

# ABSTRACT

WANG, HAN. Statistical Methods for Missing Data and Composite Outcomes in Reinforcement Learning with Electronic Health Record Data. (Under the direction of Wenbin Lu and Rui Song).

Electronic Health Record (EHR) data contains comprehensive patient information collected from routine clinical practice. The rich observational data facilitates the investigation of optimal treatment decisions. One way to learn optimal treatment policies is through the application of Reinforcement Learning (RL), a branch of machine learning that solves sequential decision-making problems. However, in practice, applying RL techniques to EHR data poses some challenges, including potential bias due to missing data and the challenge of reward construction for composite outcomes. In this dissertation, we develop novel statistical methods to effectively handle the missing data issue and derive the data-driven reward function using expert demonstrations.

In Chapter 2, we focus on a critical step of offline RL known as Off-Policy Evaluation (OPE), which aims to estimate the value of a target policy using data collected from potentially different policies. We investigate OPE in the presence of monotone missingness and theoretically demonstrate that the value estimator remains consistent under ignorable missingness but can be biased when the missing mechanism is nonignorable (informative). To this end, we propose a semiparametric inverse probability weighted value estimator, which is shown to be consistent under nonignorable missingness. Additionally, we establish the asymptotic properties of the proposed value estimator and provide the associated confidence interval. Numerical experiments are conducted to empirically demonstrate the effectiveness of our proposed method in ensuring a more reliable value inference.

In Chapter 3, we consider the challenge of reward construction in the presence of multiple outcomes that need to be optimized. Given expert demonstration data, the Inverse Reinforcement Learning (IRL) technique can be utilized to infer the underlying reward function directly from the data, bypassing the need for manually crafting the reward functions. We propose a novel offline IRL method called Inverse soft-Q and Reward Learning (IQRL), which is accurate, fast, and stable. Through empirical evaluations, our method outperforms baseline methods in various domains, including classic control tasks and real-world EHR data. On the other hand, in practice, human experts often have diverse strategies while making decisions, resulting in heterogeneous demonstrations. In light of this, we further introduce a variant of our method that incorporates reward distillation to capture heterogeneity in expert strategies.

© Copyright 2023 by Han Wang

All Rights Reserved

Statistical Methods for Missing Data and Composite Outcomes in Reinforcement Learning  
with Electronic Health Record Data

by  
Han Wang

A dissertation submitted to the Graduate Faculty of  
North Carolina State University  
in partial fulfillment of the  
requirements for the Degree of  
Doctor of Philosophy

Statistics

Raleigh, North Carolina  
2023

APPROVED BY:

---

Marie Davidian

---

Shu Yang

---

Wenbin Lu  
Co-chair of Advisory Committee

---

Rui Song  
Co-chair of Advisory Committee

# DEDICATION

To my family and friends.

## **BIOGRAPHY**

Han Wang was born in Zhejiang, China in 1996. She completed her undergraduate studies in Statistics at Zhejiang University in 2018. Afterward, she continued her education at North Carolina State University to pursue a doctoral degree in Statistics. She is advised by Dr. Wenbin Lu and Dr. Rui Song, and she completes her Ph.D. in 2023.

## ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisors, Dr. Wenbin Lu and Dr. Rui Song, for their unwavering support and guidance during my doctoral studies. They directed me towards the fascinating research area of reinforcement learning and EHR data, which has been instrumental in shaping the direction of my research. I am especially thankful for the time they dedicated to meeting with me every week, offering invaluable insights and constructive feedback that have fostered my growth as an independent researcher. Their mentorship has been pivotal in the successful completion of this work, and I feel grateful to have their guidance throughout this journey. I would also like to thank Dr. Marie Davidian and Dr. Shu Yang for serving on my dissertation committee and taking the time to provide valuable feedback that helped me refine my work.

I wish to express my appreciation to the faculty members of the Department of Statistics for providing a comprehensive collection of courses that have greatly enriched my academic journey. I am also grateful to all the staff members for their great service and timely assistance throughout my time in the department.

Furthermore, I would like to thank my friends in the department who have helped me along the way, making my time here a memorable experience. In particular, I would like to extend special thanks to Xuan Liu, Jianian Wang, Ye Shen, Hengrui Cai, and Kevin Gunn. It has been a pleasure to have crossed paths with you on this journey, and I wish you all the best in your future endeavors.

My appreciation also goes to my summer intern managers Dr. Yichi Zhang at Meta and Dr. Xiaodong Luo and Dr. Liming Shen at Sanofi. They not only offered me the opportunity to apply my knowledge and skills acquired through research to real-world industry problems but also provided valuable suggestions for my career development. I am truly thankful for their support and the valuable experiences I gained under their supervision.

Last but not least, I would like to express my deepest love and gratitude to my parents for their unwavering support and unconditional love. They have always been encouraging me and standing by me for every decision I make. Without them, I would not be here. Special thanks to my boyfriend, Erjia Cui, for enduring support and the memorable days we have shared. The past several years during the pandemic have been filled with uncertainty, and we are fortunate to have each other's companionship through these difficult times.

# TABLE OF CONTENTS

List of Tables . . . . .	vii
List of Figures . . . . .	viii
<b>Chapter 1 Introduction . . . . .</b>	<b>1</b>
1.1 Background . . . . .	1
1.1.1 Reinforcement Learning . . . . .	1
1.1.2 MIMIC-III Database . . . . .	2
1.2 Practical Challenges in Applying RL to EHR Data . . . . .	3
1.2.1 Missing Data and Dropout . . . . .	3
1.2.2 Reward Construction for Composite Outcomes . . . . .	4
1.3 Notations . . . . .	5
1.4 Overview . . . . .	7
<b>Chapter 2 Off-Policy Evaluation with Nonignorable Missing Data . . . . .</b>	<b>8</b>
2.1 Introduction . . . . .	8
2.2 An Overview of Off-Policy Evaluation Methods . . . . .	10
2.3 Off-Policy Evaluation with Incomplete Data . . . . .	13
2.3.1 Missing Data Mechanism . . . . .	13
2.3.2 Value Inference under Missing Data . . . . .	15
2.3.3 Value Inference with Inverse Weights . . . . .	17
2.4 Generalizability of Proposed Framework . . . . .	20
2.4.1 More General Dropout Patterns . . . . .	20
2.4.2 Extension to Other Off-Policy Evaluation Methods . . . . .	21
2.5 Simulation Study . . . . .	23
2.6 Real Data Application . . . . .	26
2.7 Discussion and Future Work . . . . .	28
<b>Chapter 3 Offline Inverse Reinforcement Learning via Joint Soft-Q and   Reward Learning . . . . .</b>	<b>30</b>
3.1 Introduction . . . . .	30
3.2 Preliminaries . . . . .	31
3.2.1 Maximum Entropy Reinforcement Learning . . . . .	32
3.2.2 Maximum Entropy Inverse Reinforcement Learning . . . . .	33
3.3 Offline Inverse Reinforcement Learning . . . . .	37
3.3.1 Inverse soft-Q and Reward Learning (IQRL) . . . . .	37
3.3.2 Distilled Inverse soft-Q and Reward Learning (D-IQRL) . . . . .	40
3.4 Simulation Study . . . . .	43
3.4.1 Performance of IQRL . . . . .	43
3.4.2 Performance of D-IQRL . . . . .	46
3.5 Real Data Application . . . . .	49
3.6 Discussion and Future Work . . . . .	53

<b>References</b> . . . . .	<b>54</b>
<b>APPENDICES</b> . . . . .	<b>61</b>
Appendix A    Off-Policy Evaluation with Nonignorable Missing Data . . . . .	62
A.1 Sensitivity Analysis . . . . .	62
A.2 Additional Experimental Details . . . . .	63
A.3 Additional Details for Real Data Application . . . . .	65
A.4 Assumptions . . . . .	66
A.5 Proof of Main Results . . . . .	70
A.5.1 Proof of Theorem 1 . . . . .	71
A.5.2 Proof of Theorem 2 . . . . .	76
A.5.3 Proof of Theorem 3 . . . . .	80
A.5.4 Proof of Theorem 4 . . . . .	90
A.5.5 Proof of Lemma 7 . . . . .	92
A.5.6 Connection Between LSTDQ and MWL . . . . .	94
Appendix B    Offline Inverse Reinforcement Learning via Joint Soft-Q and Re- ward Learning . . . . .	96
B.1 Additional Experimental Details . . . . .	96
B.2 Additional Details for Real Data Application . . . . .	98

## LIST OF TABLES

Table 2.1	Results of value estimates and 95% confidence intervals for policy $\pi_1$ in the 2D-Linear environment. The average bias, MSE values, ECP, and AL are reported for each estimator (with standard error in parenthesis).	25
Table 2.2	Results of value estimates and 95% confidence intervals for policy $\pi_2$ in the 2D-Linear environment. The average bias, MSE values, ECP, and AL are reported for each estimator (with standard error in parenthesis).	26
Table 2.3	Results of various value estimators for policy $\pi_1$ in the 2D-Linear environment with horizon $T = 25$ and $n = 500$ trajectories. The average bias (with standard error in parenthesis) and MSE values are reported for each estimator.	28
Table 2.4	Off-Policy evaluation results for three different target policies using the MIMIC-III sepsis dataset. The value estimates and confidence intervals are reported.	29
Table 3.1	Results of reward weight estimation. The estimated weight $\hat{\omega}$ , the correlation between ground-truth rewards and recovered rewards, and the accuracy (ACC) of action prediction are reported (with standard errors in parentheses).	45
Table 3.2	Comparison of the three training approaches. The correlation between true rewards and recovered rewards, the accuracy (ACC) of action prediction, and the obtained values under learned policies are reported (with standard errors in parentheses).	48
Table 3.3	Comparison of policy performance of different imitation learning algorithms on the MIMIC-III dataset, evaluated by the quality of action matching against a held-out test set of demonstrations with cross-validation. The accuracy (ACC), the area under the receiving operator characteristic curve (AUC), and the average precision score (APS) of action prediction are reported (with standard error in parenthesis).	49
Table 3.4	Comparison of policy performance of the three training approaches on the MIMIC-III dataset, evaluated by the site-wise accuracy (ACC) of action prediction against a held-out test set of demonstrations with cross-validation.	52
Table A.1	Results of value estimates and 95% confidence intervals for policy $\pi_1$ under dropout propensity $\lambda_3$ . The average bias, MSE values, ECP, and AL are reported for each estimator (with standard error in parenthesis). The suffix (IV $\times$ ) indicates misspecification of instrumental variable, and the suffix (mod $\times$ ) indicates misspecification of the parametric model for outcome variables.	63

## LIST OF FIGURES

Figure 2.1	The average SOFA scores for deceased patients (red) and patients who survived (blue) during ICU stay. The shadow represents the 25% to 75% quantile. . . . .	9
Figure 2.2	The ECP (top) and AL (bottom) of value confidence intervals for policy $\pi_1$ under nonignorable missingness obtained from IPW (blue) and CC (red) estimators. . . . .	27
Figure 3.1	The heatmaps of ground-truth reward (top-left), value of the ground-truth reward (top-right), recovered reward (bottom-left), and value of the recovered reward (bottom-right) in $5 \times 5$ GridWorld environment.	44
Figure 3.2	The average returns of learned policies using different imitation learning algorithms for Acrobot (left), CartPole (middle), and LunarLander (right). The number of training trajectories increases in the sequence of 1,3,7,10,15 for each algorithm. . . . .	46
Figure 3.3	The recovered rewards and ground-truth rewards for each strategy. In each subplot, the x-axis and y-axis represent the two dimensions of the state variable. The top row corresponds to the reward function for action 0, and the bottom row corresponds to that of action 1. . . . .	47
Figure 3.4	The state-action reward for an otherwise average patient as their blood pH (left) or PaCO <sub>2</sub> level (right) varies. . . . .	50
Figure 3.5	The average rewards weights for different care unit types obtained from D-IQRL. . . . .	53

# Chapter 1

## Introduction

### 1.1 Background

We begin by introducing the background of this work. In Section 1.1.1, we provide a brief overview of Reinforcement Learning, and in Section 1.1.2, we introduce the MIMIC-III database. These two key components motivate the research presented in this dissertation.

#### 1.1.1 Reinforcement Learning

Reinforcement Learning (RL) is a general technique to solve sequential decision-making problems with the goal of learning an optimal policy that maximizes cumulative rewards. Take the classical Cart-Pole control task (Barto et al. 1983) as an illustrative example. In this task, a pole is attached to a cart through an unactuated joint, and the cart can move along a frictionless track. The goal is to balance the pole in an upright position by applying forces to the cart in either the left or right direction. In this context, a policy specifies the appropriate action given the current status of the cart and pole. To learn an optimal policy that keeps the pole in the upright position for as long as possible, one can apply the RL techniques.

Recent years have seen significant progress in RL, with notable methods such as DQN (Mnih et al. 2013), DDPG (Lillicrap et al. 2015), PPO (Schulman et al. 2017), SAC (Haarnoja et al. 2018), and more. These developments have propelled the capabilities of RL and demonstrated successful applications in diverse domains ranging from game playing (Mnih et al. 2013; Silver et al. 2016) to robotic control (Kober et al. 2013). Notably, many of these successes rely on simulators to generate large amounts of interaction data for RL training. However, when applying RL in real-world scenarios, obtaining online data through direct interaction with the environment is often challenging, and deploying a new policy in the environment to evaluate its performance can sometimes be infeasible, especially in

safety-sensitive domains like healthcare (Gottesman et al. 2019; Coronato et al. 2020) and autonomous driving (Sallab et al. 2017). Consequently, there has been a growing interest in developing offline RL algorithms that rely solely on pre-collected data.

### 1.1.2 MIMIC-III Database

Electronic Health Records (EHRs) consist of data collected from routine hospital care. Initially designed for record-keeping and billing purposes, EHRs are now also used for secondary data analysis. The large volume of data available within EHRs enables researchers to explore a wide range of problems and conduct in-depth investigations.

MIMIC-III (Medical Information Mart for Intensive Care III) database (Johnson et al. 2016), an example of EHRs, is a publicly accessible intensive care database that contains de-identified health data from over 40,000 patients who were admitted to the Intensive Care Units (ICUs) at the Beth Israel Deaconess Medical Center between 2001 and 2012. The database encompasses different types of ICUs, including the Medical Intensive Care Unit (MICU), Coronary Care Unit (CCU), Cardiac Surgery Recovery Unit (CSRU), Surgical Intensive Care Unit (SICU), and Trauma Surgical Intensive Care Unit (TSICU).

Comprehensive information is captured in the MIMIC-III database, including patient demographics, vital sign measurements taken hourly at the bedside, laboratory test results, medications, procedures, caregiver notes, imaging reports, and mortality data. The rich information in the database facilitates a diverse range of analytic studies, such as mortality prediction (Pirracchio et al. 2015; Purushotham et al. 2018; Harutyunyan et al. 2019), sepsis and septic shock prediction (Kam and Kim 2017; Scherpf et al. 2019), acute kidney injury prediction (Zimmerman et al. 2019; Sun et al. 2019), and more. Among them, one area of interest is exploring optimal treatment decisions to assist clinical decision-making. In recent years, there have been some endeavors that leverage RL techniques to learn optimal treatments using the MIMIC-III database. For example, Komorowski et al. (2018) and Raghu et al. (2017) investigated the optimal treatments for intravenous fluid and vasopressor dosage for patients with sepsis, and Prasad et al. (2017) studied the optimal decisions regarding sedation dosage and ventilator support.

This dissertation primarily focuses on the application of RL to learn optimal treatment policies using the MIMIC-III database. In Section 1.2, we will discuss some practical challenges arising from real-world applications.

## 1.2 Practical Challenges in Applying RL to EHR Data

The Dynamic Treatment Regime (DTR) literature has extensively explored the derivation of optimal treatment policies from complex longitudinal data (Murphy 2003; Schulte et al. 2014; Zhao et al. 2015). However, these methods are primarily designed for studies with few decision points and may not be suitable for applications involving many decision steps. Recognizing that the EHR data entails continuous decision-making over a prolonged time, we consider the infinite-horizon setting.

While the RL literature has predominantly focused on simulation environments and game-playing, the application of RL encounters additional challenges in real-world scenarios. In this section, we discuss two practical challenges when applying RL to EHR data: the issue of missing data and the construction of appropriate reward functions.

### 1.2.1 Missing Data and Dropout

Missing data is ubiquitous in EHRs, as patients in clinical practice are not as closely monitored as those in clinical trials. Factors such as missed office visits, failure to follow up, and switching healthcare systems often lead to the occurrence of missing data in EHRs. Similar to other EHRs, the MIMIC-III database also exhibits a prevalent issue of missing data, as discussed in Che et al. (2018). When performing data analysis or training models using such datasets, it is crucial to appropriately handle missing data to avoid introducing potential bias into the results.

Identifying the source of missing data is important when handling missing data. In the missing data literature, there are three main mechanisms of missingness: Missing-Completely-At-Random (MCAR), Missing-At-Random (MAR), and Missing-Not-At-Random (MNAR).

The MCAR mechanism describes the situation where missing observations are independent of the observed and unobserved measurements. In this case, the complete data is representative of the study population since the missing data can be viewed as a random sample of all the data. It is important to note that this assumption rarely holds in real-world applications.

The MAR mechanism, also known as ignorable missingness, describes the case where the missing probability does not depend on the unobserved elements conditional on the observed data. Notably, MCAR is a special case of MAR. An example of the MAR mechanism is non-response or lack of measurement that can be attributed to certain known baseline characteristics. In practice, this assumption cannot be verified from the observed data alone, so it is necessary to include as many covariates as possible to make the MAR assumption plausible.

The MNAR mechanism, also known as nonignorable missingness, describes the case where

the missing probabilities depend on unobserved components. For example, patients with low blood pressure are more likely to have fewer blood pressure measurements, leading to missing data for the “blood pressure” variable that depends on the actual blood pressure values. Such a type of missingness is the most challenging to model for. In practice, we cannot evaluate whether missing data is MAR or MNAR from the observed data alone, the missing mechanism needs to be justified based on the context and subject-matter knowledge.

When applying RL to patient trajectories from MIMIC-III, one notable pattern of missingness arises from dropout, which occurred when patients were discharged from the ICU or experienced mortality, resulting in truncated patient trajectories. Improper handling of such dropouts can introduce potential bias into RL. In Chapter 2, we will delve deeper into this issue and propose remedies to mitigate the bias.

### 1.2.2 Reward Construction for Composite Outcomes

RL aims to maximize the cumulative reward in sequential decision-making tasks, and it relies on a reward function to guide policy learning. Unlike game environments where explicit rewards are readily available, real-world applications rely on manually defined rewards. Specifying the reward function can sometimes be challenging, as it involves balancing multiple and possibly competing outcomes of interest. For example, in medical applications, there is often a need to balance symptom reduction with the risk of an adverse event.

