate is

$$Var(\bar{\hat{\beta}}_M) = \frac{1}{M} \sum_{l=1}^{M} \text{Cov}(\hat{\beta}_{nl}) + \left(1 + \frac{1}{M}\right) \frac{\sum_{l=1}^{M} (\hat{\beta}_{nl} - \bar{\hat{\beta}}_M)^T (\hat{\beta}_{nl} - \bar{\hat{\beta}}_M)}{M - 1}.$$

where the first term is the average of the variance-covariance of $\hat{\beta}_{nl}$ estimate in the fitted frailty model (within MI) and the second term is the variance across the MI estimates $\hat{\beta}_{nl}$'s (across MI). Test statistic for the pooled estimate is defined as $T_{pooled} = \frac{\bar{\hat{\beta}}_M}{\left(Var(\bar{\hat{\beta}}_M)\right)^{1/2}}$ and the p-value follows a two-sided student t-distribution.

Another estimator we need is the estimated hazard function. We extend the Breslow estimator (Lin, 2007) to our data structure and define the hazard function estimator as the empirical survival cases across all levels of hierarchies in the data at a certain time point adjusted by the at-risk population whose at risk interval covers the current time point. Mathematically, the baseline

hazard function estimator is

$$d\hat{\Lambda}_{nl}(s) = \frac{\sum_{j=1}^{N}\sum_{i=1}^{c_j}\sum_{k=1}^{m_{ij}} dN_{jik}(s)}{\sum_{j=1}^{N}\sum_{i=1}^{c_j}\sum_{k=1}^{m_{ij}} 1\{L_{jik} < s \leq U_{jik}\} e^{\boldsymbol{Z}_{jikl}(s)\hat{\beta}_{nl}}} \tag{4.23}$$

where j, i, k are introduced in Section 4.2.2, $dN_{jik}(s)$ is the indicator of an observed survival event at time s (the derivative of the counting process), $\boldsymbol{Z}_{jikl}(s)$ represent covariates at time s for the lth MI copy, and L_{jik}, U_{jik} are the lower and upper bound of the at-risk interval for the jth level-1 group, ith level-2 group, and kth encounter. The accumulative hazard functions is cumulative over all survival time points for all levels across j, i, k. The at-risk intervals are shifted to relative days since at-risk, rather than the actual date because we care more about the relative length and the actual dates do not give more information than relative days. Hence, the first interval for each subject would always start with zero.

When there are multiple hierarchies in the survival data, which level of the hierarchy is of interest? The purpose of this paper is to monitor and predict the number of events at the highest level (e.g., the program level). This is a common task especially for infection prevention and control (IP&C). The observed number of events at the jth level-1 group (e.g., program j) is summed over the bottom two levels

$$N_j = \sum_{i=1}^{c_j}\sum_{k=1}^{m_{ij}} N_{jik} \quad . \tag{4.24}$$

The expected number of risk-adjusted events for the jth level-1 group (e.g., program j) and lth multiple imputation is

$$\hat{N}_{jl} = \sum_{i=1}^{c_j}\sum_{k=1}^{m_{ij}} \int_s 1\{L_{jik} < s \leq U_{jik}\} e^{Z_{jikl}(s)\hat{\beta}_{nl}} d\hat{\Lambda}_{nl}(s) \tag{4.25}$$

Averaging over all multiple imputations, we obtain the expected number of risk-adjusted events

$$\hat{N}_{j\cdot} = \frac{1}{M}\sum_{l=1}^{M} \hat{N}_{jl} \tag{4.26}$$

The variation of this estimator mainly comes from the two plug-in estimators, $\hat{\beta}_{nl}$ and $d\hat{\Lambda}_{nl}$, and we propose that there are three parts that contribute to this variability $\widehat{\widehat{V}}_j$:

1. Within-group variance: $\hat{N}_{j\bullet}$. The variance of the estimated events for all level-2 groups within each level-1 group. The word "group" in "within-group" refers to the level-1 group;
2. Across-group variance: $\hat{V}_{j\bullet} = \frac{1}{M}\sum_{l=1}^{M}\hat{V}\left(\hat{N}_{jl}\right)$. The variance of the estimated events across level-1 groups;
3. Multiple imputation variance: $\hat{s}^2_{j\bullet} = \frac{1}{M}\sum_{l=1}^{M}\left(\hat{N}_{jl} - \hat{N}_{j\bullet}\right)^2$.

Thus,

$$\widehat{\widehat{V}}_j = \hat{N}_{j\bullet} + \hat{V}_{j\bullet} + \hat{s}^2_{j\bullet} \quad . \tag{4.27}$$

The first and third components are readily available, but the second component is not straightforward and we apply block jackknife (Ma et al., 2005) to estimate the across-group variance. Let a fixed integer m be the number of blocks and $q_{m,N}$ be the number of elements in each block, which is defined as the largest integer such that $m \cdot q_{m,N} \leq N$, where N is the unique number of values of the level-1 grouping variable. We are interested in the events on level-1 N, not the total number of observations n. Next, the $m \cdot q_{m,N}$ observations are randomly sampled from the original data D, denoted as D^*, to prepare for splitting data into even blocks. For example, if there are $N = 271$ and $m = 10$ blocks, each block will contain $q_{m,N} = 27$ elements because $m \cdot q_{m,N} = 270$ is the largest integer divisible by m but is still less than N. The notation $*$ is used to distinguish the block jackknife data from the original data. For each block $b = 1, \ldots, m$, we obtain $\hat{\theta^*}_j^{(-b)} = \hat{N}_j\left(\hat{\beta}^{*(-b)}, d\hat{\Lambda}^{*(-b)}\right)$ based on $D^{*(-b)}$, which is the $m \cdot q_{m,N}$ randomly sampled level-1 groups after omitting the bth block (thus $b-1$ blocks remaining). We then combine the estimators and their estimated variance from all m blocks by computing

$$\bar{\theta}^*_j = \frac{1}{m}\sum_{b=1}^{m}\hat{\theta^*}_j^{(-b)}$$

and

$$S_j^* = (m-1)q_{m,N}\sum_{b=1}^{m}(\hat{\theta}_j^{*(-b)} - \hat{\theta}_j^*)(\hat{\theta}_j^{*(-b)} - \hat{\theta}_j^*)^T.$$

Although MI notation is omitted here for simplicity and easier explanation, this block jackknife algorithm should be repeated to each $l = 1, \ldots, M$ multiple imputed copies with l added to the subscript of S_j^*. The second component of the variance of $\widehat{\widehat{V}}_j$ is then

$$\hat{V}_{j\cdot} = \frac{1}{M}\sum_{l=1}^{M}\hat{V}_{jl} = \frac{1}{M}\sum_{l=1}^{M}S_{jl}^*.$$

We provide more justifications of the three components in the variance $\widehat{\widehat{V}}_j$ as well as using block jackknife in Section 4.2.4.

Now we have all the components we need to construct hypothesis testing on how precise our event estimator is compared to the true number of events and adjusted by variability. The test statistic of a Z-test is

$$\hat{T}_j = \frac{N_j - \hat{N}_{j\cdot}}{\sqrt{\widehat{\widehat{V}}_j}} \tag{4.28}$$

where N_j is defined in Eq (4.24), $\hat{N}_{j\cdot}$ in Eq (4.26), and $\widehat{\widehat{V}}_j$ in Eq (4.27). The null hypothesis is $H_o : \hat{T}_j = 0$ and the alternative hypothesis is $H_a : \hat{T}_j \neq 0$. The alternative is two-sided rather than one-sided because we not only want to flag when observed events are more than expected (although it is the main concern) but also care about how precise our expectation is even when the observed is less than expected. With α representing the significance level, the two-sided $(1-\alpha)\times 100\%$ CI for the estimated number of events is

$$\hat{N}_{j\cdot} \pm z_{1-\frac{\alpha}{2}}\sqrt{\widehat{\widehat{V}}_j} \quad . \tag{4.29}$$

Depending on the interest, this CI can be one-sided if only excessive or deficient events are wanted, or even asymmetrically two-sided where we want to be more sensitive about excessive events than deficient events. Same as the alternative, the regular two-sided CI is used here because

we care about precision on both sides. Although no actions are needed for fewer-than-expected events, we want to be notified if we are doing better than expected.

4.2.4 Theoretical Justification

The three components of $\widehat{\widehat{V}}_j$ are derived from the definitions of $\hat{N}_{jl}$ in Eq (4.25) and $\hat{N}_{j\cdot}$ in Eq (4.26). By its definition, the variability of $\hat{N}_{j\cdot}$ comes from the two plug-in estimators, $\hat{\beta}_{nl}$ estimated from the frailty model and $d\hat{\Lambda}_{nl}(s)$ in Eq (4.23), and we decompose them by hierarchical layers. First, the number of events for a level-2 group in a level-1 group follows a Poisson process and should have an inherent variability from the model. The variance of a Poisson distribution is the event rate, which is estimated by $\hat{N}_{j\cdot}$. This component records the variability from the fixed effects. Going up one level, there is variability across level-1 groups, which are captured by the random effects, and we apply block jackknife to estimate the covariance matrix. Lastly, $\hat{N}_{j\cdot}$ comes from taking the average over M and there is variation across the different MI copies. If no MI is involved, the third component can be omitted in Eq (4.27). The rest of the components in $\hat{N}_{j\cdot}$ are observed data which contribute to the variability through the parameters but have no variability on their own.

The within-group variance is tricky to estimate as there are no known studies on what algorithms to use. Before block jackknife, we have explored two other methods, bootstrap and jackknife to calculate $\hat{V}_{j\cdot}$. There are two ways of bootstrapping, one is resampling with replacement and one is weighted bootstrap. When resampling level-1 groups as a whole (meaning all subsequent level-2 groups and recurrent events) with replacement, we encounter singularity issues. We think this error is mainly due to the fact that some level-1 groups never get to be in the resampled data. After attaching the original dataset to the bootstrap sample to make sure every level-1 group is included, we encounter similar errors. After careful checks we find out that the issue is because repeating the data produces tied failures times, even with Efron approximations (Efron, 1977). This leads us to the idea of weighted bootstrap where we do not resample with replacement but apply a weight randomly drawn from exponential distribution with parameter 1 for each level-1 group. Weighted bootstrap is more applicable in general, even

when some nuisance parameters are not $\sqrt{n}-$consistent (Kosorok, 2008). However, we run into optimization error despite the different optimization methods we try, Nelder-Mead (Nelder and Mead, 1965), BFGS (Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970), or Conjugate Gradient (CG) (Fletcher and Reeves, 1964). An alternative to bootstrap is jackknife, also known as leave-one-out cross validation (LOOCV). We apply jackknife to $\hat{\beta}_{nl}^{(-v)}$ where we take out the vth level-1 grouping variable at a time. This gives us

$$d\hat{\Lambda}_{nl}^{(-v)}(s) = \frac{\sum_{j=1,j\neq v}^{N}\sum_{i=1}^{c_j}\sum_{k=1}^{m_{ij}} dN_{jik}(s)}{\sum_{j=1,j\neq v}^{N}\sum_{i=1}^{c_j}\sum_{k=1}^{m_{ij}} 1\{L_{jik} < s \leq U_{jik}\}e^{Z_{jikl}(s)\hat{\beta}_{nl}^{(-v)}}}$$

and

$$\hat{N}_{jl}^{(-v)} = \sum_{i=1}^{c_j}\sum_{k=1}^{m_{ij}}\int_s 1\{L_{jik} < s \leq U_{jik}\}e^{(\hat{\beta}_{nl}^{(-v)})^T Z_{jikl}(s)} d\hat{\Lambda}_{nl}^{(-v)}(s)$$

where s represents all unique event time points in the original dataset. Hence,

$$\hat{V}_{jl} = \hat{V}(\hat{N}_{jl}) = (N-1)\sum_{v=1}^{N}\left(\hat{N}_{jl}^{(-v)} - \hat{N}_{jl}\right)^{\otimes 2}.$$

We do not run into any optimization or singularity issues with the jackknife method but the variance is large and some lower bounds of confidence intervals are far below zero, implying it is not likely to give meaningful confidence intervals. This is not unexpected since jackknife improves on bias by using as much training data as possible compared with other cross validation methods, but this improvement is achieved by the sacrifice of variability. This inspires us to consider block jackknife, the middle ground, where we remove one block of level-1 groups at time instead of only one level-1 group. Although similar to m-fold cross validation where a fold corresponds to a block, block jackknife specifies each block to have the same number of elements whereas cross validation does not necessarily have folds of equal lengths. Block jackknife requires fewer assumptions and is computationally simpler than nonparametric bootstrap including its alternatives, m within n bootstrap and subsampling. It has been shown that when properly normalized, the block jackknife estimator converges to an F distribution at rate n, which means

that it obtains asymptotically valid confidence ellipses for the true parameter (Kosorok, 2008). Block jackknife has a hyperparameter, the number of blocks m, that could be adjusted to balance the bias-variance tradeoff. Our exploratory results in Sections 4.3 and 4.4.4 show that block jackknife gives reasonable, well-validated variance estimation.

4.3 Simulations

4.3.1 Simulation Settings

To evaluate our proposed risk-adjusted frailty model, we conduct various simulations whose parameter settings are chosen to mimic the clinical data used in Section 4.4. In a hypothetical situation, let there be $10,000$ subjects (i.e., level-2 groups, denoted by i) at 150 health care programs (i.e., level-1 groups, denoted by j), and each subject has an equal probability of going to each program. We assume a nested effect of the hierarchy where one subject belongs to one and only one program and there is no relocation involved. We study two time periods, 2012 to 2014 (3-year) and 2014 (1-year), and validate the predictions of survival events and their variation on year 2015. We use l to denote encounters. Based on CF regulations, patients are expected to go to their CF programs about four times a year, so we create 12 encounter time points ($t_l, l \in \mathcal{L} = \{1, \ldots, 12\}$) for the 3-year training period and 4 encounter time points ($t_l, l \in \mathcal{L} = \{1, 2, 3, 4\}$) for the 1-year period and assume all subjects in a simulated dataset have the same time points for each simulated dataset. The t_l's are ordered in an increasing order with $t_1 = 0$. The rest of the $l - 1$ time points are randomly generated from a uniform distribution $U(1, 1100)$ or $U(1, 400)$ across the 3-year or 1-year period, all rounded to the nearest integer to imitate number of days.

