Safe and Interpretable Estimation of Optimal Treatment Regimes

Harsh Parikh* Johns Hopkins University

Quinn Lanners* Duke University

Zade Akras Harvard Medical School

Sahar F. Zafar Massachusetts General Hospital

M Brandon Westover Beth Israel Deaconess Medical Center

Cynthia Rudin Duke University

Alexander Volfovsky Duke University

Abstract

Recent advancements in statistical and reinforcement learning methods have contributed to superior patient care strategies. However, these methods face substantial challenges in high-stakes contexts, including missing data, stochasticity, and the need for interpretability and patient safety. Our work operationalizes a safe and interpretable approach for optimizing treatment regimes by matching patients with similar medical and pharmacological profiles. This allows us to construct optimal policies via interpolation. Our comprehensive simulation study demonstrates our method's effectiveness in complex scenarios. We use this approach to study seizure treatment in critically ill patients, ultimately advocating for personalized strategies based on medical history and pharmacological features. Our findings recommend reducing medication doses for mild, brief seizure episodes and adopting aggressive treatment strategies for severe cases, leading to improved outcomes.

1 INTRODUCTION

Our study investigates optimal treatment strategies for critically ill patients suffering from seizures or epileptiform activity (EA). These conditions are associated with elevated in-hospital mortality rates and long-term disabilities (Parikh et al., 2023; Ganesan and Hahn, 2019; Kim et al., 2018). EA is commonly observed in patients with various medical conditions such as brain injuries (Lucke-Wold et al., 2015), cancer (Lee et al., 2013), organ failure (Boggs, 2002), and infections like COVID-19. Healthcare professionals in intensive care units (ICUs) frequently use anti-seizure medications (ASMs) to manage EA. However, there are concerns regarding the utilization of highly potent ASMs due to their potential adverse health effects (Farrokh et al., 2018; De Wit et al., 2016). Additionally, the relative risks and benefits of ASMs vary among patients. This variation necessitates personalized treatment strategies to achieve optimal outcomes for each individual patient, as there is no one solution that fits all.[1]

We analyze data from a large hospital to identify optimal treatment regimes and generate clinically relevant hypotheses for future investigations in critical care. However, our data faces many challenges such as (i) a relatively small dataset of 995 patients, (ii) limited observation windows resulting in unobserved or missing ASM and EA data, and (iii) highly variable brain-drug interactions. No existing optimal treatment regime estimation methods are well-suited to handle these challenges (see Table 2, Section 2, and Appendix A). While our study focuses on treating EA in critically ill patients, the underlying framework is applicable across various medical and healthcare contexts, such as addressing substance use disorder in intravenous drug users (Volkow, 2020), managing coronary heart disease in ICU patients (Guo et al., 2022) or treating chronic psychiatric disorders (Murphy et al., 2007).

Contributions. We offer a general and flexible approach that allows for consistent estimation of optimal treatment regimes in the face of these challenges. Our

Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

[1]Strategies regarding when and how to treat patients based on their recent history are referred to as treatment regimes (denoted by π_i for each patient i).

approach is divided into three main steps:

1. **Pharmacological Feature Estimation:** We estimate patient-specific pharmacological features using a mechanistic model that captures EA-ASM interaction and is motivated by the underlying biochemistry.
2. **Distance Metric Learning:** We employ distance metric learning to identify clinical and pharmacological features affecting the outcome and use it to perform nearest-neighbors estimation to account for confounding factors.
3. **Optimal Regime Estimation:** We estimate the optimal treatment regime for each patient using their matched group. The matched group is comprised of nearby points according to the learned distance metric. The optimal regime is estimated using linear interpolation over the regimes of the nearby patients with favorable outcomes.

Estimation via our approach results in personalized optimal treatment regimes that are:

- ***Interpretable***, allowing caregivers to understand, validate, and implement the regimes easily;
- ***Safe***, ensuring that patients are neither over-prescribed nor under-prescribed ASMs; and
- ***Accurate***, outperforming or performing on par with state-of-the-art black-box methods.

The simplicity and transparency of our approach coupled with its flexibility and interpretability makes it suited for high-stakes scenarios, such as the design of treatment strategies for patients experiencing epileptiform activity (EA) in the ICU. We discuss the identification of optimal treatment regimes in Section 4 and delineate our methodology to estimate them in Section 5. We validate and compare our approach with existing methods via simulation studies in Section 6 and Appendix F.

Clinical Findings. We show in Section 7 that our estimated treatment regimes *would have improved the outcomes for patients in the ICU*. The results indicate that a one-size-fits-all approach to escalating ASM usage in response to EA may not be universally beneficial. Instead, it is crucial to tailor treatment plans for each individual. For instance, patients exhibiting cognitive impairment or dementia are at a heightened risk of experiencing adverse effects from ASMs. A more cautious and lower-intensity approach to treatment may be warranted in such cases. This analysis not only characterizes beneficial approaches for treating EA in critically ill patients but also generates relevant hypotheses for future inquiry.

Methods	CA	VT	MT	LO	DE	IN
Our Method	✓	✓	✓	✓	✓	✓
Finite BI	✗	✗	✗	✓	✓	Δ
Infinite HZ	✗	✓	Δ	✗	✓	Δ
Censored DTR	✗	✓	✓	Δ	✓	Δ
Deep RL	✓	✓	Δ	Δ	Δ	✗
Causal NN	✗	✗	✗	✓	✓	✓

Table 1: Characteristics of optimal regime estimation approaches. *Finite BI*: finite timestep backward induction methods, *Infinite HZ*: infinite horizon methods, *Censored*: censored data methods, *Deep RL*: deep reinforcement learning methods, *Causal NN*: causal nearest neighbors. Columns represent *CA*: continuous action space, *VT*: variable timesteps, *MT*: missing timesteps, *LO*: targets long-term outcomes without requiring a designed reward function, *DE*: data efficiency, and *IN*: interpretability. Green cells denote desired properties and red cells indicate undesired properties in the context of our problem. Δ indicates the attribute depends on underlying modeling choices.

2 RELATED LITERATURE

Our literature survey encompasses various techniques for estimating optimal treatment regimes. We classify these techniques into five categories: (i) Finite Timestep Backward Induction (Murphy, 2003; Robins, 2004; Murphy, 2005; Moodie and Richardson, 2010; Chakraborty et al., 2010; Zhang et al., 2012; Zhao et al., 2015; Murray et al., 2018; Blumlein et al., 2022; Qian and Murphy, 2011; Moodie et al., 2014; Zhang et al., 2018), (ii) Infinite Time Horizon (Ernst et al., 2005; Ertefaie and Strawderman, 2018; Clifton and Laber, 2020), (iii) Censored Data (Goldberg and Kosorok, 2012; Lyu et al., 2023; Zhao et al., 2020), (iv) Deep Reinforcement Learning (Mnih et al., 2013; Lillicrap et al., 2015; Haarnoja et al., 2018; Fujimoto et al., 2018a; Kumar et al., 2020; Fujimoto et al., 2018b; Wang et al., 2020), and (v) Causal Nearest Neighbors (Zhou and Kosorok, 2017).

Each category of techniques has its strengths and limitations. Finite timestep backward induction methods offer interpretability and ease of implementation. However, they struggle with missing states, samples with variable timesteps, and large action spaces. Infinite time horizon and censored data methods can handle more nuanced temporal data but require a predefined reward function. Deep reinforcement learning (RL) can handle more complex regimes but lacks interpretability and requires a large sample size. There is a need for a method that can handle continuous state and action spaces, variable and missing timesteps, does not require the specification of an arbitrary reward function, and can work with a small sample size while

maintaining accuracy and interpretability. We provide a summary of each category of techniques in regard to these attributes in Table 2 and include an in-depth literature survey in Appendix A.

3 PRELIMINARIES

We now introduce our setup and notation. While our study focuses on treating EA in critically ill patients, the underlying framework is applicable across various medical and healthcare contexts, as discussed earlier.

For each patient i in a cohort of n patients, we observe (i) pre-treatment covariates $\mathbf{X}_i$, (ii) time-series of states $\{E_{i,t}\}_{t=1}^{T_i}$ (in this case the EA burden), where T_i is the duration for which the patient is under observation, (iii) sequence of actions $\{\mathbf{Z}_{i,t}\}_{t=1}^{T_i}$ (a vector of ASM drug doses given to the patient), and (iv) discharge outcome Y_i. Here, Y_i is a binary indicator for patient well-being with 1 indicating an adverse outcome based on the modified Rankin Score (mRS). The mRS was retrospectively abstracted from hospital records, specifically physician and physical therapy notes, at the time of patient discharge. The mRS assessments underwent rigorous independent review by evaluators, who were intentionally blinded to the patients' EEG measurements and antiseizure medication status to avoid bias.

The sequence of actions, $\{\mathbf{Z}_{i,t}\}$, are determined based on the administered policy π_i such that $\mathbf{Z}_{i,t} = \pi_i(\{E_{i,t'}\}_{t'=1}^{t}, \{\mathbf{Z}_{i,t'}\}_{t'=1}^{t-1}) + \mathcal{E}_{i,t}$ where $\mathcal{E}_{i,t}$ is the unobserved time-and-patient specific factor affecting the action at time t. $Y_i(\{\mathbf{z}_{i,t}\})$ denotes the potential outcome, under the action sequence $\{\mathbf{z}_{i,t}\}$. However, since $\mathbf{z}_{i,t}$'s are determined by the policy π_a, we redefine the potential outcomes as a function of the policy itself, denoted as $Y_i(\pi_a)$. We assume that the observed outcome Y_i is equal to the potential outcome under the administered treatment regime, denoted $Y_i(\pi_i)$. Note that while we observe $\mathbf{Z}_{i,t}$'s, we do not observe the underlying treatment regime π_i.

Our goal is to identify an optimal regime π^* for each patient i that minimizes their potential outcome:

$$\pi_i^* \in \arg\min_{\pi_a} \mathbb{E}[Y_i(\pi_a)|\mathbf{X}_i].$$

What makes this challenging is that we only observe the potential outcome corresponding to the treatment regime administered by the doctors. Thus, $Y_i = Y_i(\pi_i)$ for each patient while all the other potential outcomes are missing (or unknown). Importantly, the outcome is observed at a timepoint τ_i which may be substantially further down the road than the length of observation for each patient, denoted by T_i.

To address this missingness, we note that the state-action interaction and state transition are determined by underlying pharmacology that can be decoupled into two parts: (i) pharmacokinetics and (ii) pharmacodynamics. Pharmacokinetics describes the changes in drug concentration at time t as a function of the drug concentration at the previous time points along with the current drug dose at time t. Pharmacodynamics describes the changes to the EA burden at time t as a function of the current drug concentration and the past EA burden. The pharmacokinetic-pharmacodynamic (PK/PD) system is formalized as a pair of partial differential equations (described in detail in Appendix C). Since this structural system fully governs the drug-EA interaction, conditioning on it allows us to avoid complex outcome simulators while also providing context for the observed heterogeneity in outcomes.

4 IDENTIFICATION

We now discuss the underlying assumptions that allow identification of $\pi_i^* \in \arg\min_{\pi_a} \mathbb{E}[Y_i(\pi_a) \mid \mathbf{X}_i]$ for each patient i. We start by assuming conditional ignorability (Rubin, 1974; Robins, 2000), $Y_i(\pi_a) \perp \pi_i \mid \mathbf{X}_i$, an assumption standard in observational causal studies. This assumption is reasonable in our setting as we know that caregivers decide the drug regimes primarily based on the pre-treatment features $\mathbf{X}$. By the law of iterated expectations, we know that

$$\mathbb{E}[Y_i(\pi_a) \mid \mathbf{X}_i] = \sum_{\pi} \begin{pmatrix} \mathbb{E}[Y_i(\pi_a) \mid \mathbf{X}_i, \pi_i = \pi] \\ \times P(\pi_i = \pi \mid \mathbf{X}_i) \end{pmatrix}.$$

And by conditional ignorability,

$$\mathbb{E}[Y_i(\pi_a) \mid \mathbf{X}_i, \pi_i = \pi] = \mathbb{E}[Y_i(\pi_a) \mid \mathbf{X}_i, \pi_i = \pi_a]$$

for all π. Thus, if we have positivity, i.e. $P(\pi_i = \pi_a \mid \mathbf{X}_i) > 0$, then $\mathbb{E}[Y_i(\pi_a) \mid \mathbf{X}_i]$ is identifiable as $\mathbb{E}[Y_i \mid \mathbf{X}_i, \pi_i = \pi_a]$.

There are many scenarios similar to our setting where the dimensionality of π is high and experts' treatment choices are based on patients' characteristics. In these scenarios, it is *highly unlikely* that $P(\pi_i = \pi \mid \mathbf{X}_i) > 0$ for all π and $\mathbf{X}_i$. However, recall that we are particularly interested in identifying the optimal treatment regime π_i^* for each patient i and not identifying $\mathbb{E}[Y_i(\pi_a) \mid \mathbf{X}_i]$ for any arbitrary policy π_a. Thus, for our context, it is reasonable to assume that even if the clinicians' policies are suboptimal, they are sampled from the neighborhood of the optimal policy such that $P(\pi_i = \pi_i^* \mid \mathbf{X}_i) = \mathbb{E}_{\pi|\mathbf{X}_i}[P(\pi_i = \pi \mid \mathbf{X}_i)]$. We refer to this assumption as *local* positivity. This assumption is weaker than the standard positivity assumption in causal inference. The major implication

of this assumption is that $P(\pi_i = \pi_i^* \mid \mathbf{X}_i) > 0$, allowing us to identify $\mathbb{E}[Y_i(\pi_i^*) \mid \mathbf{X}_i]$ and subsequently $\pi_i^* = \arg\min_{\pi \text{ s.t. } P(\pi|\mathbf{X}_i)>0} \mathbb{E}[Y_i \mid \mathbf{X}_i, \pi_i = \pi]$.

Under the assumption of local positivity, if π_i were observed for each patient i, π_i^* is always identifiable. However, as noted in Section 3, we only observe $\{E_{i,t}\}_{t=1}^{T_i}$ and $\{\mathbf{Z}_{i,t}\}_{t=1}^{T_i}$ while the underlying π_i is unobserved. Recall that $\mathbf{Z}_{i,t} = \pi_i(\{E_{i,t'}\}_{t'=1}^{t}, \{\mathbf{Z}_{i,t'}\}_{t'=1}^{t-1}) + \mathcal{E}_{i,t}$, where $\mathcal{E}_{i,t}$ is an unobserved patient-and-time specific factor. We make a Markovian assumption, $\pi_i(\{E_{i,t'}\}_{t'=1}^{t}, \{\mathbf{Z}_{i,t'}\}_{t'=1}^{t-1}) = \pi_i(\{E_{i,t'}\}_{t'=t-12h}^{t}, \{\mathbf{Z}_{i,t'}\}_{t'=t-12h}^{t-1})$ and a sequential ignorability assumption such that $\mathcal{E}_{i,t} \perp (\{E_{i,t'}\}_{t'=1}^{t}, \{\mathcal{E}_{i,t'}\}_{t'=1}^{t-1}) \mid \mathbf{X}_i$. Under these assumptions, π_i is non-parametrically identifiable for each patient i (Matzkin, 2007). This, in turn, implies that the optimal treatment regime π_i^* is identifiable.

Remark 1. Recall that the outcome Y_i is a function of a high-dimensional vector of EA burdens $\{E_{i,t}\}_{t=1}^{\tau_i}$ and drug doses $\{\mathbf{Z}_{i,t}\}_{t=1}^{\tau_i}$, some of which are unobserved. Defining the treatment as a regime π_i is akin to exposure mapping such that even though $(\{E_{i,t}\}_{t=1}^{\tau_i}, \{\mathbf{Z}_{i,t}\}_{t=1}^{\tau_i}) \neq (\{E_{j,t}\}_{t=1}^{\tau_j}, \{\mathbf{Z}_{j,t}\}_{t=1}^{\tau_j})$ we have $\mathbb{E}[Y_i(\pi_i)|\mathbf{X}_i = \mathbf{x}] = \mathbb{E}[Y_j(\pi_j)|\mathbf{X}_j = \mathbf{x}]$ if $\pi_i = \pi_j$. This helps us address the problem with missing $E_{i,t}$'s and $\mathbf{Z}_{i,t}$'s and ensures that the local positivity assumption is more reasonable.

5 METHODOLOGY

We now outline our three-stage methodology for estimating the optimal treatment regime. The first stage involves estimating an individualized mechanistic model from observed state-action data to approximate state transition dynamics. Mechanistic modeling offers interpretability and needs much less data for fine-tuning. We also estimate the administered regimes (π_i's) if they are unobserved (as in our setup). In the second stage, we create a distance metric to match patients based on pre-treatment covariates and estimated mechanistic model parameters. Subsequently, we use the estimated distance metric to tightly match patients . Finally, in the third stage, we leverage these matched groups to estimate the optimal treatment regimes.

Our interpretable matching approach allows validation through case-based reasoning, which enhances confidence in the estimation procedure and underlying assumptions. We provide details for each stage in the following subsections, with a focus on our real-world application. However, the framework is adaptable to other applications with similar data structures.

Mechanistic State Transition Modeling. We approximate PK using a one-compartment model (Shargel et al., 1999), with half-life as the parameter, and Hill's PD model (Hill, 1910; Weiss, 1997; Nelson et al., 2008), with receptor-ligand affinity and drug dose for 50% efficacy as parameters, to model the short-term effectiveness of the ASMs in reducing EA burden. We delineate the models formally in Appendix C. For each patient i in the cohort, we estimate these individualized PK/PD parameters by minimizing the mean squared error between the predicted EA time series under the observed ASM regime using the mechanistic model and the actual observed EA time series. This step is akin to estimating a multi-dimensional propensity score.

Remark 2. We approximate state-transition dynamics via deterministic mechanistic models, but we do not use them for counterfactual simulations. Mechanistic modeling isolates clinically relevant pharmacological features from stochastic dynamics. While state-transition dynamics adjustment is not necessary for consistent estimation, accounting for PK/PD parameters aids in estimating heterogeneous effects, akin to using propensity scores with Bayesian regression trees (Hahn et al., 2020).