When expert demonstration data is available, such as clinician decisions recorded in EHRs, an alternative approach is to apply Imitation Learning (IL) techniques to learn the optimal policy without specifying the reward. However, general IL approaches solely focus on learning the expert policy and do not provide insights into the underlying motivations of the experts. To gain a deeper understanding of expert behavior, a preferred approach is Inverse Reinforcement Learning (IRL), which is a special type of IL that aims to first derive the reward function from expert demonstrations and use the retrieved reward function to guide policy search. By learning the reward function from data, IRL bypasses the need for manually designing reward functions, and the learned reward can also provide insights into the motivations of experts.

While there is a significant body of literature on IRL that allows for flexible reward models and complex state spaces (Wulfmeier et al. 2015; Finn et al. 2016; Ho and Ermon 2016; Fu et al. 2017), these methods are mainly intended for online settings that require further interaction with the environment to gather additional data. However, for safety-critical domains like medical applications, online interactions are not feasible. Therefore, it is essential to develop IRL methods that are compatible with offline settings. In Chapter 3, we will dive

into offline IRL methods and introduce our approach.

### 1.3 Notations

In this section, we prepare for the subsequent contents by introducing some key concepts, notations, and assumptions.

In the RL literature, the environment is often modeled with Markov Decision Processes (MDP). An MDP can be defined by a tuple  $(\mathcal{S}, \mathcal{A}, p_0, p, r, \gamma)$ , where  $\mathcal{S}$  is the state space,  $\mathcal{A}$  is the action space,  $p_0(s)$  is the initial state distribution,  $p(s'|a, s)$  is the Markov transition distribution which characterizes the environment dynamics,  $r(s, a)$  is the reward function with larger positive values indicating preferable outcomes, and  $\gamma \in (0, 1)$  is a discount factor that trades off long-term rewards for immediate rewards. Let  $\{(S_t, A_t, R_{t+1})\}_{t \geq 0}$  denote a trajectory generated by the MDP model, where  $(S_t, A_t, R_{t+1})$  represents the triplet of state, action, and immediate reward. Here we use the notation  $R_{t+1}$  instead of  $R_t$  to emphasize that the reward  $R_{t+1}$  and next state  $S_{t+1}$  are jointly determined. A policy  $\pi$  maps the state space  $\mathcal{S}$  to a probability mass function over the action space  $\mathcal{A}$ , which characterizes how the agent ought to take action in the environment. A stationary policy is one that remains unchanged over time and does not depend on the time step  $t$ . Denote the policy space for stationary policies as  $\Pi$ .

The following two assumptions are commonly imposed in infinite-horizon RL settings, which guarantee the existence of an optimal stationary policy (Puterman 1994).

**Assumption 1** (Time-homogeneous Markov Assumption). *The transition probability satisfies  $P(S_{t+1}|S_t = s, A_t = a, \{S_j, A_j, R_{j+1}\}_{0 \leq j < t}) = p(S_{t+1}|S_t = s, A_t = a) = p(S_1|S_0 = s, A_0 = a)$ , where  $p$  is the transition function.*

**Remark 1.** *The Markovian assumption states that future states are independent of past observations given the current state-action pair. This assumption may not hold for some applications such as chronic disease. If one suspects that the trajectory has higher-order Markovian properties, the state variable  $S_t$  can be constructed by aggregating the state information over multiple time steps, as demonstrated by Shi et al. (2020).*

**Assumption 2** (Conditional Mean Independence Assumption).  $\mathbb{E}(R_{t+1}|S_t = s, A_t = a, \{S_j, A_j, R_{j+1}\}_{0 \leq j < t}) = \mathbb{E}(R_{t+1}|S_t = s, A_t = a) := r(s, a)$ , where  $r$  is the reward function.

**Remark 2.** *This assumption states that the expected current reward is conditionally independent of the history given the current state-action pair, and the reward function is stationarity.*

In practice, the stationarity property can be guaranteed by incorporating time-associated covariates into state features.

For discounted infinite-horizon MDP, the state value function of policy  $\pi$  is defined as the expected discounted cumulative rewards from a state  $s$  following the given policy  $\pi$ ,

$$V^\pi(s) = \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} \gamma^t R_{t+1} \middle| S_0 = s \right] = \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} \gamma^t \cdot r(S_t, A_t) \middle| S_0 = s \right], \quad (1.1)$$

where  $\mathbb{E}_\pi$  denotes the expectation with respect to the trajectory distribution following policy  $\pi$ , the second equivalence in (1.1) follows from Assumption 2. Similarly, the state-action value function (better known as the Q-function) is defined as the expected cumulative rewards from taking an action  $a$  in a state  $s$  and following the given policy  $\pi$ ,

$$Q^\pi(s, a) = \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} \gamma^t R_{t+1} \middle| S_0 = s, A_0 = a \right] = \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} \gamma^t \cdot r(S_t, A_t) \middle| S_0 = s, A_0 = a \right]. \quad (1.2)$$

Under Assumption 1 and 2,  $Q^\pi$  satisfies the Bellman equation,

$$Q^\pi(s, a) = r(s, a) + \gamma \cdot \mathbb{E}_{S' \sim p(\cdot | s, a)} [V^\pi(S')] = r(s, a) + \gamma \cdot \mathbb{E}_{S' \sim p(\cdot | s, a), A' \sim \pi(\cdot | S')} [Q^\pi(S', A')]. \quad (1.3)$$

This equation plays a critical role in estimating the Q-function in many RL algorithms. The policy value for  $\pi$  is defined as an integral of state values over a reference distribution  $\mathbb{G}$ ,

$$V^\pi(\mathbb{G}) = \mathbb{E}_{s \sim \mathbb{G}} [V^\pi(s)] = \int_{s \in \mathcal{S}} V^\pi(s) \mathbb{G}(ds). \quad (1.4)$$

The reference distribution  $\mathbb{G}$  specifies the state distribution on which the policy is evaluated, and is typically set to be the initial state distribution  $p_0$ . For simplicity, we assume  $\mathbb{G}$  to be  $p_0$  for the remainder of the discussion. The integrated value  $V^\pi(\mathbb{G})$  quantifies the overall performance of a policy and thus is the focus of policy evaluation.

Another way to express the policy value is through the marginal density of state-action pairs. For any  $t \geq 0$ , define  $p_t^\pi(s)$  as the marginal density of  $S_t \in \mathcal{S}$  under the target policy  $\pi$  and the reference distribution  $\mathbb{G}$ . With  $p_t^\pi(s)$ , the normalized discounted visitation probability density, or occupancy measure, is defined as

$$d_\pi(s, a) = (1 - \gamma) \cdot \pi(a | s) \sum_{t=0}^{\infty} \gamma^t \cdot p_t^\pi(s). \quad (1.5)$$

It can be shown that there is a one-to-one correspondence between  $\pi$  and  $d_\pi(s, a)$ : given

$d_\pi(s, a)$ , the policy can be recovered as  $\pi(a|s) = d_\pi(s, a) / \sum_{a' \in \mathcal{A}} d_\pi(s, a')$ , and  $\pi$  is the only policy whose occupancy measure is  $d_\pi$  (Syed et al. 2008). Based on the definition of  $d_\pi(s, a)$ , the policy value can be equivalently reformulated as an expectation over state-action pairs with density  $d_\pi(s, a)$ ,

$$V^\pi(\mathbb{G}) = \mathbb{E}_{s \sim \mathbb{G}} [V^\pi(s)] = \mathbb{E}_\pi \left[ \sum_{t=0}^{\infty} \gamma^t \cdot r(S_t, A_t) \right] = \frac{1}{1-\gamma} \mathbb{E}_{d_\pi} [r(S_t, A_t)]. \quad (1.6)$$

The term  $1/(1-\gamma)$  appears at the front because the term  $d_\pi(s, a)$  is normalized with  $(1-\gamma)$  by definition. Note that the value expression in (1.4) leads to the direct method of OPE while (1.6) leads to the marginalized importance sampling method of OPE. We will provide an overview of existing OPE methods in Section 2.2.

In standard RL, the policy evaluation step is followed by a policy improvement step, where the optimal policy  $\pi^*$  is derived by maximizing the policy value:  $\pi^* = \operatorname{argmax}_{\pi \in \Pi} V^\pi(\mathbb{G})$ .

## 1.4 Overview

The rest of the dissertation is organized as follows. In Chapter 2, we focus on Off-Policy Evaluation (OPE), a critical step in offline RL that evaluates policy performance using off-policy data. We investigate OPE under monotone missingness and establish the key condition for the complete-case value estimators to remain valid. To address potential bias under nonignorable missingness, we propose a novel semiparametric IPW value estimator that is shown to be consistent, and we also provide the confidence interval for the proposed estimator. In Chapter 3, our focus shifts to the problem of learning a data-driven reward function using the Inverse Reinforcement Learning (IRL) approach. We introduce a novel offline IRL method that is accurate, fast, and stable. Furthermore, we present a variant of our method that incorporates reward distillation to handle heterogeneity in expert demonstrations. For each chapter, we conduct extensive simulation studies alongside a real data application on MIMIC-III data to demonstrate the effectiveness of our proposed methods.

# Chapter 2

## Off-Policy Evaluation with Nonignorable Missing Data

### 2.1 Introduction

Reinforcement learning (RL) has demonstrated many successes in various domains ranging from game playing (Mnih et al. 2013; Silver et al. 2016) to robotic control (Kober et al. 2013). These successes often rely on simulators to collect large amounts of interaction data for RL training. However, one usually does not have easy access to the environment in real-world applications. Furthermore, deploying a new policy in the environment to evaluate its performance is sometimes infeasible, especially in safety-sensitive domains such as healthcare and autonomous driving. To make real-world RL more practical, pre-collected datasets have become readily available for offline RL. For example, the widespread adoption of Electronic Health Records (EHR) has paved the way for the potential application of offline RL in healthcare, and recent years have seen many efforts toward investigating optimal treatments to assist clinical decision-making (Prasad et al. 2017; Raghu et al. 2017; Wang et al. 2018).

Off-Policy Evaluation (OPE) is a critical step in offline RL to estimate the value of a target policy using offline samples obtained from potentially different policies. In practice, the offline data is often subject to missingness. For example, the sepsis data from the MIMIC-III database (Komorowski et al. 2018) exhibits selection bias attributed to missing data: some patients have shorter trajectories due to early mortality. As shown in Figure 2.1, the average Sepsis-related Organ Failure Assessment (SOFA) score (Vincent et al. 1996) for deceased patients in ICU is higher than that of the remaining patients in the dataset. A higher SOFA score is typically associated with a more severe condition, indicating that patients with shorter trajectories due to mortality in the ICU are generally in worse conditions. When performing

policy evaluation using this dataset, such monotone missingness may lead to biased results.

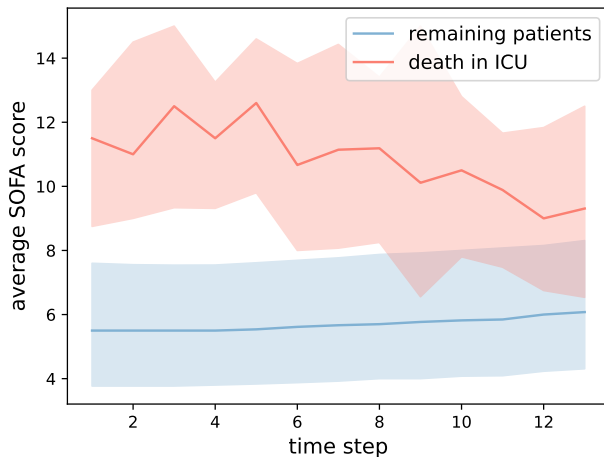


Figure 2.1: The average SOFA scores for deceased patients (red) and patients who survived (blue) during ICU stay. The shadow represents the 25% to 75% quantile.

Although OPE has been extensively studied in the literature (Jiang and Li 2016; Le et al. 2019; Liu et al. 2018; Nachum et al. 2019), OPE with incomplete data is rarely investigated. In this work, we study OPE in the presence of missing data and consider two major missing data mechanisms: ignorable and nonignorable (informative) missingness. Previous research by Goldberg and Kosorok (2012) explored multi-stage decision problems with survival time as rewards that are subject to censoring, and Dong et al. (2020) investigated general optimal treatment regimes under ignorable missingness. Both works applied the Inverse Probability Weighting (IPW) approach to address bias, but these methods relied on backward recursion and hence are vulnerable to model misspecification as the horizon grows. To handle longer horizons, we consider infinite-horizon settings under the Markov Decision Process (MDP) framework (Puterman 1994). Moreover, unlike ignorable missingness, nonignorable missingness is rarely considered in the RL or OPE literature. In this work, we study OPE under monotone missing data and consider both ignorable missingness and nonignorable (informative) missingness. We theoretically demonstrate that the complete-case value estimator remains valid under ignorable missingness but becomes biased when the missing mechanism is nonignorable (informative). To mitigate the bias, we propose a novel semiparametric Inverse Probability Weighted (IPW) value estimator that is shown to be consistent under nonignorable missingness. Furthermore, we provide the associated confidence interval for the proposed estimator to quantify the uncertainty in value estimation. The

effectiveness of the proposed method is empirically demonstrated through a simulation study and a real-world application to EHR data.

We highlight our contributions as follows:

- To the best of our knowledge, we are the first to identify the key condition for the complete-case OPE result to remain valid. In terms of practical implications, our work justifies the application of OPE methods in the presence of missing data when certain conditions are met.
- We bridge the gap in the literature by investigating nonignorable missingness in OPE problems. We theoretically demonstrate that the complete-case value estimator is biased in such cases.
- We introduce a novel semiparametric IPW value estimator that is consistent under nonignorable missingness. Additionally, we provide the associated confidence interval for this estimator.

The rest of the chapter is organized as follows. In Section 2.2, we review the three major classes of model-free OPE methods in the infinite-horizon setting. In Section 2.3, we investigate OPE under incomplete data with monotone missingness and identify the condition for preserving the validity of complete-case OPE results. To mitigate potential bias under nonignorable missingness, we propose a semiparametric IPW value estimator. In Section 2.4, we discuss the generalizability of the proposed framework to more general dropout patterns and a broader class of OPE methods. Simulation studies and a real data application are presented in Sections 2.5 and 2.6. We summarize this work and discuss potential future directions in Section 2.7.

## 2.2 An Overview of Off-Policy Evaluation Methods

Existing OPE algorithms can be categorized into three categories. The first category is the Direct Method (DM), where the policy value is estimated by directly learning the value function or Q-function via model-free function approximation. The definitions of value function  $V^\pi(s)$  and Q-function  $Q^\pi(s, a)$  are given in equations (1.1) and (1.2) respectively. After obtaining the estimator for either function, the policy value can be calculated as

$$V^\pi(\mathbb{G}) = \mathbb{E}_{s \sim \mathbb{G}} [V^\pi(s)] = \mathbb{E}_{s \sim \mathbb{G}, a \sim \pi(\cdot|s)} [Q^\pi(s, a)].$$

The key to estimating the value function or Q-function lies in the following equations:

$$\mathbb{E}_{A_t \sim \pi(\cdot|S_t), S_{t+1} \sim p(\cdot|S_t, A_t)} \left\{ R_{t+1} + \gamma V^\pi(S_{t+1}) - V^\pi(S_t) \middle| S_t \right\} = 0, \quad (2.1)$$

$$\mathbb{E}_{S_{t+1} \sim p(\cdot|S_t, A_t)} \left\{ R_{t+1} + \gamma \mathbb{E}_{a' \sim \pi(\cdot|S_{t+1})} Q^\pi(S_{t+1}, a') - Q^\pi(S_t, A_t) \middle| S_t, A_t \right\} = 0, \quad (2.2)$$

which follow from the Bellman equation (1.3) and Assumption 2. Several OPE methods have been developed based on these equations. LSTD (Bradtke and Barto 1996) estimates the value function using equation (2.1). However, this method requires the action  $A_t$  to be sampled from the policy  $\pi$ , while the actual data is often collected under a different behavior policy  $\pi_b$ . To address the distribution mismatch between the target policy and the behavior policy, V-Learning (Luckett et al. 2019) incorporates an importance sampling term  $\pi(A_t|S_t)/\pi_b(A_t|S_t)$ . However, this method relies on the correct specification of the behavior policy  $\pi_b(a|s)$ , which is not easily available from the observational data. An alternative approach is to estimate the Q-function instead. Such an approach does not impose any restrictions on the behavior policy and also does not require estimating the behavior policy. Based on equation (2.2), LSTDQ (Lagoudakis and Parr 2003; Shi et al. 2021b) approximates  $Q^\pi(s, a)$  using basis functions, which offers an analytical solution because of the linear formulation. To accommodate a more flexible function class for the Q-function, FQE (Le et al. 2019) learns the Q-function by iteratively minimizing the mean squared Bellman residuals.

The second category is the Importance Sampling-based (IS) method, which re-weights the observed rewards to correct the mismatch of data distributions under the target policy and the behavior policy. Recall that the policy value can be equivalently expressed as

$$V^\pi(\mathbb{G}) = \frac{1}{1-\gamma} \mathbb{E}_{(S_t, A_t) \sim d_\pi} \{R_{t+1}\} = \frac{1}{1-\gamma} \mathbb{E}_{(S_t, A_t) \sim d_\pi} \{r(S_t, A_t)\},$$

where  $d^\pi(s, a)$  is the normalized discounted visitation probability density defined in (1.5). In off-policy settings, the data is collected from potentially different policies than the target policy  $\pi$ . Denote the state-action visitation probability density in the observed data as  $d_{\mathcal{D}}$ . One can rewrite  $V^\pi(\mathbb{G})$  as

$$V^\pi(\mathbb{G}) = \frac{1}{1-\gamma} \mathbb{E}_{(S_t, A_t) \sim d_{\mathcal{D}}} \left\{ \frac{d_\pi(S_t, A_t)}{d_{\mathcal{D}}(S_t, A_t)} \cdot R_{t+1} \right\} = \frac{1}{1-\gamma} \mathbb{E}_{(S_t, A_t) \sim d_{\mathcal{D}}} \{\omega_\pi(S_t, A_t) \cdot R_{t+1}\}, \quad (2.3)$$

where  $\omega_\pi(s, a) := d_\pi(s, a)/d_{\mathcal{D}}(s, a)$  is called the marginalized state-action density ratio. Similarly, the marginalized state density ratio can be defined as  $\omega_\pi(s) = d_\pi(s)/d_{\mathcal{D}}(s)$ . Compared with trajectory-based importance sampling methods (Precup 2000; Rubinstein 1981; Hester-

berg 1988), such a marginalized density ratio plays a crucial role in breaking the curse of horizon (Liu et al. 2018). To estimate  $\omega_\pi(s, a)$ , it is important to notice that  $d_\pi(s, a)$  satisfies the backward Bellman recursion

$$d_\pi(s, a) = (1 - \gamma)\mathbb{G}(s)\pi(a|s) + \gamma \cdot \pi(a|s) \int_{\tilde{s} \in \mathcal{S}} \sum_{\tilde{a} \in \mathcal{A}} d_\pi(\tilde{s}, \tilde{a}) p(s|\tilde{s}, \tilde{a}) d\tilde{s}. \quad (2.4)$$

By integrating any function  $f(s, a)$  over the probability density on both sides of (2.4) and then rewriting the expectation over  $d_\pi(s, a)$  as an expectation over  $d_{\mathcal{D}}(s, a)$  with importance sampling weight  $\omega_\pi(s, a)$ , we obtain the following equation

$$\begin{aligned} \mathbb{E}_{d_{\mathcal{D}}} \{ \omega_\pi(S_t, A_t) (f(S_t, A_t) - \gamma \mathbb{E}_{S_{t+1} \sim p(\cdot|S_t, A_t), a' \sim \pi(\cdot|S_{t+1})} [f(S_{t+1}, a')]) \} \\ = (1 - \gamma) \mathbb{E}_{S_0 \sim \mathbb{G}, a \sim \pi(\cdot|S_0)} \{ f(S_0, a) \}. \end{aligned} \quad (2.5)$$

The density ratio  $\omega_\pi(s, a)$  can be estimated by minimizing the difference between the two sides of equation (2.5) over a class of function  $f : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ . This idea gives rise to several Marginalized Importance Sampling (MIS) approaches. DICE (Liu et al. 2018) models the marginalized state density ratio  $\omega_\pi(s)$  and expresses the importance sampling weight as  $\omega_\pi(s, a) = \omega_\pi(s) \cdot \frac{\pi(a|s)}{\pi_b(a|s)}$ . This approach requires a known behavior policy  $\pi_b$  or a model to approximate it accurately, which is sometimes infeasible in practice. To handle unknown behavior policies, a better way is to model the density ratio  $\omega_\pi(s, a)$  directly. Several methods have been proposed for this purpose, including DaulDICE (Nachum et al. 2019), GenDICE (Zhang et al. 2020a), and MWL (Uehara et al. 2020).