A total of 10 covariates are generated. The five time-invariant covariates, denoted as $\boldsymbol{Z}_{11}, \ldots, \boldsymbol{Z}_{15}$, follow a multivariate normal distribution $MVN(\mu_1, \Sigma_1)$ where

$\mu_1 = \begin{pmatrix} 0, 0, 0, 0, 0 \end{pmatrix}^T$ and

$$\Sigma_1 = \begin{pmatrix} 0.1 & 0.02 & 0.02 & 0.02 & 0.02 \\ 0.02 & 0.1 & 0.02 & 0.02 & 0.02 \\ 0.02 & 0.02 & 0.1 & 0.02 & 0.02 \\ 0.02 & 0.02 & 0.02 & 0.1 & 0.02 \\ 0.02 & 0.02 & 0.02 & 0.02 & 0.1 \end{pmatrix}.$$

We assume all $\boldsymbol{Z}_1$ variables are correlated somewhat weakly. The five time-variant covariates are denoted as $\boldsymbol{Z}_{21}(t), \ldots, \boldsymbol{Z}_{25}(t)$, independent and identically distributed, where each $\boldsymbol{Z}_{2p}(t)$ for $p = 1, \ldots, 5$ follows a multivariate normal distribution $MVN\left(\mu_2, \Sigma_2\right)$ across t. The mean vector is $\mu_2^1 = \begin{pmatrix} -5, -4, -3, \ldots, 4, 5, 6 \end{pmatrix}^T$ for 12 time points or $\mu_2^2 = \begin{pmatrix} -1, 0, 1, 2 \end{pmatrix}^T$ for four time points. The covariance matrix is

$$\Sigma_2^1 = c \begin{pmatrix} 12 & 11 & 10 & \ldots & 2 & 1 \\ 11 & 12 & 11 & \ldots & 3 & 2 \\ & & & \ddots & & \\ 1 & 2 & 3 & \ldots & 11 & 12 \end{pmatrix} \text{ or } \Sigma_2^2 = c \begin{pmatrix} 4 & 3 & 2 & 1 \\ 3 & 4 & 3 & 2 \\ 2 & 3 & 4 & 3 \\ 1 & 2 & 3 & 4 \end{pmatrix}$$

for 3-year and 1-year period respectively, where c is a scalar randomly sampled from uniform distribution $U(0, 0.1)$ for each simulation dataset. The covariance matrices are chosen to reflect correlation over time.

The piece-wise survival time $S(t_l)$ follows a piece-wise exponential distribution with rate being the hazard with a constant baseline hazard function h_0:

$$h_{ij}(t_l) = h_0 \exp(\boldsymbol{Z}_{1ij}\beta_1 + \boldsymbol{Z}_{2ij}(t_l)\beta_2 + b_{1i} + b_{2j}),$$

where β_1, β_2 are coefficients for the time-invariant and time-variant covariate $\boldsymbol{Z}_{1ij}, \boldsymbol{Z}_{2ij}(t_l)$ and b_{1i}, b_{2j} are the random intercept effects on level-2 (patient level) and level-1 (program-level). We assume no random slope effect for simplicity. Under the assumption that $\boldsymbol{Z}_{2ij}(t_l)$ does not change in the time interval $[t_l, t_{l+1})$ and the memoryless property of exponential distribution, the event time is $T_{ij}^1 = t_l + S_{ij}(t_l)$ if $t_l + S_{ij}(t_l) < t_{l+1}$. Otherwise, there is no event between t_l and t_{l+1} and we simulate the next piece-wise survival time $S_{ij}(t_{l+1})$ from $h_{ij}(t_{l+1})$ and repeat the process until we reach the last time point. Let W_{ij} denote the length of the pre-specified washout period for the ith patient at the jth program. For simplicity, we assume $W_{ij} = w$ for all i, j's with a constant w. For recurrent events, the second event time is $T_{ij}^2 = T_{ij}^1 + W_{ij} + S_{ij}(t_{l'})$ if $T_{ij}^1 + W_{ij} + S_{ij}(t_{l'}) < t_{l'+1}$ where $l' = \{l \in \mathcal{L} : \max(t_{l'} \leq T_{ij}^1 + W_{ij})\}$. Otherwise, there is no event between $T_{ij}^1 + W_{ij}$ and $t_{l'+1}$. We simulate the next piece-wise survival time $S_{ij}(t_{l'+1})$ and determine the second event time as $T_{ij}^2 = t_{l'+1} + S_{ij}(t_{l'+1})$ if $t_{l'+1} + S_{ij}(t_{l'+1}) < t_{l'+2}$ and repeat this process the same way as the first event time described above. The independent censoring time C is exponentially distributed with rate $1/600$ and $1/300$ for the 3-year and 1-year period. The event indicator $\delta_{ij} = 1\{T_{ij} \leq C_{ij}\}$ where 1 indicates an event and 0 indicates censoring. The observed time is $Y_{ij} = \min(T_{ij}, C_{ij})$. There could be more than one $\delta_{ij} = 1$ because of recurrent events.

Table 4.17 is a summary of all constants and parameters in the simulation introduced so far. Since we assume a nested structure between the level-1 group and level-2 group, all notations with subscript ij could be written with subscript i only because j is deterministic given i, but we use ij as it suits not nested structures (e.g., crossed) between level-1 and level-2 groups. The values of baseline hazard and the censoring time parameter are determined by the censoring rate.

Since there are only 10 covariates and no missing data, we skip the multiple imputation and feature selection steps in the pipeline illustrated in Figure 4.8. After the event/censoring time and the at-risk intervals are generated, we move on to survival modeling directly. The modeling involves two parts. For each iteration, we start with fitting a Cox proportional hazards

Table 4.17: Simulation constants and parameters

Parameter	Notation	Value	
		3-year	1-year
No. of groups, level-1	n_2	150	
No. of groups, level-2	n_1	10,000	
No. of time points	$\lvert\mathcal{L}\rvert$	12	4
Length of study period		1100	400
Encounter time	$t_l, l \in \mathcal{L}$	$U(12, 1100)$	$U(4, 400)$
No. of time-invariant covariates	p_1	5	
No. of time-variant covariates	p_2	5	
Time-invariant covariate	$\boldsymbol{Z}_1$	$MVN(\mu_1, \Sigma_1)$	
Time-variant covariate (i.i.d. across $p = 1, \ldots, p_2$)	$\boldsymbol{Z}_{2p}(t)$	$MVN(\mu_2^1, \Sigma_2^1)$	$MVN(\mu_2^2, \Sigma_2^2)$
Baseline hazard	h_0	e^{-8}	
Covariate coefficient	β_1, β_2	all 0.5	
Random intercept effect, level-1	b_j	$N(0, G_j), G_j \sim N(0, 0.001)$	
Random intercept effect, level-2	b_i	$N(0, G_i), G_i \sim N(0, 0.001)$	
Piece-wise hazard function	$h_{ij}(t)$	$h_0 \exp(\boldsymbol{Z}_{1ij}\beta_1 + \boldsymbol{Z}_{2ij}(t_l)\beta_2 + b_{1i} + b_{2j})$	
Piece-wise survival time	$S_{ij}(t)$	$\exp(h_{ij}(t))$	
Censoring time	C_{ij}	$\exp(1/600)$	$\exp(1/300)$
Washout period	W_{ij}	730 (2 years)	
No. of Simulations		100	

model to acquire initial values for the coefficients of the later frailty model. Taking into account correlated observations, we utilize robust standard errors by identifying level-1 (e.g., program) and level-2 (e.g., patient) as correlated groups. R package `survival` is used for the Cox model. The mixed effect survival model is applied to the same covariates using the initial values from the Cox model to help with convergence. The computation tool we use to fit the frailty model is the R package `coxme`, which assumes the random effects follow a Gaussian distribution (Therneau, 2019) and is more efficient because of the use of semiparametric estimation in the lognormal frailty model. Instead of treating random effects as missing data and applying EM algorithm which has been proven to be slow, it incorporates random effects by penalizing the partial likelihood which can be easily implemented by adding a penalty term to standard Cox semiparametric models (Therneau et al., 2003). The rest of the pipeline (resampling, variance

estimation, and validation) are followed as described in the diagram. Section 4.4 will explore all steps in the pipeline. Simulation calculations are performed in R 3.6.1 (R Core Team, 2019).

4.3.2 Simulation Results

The 3-year training period is set to be from January 1, 2012 to December 31, 2014. The 1-year training period is from January 1, 2014 to December 31, 2014. The 1-year test period for both training periods is January 1, 2015 to December 31, 2015. For each of the three periods, 100 simulated datasets have been generated following the assumptions and definitions in Section 4.3.1. The observed censoring rate in Section 4.4 is between 0.67 and 0.83 for 2012-2014 and 0.86 and 0.95 for 2013, 2014, 2015. The observed second event rate for 2012-2014 period is between 0.0001 and 0.005. The censoring rate in our 100 simulated data has mean 0.78 and standard deviation (SD) 0.05 for the 3-year period and mean 0.90 and SD 0.02 for the two 1-year periods. The second event rate has mean 0.004 with SD 0.003.

The estimated coefficients for $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2(t)$ with SDs are displayed in Table 4.18 using a Cox model and Table 4.19 using a frailty model. Overall, all estimated coefficients are reasonably close to the true value 0.5 regardless of time-invariant or time-variant covariate or survival model. The two 1-year periods have smaller bias than the 3-year period but higher SDs. The 2012-2014 period has relatively lower estimates than 0.5 and we speculate this is due to the interference of recurrent events. The coefficients in the mixed effect model have smaller biases for the corresponding coefficients in the the Cox model, indicating that adjusting for the random effects helps with estimation and it learns the structure in the data better. This confirms that our simulated data and models are reasonably generated and well-fit. All covariates have p-values less than 0.05, implying that the coefficients are significantly different from 0, which is expected as all covariates are involved in the definition of hazard function and survival time.

Next, we apply block jackknife resampling to estimate the variance as well as the CIs. To observe results for various situations and settings, three values of m (the number of blocks in the block jackknife) are used and several levels of the confidence intervals are explored spanning from 0.7 to 0.995. Tables 4.20 and 4.21 contain mean coverage of the estimated risk-adjusted

Table 4.18: Mean (SD) of estimated covariate coefficients across 100 simulations for three time periods based on a Cox proportional hazards model

Covariate	2012-2014	2014	2015
$\boldsymbol{Z}_{11}$	0.46 (0.07)	0.51 (0.11)	0.51 (0.11)
$\boldsymbol{Z}_{12}$	0.47 (0.08)	0.51 (0.11)	0.51 (0.11)
$\boldsymbol{Z}_{13}$	0.46 (0.07)	0.49 (0.11)	0.49 (0.11)
$\boldsymbol{Z}_{14}$	0.45 (0.08)	0.51 (0.12)	0.50 (0.12)
$\boldsymbol{Z}_{15}$	0.46 (0.08)	0.49 (0.10)	0.49 (0.10)
$\boldsymbol{Z}_{21}$	0.46 (0.09)	0.49 (0.12)	0.50 (0.14)
$\boldsymbol{Z}_{22}$	0.47 (0.05)	0.50 (0.11)	0.51 (0.12)
$\boldsymbol{Z}_{23}$	0.47 (0.09)	0.50 (0.16)	0.49 (0.15)
$\boldsymbol{Z}_{24}$	0.47 (0.10)	0.53 (0.16)	0.52 (0.16)
$\boldsymbol{Z}_{25}$	0.47 (0.04)	0.48 (0.14)	0.48 (0.14)

Table 4.19: Mean (SD) of estimated covariate coefficients across 100 simulations for three time periods based on a mixed effects Cox model

Covariate	2012-2014	2014	2015
$\boldsymbol{Z}_{11}$	0.47 (0.07)	0.51 (0.11)	0.52 (0.11)
$\boldsymbol{Z}_{12}$	0.48 (0.08)	0.52 (0.11)	0.51 (0.11)
$\boldsymbol{Z}_{13}$	0.47 (0.07)	0.49 (0.11)	0.50 (0.11)
$\boldsymbol{Z}_{14}$	0.46 (0.07)	0.52 (0.12)	0.50 (0.12)
$\boldsymbol{Z}_{15}$	0.47 (0.07)	0.49 (0.10)	0.49 (0.10)
$\boldsymbol{Z}_{21}$	0.47 (0.09)	0.49 (0.12)	0.50 (0.14)
$\boldsymbol{Z}_{22}$	0.47 (0.05)	0.51 (0.12)	0.51 (0.12)
$\boldsymbol{Z}_{23}$	0.48 (0.09)	0.50 (0.16)	0.50 (0.15)
$\boldsymbol{Z}_{24}$	0.48 (0.09)	0.53 (0.16)	0.52 (0.16)
$\boldsymbol{Z}_{25}$	0.48 (0.04)	0.48 (0.15)	0.48 (0.14)

CIs for the 3-year period and 1-year period, both validated on the 2015 data. When trained and tested on the same 3-year period, the CI is generally good for all α's and m's with the largest absolute value of mean coverage difference being 0.016 and the majority under 0.006. There is more coverage for higher confidence levels. When tested on a validation period 2015, the CI coverage is relatively worse, which is not surprising because the model has not seen the validation data, but it is within an absolute value of 0.032. For this validation, there is no clear trend that higher confidence levels have more coverage or that certain value of m gives higher coverage. When trained and tested on the same 1-year period, the CI coverage is exceptionally good for all α's and m's with the largest absolute value of mean coverage difference being 0.005.

Table 4.20: Results of risk-adjusted model of 2012-2014 simulated data validated on 2012-2014 and 2015 simulated data separately, where α is significance level, m is number of blocks in block jackknife estimation of variance, meanCoverage is mean proportion of level-1 groups whose true number of survival cases in the validation set is contained in the risk-adjusted $1-\alpha$ confidence interval estimated from the training data averaged across 100 simulations, and AbsCovDiff is absolute difference between the mean coverage and $1-\alpha$.