Characterizing Administered Policies. In our study, we focus on treatment regimes for two commonly used anti-seizure medications (ASMs): propofol and levetiracetam. For our application, we employ the policy template that is defined by the drug administration protocols used in hospitals, to ensure interpretability, although our framework can accommodate non-parametric policy functions such as trees or forests. Propofol, a sedating ASM, is administered as a continuous infusion based on the past 1hr, 6hrs, and 12hrs of seizure levels using policy π^{prop}. In contrast, non-sedative ASM levetiracetam is given as a bolus every 12 hours, with dosages varying according to recent EA burden and drug history through policy π^{lev}. The regime for patient i is denoted by

$$\pi_i = \begin{Bmatrix} \pi_i^{prop}\left(\{E_{i,t'}\}_{t'=1}^{t}, \{\mathbf{Z}_{i,t'}\}_{t'=1}^{t-1}; \mathbf{a}_i^p\right) \\ \pi_i^{lev}\left(\{E_{i,t'}\}_{t'=1}^{t}, \{\mathbf{Z}_{i,t'}\}_{t'=1}^{t-1}; \mathbf{a}_i^l\right) \end{Bmatrix}.$$

We provide the functional forms of the policies in Appendix H. We use the observed EA burdens ($\{E_{i,t}\}$) and ASM doses ($\{\mathbf{Z}_{i,t}\}$) to deduce the administered policy π_i for each patient i by minimizing the mean squared error loss between the predicted and observed drug doses at each time t.

Distance Metric Learning and Matching To adjust for confounding, we need to account for pre-treatment covariates and pharmacological features. We do this by grouping patients who are similar in

these features but are treated differently. This procedure is called matching, a commonly used approach to nonparametrically estimate potential outcomes (Ho et al., 2007; Stuart, 2010; Parikh et al., 2022). For the sake of simplicity, let $\mathbf{V}_i$ denote a vector of pretreatment and pharmacological features for each patient i. Then, the estimate for $\mathbb{E}[Y(\pi_a)|\mathbf{V}=\mathbf{v}]$ is given by $\widehat{Y}_{\mathbf{v}}(\pi_a) = m(MG_d(\mathcal{D}, r, \mathbf{v}), \pi_a)$ where $MG_d(\mathcal{D}, r, \mathbf{v})$ is the matched group of units from dataset $\mathcal{D}$ that are r distance away from $\mathbf{v}$ under distance metric d, and m is a regression on the units in the matched group evaluated at π_a.

In high-dimensional scenarios with limited data, it is not possible to precisely match all covariates. Thus, we want to match tightly on important covariates that affect patients' prognoses. Recent matching approaches have explored distance metric learning before matching for more accurate and interpretable causal effect estimation (Parikh et al., 2022; Diamond and Sekhon, 2013; Lanners et al., 2023, see Appendix B for further details). We extend the Variable Importance Matching (VIM) framework (Lanners et al., 2023) to our problem setting. Our distance metric d is parameterized by a positive semi-definite matrix $\mathcal{M}$ such that $d_{\mathcal{M}}(\mathbf{v}_i, \mathbf{v}_k) = (\mathbf{v}_i - \mathbf{v}_k)^T \mathcal{M} (\mathbf{v}_i - \mathbf{v}_k)$. We constrain $\mathcal{M}$ to a diagonal matrix, enabling domain experts to interpret these entries as feature importance values. Consequently, we set $\mathcal{M}_{j,j}$ equal to the gini impurity importance of the j-th feature in the model for $\mathbb{E}[Y|\mathbf{V}]$ (as defined in Nembrini et al. (2018) and Ishwaran (2015)). To ensure the "honesty" of our approach, we split the dataset $\mathcal{D}$ into two parts: the training set $\mathcal{D}_{tr}$ and the estimation set $\mathcal{D}_{est}$ (Ratkovic, 2019). We fit gradient-boosting trees with 100 estimators on $\mathcal{D}_{tr}$, each with a maximum depth of 2. Henceforth, we denote the learned distance metric as $d^\dagger$.

Estimating Optimal Regimes. For each matched group centered around patient $i \in \mathcal{D}_{est}$, we consider the administered regimes π_k and outcomes Y_k for all $k \in MG_{d^\dagger}(\mathcal{D}_{est}, r, \mathbf{V}_i)$, where $d^\dagger$ is the learned distance metric. For the sake for simplicity, we will denote $MG_{d^\dagger}(\mathcal{D}_{est}, r, \mathbf{V}_i)$ as MG_i. We estimate the conditional expected outcome $\nu_i(\pi) := \mathbb{E}[Y_i \mid \pi, \mathbf{V}_i]$ using only the units in MG_i. The estimate is denoted as $\widehat{\nu}_i(\pi)$. Further, consider a *new* operator $\bigoplus$ such that if $\pi_1 \in \text{Dom}(\pi)$ (a function that maps states to a vector of ASM doses) and $\pi_2 \in \text{Dom}(\pi)$ (another function that maps states to a vector of ASM doses) then $\pi_3 = \pi_1 \bigoplus \pi_2 \in \text{Dom}(\pi)$. This operation is defined so that if $\pi_3 = \pi_1 \bigoplus \pi_2$ then $\pi_3(s) := \pi_1(s) + \pi_2(s)$ for all s in the domain of states. Then, our estimate of the optimal treatment regime for unit i is $\widehat{\pi}_i^* \in \arg\min_{\pi_{c,i}} \widehat{\nu}_i(\pi_{c,i})$ where, $\pi_{c,i} = \bigoplus_{k \in MG_i} c_k \pi_k$, $\sum_{k \in MG_i} c_k = 1$ and $0 \le c_k \le 1$.

Consistency. We now discuss a smoothness of outcomes assumption under which our estimated optimal regime is consistent. Let's first define an $(\mathcal{S}, p)$-norm on the space of policies such that $\|\pi_1 - \pi_2\|_{\mathcal{S},p} = \left(\int_{s \in \mathcal{S}} |\pi_1(s) - \pi_2(s)|^p\right)^{1/p} ds$ where $\mathcal{S}$ is the state space for the policies and p is some positive integer. The smoothness of outcomes assumption is given as follows: given constants $\lambda_\pi \ge 0$ and $\lambda_{\mathbf{V}} \ge 0$ such that for any two units 1 and 2, if $\|\pi_1 - \pi_2\|_{\mathcal{S},\infty} \le \lambda_\pi$ and $\|\mathbf{V}_1 - \mathbf{V}_2\|_2 \le \lambda_{\mathbf{V}}$ then $\|\mathbb{E}[Y(\pi_1) \mid \mathbf{V}_1] - \mathbb{E}[Y(\pi_2) \mid \mathbf{V}_2]\| \le \delta(\lambda_\pi, \lambda_{\mathbf{V}})$ where δ is a monotonically decreasing function in both the arguments with $\delta(0,0) = 0$.

This assumption essentially implies that if $\mathbf{V}_1$ and $\mathbf{V}_2$ are close and if π_1 and π_2 are also close then the expected potential outcomes are also close.

Proposition 1. *Given the conditional ignorability, local positivity, and smoothness of outcomes assumptions, $\widehat{\pi}_i^*$ is a consistent estimate of π_i^*, such that*

$$\lim_{n \to \infty} \mathbb{E}[Y(\widehat{\pi}_i^*) \mid \mathbf{V}_i] \to \mathbb{E}[Y(\pi_i^*) \mid \mathbf{V}_i].$$

We provide the proof of this proposition in Appendix I

Remark 3. As our regimes π are linear score functions with parameter vector $\mathbf{a}$ (see Appendix H), $\pi_{k_3} = \pi_{k_1} \bigoplus \pi_{k_2}$ corresponds to defining new policy π_{k_3} with parameters $\mathbf{a}_{k_3} = \mathbf{a}_{k_1} + \mathbf{a}_{k_2}$. This property comes in handy when comparing the administered policy's parameters with the estimated optimal policy.

6 SYNTHETIC EXPERIMENTS

Comparison Baselines. We compare our approach to 49 approaches based on 10 different state-of-the-art finite timestep backward induction, infinite horizon, and deep reinforcement learning frameworks. The vast majority of methods cannot be run on our data setup out of the box and often require major modifications. The various approaches we compare use different underlying models, ways to discretize continuous outcomes, and predefined reward functions. We outline the methods we compare to and the implementation details in Appendix E.

Data Generation Procedure. Our data-generative procedure is designed to emulate the real-world scenario where critically ill patients undergo drug treatment that affects their state. We design the data generation process to be customizable in five important aspects to discern how various methods perform with the challenges present in our real-world data: (i) number of covariates; (ii) number of total timesteps τ_i, for each patient; (iii)

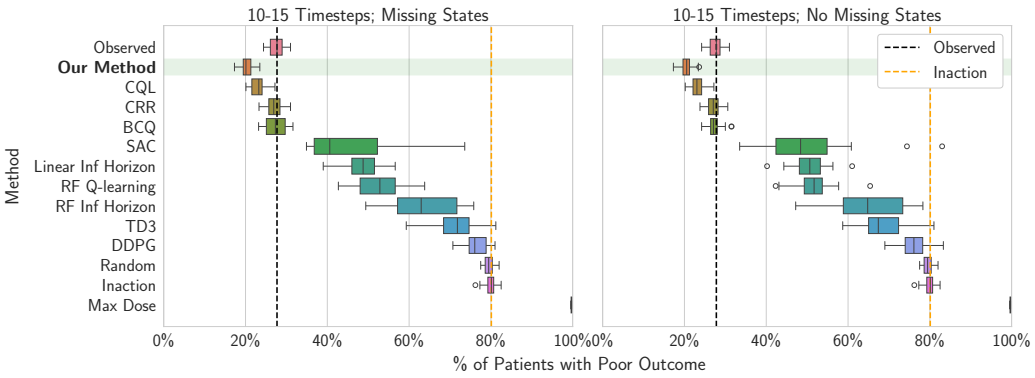
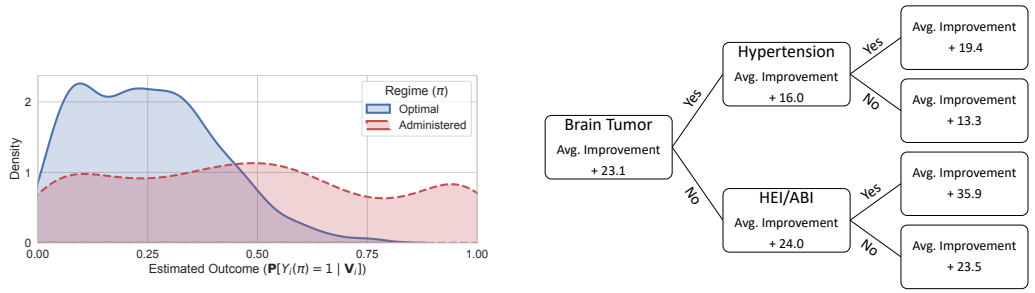


Figure 1: Percent of patients with poor outcomes under each method's proposed policy (*lower is better*). Boxplots show the distribution of the average outcomes over 20 iterations. *Observed* shows average observed outcomes. *Inaction* and *Max Dosing* administer no drugs and the max amount of drugs to each patient at each timestep, respectively. *RF Q-learning* is a finite timestep backward induction method using random forests. *Infinite (Inf) Horizon* methods use fitted Q-iteration (see Clifton and Laber, 2020) with either linear models or random forests. *Q-learning* and *Inf Horizon* discretize the treatment into five bins. *BCQ, CQL, CRR, GGPQ, SAC*, and *TD3* are Deep RL methods. *Inf Horizon* and Deep RL methods use an insightful reward function, see Appendix E.



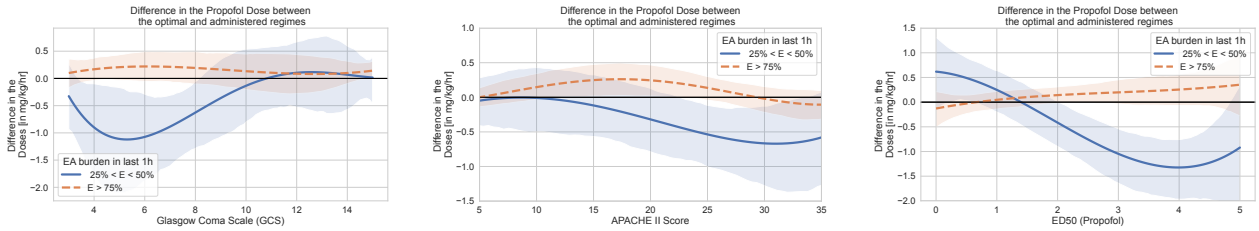
(a) (b)

Figure 2: (a) Estimated density of the outcome probabilities under optimal and clinician's administered policies. (b) Tree characterizing the subpopulations that would have benefited the most by switching to the optimal policy. The value at each node in the tree shows the percentage point *improvement* in the outcome. Here, HEI/ABI refers to hypoxic-ischemic encephalopathy (HIE) and anoxic brain injury (ABI).

number of unobserved timesteps, $\tau_i - T_i$, for each patient; (iv) cardinality of the action space; and (v) observed policies. We construct a total of 32 different experimental setups by varying these aspects. We provide the full details of our data generation process and experimental setups in Appendix D.

Results. For a real-world data simulation, we use 1000 simulated "patients" with (i) 100 pre-treatment covariates, (ii) varying lengths of stay (10-15 timesteps), and (iii) unobserved timesteps (2-5 steps), where (iv) drug doses at each timestep are between 0 to 100 and (v) determined using an educated policy akin to one doctors use in the ICU. We display the percent of patients with poor outcomes under the proposed policies of our method, representative approaches from each of our comparison baseline categories, and predetermined approaches like inaction, random assignment, and max-dose in the left plot of Figure 1. The right plot of Figure 1 shows results for the same setting except with (iii) no missing timesteps. In each of these complex setups, our matching-based method consistently yields optimal treatment policies, surpassing all comparison methods. Notably, among the 8 setups with 10-15 total timesteps and observed data generated from an educated policy, our method is consistently the top performer.

Analysis. Existing methods falter on simulated data emulating our real-world setup for various reasons. The suboptimal performance of Q-learning is likely caused by its inability to handle missing states as well as continuous action spaces (Huang et al., 2022). Infinite horizon methods like fitted Q-iteration mainly rely on a predefined reward function, often focusing on short-term objectives, and cannot handle contin-



(a) (b) (c)

Figure 3: Difference in the propofol drug doses between the optimal and the administered regimes for mild and severe EA burden in last 1h for (a) patients on various levels of Glasgow coma scale (GCS); (b) patients with various levels of APACHE II scores; and (c) patients with various levels of ED50 for propofol, an important pharmacodynamic parameter determining the amount of drug required to reduce EA burden by 50%.

uous action spaces (Clifton and Laber, 2020). Deep RL methods like DDPG are also likely struggling with having to rely on a predefined reward function and the relatively small dataset size (Riachi et al., 2021; Kondrup et al., 2023; Kang et al., 2023; Kalweit and Boedecker, 2017). More modern Deep RL methods like CQL, CRR, and BCQ mediate the deficiencies of DDPG. However, unlike our approach, these methods are inherently uninterpretable and, therefore, are unsuitable for high-stakes problems.

Other methods can perform as good or better than our method when aspects of the data generating process are varied to look less like our real-world data. For example, approaches like infinite horizon and some Deep RL methods perform better when the observed data is generated from a random, rather than an educated, policy and backward induction methods perform better when there are fewer and no unobserved timesteps.

In Appendix F, we thoroughly compare our method to the 49 baselines using 32 simulation setups. These results underscore the suboptimal performance of existing methods in scenarios with missing data, continuous action space, and highly stochastic state dynamics. Our method can handle these various challenges, allowing it to accurately estimate interpretable optimal regimes that are safe for high-stakes settings.

7 TREATING SEIZURES IN CRITICALLY ILL PATIENTS

We now present the analysis and insights derived from our optimal treatment estimation approach when applied to a cohort of 995 critically ill patients. This cohort is comprised of individuals aged 18 and older with confirmed electrographic EA as diagnosed by clinical neurophysiologists or epileptologists.

We evaluate our approach by comparing the estimated optimal treatment policy $P(Y_i(\pi_i^*) = 1|\mathbf{V}_i)$ with the clinician's administered policy $P(Y_i(\pi_i) = 1|\mathbf{V}_i)$ for each patient. Our analysis indicates a significant improvement in patient outcomes, with a 23.6 ± 1.9 percentage point reduction in the probability of adverse events under the optimal regimen. Few patients under the optimal policy had over a 50% chance of an adverse outcome (Figure 2(a)). Figure 2(b) reveals that patients with hypoxic-ischemic encephalopathy (HIE) or anoxic brain injury (ABI) experienced a substantial 35.9 percentage point decrease in the likelihood of an adverse outcome, highlighting those who benefited most from our estimated optimal treatment policies.

Table 2: APACHE II scores and corresponding non-operative mortality or death rate from Knaus et al. (1986), as well as estimated Y under estimated administered regime and optimal regime.

APACHE II Score	Death Rate	Est. $\mathbb{E}[Y_i(\pi_i)]$	Est. $\mathbb{E}[Y_i(\widehat{\pi}_i^*)]$
0 to 4	4%	17%	6%
5 to 9	8%	22%	8%
10 to 14	15%	35%	17%
15 to 19	24%	48%	25%
20 to 24	40%	56%	31%
25 to 29	55%	61%	35%
30 to 34	73%	73%	36%

We compare and contrast the optimal regimes with the administered regimes for each drug. We consider the variability of each drug's regime with respect to patients' pre-treatment prognosis measured as APACHE II score (Knaus et al., 1986) and Glasgow coma scale (GCS) (Jain and Iverson, 2018). APACHE II score quantifies disease severity in ICU patients and GCS measures impaired consciousness in acute medical and trauma patients. Both of these measures are clinically relevant for deciding treatment strategies (Mumtaz et al., 2023). Table 2 displays mortality rates from Knaus et al. (1986) and estimated Y under administered and optimal regimes for different APACHE II

scores. The optimal regime improves outcomes across all levels, with the most benefits seen in patients with high APACHE II scores (i.e., with worse prognoses).