The last category is the Doubly Robust (DR) method, which combines DM and MIS methods for more robust and efficient value evaluation. Specifically, Kallus and Uehara (2022) introduces the doubly robust value estimator based on the following expression of policy value

$$\begin{aligned} V_{\text{DR}}^\pi(\mathbb{G}) = \frac{1}{1 - \gamma} \mathbb{E}_{d_{\mathcal{D}}} \{ \omega_\pi(S_t, A_t) (R_{t+1} + \gamma \cdot \mathbb{E}_{a \sim \pi(\cdot|S_{t+1})} Q^\pi(S_{t+1}, a) - Q^\pi(S_t, A_t)) \} \\ + \mathbb{E}_{S_0 \sim \mathbb{G}, a \sim \pi(\cdot|S_0)} \{ Q^\pi(S_0, a) \}. \end{aligned}$$

Then  $\widehat{V}_{\text{DR}}^\pi(\mathbb{G})$  can be calculated by plugging in the estimator for Q-function  $\widehat{Q}^\pi$  obtained from direct methods and marginalized state-action density ratio  $\widehat{\omega}_\pi$  obtained from MIS methods. The term ‘‘doubly robust’’ refers to the property that when either  $\widehat{Q}^\pi$  or  $\widehat{\omega}_\pi$  is consistent,  $\widehat{V}_{\text{DR}}^\pi(\mathbb{G})$  is also consistent, offering two chances to ensure consistency in the estimation process.

In addition to obtaining point estimates of value, many applications would benefit from quantifying the level of uncertainty in the OPE estimates. This type of OPE method is referred to as High-Confidence Off-Policy Evaluation (HCOPE). Dai et al. (2020) estimated the value

confidence interval (CI) using the empirical likelihood approach under the assumption of i.i.d. transitions, which is often violated in practice (Shi et al. 2021a). Recently, Lockett et al. (2019) and Shi et al. (2021b) derive the value CI based on asymptotic distribution, which holds for  $\beta$ -mixing data (Bradley 2005).

## 2.3 Off-Policy Evaluation with Incomplete Data

In this section, we discuss OPE under incomplete data with monotone missingness. Such a missing pattern often occurs when some subjects drop out of the study before the end of the follow-up time window.

Let  $\mathcal{D} = \{\tau_i\}_{1 \leq i \leq n}$  denote the observed data consisting of  $n$  independent and identically distributed trajectories, where each trajectory  $\tau_i = \{(S_{i,t}, A_{i,t}, R_{i,t+1}, S_{i,t+1})\}_{0 \leq t < T_i}$  terminates at time  $T_i$ , and the immediate rewards are uniformly bounded. For simplicity, we assume that all trajectories have the same number of observed time steps, i.e.,  $T_i = T$  for  $i = 1, \dots, n$ . Let  $\boldsymbol{\eta} = (\eta_0, \eta_1, \eta_2, \dots, \eta_T)^\top$  denote the vector of binary response indicators.  $\eta_{i,t}$  is a sample of  $\eta_t$  that represents the response indicator for subject  $i$  at time  $t$ :  $\eta_{i,t} = 1$  if subject  $i$  is still in the study at time  $t$  and we observe the corresponding data  $(R_{i,t}, S_{i,t}, A_{i,t})$ , otherwise  $\eta_{i,t} = 0$ . Assume the baseline covariates and initial treatment assignment are always observable, i.e.,  $\eta_{i,0} = 1$ . We consider a general setting where the reward  $R_{t+1}$  depends on  $(S_t, A_t, S_{t+1})$ : if  $S_{t+1}$  is unobserved,  $R_{t+1}$  is also missing. Therefore, a trajectory can be represented as  $\tau_i = \{(\eta_{i,t}S_{i,t}, \eta_{i,t}A_{i,t}, \eta_{i,t+1}R_{i,t+1}, \eta_{i,t+1}S_{i,t+1})\}_{0 \leq t < T}$ . Under monotone missingness,  $\boldsymbol{\eta}_i$  is a decreasing sequence: if  $\eta_{i,t} = 0$ , then  $\eta_{i,s} = 0$  for all  $s > t$ . To describe the lengths of observed trajectories, we also define the dropout time  $C$ :  $C = t$  if the subject dropped out right after action  $A_t$ , which corresponds to  $(\eta_t, \eta_{t+1}) = (1, 0)$ . If the trajectory is fully observed,  $C$  is set to  $T$ . Given the offline data  $\mathcal{D}$ , our goal is to estimate the value of target policy  $\pi$ .

**Remark 3.** *In this section, we focus on a particular dropout pattern where the dropout occurs after observing an action but before observing the next state. Nevertheless, the proposed framework and theoretical results apply to more general dropout patterns; see Section 2.4.1 for more discussions.*

### 2.3.1 Missing Data Mechanism

There are two major types of missing data mechanisms: ignorable and nonignorable missingness. Ignorable missingness refers to the case where the missingness can be fully explained by the observed information, which is also referred to as Missing-At-Random (MAR). An example of MAR would be dropout in a clinical trial due to recorded side effects and lack

of efficacy, or other known baseline characteristics. The term “randomness” in MAR implies that once one has conditioned on all the available data, any remaining missingness is completely random (Graham et al. 2009). On the other hand, if the missingness depends on unobserved components, the missing data mechanism is referred to as nonignorable, or Missing-Not-At-Random (MNAR). Dropout in a clinical trial due to the unobserved current health status or other unrecorded factors is an example of an MNAR.

We give the formal definition of the two mechanisms under our MDP framework as follows.

**Definition 1** (Ignorable Missingness, MAR). *The missingness can be fully accounted for by the observed information, that is,*

$$\eta_{t+1} \perp\!\!\!\perp (R_{t+1}, S_{t+1}) \mid (S_t, A_t, \{(S_j, A_j, R_{j+1})\}_{0 \leq j < t}, \eta_t)$$

for  $t = 0, \dots, T - 1$ . Here  $\perp\!\!\!\perp$  means independence.

**Definition 2** (Nonignorable Missingness, MNAR). *The missingness depends on the next state regardless of whether it is observed or not, that is,*

$$\eta_{t+1} \not\perp\!\!\!\perp (R_{t+1}, S_{t+1}) \mid (S_t, A_t, \{(S_j, A_j, R_{j+1})\}_{0 \leq j < t}, \eta_t)$$

for  $t = 0, \dots, T - 1$ .

**Remark 4.** *In the special case where  $S_{t+1}$  and  $R_{t+1}$  are fully determined by  $S_t$  and  $A_t$  (e.g., deterministic dynamics), nonignorable missingness reduces to ignorable missingness. In practice, we cannot evaluate whether the missing data mechanism is ignorable or nonignorable from the observed data alone, the missing mechanism needs to be justified based on the context and subject-matter knowledge.*

For nonignorable missingness in longitudinal data, it is common to also assume that the dropout right after time  $t$  is conditionally independent of future observations after time  $t + 1$  (Diggle and Kenward 1994). Thus, the dropout propensity can be expressed as  $P(C = t \mid \{(S_j, A_j, R_{j+1}, S_{j+1})\}_{0 \leq j < t+1}, C \geq t) = P(\eta_{t+1} = 0 \mid \{(S_j, A_j, R_{j+1}, S_{j+1})\}_{0 \leq j < t+1}, \eta_t = 1)$ , which is the probability of the subject dropping out right after time  $t$ . We impose the following assumption regarding the dropout propensity.

**Assumption 3.** *The dropout propensity satisfies  $P(\eta_{t+1} = 0 \mid S_t = s_t, A_t = a_t, R_{t+1} = r_{t+1}, S_{t+1} = s_{t+1}, \{(S_j, A_j, R_{j+1})\}_{0 \leq j < t}, \eta_t = 1) = P(\eta_{t+1} = 0 \mid S_t = s_t, A_t = a_t, R_{t+1} = r_{t+1}, S_{t+1} = s_{t+1}, \eta_t = 1) := \lambda(s_t, a_t, r_{t+1}, s_{t+1})$ , where  $\lambda$  is the dropout propensity function.*

**Remark 5.** *This assumption states that whether a subject will drop out right after receiving  $A_t$  depends on the history only through the current state-action pair  $(S_t, A_t)$ . Moreover,*

such dependency is stationary over time. The assumption shares some similarities with the Markovian assumption. Therefore, if one suspects the missing probability depends on several past steps, one can instead aggregate that information into the state variable.

### 2.3.2 Value Inference under Missing Data

We use the value inference method of Shi et al. (2021b) as the base algorithm, which utilizes linear sieves to approximate the Q-function and estimates the parameters based on the Bellman equation. The linear formulation leads to an explicit expression for the parameter estimator as well as the policy value. Its simplicity, along with strong theoretical guarantees, makes it well-suited for studying further theoretical properties. This approach is an extension of Least-Square Temporal Difference Q (LSTDQ) (Lagoudakis and Parr 2003) and is also a special case of Minimax Q-Function Learning (MQL) (Uehara et al. 2020). Because of its close connection to LSTDQ, we still refer to it as LSTDQ without loss of generality.

Specifically, the Q-function is approximated with linear sieves as  $Q^\pi(s, a) \approx \Phi_L^\top(s) \boldsymbol{\beta}_{\pi, a}$ , where  $\Phi_L(\cdot) = \{\phi_{L,1}(\cdot), \dots, \phi_{L,L}(\cdot)\}^\top$  denotes a vector of  $L$  sieve basis functions, one can use splines (De Boor 1976) or wavelet basis (Huang et al. 1998). The number of basis functions  $L$  is allowed to grow with the sample size to reduce the approximation error. Let  $\boldsymbol{\beta}_\pi = (\boldsymbol{\beta}_{\pi,1}, \dots, \boldsymbol{\beta}_{\pi,m})^\top \in \mathbb{R}^{mL}$ , then the Q-function can be expressed as  $Q^\pi(s, a) = \boldsymbol{\xi}(s, a)^\top \boldsymbol{\beta}_\pi$ , where  $\boldsymbol{\xi}(s, a) = \{\Phi_L^\top(s) \mathbb{1}(a = 1), \dots, \Phi_L^\top(s) \mathbb{1}(a = m)\}^\top$ . The corresponding value function is given by  $V^\pi(s) = \mathbb{E}_{a \sim \pi(\cdot|s)} [Q^\pi(s, a)] = \mathbf{U}_\pi(s)^\top \boldsymbol{\beta}_\pi$ , where  $\mathbf{U}_\pi(s) = \{\Phi_L^\top(s) \pi(1|s), \dots, \Phi_L^\top(s) \pi(m|s)\}^\top$ . By Bellman equation and Assumption 2, we have

$$\mathbb{E} \{R_{t+1} + \gamma V^\pi(S_{t+1}) - Q^\pi(S_t, A_t) | S_t, A_t\} = 0. \quad (2.6)$$

Replace  $Q^\pi$ ,  $V^\pi$  with the linear expression and use  $\boldsymbol{\xi}_t$ ,  $\mathbf{U}_{\pi,t}$  to represent  $\boldsymbol{\xi}(S_t, A_t)$ ,  $\mathbf{U}_\pi(S_t)$ , we obtain  $\mathbb{E} [\mathbf{M}_t(\boldsymbol{\beta}_\pi)] = \mathbf{0}$ , where

$$\mathbf{M}_t(\boldsymbol{\beta}_\pi) = \boldsymbol{\xi}_t \{R_{t+1} + \gamma V^\pi(S_{t+1}) - Q^\pi(S_t, A_t)\} = \boldsymbol{\xi}_t \{R_{t+1} - (\boldsymbol{\xi}_t - \gamma \mathbf{U}_{\pi,t+1})^\top \boldsymbol{\beta}_\pi\}.$$

The true parameter  $\boldsymbol{\beta}_\pi^*$  can be estimated by solving the estimating equations  $\mathbb{E}_{nT} [\mathbf{M}_t(\boldsymbol{\beta}_\pi)] = \mathbf{0}$ , where  $\mathbb{E}_{nT}[\cdot]$  denotes the empirical average over  $nT$  transition pairs  $(S_{i,t}, A_{i,t}, R_{i,t+1}, S_{i,t+1})$ . The problem reduces to a linear regression.

When the data is incomplete, the response indicator  $\eta_t$  needs to be taken into account. One approach is to estimate the parameters using only the observed data, which is known as the Complete-Case (CC) estimator. In general, the CC estimator refers to the value estimator using standard OPE methods without any adjustment for missingness. In our discussion, it

corresponds to the base algorithm proposed by Shi et al. (2021b) without applying any adjustment. The corresponding estimating equation becomes  $\mathbb{E}_{nT} \{\mathbf{M}_t(\boldsymbol{\beta}_\pi) \mid \eta_{t+1} = 1\} = \mathbf{0}$ . Here, the expectation is conditioned on  $\eta_{t+1} = 1$  because  $\mathbf{M}_t(\boldsymbol{\beta}_\pi)$  involves the next state and reward: if  $\eta_{t+1} = 0$ , then  $(R_{t+1}, S_{t+1})$  is unobserved, and the transition  $(S_t, A_t, R_{t+1}, S_{t+1})$  does not contribute to the estimation of  $\boldsymbol{\beta}_\pi$ . Note that  $\mathbb{E}_{nT} \{\mathbf{M}_t(\boldsymbol{\beta}_\pi) \mid \eta_{t+1} = 1\} = \mathbb{E}_{nT} \{\eta_{t+1} \mathbf{M}_t(\boldsymbol{\beta}_\pi)\}$ . Therefore, given the observed data  $\mathcal{D}$ , the complete-case estimator of  $\boldsymbol{\beta}_\pi$  is given as follows

$$\widehat{\boldsymbol{\beta}}_{\pi, \text{CC}} = \underbrace{\left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top \right\}^{-1}}_{\widehat{\boldsymbol{\Sigma}}_{\pi, \text{CC}}} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} R_{i,t+1} \right) \quad (2.7)$$

The estimators for the Q-function and value function are given by  $\widehat{Q}_{\text{CC}}^\pi(s, a) = \boldsymbol{\xi}^\top(s, a) \widehat{\boldsymbol{\beta}}_{\pi, \text{CC}}$ ,  $\widehat{V}_{\text{CC}}^\pi(s) = \mathbf{U}_\pi^\top(s) \widehat{\boldsymbol{\beta}}_{\pi, \text{CC}}$ , respectively. Given a reference distribution  $\mathbb{G}$  on state space  $\mathcal{S}$ , the policy value can be estimated as

$$\widehat{V}_{\text{CC}}^\pi(\mathbb{G}) = \int_{s \in \mathcal{S}} \widehat{V}_{\text{CC}}^\pi(s) \mathbb{G}(ds) = \left\{ \int_{s \in \mathcal{S}} \mathbf{U}_\pi(s) \mathbb{G}(ds) \right\}^\top \widehat{\boldsymbol{\beta}}_{\pi, \text{CC}}.$$

In practice, the integration  $\int_{s \in \mathcal{S}} \mathbf{U}_\pi(s) \mathbb{G}(ds)$  can be approximated with a sample average of  $\mathbf{U}_\pi(s)$  over the reference distribution  $\mathbb{G}$ .

When there are no missing data, the value estimator is shown to be asymptotically normal as either  $n \rightarrow \infty$  or  $T \rightarrow \infty$  under some mild conditions (see Theorem 1 of Shi et al. (2021b)). However, when the data is incomplete, the solution to  $\mathbb{E} \{\eta_{t+1} \mathbf{M}_t(\boldsymbol{\beta}_\pi)\} = \mathbf{0}$  may differ from the solution to  $\mathbb{E}_{nT} \{\mathbf{M}_t(\boldsymbol{\beta}_\pi)\} = \mathbf{0}$ . The following theorem outlines when the complete-case value estimator remains valid and when it may not.

**Theorem 1.** *Suppose Assumption 1-4 holds.  $\widehat{V}_{\text{CC}}^\pi(\mathbb{G})$  is a consistent estimator if the missing mechanism is ignorable (MAR). However, if the missing mechanism is nonignorable (MNAR),  $\widehat{V}_{\text{CC}}^\pi(\mathbb{G})$  can be biased.*

Assumption 1-3 have been previously discussed. Assumption 4 encompasses the necessary conditions that ensure the consistency and asymptotic distribution of value estimation when there is no missing data. The complete proof of Theorem 1 can be found in Appendix A.5.1. Here, we sketch the big idea behind the proof. The key to consistency under MAR is the conditional independence between  $\eta_{t+1}$  and  $\mathbf{M}_t(\boldsymbol{\beta}_\pi)$  given  $(S_t, A_t, \eta_t)$ , which allows the two terms to be separated. Specifically,  $\mathbb{E} \{\eta_{t+1} \mathbf{M}_t(\boldsymbol{\beta}_\pi)\} = \mathbb{E} \{\mathbb{E}(\eta_{t+1} \mid S_t, A_t, \eta_t) \mathbb{E}(\mathbf{M}_t(\boldsymbol{\beta}_\pi) \mid S_t, A_t)\}$ . Let  $\boldsymbol{\beta}_\pi^*$  denote the true parameter. It follows from (2.6) that  $\mathbb{E}\{\mathbf{M}_t(\boldsymbol{\beta}_\pi^*) \mid S_t, A_t\} = \mathbf{0}$ . Consequently,  $\mathbb{E} \{\eta_{t+1} \mathbf{M}_t(\boldsymbol{\beta}_\pi^*)\} = \mathbf{0}$ , indicating that  $\boldsymbol{\beta}_\pi^*$  is still the solution to  $\mathbb{E} \{\eta_{t+1} \mathbf{M}_t(\boldsymbol{\beta}_\pi)\} = \mathbf{0}$ .