2012-2014 Training		2012-2014 Validation		2015 Validation	
$1-\alpha$	m	meanCoverage	AbsCovDiff	meanCoverage	AbsCovDiff
0.7	5	0.706	0.006	0.724	0.024
	10	0.711	0.011	0.729	0.029
	15	0.716	0.016	0.732	0.032
0.8	5	0.800	0.000	0.805	0.005
	10	0.806	0.006	0.808	0.008
	15	0.811	0.011	0.812	0.012
0.9	5	0.895	0.005	0.882	0.018
	10	0.901	0.001	0.884	0.016
	15	0.905	0.005	0.886	0.014
0.95	5	0.946	0.004	0.919	0.031
	10	0.948	0.002	0.921	0.029
	15	0.951	0.001	0.923	0.027
0.995	5	0.991	0.001	0.962	0.033
	10	0.992	0.002	0.964	0.031
	15	0.992	0.002	0.964	0.031

When tested on a validation period 2015, the CI coverage is relatively worse again, but within 0.060 below the true confidence interval. The coverage is uniformly better as confidence level $1-\alpha$ increases. In addition, $m = 15$ seems to produce low coverage differences compared with $m = 5, 10$ but not by very much. The 2015 validation results are better using 3-year data than 1-year data in terms of absolute coverage difference and we think this is is because 3-year data are richer and contain more heterogeneity information for the proposed model to learn. In general, it is not useful to look at results trained and validated on the same dataset as it almost guarantees overfitting but we study such results as a comparison reference in addition to the more important 2015 validation results.

To visually inspect the estimation of survival events across simulations, Figure 4.9 contains four histograms of estimated versus true number of events across 100 simulated datasets. The four subplots are results trained and validated on 2012-2014 (top left), trained on 2012-2014 but

Table 4.21: Results of risk-adjusted model of 2014 simulated data validated on 2014 and 2015 simulated data separately, where α is significance level, m is number of blocks in block jackknife estimation of variance, meanCoverage is mean proportion of level-1 groups whose true number of survival cases in the validation set is contained in the risk-adjusted $1-\alpha$ confidence interval estimated from the training data averaged across 100 simulations, and AbsCovDiff is absolute difference between the mean coverage and $1-\alpha$.

2014 Training		2014 Validation		2015 Validation	
$1-\alpha$	m	meanCoverage	AbsCovDiff	meanCoverage	AbsCovDiff
0.7	5	0.695	0.005	0.646	0.054
	10	0.697	0.003	0.649	0.051
	15	0.700	0.000	0.651	0.049
0.8	5	0.802	0.002	0.742	0.058
	10	0.803	0.003	0.745	0.055
	15	0.806	0.006	0.747	0.053
0.9	5	0.901	0.001	0.840	0.060
	10	0.903	0.003	0.842	0.058
	15	0.904	0.004	0.844	0.056
0.95	5	0.951	0.001	0.895	0.055
	10	0.952	0.002	0.897	0.053
	15	0.953	0.003	0.898	0.052
0.995	5	0.992	0.002	0.967	0.028
	10	0.992	0.002	0.968	0.027
	15	0.992	0.002	0.968	0.027

validated on 2015 (top right), trained and validated on 2014 (bottom left), trained on 2015 but validated on 2015 (bottom right). The estimated Spearman correlation coefficients between the estimated and observed for each of the four subplots are $0.71, 0.24, 0.50, 0.37$, indicating weak to moderate positive correlations. The first histogram has the highest number of survival events because it spans over three years of period. All four distributions of the estimated events have roughly similar means and ranges as the four distributions of observed events, but the heights of the modes vary. The top left histogram has the closest observed nad estimated distributions because it uses and overfits three years of data. The top right histogram is the second closest with the estimated distribution denser at the mode with estimated distribution denser near the mode than the observed distribution. The estimated and true distributions differ more in the bottom histograms. This phenomenon can be explained by regression to the mean.

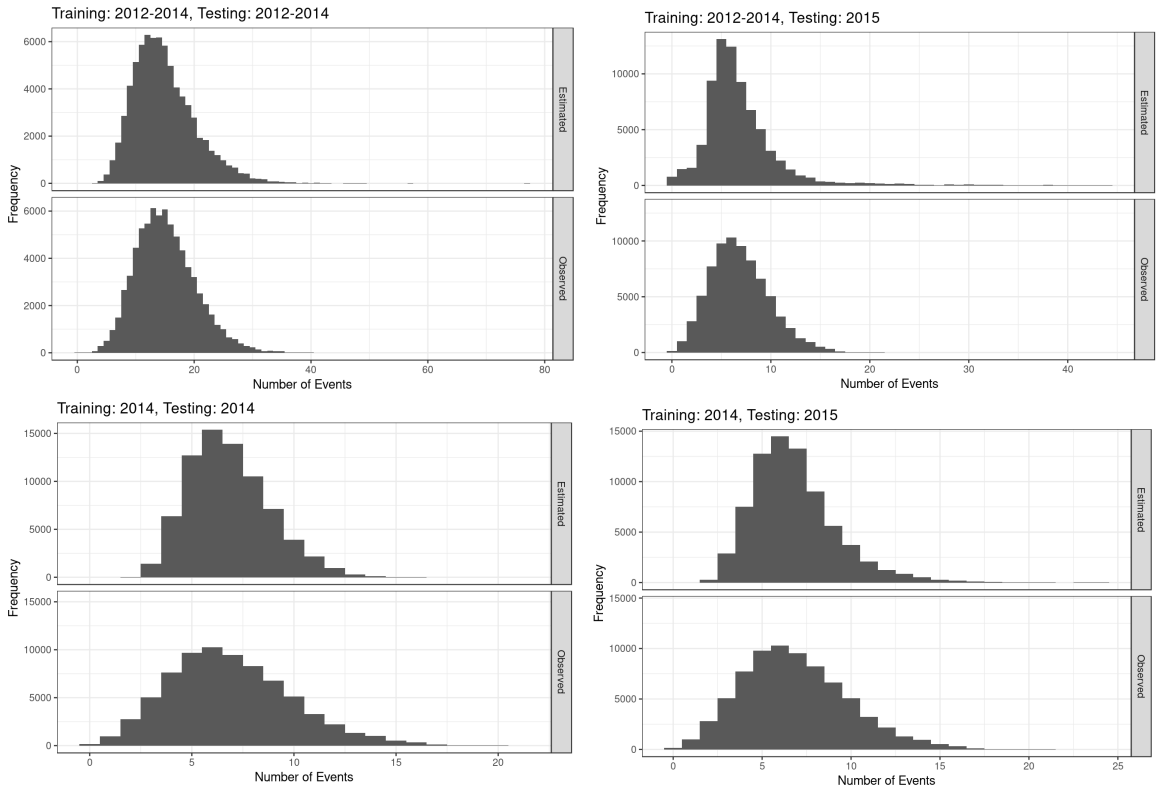

Figure 4.9: Histograms of estimated versus observed (true) number of survival events for different training and validation periods cumulative across 100 simulations (top left: 2012-2014 training and validation, topright 2012-2014 training and 2015 validation, bottom left: 2014 training and validation, bottom right: 2014 training and 2015 validation)

4.4 Clinical Implementation

Cystic fibrosis (CF) is a chronic, genetic disorder where defects in a chloride ion channel lead to excessively thick and sticky mucus throughout the respiratory and digestive systems. This impaired mucus clearance predisposes patients to respiratory infections and chronic lung damage, the main cause of morbidity and mortality in patients with cystic fibrosis. Two bacteria in particular, methicillin-resistant Staphylococcus aureus (MRSA) and pseudomonas aeruginosa (PA) are particularly harmful and lead to decreased lung functions and poor outcomes for patients with CF. These respiratory pathogens can be acquired and spread in healthcare institutions for both the inpatient and outpatient settings (Saiman et al., 2003). Monitoring such transmission is much needed for CF IP&C. There is considerable variability in IP&C among CF programs and they currently do not have an accurate, timely way of estimating their incidence rates of

key pathogens such as MRSA and PA (Stoudemire et al., 2019). The incidence rate is defined as the number of incidence cases (with a 2-year look-back free of infection) divided by the at-risk population of the bacterial infection. We apply the new proposed pipeline to the U.S. CF Foundation Patient Registry (CFFPR) data for the period of 2012-2015 to study, predict, and monitor the incidence of bacteria infection. The CFFPR fits our setup because it gives right-censored data with i) a 3-level hierarchy, where level 1 is CF program, level 2 is CF patient who goes to a CF program, and level 3 is an encounter observation from a CF patient who goes to a CF program; 2) repeated events as infections could occur again after a washout period. All analyses are performed in R 3.6.1 (R Core Team, 2019).

4.4.1 Preprocessing

We start with defining MRSA and PA infections. MRSA infection is defined as having a positive respiratory culture for methicillin-resistant staphylococcus aureus - a bacteria that has developed resistance to penicillin-based antibiotics. PA infection is defined as having a positive respiratory culture to any of the different species of Pseudomonas. Sensitivity analyses had been done preliminarily to compare 2, 5, 10 years of a washout period and concluded that shorter look-back intervals did not alter overall estimates of incidence or changes in incidence. Hence, the start date of an at-risk period for a patient is defined as the first day that the patient has 2 years free of infections since last infection. We apply the following exclusion criteria to define the at-risk population. We exclude all encounters of patients 1) who do not have encounter date data; 2) whose start date of at-risk day is the infection date; and we exclude encounters (not the entire patient) 3) who have gaps in encounter data for more than 18 months (otherwise we assume the patient's infection status stays the same during the gap); 4) after organ transplants. Babies less than two years old are considered at-risk even though they do not have two years of data to look back at. Relocation and change of program are allowed. After data cleaning, we construct at-risk intervals for each patients consisted of start and end dates of being at-risk as well as a censoring indicator. For a time period of three years, each patient could have 0, 1, or 2

infections (because the two year look-back is contained in three years) but each patient should have at most one infection case for a time period less than two years.

Potential risk factors are also included in the preprocessed, right-censored data. The CFFPR data contain different modalities of data on prevalence of CF related pathogens, provided by CF programs once a year. We combine three modalities demographics and diagnosis, encounter, and annualized data by unique patient ID and review year. The mean number of yearly encounters per patient is around 5.6. The potential risk factors could be time invariant (demographic data) or time varying covariates (often found in encounter and annualized data). The values of time varying covariates are associated with the start dates of the corresponding at-risk intervals, as we assume that the time varying covariates do not change between the current encounter and the next consecutive encounter (for gaps less than 18 months). The time scales of the risk factors vary, which could be by year or by day. Note that the start date of one at-risk interval may not be an actual encounter date because we derive the hypothetical at-risk start date by going back two years from an infection date. Each of the potential risk factors is carefully chosen, preprocessed, and transformed (if necessary) by both the clinician and biostatistician on board. The reasons that we exclude some covariates are but not limited to correlation and multicollinearity, rare events, and more than 50% missingness. As a result, the number of covariates in the preprocessed data is 79 for MRSA and 72 for PA. The number of observations vary by the time period. For 2012-2014 (the longest period that we look at) MRSA has 219,251 observations of 18,366 patients and PA has 123,341 observations of 13,228 patients. For 2014, MRSA has 70,363 observations of 15,810 patients and PA has 41,085 observations of 10,505 patients.

4.4.2 Multiple Imputation

To handle missing data in the preprocessed covariates, different imputation methods are applied to each modality separately before merging. We generate $M = 10$ copies of imputed datasets.

Encounter data. First we apply last observation carried forward (LOCF) first and then next observation carried backward (NOCB) to variables in the encounter data. Missing FEV1 (forced

expiratory volume for the first forced breath) values are replaced by the max FEV1 value in the past 365 days for each patient. After LOCF and NOCB, what is left should be missing for all observations/encounters of the patients. Taking into account the longitudinal nature in the encounter data, we apply joint modeling imputation (Carpenter and Kenward, 2012) in the `mitml` R package (Grund et al., 2019) with unique patient ID as the random effect. This is a MCMC imputation algorithm suited for multilevel data continuous and categorical data, which match with the longitudinal nature in our encounter data.

Annualized data. We do not apply LOCF or NOCV to annualized data because a year is considered too long to be carried forward or backward, whereas encounter data are more frequent. Instead, joint modeling is applied again with patient ID as a random effect because annualized data are longitudinal as well, only less frequent than encounters.

Demographic data. We apply multiple imputation by chained equations (MICE) with random forests (RF) because it is able to impute nominal variables (e.g., mutation information such as F508 - the most common disease causing mutation in CF) and there are no random effects in demographic data.

4.4.3 Survival Model and Variable Selection

With more than 70 potential risk factors in the preprocessed data, we want to narrow them down and identify only the important risk factors that are associated with incidence rates of MRSA and PA. Stepdown selection is used for this purpose. Stepdown selection typically starts with modeling on all covariates and remove one insignificant covariate at a time until all covariates left have p-value less than the significance level. The remaining covariates are called the important risk factors. It has been shown that under several reasonable, generalized assumptions, meaningful recursive feature elimination methods with kernel machines can find the correct feasible feature space with uniform consistency (Dasgupta et al., 2019).

As in the simulations, we use Cox model first to acquire initial values for the coefficients before fitting the frailty model. The same R packages (`survival` and `coxme`) are applied to the CF data. We make two changes to the stepdown selection procedure to fit our situation

better. First, removing variables one at a time for over 70 covariates would be time consuming. We changed the classic stepdown selection to allow it to drop more than one variable for each iteration in order to speed up the variable selection process. More specifically, we remove three variables at a time in the early selection stage where the p-values of the dropped variables are greater than or equal to 0.5 then drop two variables at a time when p-values are strictly between 0.25 and 0.5. When the p-values are less than or equal to 0.25, we slow down the elimination process and drop one variable at a time. We do not drop any variable when the highest p-values are all below 0.1. Second, due to MI, we fit the two survival models to each imputation copy separately then pool the estimates from all M multiple imputation datasets together and determine which variable(s) to drop based on the p-values described above. This is one iteration and we repeat with newly dropped covariates. When all p-values are below 0.1, we pool the coefficient estimates one last time and use the results to determine important risk factors. Note that stepwise selection is performed separately for MRSA and PA since different infections do not necessarily have the same risk factors.