Propofol Regimes. Figures 3(a) and 3(b) show that, on average, the estimated optimal propofol dose for individuals with low EA burden is generally lower than the administered dose, especially for those with worse prognoses (lower GCS or higher APACHE II scores). Conversely, when patients have a severe EA burden in the last hour and an APACHE II score below 30, the optimal dose is marginally higher than the administered dose. Also, one must adjust propofol dosages based on patients' PK/PD, specifically, based on the ED50 values – a PD parameter quantifying the amount of drug required to reduce the EA burden by 50%. When the EA burden is low, we recommend increasing the dosage for patients with low ED50 values to alleviate EA and decreasing it for those with high ED50 values, as an excess of propofol may lead to adverse effects (see Figure 3(c)).

Levetiracetam Regimes. The optimal and administered levetiracetam regimes generally align, except for patients with sustained 12-hour EA burden. In such cases, the optimal regime recommends a lower dose (0.50 mg/kg on average) compared to the administered regime (0.82 mg/kg on average). For dementia patients, the difference is more pronounced, with the optimal regime suggesting a dose of 4.2 mg/kg lower (see Figure 4(a)). Conversely, subarachnoid hemorrhage patients with a 6-hour sustained EA burden receive a 1 mg/kg higher dose with the optimal regime (see Figure 4(b)).

To summarize, our findings indicate that patients in this study would, on average, be less likely to have an adverse outcome under the optimal regimes estimated by using our method. These optimal regimes would lead us to advocate for an assertive approach to managing the high EA burden in more critically ill patients while reducing propofol and levetiracetam dosages for relatively healthier patients or those with mild EA.

8 DISCUSSION & CONCLUSION

We present an approach that is capable of handling many challenges with real-world observational data like variable timesteps, missing states, a continuous action space, and small data size. Our approach balances accuracy and interpretability and demonstrates superior performance through simulation. We ultimately operationalize our approach to learn treatment regimes for ICU patients with EA, showcasing its ability to solve real-world problems.

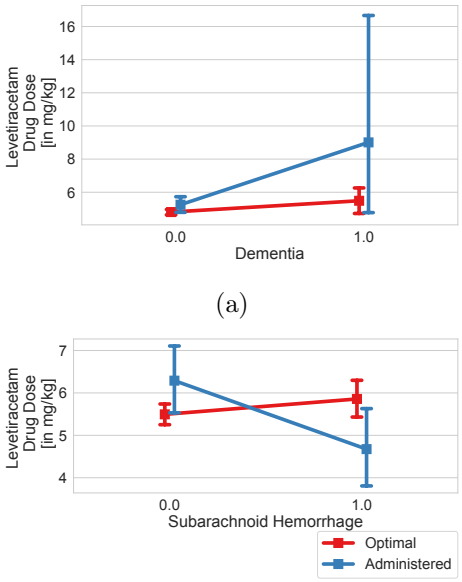


Figure 4: Difference in the levetiracetam doses between the optimal and the administered regimes for (a) patients with and without dementia experiencing a sustained EA burden for 12 hours; and (b) patients with and without subarachnoid hemorrhage experiencing a sustained EA burden for 6 hours.

Clinical Relevance. The current absence of evidence-based guidelines to inform ASM regimes (drug type and dosing) in patients with EA results in frequent overprescription of ASMs in response to EA (Zafar et al., 2020; Rubinos et al., 2018). High EA-burden is frequently treated with escalating doses of ASMs and anesthetics, and many of these patients are also discharged on ASM treatment (Zafar et al., 2018; Tabaeizadeh et al., 2020; Dhakar et al., 2022; Alvarez et al., 2017; Kilbride et al., 2009; Punia et al., 2020). Our findings suggest that not all patients may benefit from such ASM escalation. Thus, careful consideration of the baseline illness severity, injury type, and patient comorbidities is important to determine the risk-benefit trade-off of initiating treatment and selecting treatment intensity. For example, patients with cognitive impairment and dementia have a higher risk of ASM adverse effects (Mendez and Lim, 2003; Cretin, 2021) and may require lower-intensity treatment, which is supported by our findings. Finally, as shown in Figure 3 and Figure 4, heterogeneous treatment responses need to be considered in selecting drug dosing. Current clinical practice relies on population-level pharmacological data to infer standardized dosing regimens used for all patients. However, this one-size-fits-all approach is suboptimal due to the patient-

level PK/PD heterogeneity shown in our study (see Figure 3(c)). Our findings strongly support the need for *clinical trials* to reveal heterogeneous causal effects and construct individualized optimal treatment. Such efforts can guide evidence-based clinical practice and improve patient care in the ICU.

Limitations. Like all causal research, our study relies on untestable assumptions. We assume there are no hidden variables affecting both EA burden and patient discharge outcomes, though unmeasured disease characteristics might violate this assumption. Additionally, the misspecification of our predefined policy template, intended for doctor interpretability, could affect real-world drug administration, akin to issues discussed in recent work Sävje (2023). Furthermore, while we focus on point estimation for personalized optimal treatment regimes, handling uncertainty, especially when estimating the exposure map from observed data, remains an open question.

Future Direction. Addressing the limitations inherent in our approach, we identify three promising areas for future work. First, there is a need for research into uncertainty quantification for estimated personalized optimal treatment regimes, with broader implications for situations where exposure mapping is data-driven. Second, developing a non-parametric approach for sensitivity analysis and partial identification has the potential to advance research in this area. Third, interpretable Deep RL has been explored to a limited capacity in works like Lyu et al. (2019) and Li et al. (2023). Given the relatively strong performance of these methods, future work that optimizes Deep RL for offline and off-policy tasks is a promising direction for future work in this area.

Acknowledgments

We acknowledge funding from the National Science Foundation and Amazon under grant NSF IIS-2147061, and the National Institute on Drug Abuse under grant DA056407. National Institute on Drug Abuse R01DA056407 supports Harsh Parikh. NSF grant DMS-2046880 supports Cynthia Rudin, Alexander Volfovsky, and Quinn Lanners. Alexander Volfovsky is also supported by a National Science Foundation Faculty Early Career Development Award (CAREER: Design and analysis of experiments for complex social processes). The authors want to thank Dr. Lina Montoya and Srikar Katta for their insightful and constructive comments.

References

Alvarez, V., Ruiz, A. A. R., LaRoche, S., Hirsch, L. J., Parres, C., Voinescu, P. E., Fernandez, A., Petroff, O. A., Rampal, N., Haider, H. A., et al. (2017). The use and yield of continuous eeg in critically ill patients: a comparative study of three centers. *Clinical Neurophysiology*, 128(4):570–578.

Arora, S. and Doshi, P. (2021). A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297:103500.

Blumlein, T., Persson, J., and Feuerriegel, S. (2022). Learning optimal dynamic treatment regimes using causal tree methods in medicine. In *Machine Learning for Healthcare Conference*, pages 146–171. PMLR.

Boggs, J. (2002). Seizures and organ failure. In *Seizures*, pages 71–83. Springer.

Chakraborty, B., Murphy, S., and Strecher, V. (2010). Inference for non-regular parameters in optimal dynamic treatment regimes. *Statistical methods in medical research*, 19(3):317–343.

Clifton, J. and Laber, E. (2020). Q-learning: Theory and applications. *Annual Review of Statistics and Its Application*, 7:279–301.

Cretin, B. (2021). Treatment of seizures in older patients with dementia. *Drugs & Aging*, 38(3):181–192.

De Wit, F., Van Vliet, A., De Wilde, R., Jansen, J., Vuyk, J., Aarts, L., De Jonge, E., Veelo, D., and Geerts, B. (2016). The effect of propofol on haemodynamics: cardiac output, venous return, mean systemic filling pressure, and vascular resistances. *British Journal of Anaesthesia*, 116(6):784–789.

Devroye, L., Gyorfi, L., Krzyzak, A., and Lugosi, G. (1994). On the strong universal consistency of nearest neighbor regression function estimates. *The Annals of Statistics*, 22(3):1371–1385.

Dhakar, M. B., Sheikh, Z., Kumari, P., Lawson, E. C., Jeanneret, V., Desai, D., Ruiz, A. R., and Haider, H. A. (2022). Epileptiform abnormalities in acute ischemic stroke: impact on clinical management and outcomes. *Journal of Clinical Neurophysiology*, 39(6):446–452.

Diamond, A. and Sekhon, J. S. (2013). Genetic matching for estimating causal effects: A general multivariate matching method for achieving balance in observational studies. *Review of Economics and Statistics*, 95(3):932–945.

Einmahl, U. and Mason, D. M. (2005). Uniform in bandwidth consistency of kernel-type function estimators.

Ernst, D., Geurts, P., and Wehenkel, L. (2005). Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6.

Ertefaie, A. and Strawderman, R. L. (2018). Constructing dynamic treatment regimes over indefinite time horizons. *Biometrika*, 105(4):963–977.

Farrokh, S., Tahsili-Fahadan, P., Ritzl, E. K., Lewin, J. J., and Mirski, M. A. (2018). Antiepileptic drugs in critically ill patients. *Critical Care*, 22(1):1–12.

Ferraty, F., Laksaci, A., Tadj, A., and Vieu, P. (2010). Rate of uniform consistency for nonparametric estimates with functional variables. *Journal of Statistical planning and inference*, 140(2):335–352.

Fujimoto, S., Hoof, H., and Meger, D. (2018a). Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR.

Fujimoto, S., Meger, D., and Precup, D. (2018b). Off-policy deep reinforcement learning without exploration. corr abs/1812.02900 (2018). *arXiv preprint arXiv:1812.02900*.

Ganesan, S. L. and Hahn, C. D. (2019). Electrographic seizure burden and outcomes following pediatric status epilepticus. *Epilepsy & Behavior*, 101:106409.

Goldberg, Y. and Kosorok, M. R. (2012). Q-learning with censored data. *Annals of statistics*, 40(1):529.

Guo, H., Li, J., Liu, H., and He, J. (2022). Learning dynamic treatment strategies for coronary heart diseases by artificial intelligence: real-world data-driven study. *BMC Medical Informatics and Decision Making*, 22(1):1–16.

Haarnoja, T., Zhou, A., Hartikainen, K., Tucker, G., Ha, S., Tan, J., Kumar, V., Zhu, H., Gupta, A., Abbeel, P., et al. (2018). Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*.

Hahn, P. R., Murray, J. S., and Carvalho, C. M. (2020). Bayesian regression tree models for causal inference: Regularization, confounding, and heterogeneous effects (with discussion). *Bayesian Analysis*, 15(3):965–1056.

Hill, A. V. (1910). The possible effects of the aggregation of the molecules of hemoglobin on its dissociation curves. *j. physiol.*, 40:iv–vii.

Ho, D. E., Imai, K., King, G., and Stuart, E. A. (2007). Matching as nonparametric preprocessing for reducing model dependence in parametric causal inference. *Political analysis*, 15(3):199–236.

Holloway, S., Laber, E., Linn, K., Zhang, B., Davidian, M., and Tsiatis, A. (2020). Dyntxregime: Methods for estimating optimal dynamic treatment regimes. *R package version*, 49:3.

Huang, Y., Cao, R., and Rahmani, A. (2022). Reinforcement learning for sepsis treatment: A continuous action space solution. In *Machine Learning for Healthcare Conference*, pages 631–647. PMLR.

Ishwaran, H. (2015). The effect of splitting on random forests. *Machine learning*, 99:75–118.

Jain, S. and Iverson, L. M. (2018). Glasgow coma scale.

Jiang, H. (2019). Non-asymptotic uniform rates of consistency for k-nn regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3999–4006.

Kalweit, G. and Boedecker, J. (2017). Uncertainty-driven imagination for continuous deep reinforcement learning. In *Conference on Robot Learning*, pages 195–206. PMLR.

Kang, Y., Shi, D., Liu, J., He, L., and Wang, D. (2023). Beyond reward: Offline preference-guided policy optimization. *arXiv preprint arXiv:2305.16217*.

Kara, L.-Z., Laksaci, A., Rachdi, M., and Vieu, P. (2017). Data-driven knn estimation in nonparametric functional data analysis. *Journal of Multivariate Analysis*, 153:176–188.

Kilbride, R. D., Costello, D. J., and Chiappa, K. H. (2009). How seizure detection by continuous electroencephalographic monitoring affects the prescribing of antiepileptic medications. *Archives of Neurology*, 66(6):723–728.

Kim, J. A., Boyle, E. J., Wu, A. C., Cole, A. J., Staley, K. J., Zafar, S., Cash, S. S., and Westover, M. B. (2018). Epileptiform activity in traumatic brain injury predicts post-traumatic epilepsy. *Annals of Neurology*, 83(4):858–862.

Knaus, W. A., Draper, E. A., Wagner, D. P., and Zimmerman, J. E. (1986). Apache ii-a severity of disease classification system: Reply. *Critical Care Medicine*, 14(8):755.

Koenig, S. and Simmons, R. G. (1996). The effect of representation and knowledge on goal-directed exploration with reinforcement-learning algorithms. *Machine Learning*, 22:227–250.

Kondrup, F., Jiralerspong, T., Lau, E., de Lara, N., Shkrob, J., Tran, M. D., Precup, D., and Basu, S. (2023). Towards safe mechanical ventilation treatment using deep offline reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 15696–15702.

Kudraszow, N. L. and Vieu, P. (2013). Uniform consistency of knn regressors for functional variables. *Statistics & Probability Letters*, 83(8):1863–1870.

Kumar, A., Zhou, A., Tucker, G., and Levine, S. (2020). Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 33:1179–1191.

Lanners, Q., Parikh, H., Volfovsky, A., Rudin, C., and Page, D. (2023). Variable importance matching for

causal inference. In *Uncertainty in Artificial Intelligence*, pages 1174–1184. PMLR.

Lee, M. H., Kong, D.-S., Seol, H. J., Nam, D.-H., and Lee, J.-I. (2013). Risk of seizure and its clinical implication in the patients with cerebral metastasis from lung cancer. *Acta neurochirurgica*, 155(10):1833–1837.

Li, K.-C. (1984). Consistency for cross-validated nearest neighbor estimates in nonparametric regression. *The Annals of Statistics*, pages 230–240.

Li, X., Lei, H., Zhang, L., and Wang, M. (2023). Differentiable logic policy for interpretable deep reinforcement learning: A study from an optimization perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Lucke-Wold, B. P., Nguyen, L., Turner, R. C., Logsdon, A. F., Chen, Y.-W., Smith, K. E., Huber, J. D., Matsumoto, R., Rosen, C. L., Tucker, E. S., et al. (2015). Traumatic brain injury and epilepsy: underlying mechanisms leading to seizure. *Seizure*, 33:13–23.

Luo, Y., Kay, J., Grefenstette, E., and Deisenroth, M. P. (2023). Finetuning from offline reinforcement learning: Challenges, trade-offs and practical solutions. *arXiv preprint arXiv:2303.17396*.

Lyu, D., Yang, F., Liu, B., and Gustafson, S. (2019). Sdrl: interpretable and data-efficient deep reinforcement learning leveraging symbolic planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2970–2977.

Lyu, L., Cheng, Y., and Wahed, A. S. (2023). Imputation-based q-learning for optimizing dynamic treatment regimes with right-censored survival outcome. *Biometrics*.

Mataric, M. J. (1994). Reward functions for accelerated learning. In *Machine learning proceedings 1994*, pages 181–189. Elsevier.

Matzkin, R. L. (2007). Nonparametric identification. *Handbook of econometrics*, 6:5307–5368.

Mendez, M. F. and Lim, G. T. (2003). Seizures in elderly patients with dementia: epidemiology and management. *Drugs & aging*, 20:791–803.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Moodie, E. E., Chakraborty, B., and Kramer, M. S. (2012). Q-learning for estimating optimal dynamic treatment rules from observational data. *Canadian Journal of Statistics*, 40(4):629–645.

Moodie, E. E., Dean, N., and Sun, Y. R. (2014). Q-learning: Flexible learning about useful utilities. *Statistics in Biosciences*, 6:223–243.

Moodie, E. E. and Richardson, T. S. (2010). Estimating optimal dynamic regimes: Correcting bias under the null. *Scandinavian Journal of Statistics*, 37(1):126–146.

Mumtaz, H., Ejaz, M. K., Tayyab, M., Vohra, L. I., Sapkota, S., Hasan, M., and Saqib, M. (2023). Apache scoring as an indicator of mortality rate in icu patients: a cohort study. *Annals of Medicine and Surgery*, 85(3):416.

Murphy, S. A. (2003). Optimal dynamic treatment regimes. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 65(2):331–355.

Murphy, S. A. (2005). A generalization error for q-learning.

Murphy, S. A., Oslin, D. W., Rush, A. J., and Zhu, J. (2007). Methodological challenges in constructing effective treatment sequences for chronic psychiatric disorders. *Neuropsychopharmacology*, 32(2):257–262.

Murray, T. A., Yuan, Y., and Thall, P. F. (2018). A bayesian machine learning approach for optimizing dynamic treatment regimes. *Journal of the American Statistical Association*, 113(523):1255–1267.

Nelson, D. L., Lehninger, A. L., and Cox, M. M. (2008). *Lehninger principles of biochemistry*. Macmillan.

Nembrini, S., König, I. R., and Wright, M. N. (2018). The revival of the gini importance? *Bioinformatics*, 34(21):3711–3718.

Ng, A. Y., Russell, S., et al. (2000). Algorithms for inverse reinforcement learning. In *Icml*, volume 1, page 2.

Parikh, H., Hoffman, K., Sun, H., Zafar, S. F., Ge, W., Jing, J., Liu, L., Sun, J., Struck, A., Volfovsky, A., et al. (2023). Effects of epileptiform activity on discharge outcome in critically ill patients in the usa: a retrospective cross-sectional study. *The Lancet Digital Health*.

Parikh, H., Rudin, C., and Volfovsky, A. (2022). Malts: Matching after learning to stretch. *The Journal of Machine Learning Research*, 23(1):10952–10993.

Punia, V., Chandan, P., Fesler, J., Newey, C. R., and Hantus, S. (2020). Post-acute symptomatic seizure (pass) clinic: a continuity of care model for patients impacted by continuous eeg monitoring. *Epilepsia Open*, 5(2):255–262.

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

Ratkovic, M. T. (2019). Rehabilitating the regression : Honest and valid causal inference through machine learning.

Riachi, E., Mamdani, M., Fralick, M., and Rudzicz, F. (2021). Challenges for reinforcement learning in healthcare. *arXiv preprint arXiv:2103.05612*.