As a result, the corresponding value estimator remains consistent. However, when the missingness is nonignorable,  $\eta_{t+1}$  and  $\mathbf{M}_t(\boldsymbol{\beta}_\pi)$  are no longer conditionally independent. Consequently,  $\mathbb{E}\{\eta_{t+1}\mathbf{M}_t(\boldsymbol{\beta}_\pi^*)\} = \mathbf{0}$  does not hold.

To the best of our knowledge, this is the first result establishing the validity of the OPE method in the presence of missing data. As per Theorem 1, the complete-case value estimator remains valid if the missing data mechanism is ignorable. However, for nonignorable missingness, further adjustments are required to retrieve consistency.

### 2.3.3 Value Inference with Inverse Weights

To address the bias in value estimation under nonignorable missingness, we adopt the Inverse Probability Weighting (IPW) approach that is widely used in the missing data literature. Consider the IPW estimating equation as follows

$$\mathbb{E}_{nT} \left\{ \frac{\eta_{t+1}}{1 - \lambda(S_t, A_t, R_{t+1}, S_{t+1}; \boldsymbol{\psi})} \mathbf{M}_t(\boldsymbol{\beta}_\pi) \right\} = \mathbf{0},$$

where  $\lambda(S_t, A_t, R_{t+1}, S_{t+1}; \boldsymbol{\psi})$  is the dropout propensity model parameterized by  $\boldsymbol{\psi}$ .

To solve for  $\boldsymbol{\beta}_\pi$ , the first step is to fit the dropout propensity model. For ignorable missingness, the dropout propensity function can be simplified to  $\lambda(S_t, A_t)$  since  $\eta_{t+1}$  is conditionally independent of  $S_{t+1}$  and  $R_{t+1}$ . In such cases, the propensity can be modeled with any binary classification method. However, unlike ignorable missingness, modeling the nonignorable missingness is much more challenging. The difficulty lies in that if both the dropout propensity  $\lambda(S_t, A_t, R_{t+1}, S_{t+1})$  and the conditional density function  $f(R_{t+1}, S_{t+1}|S_t, A_t)$  are completely unknown, the joint distribution of  $(\eta_{t+1}, R_{t+1}, S_{t+1})$  given  $(S_t, A_t)$  is non-identifiable (Rotnitzky and Robins 1997). One solution is to posit a model on  $\lambda(S_t, A_t, R_{t+1}, S_{t+1})$  and allow  $f(R_{t+1}, S_{t+1}|S_t, A_t)$  to be unspecified. Inspired by the recent development of the semiparametric framework to model nonignorable missing data (Kim and Yu 2011; Shao and Wang 2016), we consider a semiparametric exponential tilting model for the dropout propensity as

$$\lambda(S_t, A_t, R_{t+1}, S_{t+1}; \boldsymbol{\psi}) = \left\{ 1 + \exp[g(S_t, A_t) + Z_{t+1}^\top \boldsymbol{\psi}] \right\}^{-1},$$

where  $\boldsymbol{\psi} \in \mathbb{R}^q$  is an unknown tilting parameter,  $Z_{t+1} \in \mathbb{R}^q$  are features constructed from  $(R_{t+1}, S_{t+1})$ ,  $g(\cdot)$  is a non-parametric function of observed variables  $(S_t, A_t)$ . For succinctness, we suppress the data arguments in  $\lambda(S_t, A_t, R_{t+1}, S_{t+1}; \boldsymbol{\psi})$  and write it as  $\lambda_{t+1}(\boldsymbol{\psi})$ .

Note that the semiparametric model alone still suffers from non-identifiability issues, as there is only one estimating equation to estimate both  $g$  and  $\boldsymbol{\psi}$ . One way to resolve such non-identifiability issue is by using the instrumental variable (Shao and Wang 2016). An

instrumental variable  $\mathcal{V}_t$  is a covariate in  $(S_t, A_t)$  that is related to the outcome  $(R_{t+1}, S_{t+1})$  but not related to the dropout propensity given other covariates. For instance, in the clinical trial setting, it is reasonable to believe that some baseline measurements do not provide additional information about dropout given the outcomes, hence some baseline measurements prior to the treatments can often serve as instrumental variables. In practice, identifying instrumental variables can be a challenging task and often requires subject-matter knowledge. A discussion on how to find such instrumental variables is provided in Section 6 of Shao and Wang (2016).

Based on the definition of the instrumental variable,  $\mathcal{V}_t$  can be removed from the model. Denote the non-instrumental part of  $(S_t, A_t)$  as  $\mathcal{U}_t$ , the exponential tilting model can be rewritten as

$$\lambda_{t+1}(\boldsymbol{\psi}) = \{1 + \exp[g(\mathcal{U}_t) + \boldsymbol{\psi}^\top Z_{t+1}]\}^{-1}. \quad (2.8)$$

With the instrumental variable, multiple estimating equations can be constructed to estimate the parameters of interest. If the instrumental variable  $\mathcal{V}_t$  is discrete with  $\tilde{L}$  levels, the  $\tilde{L}$  estimating equations can be constructed as

$$\mathbb{E}_{nT} \left\{ \mathbb{1}(\mathcal{V}_t = l) \left( \frac{\eta_{t+1}}{1 - \lambda_{t+1}(\boldsymbol{\psi})} - 1 \right) \right\} = 0, \quad l \in \{1, \dots, \tilde{L}\}.$$

Here we use the notation  $\tilde{L}$  to differentiate it from the notation  $L$ , which represents the number of basis functions. In the case of  $\mathcal{V}_t$  being a continuous variable, it can be first discretized into  $\tilde{L}$  bins. Given the definition of the instrumental variable,  $\mathcal{V}_t$  is conditionally independent of  $\eta_{t+1}$  given  $(\mathcal{U}_t, S_{t+1}, R_{t+1})$ , meanwhile,  $\mathbb{E}(\mathbb{1}(\mathcal{V}_t = l) \mid \mathcal{U}_t, S_{t+1}, R_{t+1})$  is not a constant since  $\mathcal{V}_t$  is related to  $(S_{t+1}, R_{t+1})$ , hence the  $\tilde{L}$  estimating equations will not reduce to a single one. Therefore, this approach effectively addresses the aforementioned non-identifiability issue. To estimate  $\boldsymbol{\psi}$ , the non-parametric component  $g$  is first profiled with a kernel estimator. The remaining  $\tilde{L} - 1$  estimating equations are used to solve for  $\boldsymbol{\psi}$  using the Generalized Method of Moments (GMM) (Hansen 1982). More estimation details are provided in Appendix A.4.

Denote the estimated dropout propensity parameter as  $\widehat{\boldsymbol{\psi}}_{nT}$ . The proposed IPW estimator for  $\boldsymbol{\beta}_\pi$  is given by

$$\widehat{\boldsymbol{\beta}}_{\pi, \text{IPW}} = \underbrace{\left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \frac{\eta_{i,t+1}}{1 - \lambda_{i,t+1}(\widehat{\boldsymbol{\psi}}_{nT})} \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top \right\}^{-1}}_{\widehat{\boldsymbol{\Sigma}}_{\pi, \text{IPW}}} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \frac{\eta_{i,t+1}}{1 - \lambda_{i,t+1}(\widehat{\boldsymbol{\psi}}_{nT})} \boldsymbol{\xi}_{i,t} R_{i,t+1} \right). \quad (2.9)$$

Compare with the complete-case estimator in Equation (2.7), both terms of  $\widehat{\boldsymbol{\beta}}_{\pi, \text{IPW}}$  are multiplied with an inverse weighting term,  $\{1 - \lambda_{i, t+1}(\widehat{\boldsymbol{\psi}}_{nT})\}^{-1}$ . Intuitively, observations with higher dropout propensities are assigned higher weights to adjust the data distribution. The other part of the estimation procedure is similar to the complete-case estimator. The estimator of value function is  $\widehat{V}_{\text{IPW}}^{\pi}(s) = \mathbf{U}_{\pi}^{\top}(s)\widehat{\boldsymbol{\beta}}_{\pi, \text{IPW}}$ , and the policy value is given by

$$\widehat{V}_{\text{IPW}}^{\pi}(\mathbb{G}) = \int_{s \in \mathcal{S}} \widehat{V}_{\text{IPW}}^{\pi}(s)\mathbb{G}(ds) = \left\{ \int_{s \in \mathcal{S}} \mathbf{U}_{\pi}(s)\mathbb{G}(ds) \right\}^{\top} \widehat{\boldsymbol{\beta}}_{\pi, \text{IPW}}.$$

Theorem 2 and 3 establish the consistency and asymptotic distribution of the proposed estimator under nonignorable missingness.

**Theorem 2** (Bidirectional Consistency). *Assume conditions 1-5 and 6(a)-(d) hold.  $\widehat{V}_{\text{IPW}}^{\pi}(\mathbb{G})$  is a consistent value estimator, that is,  $\widehat{V}_{\text{IPW}}^{\pi}(\mathbb{G}) \xrightarrow{p} V^{\pi}(\mathbb{G})$  as either  $n \rightarrow \infty$  or  $T \rightarrow \infty$ .*

**Theorem 3** (Bidirectional Asymptotics). *Assume conditions 1-6 hold. As either  $n \rightarrow \infty$  or  $T \rightarrow \infty$ , we have*

$$\sqrt{nT}\widehat{\sigma}_{\pi, \text{IPW}}^{-1}(\mathbb{G})\{\widehat{V}_{\text{IPW}}^{\pi}(\mathbb{G}) - V^{\pi}(\mathbb{G})\} \xrightarrow{d} \mathcal{N}(0, 1),$$

where  $\widehat{\sigma}_{\pi, \text{IPW}}^2(\mathbb{G})$  is given by (A.24) in Appendix A.5.3.

Assumption 5 is the boundedness condition imposed on the dropout propensities, while Assumption 6 ensures the consistency and asymptotic normality of the dropout propensity estimation. The proofs for these two theorems can be found in Appendix A.5.2 and A.5.3. Different from the asymptotic result presented in Shi et al. (2021b), we now take into account the response indicator  $\eta_t$  and the uncertainty associated with dropout propensity estimation. Based on Theorem 3, the 2-sided CI for  $V^{\pi}(\mathbb{G})$  with significance level  $\alpha$  can be constructed as

$$\left[ \widehat{V}_{\text{IPW}}^{\pi}(\mathbb{G}) - z_{\alpha/2} \cdot \frac{\widehat{\sigma}_{\pi, \text{IPW}}(\mathbb{G})}{\sqrt{nT}}, \widehat{V}_{\text{IPW}}^{\pi}(\mathbb{G}) + z_{\alpha/2} \cdot \frac{\widehat{\sigma}_{\pi, \text{IPW}}(\mathbb{G})}{\sqrt{nT}} \right],$$

where  $z_{\alpha/2}$  is the  $(1 - \alpha/2)$ -quantile of the standard normal distribution. The original form of  $\widehat{\sigma}_{\pi, \text{IPW}}^2(\mathbb{G})$  has a complicated form. For ease of computation, we suggest using an approximation of  $\widehat{\sigma}_{\pi, \text{IPW}}^2(\mathbb{G})$  given by

$$\widetilde{\sigma}_{\pi, \text{IPW}}^2(\mathbb{G}) = \left\{ \int_{s \in \mathcal{S}} \mathbf{U}_{\pi}(s)\mathbb{G}(ds) \right\}^{\top} \widehat{\boldsymbol{\Sigma}}_{\pi, \text{IPW}}^{-1} \widetilde{\boldsymbol{\Omega}}_{\pi, \text{IPW}} (\widehat{\boldsymbol{\Sigma}}_{\pi, \text{IPW}}^{\top})^{-1} \left\{ \int_{s \in \mathcal{S}} \mathbf{U}_{\pi}(s)\mathbb{G}(ds) \right\}. \quad (2.10)$$

Here,  $\tilde{\Omega}_{\pi, \text{IPW}}$  is an approximation of  $\hat{\Omega}_{\pi, \text{IPW}}$  and can be calculated as follows

$$\tilde{\Omega}_{\pi, \text{IPW}} = \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \left\{ \frac{\eta_{i,t+1}}{1 - \lambda_{i,t+1}(\hat{\boldsymbol{\psi}}_{nT})} \left( R_{i,t+1} + \gamma \hat{V}_{\text{IPW}}^\pi(S_{i,t+1}) - \hat{Q}_{\text{IPW}}^\pi(S_{i,t}, A_{i,t}) \right) \right\}^2.$$

Our experiments suggest that this approximation is close to the result of bootstrapping, so we deploy it in our implementation.

The outline for the proposed estimator is presented in Algorithm 1.

---

**Algorithm 1** Off-Policy Evaluation with Nonignorable Monotone Missingness

---

- 1: **Input:** Observed dataset  $\mathcal{D} = \{\tau_i\}_{i=1}^n$ , target policy  $\pi$ , discount factor  $\gamma$ , number of basis  $L$
  - 2: Fit dropout propensity model (2.8) using the semiparametric approach
  - 3: Construct a set of basis  $\Phi_L(s)$  from state variables and estimate  $\hat{\boldsymbol{\beta}}_{\pi, \text{IPW}}$  by (2.9)
  - 4: Estimate policy value:  $\hat{V}_{\text{IPW}}^\pi(s) = \mathbf{U}_\pi^\top(s) \hat{\boldsymbol{\beta}}_{\pi, \text{IPW}}$ ,  $\hat{V}_{\text{IPW}}^\pi(\mathbb{G}) = \int_{s \in \mathcal{S}} \hat{V}_{\text{IPW}}^\pi(s) \mathbb{G}(ds)$
  - 5: Calculate the approximated asymptotic variance  $\tilde{\sigma}_{\pi, \text{IPW}}^2(\mathbb{G})$  given by (2.10)
  - 6: **Return:** The CI of  $V_{\text{IPW}}^\pi(\mathbb{G})$ :  $\left[ \hat{V}_{\text{IPW}}^\pi(\mathbb{G}) \pm z_{\alpha/2} (nT)^{-1/2} \tilde{\sigma}_{\pi, \text{IPW}}(\mathbb{G}) \right]$
- 

## 2.4 Generalizability of Proposed Framework

In the previous section, we focus on the scenario where dropout occurs after the action is observed but before the reward and the next state are observed. Additionally, we use LSTDQ as the base OPE algorithm to investigate the effect of missing data. In this section, we will expand our scope to more general dropout patterns and other OPE methods.

### 2.4.1 More General Dropout Patterns

The proposed framework is universally applicable to a broader class of dropout patterns. Specifically, the theoretical results for our IPW estimator are valid when dropout occurs after the observed action, regardless of whether the reward is observed or not. This is because the key idea behind the proposed IPW estimator is to assign weights to each transition based on the inverse probability of observing the complete transition quadruple  $(S_t, A_t, R_{t+1}, S_{t+1})$  given observed  $(S_t, A_t)$ . On the other hand, when dropout occurs after an observed state but before an observed action, the proposed framework also applies. The distinction lies in that MAR and MNAR are now defined with respect to  $S_t$  instead of  $(S_t, A_t)$ . If the missingness of the current

action only depends on the current state and not on the action itself, it is considered ignorable. In such cases, the CC estimator remains valid, and no further adjustment is required. This can be seen from the decomposition  $\mathbb{E} \{ \eta_{t+1} \mathbf{M}_t(\boldsymbol{\beta}_\pi^*) \} = \mathbb{E} \{ \mathbb{E}(\eta_{t+1} | S_t, \eta_t) \mathbb{E}(\mathbf{M}_t(\boldsymbol{\beta}_\pi^*) | S_t) \} = \mathbf{0}$ . Here,  $\mathbb{E}(\mathbf{M}_t(\boldsymbol{\beta}_\pi^*) | S_t) = \mathbf{0}$  follows from the law of total probability together with equation (2.6). In the case of nonignorable missingness where the dropout is dependent on the potential action, the CC estimator can be biased, and the proposed IPW estimator can still be used to mitigate such bias.

Moreover, the idea of IPW adjustment can potentially be extended to handle intermittent missingness. The key distinction from monotone missingness lies in estimating the dropout propensity, which should be determined on a case-by-case basis and sometimes requires additional assumptions regarding the missing pattern. We leave this for future investigation.

## 2.4.2 Extension to Other Off-Policy Evaluation Methods

In Section 2.3, we use LSTDQ as the base OPE algorithm. However, one limitation of LSTDQ is that it is only applicable to discrete action spaces and low-dimensional state spaces due to basis approximation. As the dimension grows, the instability of the matrix inversion step will become an issue. Nevertheless, it is worth noting that the IPW adjustment is flexible enough to be potentially combined with other direct methods of OPE. Take Fitted Q Evaluation (FQE) (Le et al. 2019) as an example. The Q-function is approximated with some function class  $\mathcal{Q}$  and iteratively estimated by minimizing the following loss function

$$\mathcal{L}_{nT}(Q^\pi) = \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \left\{ Q^\pi(S_{i,t}, A_{i,t}) - \left( R_{i,t+1} + \gamma \sum_{a \in \mathcal{A}} \pi(a|S_{i,t+1}) Q_{k-1}^\pi(S_{i,t+1}, a) \right) \right\}^2,$$

where  $Q_{k-1}$  is the estimated Q-function obtained from the last iteration. To handle nonignorable missingness, we can incorporate the inverse weighting term into the loss function as follows

$$\begin{aligned} \tilde{\mathcal{L}}_{nT}(Q^\pi) &= \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \frac{\eta_{i,t+1}}{1 - \lambda_{i,t+1}(\widehat{\boldsymbol{\psi}}_{nT})} \times \\ &\quad \left\{ Q^\pi(S_{i,t}, A_{i,t}) - \left( R_{i,t+1} + \gamma \sum_{a \in \mathcal{A}} \pi(a|S_{i,t+1}) Q_{k-1}^\pi(S_{i,t+1}, a) \right) \right\}^2. \end{aligned}$$

Besides direct methods, we can further extend the proposed framework to Marginalized Importance Sampling-based (MIS) methods. Based on equation (2.5), several methods have been developed to estimate the density ratio  $\omega_\pi(s, a)$ , as discussed in Section 2.2. These

methods typically learn  $\omega_\pi(s, a)$  by minimizing the difference between the two sides of the equation (2.5) within the chosen function classes for  $Q^\pi(s, a)$  and  $\omega_\pi(s, a)$ . Denote the function class for  $Q^\pi(s, a)$  as  $\mathcal{Q}$  and the function class for  $\omega_\pi(s, a)$  as  $\Omega$ . To illustrate the estimation process, we use Minimax Weight Learning (MWL) (Uehara et al. 2020) as an example, where  $\omega_\pi$  is estimated by solving  $\hat{\omega}_{\pi, nT}(s, a) = \operatorname{argmin}_{\omega_\pi \in \Omega} \sup_{Q^\pi \in \mathcal{Q}} \mathcal{L}_{nT}(\omega_\pi, Q^\pi)^2$  with  $\mathcal{L}_{nT}(\omega_\pi, Q^\pi)$  defined as follows

$$\begin{aligned} \mathcal{L}_{nT}(\omega_\pi, Q^\pi) = & \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \omega_\pi(S_{i,t}, A_{i,t}) \left( \gamma \sum_{a' \in \mathcal{A}} \pi(a' | S_{i,t+1}) Q^\pi(S_{i,t+1}, a') - Q^\pi(S_{i,t}, A_{i,t}) \right) \\ & + (1 - \gamma) \cdot \mathbb{E}_{S_0 \sim \mathbb{G}} \left\{ \sum_{a \in \mathcal{A}} \pi(a | S_0) Q^\pi(S_0, a) \right\}. \end{aligned} \quad (2.11)$$

The complete-case MIS value estimator can be obtained by plugging in  $\hat{\omega}_{\pi, nT}(s, a)$  as follows,

$$\hat{V}_{\text{CC}}^\pi(\mathbb{G}) = \frac{1}{1 - \gamma} \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \hat{\omega}_{\pi, nT}(S_{i,t}, A_{i,t}) R_{i,t+1}. \quad (2.12)$$

Next, we present the consistency results under the two missingness mechanisms.