4.4.4 Results

There are six combinations of training and testing for the period of 2012-2015: training 2012 to validate 2013, training 2013 to validate 2014, training 2014 to validate 2015, training 2012-2013 to validate 2014, training 2013-2014 to validate 2015, and training 2012-2014 to validate 2015. For the purpose of concise result presentation, we select two exemplary combinations. We present year 2014 as the training set and year 2015 as the test set because they are the most recent data, and later look at a training set of three years 2012-2014. Other combinations are omitted because results are similar if the numbers of years in the training set are the same but results are more different when the number of years differs. The training set and testing set are preprocessed in the same way as described in Section 4.4.1. Only CF programs in the training set are included in the testing set, and only the important risk factors in the training set are included in the testing set. After stepdown selection of 2014 data, there are 19 out of 79 risk factors selected for MRSA and 34 out of 72 for PA. For 2012-2014, as a comparison, 41 out of 79 (MRSA) and 45 out of

72 (PA) risk factors are selected because this time period has more data. This shows that PA infection is associated with more factors than MRSA. Important risk factors of MRSA include season, region, whether or not bacterial culture was done, need-based insurance, number of hospitalization/outpatient visit in the past year, feeding, smoking, birth year, etc. PA has more important risk factors such as mutation class, hispanic race, sale supplement, FEV1, in addition to season and region. More results on variable selection and clinical interpretation of the selected risk factors can be found in Stoudemire et al. (shed), which provides the clinical prospective of our risk-adjusted survival analysis. Other specifications in the clinical analysis include $M = 10$ multiple imputed copies and $m = 5, 10, 15$ numbers of blocks.

Risk-Adjusted Incidence Modeling. After preprocessing, multiple imputation, and variable selection, we follow the proposed pipeline in Figure 4.8. We perform the risk-adjuted survival models and calculate the CIs as defined in Eq (4.29) for 2014 CFFPR data. Prior to the validation stage, we check to see if the risk-adjusted CIs trained from 2014 contain the true number of events for the same dataset. Based on Table 4.22, we see that the coverage of the risk-adjusted confidence intervals is close to the true $1-\alpha$ as almost all absolute absolute coverage differences are less than 0.05 (except 2 places). This is expected as the training and testing data are the same. What is more interesting is m. For PA, $m = 10$ has the lowest absolute difference for low confidence levels ($1-\alpha \leq 0.9$) and the difference among different values of m starts to decrease for high confidence levels ($1 - \alpha \geq 0.95$). High confidence levels require wider confidence intervals and thus the bias-variance tradeoff controlled by m is no so prominent. For MRSA, $m = 5$ has the lowest absolute difference for all $1 - \alpha$'s but the differences are still larger when confidence levels are high. Figure 4.10 shows the distribution of the observed incidence cases is similar to the estimated incidence cases for both bacteria even at the tails. The correlation between the estimated and observed incidence is strong as the estimated Spearman correlation coefficients are 0.68 for MRSA and 0.77 for PA.

Validation with New Data. We now validate the model trained from 2014 data on 2015 data. Table 4.23 shows that the validation coverages are relatively worse by a small amount

Table 4.22: Results of risk-adjusted incidence model of 2014 CFFPR data validated on 2014 data, where α is significance level, m is number of blocks in block jackknife estimation of variance, Coverage is proportion of programs whose true number of 2014 incidence cases is contained in the risk-adjusted $1-\alpha$ confidence interval trained on 2014 data, and AbsCovDiff is absolute difference between the coverage and $1-\alpha$.

2014 on 2014		PA		MRSA	
$1-\alpha$	m	Coverage	AbsCovDiff	Coverage	AbsCovDiff
0.7	5	0.62	0.080	0.68	0.020
	10	0.70	0.000	0.72	0.020
	15	0.74	0.040	0.75	0.050
0.8	5	0.76	0.040	0.82	0.020
	10	0.81	0.010	0.84	0.040
	15	0.83	0.030	0.87	0.070
0.9	5	0.88	0.020	0.93	0.030
	10	0.91	0.010	0.94	0.040
	15	0.92	0.020	0.95	0.050
0.95	5	0.92	0.030	0.96	0.010
	10	0.93	0.020	0.97	0.020
	15	0.94	0.010	0.97	0.020
0.995	5	0.98	0.015	0.99	0.005
	10	0.98	0.015	0.99	0.005
	15	0.98	0.015	0.99	0.005

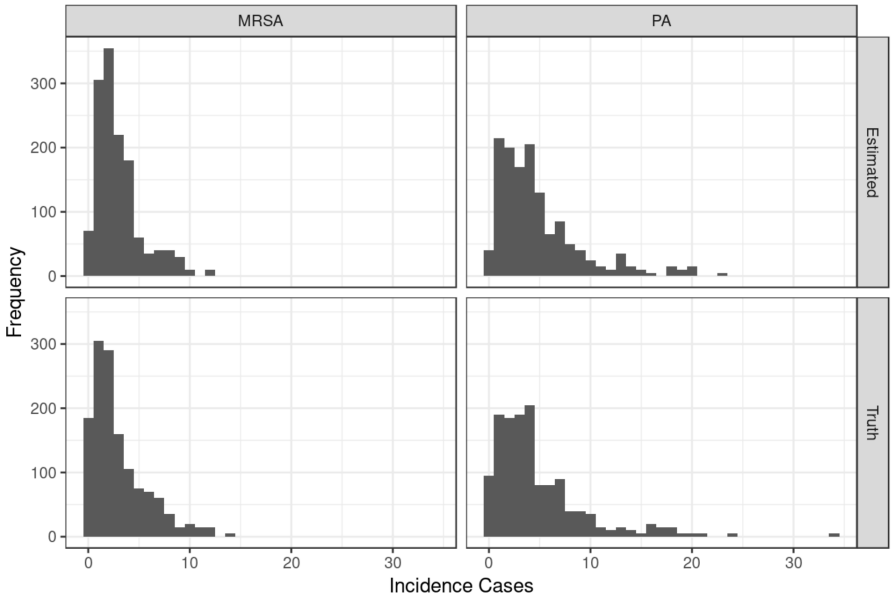

Figure 4.10: Histogram of estimated versus observed (true) number of 2014 MRSA and PA incidence cases where the risk-adjusted model is trained from 2014 data

Table 4.23: Results of risk-adjusted incidence model of 2014 CFFPR data validated on 2015 data, where α is significance level, m is number of blocks in block jackknife estimation of variance, Coverage is proportion of programs whose true number of 2015 incidence cases is contained in the risk-adjusted $1-\alpha$ confidence interval trained on 2014 data, and AbsCovDiff is absolute difference between the coverage and $1-\alpha$.

2014 on 2015		PA		MRSA	
$1-\alpha$	m	Coverage	AbsCovDiff	Coverage	AbsCovDiff
0.7	5	0.61	0.090	0.64	0.060
	10	0.67	0.030	0.65	0.050
	15	0.71	0.010	0.70	0.000
0.8	5	0.72	0.080	0.76	0.040
	10	0.77	0.030	0.79	0.010
	15	0.80	0.000	0.80	0.000
0.9	5	0.84	0.060	0.85	0.050
	10	0.87	0.030	0.86	0.040
	15	0.91	0.010	0.87	0.030
0.95	5	0.92	0.030	0.88	0.070
	10	0.94	0.010	0.90	0.050
	15	0.94	0.010	0.91	0.040
0.995	5	0.98	0.015	0.94	0.055
	10	0.99	0.005	0.95	0.045
	15	0.99	0.005	0.96	0.035

compared with the results in Table 4.22, because the test set is now new data that the model has not seen. As confidence levels go up, the coverage difference decreases for both MRSA and PA. The overall coverage of our risk-adjusted intervals is close to the desired $1-\alpha$ level as the absolute coverage differences are all less than 0.1 for all three m's. All differences are less than 0.05 when $m = 10, 15$. For this test set, $m = 15$ gives the best coverage for both bacteria. The distributions of observed versus estimated incidence cases are similar as well (Figure 4.11) including the right-skewed tails. The estimated Spearman correlation coefficient between the estimated and observed incidence is 0.59 for MRSA and 0.79 for PA (moderate to strong correlations). Results are depended on the hyperparameter m to some extent but are within reasonable range overall. Overall, block jackknife is able to estimate variabilities in the incidence cases well enough for the coverage difference to be small regardless of the value of m.

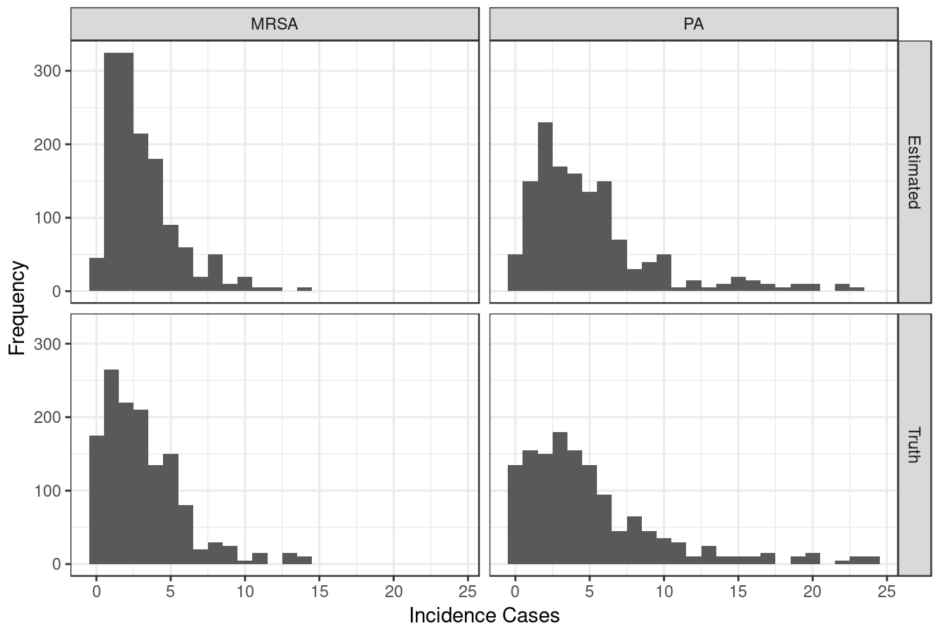

Figure 4.11: Histogram of estimated versus observed (true) number of 2015 MRSA and PA incidence cases where the risk-adjusted model is trained from 2014 data

Multiple years of training data. When the training set contains more than one year of CFFPR data, overcoverage is observed in the validation stage while the training coverage differences stay small. This is concluded from the summarized MRSA and PA coverage results, where we pick the 2012-2014 period as training set and validate on both 2012-2014 and 2015. We present MRSA results in Table 4.24. PA results are similar and can be found in Appendix C, which also contains the histograms of estimated versus observed number of 2014 and 2015 incidence cases using models trained from 2012-2014 data for both MRSA and PA. Simulation results in Section 4.3.2 show that three years of training data produce CIs with higher coverage than one year of data. Indeed, having three years of data allows recurrent events and increases sample size but the data seem to be noisier and contain more complex situations in our real world clinical situation. For instance, recall that the gaps between two consecutive encounters are allowed to be as long as 18 months, which is carefully determined with the clinicians on board. When the training set contains more than one year, it is possible that we include more encounters (encounters with longer gaps) which increases the variability of data. The proposed model learns from this complexity and estimates larger variance and wider CIs that overcover

Table 4.24: Results of risk-adjusted MRSA incidence model of 2012-2014 CFFPR data validated on 2012-2014 and 2015 data separately, where α is significance level, m number of blocks in block jackknife estimation of variance, Coverage is proportion of programs whose true number of incidence cases in the test set is contained in the risk-adjusted $1-\alpha$ confidence interval trained on 2012-2014 data, and AbsCovDiff is absolute difference between the coverage and $1-\alpha$.

MRSA		2012-2014 Validation		2015 Validation	
$1-\alpha$	m	Coverage	AbsCovDiff	Coverage	AbsCovDiff
0.7	5	0.60	0.100	0.87	0.170
	10	0.63	0.070	0.89	0.190
	15	0.68	0.020	0.90	0.200
0.8	5	0.70	0.100	0.93	0.130
	10	0.73	0.070	0.94	0.140
	15	0.77	0.030	0.94	0.140
0.9	5	0.82	0.080	0.97	0.070
	10	0.85	0.050	0.97	0.070
	15	0.88	0.020	0.97	0.070
0.95	5	0.88	0.070	0.98	0.030
	10	0.90	0.050	0.98	0.030
	15	0.93	0.020	0.98	0.030
0.995	5	0.97	0.025	0.99	0.005
	10	0.98	0.015	0.99	0.005
	15	0.98	0.015	0.99	0.005

the test set. Nonetheless, the coverage of 2015 test set is below 0.1 when confidence level is less than or equal to 0.9 and the difference becomes smaller when confidence level is higher. Overcoverage is noticeable when confidence level is low ($\approx 0.7, 0.8$). For common confidence levels (e.g., 0.95) the validation coverage is under control despite slight overcoverage.

4.5 Discussion

A risk-adjusted model is developed to learn from right-censored hierarchical data with recurrent events and estimate the survival events with CIs. The variability of predicted incidence cases has three sources and one of which is obtained with the block jackknife method. We propose to carefully preprocess the data and recommend useful tools such as multiple imputation and variable selection to produce clean and concise input data before modeling. Simulations are conducted to evaluate our methodology of the risk-adjusted CIs. CF Foundation Patient Registry data are used as clinical application to evaluate the practicality of our pipeline. Overall, the

results show promising usage of our incidence models when we train with shorter periods of CF data or larger periods of simulated data. This slight discrepancy implies that larger data are beneficial when they are clean and well-organized but larger data often bring in more unknown noise which may increase the difficulty of variance estimation. For the CF clinical data, data recency and quality trump data quantity. There is potential cohort effect across different time periods in the CF patients which is worth further investigation for generalizability.

Our methodology estimates the survival events and its variability for each program and packages everything into a CI which is easy to understand and acquire. The risk-adjusted incidence estimates and confidence intervals are organized into a spreadsheet. This is a ready-to-use incidence monitoring tool as the only inputs needed from CF programs are the observed incidence cases and the built-in formulae would be able to calculate whether or not to flag the programs if the observed incidence falls out of expectation. More explanation of this monitoring tool is discussed in Stoudemire et al. (shed). Stoudemire et al. (shed) also looks into coverage results when the test year has not ended, i.e., validating on 3, 6, 9, months instead waiting for the 12 whole months of data. We are able to provide quarterly reports on MRSA and PA incidence for each program, so unusual incidence rates can be detected earlier and infection control procedures can be implemented sooner. Limitations and future research are located in Chapter 5.