Robins, J. M. (2000). Robust estimation in sequentially ignorable missing data and causal inference models. In *Proceedings of the American Statistical Association*, volume 1999, pages 6–10. Indianapolis, IN.

Robins, J. M. (2004). Optimal structural nested models for optimal sequential decisions. In *Proceedings of the Second Seattle Symposium in Biostatistics: analysis of correlated data*, pages 189–326. Springer.

Rubin, D. B. (1974). Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5):688.

Rubinos, C., Reynolds, A. S., and Claassen, J. (2018). The ictal–interictal continuum: to treat or not to treat (and how)? *Neurocritical care*, 29:3–8.

Sävje, F. (2023). Causal inference with misspecified exposure mappings: separating definitions and assumptions. *Biometrika*, page asad019.

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.

Seno, T. and Imai, M. (2022). d3rlpy: An offline deep reinforcement learning library. *Journal of Machine Learning Research*, 23(315):1–20.

Shargel, L., Andrew, B., and Wu-Pong, S. (1999). *Applied biopharmaceutics & pharmacokinetics*, volume 264. Appleton & Lange Stamford.

Singh, S., Lewis, R. L., Barto, A. G., and Sorg, J. (2010). Intrinsically motivated reinforcement learning: An evolutionary perspective. *IEEE Transactions on Autonomous Mental Development*, 2(2):70–82.

Song, R., Wang, W., Zeng, D., and Kosorok, M. R. (2015). Penalized q-learning for dynamic treatment regimens. *Statistica Sinica*, 25(3):901.

Stuart, E. A. (2010). Matching methods for causal inference: A review and a look forward. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 25(1):1.

Tabaeizadeh, M., Aboul Nour, H., Shoukat, M., Sun, H., Jin, J., Javed, F., Kassa, S., Edhi, M., Bordbar, E., Gallagher, J., et al. (2020). Burden of epileptiform activity predicts discharge neurologic outcomes in severe acute ischemic stroke. *Neurocritical care*, 32:697–706.

Tschantz, A., Baltieri, M., Seth, A., and Buckley, C. (2019). Scaling active inference. arxiv. *arXiv preprint arXiv:1911.10601*.

Volkow, N. D. (2020). Personalizing the treatment of substance use disorders. *American Journal of Psychiatry*, 177(2):113–116.

Wang, Z., Novikov, A., Zolna, K., Merel, J. S., Springenberg, J. T., Reed, S. E., Shahriari, B., Siegel, N., Gulcehre, C., Heess, N., et al. (2020). Critic regularized regression. *Advances in Neural Information Processing Systems*, 33:7768–7778.

Weiss, J. N. (1997). The hill equation revisited: uses and misuses. *The FASEB Journal*, 11(11):835–841.

Zafar, S. F., Postma, E. N., Biswal, S., Boyle, E. J., Bechek, S., O'Connor, K., Shenoy, A., Kim, J., Shafi, M. S., Patel, A. B., et al. (2018). Effect of epileptiform abnormality burden on neurologic outcome and antiepileptic drug management after subarachnoid hemorrhage. *Clinical Neurophysiology*, 129(11):2219–2227.

Zafar, S. F., Subramaniam, T., Osman, G., Herlopian, A., and Struck, A. F. (2020). Electrographic seizures and ictal–interictal continuum (iic) patterns in critically ill patients. *Epilepsy & Behavior*, 106:107037.

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

Zhang, Y., Laber, E. B., Davidian, M., and Tsiatis, A. A. (2018). Interpretable dynamic treatment regimes. *Journal of the American Statistical Association*, 113(524):1541–1549.

Zhao, Y.-Q., Zeng, D., Laber, E. B., and Kosorok, M. R. (2015). New statistical learning methods for estimating optimal dynamic treatment regimes. *Journal of the American Statistical Association*, 110(510):583–598.

Zhao, Y.-Q., Zhu, R., Chen, G., and Zheng, Y. (2020). Constructing dynamic treatment regimes with shared parameters for censored data. *Statistics in medicine*, 39(9):1250–1263.

Zhou, X. and Kosorok, M. R. (2017). Causal nearest neighbor rules for optimal treatment regimes. *arXiv preprint arXiv:1711.08451*.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]/No/Not Applicable]. Section 3 and Section 4 discuss the mathematical settings and the relevant assumptions. Section 5 discusses our methodology.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes/No/Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]/No/Not Applicable]. GitHub Link (https://github.com/almost-matching-exactly/opt_tx_regime_matching)
2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]/No/Not Applicable]. Section 3, Section 4 and Section 5 state the relevant assumptions for our Proposition 1.
   (b) Complete proofs of all theoretical results. [Yes]/No/Not Applicable]. Appendix I provides the proof of the proposition.
   (c) Clear explanations of any assumptions. [Yes]/No/Not Applicable]. Section 3, Section 4 and Section 5 discuss the relevant assumptions for our Proposition 1.
3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]/No/Not Applicable]. Click here to access our GitHub containing the code. The patients' data will be provided on appropriate request to Dr M Brandon Westover, and Dr Sahar F Zafar.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]/No/Not Applicable] Appendix E and Appendix F.2 discuss training details. Code also includes comments.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]/No/Not Applicable] Contents of each plot is described in the corresponding caption. Further discussion of experimental details is in Appendix E.
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]/No/Not Applicable] Discussed in Appendix F.2
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]/No/Not Applicable] Citations to existing assets are in Appendix E.
   (b) The license information of the assets, if applicable. [Yes/No/Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes/No/Not Applicable]
   (d) Information about consent from data providers/curators. [Yes]/No/Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Yes]/No/Not Applicable]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Yes/No/Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Yes]/No/Not Applicable] We will add the IRB approvals from involved institution with the camera-ready version (post-acceptance)
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Yes/No/Not Applicable]

Safe and Interpretable Estimation of Optimal Treatment Regimes: Supplementary Materials

Appendix A Dynamic Treatment Regime & Reinforcement Learning Literature Survey

There are a number of different techniques for estimating optimal treatment regimes. Prior methods include parametric, semi-parametric, and non-parametric modeling approaches and are often combined with reinforcement learning (RL) frameworks such as Q-learning and policy gradient. We categorize the existing methods into five categories: *Finite Timestep Backward Induction*, *Infinite Time Horizon*, *Censored Data*, *Deep Reinforcement Learning*, and *Causal Nearest Neighbors*. Methods from each of these categories excel in certain settings. However, in this section, we highlight the limitations of each approach that ultimately make them unsuitable for our complex, high-stakes problem.

Finite timestep backward induction methods make up the majority of optimal treatment policy estimation methods. Murphy (2003) and Robins (2004) were some of the first ones to utilize backward induction in a semiparametric approach using approximate dynamic programming. Murphy (2005) introduced the now widely used Q-learning, of which initial extensions focused on using parametric and semi-parametric modeling of the Q-functions (Moodie and Richardson, 2010; Chakraborty et al., 2010; Song et al., 2015). These approaches can produce interpretable policies and can be easier to implement. However, correct specification of the Q-functions can be difficult, particularly with observational data (Moodie et al., 2014). This can lead to poor estimates of the optimal policy when a misspecified linear model is used. For this reason, recent work has focused on using flexible non-parametric machine learning methods (Zhang et al., 2012; Zhao et al., 2015; Murray et al., 2018; Blumlein et al., 2022), particularly within the Q-learning framework (Qian and Murphy, 2011; Moodie et al., 2014; Zhang et al., 2018). While these methods are less prone to model misspecification, they often result in complex treatment regimes for which the rationale behind the treatment decision is difficult to discern. Although, Blumlein et al. (2022) and Zhang et al. (2018) proposed more explainable nonparametric approaches.

The majority of backward induction methods assume all patients have the same number of fixed timesteps, which presents difficulty when working with variable timesteps across patients and unobserved states. Infinite horizon methods, like fitted Q-iteration (Ernst et al., 2005; Clifton and Laber, 2020) and Ertefaie and Strawderman (2018)'s Q-learning approach, are better suited to handle these complexities. However, these methods necessitate a reward value to be associated with every action taken by each unit. These reward values are often assumed to be a measurable value that is intrinsically linked to the optimization problem. When this is not the case, they need to be calculated using a predefined function over the observed variables. Having to create such a reward function is often a difficult task that can lead to poor optimal regime estimates (Mataric, 1994; Koenig and Simmons, 1996; Singh et al., 2010). Other work has investigated using backward induction with censored data (Goldberg and Kosorok, 2012; Lyu et al., 2023; Zhao et al., 2020). However, these methods have focused on survival analysis time-to-event tasks, which differ from our setup where we have a labeled outcome for each patient.

Regardless of time-step constraints, all of the methods discussed thus far assume that there are a discrete number of treatment options at each time point. Furthermore, while there is extensive work on backward induction methods for observational data (Moodie et al., 2012), many methods impose a strong positivity assumption over all of the treatments at each timepoint (Qian and Murphy, 2011; Zhao et al., 2015; Blumlein et al., 2022). This assumption is often broken in observational data. For example, in the medical setting patient care is given under the supervision of a trained professional and thus, unless randomized, at any given time point a patient in a particular state may have a near-zero chance of receiving a particular treatment. While approaches like Schulte et al. (2014) do employ weaker positivity assumptions, there is limited discussion on how various backward induction methods handle extremal propensity scores.

Deep reinforcement learning (RL) methods are a fast-growing area of research for optimal treatment regime estimation. Deep RL methods can be categorized as online or offline and on-policy or off-policy. In real-world high-stakes settings online and on-policy methods are infeasible, limiting the scope of applicable methods to offline, off-policy approaches. Mnih et al. (2013) introduced Deep Q-Learning as an effective method for off-policy RL and Lillicrap et al. (2015) extended this method to a continuous action space with deep deterministic policy gradient (DDPG). More recent work has focused on improving upon DDPG by improving sampling efficiency (Haarnoja et al., 2018), limiting overestimation bias (Fujimoto et al., 2018a; Kumar et al., 2020), overcoming extrapolation error (Fujimoto et al., 2018b), and using a critic-regularized approach (Wang et al., 2020).

Deep RL methods are capable of learning complex optimal treatment regimes and can handle variable and infinite timesteps. These methods are significantly more data and resource-hungry than non-deep learning approaches. Although, Deep RL methods like Tschantz et al. (2019) and Haarnoja et al. (2018) offer improvements in this area. A larger issue with Deep RL is that it requires reward values to be associated with each action. This can cause issues similar to those discussed with infinite horizon methods. A possible solution to not having reward values for infinite horizon and Deep RL methods is to use inverse reinforcement learning to learn a good reward function (Ng et al., 2000; Arora and Doshi, 2021). However, such an approach would add an additional layer of complexity to the estimation procedure. In the case of Deep RL, this would further exacerbate what is already its most crucial limitation in its inherent lack of interpretability. The black-box nature of Deep RL makes it a poor choice for optimal treatment regime estimation in high-stakes applications.

The previously mentioned methods either assume (i) the correct specification of a reward function or (ii) that there are no missing states or actions leading up to the final outcome. These assumptions do not align with our real-world scenario.

Matching is an intuitive method for optimal treatment regime estimation. Despite its inherent interpretability, little work has been done in this area. Zhou and Kosorok (2017) used a nearest-neighbor approach that examined the causal treatment effects within neighborhoods of similar patients to estimate optimal treatment regimes. While mentioning that their method can be extended to observational studies, they focus on randomized controlled trials - lacking theoretical or experimental results for the observational setting. Furthermore, they only consider a singular timestep with discrete treatment options and use a limited univariate approach for matching in high dimensions. Ultimately, their matching approach shows promise as an accurate and interpretable approach to optimal treatment regime estimation but is unable to handle the complexities commonly found in real-world problems.

Ideally, we want a method that can handle continuous action and state spaces, missing timesteps, does not require a reward function to be specified, and can be trained on a small number of samples. Furthermore, we want a method that is interpretable given the high-stakes setting. Table 2, in the main text, summarizes the different optimal treatment regime estimation approaches in regard to these desired attributes. In Section 5, we present our matching approach for optimal treatment regime estimation. We subsequently present results showing our method's superior performance over a number of comparison approaches across various settings (see Section 6 and Appendix F.1). Ultimately, to the best of our knowledge, our method is the only approach that possesses all of the qualities needed to effectively address our problem.

Appendix B Distance Metric Learning and Almost Exact Matching

In this section, we discuss some recent and relevant work in the almost exactly matching and distance metric learning literature. In an ideal scenario, we would achieve exact matches for some units. However, in high-dimensional contexts with continuous covariates, exact matches are rare. When performing nearly exact matching with a caliper of r, the objective is to achieve a close match on relevant features while not being overly concerned about matching on irrelevant ones. Therefore, especially in cases with limited data, the choice of the distance metric d for matching becomes crucial. Recent matching approaches have focused on distance metric learning before the matching process. One such approach, Genetic Matching (Diamond and Sekhon, 2013), employs a genetic algorithm to learn an appropriate distance metric. However, it has been found to perform poorly for individualized estimation and is limited to binary or categorical exposures. Another method, Matching After Learning to Stretch (MALTS) (Parikh et al., 2022), is effective for individualistic estimation but struggles to converge in high-dimensional settings with small datasets. A recent approach called Variable Importance Matching (VIM) (Lanners et al., 2023) uses a highly regularized model like LASSO or a shallow decision tree

to model $\mathbb{E}[Y|\mathbf{V}]$. It then utilizes the variable importance scores from the fitted model to guide the selection of the distance metric. This approach is both fast and interpretable and works well in high-dimensional scenarios, making it well-suited for our problem.

Appendix C Pharmacokinetics and Pharmacodynamics

In this section, we discuss our modeling choice for PK and PD mechanistic models.

Pharmacokinetics. We use a one-compartment PK model to estimate the concentration of drug j for patient i at time t ($D_{j,i,t}$) as:

$$g_i(\{\mathbf{D}_{i,t'}\}_{t'=1}^{t-1}, \mathbf{Z}_{i,t}) = e^{-\gamma_{j,i}} D_{j,i,t-1} + Z_{j,i,t}, \tag{1}$$

where pharmacokinetic parameter $\gamma_{j,i}$ is proportional to the half-life of the drug j in patient i.

Pharmacodynamics. We model PD using Hill's model (Nelson et al., 2008) to estimate the short-term effectiveness of the ASMs in reducing EA burden:

$$f_i(\{E_{i,t'}\}_{t'=1}^{t-1}, \mathbf{D}_{i,t}) = \beta_i \left(1 - \sum_j \frac{D_{j,i,t}^{\alpha_{j,i}}}{D_{j,i,t}^{\alpha_{j,i}} + ED50_{j,i}^{\alpha_{j,i}}}\right), \tag{2}$$

where β_i is patient i's EA burden when no drugs are administered, $\alpha_{j,i}$ models the affinity of drug j's ligand to a receptor for patient i, and $ED50_{j,i}$ is the amount of drug concentration necessary to reduce EA burden by 50% from the maximum level.

Appendix D Data Generative Mechanism for the Simulation Study

We base our synthetic data experiments on our real world application where patients experiencing seizures are treated with anti-seizure medications. For our synthetic experiments, we let the first-order pharmacological state-transition model outlined in Appendix C be the true model for each patient's drug response and EA burden progression.

For each patient $i \in \{1, \ldots, n\}$, the PK/PD model is defined by the following parameters: β_i, $\gamma_{i,j}$, $\alpha_{i,j}$, and $ED50_{i,j}$ for each drug j. For simplicity, and to allow for comparison to more methods, we consider a setting with only one drug. Associated with each patient are p pre-treatment covariates, $X_{i,1}, \ldots, X_{i,p} \overset{iid}{\sim} \text{Normal}(0, 1)$. We let the PK/PD parameters be correlated with the pre-treatment covariates $\mathbf{X}_i$ such that $\beta_i \sim \text{Normal}\,(100 + 10X_{i,1}, 5)$ and $ED50_i \sim \text{Normal}\,(15 - 2X_{i,3}, 1)$. Further, $\gamma_i, \alpha_i \overset{iid}{\sim} \text{Normal}(1, 0.1)$.

From here, we let the total number of timesteps, τ_i, be a random integer in $[T_{min}, T_{max}]$ and set the number of observed states as $T_i = \tau_i - m_i$, where m_i is the number of unobserved timesteps and is a random integer in $[M_{min}, M_{max}]$. Finally, $E_{i,0}$, the initial burden for patient i, is sampled as $E_{i,0} \sim \text{Normal}(75 + 5X_{i,2}, 5)$, and is lower bounded by 0 and upper bounded by β_i.

We simulate a complete sequence of states $\{E_{i,t}\}_{t=1}^{\tau_i}$ and actions $\{Z_{i,t}\}_{t=1}^{\tau_i}$ given the initial burden $E_{i,0}$, a policy π_i, and the patient's corresponding PK/PD parameters. We use the same PK/PD equations outlined in Appendix C with a small amount of noise added to the patient's EA burden at each timestep. In particular, we calculate the EA burden for patient i at timestep t by slighltly modifying Equation 2 in Appendix C so that

$$E_{i,t} = \beta_i \left(1 - \sum_j \frac{D_{i,t}^{\alpha_i}}{D_{i,t}^{\alpha_i} + ED50_i^{\alpha_i}}\right) + \epsilon_{E_{i,t}}. \tag{3}$$

where $\epsilon_{E_{i,t}} \sim \text{Normal}(0, 2.5)$. This produces a series of EA burdens $\{E_{i,t}\}_{t=1}^{\tau_i}$ drug doses $\{Z_{i,t}\}_{t=1}^{\tau_i}$ and drug concentrations $\{D_{i,t}\}_{t=1}^{\tau_i}$ corresponding to each patient i. The outcome is related to the patient's pre-treatment covariates, EA burdens, and drug concentrations - thus inducing a level of confounding. In particular, we calculate the continuous outcome value as

$$O_i = \frac{1}{\tau_i}\left[\exp\left(\sum_{j=1}^{2}\frac{X_{i,j}}{2}\right)\left(\sum_{t=1}^{\tau_i}\exp\left(\frac{E_{i,t}}{50}\right)-1\right)+\exp\left(\sum_{j=3}^{4}\frac{X_{i,j}}{2}\right)\left(\sum_{t=1}^{\tau_i}\exp\left(\frac{D_{i,t}}{50}\right)-1\right)\right] \tag{4}$$

Note that we desire a smaller continuous outcome value. This outcome function represents a scenario where patients with a large average value in $X_{i,1}$ and $X_{i,2}$ are more at risk from high levels of EA burden. Whereas, patients with a large average value in $X_{i,3}$ and $X_{i,4}$ are more at risk from high drug concentrations. Finally, to emulate the real-world setting where we observe a binary outcome, we discretize the continuous outcomes to a binary outcome for each patient, setting $Y_i = \mathbf{1}[O_i > 3]$.