**Theorem 4.** *Assume conditions 1-3 and 4(a)(f) hold. Let  $\omega_\pi(s, a)$  denote the true density ratio under missing data and  $\hat{\omega}_\pi(s, a)$  denote the estimated density ratio from the observed data. Further, assume*

- (a) *There exists a constant  $c_\omega > 0$  such that  $\sup_{s,a} |\omega_\pi(s, a)| \leq c_\omega$  and the function class  $\Omega$  satisfies  $\|\omega\|_\infty \leq c_\omega$  for all  $\omega \in \Omega$ .*
- (b)  *$\mathcal{L}_{nT}(\hat{\omega}_\pi, Q^\pi) = o_p(1)$ , where  $Q^\pi$  represents the true Q-function.*

*Under ignorable missingness (MAR), the value estimate (2.12) remains consistent. On the other hand, if the missingness is nonignorable (MNAR), the value estimator (2.12) can be biased.*

In Assumptions (a), the boundedness of marginalized state-action density ratio  $\omega_\pi$  can be ensured if the numerator  $d_\pi$  is bounded above and the denominator  $d_{\mathcal{D}}$  is bounded away from 0. Such an assumption is commonly made in the literature related to importance sampling or inverse weighting. Additionally, the boundedness of function class  $\Omega$  can be guaranteed through a truncation argument. Assumption (b) states that  $\hat{\omega}_\pi$  ensures equation (2.5) approximately holds when substituting  $f(s, a)$  with the true Q-function  $Q^\pi(s, a)$ . This assumption can be achieved when the function class  $\mathcal{Q}$  captures the true Q-function, i.e.,  $Q^\pi \in \mathcal{Q}$ , and the OPE algorithm minimizes  $\sup_{Q^\pi \in \mathcal{Q}} \mathcal{L}_{nT}(\omega_\pi, Q^\pi)^2$  sufficiently close to 0.

The proof for Theorem 4 can be found in Appendix A.5.4. It is noteworthy that the statement in Theorem 4 can also be viewed from a special case of MWL, where  $\omega_\pi(s, a)$  and  $Q^\pi(s, a)$  are modeled with the same set of basis functions, i.e.,  $\omega_\pi(s, a) = \Phi_L(s)^\top \boldsymbol{\alpha}_{\pi, a}$  and  $Q^\pi(s, a) = \Phi_L(s)^\top \boldsymbol{\beta}_{\pi, a}$ . The corresponding value estimator can be shown to be

$$\widehat{V}_{\text{CC}}^\pi(\mathbb{G}) = \left\{ \int_s \mathbf{U}_\pi(s) \mathbb{G}(ds) \right\} \left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi, i, t+1})^\top \right\}^{-1} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} R_{i,t+1} \right).$$

which is identical to the complete-case LSTDQ estimator discussed in Section 2.3.2; see Appendix A.5.6 for a detailed derivation. Consequently, these two estimators share the same theoretical properties described in Theorem 1. In the case of nonignorable missingness, the IPW adjustment discussed in Section 2.3.3 can be applied to this estimator as well.

## 2.5 Simulation Study

We conduct a simulation study to investigate the finite-sample performance of the proposed estimator and the associated confidence interval. We consider the 2D-Linear environment used in Lockett et al. (2019) and Shi et al. (2021b). The environment is characterized by 2-dimensional state variable  $S_t = (S_t^{(1)}, S_t^{(2)})^\top$  and binary action  $A_t \in \{0, 1\}$ . The initial states are generated from the standard bivariate normal distribution  $\mathcal{N}(\mathbf{0}_2, \mathbf{I}_2)$ . For  $t \geq 0$ , we slightly modify the original dynamics and consider the following transition:  $S_{t+1}^{(1)} = (2A_t - 1)S_t^{(1)} + \varepsilon_t^{(1)}$ ,  $S_{t+1}^{(2)} = (1 - 2A_t)S_t^{(2)} + \varepsilon_t^{(2)}$ , where  $\varepsilon_t^{(1)}$  and  $\varepsilon_t^{(2)}$  are independent  $\mathcal{N}(0, 0.25)$  random variables. The immediate reward is  $R_{t+1} = 2S_{t+1}^{(1)} + S_{t+1}^{(2)} + 0.5S_t^{(2)} - 0.25(2A_t - 1) + \varepsilon_t^{(3)}$ , where  $\varepsilon_t^{(3)} \sim \mathcal{N}(0, 10^{-4})$ . The behavior policy follows a Bernoulli distribution with a mean of 0.5. Throughout the simulation studies, the reference distribution  $\mathbb{G}$  is set to the initial state distribution, and the discount factor  $\gamma$  is set to 0.9. We evaluate the following two different target policies:

- (a)  $\pi_1(a = 1|s) = \mathbb{1}\{s^{(1)} + s^{(2)} > 0\}$ , which is a deterministic policy characterized by a discontinuous function with respect to the state;
- (b)  $\pi_2(a = 1|s) = \exp(s^{(1)} + s^{(2)}) / \{1 + \exp(s^{(1)} + s^{(2)})\}$ , which is a stochastic policy characterized by a smooth function of the state.

For each target policy, the true policy values are estimated with 100,000 Monte Carlo approximations. To generate incomplete data, we consider both ignorable (MAR) and nonignorable (MNAR) mechanisms. Assume  $S_t^{(2)}$  is an instrument variable such that it is correlated with

$(R_{t+1}, S_{t+1})$  but uncorrelated with dropout propensity. The MNAR dropout model is constructed as  $\lambda_1(S_t, A_t, R_{t+1}, S_{t+1}) = \{1 + \exp(7 + 0.8S_t^{(1)} - 1.5R_{t+1})\}^{-1}$  and the MAR dropout model is  $\lambda_2(R_t, S_t, A_t) = \{1 + \exp(7 + 0.8S_t^{(1)} - 1.5R_t)\}^{-1}$ . The difference between the two models is that  $\lambda_1$  relies on the next state  $S_{t+1}$  through reward  $R_{t+1}$  while  $\lambda_2$  does not. In this setting, higher reward leads to higher dropout propensity, so the distribution of the observed data is biased towards the low-reward region. More implementation details are reported in Appendix A.2.

To the best of our knowledge, no other method has ever considered OPE under nonignorable missingness, so we mainly focus on the comparison between the IPW and CC estimator. We consider four different combinations of  $n$  and  $T$ : (500, 10), (1000, 10), (500, 25), (1000, 25). For each setting, we run 250 experiments. In each experiment, we generate a new dataset and estimate the value as well as its confidence interval. The Empirical Coverage Probability (ECP) is calculated as the percentage of intervals out of 250 that contain the true value of the target policy. The Average Length (AL) is the average length of the 250 computed intervals.

Tables 2.1 and 2.2 present the complete results of value estimation for both target policies. We observe that the CC estimator under ignorable missingness remains consistent with low bias, which aligns with our theoretical findings. The associated confidence intervals also achieve nominal coverage probability, indicating that no further adjustment is necessary in this scenario. However, when it comes to nonignorable missingness, the CC estimator exhibits high bias, resulting in poor coverage probability of the associated confidence intervals. This under-coverage issue gets worse as the sample size grows. In contrast, the proposed IPW estimator effectively reduces the bias and yields more accurate confidence intervals, which is consistent with our theoretical results. Moreover, we performed a sensitivity analysis in Appendix A.1 to evaluate the robustness of our proposed estimator in case of misspecification of the missing propensity. The analysis revealed that, even if there is model misspecification, the proposed IPW estimator still yields satisfying results, provided that the estimated propensities are approximately accurate.

We also visualize the ECP and AL of the estimated confidence intervals across different confidence levels in Figure 2.2. As depicted in the figure, the confidence intervals obtained by the CC estimator suffer from the poor coverage issue, while the IPW intervals achieve ECPs close to the intended coverage, even though the lengths of the two intervals are very close to each other.

To illustrate the extension of the proposed framework to other OPE methods, we further compare the value estimation results obtained from LSTDQ (Lagoudakis and Parr 2003; Shi et al. 2021b), FQE (Le et al. 2019), and MWL (Uehara et al. 2020). To ensure a fair comparison, we utilize the same set of basis functions to approximate the Q-function and

Table 2.1: Results of value estimates and 95% confidence intervals for policy  $\pi_1$  in the 2D-Linear environment. The average bias, MSE values, ECP, and AL are reported for each estimator (with standard error in parenthesis).

$T$	$n$	Dropout	Method	Bias	MSE	ECP	AL
10	500	no dropout	CC	0.111 (1.762)	3.105	0.960	7.362 (0.353)
		MAR	CC	0.090 (2.253)	5.062	0.960	9.139 (2.131)
		MAR	IPW	-0.007 (2.285)	5.202	0.968	9.404 (3.511)
		MNAR	CC	-2.157 (2.191)	9.434	0.843	9.062 (1.052)
		MNAR	IPW	0.559 (2.408)	6.090	0.952	10.270 (1.494)
	1000	no dropout	CC	-0.069 (1.313)	1.722	0.968	5.172 (0.178)
		MAR	CC	0.056 (1.505)	2.258	0.968	6.181 (0.328)
		MAR	IPW	0.031 (1.561)	2.428	0.956	6.282 (0.377)
		MNAR	CC	-2.234 (1.560)	7.414	0.704	6.263 (0.403)
		MNAR	IPW	0.320 (1.909)	3.733	0.920	6.963 (0.563)
25	500	no dropout	CC	-0.004 (1.236)	1.523	0.932	4.740 (0.150)
		MAR	CC	0.086 (1.793)	3.209	0.964	7.355 (2.038)
		MAR	IPW	-0.006 (1.915)	3.653	0.948	7.366 (1.740)
		MNAR	CC	-2.334 (1.933)	9.172	0.768	7.801 (1.204)
		MNAR	IPW	0.210 (2.302)	5.320	0.932	8.916 (1.928)
	1000	no dropout	CC	-0.011 (0.728)	0.528	0.976	3.344 (0.078)
		MAR	CC	0.054 (1.159)	1.340	0.968	4.974 (0.239)
		MAR	IPW	-0.009 (1.174)	1.373	0.968	5.004 (0.313)
		MNAR	CC	-2.313 (1.338)	7.134	0.604	5.384 (0.342)
		MNAR	IPW	0.184 (1.614)	2.630	0.956	6.040 (0.549)

density ratio for all three methods. Given that not all estimators provide interval estimation, we only compare the point estimators of the policy value. The evaluation focuses on the scenario with  $T = 25$ ,  $n = 500$ , and target policy  $\pi_1$ , with the results summarized in Table 2.3. The results indicate that MWL and LSTDQ yield identical results, validating our derivation that these two estimators are equivalent when using the same set of basis functions for approximation. For both FQE and MWL, the complete-case value estimator demonstrates approximate unbiasedness under ignorable missingness but exhibits higher bias and MSE under nonignorable missingness. By incorporating the proposed IPW adjustment, both the bias and MSE can be effectively reduced. These results align with our discussion in Section 2.4.2.

Table 2.2: Results of value estimates and 95% confidence intervals for policy  $\pi_2$  in the 2D-Linear environment. The average bias, MSE values, ECP, and AL are reported for each estimator (with standard error in parenthesis).

$T$	$n$	Dropout	Method	Bias	MSE	ECP	AL
10	500	no dropout	CC	0.076 (1.038)	1.078	0.960	4.218 (0.210)
		MAR	CC	0.080 (1.312)	1.720	0.956	5.152 (0.765)
		MAR	IPW	0.042 (1.401)	1.958	0.936	5.255 (1.238)
		MNAR	CC	-2.142 (1.337)	6.371	0.560	5.085 (0.717)
		MNAR	IPW	0.144 (1.577)	2.497	0.936	6.064 (1.268)
	1000	no dropout	CC	0.017 (0.761)	0.577	0.964	2.967 (0.103)
		MAR	CC	0.088 (0.892)	0.800	0.952	3.568 (0.189)
		MAR	IPW	0.060 (0.916)	0.838	0.960	3.610 (0.207)
		MNAR	CC	-2.172 (0.893)	5.514	0.304	3.525 (0.243)
		MNAR	IPW	0.050 (1.183)	1.395	0.928	4.156 (0.417)
25	500	no dropout	CC	0.011 (0.725)	0.523	0.940	2.715 (0.087)
		MAR	CC	0.038 (1.030)	1.059	0.960	4.138 (0.519)
		MAR	IPW	-0.039 (1.088)	1.180	0.956	4.154 (0.593)
		MNAR	CC	-2.322 (1.113)	6.625	0.404	4.369 (0.628)
		MNAR	IPW	-0.084 (1.415)	2.002	0.924	5.391 (2.108)
	1000	no dropout	CC	0.045 (0.419)	0.177	0.976	1.916 (0.044)
		MAR	CC	0.089 (0.705)	0.503	0.968	2.885 (0.132)
		MAR	IPW	0.038 (0.709)	0.502	0.968	2.886 (0.148)
		MNAR	CC	-2.293 (0.754)	5.824	0.148	3.042 (0.216)
		MNAR	IPW	-0.014 (0.957)	0.912	0.960	3.663 (0.399)

## 2.6 Real Data Application

We apply the proposed method to a sepsis dataset from the Medical Information Mart for Intensive Care (MIMIC-III v1.4) database (Johnson et al. 2016), following the cohort definition and inclusion/exclusion criteria of Komorowski et al. (2018). Sepsis is a severe medical condition that occurs when the body’s response to infection causes damage to its tissues and organs (Singer et al. 2016). Intravenous fluids (IV) and vasopressors (VASO) are two commonly administered interventions to correct hypotension caused by infection (Gotts and Matthay 2016), and we are interested in evaluating different IV and VASO management policies using this offline dataset.

The state space is constructed using 15 features including demographics (e.g., age), lab

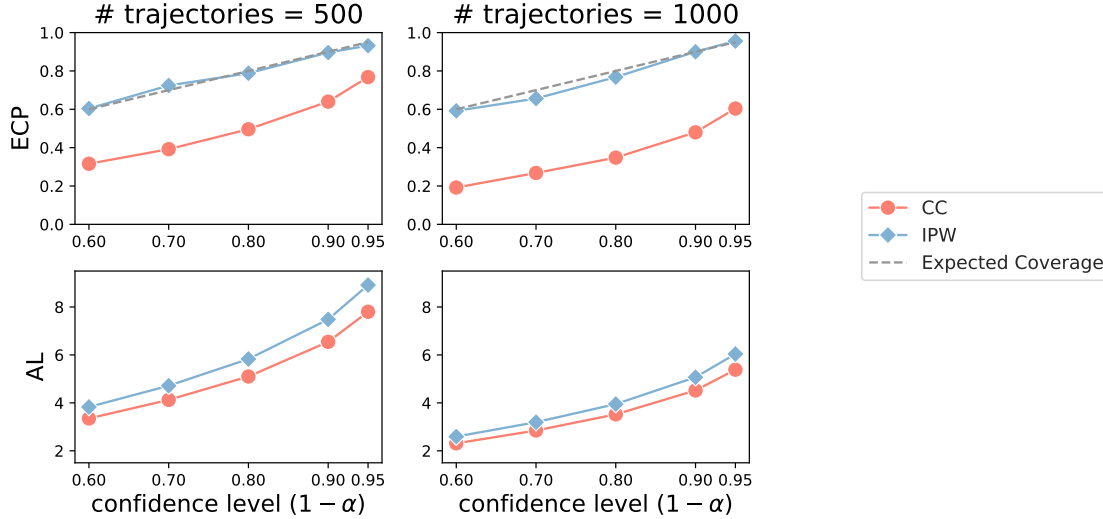


Figure 2.2: The ECP (top) and AL (bottom) of value confidence intervals for policy  $\pi_1$  under nonignorable missingness obtained from IPW (blue) and CC (red) estimators.

measurements (e.g., arterial pH, arterial lactate), and vital signs (e.g., heart rate, respiratory rate). These are important features that clinicians would examine when determining appropriate treatment and dosage for patients. Our action space consists of three actions: no intravenous fluids and no vasopressors, intravenous fluids only, and vasopressors. The Sequential Organ Failure Assessment (SOFA) score reflects sepsis-related organ dysfunction, with a lower score being preferable during sepsis treatment. Thus, we consider a reward of  $R_{t+1} = 1 - 2 \cdot \mathbb{1}(S_{t+1}^{\text{SOFA}} \geq 12)$ , which penalizes high SOFA scores that exceed some threshold.

As shown in Figure 2.1, there is a portion of dropouts in the dataset that is closely related to the patient’s health status, necessitating proper adjustment in OPE. Given that mortality is commonly correlated with the current status of patients, it is reasonable to assume a nonignorable missing mechanism, i.e., the missingness cannot be fully explained by the observed data. To investigate how such dropouts impact the OPE results, we apply the proposed value estimator and compare it to the complete-case estimator. Target policies include a fitted behavior policy and optimal policies trained from Dueling Double Deep Q-Network (Wang et al. 2016) and Batch-Constrained Deep Q-Learning (BCQ) (Fujimoto et al. 2019). More details on this experiment can be found in Appendix A.3. The value estimation results are presented in Table 2.4. In general, the IPW estimator yields lower value estimates than the CC estimator. This aligns with our intuition since patients who dropped out early due to mortality were considered to be in worse condition with lower rewards than those who did not, hence the CC estimator tends to overestimate the value.

Table 2.3: Results of various value estimators for policy  $\pi_1$  in the 2D-Linear environment with horizon  $T = 25$  and  $n = 500$  trajectories. The average bias (with standard error in parenthesis) and MSE values are reported for each estimator.