CHAPTER 5: FUTURE RESEARCH

This chapter will discuss future research directions for methods proposed in the three preceding chapters.

In Chapter 2, jackknife estimators of value functions were applied to compare performance of various machine learning models and select the optimal individualized treatment rule for knee osteoarthritis patients. We expect the following future studies to be useful: 1) Exploring more robust models that directly determine the optimal treatment rules. We have observed from the results that, in general, machine learning models which predict well do not necessarily find the optimal treatment rule. The objectives are different: better prediction aims to lower mean squared errors whereas optimal treatment rules aim to increase value functions. For example, there has been recent advances in super learning that directly learn the optimal treatment regime (van der Laan and Luedtke, 2014, 2015; Luedtke and van der Laan, 2016), and we believe it is worth investigating further such robust models; 2) Finding optimal treatment regimes in the setting of multiple decision time points, where data can vary regularly by time and treatment plans can be adjusted periodically rather than fixed for the entire intervention period. This would make the individualized treatment recommendations more up-to-date and adaptive. There are many dynamic models that can be applied to this setting, and we recommend reinforcement learning (Schulte et al., 2014) and Gaussian processes (Wilson and Adams, 2013); 3) Although external validation was investigated via simulations, a new randomized trial on a similar population would be needed for external validation of our findings.

In Chapter 3, we proposed DDROWL where deep neural networks (DNN) were applied to a doubly robust optimization learning problem in the area of precision medicine. The limitation and future research of DDROWL can be summarized in the following directions: i) We imposed

strict inclusion criteria on the NACC data and performed basic preprocessing to the MRI scans. A future research direction would be to relax the inclusion criteria or add similar neuroimaging and clinical data from existing database to increase the size of available data, and apply more flexible powerful image processing tool boxes. The package Nipype in Python (Gorgolewski et al., 2011) possesses many preprocessing tools, such as co-registration, segmentation, and region of interests, that could remove unrelated noise and further standardize the size and region of interest; ii) The grid and random search that we applied in this paper is commonly used but not the most efficient way to tune hyperparameters for all cases, especially when the level of tuning is high. Bayesian optimization (Bergstra et al., 2011) has been shown to be "smarter" in hyperparameter tuning because it uses priors to learn from past evaluations and avoid wasting time in the "bad" part of the hyperparameter space that do not perform well (Bergstra et al., 2013); iii) A sensitivity analysis of different surrogate loss functions provides insights in the influence of loss functions. Such analysis can closely look at whether the performance of Cauchy-Schwarz divergence loss carries over from classification to regression and fully applies to different situations. This can be done using simulated data and clinical data; iv) The current setting is restricted on two interventions and it is possible to extend DDROWL to more than two interventions. This extension can be achieved using balanced policy evaluation (Kallus, 2018; Leete et al., 2019) where the conditional mean squared error can be decomposed into a bias component and a variance component with the goal of minimizing the worse case bias and variance. We hope our work encourages more methodology research in empowering precision medicine using high performance machine learning tools. Although we presented several types of deep learning structure in DDROWL (FFNN, DKL, ResNet), they can be replaced with other network models depending on input data; for example, recurrent neural networks (RNN) for time-series data and generative adversarial network (GAN). We encourage future research to implement more complex DL architectures in DDROWL.

In Chapter 4, a risk-adjusted Anderson-Gill frailty model was applied to monitor survival events. Additionally, a three-component variance estimator was designed to generate confidence

intervals allowing missing data, hierarchies, and recurrent events. There are several limitations and extensions: The level-1 group, the highest level in the hierarchy that we focused on, is usually a social entity or some group of subjects that can vary in size. For example, CF programs in the U.S. could be as large as more than 300 patients or as small as less than 10 patients. Our model did not treat small programs (size less than 50) differently in preprocessing, and the feature selection did not necessarily select program size as an important risk factors. Smaller programs are more likely to create outliers. Although our results imply that it is not always small programs whose risk-adjusted intervals fail to contain the observed number of incidence cases (especially when training period is one year), we think dealing with small programs has potential benefits of improving our model performance and generalizability. More simulations need to be done to study the underlying intensity process. Another potential generalization of our methods would be in the area of multiple event types. The event types could be a mixture of competing risks and not competing risks. Taking the general health of elderly people as an example, both hospitalization and death are considered events with hospitalization as recurrent events and death as a competing risk for hospitalization. In CF, clinicians are often interested in multiple events at the same time, such as CF-related diabetes and liver diseases, which may or may not be competing. Bivariate or multivariate survival models could be applied for multiple events. Finally, this paper serves as the first step into the precision public health paradigm for general IP&C. Precision medicine tailors patient characteristics into the decision making of clinical interventions, where as precision health focuses on precise recommendatiosn on a broader level for larger entities such as schools and hospitals (Kosorok and Laber, 2018; Sperger et al., shed). Since our proposed methods provide contextual intelligence of bacterial infection rates on a program-level for CF, a natural continuum of this in the realm of precision public health is to identify modifiable program-level risk factors and provide strategies tailored to each program to proactively prevent infections. Precision health is a new and growing area that are in need of methodology development.

APPENDIX A: TECHNICAL DETAILS FOR CHAPTER 1

This chapter contains technical details supplemental to the main text of Chapter 2.

Derivation of Influence Function-Inspired Value Function

R_i^{jk} and R_{ji}^{cv} have similar derivations so we omit the notation of jk or cv for cleaner notations. Assume $Y = O_p(1)$ and $E[W] \in (\epsilon, 1-\epsilon)$ for $0 < \epsilon < 0.5$.

$$\begin{aligned}
& \hat{V}(d) - V_0(d) \\
= & \frac{\sum_{i=1}^n U_i}{\sum_{i=1}^n W_i} - \frac{E[U]}{E[W]} \\
= & \frac{n^{-1}\sum_{i=1}^n (U_i - E[U])}{n^{-1}\sum_{i=1}^n W_i} - \frac{n^{-1}E[U]\cdot\sum_{i=1}^n (W_i - E[W])}{(n^{-1}\sum_{i=1}^n W_i)\, E[W]} \\
= & \frac{n^{-1}\sum_{i=1}^n (U_i - E[U])}{E[W] + o_P(1)} - \frac{n^{-1}E[U]\cdot\sum_{i=1}^n (W_i - E[W])}{E[W](E[W] + o_P(1))} \\
= & \frac{n^{-1}\sum_{i=1}^n (U_i - E[U])}{E[W]} - \frac{n^{-1}E[U]\cdot\sum_{i=1}^n (W_i - E[W])}{(E[W])^2} + o_P(1)
\end{aligned}$$

According to (ii) of Theorem 18.7 in Kosorok (2008)

$$\sqrt{n}(\hat{V}(d) - V_0(d)) = \sqrt{n}\sum_{i=1}^n \check{\psi}_i + o_p(1)$$

for a fixed d, the influence function and its estimator would be

$$\begin{aligned}
\check{\psi}_i &= \frac{(U_i - E[U])}{E[W]} - \frac{E[U](W_i - E[W])}{(E[W])^2} = \frac{1}{E[W]}U_i - \frac{E[U]}{(E[W])^2}W_i \\
\ddot{\psi}_i &= \frac{1}{\bar{W}}U_i - \frac{\bar{U}}{\bar{W}^2}W_i
\end{aligned}$$

where R follows a similar form as $\check{\psi}_i$.

Proof of Theorem 2.1

Proof. Let $U_i = \frac{Y_i 1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}}{P(A_i|\boldsymbol{X}_i)}$, $W_i = \frac{1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}}{P(A_i|\boldsymbol{X}_i)}$, $U_n = n^{-1}\sum_{i=1}^n U_i$, and $W_n = n^{-1}\sum_{i=1}^n W_i$. First,

$$\mu_n = \mathbb{E}[U_n] = n^{-1}\sum_{i=1}^n \mathbb{E}\left[\frac{Y_i 1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}}{P(A_i|\boldsymbol{X}_i)}\right] = \mathbb{E}\left[\frac{Y 1\{A = \hat{d}_{n-1}(\boldsymbol{X})\}}{P(A|\boldsymbol{X})}\right].$$

Denote $\tilde{\mu}_n = \mathbb{E}\left[\frac{Y 1\{A=\hat{d}_n(\boldsymbol{X})\}}{P(A|\boldsymbol{X})}\right]$, then

$$\begin{aligned}
\mu_n - \tilde{\mu}_n &= \mathbb{E}\left[\frac{Y}{P(A|\boldsymbol{X})}\left(1\{A = \hat{d}_{n-1}(\boldsymbol{X})\} - 1\{A = \hat{d}_n(\boldsymbol{X})\}\right)\right] \\
&\leq M\mathbb{E}\left[1\{A = \hat{d}_{n-1}(\boldsymbol{X})\} - 1\{A = \hat{d}_n(\boldsymbol{X})\}\right] \\
&\quad + \mathbb{E}\left[\frac{|Y|}{P(A|\boldsymbol{X})} 1\left\{\frac{|Y|}{P(A|\boldsymbol{X})} > M\right\}\right] \\
&\to 0
\end{aligned}$$

where the convergence is based on Assumption 2.1 for the first term and Assumption 2.2, which implies finite first moment, for the second term. Given the first term in Assumption 2.2, we have the following property of the variance

$$\begin{aligned}
\text{Var}[U_n] &= n^{-1}\text{Var}\left[\sum_{i=1}^n U_i\right] \\
&= n^{-2}\sum_{i=1}^n\sum_{j=1}^n [\mathbb{E}(U_i U_j) - \mathbb{E}(U_i)\mathbb{E}(U_j)] \\
&= n^{-2}\sum_{i=1}^n\sum_{j=1}^n \left[\mathbb{E}\left(\frac{Y_i Y_j 1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}1\{A_j = \hat{d}_n^{(-j)}(\boldsymbol{X}_j)\}}{P(A_i|\boldsymbol{X}_i)P(A_j|\boldsymbol{X}_j)}\right) - \mu_n^2\right] \\
&\to n^{-2}\sum_{i=1}^n\sum_{j=1}^n \left[\mathbb{E}\left(\frac{Y_i Y_j 1\{A_i = \hat{d}_n^{(-i,-j)}(\boldsymbol{X}_i)\}1\{A_j = \hat{d}_n^{(-i,-j)}(\boldsymbol{X}_j)\}}{P(A_i|\boldsymbol{X}_i)P(A_j|\boldsymbol{X}_j)}\right) - \mu_n^2\right] \\
&= n^{-2}\sum_{i=1}^n\sum_{j=1}^n \left\{\left[\mathbb{E}\left(\frac{Y 1\{A = \hat{d}_{n-2}(\boldsymbol{X})\}}{P(A|\boldsymbol{X})}\right)\right]^2 - \mu_n^2\right\}
\end{aligned}$$

$$\rightarrow \quad n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\mu_n^2 - \mu_n^2) = 0$$

where the convergences are based on Assumption 2.1. Thus, we have shown that

$$\begin{aligned}\mathbb{E}[U_n] - \tilde{\mu}_n &\rightarrow 0 \\ \text{Var}[U_n] &\rightarrow 0\end{aligned}$$

Applying the same arguments as above to W_n with Assumption 2.1 and the second term in Assumption 2.2,

$$\begin{aligned}\tau_n &= \mathbb{E}[W_n] = \mathbb{E}\left[\frac{1\{A = \hat{d}_{n-1}(\boldsymbol{X})\}}{P(A|\boldsymbol{X})}\right] \\ \tilde{\tau}_n &= \mathbb{E}\left[\frac{1\{A = \hat{d}_n(\boldsymbol{X})\}}{P(A|\boldsymbol{X})}\right] = \mathbb{E}\left\{\mathbb{E}\left[\frac{1\{A = \hat{d}_n(\boldsymbol{X})\}}{P(A = \hat{d}_n(\boldsymbol{X})|\boldsymbol{X})}\middle|\boldsymbol{X}\right]\right\} = 1,\end{aligned}$$

and similarly

$$\begin{aligned}\mathbb{E}[W_n] - 1 &\rightarrow 0 \\ \text{Var}[W_n] &\rightarrow 0\end{aligned}$$

Thus by the weak law of large numbers (WLLN),

$$U_n - \tilde{\mu}_n \underset{p}{\rightarrow} 0 \text{ and } W_n - 1 \underset{p}{\rightarrow} 0$$

which yields

$$\frac{U_n}{W_n} - \tilde{\mu}_n \underset{p}{\rightarrow} 0$$

by the multivariate continuous mapping theorem. This completes the proof because

$$\tilde{\mu}_n = \mathbb{E}\left[\frac{Y1\{A = \hat{d}_n(\boldsymbol{X})\}}{P(A|\boldsymbol{X})}\right] = \mathbb{E}[Y^{\hat{d}_n(\boldsymbol{X})}] = \mathbb{E}[Y|A = \hat{d}_n(\boldsymbol{X})]$$

by applying a version of Radon-Nikodym derivative (i.e. $\frac{dP^d}{dP} = 1\{a = d(\boldsymbol{x})\}/P(a|\boldsymbol{x})$ where P denotes the distribution of $(\boldsymbol{X}, A, Y)$ and P^d denotes the distribution of $(\boldsymbol{X}, A, Y)$ under the decision rule d (Qian and Murphy, 2011)) and since

$$\frac{\sum_{i=1}^{n} \frac{Y_i 1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}}{P(A_i|\boldsymbol{X}_i)}}{\sum_{i=1}^{n} \frac{1\{A_i = \hat{d}_n^{(-i)}(\boldsymbol{X}_i)\}}{P(A_i|\boldsymbol{X}_i)}} - E[Y|A = \hat{d}_n(\boldsymbol{X})] = \frac{U_n}{W_n} - \tilde{\mu}_n.$$

■

APPENDIX B: TECHNICAL DETAILS FOR CHAPTER 2

This chapter contains technical details including as assumptions, proofs, definitions, and other materials supplemental to the main text of Chapter 3.