Remark 4. Three was chosen as our cutoff value for the binary outcomes to create a setting where about 50% of patients experience a bad outcome (i.e. $Y = 1$). By using a static value, we could more easily compare the binary outcomes across a variety of data generation setups.

Ultimately, the observed data for each patient i is $\{X_i, \{E_{i,t}\}_{t=1}^{T_i}, \{Z_{i,t}\}_{t=1}^{T_i}, Y_i\}$. Note that the observed history only includes the states and actions up to timestep T_i, not τ_i, and only includes the binary outcome Y_i, not O_i.

D.1 Data Generation Process Setups

We vary the data generation process in five important aspects to create a comprehensive synthetic experiment under these conditions.

1. Number of pre-treatment covariates.
2. Number of total timesteps.
3. Number of missing timesteps.
4. Size of the action space.
5. Policy creation method (i.e. how we generate π_i).

For each of these five aspects, we consider two separate settings. **We enumerate over all possible combinations for a total of 32 experimental setups.** To align with our real-world dataset size, we set the number of patients $n = 1000$ for all setups. We outline the two options for each aspect below.

1. Number of pre-treamtent covariates.
   (a) 10 pre-treatment covariates ($p = 10$).
   (b) 100 pre-treatment covariates ($p = 100$).
2. Number of total timesteps.
   (a) Each patient has two total timesteps ($\tau_i = 2$ for all i).
   (b) Each patient has between 10 and 15 total timesteps ($T_{min} = 10$, $T_{max} = 15$).
3. Number of missing timesteps.
   (a) No missing timesteps for any patients ($T_i = \tau_i$ for all i).
   (b) Patients are missing a variable number of timesteps. If the number of total timesteps is 2(a), then patients are missing between zero and one timesteps ($M_{min} = 0$, $M_{max} = 1$). Otherwise, if the total number of timesteps is 2(b), then patients are missing between two and five timesteps ($M_{min} = 2$, $M_{max} = 5$)
4. Size of the action space.
   (a) A continuous action space with drug doses allowed in $[0, 100]$.
   (b) A binary action space with only two drug doses allowed $\{0, 50\}$.
5. Policy creation method (i.e. how we generate π_i).
   (a) Random policy. If the action space is continuous, 4(a), then $\pi_i\left(\{E_{i,t'}\}_{t'=1}^{t-1}, \{Z_{i,t'}\}_{t'=1}^{t-1}\right) = \epsilon_{\pi_{i,t}}$ where $\epsilon_{\pi_{i,t}} \sim \text{Uniform}(0, 100)$. If the action space is binary, 4(b), then $\pi_i\left(\{E_{i,t'}\}_{t'=1}^{t-1}, \{Z_{i,t'}\}_{t'=1}^{t-1}\right) = 50\epsilon_{\pi_{i,t}}$ where $\epsilon_{\pi_{i,t}} \sim \text{Bernoulli}(0.5)$.
   (b) An informed policy that is an additive model using ten binary features $F^1, \ldots, F^{10}$. For a patient i at timestep t, the ten features are calculated as:
      i. $F_{i,t}^1 = \mathbf{1}[E_{i,t-1} > 10]$
      ii. $F_{i,t}^2 = \mathbf{1}[E_{i,t-1} > 20]$
      iii. $F_{i,t}^3 = \mathbf{1}[E_{i,t-1} > 30]$
      iv. $F_{i,t}^4 = \mathbf{1}[E_{i,t-1} > 40]$

Policy Type		**Aggressive**		**Moderate**		**Conservative**	
Action Space		**Continuous**	**Binary**	**Continuous**	**Binary**	**Continuous**	**Binary**
Coefficient Values	ζ_1	$10+\epsilon_{\zeta_{a1}}$	0	$\epsilon_{\zeta_{m1}}$	0	$\epsilon_{\zeta_{c1}}$	0
	ζ_2	$10+\epsilon_{\zeta_{a2}}$	50	$\epsilon_{\zeta_{m2}}$	0	$\epsilon_{\zeta_{c2}}$	0
	ζ_3	$20+\epsilon_{\zeta_{a3}}$	0	$10+\epsilon_{\zeta_{m3}}$	0	$\epsilon_{\zeta_{c3}}$	0
	ζ_4	$20+\epsilon_{\zeta_{a4}}$	0	$10+\epsilon_{\zeta_{m4}}$	0	$\epsilon_{\zeta_{c4}}$	0
	ζ_5	$20+\epsilon_{\zeta_{a5}}$	0	$20+\epsilon_{\zeta_{m5}}$	50	$10+\epsilon_{\zeta_{c5}}$	50
	ζ_6	$20+\epsilon_{\zeta_{a6}}$	0	$20+\epsilon_{\zeta_{m6}}$	0	$20+\epsilon_{\zeta_{c6}}$	0
	ζ_7	$\epsilon_{\zeta_{a7}}$	0	$-10+\epsilon_{\zeta_{m7}}$	0	$-10+\epsilon_{\zeta_{c7}}$	-50
	ζ_8	$\epsilon_{\zeta_{a8}}$	0	$-20+\epsilon_{\zeta_{m8}}$	0	$-20+\epsilon_{\zeta_{c8}}$	0
	ζ_9	$20+\epsilon_{\zeta_{a9}}$	0	$20+\epsilon_{\zeta_{m9}}$	0	$20+\epsilon_{\zeta_{c9}}$	0
	ζ_{10}	$\epsilon_{\zeta_{a10}}$	0	$-20+\epsilon_{\zeta_{m10}}$	0	$-20+\epsilon_{\zeta_{c10}}$	0

Table 3: Coefficient values for aggressive, moderate, and conservative policies. All $\epsilon_{\zeta_*} \overset{iid}{\sim} \text{Normal}(0,1)$ and are added to emulate the liberty that experts take to slightly deviate from the preset policies.

v. $F_{i,t}^5 = \mathbf{1}[E_{i,t-1} > 60]$
vi. $F_{i,t}^6 = \mathbf{1}[E_{i,t-1} > 80]$
vii. $F_{i,t}^7 = \mathbf{1}[Z_{i,t-1} > 25]$
viii. $F_{i,t}^8 = \mathbf{1}[Z_{i,t-1} > 50]$
ix. $F_{i,t}^9 = \mathbf{1}[t \geq 3]\mathbf{1}[E_{i,t-1} > 40]\mathbf{1}[\frac{1}{3}\sum_{t'=t-3}^{t-1} E_{i,t'} > 20]$
x. $F_{i,t}^{10} = \mathbf{1}[t \geq 3]\mathbf{1}[Z_{i,t-1} > 40]\mathbf{1}[\frac{1}{3}\sum_{t'=t-3}^{t-1} Z_{i,t'} > 20]$

Then, $\pi_i\left(\{E_{i,t'}\}_{t'=1}^{t-1}, \{Z_{i,t'}\}_{t'=1}^{t-1}\right) = \pi_i\left(\{F_{i,t}^j\}_{j=1}^{10}\right) = \sum_{j=1}^{10} \zeta_j F_{i,t}^j$, where $\zeta_1, \ldots, \zeta_{10}$ are determined by the type of policy assigned to patient i. We define three separate policy types: aggressive (π_i^a), moderate (π_i^m), and conservative (π_i^c). Depending on the size of the action space, the coefficients corresponding to each of the policy types are shown in Table 3.

We then assign a policy to each patient i such that if the patient has a larger average value in $X_{i,1}$ and $X_{i,2}$ then they are assigned an aggressive policy with high probability. And similarly, if the patient has a larger average value in $X_{i,3}$ and $X_{i,4}$ then they are assigned a conservative policy with high probability.

Finally, to emulate a doctor occasionally deviating from the informed policy, at each timestep there is a small chance that the administered dose does not follow the assigned policy π_i. In particular, if the action space is continuous, 4(a), there is a 5% chance that $\mathbf{Z}_{i,t} = \xi_{i,t}$ where $\xi_{i,t} \sim \text{Normal}(E_{i,t}, 10)$. And if the action space is binary, 4(b), there is a 5% chance that $\mathbf{Z}_{i,t} = 50\xi_{i,t}$ where $\xi_{i,t} \sim \text{Bernoulli}(0.5)$.

Varying these five aspects of the data generation process, we generate a suite of results that provide a comprehensive analysis of the strengths and weaknesses of a variety of optimal policy estimation methods. We outline the methods we compare to, and provided implementation details, in Appendix E. Results for all experiments are shown in Appendix F.

Appendix E Comparison Methods and Implementation Details

We compare our matching method to *Finite Timestep Backward Induction Methods*, *Infinite Time Horizon Methods*, and *Deep Reinforcement Learning Methods*. Many of the methods we compare to are not configured to handle all of the complexities present in our data. For this reason, we make adaptations to each of the methods where necessary. In this section, we outline the methods we implement and any adaptations we make. We omit censored data methods due to their focus on survival analysis time-to-event tasks. We also omit the matching method of Zhou and Kosorok (2017) as they do not consider multiple timesteps and only discuss discrete treatment options.

Note One: Many of the methods we compare to can only handle binary or discrete actions spaces. For binary action space methods, we let $Z_{i,t} \in \{0, 50\}$ *and we binarize the doses such that* $Z_{i,t} = 50\,(\mathbf{1}\,[Z_{i,t} > 25])$. *For discrete action space methods, we let* $Z_{i,t} \in \{0, 25, 50, 75, 100\}$ *and we discretize the doses such that* $Z_{i,t} = 25\,(\mathbf{1}\,[Z_{i,t} > 12.5] + \mathbf{1}\,[Z_{i,t} > 37.5] + \mathbf{1}\,[Z_{i,t} > 62.5] + \mathbf{1}\,[Z_{i,t} > 87.5])$.

Note Two: The optimal treatment regime estimation literature normally focuses on maximizing outcomes, not minimizing like we do in our setup. We flip the outcomes in our data for methods that try to maximize in order to account for this.

Note Three: A number of the methods we compare to require a reward value corresponding to each patient i at timestep t, $\{R_{i,t'}\}_{t'=1}^{T_i}$. To calculate these values, we define three separate reward functions: naive, insightful, and oracle. The naive reward function prioritizes reducing EA burden while avoiding large drug doses, but does not consider the patient's pre-treatment covariates. The insightful reward function considers the interaction between $X_{i,1}$ and EA burdens and $X_{i,3}$ and drug doses, but assumes a linear relationship and does not account for $X_{i,2}$ nor $X_{i,4}$. The oracle reward function is of the same form as our outcome function defined in Equation 4. We compare to three configurations of each method that requires reward values, where each configuration uses reward values calculated from a different reward function. The exact reward functions are outlined below. Note that all methods aim to maximize the reward function.

$$\text{Naive: } R_{i,t}\left(\{E_{i,t'}\}_{t'=1}^{t}, \{Z_{i,t'}\}_{t'=1}^{t}\right) = [E_{i,t-1} - E_{i,t}] + \frac{50 - Z_{i,t}}{4} \tag{5}$$

$$\text{Insightful: } R_{i,t}\left(\{E_{i,t'}\}_{t'=1}^{t}, \{Z_{i,t'}\}_{t'=1}^{t}\right) = -\left[\exp\left(X_{i,1}\right)E_{i,t} + \exp\left(X_{i,3}\right)Z_{i,t}\right] \tag{6}$$

$$\text{Oracle: } R_{i,t}\left(\{E_{i,t'}\}_{t'=1}^{t}, \{D_{i,t'}\}_{t'=1}^{t}\right) = -\left[\exp\left(\sum_{j=1}^{2}\frac{X_{i,j}}{2}\right)\left(\exp\left(\frac{E_{i,t}}{50}\right) - 1\right) + \exp\left(\sum_{j=3}^{4}\frac{X_{i,j}}{2}\right)\left(\exp\left(\frac{D_{i,t}}{50}\right) - 1\right)\right] \tag{7}$$

- **Finite Timestep Backward Induction Methods:** We compare to a wide array of finite timestep backward induction methods. The methods we compare to are: Q-learning Murphy (2005); Moodie et al. (2012); Clifton and Laber (2020), BOWL (Zhao et al., 2015), and optimal classifier (Zhang et al., 2012). We used the R package `DynTxRegime` (Holloway et al., 2020) to implement each of these methods. These methods all require a discrete treatment space and the `DynTxRegime` package only handles the binary case. Given that there is a large literature on Q-learning for discrete action spaces with more than two actions, we also implement our own version of Q-learning for multilevel treatments. For these methods, we followed the Q-learning implementation for observational data as outlined by Moodie et al. (2012).
  Finite timestep backward induction methods assume full observation of all states and actions for each patient and that the number of timesteps for each patient is the same. To implement these methods when patients have varying numbers of observable timesteps, we truncate the state and action space to only include the timesteps for which all samples have observed data, $\hat{T} = \min_{i\in\{1,\ldots,n\}} T_i$. We then carry out each method on this subset of the data to generate estimated optimal treatments for timesteps $t \in \{1,\ldots,\hat{T}\}$. From here, we use the model generated at the last observed timestep, $\hat{T}$, to estimate optimal treatments for the remaining $t \in \{\hat{T},\ldots,\tau_i\}$ for each patient i.
  For the binary Q-learning methods implemented using the `DynTxRegime` R package we run two versions. One where the contrasts model is a linear model and one where the contrasts model is a decision tree model. For both versions, we use a linear model for the main effects component of the outcome regression. This results in two binary Q-learning varieties.
  For the optimal classifier method we also run two versions. One where the contrasts model is a linear model and one where the contrasts model is a decision tree model. For both versions, we use a linear model for the propensity score model and main effects component of the outcome regression. We use a decision tree classifier for the classification model. This results in two optimal classification varieties.
  BOWL requires reward values and thus we run a version for each of the three reward functions. We also run a linear kernel and second degree polynomial kernel version of BOWL for each reward function. All versions use a linear model for the propensity score model. This results in six BOWL varieties.
  For the multilevel Q-learning methods, we incorporate the propensity score at each timestep as a term in our Q-function model (Moodie et al., 2012). All propensity scores are estimated with a linear model. We consider three cases: linear model Q-functions, support vector machine with RBF kernels Q-functions, and random forest Q-functions. This results in three multivel Q-learning varieties.
  In total, we generate results from 13 varieties of finite timestep backward induction methods.
- **Infinite Time Horizon Methods:** We compare to infinite time horizon Q-learning. We implement this method using *Fitted Q-iteration* as outlined in Algorithm 2 of Section 4 of Clifton and Laber (2020). Similar to multilevel backward induction Q-learning, we use a linear model to estimate propensity scores and include

them as a term to the Q-funcion. We consider using three different types of models for the Q-functions: linear models, support vector machines with RBF kernels, and random forests. For each model type, we also consider the case of binarizing the doses into $\{0, 50\}$ and discretizing the doses into $\{0, 25, 50, 75, 100\}$. Finally, infinite horizon methods need a reward for each action, so we run each configuration under each of the three reward functions.
In total, we generate results from 18 varieties of infinite time horizon methods.

- **Deep Reinforcement Learning Methods:** We compare to Batch Constrained Q-learning (BCQ) (Fujimoto et al., 2018b), Conservative Q-learning (CQL) (Kumar et al., 2020), Critic Regularized Regression (CRR) (Wang et al., 2020), Deep Deterministic Policy Gradients (DDPG) (Lillicrap et al., 2015), Soft Actor-Critic (SAC) (Haarnoja et al., 2018), and Twin Delayed Deep Deterministic Policy Gradients (TD3) (Fujimoto et al., 2018a). We implement these methods using the `d3rlpy` Python package (Seno and Imai, 2022). All of these methods require reward values, so each was run three separate times – one for each of the three reward functions. We set the number of steps for each model to 10,000 and kept the remaining parameters at their default values.
In total, we generate results from 18 varieties of deep reinforcement learning methods.

In addition to the optimal treatment regime estimation methods outlined above, we compare to a handful of other baselines. We refer to these as **preset policies** and outline each of them below.

- **Expert:** This baseline is meant to emulate an educated doctor strictly following the informed policy with no deviation. Here we assign policies to each patient i as done in the informed policy creation method 5(b). However, we remove all the noise we added to 5(b).In particular, $\epsilon_{\eta_*} = 0$ and there is a 0% chance that the doctor deviates from the assigned policy at each timestep.
- **Random:** Random dosing at each timestep. If the action space is continuous, 4(a), then $Z_{i,t} = \xi_{i,t}$ where $\xi_{i,t} \sim \text{Uniform}(0, 100)$. Otherwise, if the action space is binary, 4(b), then $Z_{i,t} = 50\xi_{i,t}$ where $\xi_{i,t} \sim \text{Bernoulli}(0.5)$.
- **Inaction:** No drug is administered to any patients at any timesteps. $Z_{i,t} = 0$ for all i and t.
- **Full Dosing:** If the action space is continuous, 4(a), then a dose of 100 is given at every timestep. $Z_{i,t} = 100$ for all i and t. If the action space is binary, 4(b), then a dose of 50 is given at every timestep. $Z_{i,t} = 50$ for all i and t.

We implement our method as outlined in Section 5. Since here we know the true underlying PK/PD parameters, we omit Step 1 from our method to ensure a fair comparison. We first estimate each patient's observed regime with a linear model, using the ten features in 5(b) of Appendix D as our policy template. We then learn a distance metric with a linear model and use that distance metric to perform nearest neighbors matching. We create matched groups of size five for each patient, where we match to the five closest patients with good outcomes. Finally, we perform linear interpolation over the patients' policies in each matched group to estimate the optimal policy, $\hat{\pi}_i^*$, for each patient i.

Appendix F Synthetic Data Experiments: Additional Results and Implementation Details

In Section 6 we present just a small selection of the results from our synthetic data experiment. Here we provide all of our results and further implementation details. We give a comprehensive analysis of key findings in Section F.1. We provide additional experimental implementation details in Section F.2. Given the number of approaches (54) and data generation process setups (32) we ran tests for, we include our full results in our publically available GitHub repository (https://github.com/almost-matching-exactly/opt_tx_regime_matching). We outline each file and its contents in Section F.3.