OPE	Dropout	Method	Bias	MSE
LSTDQ-spline	no dropout	CC	-0.004 (1.236)	1.523
	MAR	CC	0.086 (1.793)	3.209
	MAR	IPW	-0.006 (1.915)	3.653
	MNAR	CC	-2.334 (1.933)	9.172
	MNAR	IPW	0.210 (2.302)	5.320
FQE-spline	no dropout	CC	0.025 (1.249)	1.555
	MAR	CC	0.128 (1.811)	3.283
	MAR	IPW	0.061 (1.957)	3.818
	MNAR	CC	-2.298 (1.951)	9.071
	MNAR	IPW	0.366 (2.341)	5.594
MWL-spline	no dropout	CC	-0.004 (1.236)	1.523
	MAR	CC	0.086 (1.793)	3.209
	MAR	IPW	-0.006 (1.915)	3.653
	MNAR	CC	-2.334 (1.933)	9.172
	MNAR	IPW	0.210 (2.302)	5.320

In this case study, our primary goal is to illustrate the application of the proposed method and raise the concern for nonignorable (informative) missingness in real-world offline data. The construction of states, actions, rewards, and the dropout model requires further examination by domain experts. Besides the presented medical application, our proposed method is general enough and does not target any specific application area.

## 2.7 Discussion and Future Work

This work studies OPE with monotone missing data. We theoretically show that the complete-case value estimator is still valid under ignorable missingness but can be biased if the missing mechanism is nonignorable. To address the bias, we propose a semiparametric IPW value estimator, which is shown to be consistent and asymptotically normal under nonignorable missingness. The effectiveness of the proposed method is empirically demonstrated through several simulation studies. One limitation of our method results from the difficulty of justifying the missingness mechanism and identifying the instrumental variables, which often require

Table 2.4: Off-Policy evaluation results for three different target policies using the MIMIC-III sepsis dataset. The value estimates and confidence intervals are reported.

Policy	Method	$\widehat{V}^\pi$	CI
Behavior	CC	4.274	(4.191,4.357)
	IPW	4.259	(4.177,4.341)
Dueling DQN	CC	4.561	(4.420,4.702)
	IPW	4.479	(4.342,4.615)
BCQ	CC	4.566	(4.427,4.705)
	IPW	4.482	(4.348,4.616)

context and subject-matter knowledge. Handling nonignorable missingness is still an ongoing research area in the field of missing data methodology, and we leave the integration of these evolving advancements for future investigations. Besides, We believe extending this work to learning optimal policies using incomplete data can be an interesting direction worthy of future work.

# Chapter 3

## Offline Inverse Reinforcement Learning via Joint Soft-Q and Reward Learning

### 3.1 Introduction

Reinforcement Learning (RL) is a general technique to derive optimal policies for sequential decision-making problems with the goal of maximizing cumulative rewards. In real-world applications, manually specifying the reward function can sometimes be challenging, as it often involves balancing multiple and possibly competing outcomes. For example, in medical applications, there is often a need to balance symptom reduction with the risk of an adverse event. When expert demonstration data is available, Imitation Learning (IL) techniques can be applied to learn the optimal policy without specifying the reward. Inverse Reinforcement Learning (IRL) is a special type of IL that seeks to first derive the reward function from expert demonstrations and then use the retrieved reward function to guide policy search. In practice, IRL is often preferred over general IL, as it provides insights into the motivations of experts and facilitates knowledge transfer to similar environments or tasks.

Our work is motivated by the application of RL to Electronic Health Record (EHR) data. EHR data has gained increasing popularity in the research community due to its rich information collected from daily clinical practice, allowing for the study of a wide range of problems. The MIMIC-III database (Johnson et al. 2016), an example of EHRs, contains intensive care patient trajectories that record patients' conditions and therapeutic interventions. The extensive longitudinal data along with treatment information facilitates the investigation of optimal treatment decisions (Prasad et al. 2017; Raghu et al. 2017). When applying IRL to EHR data, a notable challenge is that the methods should be applicable to offline settings since online interaction is infeasible in such a safety-critical domain. While

there is a significant body of literature on IRL that allows for flexible reward models and complex state spaces (Wulfmeier et al. 2015; Finn et al. 2016; Ho and Ermon 2016; Fu et al. 2017), these methods are mainly intended for online settings that require further interaction with the environment. Furthermore, existing IRL methods typically presume homogeneous demonstrations for a single task, whereas in practice it may be easier to collect data on heterogeneous but related behaviors. For example, patients are admitted to different types of intensive care units (e.g., medical ICU or surgical ICU) depending on their conditions, and the optimal treatment rules may vary across different types of care units (Jabaley et al. 2018; Zhang et al. 2020b). In such cases, a single reward function and optimal policy are insufficient to account for all the demonstrations.

To address the aforementioned challenges in EHR applications, we developed a novel offline IRL method called Inverse soft-Q and Reward Learning (IQRL), which offers several advantages over existing approaches. First, our method is more stable as it does not rely on adversarial training and has good convergence performance. Second, it avoids computationally expensive inner-loop operations by jointly learning the reward and soft Q-function, thereby improving computational efficiency. Third, our method provides a portable reward function and offers flexibility in applying potential constraints to the reward model. The proposed method achieves state-of-the-art performance in numerical experiments and also provides a reliable estimate of the reward function. To handle heterogeneity in expert demonstrations, we further propose a variant of IQRL called Distilled-IQRL (D-IQRL), which inherits the advantages of IQRL while incorporating a reward distillation module that jointly learns task rewards and strategic-specific rewards.

The rest of the chapter is organized as follows. In Section 3.2, we provide a review of the preliminary framework for RL and maximum entropy IRL. In Section 3.3.1, we introduce our IQRL algorithm, and in Section 3.3.2, we propose a reward-distilled extension of IQR to handle heterogeneous demonstrations. Section 3.4 presents simulation studies, and Section 3.5 showcases a real data application to demonstrate the effectiveness of our proposed methods. We summarize our work and discuss future directions in Section 3.6.

## 3.2 Preliminaries

In this section, we provide an overview of maximum entropy RL and maximum entropy IRL.

### 3.2.1 Maximum Entropy Reinforcement Learning

The framework of standard RL is reviewed in Section 1.3. Recall that the optimal policy  $\pi^*$  is learned by maximizing the expected discounted cumulative rewards

$$\pi^* = \operatorname{argmax}_{\pi \in \Pi} \mathbb{E}_{\tau \sim \pi} \left\{ \sum_{t=0}^{\infty} \gamma^t \cdot r(S_t, A_t) \right\} = \operatorname{argmax}_{\pi \in \Pi} \frac{1}{1 - \gamma} \mathbb{E}_{d_\pi} \{r(S_t, A_t)\},$$

where  $d_\pi(s, a)$  is the normalized discounted visitation probability density defined in (1.5).

Another RL framework is Maximum Entropy RL (MaxEnt-RL), which encourages exploration by augmenting the reward with an entropy of  $\pi$  defined as follows,

$$\mathcal{H}(\pi(\cdot|s)) = \mathbb{E}_{a \sim \pi(\cdot|s)} \{-\log \pi(a|s)\}.$$

A higher entropy value indicates a more stochastic policy. In the MaxEnt-RL framework, the goal is to learn a policy that not only maximizes rewards but also exhibits a certain level of randomness. Therefore, the optimal policy is obtained by maximizing the expected entropy-regularized cumulative reward

$$\pi^* = \operatorname{argmax}_{\pi \in \Pi} \mathbb{E}_{\tau \sim \pi} \left\{ \sum_{t=0}^{\infty} \gamma^t [r(S_t, A_t) + \alpha \cdot \mathcal{H}(\pi(\cdot|S_t))] \right\},$$

where  $\alpha$  is a temperature parameter that determines the relative weight of entropy and reward. Slightly different from standard RL, the optimal soft Q-function and soft value function for MaxEnt-RL are given by

$$\begin{aligned} Q_{\text{soft}}^*(s, a) &= r(s, a) + \mathbb{E}_{\tau \sim \pi} \left\{ \sum_{t=1}^{\infty} \gamma^t [r(S_t, A_t) + \alpha \cdot \mathcal{H}(\pi^*(\cdot|S_t))] \middle| S_0 = s, A_0 = a \right\}, \\ V_{\text{soft}}^*(s) &= \alpha \cdot \log \sum_{a \in \mathcal{A}} \exp \left\{ \frac{1}{\alpha} Q_{\text{soft}}^*(s, a) \right\}. \end{aligned} \quad (3.1)$$

For discrete action space, the optimal soft value for a given state is obtained by taking the log-sum-exp of the Q-values, serving as a “soft” maximum. The corresponding soft Bellman equation (Haarnoja et al. 2017) is given by

$$Q_{\text{soft}}^*(s, a) = r(s, a) + \gamma \cdot \mathbb{E}_{S' \sim p(\cdot|s, a)} \{V_{\text{soft}}^*(S')\}. \quad (3.2)$$

A nice property of MaxEnt-RL is that the optimal policy can be explicitly expressed with

the optimal soft Q-function as follows

$$\pi^*(a|s) = \frac{\exp\left\{\frac{1}{\alpha}Q_{\text{soft}}^*(s, a)\right\}}{\sum_{a' \in \mathcal{A}} \exp\left\{\frac{1}{\alpha}Q_{\text{soft}}^*(s, a')\right\}} = \frac{\exp\left\{\frac{1}{\alpha}Q_{\text{soft}}^*(s, a)\right\}}{\exp\left\{\frac{1}{\alpha}V_{\text{soft}}^*(s)\right\}} = \exp\left\{\frac{1}{\alpha}A_{\text{soft}}^*(s, a)\right\}, \quad (3.3)$$

where  $A_{\text{soft}}^*$  is the soft advantage function defined as  $A_{\text{soft}}^*(s, a) = Q_{\text{soft}}^*(s, a) - V_{\text{soft}}^*(s)$ . This expression indicates that the optimal policy is a stochastic policy characterized by the soft advantage function. Intuitively, for a given state, the action with the highest soft Q-value is more likely to be selected. The temperature parameter  $\alpha$  controls the randomness of the optimal policy: a higher  $\alpha$  leads to a more uniform policy, while a lower  $\alpha$  will bring  $\pi^*(a|s)$  closer to a deterministic policy. This nice property gives rise to an important IRL framework known as Maximum Entropy IRL (MaxEnt-IRL).

### 3.2.2 Maximum Entropy Inverse Reinforcement Learning

The Maximum Entropy IRL (MaxEnt-IRL) framework (Ziebart et al. 2008, 2010) aims to learn a reward function that generates an optimal policy with similar performance to expert demonstrations while also promoting higher policy entropy. Denote the expert policy as  $\pi_E$ , and assume the observed trajectories have a horizon of  $T$ . Parameterize the reward function with  $\omega$ , MaxEnt-IRL learns the reward function  $r_\omega(s, a)$  by solving the following optimization problem

$$\max_{\omega} \min_{\pi \in \Pi} \mathcal{L}(\omega, \pi) := \mathbb{E}_{\tau \sim \pi_E} \left[ \sum_{t=0}^{T-1} \gamma^t \cdot r_\omega(S_t, A_t) \right] - \left\{ \mathbb{E}_{\tau \sim \pi} \left[ \sum_{t=0}^{T-1} \gamma^t \cdot r_\omega(S_t, A_t) \right] + \alpha \cdot \mathcal{H}(\pi) \right\}, \quad (3.4)$$

where

$$\mathcal{H}(\pi) = \mathbb{E}_{\tau \sim \pi} \left[ \sum_{t=0}^{T-1} \gamma^t \cdot \mathcal{H}(\pi(\cdot|S_t)) \right] = \mathbb{E}_{\tau \sim \pi} \left[ - \sum_{t=0}^{T-1} \gamma^t \cdot \log \pi(A_t|S_t) \right]$$

is known as the causal entropy (Ziebart et al. 2010). The inner optimization in (3.4) learns the optimal policy by solving a forward MaxEnt-RL under the current reward function, and the outer optimization updates the reward function to distinguish expert policy from the learned optimal policy. The gradient of  $\mathcal{L}(\omega, \pi)$  with respect to the reward parameter  $\omega$  is

$$\nabla_{\omega} \mathcal{L}(\omega, \pi) = \mathbb{E}_{\tau \sim \pi_E} \left[ \sum_{t=0}^{T-1} \gamma^t \cdot \nabla_{\omega} r_\omega(S_t, A_t) \right] - \mathbb{E}_{\tau \sim \pi} \left[ \sum_{t=0}^{T-1} \gamma^t \cdot \nabla_{\omega} r_\omega(S_t, A_t) \right], \quad (3.5)$$

which can be interpreted as the discrepancy of expected cumulative reward gradients between the expert policy and the learned policy.

An alternative formulation is from the perspective of maximum likelihood. Denote the optimal policy under reward  $r_\omega$  as  $\pi_\omega$ , the expected discounted log-likelihood of expert trajectories is given by

$$\begin{aligned} & \mathbb{E}_{\tau \sim \pi_E} \left[ \log \left( p_0(S_0) \prod_{t=0}^{T-1} [p(S_{t+1} | S_t, A_t) \pi_\omega(A_t | S_t)]^{\gamma^t} \right) \right] \\ = & \mathbb{E}_{\tau \sim \pi_E} \left[ \sum_{t=0}^{T-1} \gamma^t \cdot \log \pi_\omega(A_t | S_t) \right] + \mathbb{E}_{\tau \sim \pi_E} \left[ \sum_{t=0}^{T-1} \gamma^t \cdot \log p(S_{t+1} | S_t, A_t) \right] + \mathbb{E}_{\tau \sim \pi_E} [\log p_0(S_0)]. \end{aligned}$$

Since only the first term involves reward parameter  $\omega$ , the maximum likelihood objective can be converted to the following optimization problem

$$\begin{aligned} \max_{\omega} \quad & \mathcal{L}(\omega) = \mathbb{E}_{\tau \sim \pi_E} \left[ \sum_{t=0}^{T-1} \gamma^t \cdot \log \pi_\omega(A_t | S_t) \right] \\ \text{s.t.} \quad & \pi_\omega = \operatorname{argmax}_{\pi_\omega} \mathbb{E}_{\tau \sim \pi_\omega} \left[ \sum_{t=0}^{T-1} \gamma^t (r_\omega(S_t, A_t) + \alpha \cdot \mathcal{H}(\pi(\cdot | S_t))) \right]. \end{aligned} \quad (3.6)$$

By plugging in the expression for the optimal policy from (3.3), the gradient of  $\mathcal{L}(\omega)$  can be shown to be

$$\nabla_{\omega} \mathcal{L}(\omega) = \mathbb{E}_{\tau \sim \pi_E} \left[ \sum_{t=0}^{T-1} \gamma^t \cdot \nabla_{\omega} r_{\omega}(S_t, A_t) \right] - \mathbb{E}_{\tau \sim \pi_\omega} \left[ \sum_{t=0}^{T-1} \gamma^t \cdot \nabla_{\omega} r_{\omega}(S_t, A_t) \right] \quad (3.7)$$

$$= \sum_{t=0}^{T-1} \gamma^t \cdot \mathbb{E}_{p_t^E(s,a)} [\nabla_{\omega} r_{\omega}(S_t, A_t)] - \sum_{t=0}^{T-1} \gamma^t \cdot \mathbb{E}_{p_t^\pi(s,a)} [\nabla_{\omega} r_{\omega}(S_t, A_t)], \quad (3.8)$$

where  $p_t^E(s, a)$  and  $p_t^\pi(s, a)$  denote the marginal density of  $(S_t, A_t) \in \mathcal{S} \times \mathcal{A}$  under expert policy  $\pi_E$  and learned policy  $\pi$ . A detailed derivation is provided in Section 3.2 of Gleave and Toyer (2022). Notably, the gradient in (3.7) is identical to the gradient in (3.5), indicating the equivalence between these two formulations. When applying gradient ascent to update  $\omega$ , the first term of  $\nabla_{\omega} \mathcal{L}(\omega)$  can be easily estimated using the expert trajectories. However, the difficulty arises in estimating the second expectation with respect to trajectory distribution under the learned optimal policy.

The original proposal of MaxEnt-IRL (Ziebart et al. 2008, 2010) considers the undiscounted setting with  $\gamma = 1$  and a linear reward function  $r_\omega(s, a) = \phi(s, a)^\top \omega$ , where  $\phi(s, a)$  is a pre-specified feature mapping. By converting the expectation with respect to trajectories to an expectation over occupancy measure as in (3.8), the gradient can be estimated by explicitly calculating the state-action visitation probabilities under the learned optimal policy.

Wulfmeier et al. (2015) extends the linear reward model to neural networks, allowing for the representation of more complex reward structures. However, both methods rely on the explicit calculation of state-action visitation probability, which is only feasible in small and discrete domains with known state transition dynamics.

To handle larger or even continuous state and action spaces, one can apply the sampling approach to estimate the second expectation term of  $\nabla_{\omega} \mathcal{L}(\omega)$ . GCL (Finn et al. 2016) samples trajectories under a baseline policy and couples reward learning with policy optimization, which allows the baseline policy to gradually approach the optimal policy throughout the training process. However, as pointed out by Fu et al. (2017), one drawback of GCL is that it operates at the level of entire trajectories, which often results in high variance in gradient estimation and is not flexible enough to be applied to more complex problem settings. To address this issue, a better approach is to operate on the state-action pairs instead of the whole trajectories. Define the average visitation probability as follows:

$$\rho_{\pi}(s, a) = \frac{1}{T} \sum_{t=0}^{T-1} p_t^{\pi}(s) \pi(a | s).$$

Note that the distinction between  $\rho_{\pi}(s, a)$  and previously defined  $d_{\pi}(s, a)$  lies in that  $\rho_{\pi}(s, a)$  only involves finite horizons, and it takes the average of  $p_t^{\pi}(s, a)$  instead of the normalized discounted cumulative summation. Therefore,  $\rho_{\pi}(s, a)$  can be seen as the marginalized distribution of state-action pair under some policy  $\pi$ , and we still refer to it as occupancy measure. Denote the marginalized state-action distribution in expert trajectories as  $\rho_E$ , and set  $\gamma = 1$ . The objective and its gradient can be rewritten as

$$\begin{aligned} \mathcal{L}(\omega, \pi) &= \mathbb{E}_{\tau \sim \pi_E} \left[ \sum_{t=0}^{T-1} r_{\omega}(S_t, A_t) \right] - \left\{ \mathbb{E}_{\tau \sim \pi} \left[ \sum_{t=0}^{T-1} r_{\omega}(S_t, A_t) \right] + \alpha \cdot \mathcal{H}(\pi) \right\} \\ &= \{ \mathbb{E}_{\rho_E} [r_{\omega}(S_t, A_t)] - \mathbb{E}_{\rho_{\pi}} [r_{\omega}(S_t, A_t) + \alpha \cdot \mathcal{H}(\pi(\cdot | S_t))] \} \cdot T, \\ \nabla \mathcal{L}_{\omega}(\omega, \pi) &= \mathbb{E}_{\tau \sim \pi_E} \left[ \sum_{t=0}^{T-1} \nabla_{\omega} r_{\omega}(S_t, A_t) \right] - \mathbb{E}_{\tau \sim \pi} \left[ \sum_{t=0}^{T-1} \nabla_{\omega} r_{\omega}(S_t, A_t) \right] \\ &= \{ \mathbb{E}_{\rho_E} [\nabla_{\omega} r_{\omega}(S_t, A_t)] - \mathbb{E}_{\rho_{\pi}} [\nabla_{\omega} r_{\omega}(S_t, A_t)] \} \cdot T \end{aligned}$$

Ho and Ermon (2016) demonstrate that by introducing a convex reward regularizer  $\psi(r_{\omega})$  to the objective function  $\mathcal{L}(\omega, \pi)$ , the optimization problem can be reformulated as an occupancy measure matching problem:

$$\min_{\pi \in \Pi} d_{\psi}(\rho_{\pi}, \rho_E) - \alpha \cdot \mathbb{E}_{\rho_{\pi}} [\mathcal{H}(\pi(\cdot | S_t))],$$

where  $d_\psi(\rho_\pi, \rho_E) = \psi^*(\rho_\pi - \rho_E)$  reflects the discrepancy between occupancy measures under the learned policy and expert policy,  $\psi^*$  is the convex conjugate of  $\psi$  given by  $\psi^*(\rho) = \sup_{r \in \mathbb{R}^{S \times A}} \rho^\top r - \psi(r)$ . This leads to a branch of MaxEnt-IRL that focuses on occupancy matching using various divergence measures (Ghasemipour et al. 2020).