Derivation of Cauchy-Schwarz Divergence Loss

The Cauchy-Schwarz divergence loss was originally derived from Cauchy-Schwarz inequality and have the following forms to describe the "distance" between two vectors or two probability density functions (PDFs), respectively

$$D_{cs}(\boldsymbol{x}, \boldsymbol{y}) = -\log \frac{\boldsymbol{x}^T \boldsymbol{y}}{||\boldsymbol{x}||_2 ||\boldsymbol{y}||_2}$$

$$D_{cs}(p, q) = -\log \frac{\int p(x)q(x)dx}{\sqrt{\int p^2(x)dx \int q^2(x)dx}}.$$

We first convert the random variables in (3.16) from $\{-1, 1\}$ to $\{0, 1\}$ so they match with the notations of Bernoulli distributions, i.e., $U(d) = 1\{A \cdot \text{sign}(Y - \hat{r}(X)) = 1\} = U \perp d$ and $W(d) = 1\{d(X) = 1\}$, where d is short for $d(x)$. Thus,

$$1\{A \cdot sign(Y - \hat{r}(X)) \neq d(X)\} = 1\{U(d) \neq W(d)\}.$$

Let $w(d) = P(W(d) = 1) = P(d(x) = 1) = \frac{e^{f(x)}}{1+e^{f(x)}} = p(f(x))$ for some function $f(x)$ and $u(d) = u$. Note that $u \perp d$. We use the PDF version of Cauchy-Schwarz divergence loss to rewrite the 0-1 loss into

$$
\begin{aligned}
D_{cs}(u,w) &= -\log\left\{\frac{\sum_d uw(d)}{\sqrt{\sum_d u^2 \sum_d w^2(d)}}\right\} \\
&= -\log\left\{\frac{\sum_d w(d)}{\sqrt{\sum_d w^2(d)}}\right\} + c \\
&= -\log\left\{\frac{\left(\frac{e^{f(x)}}{1+e^{f(x)}}\right)^u \left(\frac{1}{1+e^{f(x)}}\right)^{1-u}}{\sqrt{\left(\frac{e^{f(x)}}{1+e^{f(x)}}\right)^2 + \left(\frac{1}{1+e^{f(x)}}\right)^2}}\right\} + c \\
&= -\log\left\{\frac{\frac{e^{uf(x)}}{1+e^{f(x)}}}{\frac{(1+e^{2f(x)})^{1/2}}{1+e^{f(x)}}}\right\} + c \\
&= -uf(x) + \frac{1}{2}\log\left(1+e^{2f(x)}\right) + c \\
&= -1\{a \cdot \text{sign}(y - \hat{r}(x)) = 1\} f(x) + \frac{1}{2}\log\left(1+e^{2f(x)}\right) + c
\end{aligned}
$$

where c is a constant. The third equality comes from the reasoning that $d(x) = 1$ when $P(d(x) = 1) = \frac{e^{f(x)}}{1+e^{f(x)}} > \frac{1}{2}$ which is when $f(x) > 0$. Thus, $d(x) = \text{sign}\, f(x)$. The second term in the last line is a function in the form $\phi(t) = \frac{1}{k}\log(1 + e^{kt})$ which serves as a penalty term for regularization of $f(x)$. For a fixed $k > 0$, it approaches to 0 when $t \to -\infty$ and grows infinitely large when $t \to \infty$. The derivative of this function when $k = 1$ is the famous sigmoid function $\frac{e^t}{1+e^t} = (1 + e^{-t})^{-1}$.

Proof of Theorem 3.2

This is a proof of the consistency of weighted bootstrap for fixed $\theta \in \Theta$, which is a special, simpler case of the original consistency statement $\forall\theta \in \Theta$.

Proof. For simplicity, denote $f_\theta = f_\theta(\boldsymbol{X}_i)$. First, we show that the expectation is 0.

$$
\begin{aligned}
E\left|n^{-1}\sum_{i=1}^n \left(\frac{\omega_i}{m} - \hat{p}_i\right) h(f_\theta, \boldsymbol{X}_i, A_i)\right| &\leq n^{-1}\sum_{i=1}^n E\left|\frac{\omega_i}{m} - \hat{p}_i\right| \cdot |h(f_\theta, \boldsymbol{X}_i, A_i)| \\
&= n^{-1}\sum_{i=1}^n \left|\frac{m\hat{p}_i}{m} - \hat{p}_i\right| \cdot |h(f_\theta, \boldsymbol{X}_i, A_i)| = 0
\end{aligned}
$$

Second, we show the variance converges to 0.

$$
\begin{aligned}
& Var\left|n^{-1}\sum_{i=1}^{n}\left(\frac{\omega_i}{m}-\hat{p}_i\right)h(f_\theta,\boldsymbol{X}_i,A_i)\right| \\
\leq\ & Var\left(n^{-1}\sum_{i=1}^{n}\left|\frac{\omega_i}{m}-\hat{p}_i\right|\cdot|h(f_\theta,\boldsymbol{X}_i,A_i)|\right) \\
=\ & n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}Cov\left[\left|\frac{\omega_i}{m}-\hat{p}_i\right|\cdot|h(f_\theta,\boldsymbol{X}_i,A_i)|,\left|\frac{\omega_j}{m}-\hat{p}_j\right|\cdot|h(f_\theta,\boldsymbol{X}_j,A_j)|\right] \\
\leq\ & n^{-4}\sum_{i=1}^{n}\sum_{j=1}^{n}Cov\left(\omega_i,\omega_j\right)\cdot|h(f_\theta,\boldsymbol{X}_i,A_i)|\cdot|h(f_\theta,\boldsymbol{X}_j,A_j)| \\
=\ & n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}(m\hat{p}_i(1-\hat{p}_i)1\{i=j\}-m\hat{p}_i\hat{p}_j1\{i\neq j\})\cdot|h(f_\theta,\boldsymbol{X}_i,A_i)|\cdot|h(f_\theta,\boldsymbol{X}_j,A_j)| \\
=\ & n^{-4}\Bigg[\sum_{i=1}^{n}\left(m\hat{p}_i(1-\hat{p}_i)h^2(f_\theta,\boldsymbol{X}_i,A_i)\right) \\
& \quad -\sum_{i,j=1,i\neq j}^{n}m\hat{p}_i\hat{p}_j|h(f_\theta,\boldsymbol{X}_i,A_i)|\cdot|h(f_\theta,\boldsymbol{X}_j,A_j)|\Bigg] \\
\leq\ & n^{-3}\Bigg[\sum_{i=1}^{n}\hat{p}_i(1-\hat{p}_i)h^2(f_\theta,\boldsymbol{X}_i,A_i)-\left(\sum_{i=1}^{n}\hat{p}_i|h(f_\theta,\boldsymbol{X}_i,A_i)|\right)^2 \\
& \quad +\sum_{i=1}^{n}\hat{p}_i^2h^2(f_\theta,\boldsymbol{X}_i,A_i)\Bigg] \\
=\ & n^{-3}\left[\sum_{i=1}^{n}\hat{p}_ih^2(f_\theta,\boldsymbol{X}_i,A_i))-\left(\sum_{i=1}^{n}\hat{p}_i|h(f_\theta,\boldsymbol{X}_i,A_i)|\right)^2\right] \\
\leq\ & n^{-3}\sum_{i=1}^{n}h^2(f_\theta,\boldsymbol{X}_i,A_i) \\
\leq\ & n^{-3}\sum_{i=1}^{n}H(\boldsymbol{X}_i)=o_p(1)
\end{aligned}
$$

The last inequality is because $H(\boldsymbol{X})$ is the envelope of $\{h(f_\theta,\boldsymbol{X},A:\theta\in\Theta)\}$. ■

Definitions of Value Functions

We use the same CV definition of value function estimate and its variance estimator as in Jiang et al. (2020b). Let $j=1,\ldots,MK$ denote all MK tuning folds regardless repetition

across M repetitions and K CV folds and $i = 1, \ldots, n_j$ be the ith observation in the jth overall fold. The CV estimated value function was used to compare tuning performance as in

$$\widehat{V}^{cv}(\hat{d}^{(-j)}_{n_{tr}}) = \frac{\sum_{j=1}^{MK}\sum_{i=1}^{n_j} U_{ji}}{\sum_{j=1}^{MK}\sum_{i=1}^{n_j} W_{ji}}$$

where $W_{ji} = \frac{1\{A_{ji}=\hat{d}^{(-j)}_{n_{tr}}(\boldsymbol{X}_{ji})\}}{\hat{P}(A_{ji}|\boldsymbol{X}_{ji})}$, $U_{ji} = Y_{ji}W_{ji}$, $\hat{d}^{(-j)}_{n_{tr}}$ is the decision rule estimated from a training set of size n_{tr} with the jth fold left out, and $\hat{P}(A_{ji}|\boldsymbol{X}_{ji})$ is the estimated propensity score (known for randomized trials). Its standard deviation was used to measure the estimation uncertainty

$$\widehat{\text{Var}}[\,\widehat{V}^{cv}(\hat{d}^{(-j)}_{n_{tr}})] = \frac{1}{K(MK-1)}\sum_{j=1}^{MK}\sum_{i=1}^{n_j} R_{ji}^2$$

where $R_{ji} = \frac{1}{W_j}U_{ji} - \frac{\bar{U}_j}{\bar{W}_j^2}W_{ji}$ is an influence function-inspired form of the value function with $\bar{U}_j = \sum_{i=1}^{n_j} U_{ji}$ and $\bar{W}_j = \sum_{i=1}^{n_j} W_{ji}$. By definition, $\sum_{j=1}^{MK}\sum_{i=1}^{n_j} R_{ji} = 0$.

For testing results, there is no CV so $j = 1, \ldots, M$. The estimated value function is

$$\widehat{V}(\hat{d}_{n_{te}}) = \frac{\sum_{i=1}^{n_{te}} U_i}{\sum_{i=1}^{n_{te}} W_i}$$

where U_i, W_i are defined similarly as the U_{ji}, W_{ji} but with $i = 1, \ldots, n_{te}$ and decision rule $\hat{d}_{n_{te}}$. Its standard deviation is

$$\widehat{\text{Var}}[\widehat{V}(\hat{d}_{n_{te}})] = \frac{\sum_{j=1}^{M}(\widehat{V}(\hat{d}_{n_{te},j}) - \bar{V}(\hat{d}_{n_{te},M}))^2}{M-1}$$

where $\hat{d}_{n_{te},j}$ is the single estimated decision rule from the jth repetition and $\bar{V}(\hat{d}_{n_{te},M}) = \sum_{j=1}^{M}\widehat{V}(\hat{d}_{n_{te},j})$ is the average estimated value functions over M single estimated decision rules.

Computation Time

Below is computation time in seconds for all the dimensions and scenarios in Section 3.3 simulations. Trends are generally consistent across dimensions and scenarios. More specifically, ℓ_1-PLS is the fastest for all scenarios and dimensions, followed by Q-RF, AOL-L, and RWL-L.

AOL-L or AOL-VSL is faster than RWL-L, which confirms that AOL is an improvement of RWL in terms of speed. The computation time increases by a lot for RWL-L when the dimension increases from 5 to 100. Both DDROWL methods are faster than AOL methods. Note that number of simulations vary. For example, $p = 5$ has large DKL-1L computation time because we tuned on 100 simulations whereas the rest of DKL were tuned on 10 simulations. Computation time is roughly on the same scale across scenarios and vary within reasonable margins. So far the different scenarios and dimensions have been programmed to multiple processed, and one future improvement is to parallelize the 100 simulations to speed up the computation.

More Preprocessing Details of the NACC Clinical and Imaging Data

Variables related to smoking (e.g., consumption of tobacco in the past 30 days, number of years smoking, whether or not the subject quit smoking, etc.) are combined into two variables: an indicator of current smoker (has quit smoking, or no tobacco consumed in the last 30 days) and an indicator of ever smoked (smoked cigarettes in the last 30 days, or more than 100 cigarettes in life, or non-zero years of smoking, or at least 1 cigarette smoked per day on average).

Diabetes is redefined as a binary variables with 1 representing Type 1, Type 2, or other type of diabetes such as diabetes insipidus, latent autoimmune diabetes /Type 1.5, and gestational diabetes, and 0 representing no diabetes reported.

Table B.25: Computation time in seconds for each scenario and model, $n_{tr} = 800$

$p = 5$				
Model/ Scenario	1	2	3	4
ℓ_1-PLS	190	170	170	168
Q-RF	4609	4067	4249	4227
RWL-L	87421	223820	70889	157953
AOL-L	726	750	444	473
AOL-G	395415	416133	352826	307203
FFNN-1L	178469	158252	183132	183061
FFNN-2L	158845	157140	159840	159615
DKL-1L	855820	829547	783503	800124
DKL-2L	212892	220662	214471	212960
$p = 25$				
Model/ Scenario	1	2	3	4
ℓ_1-PLS	311	311	308	314
Q-RF	5595	5578	5764	5830
RWL-L	315525	301377	363569	369956
AOL-VSL	18769	22477	17027	19253
FFNN-1L	149716	112517	101992	113389
FFNN-2L	139633	152411	399695	300425
DKL-1L	115756	118899	108750	118899
DKL-2L	232065	249131	241691	229938
$p = 100$				
Model/ Scenario	5	6	7	8
ℓ_1-PLS	773	720	3495	3517
Q-RF	47960	49174	46692	69191
RWL-L	133812	136468	270067	1066383
FFNN-1L	511830	513246	602744	615860
FFNN-2L	328991	342228	277136	324328
DKL-1L	172889	135249	126635	162101
DKL-2L	280842	363113	339546	326032
$p = 800$				
Model/ Scenario	5	6	7	8
ℓ_1-PLS	3157	2947	1449	1763
Q-RF	10092	10837	10077	22648
FFNN-1L	615814	637164	626503	627302
FFNN-2L	534719	577726	576447	434781
DKL-1L	525672	548699	540052	524156
DKL-2L	269259	340318	370372	363733

APPENDIX C: TECHNICAL DETAILS FOR CHAPTER 3

This chapter contains technical details including as definitions, extra analysis results, and other materials supplemental to the main text of Chapter 4.