F.1 Additional Results for Synthetic Data Experiments

Summary of our Analysis. We first compare our method with the 39 approaches that do not use the oracle reward function and are not a preset policy. As noted in Section 6, on the 8 setups with 10-15 timesteps where the observed data is generated from educated policies, our method is consistently the top performer. Looking at the 8 setups where we have 10-15 timesteps and 2-5 missing timesteps, our method outperforms all other approaches in the majority of setups (5 of 8) and is always among the top four performing approaches – never more than 4.5 percentage points worse than the best approach. In the 16 setups with 10-15 timesteps we are

the best performing method 9 of 16 times and among the top 4 approaches 16 of 16 times - never more than 7 percentage points worse than the top approach. When compared on all 32 simulations setups, where some are specifically designed for finite-timestep backward induction methods to perform well, our method outperforms all of the comparison approaches in 17 of the 32 setups and is among the top 4 approaches 29 times – never more than 10.1 percentage points worse than the top approach.

When we also consider the oracle reward functions, our method is never more than 12 percentage points worse than the top performing approach on the 8 setups with 10-15 timesteps and 2-5 missing timesteps and never more than 15 percentage points worse than the top performing approach across all 32 setups.

All of these upper limits on the number of percentage points between our method and the top performing approach are the lowest such values for any method. Ultimately, our simulation results show that our method is frequently the best approach and that its performance is consistent across a variety of scenarios.

In the remainder of this section, we perform an in-depth analysis comparing our method to each of the categories of existing DTR and RL methods that we implemented. We focus on finite-timestep backward induction methods Q-learning and optimal classifier in Appendix F.1.1, infinite horizon methods in Appendix F.1.2, Deep RL in Appendix F.1.3, and BOWL in Appendix F.1.4. In each subsection, we comment on the strengths and weaknesses of the methods, ultimately highlighting how our approach is superior for estimating optimal treatment regimes in complex high-stakes settings.

F.1.1 Analyzing Q-learning and Optimal Classifier Performance

Q-learning and **Optimal Classifier** methods implemented using the `DynTxRegime` R package struggle in complex settings for what we presume is a variety of reasons. Figure 5 details the performance of our method, Q-learning, and optimal classifier with varying action spaces (binary vs. continuous), number of timesteps (2 vs. 10-15), and missing states (missing vs. no missing). These plots highlight how our method drastically outperforms Q-learning and optimal classifier in continuous action spaces. It makes sense that Q-learning and optimal classifier struggle with continuous actions spaces, as they are forced to binarize continuous actions spaces, thereby losing important information. Note that the best results across all the plots in Figure 5 are achieved by our method when we allow the doses to be continuous, suggesting that binarizing the treatment is not a good strategy to optimize outcomes for patients.

We also note that our method is far superior in settings with longer time horizons. This aligns with the fact that previous work on finite-timestep backward induction methods has largely focused on the two timestep setting, paying less attention to longer time horizons (Clifton and Laber, 2020). As outlined in Appendix E, when implementing these methods we truncate all of the states to only include timesteps for which all individuals have an observed state and action. This removes a large amount of information from the data and most likely impacts the performance of these methods.

We further note that the finite-timestep backward induction methods perform better, on average, when there are no missing timesteps. Whereas, our method is quite robust to the missingness of states.

As a sanity check, we show the performance of Q-learning and optimal classifier under the conditions that it was primarily designed for in Figure 6. These results show how effective Q-learning and optimal classifer can be in a more conducive setting, with all varieties outperforming our method. However, this performance does not translate to our challenging high-stakes setting, ultimately making these methods ill-suited for our application.

One obvious way to try to improve finite-timestep backward induction Q-learning is to decrease the amount of information loss by discretizing the continuous treatments into more bins. Since the `DynTxRegime` R package does not support multilevel treatments we implemented our own version of Q-learning to handle this. We outline our implementation in Appendix E.

Figure 7 shows a comparison between binary Q-learning with two treatment options and discrete Q-learning with five treatment options. All plots in Figure 7 are in settings with 10 pre-treatment covariates, a continuous action space, and observed data generated using an informed policy. While we do see a gain in performance, this gain is less substantial when there are more timesteps and missing states. Ultimately, the multi-level treatment form of Q-learning still fails to match the performance of our method. This suggests that while Q-learning can improve by increasing the number of discrete dose options, it still struggles with long time horizons and missing states. Furthermore, at some point the small sample size limits the gain in performance Q-learning can achieve

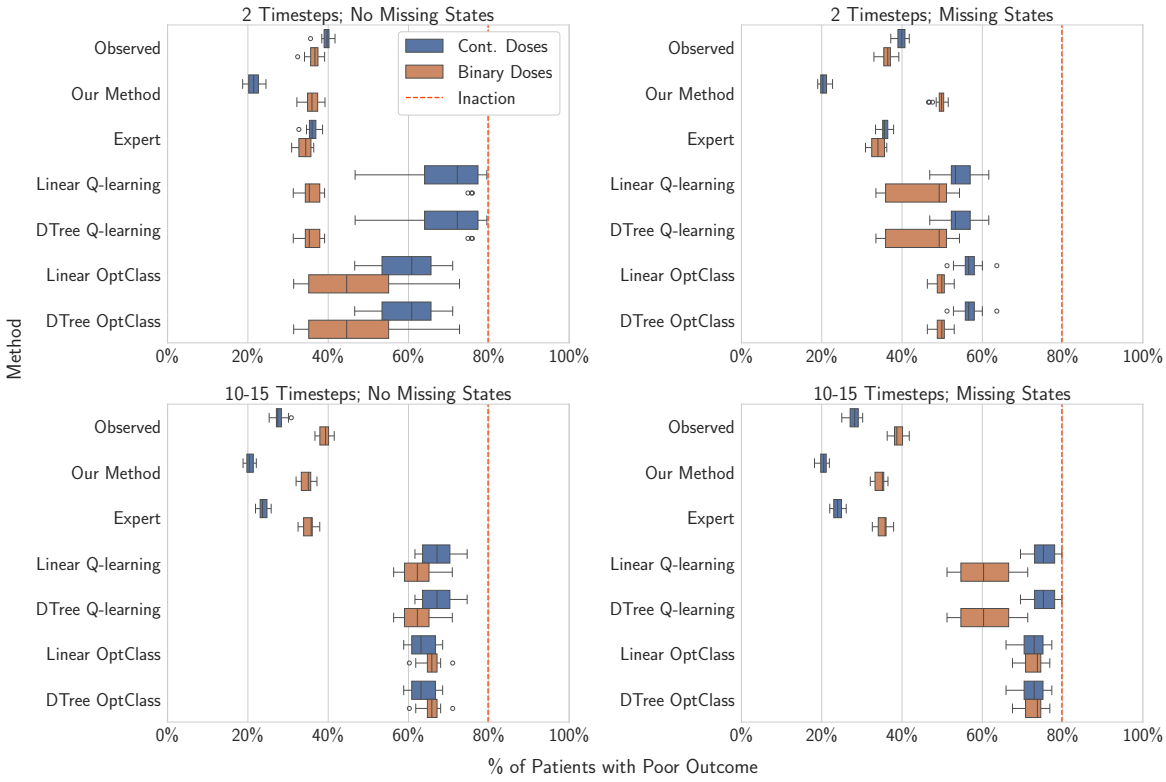


Figure 5: Percent of patients with poor outcomes under different proposed policies (*lower is better*). Boxplots show the distribution of the average outcomes over 20 iterations. *Observed* shows average observed outcomes. *Expert* shows outcomes under the expert policies. *Linear* and *DTree Q-learning* are finite-timestep backward induction Q-learning using either linear models or decision trees. *Linear* and *DTree OptClass* are optimal classifier using either linear models or decision trees. See Appendix E for further details of each method. Note that not all backward induction methods converged for all 20 iterations of each setup. See `all_sims_nan.csv` and Appendix F.3 for details.

by creating more treatment bins.

F.1.2 Analyzing Infinite Horizon Performance

Infinite Horizon methods can overcome the issue finite-timestep backward induction methods face with longer time horizons and missing states/actions. Figure 8 shows a comparison of our method, the infinite horizon method fitted Q-iteration (see Clifton and Laber (2020)), and the finite-timestep backward induction methods Q-learning and optimal classifier. The subplots in this figure show how each method performs with different numbers of missing states and different size action spaces.

Figure 8 highlights how infinite horizon methods can handle long time horizons and missing states much better than finite-timestep backward induction Q-learning and optimal classifier. Fitted Q-iteration can outperform our method when the action space is binary and does especially well when the observed data is generated from a random policy.

However, we still see that fitted Q-iteration struggles with a continuous action space. This is particularly true when the observed data is generated from an informed policy (first row plots of Figure 8). Conversely, our method can handle these added complexities, producing much better results in the setups most resembling a complex real-world setting.

Similar to Figure 7 for backward induction Q-learning, Figure 9 shows how infinite horizon methods can alleviate the problem of a continuous action space by using a multi-level treatment version of fitted Q-iteration instead of a binary version. The bottom row of Figure 9 shows the strong performance of infinite horizon methods

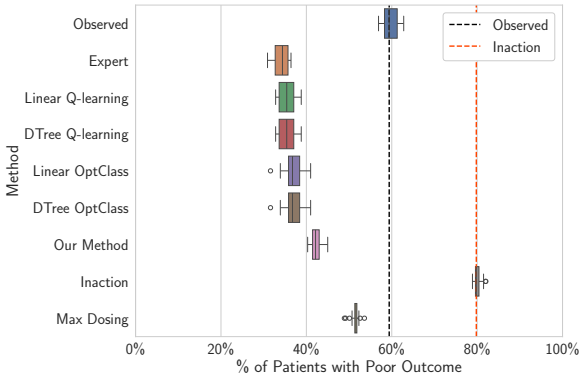


Figure 6: Percent of patients with poor outcomes under different proposed policies (*lower is better*) in a setting more conducive to finite-timestep backward induction methods. Here we set the (i) number of covariates to 10, (ii) number of timesteps to 2, (iii) have no missing states, (iv) only allow binary doses, and (v) generate the observed data from a random policy. Boxplots show the distribution of the average outcomes over 20 iterations. *Observed* shows average observed outcomes. *Expert* shows outcomes under expert policies. *Inaction* and *Max Dosing* administer no drugs and the max amount of drugs to each patient at each timestep, respectively. *Linear* and *DTree Q-learning* are finite-timestep backward induction Q-learning using either linear models or decision trees. *Linear* and *DTree OptClass* are optimal classifier using either linear models or decision trees. See Appendix E for further details of each method.

when using observational data generated from a random policy. Fitted Q-iteration outperforms our method in these setups. However, when the training data is generated from an informed policy (top row of Figure 9), as observational data typically is, our method has superior performance. This could be due to the fact that infinite horizon methods have to deal with the notion of exploration vs. exploitation (Clifton and Laber, 2020), leading to worse performance when the data is collected following a relatively stagnant and educated policy. Observed data collected under such policies essentially has less "exploration" built into it. This struggle could also be due to the fact that infinite horizon methods often work under the assumption that there is a non-zero probability of each action at each timestep (Ertefaie and Strawderman, 2018). However, under the informed policy there are certain states for which certain actions are near-impossible.

Infinite horizon methods are a promising technique, but face a key challenge in our data setup as they require a reward value to be assigned to each action. In our setup, we only observe an outcome at the end of a patient's timesteps. Therefore, we are forced to define a reward function ourselves. We outline the three different reward functions we consider in Appendix E. Figure 9 showed results using the oracle reward function. Figure 10 depicts the stark differences in performance we see using infinite horizon methods with different reward functions. We observe that the performance of infinite horizon methods suffers as the reward function gets farther away from the truth. Researchers typically do not know the oracle, or true, reward function and while we can compare the different reward functions since we know the underlying simulation setup, this is not the case with real-world observational data. Thus, the researcher has to carefully consider the reward function when using infinite horizon methods. This ultimately limits the usefulness of infinite horizon methods in high-stakes applications where reward values are not available for each action that is observed.

F.1.3 Analyzing Deep RL Performance

The performance capabilities of **Deep Reinforcement Learning** is already depicted in Section 6's Figure 1. While DDPG, SAC, and TD3 struggle with the smaller sample sizes and/or the lack of randomness in informed policies, the more modern architectures like BCQ, CQL, and CRR perform well, although slightly worse than our method, on a simulated dataset that resembles our real-world data. However, we note that these Deep RL methods struggle when a random policy is used to generate the observed data. Figure 11 shows how BCQ, CQL, and CRR perform worse when the training data is generated from a random policy. While our method's performance also suffers in this setting, the dip in performance is less severe than Deep RL methods. We hypothesize that Deep RL struggles when using data generated from a random policy because they all use an evaluation set to guide the learning process (Seno and Imai, 2022). Thus, with only a small amount of data

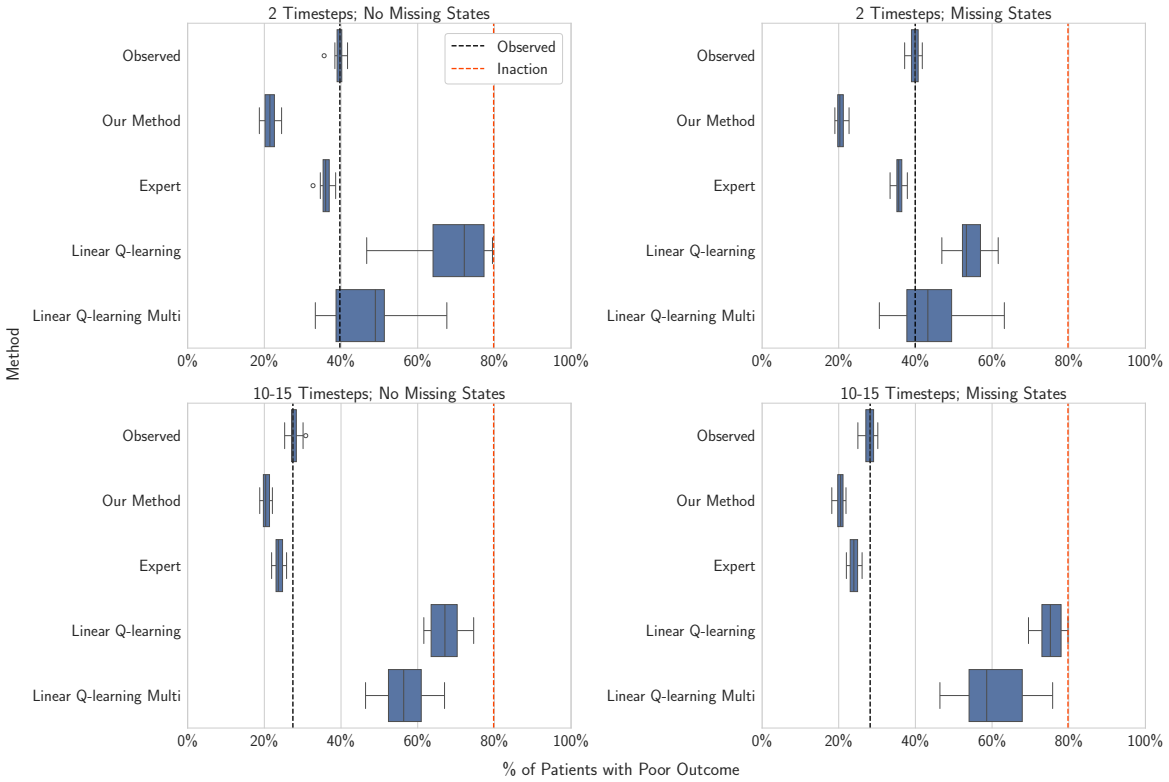


Figure 7: Percent of patients with poor outcomes under different proposed policies (*lower is better*). In all plots, the setup has (i) 10 pre-treatment covariates, (iv) a continuous action space, and (v) generates the observed data using an informed policy. Boxplots show the distribution of the average outcomes over 20 iterations. *Observed* shows average observed outcomes. *Expert* shows outcomes under expert policies. *Linear Q-learning* and *Linear Q-learning Multi* are finite-timestep backward induction Q-learning using linear models. *Linear Q-learning* binarizes the continuous treatment values into two values whereas *Linear Q-learning Multi* discretizes the treatments into five bins. See Appendix E for further details of each method.

generated via a random policy, it is difficult to evaluate the model's performance. Deep RL methods would likely improve if we had significantly more randomly generated data or had the ability to do online learning (Luo et al., 2023).

Deep RL approaches, like infinite horizon methods, also require a reward to be specified for each action. Figure 12 shows that the performance of the best Deep RL methods when using each of the three different reward functions outlined in Appendix E. We observe that the performance is stable across these three reward functions when training on data generated from informed policies. Although, we note that all three reward functions are at least slightly related to the outcome, and thus performance could suffer if the reward function was badly misspecified.

Ultimately, deep reinforcement learning methods show promise for optimal treatment regime estimation from observational data generated by domain experts. The main drawbacks of Deep RL in our setting is its fundamental lack of interpretability. The inability to explain the estimates generated by Deep RL makes it ill-suited for high-stakes applications in the medical field.

As an aside, we also note that Deep RL methods require substantially more compute power to train than our method and any of the other methods we compare to. We train these models using significantly more compute power and GPUs Even with the enhanced computing power, these methods have substantially longer runtimes. See Appendix F.2 for further details.