Based on this insight, GAIL (Ho and Ermon 2016) harnesses generative adversarial training (Goodfellow et al. 2020), where a discriminator is trained to distinguish between the distributions  $\rho_\pi$  and  $\rho_E$ , while a policy is trained to generate samples that resemble expert behaviors. Specifically, GAIL parameterizes the policy with  $\theta$  and the discriminator with  $\omega$ , and solves the following optimization problem

$$\min_{\theta} \max_{\omega} \mathbb{E}_{\rho_{\pi_\theta}} [\log(D_\omega(S_t, A_t))] + \mathbb{E}_{\rho_E} [\log(1 - D_\omega(S_t, A_t))] - \alpha \cdot \mathbb{E}_{\rho_{\pi_\theta}} [\mathcal{H}(\pi_\theta(\cdot | S_t))].$$

The algorithm alternates between a gradient step on  $\omega$  to update the discriminator using the cross-entropy loss and a forward MaxEnt-RL step to update the policy using  $r(s, a) = -\log D_\omega(s, a)$  as the pseudo-reward function. At convergence, the learned policy  $\pi_\theta$  produces a similar occupancy measure to that of expert trajectories. However, GAIL only focuses on learning the policy and does not explicitly estimate the reward function through the discriminator. Therefore, this approach does not strictly fall under the category of IRL. To explicitly learn the reward function, AIRL (Fu et al. 2017) inherits the adversarial training procedure and constructs the discriminator as

$$D_{\omega, \theta}(s, a) = \frac{\exp[f_\omega(s, a)/\alpha]}{\exp[f_\omega(s, a)/\alpha] + \pi_\theta(a | s)},$$

where  $f_\omega(s, a)$  is a function related with the reward function. Similar to GAIL, this algorithm alternates between training the discriminator to classify expert data from policy samples and updating the policy to confuse the discriminator. In the policy optimization step, the pseudo-reward function is constructed as

$$r(s, a) = \log[D_{\omega, \theta}(s, a)] - \log[1 - D_{\omega, \theta}(s, a)] = \frac{1}{\alpha} f_\omega(s, a) - \log \pi_\theta(a | s),$$

which is an entropy-regularized version of  $f_\omega$ , and thus  $f_\omega$  plays the role of reward function. At optimality,  $f_\omega$  recovers the optimal soft advantage function  $A_{\text{soft}}^*$  defined as  $Q_{\text{soft}}^*(s, a) - V_{\text{soft}}^*(s)$ . It can be shown that the gradient of the discriminator objective aligns with the gradient of the general MaxEnt-IRL objective (3.5) at optimality when  $\gamma = 1$ .

It is worth mentioning that the aforementioned MaxEnt-IRL methods for continuous state space all require online sampling from the environment. This is because either the objective (3.4) or the gradient (3.5) involves estimating the expectation with respect to the state-

action visitation probability density under the learned policy. This restriction makes these algorithms not suitable for offline settings. Furthermore, in terms of computational stability, the adversarial training procedure employed by GAIL and AIRL is prone to convergence issues.

### 3.3 Offline Inverse Reinforcement Learning

In this section, we present our proposed offline IRL algorithm. In Section 3.3.1, we introduce the Inverse soft-Q and Reward Learning (IQRL) algorithm and highlight its advantages over existing methods. In Section 3.3.2, we introduce an extension of IQRL called Distilled-IQRL (D-IQRL) to handle heterogeneous demonstration data.

#### 3.3.1 Inverse soft-Q and Reward Learning (IQRL)

Let  $\mathcal{D} = \{\tau_i\}_{1 \leq i \leq n}$  denote the expert demonstrations, where  $n$  is the number of trajectories, and each trajectory can be represented as  $\tau_i = \{(S_{i,t}, A_{i,t}, S_{i,t+1})\}_{0 \leq t < T_i}$ . In order to learn a reward function that produces an optimal policy resembling the demonstrated behavior, we utilize the maximum likelihood formulation and attempt to maximize the log-likelihood of observing the expert trajectories. Unlike the MaxEnt-IRL methods discussed in Section 3.2.2, we specifically focus on the objective of the log-likelihood instead of its gradient form (3.5), and we demonstrate that the likelihood objective can be estimated solely from the observed trajectories.

Parameterize the reward as  $r_\omega(s, a)$  and denote the optimal policy under reward  $r_\omega$  as  $\pi_\omega$ , the log-likelihood of observing  $\mathcal{D}$  is given by

$$\begin{aligned} \ell(\omega; \mathcal{D}) &= \log \left\{ \prod_{i=1}^n p_0(S_{i,0}) \prod_{t=0}^{T_i-1} p(S_{i,t+1} | S_{i,t}, A_{i,t}) \pi_\omega(A_{i,t} | S_{i,t}) \right\} \\ &= \sum_{i=1}^n \sum_{t=0}^{T_i-1} \log \pi_\omega(A_{i,t} | S_{i,t}) + \sum_{i=1}^n \sum_{t=0}^{T_i-1} \log p(S_{i,t+1} | S_{i,t}, A_{i,t}) + \sum_{i=1}^n \log p_0(S_{i,0}). \end{aligned}$$

By dropping the constant terms with respect to  $\omega$ , we obtain the following objective

$$\begin{aligned} \max_{\omega} \quad & \mathcal{L}(\omega) := \mathbb{E}_{\mathcal{D}} \{ \log \pi_\omega(A_{i,t} | S_{i,t}) \} \\ \text{s.t.} \quad & \pi_\omega = \operatorname{argmax}_{\pi_\omega} \mathbb{E}_{\tau \sim \pi_\omega} \left\{ \sum_{t=0}^{\infty} \gamma^t [r_\omega(S_t, A_t) + \alpha \cdot \mathcal{H}(\pi(\cdot | S_t))] \right\}, \end{aligned}$$

where  $\mathbb{E}_{\mathcal{D}}[\cdot] := \frac{1}{\sum_{i=1}^n T_i} \sum_{i=1}^n \sum_{t=0}^{T_i-1} [\cdot]$  denotes the empirical average. By plugging in the

analytical expression of the optimal policy as shown in (3.3),  $\mathcal{L}(\omega)$  can be expressed as follows

$$\begin{aligned}
\mathcal{L}(\omega) &= \mathbb{E}_{\mathcal{D}} \{ \log \pi_{\omega}(A_{i,t} | S_{i,t}) \} = \mathbb{E}_{\mathcal{D}} \left\{ \frac{1}{\alpha} [Q_{\text{soft}}^{\pi_{\omega}}(S_{i,t}, A_{i,t}) - V_{\text{soft}}^{\pi_{\omega}}(S_{i,t})] \right\} \\
&\stackrel{(i)}{=} \frac{1}{\alpha} \mathbb{E}_{\mathcal{D}} \{ r_{\omega}(S_{i,t}, A_{i,t}) + \gamma \cdot \mathbb{E}_{s_{t+1} \sim p(\cdot | S_{i,t}, A_{i,t})} V_{\text{soft}}^{\pi_{\omega}}(s_{t+1}) - V_{\text{soft}}^{\pi_{\omega}}(S_{i,t}) \} \\
&\stackrel{(ii)}{\approx} \frac{1}{\alpha} \mathbb{E}_{\mathcal{D}} \{ r_{\omega}(S_{i,t}, A_{i,t}) + \gamma \cdot V_{\text{soft}}^{\pi_{\omega}}(S_{i,t+1}) - V_{\text{soft}}^{\pi_{\omega}}(S_{i,t}) \} \\
&\stackrel{(iii)}{=} \frac{1}{\alpha} \mathbb{E}_{\mathcal{D}} \left\{ r_{\omega}(S_{i,t}, A_{i,t}) + \gamma \cdot \alpha \log \sum_{a \in \mathcal{A}} \exp \left( \frac{1}{\alpha} Q_{\text{soft}}^{\pi_{\omega}}(S_{i,t+1}, a) \right) \right. \\
&\quad \left. - \alpha \log \sum_{a \in \mathcal{A}} \exp \left( \frac{1}{\alpha} Q_{\text{soft}}^{\pi_{\omega}}(S_{i,t}, a) \right) \right\} \tag{3.9}
\end{aligned}$$

In the above derivation, step (i) follows from the soft Bellman equation (3.2), step (ii) is a reasonable approximation if the demonstration dataset  $\mathcal{D}$  is sufficiently large, and step (iii) is obtained by expressing the soft value using soft Q-function as shown in (3.1). We remark that the reward function can be relaxed to include the next state, allowing for dependence on  $(s_t, a_t, s_{t+1})$ . Such a reward function facilitates easier interpretation on some occasions, especially when the reward is determined by the changes in consecutive state features. This extension is justified by an alternative formulation of the soft Bellman equation,  $Q_{\text{soft}}^{\pi_{\omega}}(s_t, a_t) = \mathbb{E}_{s_{t+1} \sim p(\cdot | s_t, a_t)} \{ r(s_t, a_t, s_{t+1}) + \gamma \cdot V_{\text{soft}}^{\pi_{\omega}}(s_{t+1}) \}$ , and then the RHS of (ii) can also be written as  $(1/\alpha) \cdot \mathbb{E}_{\mathcal{D}} \{ r_{\omega}(S_{i,t}, A_{i,t}, S_{i,t+1}) + \gamma \cdot V_{\text{soft}}^{\pi_{\omega}}(S_{i,t+1}) - V_{\text{soft}}^{\pi_{\omega}}(S_{i,t}) \}$ . For the remaining discussion, we will primarily focus on the base case where the reward is solely a function of  $(s_t, a_t)$ .

The objective function (3.9) is well-suited for offline settings as it only requires sampling from the observed data. To learn the reward function based on this objective, it is necessary to estimate the soft Q-function as well. One approach is to use a nested loop, where the inner loop performs forward RL to learn the soft-Q function for the current reward while the outer loop updates the reward function. However, this approach can be computationally expensive. In light of this, we propose to learn the reward function and the soft-Q function simultaneously with a constraint on the soft Bellman residual to ensure the reward function and soft-Q function are aligned and satisfy the soft-Bellman equation (3.2).

We parameterize the reward function and soft-Q function with  $\omega$  and  $\beta$  respectively. For succinctness, we use  $V_{\beta}(s) := \alpha \log \sum_{a \in \mathcal{A}} \exp [(1/\alpha) Q_{\beta}(s, a)]$  to represent the soft value

function as a function of  $Q_\beta$ . Our objective function can be written as follows

$$\begin{aligned} \max_{\omega, \beta} \mathcal{J}(\omega, \beta) &:= \mathbb{E}_{\mathcal{D}} [r_\omega(S_{i,t}, A_{i,t}) + \gamma \cdot V_\beta(S_{i,t+1}) - V_\beta(S_{i,t})] \\ \text{s.t. } \mathbb{E}_{\mathcal{D}} [r_\omega(S_{i,t}, A_{i,t}) + \gamma \cdot V_\beta(S_{i,t+1}) - Q_\beta(S_{i,t}, A_{i,t})]^2 &\leq \delta. \end{aligned} \quad (3.10)$$

Here,  $\delta$  controls the strength of the constraint applied to the loss function of soft-Q iteration (Haarnoja et al. 2017). Rewriting the above objective as the Lagrangian dual yields

$$\begin{aligned} \max_{\omega, \beta} \mathcal{J}(\omega, \beta) &:= \mathbb{E}_{\mathcal{D}} [r_\omega(S_{i,t}, A_{i,t}) + \gamma \cdot V_\beta(S_{i,t+1}) - V_\beta(S_{i,t})] \\ &\quad - \lambda \cdot \mathbb{E}_{\mathcal{D}} [r_\omega(S_{i,t}, A_{i,t}) + \gamma \cdot V_\beta(S_{i,t+1}) - Q_\beta(S_{i,t}, A_{i,t})]^2, \end{aligned} \quad (3.11)$$

where the Lagrangian multiplier  $\lambda \geq 0$  controls the strength of constraint. Based on the objective (3.11), the parameters  $\omega$  and  $\beta$  are updated through stochastic gradient ascent. For the reward function, when there are several pre-specified reward components, we can use a weighted average to represent the reward as  $r_\omega(s, a) = \phi(s, a)^\top \omega$ , where  $\phi(s, a)$  is a vector of reward components constructed from the state-action pair. Such a linear model offers better interpretability, and the learned weight  $\omega$  provides insights into how to balance different aspects in a composite reward. When knowledge about potential reward components is limited, a more flexible way is to directly model the reward function using neural networks or other non-parametric models, which allows for capturing complex patterns within the data and does not rely on predefined reward features. In terms of the soft Q-function, since it is not the primary focus of the IRL algorithm, it is preferable to impose minimal assumptions on it by using non-parametric models such as neural networks or spline approximation. The pseudo-code for the proposed algorithm is outlined in Algorithm 2.

---

**Algorithm 2** Inverse soft-Q and Reward Learning (IQRL)

---

- 1: **Input:** Expert trajectories  $\mathcal{D} = \{\tau_i\}_{i=1}^n$ , discount factor  $\gamma$ , penalty weight  $\lambda$ , mini-batch size  $b$ , learning rate  $\eta$
  - 2: Initialize the parameters  $\omega, \beta$
  - 3: **while** not converged **do**
  - 4:     Randomly sample a mini-batch  $\mathcal{D}_{\text{mini}}$  with  $b$  transitions  $(s_t, a_t, s_{t+1})$  from  $\mathcal{D}$
  - 5:     Calculate the objective  $\mathcal{J}(\omega, \beta)$  using the sampled transitions  $\mathcal{D}_{\text{mini}}$
  - 6:     Perform one step of gradient ascent  $(\omega', \beta') \leftarrow (\omega, \beta) + \eta \cdot \nabla_{\omega, \beta} \mathcal{J}(\omega, \beta)$
  - 7:     Update parameters  $(\omega, \beta) \leftarrow (\omega', \beta')$
  - 8: **end while**
  - 9: **Return:** Parameters  $\omega$  of reward function and  $\beta$  of soft Q-function
-

Our method offers several advantages over existing methods. First, it is more stable in convergence because it does not involve adversarial training with mini-max optimization as in GAIL (Ho and Ermon 2016), AIRL (Fu et al. 2017), and ValueDICE (Kostrikov et al. 2019). Second, it is suitable for offline settings as it only needs to sample from observed data, and does not rely on state-action distribution under the learned policy. As such, it does not need to sample new trajectories for estimation, unlike ML-IRL (Zeng et al. 2022) and other on-policy distribution matching approaches (Ho and Ermon 2016; Fu et al. 2017). Additionally, our method is efficient in computation as we estimate the reward and soft Q-function simultaneously, avoiding the need to repeatedly solve for the soft Q-function or optimal policy in the inner loop. A similar idea is utilized by AVRIL (Chan and van der Schaar 2021), which simultaneously learns a variational distribution of reward posterior and Q-function, but its objective is the Evidence Lower BOund (ELBO) of the log-likelihood under the Bayesian IRL framework. Our numerical experiments indicate that AVRIL is sensitive to parameter initialization, and its performance is comparatively weaker in more complex domains compared to our proposed method. Recently, IQ-Learn (Garg et al. 2021) optimizes a regularized objective function solely with respect to the soft Q-function and expresses the reward with soft Q-function as  $Q_{\text{soft}}^{\pi_{\omega}}(s_t, a_t) - \gamma V_{\text{soft}}^{\pi_{\omega}}(s_{t+1})$ . While this method is compatible with offline settings and demonstrates stable convergence, it compromises the accuracy of reward estimation as it indirectly recovers the rewards from a soft Q-function approximator, which heavily relies on the environment dynamics and does not strictly adhere to the soft-Bellman equation, as pointed out by Zeng et al. (2022). Without an explicit and accurate reward estimation, it might fail to provide insights into how experts make decisions in the given context. Another advantage of modeling an explicit reward function is that it enables the incorporation of various constraints and structures into the reward function.

### 3.3.2 Distilled Inverse soft-Q and Reward Learning (D-IQRL)

Existing IRL methods typically assume expert demonstrations to be homogeneous, while in practice, the logged trajectories may exhibit heterogeneity due to slightly different strategies. In this section, we present a variant of the proposed IQRL algorithm that handles heterogeneous demonstrations. Specifically, we consider the scenario with a single high-level task and  $K$  types of strategies, the strategy label is assumed to be known. The expert trajectories for strategy  $k$  is denoted as  $\mathcal{D}_k = \{\tau_{k,i}\}_{i=1}^{n_k}$ ,  $k = 1, \dots, K$ .

If the reward functions for each strategy are unrelated, one cannot do better than applying the base IRL algorithm to each strategy separately. However, in practice, similar tasks often have reward functions with similar structures, and joint training can lead to more

sample-efficient inference of the reward functions. Furthermore, the knowledge obtained from joint training can facilitate knowledge transfer to similar tasks. To leverage the common task reward structure and borrow information from each other, a common approach is to share parameters across strategies to improve data efficiency. However, it may result in negative transfer, where the training of some strategies adversely affects others and hence makes training unstable (Sun et al. 2020). One way to mitigate negative transfer is through knowledge distillation (Hinton et al. 2015). The key idea is to “distill” knowledge from multiple models into a single model that consolidates the common knowledge. The distillation technique has been studied in the context of RL for multi-task learning and transfer learning (Rusu et al. 2015; Teh et al. 2017). Inspired by this line of research, we perform distillation on the reward functions and decompose the strategy reward function into two parts,

$$r^{(k)}(s, a) = r^{(0)}(s, a) + \tilde{r}^{(k)}(s, a),$$

where  $r^{(0)}(s, a)$  is the “distilled” task reward that captures common reward across strategies, and  $\tilde{r}^{(k)}(s, a)$  is the strategy-specific reward that allows for a certain degree of heterogeneity among strategies. The two reward sources  $r^{(0)}(s, a)$  and  $\tilde{r}^{(k)}(s, a)$  are modeled separately. Such two-column architecture is shown to yield faster convergence (Teh et al. 2017). We parameterize the task reward with  $\omega_0$ , the  $k$ -th strategy-specific reward with  $\omega_k$ , and the  $k$ -th soft Q-function with  $\beta_k$ .