Additional Clinical Results

When there are three years of training data, Section 4.4.4 shows estimated coverage of MRSA incidence in 2012-2014 and 2015 using 2012-2014 training data. The corresponding PA incidence coverage is summarized in Table C.26 and we see that PA has similar patterns as MRSA. There is again overcoverage for the 2015 validation but not the 2012-2014 validation. The absolute coverage difference decreases as confidence level increases, with almost no overcoverage when confidence level is 0.995. This section also contains distribution comparison between estimated and observed survival events for both MRSA and PA (see Figure C.12). The top figure has more incidence cases because it has 2012-2014, three years of data, compared with the bottom figure which has only 2015 data. Visually inspecting the histograms, we conclude that the estimated and observed distributions are both skewed to the right with similar modes and tails, regardless of bacteria and training data. Additionally, the estimated Spearman correlation coefficients are 0.87 (MRSA) and 0.91 (PA) for 2012-2014 validation and 0.58 (MRSA) and 0.80 (PA) for 2015 validation, all of which are moderate to strong correlations. Overall the correlations are stronger for CF data than simulated data and we speculate this is due to model setting, parameter estimation, and how much variance can be explained by the covariates.

Table C.26: Results of risk-adjusted PA incidence model of 2012-2014 CFFPR data validated on 2012-2014 and 2015 data separately, where α is significance level, m number of blocks in block jackknife estimation of variance, Coverage is proportion of programs whose true number of incidence cases in the test set is contained in the risk-adjusted $1-\alpha$ confidence interval trained on 2012-2014 data, and AbsCovDiff is absolute difference between the coverage and $1-\alpha$.

PA		2012-2014 Validation		2015 Validation	
$1-\alpha$	m	Coverage	AbsCovDiff	Coverage	AbsCovDiff
0.7	5	0.59	0.110	0.89	0.190
	10	0.64	0.060	0.93	0.230
	15	0.67	0.030	0.94	0.240
0.8	5	0.69	0.110	0.94	0.140
	10	0.73	0.070	0.95	0.150
	15	0.78	0.020	0.95	0.150
0.9	5	0.83	0.070	0.97	0.070
	10	0.86	0.040	0.97	0.070
	15	0.89	0.010	0.97	0.070
0.95	5	0.90	0.050	0.98	0.030
	10	0.92	0.030	0.98	0.030
	15	0.93	0.020	0.99	0.040
0.995	5	0.96	0.035	0.99	0.005
	10	0.96	0.035	0.99	0.005
	15	0.97	0.025	0.99	0.005

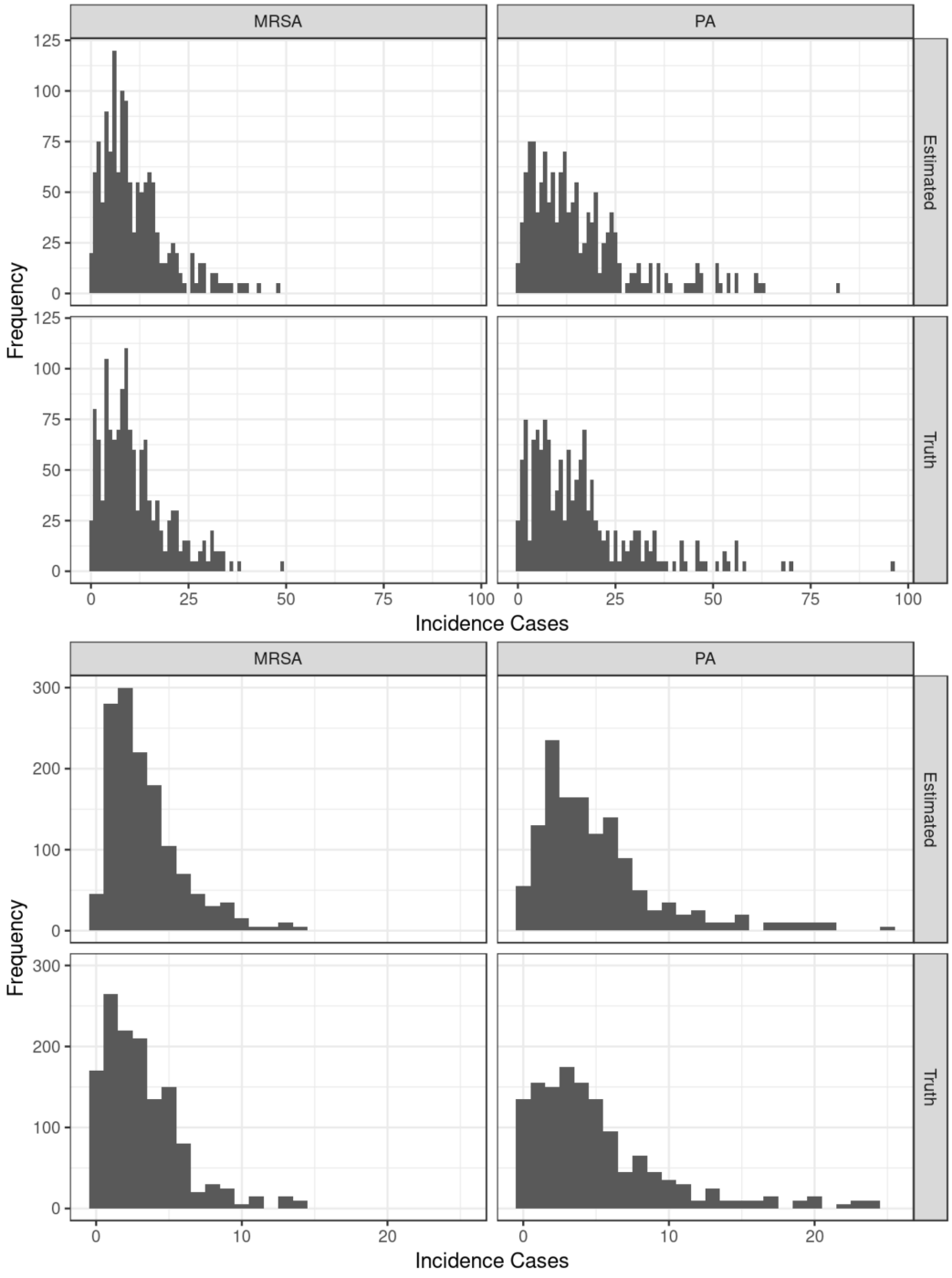

Figure C.12: Histogram of estimated versus observed (true) number of 2012-2014 (top) and 2015 (bottom) MRSA and PA incidence cases where the risk-adjusted model is trained from 2012-2014 data

Glossary for Key Notations Introduced in the Methods Section 4.2

1. $*$: The indicator of the block jackknife data, as opposed to the original data.
2. b: The indicator of block in the block jackknife.
3. β_0: The covariate coefficient in the intensity process R_{0j}, assuming effects are the same for all i, j, k's.
4. $\hat{\beta}_l$: The indicator of multiple imputation copy.
5. $\hat{\bar{\beta}}_M$: The pooled estimator of β_0 averaged over across M $\hat{\beta}_{nl}$'s.
6. $\hat{\beta}_{nl}$: The estimator of β_0 from the frailty model for the lth multiple imputation dataset.
7. $\hat{\beta}_{nl}^{(-v)}$: The estimator of β_0 from the frailty model for the lth multiple imputation dataset but with the vth level-1 group taken out.
8. c_j: Total number of unique level-2 group variable (i) in the jth level-1 group.
9. $d\hat{\Lambda}_{nl}(t)$: The estimator of the hazard function at time t for the lth multiple imputed data.
10. $d\hat{\Lambda}_{nl}(t)$: The estimator of the hazard function at time t for the lth multiple imputed data.
11. $d\hat{\Lambda}_{nl}^{(-v)}(t)$: The estimator of the hazard function at time t for the lth multiple imputation data but with the vth level-1 group taken out.
12. $dN_{ijk}(t)$: The indicator of an observed survival event at time t.
13. i: The indicator of level-2 group variable in the survival data with a three-level hierarchical structure, e.g., CF patient (the middle level).
14. j: The indicator of level-1 group variable in the survival data with a three-level hierarchical structure, e.g., CF program (the highest level).
15. k: The indicator of observations in a level-2 group, e.g., visits at a CF program.

16. L_{ijk}: The lower bound of the at-risk interval for the jth level-1, ith level-2, and kth encounter; used in $d\hat{\lambda}_{nl}(s)$.

17. M: Number of multiple imputation copies.

18. m: Number of blocks in the block jackknife.

19. m_{ij}: Total number of observations (denoted by k) in the ith level-2 group and the jth level-1 group.

20. N: Total number of unique level-1 group variable (j).

21. N_j: The observed number of survival events for the j level-1 group.

22. $\hat{N}_{jl}$: The estimated number of survival events for the jth level-1 group and lth multiple imputed data.

23. $\hat{N}_{j\bullet}$: Th e estimated number of survival events for the jth level-1 group across M multiple imputations.

24. $\hat{N}_j\left(\hat{\beta^*}^{(-b)}, d\hat{\Lambda^*}^{(-b)}\right)$: The estimated survival events for the jth level-1 group based on estimated parameters of data with the bth block removed.

25. $\hat{N}_{jl}^{(-v)}$: The estimated survival events for the jth level-1 group based on estimated parameters of data with the vth level-1 group removed.

26. n: Total sample size, $n = \sum_{j=1}^{N}\sum_{i=1}^{c_j} m_{ij}$.

27. p: The dimension of $\boldsymbol{Z}_{ijk}(t)$; number of all covariates of interest.

28. $q_{m,N}$: The number of elements in each block of the block jackknife method.

29. R_{0j}: The true intensity process for the jth level-1 group.

30. S_j^*: The estimated variance of the survival events pooled across all m block jackknife datasets.

31. S^*_{jl}: The estimated variance of the survival events pooled across all m block jackknife datasets using the lth multiple imputation data as the original data.

32. $\hat{s}^2_{j\bullet}$: The estimated multiple imputation variance in the jth level-1 group.

33. T_{ijk}: At-risk time interval for the kth encounter, the ith level-2 group, and the jth level-1 group.

34. $\hat{T}_j$: The test statistic of the Z-test comparing the estimated number of events with the observed number of events.

35. t: Time.

36. $\hat{\theta^*}_j^{(-b)}$: The estimator of the survival events for the jth level-1 group based on estimated parameters of data with the bth block removed.

37. $\bar{\theta}^*_j$: The estimated survival events pooled across all m block jackknife datasets.

38. U_{ijk}: The upper bound of the at-risk interval for the jth level-1, ith level-2, and kth encounter; used in $d\hat{\lambda}_{nl}(s)$.

39. $\hat{\bar{V}}_j$: The estimated variance of the estimated number of survival events $\hat{N}_{j\bullet}$.

40. $\hat{V}_{j\bullet}$: The estimated across-group variance in the jth level-1 group.

41. $Var\left(\hat{\bar{\beta}}_M\right)$: The pooled estimator of the variance of β_0 across M multiple imputations of β_{nl}.

42. v: Indicator of the level-1 group that is taken out of training set in the jackknife method.

43. $\boldsymbol{Z}_{ijk}(t)$: The p-dimensional covariate variable at time t including both time-invariant and time-varying variables, for the kth event, the ith level-2 group, and the jth level-1 group.

44. $\boldsymbol{Z}_{ijkl}(t)$ The p-dimensional covariate variable in the lth multiple imputed data at time t including both time-invariant and time-varying variables, for the kth encounter, the ith level-2 group, the jth level-1 group.

BIBLIOGRAPHY

Amorim, L. D. and Cai, J. (2015). Modelling recurrent events: a tutorial for analysis in epidemiology. *International journal of epidemiology*, 44(1):324–333.

Andersen, P. K. and Gill, R. D. (1982). Cox's regression model for counting processes: a large sample study. *The annals of statistics*, pages 1100–1120.

Beekly, D. L., Ramos, E. M., van Belle, G., Deitrich, W., Clark, A. D., Jacka, M. E., Kukull, W. A., et al. (2004). The national alzheimer's coordinating center (nacc) database: an alzheimer disease database. *Alzheimer Disease & Associated Disorders*, 18(4):270–277.

Bengio, Y. and Grandvalet, Y. (2004). No unbiased estimator of the variance of k-fold cross-validation. *Journal of machine learning research*, 5(Sep):1089–1105.

Bergstra, J., Yamins, D., and Cox, D. D. (2013). Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. *Jmlr*.

Bergstra, J. S., Bardenet, R., Bengio, Y., and Kégl, B. (2011). Algorithms for hyper-parameter optimization. In *Advances in neural information processing systems*, pages 2546–2554.

Besser, L., Kukull, W., Knopman, D. S., Chui, H., Galasko, D., Weintraub, S., Jicha, G., Carlsson, C., Burns, J., Quinn, J., et al. (2018). Version 3 of the national alzheimer's coordinating center's uniform data set. *Alzheimer disease and associated disorders*, 32(4):351.

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

Breiman, L. et al. (1996). Heuristics of instability and stabilization in model selection. *The annals of statistics*, 24(6):2350–2383.

Broyden, C. G. (1970). The convergence of a class of double-rank minimization algorithms: 2. the new algorithm. *IMA journal of applied mathematics*, 6(3):222–231.

Burman, P. (1989). A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods. *Biometrika*, 76(3):503–514.

Carpenter, J. and Kenward, M. (2012). *Multiple imputation and its application*. John Wiley & Sons.

Center for Disease Control, P. (2018). The nhsn standardized infection ratio (sir): a guide to the sir. *Available from:)(Accessed November 1, 2018) https://www. cdc. gov/nhsn/pdfs/ps-analysis-resources/nhsn-sir-guide. pdf Date*.

Chipman, H. A., George, E. I., McCulloch, R. E., et al. (2010). Bart: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298.

Cox, D. R. (1975). Partial likelihood. *Biometrika*, 62(2):269–276.

Cross, M., Smith, E., Hoy, D., Nolte, S., Ackerman, I., Fransen, M., Bridgett, L., Williams, S., Guillemin, F., Hill, C. L., et al. (2014). The global burden of hip and knee osteoarthritis: estimates from the global burden of disease 2010 study. *Annals of the rheumatic diseases*, 73(7):1323–1330.

Dasgupta, S., Goldberg, Y., Kosorok, M. R., et al. (2019). Feature elimination in kernel machines in moderately high dimensions. *The Annals of Statistics*, 47(1):497–526.