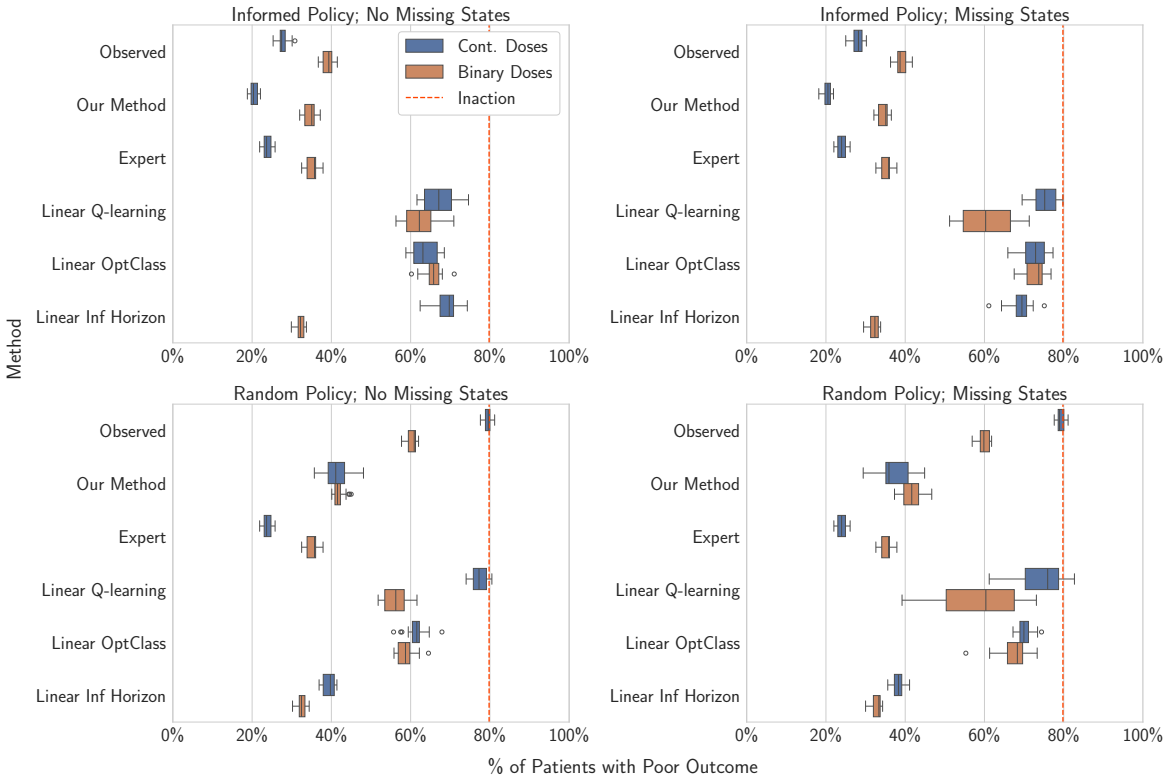


Figure 8: Percent of patients with poor outcomes under different proposed policies (*lower is better*). In all plots, the setup has (i) 10 pre-treatment covariates and (ii) 10-15 total timesteps. Boxplots show the distribution of the average outcomes over 20 iterations. *Observed* shows average observed outcomes. *Expert* shows outcomes under expert policies. *Linear Q-learning* is finite-timestep backward induction Q-learning using linear models. *Linear OptClass* is optimal classifier using linear models. *Linear Inf* is the infinite horizon method fitted Q-iteration using linear models. *Linear Inf* uses the oracle reward function. See Appendix E for further details of each method and the reward functions.

F.1.4 Analyzing BOWL Performance

The final method we compare to is Backward Outcome Weighted Learning, **BOWL**. We find that the BOWL method implemented in `DynTxRegime` struggles to consistently converge, especially when the training data has more timesteps. We show the frequency in which BOWL fails to run for different configurations and reward functions in Figure 13. We further discuss the most likely reasons for these runtime issues, and the steps we took to avoid them, in Appendix F.2. The instability of BOWL for the vast majority of our data configuration setups makes it difficult to discern what aspects of the data are causing it the most problems. We ultimately conclude that BOWL, as implemented in the `DynTxRegime` R package, is ill equipped to handle the challenges present in our simulated data.

F.2 Additional Implementation Details for Synthetic Data Experiments

Code to reproduce the results in this paper is available at https://github.com/almost-matching-exactly/opt_tx_regime_matching.

We run each of the methods outlined in Section E for a total of 20 iterations for each data generation setup. Tests are run on a Slurm cluster with VMware, where each VM is an Intel(R) Xeon(R) Gold CPU (either 5317 @ 3.00GHz, 5320 @ 2.20GHz, 6142 @ 2.60GHz, 6152 @ 2.10GHz, 6226 @ 2.70GHz, or 6252 @ 2.10GHz). Deep RL methods are run on machines with RTX2080 GPUs. Slurm jobs are allocated a single core with 2 GB of RAM for non-Deep RL methods and 16 GB of RAM for Deep RL methods. We always set the random seed to match the iteration number of each setup.

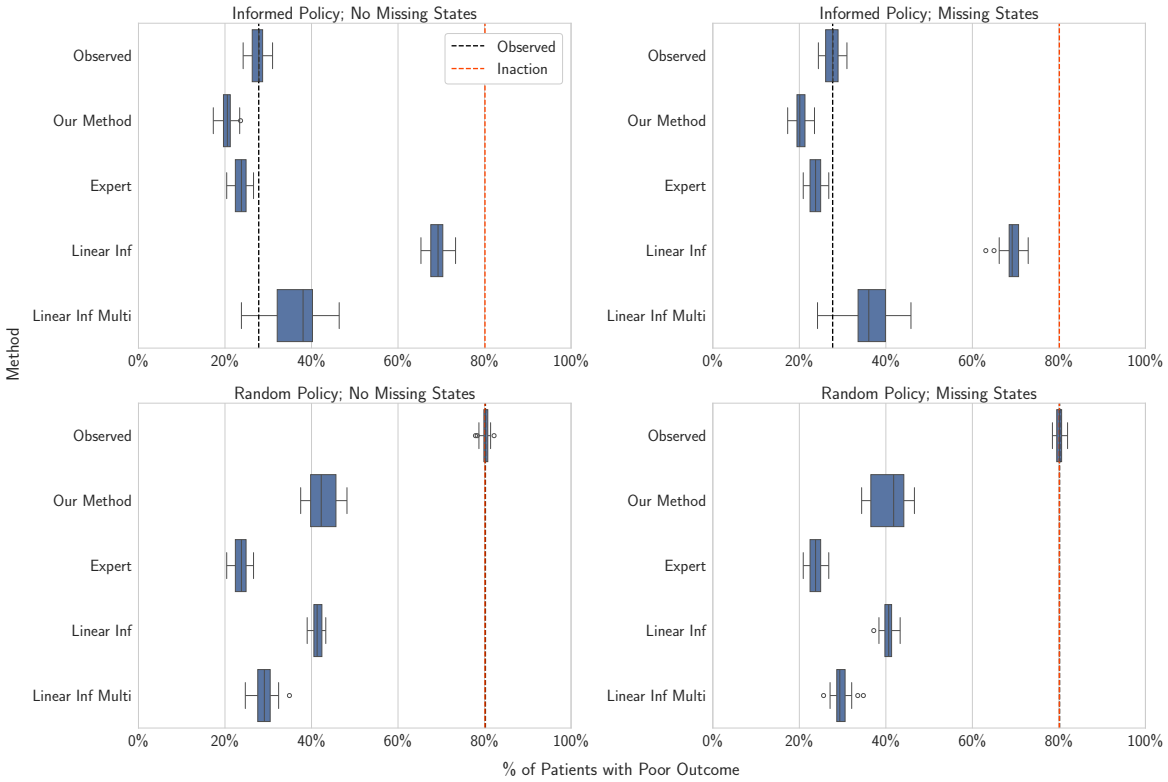


Figure 9: Percent of patients with poor outcomes under different proposed policies (*lower is better*). In all plots, the setup has (i) 100 pre-treatment covariates, (ii) 10-15 total timesteps, and (iv) a continuous action space. Boxplots show the distribution of the average outcomes over 20 iterations. *Observed* shows average observed outcomes. *Expert* shows outcomes under expert policies. *Linear Inf* is Fitted Q-iteration where the treatments are binarized and *Linear Inf Multi* is Fitted Q-iteration where the treatments are discretized into five bins. Both methods use linear models and the oracle reward function (see Appendix E for details on reward functions).

We split the dataset into 5 folds to perform estimation using our method and Deep RL methods. For our method, we use 1 fold to learn the distance metric and perform estimation on the remaining 4 folds. We then average across the 4 outcomes for each sample. For Deep RL methods, we use 4 folds for training and perform estimation on the remaining fold - doing this 5 times to get estimates for each sample.

There are some data generation processes for which we did not generate results for each method for all 20 iterations. You can find details on which methods failed to run for which setups in the `all_sims_nan.csv` file described in Appendix F.3. We outline which methods we are missing results for and provide potential explanations below.

- Q-linear, Optimal Classifier, and BOWL implemented using the *DynTxRegime* R package: Each of these methods is missing results for some of the setups because they failed to converge or produced a runtime error. We attempted to alleviate these issues by running both Q-learning and optimal classifier with decision trees and linear models and running BOWL with a linear kernel and a second degree polynomial kernel. We performed five-fold cross-validation to choose the lambda for BOWL. However, the package errored out if any of the folds failed to converge. We added exception handling to account for this, where we attempted to fit BOWL with preset lambda values of 2 and then 0.5 if it produced an error while performing cross-validation. After investigation, we hypothesized that Q-learning and optimal classification failed to converge for extremal propensity scores in observational data. This is supported by the fact that their errors only occurred when the policy was semi-random. For this policy choice, there are timesteps where a patient's next dose is mostly predetermined by their current state - thus leading to very small or large propensity scores.
  We believe that BOWL struggles with a similar issue, given that it also employs the use of a propensity score. However, BOWL failed to converge for a number of the setups that used a random policy. We acknowledge

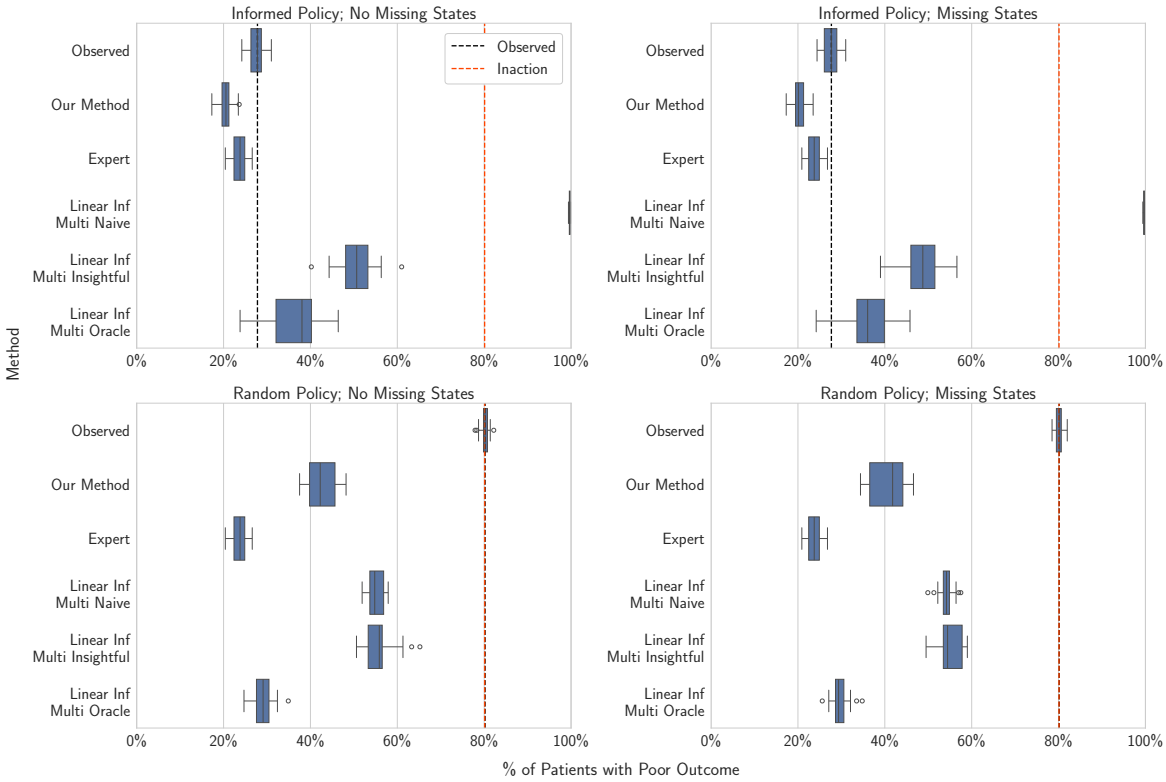


Figure 10: Percent of patients with poor outcomes under different proposed policies (*lower is better*). In all plots, the setup has (i) 100 pre-treatment covariates, (ii) 10-15 total timesteps, and (iv) a continuous action space. Boxplots show the distribution of the average outcomes over 20 iterations. *Observed* shows average observed outcomes. *Expert* shows outcomes under expert policies. All *Linear Inf Multi* methods are fitted Q-iteration approaches that discretize the treatment into five bins and use linear models. *Naive*, *Insightful*, or *Oracle* at the end of each *Linear Inf Multi* method specifies which reward function is uses to calculate the reward values. See Appendix E for details on reward functions.

that the software package made it difficult to discern if the errors were being produced due to an issue with how we were implementing it in the `DynTxRegime` R package or with the BOWL method itself. Thus, we are less sure of the exact reasons why BOWL struggled for so many of our setups.

- Multilevel Q-learning and Infinite Horizon methods: We only ran multilevel treatment method when the treatment was continuous, as running these methods when the treatment was binary was equivalent to the binary version of the method.
- Deep RL methods implemented using the `d3rlpy` Python package: The Deep RL methods we compare to only accept continuous actions spaces. Therefore, we do not have results for any of the setups where the action space was discrete. Also, for two of the setups with continuous action spaces, 2 total timesteps, and 0-1 missing timesteps the number of realized doses was such that the methods interpreted the action space as discrete in some of the iterations, causing it to error out.

F.3 Synthetic Data Experiments Results Files

We include files with results for all 54 approaches and 32 simulation setups in our public GitHub repository (https://github.com/almost-matching-exactly/opt_tx_regime_matching/tree/main/simulations_data). The files use seven columns to indicate the settings of the data generation process for that run.

- `Sim`: Indicates the assigned simulation number. All rows with the same sim number are run under the same data generation configuration, except for the random seed.
- `Iter`: Indicates the iteration number of the corresponding `Sim`. The `Iter` value is also used as the random seed for that run.

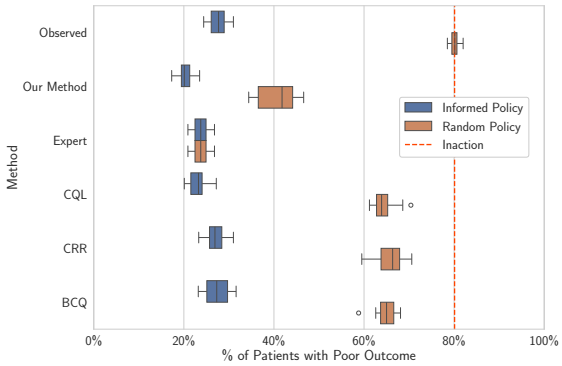


Figure 11: Percent of patients with poor outcomes under different proposed policies (*lower is better*). In all plots, the setup has (i) 100 pre-treatment covariates, (ii) 10-15 total timesteps, (iii) 2-5 missing states, and (iv) a continuous action space. Boxplots show the distribution of the average outcomes over 20 iterations. *Observed* shows average observed outcomes. *Expert* shows outcomes under expert policies. *CQL*, *CRR*, and *BCQ* are all Deep RL methods using the insightful reward function. See Appendix E for details on reward functions.

- `Covs`: The number of pre-treatment covariates.
- `T Setting`: The number of total timesteps setting, where `a` corresponds to setup 2(a) and `b` corresponds to setup 2(b) (see Appendix D.1).
- `T Drop Setting`: The number of unobserved timesteps setting, where `a` corresponds to setup 3(a) and `b` corresponds to setup 3(b) (see Appendix D.1).
- `Binary Dose`: Whether the treatment space is binary or not (if `FALSE` then treatment space is continuous).
- `Policy`: The policy used to generate the observed data. If `random`, than a random policy was used to generate the data. Else if `informed`, than an informed policy was used (see Appendix D.1).

For all methods that use a reward function, we refer to to the Naive reward function as R1, the insightful reward function as R2, and the oracle reward function as R3. We outline the contents of each file below.

- `all_sims_binary_outcomes.csv`: This file contains the average binary outcome value, $\frac{1}{n}\sum_{i=1}^{n} Y_i$, under the proposed policies of each approach. Each row corresponds to the average value for a single iteration of the specified simulation setup.
- `all_sims_cont_outcomes.csv`: This file contains the average continuous outcome value under the proposed policies of each approach. The continuous outcome is simply O_i rather than Y_i in our data generation process outlined in Appendix D. We can report these values since we know the true underlying data generation process. Each row corresponds to the average value for a single iteration of the specified simulation setup.
- `all_sims_nan.csv`: This file contains the number of iterations that each approach failed to produce policy estimates for the 32 simulation setups. See details in Appendix F.2 for explanation on why methods may have failed.
- `sims_binary_outcomes_mean.csv`: This file contains the average binary outcome value across all iterations of each simulation setup for each method. Note that not all methods ran for 20 iterations for each setup. See `all_sims_nan.csv`.
- `sims_binary_outcomes_std.csv`: This file contains the standard deviation of the average binary outcome value across all iterations of each simulation setup for each method. Note that not all methods ran for 20 iterations for each setup. See `all_sims_nan.csv`.
- `sims_binary_outcomes_median.csv`: This file contains the median of the average binary outcome value across all iterations of each simulation setup for each method. Note that not all methods ran for 20 iterations for each setup. See `all_sims_nan.csv`.

We also include `sims_cont_outcomes_mean.csv`, `sims_cont_outcomes_std.csv`, and `sims_cont_outcomes_median.csv` which contain the same content but for the continuous outcome.

Appendix G Data Summary

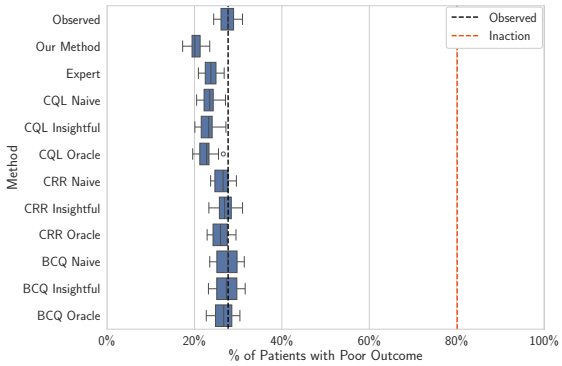


Figure 12: Percent of patients with poor outcomes under different proposed policies (*lower is better*). In all plots, the setup has (i) 100 pre-treatment covariates, (ii) 10-15 total timesteps, (iii) 2-5 missing states, (iv) a continuous action space, and (v) generates data from an informed policy. Boxplots show the distribution of the average outcomes over 20 iterations. *Observed* shows average observed outcomes. *Expert* shows outcomes under expert policies. *CQL*, *CRR*, and *BCQ* are all Deep RL methods where *Naive*, *Insightful*, or *Oracle* at the end of each method specifies which reward function is uses to calculate the reward values. See Appendix E for details on reward functions.