Delgado-Rodríguez, M. and Llorca, J. (2005). Caution should be exercised when using the standardized infection ratio. *Infection Control & Hospital Epidemiology*, 26(1):8–9.

Efron, B. (1977). The efficiency of cox's likelihood function for censored data. *Journal of the American statistical Association*, 72(359):557–565.

Efron, B. and Tibshirani, R. J. (1994). *An introduction to the bootstrap*. CRC press.

Fletcher, R. (1970). A new approach to variable metric algorithms. *The computer journal*, 13(3):317–322.

Fletcher, R. and Reeves, C. M. (1964). Function minimization by conjugate gradients. *The computer journal*, 7(2):149–154.

Friedman, J., Hastie, T., and Tibshirani, R. (2001). *The elements of statistical learning*, volume 1. Springer series in statistics New York.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1.

Gardner, J., Pleiss, G., Weinberger, K. Q., Bindel, D., and Wilson, A. G. (2018). Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. In *Advances in Neural Information Processing Systems*, pages 7587–7597.

Givens, G. and Hoeting, J. (2005). Computational statistics (wiley series in computation statistics). *Wiley New Jersey*.

Goldfarb, D. (1970). A family of variable-metric methods derived by variational means. *Mathematics of computation*, 24(109):23–26.

Gorgolewski, K., Burns, C. D., Madison, C., Clark, D., Halchenko, Y. O., Waskom, M. L., and Ghosh, S. S. (2011). Nipype: a flexible, lightweight and extensible neuroimaging data processing framework in python. *Front Neuroinform*, 5.

Grund, S., Robitzsch, A., and Luedtke, O. (2019). *mitml: Tools for Multiple Imputation in Multilevel Modeling*. R package version 0.3-7.

Gustafson, T. L. (2006). Three uses of the standardized infection ratio (sir) in infection control. *Infection Control & Hospital Epidemiology*, 27(4):427–430.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Holloway, S. T., Laber, E. B., Linn, K. A., Zhang, B., Davidian, M., and Tsiatis, A. A. (2018). *DynTxRegime: Methods for Estimating Optimal Dynamic Treatment Regimes*. R package version 3.2.

Jamshidi, A., Pelletier, J.-P., and Martel-Pelletier, J. (2018). Machine-learning-based patient-specific prediction models for knee osteoarthritis. *Nature Reviews Rheumatology*, page 1.

Janocha, K. and Czarnecki, W. M. (2017). On loss functions for deep neural networks in classification. *arXiv preprint arXiv:1702.05659*.

Jiang, X., Nelson, A., Cleveland, R., Beaver, D., Schwartz, T., Arbeeva, L., Alvarez, C., Callahan, L., Messier, S., Loeser, R., et al. (2020a). Technical background for "a precision medicine approach to develop optimal exercise and weight loss treatments for overweight and obese adults with knee osteoarthritis". *arXiv preprint arXiv:2001.09930*.

Jiang, X., Nelson, A. E., Cleveland, R. J., Beavers, D. P., Schwartz, T. A., Arbeeva, L., Alvarez, C., Callahan, L. F., Messier, S., Loeser, R., et al. (2020b). A precision medicine approach to develop and internally validate optimal exercise and weight loss treatments for overweight and obese adults with knee osteoarthritis. *Arthritis Care & Research*.

Kallus, N. (2018). Balanced policy evaluation and learning. In *Advances in Neural Information Processing Systems*, pages 8909–8920.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kohavi, R. et al. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada.

Kosorok, M. R. (2008). Introduction to empirical processes. *Introduction to Empirical Processes and Semiparametric Inference*, pages 77–79.

Kosorok, M. R. and Laber, E. B. (2018). Precision medicine. *Annual reviews of statistics and its application*. In press.

Laber, E. B., Lizotte, D. J., Qian, M., Pelham, W. E., and Murphy, S. A. (2014). Dynamic treatment regimes: Technical challenges and applications. *Electronic journal of statistics*, 8(1):1225.

Lawless, J. F. (1987). Regression methods for poisson process data. *Journal of the American Statistical Association*, 82(399):808–815.

Leete, O. E., Kallus, N., Hudgens, M. G., Napravnik, S., and Kosorok, M. R. (2019). Balanced policy evaluation and learning for right censored data. *arXiv preprint arXiv:1911.05728*.

Liaw, A., Wiener, M., et al. (2002). Classification and regression by randomforest. *R news*, 2(3):18–22.

Liaw, R., Liang, E., Nishihara, R., Moritz, P., Gonzalez, J. E., and Stoica, I. (2018). Tune: A research platform for distributed model selection and training. *arXiv preprint arXiv:1807.05118*.

Lin, D. (2007). On the breslow estimator. *Lifetime data analysis*, 13(4):471–480.

Liu, Y., Wang, Y., Kosorok, M. R., Zhao, Y., and Zeng, D. (2016). Robust hybrid learning for estimating personalized dynamic treatment regimens. *arXiv preprint arXiv:1611.02314*.

Logan, B. R., Zhang, M.-J., and Klein, J. P. (2011). Marginal models for clustered time-to-event data with competing risks using pseudovalues. *Biometrics*, 67(1):1–7.

Luedtke, A. R. and van der Laan, M. J. (2016). Super-learning of an optimal dynamic treatment rule. *The international journal of biostatistics*, 12(1):305–332.

Ma, S., Kosorok, M. R., et al. (2005). Penalized log-likelihood estimation for partly linear transformation models with current status data. *The Annals of Statistics*, 33(5):2256–2290.

Messier, S. P., Legault, C., Mihalko, S., Miller, G. D., Loeser, R. F., DeVita, P., Lyles, M., Eckstein, F., Hunter, D. J., Williamson, J. D., et al. (2009). The intensive diet and exercise for arthritis (idea) trial: design and rationale. *BMC musculoskeletal disorders*, 10(1):93.

Messier, S. P., Mihalko, S. L., Legault, C., Miller, G. D., Nicklas, B. J., DeVita, P., Beavers, D. P., Hunter, D. J., Lyles, M. F., Eckstein, F., et al. (2013). Effects of intensive diet and exercise on knee joint loads, inflammation, and clinical outcomes among overweight and obese adults with knee osteoarthritis: the idea randomized clinical trial. *Jama*, 310(12):1263–1273.

Messier, S. P., Resnik, A. E., Beavers, D. P., Mihalko, S. L., Miller, G. D., Nicklas, B. J., DeVita, P., Hunter, D. J., Lyles, M. F., Eckstein, F., et al. (2018). Intentional weight loss in overweight and obese patients with knee osteoarthritis: Is more better? *Arthritis care & research*, 70(11):1569–1575.

Michel, J. (2020). ENNUI: An Elegant Neural Network User Interface.

Nelder, J. A. and Mead, R. (1965). A simplex method for function minimization. *The computer journal*, 7(4):308–313.

Nelson, A. E., Allen, K. D., Golightly, Y. M., Goode, A. P., and Jordan, J. M. (2014). A systematic review of recommendations and guidelines for the management of osteoarthritis: the chronic osteoarthritis management initiative of the us bone and joint initiative. In *Seminars in arthritis and rheumatism*, volume 43, pages 701–712. Elsevier.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in pytorch. *NIPS*.

Paterek, A. (2007). Improving regularized singular value decomposition for collaborative filtering. In *Proceedings of KDD cup and workshop*, volume 2007, pages 5–8.

Pickles, A. and Crouchley, R. (1995). A comparison of frailty models for multivariate survival data. *Statistics in Medicine*, 14(13):1447–1461.

Polley, E. C. and Van Der Laan, M. J. (2010). Super learner in prediction. *U.C. Berkeley Division of Biostatistics Working Paper Series. Working Paper 266.*

Qian, M. and Murphy, S. A. (2011). Performance guarantees for individualized treatment rules. *The Annals of Statistics*, 39(2):1180.

Quenouille, M. H. (1949). Approximate tests of correlation in time-series 3. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 45, pages 483–484. Cambridge University Press.

R Core Team (2019). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rashid, N. U., Luckett, D. J., Chen, J., Lawson, M. T., Wang, L., Zhang, Y., Laber, E. B., Liu, Y., Yeh, J. J., Zeng, D., et al. (2019). High dimensional precision medicine from patient-derived xenografts. *arXiv preprint arXiv:1912.06667*.

Robins, J. M., Rotnitzky, A., and Zhao, L. P. (1994). Estimation of regression coefficients when some regressors are not always observed. *Journal of the American statistical Association*, 89(427):846–866.

Rotnitzky, A., Robins, J. M., and Scharfstein, D. O. (1998). Semiparametric regression for repeated outcomes with nonignorable nonresponse. *Journal of the american statistical association*, 93(444):1321–1339.

Rubin, D. B. (2004). *Multiple imputation for nonresponse in surveys*, volume 81. John Wiley & Sons.

Saiman, L., Siegel, J., et al. (2003). Infection control recommendations for patients with cystic fibrosis: microbiology, important pathogens, and infection control practices to prevent patient-to-patient transmission. *Infection Control & Hospital Epidemiology*, 24(S5):S6–S52.

Schulte, P. J., Tsiatis, A. A., Laber, E. B., and Davidian, M. (2014). Q-and a-learning methods for estimating optimal dynamic treatment regimes. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 29(4):640.

Shanno, D. F. (1970). Conditioning of quasi-newton methods for function minimization. *Mathematics of computation*, 24(111):647–656.

Sparapani, R. A., Logan, B. R., McCulloch, R. E., and Laud, P. W. (2016). Nonparametric survival analysis using bayesian additive regression trees (bart). *Statistics in medicine*, 35(16):2741–2753.

Sperger, J., Freeman, N. L. B., Jiang, X., Bang, D., de Marchi Daniel, and Kosorok, M. R. (unpublished). The future of precision health is data-driven decision support. Statistical Analysis and Data Mining. Under review.

Stekhoven, D. J. and Bühlmann, P. (2011). Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118.

Stoudemire, W., Jiang, X., Kosorok, M. R., and Muhlebach, M. S. (unpublished). A novel tool to monitor risk-adjusted incidence rates for mrsa and p. aeruginosa in cystic fibrosis. Unpublished.

Stoudemire, W., Jiang, X., Zhou, J. J., Maykowski, P., Kosorok, M. R., Muhlebach, M. S., and Saiman, L. (2019). Cystic fibrosis program characteristics associated with adoption of 2013 infection prevention and control recommendations. *American journal of infection control*, 47(9):1090–1095.

Talo, M., Baloglu, U. B., Yıldırım, Ö., and Acharya, U. R. (2019). Application of deep transfer learning for automated brain abnormality classification using mr images. *Cognitive Systems Research*, 54:176–188.

Therneau, T. M. (2019). *coxme: Mixed Effects Cox Models*. R package version 2.2-14.

Therneau, T. M., Grambsch, P. M., and Pankratz, V. S. (2003). Penalized survival models and frailty. *Journal of computational and graphical statistics*, 12(1):156–175.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Tukey, J. (1958). Bias and confidence in not quite large samples. *Ann. Math. Statist.*, 29:614.

van der Laan, M. J. and Luedtke, A. R. (2014). Targeted learning of an optimal dynamic treatment, and statistical inference for its mean outcome. *U.C. Berkeley Division of Biostatistics Working Paper Series*.

van der Laan, M. J. and Luedtke, A. R. (2015). Targeted learning of the mean outcome under an optimal dynamic treatment rule. *Journal of causal inference*, 3(1):61–95.

Van der Laan, M. J., Polley, E. C., and Hubbard, A. E. (2007). Super learner. *Statistical applications in genetics and molecular biology*, 6(1).

Vina, E. R. and Kwoh, C. K. (2018). Epidemiology of osteoarthritis: literature update. *Current opinion in rheumatology*, 30(2):160–167.

Wei, L., , and Glidden, D. V. (1997). An overview of statistical methods for multiple failure time data in clinical trials. *Statistics in medicine*, 16(8):833–839.

Wilson, A. and Adams, R. (2013). Gaussian process kernels for pattern discovery and extrapolation. In *International conference on machine learning*, pages 1067–1075.

Wilson, A. G., Hu, Z., Salakhutdinov, R., and Xing, E. P. (2016). Deep kernel learning. In *Artificial Intelligence and Statistics*, pages 370–378.

Yang, W., Jepson, C., Xie, D., Roy, J. A., Shou, H., Hsu, J. Y., Anderson, A. H., Landis, J. R., He, J., Feldman, H. I., et al. (2017). Statistical methods for recurrent event analysis in cohort studies of ckd. *Clinical Journal of the American Society of Nephrology*, 12(12):2066–2073.

Yap, B. W. and Sim, C. H. (2011). Comparisons of various types of normality tests. *Journal of Statistical Computation and Simulation*, 81(12):2141–2155.

Yau, K. K. (2001). Multilevel models for survival analysis with random effects. *Biometrics*, 57(1):96–102.

Zhang, B., Tsiatis, A. A., Laber, E. B., and Davidian, M. (2012). A robust method for estimating optimal treatment regimes. *Biometrics*, 68(4):1010–1018.

Zhang, Y., Laber, E. B., Davidian, M., and Tsiatis, A. A. (2018). Estimation of optimal treatment regimes using lists. *Journal of the American Statistical Association*, pages 1–9.

Zhang, Y. and Yang, Y. (2015). Cross-validation for selecting a model selection procedure. *Journal of Econometrics*, 187(1):95–112.

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(499):1106–1118.

Zhao, Y.-Q., Laber, E. B., Ning, Y., Saha, S., and Sands, B. E. (2019). Efficient augmentation and relaxation learning for individualized treatment rules using observational data. *Journal of Machine Learning Research*, 20(48):1–23.

Zhou, X. and Kosorok, M. R. (2017). Augmented outcome-weighted learning for optimal treatment regimes. *arXiv preprint arXiv:1711.10654*.

Zhou, X., Mayer-Hamblett, N., Khan, U., and Kosorok, M. R. (2017). Residual weighted learning for estimating individualized treatment rules. *Journal of the American Statistical Association*, 112(517):169–187.

Zhu, R., Zeng, D., and Kosorok, M. R. (2015). Reinforcement learning trees. *Journal of the American Statistical Association*, 110(512):1770–1784.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320.