Table 4: Full cohort characteristics and data description.

Variable	**Value**
Age, year, median (IQR)	61 (48 – 73)
Male gender, n (%)	475 (47.7%)
Race	
Asian, n (%)	33 (3.3%)
Black / African American, n (%)	72 (7.2%)
White / Caucasian, n (%)	751 (75.5%)
Other, n (%)	50 (5.0%)
Unavailable / Declined, n (%)	84 (8.4%)
Married, n (%)	500 (50.3%)
Premorbid mRS before admission, median (IQR)	0 (0 – 3)
APACHE II in first 24h, median (IQR)	19 (11 – 25)
Initial GCS, median (IQR)	11 (6 – 15)
Initial GCS is with intubation, n (%)	415 (41.7%)
Worst GCS in first 24h, median (IQR)	8 (3 – 14)
Worst GCS in first 24h is with intubation, n (%)	511 (51.4%)
Admitted due to surgery, n (%)	168 (16.9%)
Cardiac arrest at admission, n (%)	79 (7.9%)
Seizure at presentation, n (%)	228 (22.9%)
Acute SDH at admission, n (%)	146 (14.7%)
Take anti-epileptic drugs outside hospital, n (%)	123 (12.4%)
Highest heart rate in first 24h, /min, median (IQR)	92 (80 – 107)
Lowest heart rate in first 24h, /min, median (IQR)	71 (60 – 84)
Highest systolic BP in first 24h, mmHg, median (IQR)	153 (136 – 176)
Lowest systolic BP in first 24h, mmHg, median (IQR)	116 (100 – 134)
Highest diastolic BP in first 24h, mmHg, median (IQR)	84 (72 – 95)
Lowest diastolic BP in first 24h, mmHg, median (IQR)	61 (54 – 72)
Mechanical ventilation on the first day of EEG, n (%)	572 (57.5%)
Systolic BP on the first day of EEG, mmHg, median (IQR)	148 (130 – 170)
GCS on the first day of EEG, median (IQR)	8 (5 – 13)

History	
Stroke, n (%)	192 (19.3%)
Hypertension, n (%)	525 (52.8%)
Seizure or epilepsy, n (%)	182 (18.3%)
Brain surgery, n (%)	109 (11.0%)
Chronic kidney disorder, n (%)	112 (11.3%)
Coronary artery disease and myocardial infarction, n (%)	160 (16.1%)
Congestive heart failure, n (%)	90 (9.0%)
Diabetes mellitus, n (%)	201 (20.2%)
Hypersensitivity lung disease, n (%)	296 (29.7%)
Peptic ulcer disease, n (%)	50 (5.0%)
Liver failure, n (%)	46 (4.6%)
Smoking, n (%)	461 (46.3%)
Alcohol abuse, n (%)	231 (23.2%)
Substance abuse, n (%)	119 (12.0%)
Cancer (except central nervous system), n (%)	180 (18.1%)
Central nervous system cancer, n (%)	85 (8.5%)
Peripheral vascular disease, n (%)	41 (4.1%)
Dementia, n (%)	45 (4.5%)
Chronic obstructive pulmonary disease or asthma, n (%)	139 (14.0%)
Leukemia or lymphoma, n (%)	22 (2.2%)
AIDS, n (%)	12 (1.2%)
Connective tissue disease, n (%)	47 (4.7%)
Primary diagnosis	
Septic shock, n (%)	131 (13.2%)
Ischemic stroke, n (%)	85 (8.5%)
Hemorrhagic stroke, n (%)	163 (16.4%)
Subarachnoid hemorrhage (SAH), n (%)	188 (18.9%)
Subdural hematoma (SDH), n (%)	94 (9.4%)
SDH or other traumatic brain injury including SAH, n (%)	52 (5.2%)
Traumatic brain injury including SAH, n (%)	21 (2.1%)
Seizure/status epilepticus, n (%)	258 (25.9%)
Brain tumor, n (%)	113 (11.4%)
CNS infection, n (%)	64 (6.4%)
Ischemic encephalopathy or Anoxic brain injury, n (%)	72 (7.2%)
Toxic metabolic encephalopathy, n (%)	104 (10.5%)
Primary psychiatric disorder, n (%)	35 (3.5%)
Structural-degenerative diseases, n (%)	35 (3.5%)
Spell, n (%)	5 (0.5%)
Respiratory disorders, n (%)	304 (30.6%)
Cardiovascular disorders, n (%)	153 (15.4%)
Kidney failure, n (%)	65 (6.5%)
Liver disorder, n (%)	30 (3.0%)
Gastrointestinal disorder, n (%)	18 (1.8%)
Genitourinary disorder, n (%)	34 (3.4%)
Endocrine emergency, n (%)	28 (2.8%)
Non-head trauma, n (%)	13 (1.3%)
Malignancy, n (%)	65 (6.5%)
Primary hematological disorder, n (%)	24 (2.4%)

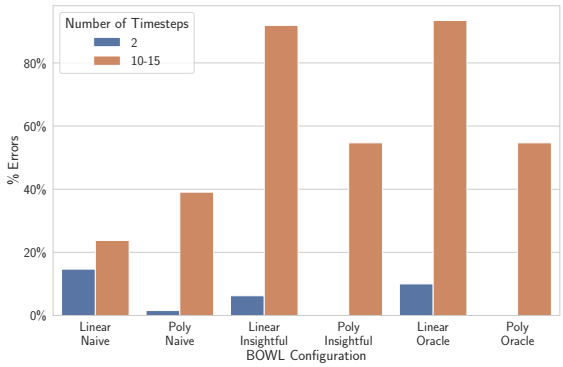


Figure 13: Percentage of total simulation iterations for which different BOWL variations produced either a runtime or convergence error. The different colored bars show the percentage of runs that were errors when the data had either 2 total timesteps or 10-15 total timesteps (see Appendix D.1). *Linear* or *Poly* refer to the kernel type BOWL uses. *Naive*, *Insightful*, or *Oracle* at the end of each method specifies which reward function is used to calculate the reward values. See Appendix E for details on BOWL implementation and reward functions. See Appendix F.2 for further details on BOWL and `DynTxRegime` errors. See `all_sim_nan.csv` and Appendix F.3 for full results on which simulation setups BOWL failed to run for.

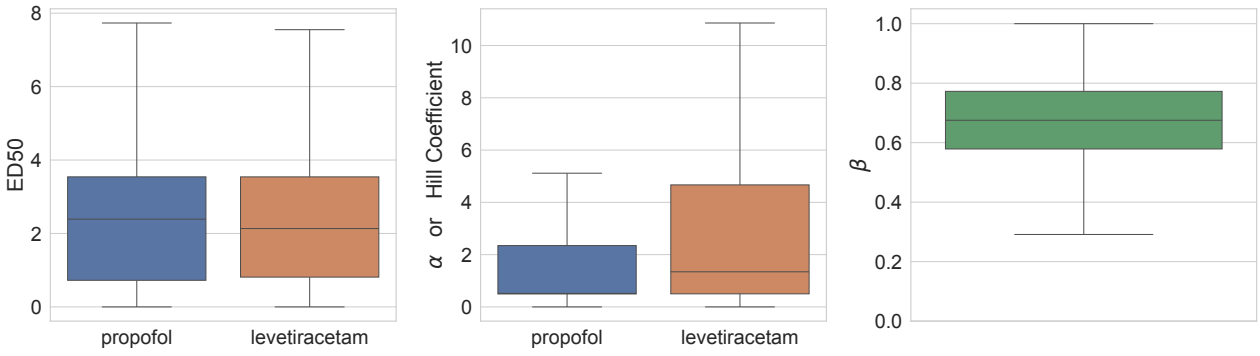


Figure 14: Boxplots showing the distribution of the estimated pharmacodynamics parameters.

Appendix H Anti-Seizure Medications and Policy Templates

H.1 Anti-Seizure Medications

Two drugs were studied: propofol and levetiracetam, Propofol is a sedative antiseizure medication and is given as a continuous infusion, while levetiracetam is a non-sedative antiseizure medication given as a bolus. The doses are normalized by body weight (kg). We use the half-lives from the literature for estimating the drug concentrations $\mathbf{D}$ and estimate the PD parameters using the E and $\mathbf{D}$ for each patient in our cohort (see Table 5 and Figure 14).

Table 5: PK and the estimated average PD parameters for the anti-seizure medications.

Drug	**Half-Life**	**avg.** $\widehat{ED50}$	**avg.**$\widehat{\alpha}$
Propofol	20 minutes	2.41 mg/kg/hr	2.96
Levetiracetam	8 hours	2.26 mg/kg	3.33

H.2 Policy Templates

The regime determining the dose for patient i, for propofol at time t is given by:

$$\begin{aligned}
&\pi_i^{prop}\left(\{E_{i,t'}\}_{t'=1}^{t-1}, \{\mathbf{Z}_{i,t'}\}_{t'=1}^{t-1}; \mathbf{a}_i^p\right) \\
&= a_{1,i}^p \mathbf{1}[E_{i,t-1hr} > 25\%] + a_{2,i}^p \mathbf{1}[E_{i,t-1hr} > 50\%] \\
&+ a_{3,i}^p \mathbf{1}[E_{i,t-1hr} > 75\%] \\
&+ a_{4,i}^p \mathbf{1}[E_{i,t-6hr} > 25\%] + a_{5,i}^p \mathbf{1}[E_{i,t-6hr} > 50\%] \\
&+ a_{6,i}^p \mathbf{1}[E_{i,t-1hr} > 25\%]\mathbf{1}[E_{i,t-6hr} > 25\%] \\
&+ a_{7,i}^p \mathbf{1}[E_{i,t-6hr} > 25\%]\mathbf{1}[E_{i,t-12hr} > 25\%],
\end{aligned} \tag{8}$$

where $\mathbf{a}^p$ is a vector of the parameters for propofol's regime, $E_{i,t-t'}$ is the average EA burden between time $t-t'$ and t, and $Z_{i,j',t-t'}$ is the total dose of drug j' administered between time $t-t'$ and t.

The regime determining the dose for patient i, for levetiracetam at time t is given by:

$$\begin{aligned}
&\pi_i^{lev}\left(\{E_{i,t'}\}_{t'=1}^{t-1}, \{\mathbf{Z}_{i,t'}\}_{t'=1}^{t-1}; \mathbf{a}_i^l\right) \\
&= \mathbf{1}\left[Z_{\text{lev},i,t-12hr} = 0\right] \times \\
&\Big(a_{0,i}^l + a_{1,i}^l \mathbf{1}[E_{i,t-1hr} > 25\%] \\
&+ a_{2,i}^l \mathbf{1}[E_{i,t-1hr} > 50\%] + a_{3,i}^l \mathbf{1}[E_{i,t-1hr} > 75\%] \\
&+ a_{4,i}^l \mathbf{1}[E_{i,t-6hr} > 25\%] + a_{5,i}^l \mathbf{1}[E_{i,t-6hr} > 50\%] \\
&+ a_{6,i}^l \mathbf{1}[E_{i,t-1hr} > 25\%]\mathbf{1}[E_{i,t-6hr} > 25\%] \\
&+ a_{7,i}^l \mathbf{1}[E_{i,t-6hr} > 25\%]\mathbf{1}[E_{i,t-12hr} > 25\%]\Big),
\end{aligned} \tag{9}$$

where $\mathbf{a}^l$ is a vector of the parameters for levetiracetam's regime,

Thus, the regime for patient i, denoted by

$$\pi_i = \left\{ \begin{matrix} \pi_i^{prop}\left(\{E_{i,t'}\}_{t'=1}^{t-1}, \{\mathbf{Z}_{i,t'}\}_{t'=1}^{t-1}; \mathbf{a}_i^p\right) \\ \pi_i^{lev}\left(\{E_{i,t'}\}_{t'=1}^{t-1}, \{\mathbf{Z}_{i,t'}\}_{t'=1}^{t-1}; \mathbf{a}_i^l\right) \end{matrix} \right\}$$

We estimate $\mathbf{a}^p$ and $\mathbf{a}^l$ by minimizing the mean squared error loss between the predicted drug doses and the observed drug doses, $Z_{\text{prop},i,t}$ and $Z_{\text{lev},i,t}$ at each time t.

Appendix I Consistency Proposition and Proof

Before proceeding to the proof of Proposition 1, we note that we consider optimality with respect to linear score functions inside the convex hull of locally observed policies. Our methodology is versatile but operationalized with a linear score function that aligns with the policy template of a prominent tertiary hospital we target (details in Appendix H.2). In this context, our approach identifies an optimal treatment regime aimed at minimizing the probability of adverse outcomes, such as death. Emphasizing patient safety, our search is confined within the convex hull of observed policies for "similar" patients. We now present the proof for Proposition 1.

Proposition 1 (Consistency of Treatment Regime Estimator)**.** *Consider a nest sequence of datasets $\{\mathcal{D}_n\}$ such that $|\mathcal{D}_n| = n$. Then, given conditional ignorability, local positivity, and the smooth outcomes assumptions,*

$$\lim_{n\to\infty} \mathbb{E}[Y_i(\widehat{\pi}_i^{*,(n)}) \mid \mathbf{V}_i] \to \mathbb{E}[Y_i(\pi_i^*) \mid \mathbf{V}_i],$$

where $\widehat{\pi}_i^{*,(n)}$ is the estimate of the optimal treatment regime for unit i estimated using the caliper nearest neighbors interpolation on dataset $\mathcal{D}_n$ with caliper r_n.

Proof. Let $\mu_i(\mathbf{v}, \pi) := \mathbb{E}[Y_i(\pi) \mid \mathbf{V}_i = \mathbf{v}]$ be the expected potential outcome for unit i for which we are interested in estimating the optimal policy, and

$$A_i^{(n)} := \mu_i(\mathbf{V}_i, \widehat{\pi}_i^{*,(n)}) - \mu_i(\mathbf{V}_i, \pi_i^*).$$

By conditional ignorability, $\mu_i(\mathbf{v}, \pi) = \mathbb{E}[Y_i \mid \mathbf{V}_i = \mathbf{v}, \pi_i = \pi]$. Also, let $\widehat{\mu}_i^{(n)}(\mathbf{v}, \pi)$ denote the r_n-caliper nearest neighbor estimate of $\mu_i(\mathbf{v}, \pi)$ on dataset $\mathcal{D}_n$ and $MG_i^{(n)}$ denote the set of all units in $\mathcal{D}_n$ that are at max r_n distance away from $\mathbf{V}_i$. Then, by definition, π_i^* is the policy that, given $\mathbf{V}_i$, minimizes $\mu_i(\cdot, \cdot)$, and $\widehat{\pi}_i^{*,(n)}$ is the policy that, given $\mathbf{V}_i$, minimizes $\widehat{\mu}_i^{(n)}(\cdot, \cdot)$. Thus,

$$\begin{aligned} A_i^{(n)} &\leq \left(\mu_i(\mathbf{V}_i, \widehat{\pi}_i^{*,(n)}) - \widehat{\mu}_i^{(n)}(\mathbf{V}_i, \widehat{\pi}_i^{*,(n)})\right) - \left(\mu_i(\mathbf{V}_i, \pi_i^*) - \widehat{\mu}_i^{(n)}(\mathbf{V}_i, \pi_i^*)\right) \\ &\leq \left|\left(\mu_i(\mathbf{V}_i, \widehat{\pi}_i^{*,(n)}) - \widehat{\mu}_i^{(n)}(\mathbf{V}_i, \widehat{\pi}_i^{*,(n)})\right) - \left(\mu_i(\mathbf{V}_i, \pi_i^*) - \widehat{\mu}_i^{(n)}(\mathbf{V}_i, \pi_i^*)\right)\right| \\ &\leq \left|\left(\mu_i(\mathbf{V}_i, \pi_i^*) - \widehat{\mu}_i^{(n)}(\mathbf{V}_i, \pi_i^*)\right)\right| + \left|\left(\mu_i(\mathbf{V}_i, \widehat{\pi}_i^{*,(n)}) - \widehat{\mu}_i^{(n)}(\mathbf{V}_i, \widehat{\pi}_i^{*,(n)})\right)\right|. \end{aligned}$$

As $n \to \infty$ we shrink $r_n \to 0$ such that $|MG_i^{(n)}| \to \infty$. Then, by the consistency of the caliper nearest-neighbors estimator under smoothness of outcomes, $\widehat{\mu}_i^{(n)}(\mathbf{V}_i, \pi) \to \mu_i(\mathbf{V}_i, \pi)$ (see Remark 5). This implies that, as $n \to \infty$, $A^{(n)} = \mu_i(\mathbf{V}_i, \widehat{\pi}_i^{*,(n)}) - \mu_i(\mathbf{V}_i, \pi_i^*) \to A^{(\infty)} \leq 0$. Further, by definition of , $\mu_i(\mathbf{V}_i, \pi_i^*) \leq \mu_i(\mathbf{V}_i, \widehat{\pi}_i^{*,(n)})$. Thus, we get, $\mu_i(\mathbf{V}_i, \widehat{\pi}_i^{*,(n)}) \to \mu_i(\mathbf{V}_i, \pi_i^*)$, as $n \to \infty$. **QED.**

Remark 5. The consistency of the caliper nearest-neighbors estimator is a standard and well-explored result in the literature (Parikh et al., 2022; Devroye et al., 1994; Kudraszow and Vieu, 2013; Li, 1984; Jiang, 2019; Ferraty et al., 2010; Kara et al., 2017; Einmahl and Mason, 2005). Our context is similar to the one discussed in Theorem 1 of Parikh et al. (2022) and Theorem 2 of Kudraszow and Vieu (2013).

Remark 6. The results in Theorem 2.2 from Zhou and Kosorok (2017), shows similar consistency result of the optimal treatment regime estimator.