We first consider linear reward models:  $r^{(0)}(s, a) = \phi_0(s, a)^\top \omega_0$ ,  $\tilde{r}^{(k)}(s, a) = \phi_1(s, a)^\top \omega_k$ , where  $\phi_0(s, a)$  represents a vector of task features constructed from the state-action pair while  $\phi_1(s, a)$  represents the strategy features. Then the strategy reward can be expressed as  $r_{\omega_0, \omega_k}^{(k)}(s, a) = r_{\omega_0}^{(0)}(s, a) + \tilde{r}_{\omega_k}^{(k)}(s, a) = \phi_0(s, a)^\top \omega_0 + \phi_1(s, a)^\top \omega_k$ . To facilitate knowledge sharing among strategies, we impose a constraint on the magnitude of  $\omega_k$  in the form of  $\mathcal{L}_{\text{reg}, k} = \|\omega_k\|_2^2$ , which ensures that the shared task reward resides at the centroid of all strategy rewards and that each strategy reward stays close to the shared reward. The proposed objective is given as follows

$$\begin{aligned} & \max_{\omega_0, \{\omega_k, \beta_k\}_{k=1}^K} \frac{1}{K} \sum_{k=1}^K \mathbb{E}_{\mathcal{D}_k} \left[ r_{\omega_0, \omega_k}^{(k)}(S_{i,t}, A_{i,t}) + \gamma \cdot V_{\beta_k}^{(k)}(S_{i,t+1}) - V_{\beta_k}^{(k)}(S_{i,t}) \right], \\ \text{s.t. } & \frac{1}{K} \sum_{k=1}^K \mathbb{E}_{\mathcal{D}_k} \left[ r_{\omega_0, \omega_k}^{(k)}(S_{i,t}, A_{i,t}) + \gamma \cdot V_{\beta_k}^{(k)}(S_{i,t+1}) - Q_{\beta_k}^{(k)}(S_{i,t}, A_{i,t}) \right]^2 \leq \delta_1, \\ & \frac{1}{K} \sum_{k=1}^K \|\omega_k\|_2^2 \leq \delta_2, \end{aligned}$$

where  $\delta_1$  and  $\delta_2$  control the strength of constraints imposed on the average soft Bellman residual loss and the average squared  $L_2$ -norm of strategy-specific reward weights, respectively. Note that we do not perform distillation on the soft Q-function, as our primary goal is to learn the reward function. The corresponding dual Lagrangian can be expressed as follows, with  $\lambda_1$  and  $\lambda_2$  being the penalty weights:

$$\begin{aligned}
& \max_{\omega_0, \{\omega_k, \beta_k\}_{k=1}^K} \mathcal{J}(\omega_0, \{\omega_k, \beta_k\}_{k=1}^K) \\
& := \frac{1}{K} \sum_{k=1}^K \mathbb{E}_{\mathcal{D}_k} \left[ r_{\omega_0, \omega_k}^{(k)}(S_{i,t}, A_{i,t}) + \gamma \cdot V_{\beta_k}^{(k)}(S_{i,t+1}) - V_{\beta_k}^{(k)}(S_{i,t}) \right] \\
& \quad - \lambda_1 \cdot \frac{1}{K} \sum_{k=1}^K \mathbb{E}_{\mathcal{D}_k} \left[ r_{\omega_0, \omega_k}^{(k)}(S_{i,t}, A_{i,t}) + \gamma \cdot V_{\beta_k}^{(k)}(S_{i,t+1}) - Q_{\beta_k}^{(k)}(S_{i,t}, A_{i,t}) \right]^2 \\
& \quad - \lambda_2 \cdot \frac{1}{K} \sum_{k=1}^K \|\omega_k\|_2^2.
\end{aligned}$$

The limitation of linear reward models is that they rely on hand-crafted reward features, which can be challenging to specify in practice. To obviate the need for explicitly specifying a reward feature map, an alternative approach is to use neural networks to directly model both  $r^{(0)}(s, a)$  and  $\tilde{r}^{(k)}(s, a)$ . However, the aforementioned regularization is not directly applicable to neural networks because regularizing the parameter space may not necessarily regularize the output space, and small changes in certain parameters can have a large impact on the predicted reward. Therefore, instead of regularizing the parameter space, we penalize the magnitude of the strategy-specific reward as follows

$$\mathcal{L}_{\text{reg},k} = \mathbb{E}_{\mathcal{D}_k} \left[ \tilde{r}_{\omega_k}^{(k)}(S_{i,t}, A_{i,t}) \right]^2, k = 1, \dots, K.$$

The corresponding objective function is then given by

$$\begin{aligned}
& \max_{\omega_0, \{\omega_k, \beta_k\}_{k=1}^K} \mathcal{J}(\omega_0, \{\omega_k, \beta_k\}_{k=1}^K) \\
& := \frac{1}{K} \sum_{k=1}^K \mathbb{E}_{\mathcal{D}_k} \left[ r_{\omega_0, \omega_k}^{(k)}(S_{i,t}, A_{i,t}) + \gamma \cdot V_{\beta_k}^{(k)}(S_{i,t+1}) - V_{\beta_k}^{(k)}(S_{i,t}) \right] \\
& \quad - \lambda_1 \cdot \frac{1}{K} \sum_{k=1}^K \mathbb{E}_{\mathcal{D}_k} \left[ r_{\omega_0, \omega_k}^{(k)}(S_{i,t}, A_{i,t}) + \gamma \cdot V_{\beta_k}^{(k)}(S_{i,t+1}) - Q_{\beta_k}^{(k)}(S_{i,t}, A_{i,t}) \right]^2 \\
& \quad - \lambda_2 \cdot \frac{1}{K} \sum_{k=1}^K \mathbb{E}_{\mathcal{D}_k} \left[ \tilde{r}_{\omega_k}^{(k)}(S_{i,t}, A_{i,t}) \right]^2.
\end{aligned}$$

Such regularization shares some similarities to Meta-AIRL (Gleave and Habryka 2018) and MSRD (Chen et al. 2020), both of which attempt to exploit similarities among reward functions. However, they are built on top of AIRL (Fu et al. 2017), which involves online sampling and adversarial training. In contrast, our method applies to fully offline settings and offers better computational stability.

The pseudo-code for the proposed algorithm is outlined in Algorithm 3.

---

**Algorithm 3** Distilled Inverse soft-Q and Reward Learning (D-IQRL)

---

- 1: **Input:** Expert trajectories for each strategy  $\mathcal{D}_k = \{\tau_{k,i}\}_{i=1}^{n_k}$ ,  $k = 1, \dots, K$ , discount factor  $\gamma$ , penalty weights  $\lambda_1$  and  $\lambda_2$ , mini-batch size  $b$ , learning rate  $\eta$
  - 2: Initialize the parameters  $\omega_0, \{\omega_k, \beta_k\}_{k=1}^K$
  - 3: **while** not converged **do**
  - 4:     **for**  $k = 1, \dots, K$  **do**
  - 5:         Randomly sample a mini-batch  $\mathcal{D}_{k,\text{mini}}$  with  $b$  transitions  $(s_t, a_t, s_{t+1})$  from  $\mathcal{D}_k$
  - 6:     **end for**
  - 7:     Calculate the objective  $\mathcal{J}(\omega_0, \{\omega_k, \beta_k\}_{k=1}^K)$  using the sampled transitions  $\{\mathcal{D}_{k,\text{mini}}\}_{k=1}^K$
  - 8:     Perform one step of gradient ascent
 
$$(\omega'_0, \{\omega'_k, \beta'_k\}_{k=1}^K) \leftarrow (\omega_0, \{\omega_k, \beta_k\}_{k=1}^K) + \eta \cdot \nabla \mathcal{J}(\omega_0, \{\omega_k, \beta_k\}_{k=1}^K)$$
  - 9:     Update parameters  $(\omega_0, \{\omega_k, \beta_k\}_{k=1}^K) \leftarrow (\omega'_0, \{\omega'_k, \beta'_k\}_{k=1}^K)$
  - 10: **end while**
  - 11: **Return:** Parameters  $\omega_0$  of task reward,  $\omega_k$  of  $k$ -th strategy-specific reward, and  $\beta_K$  of  $k$ -th soft Q-function
- 

## 3.4 Simulation Study

In this section, we demonstrate the effectiveness of our methods in simulation environments. We first conduct a range of numerical experiments in Section 3.4.1 to evaluate the performance of IQRL. In Section 3.4.2, we investigate a more challenging scenario with heterogeneous demonstrations to assess the performance of D-IQRL.

### 3.4.1 Performance of IQRL

To investigate whether the learned reward function can recover the ground-truth rewards with high accuracy, we conduct a simulation study using two simple environments. The first environment is a tabular GridWorld with a  $5 \times 5$  grid, the reward is 1 at the bottom-left,

bottom-right, and upper-right corners, and 0 otherwise. The action space consists of five possible actions: (up, down, left, right, stay). The optimal policy is derived through value iteration (Bellman 1957), then  $n = 30$  expert trajectories are generated with random initial states and a horizon of  $T = 20$  following the learned optimal policy. We apply a tabular version of our IQRL to learn the reward for each grid and normalize the rewards to range  $[0, 1]$ . More experiment details are provided in Appendix B.1. Figure 3.1 presents the learned rewards and the ground-truth rewards, along with their corresponding values. By comparing the top and bottom rows, we observe that the learned rewards effectively capture the pattern of the ground-truth rewards with satisfactory accuracy, and the corresponding values closely align with the ground-truth values. This result demonstrates the capability of our method to recover the underlying reward function using expert demonstrations.

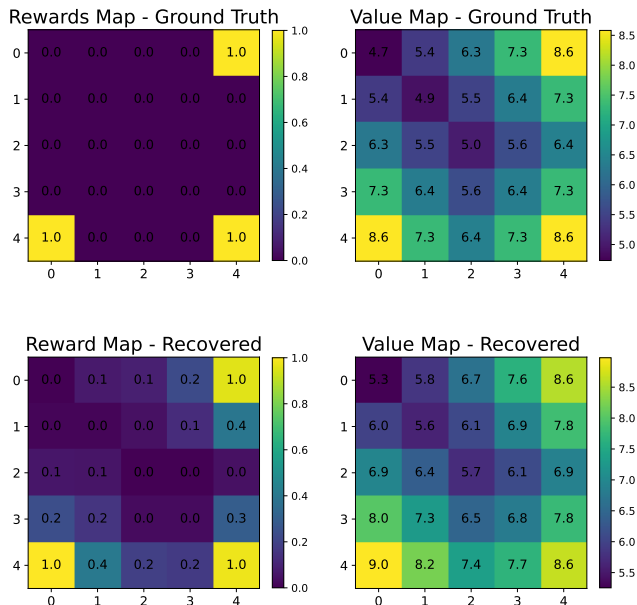


Figure 3.1: The heatmaps of ground-truth reward (top-left), value of the ground-truth reward (top-right), recovered reward (bottom-left), and value of the recovered reward (bottom-right) in  $5 \times 5$  GridWorld environment.

The second environment is a continuous 2D-Linear environment with binary action  $\mathcal{A} = \{0, 1\}$ . The initial state  $\mathbf{S}_0$  follows a standard bivariate normal distribution, and the transition dynamics are given by  $S_{t+1}^{(1)} = (3/4)(2A_t - 1)S_t^{(1)} + \varepsilon_t^{(1)}$ ,  $S_{t+1}^{(2)} = (3/4)(1 - 2A_t)S_t^{(2)} + \varepsilon_t^{(2)}$ , where  $S_t^{(i)}$  represents the  $i$ -th feature of  $\mathbf{S}_t$ ,  $\varepsilon_t^{(1)}$  and  $\varepsilon_t^{(2)}$  are independent normal random

variables with mean 0 and variance 0.04. This environment was previously used in Lockett et al. (2019) and Shi et al. (2021b). Here we slightly modify the reward to suit our IRL experiment. Specifically, the immediate reward is a weighted average of two components,  $R_{t+1} = \omega \cdot R_{t+1}^{(1)} + (1 - \omega) \cdot R_{t+1}^{(2)}$ , where  $R_{t+1}^{(1)} = 2S_{t+1}^{(1)} + S_{t+1}^{(2)}$ ,  $R_{t+1}^{(2)} = S_{t+1}^{(1)} - 3S_{t+1}^{(2)}$ . As a result, the reward function can also be expressed as a linear combination of two components,  $r_\omega(S_t, A_t) = \mathbb{E}\{R_{t+1}|S_t, A_t\} = \omega \cdot \mathbb{E}\{R_{t+1}^{(1)}|S_t, A_t\} + (1 - \omega) \cdot \mathbb{E}\{R_{t+1}^{(2)}|S_t, A_t\}$ , where  $\mathbb{E}\{R_{t+1}^{(1)}|S_t, A_t\} = (3/4)(2A_t - 1)(2S_t^{(1)} - S_t^{(2)})$ ,  $\mathbb{E}\{R_{t+1}^{(2)}|S_t, A_t\} = (3/4)(2A_t - 1)(S_t^{(1)} + 3S_t^{(2)})$ . We experiment on three different weight values,  $\omega \in \{0.3, 0.6, 0.8\}$ . For each value of  $\omega$ , we first learn an optimal policy using Proximal Policy Optimization (PPO) (Schulman et al. 2017), and then generate  $n$  synthetic expert trajectories with a horizon of  $T$  following the learned policy. The estimated weight  $\hat{\omega}$  is obtained by running the proposed IQRL algorithm with the constraint that the weights add up to 1. We fix  $n = 100$  and experiment on  $T \in \{10, 20\}$ . Additional experiment details can be found in Appendix B.1. The experiment is repeated 100 times, and the results are presented in Table 3.1. Overall, our method is capable of estimating the reward weight with satisfying accuracy, and the estimated rewards exhibit a strong correlation with the ground-truth rewards. Additionally, the learned optimal policy aligns with the actions taken in the expert demonstrations.

Table 3.1: Results of reward weight estimation. The estimated weight  $\hat{\omega}$ , the correlation between ground-truth rewards and recovered rewards, and the accuracy (ACC) of action prediction are reported (with standard errors in parentheses).

$T$	$\omega$	$\hat{\omega}$	$Corr(R, \hat{R})$	ACC
10	0.3	0.283 (0.021)	0.999 (0.002)	0.983 (0.005)
	0.6	0.572 (0.023)	0.993 (0.009)	0.979 (0.005)
	0.8	0.798 (0.026)	0.997 (0.005)	0.985 (0.004)
20	0.3	0.292 (0.015)	0.999 (0.001)	0.979 (0.003)
	0.6	0.581 (0.020)	0.996 (0.005)	0.977 (0.003)
	0.8	0.807 (0.014)	0.999 (0.001)	0.982 (0.003)

We further experiment on three classic control tasks from OpenAI gym (Brockman et al. 2016): (a) CartPole, which aims to balance the pole by pushing the cart to the left or right; (b) Acrobot, which tries to swing the free end of the chain above a given height; (c) LunarLander, which guides a landing module to a safe touchdown on the moon surface. In

this experiment, we focus on the learned policies and compare our method with other baseline IL or IRL methods, including (a) Behavior Cloning (BC), which casts IL as a supervised learning problem but does not leverage any dynamics information; (b) ValueDICE (Kostrikov et al. 2019), which minimizes a variational representation of the KL-divergence of occupancy measures between the expert and learned policies but has a biased gradient estimator and suffers from instability due to adversarial optimization; (c) AVRIL (Chan and van der Schaar 2021), which adopts a variational approach within the Bayesian IRL framework and learns a posterior distribution over the reward function; (d) IQ-Learn (Garg et al. 2021), which learns a single soft Q-function that implicitly represents both reward and policy. Additional implementation details can be found in Appendix B.1. This experiment primarily focuses on comparing the average returns of learned policies against the number of trajectories used for training. The results are visualized in Figure 3.2. In all three environments, the proposed IQRL demonstrates the capability to learn an optimal policy that achieves comparable performance to the expert when given a sufficient amount of data. Notably, both IQRL and IQ-Learn outperform other baseline algorithms in the more challenging LunarLander environment. However, when the sample size is small, IQRL exhibits better performance than IQ-Learn, as indicated by their performance on Acrobot and Cartpole. These results demonstrate the competitive performance of our method across a variety of tasks.

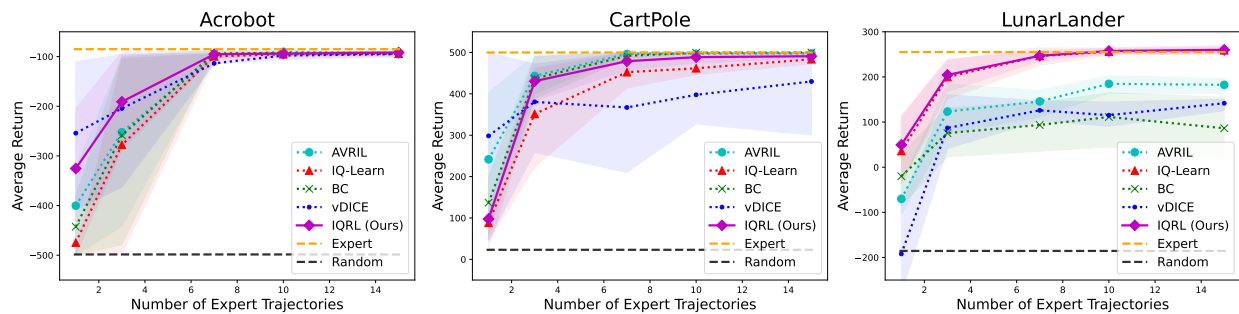


Figure 3.2: The average returns of learned policies using different imitation learning algorithms for Acrobot (left), CartPole (middle), and LunarLander (right). The number of training trajectories increases in the sequence of 1,3,7,10,15 for each algorithm.

### 3.4.2 Performance of D-IQRL

We also conducted a simulation study to evaluate the performance of the proposed D-IQRL. We utilize the 2D-Linear environment described in Section 3.4.1, and construct the reward

function for strategy  $k$  to be  $r^{(k)}(S_t, A_t) = r_{\omega_0}(S_t, A_t) + \omega_k^\top \phi_1(S_t, A_t)$ ,  $k = 1, \dots, K$ . In this formulation,  $r_{\omega_0}(S_t, A_t) = (\omega_0, 1 - \omega_0)^\top \phi_0(S_t, A_t)$  is the task reward function, where  $\phi_0 : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}^2$  is the task feature map used in the previous simulation,  $\phi_1 : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}^{d_1}$  represents the strategy-level feature map, and  $K$  is the number of strategies. We consider  $K = 3$  strategies, the strategy-level feature map is given by  $\phi_1(S_t, A_t) = S_t^{(2)}$ . The weight for the task reward is set to  $\omega_0 = 0.8$ , and the weights for the strategy-specific rewards are  $(\omega_1, \omega_2, \omega_3) = (1, 0.2, -1)$ . To generate heterogeneous expert demonstration data, we first train the optimal policy for each strategy using PPO (Schulman et al. 2017), then generate 100 trajectories for each strategy. The ground-truth rewards under strategy  $k$  are obtained from a Gaussian distribution centered at  $r^{(k)}(S_t, A_t)$  with a standard deviation of 0.2. More experiment details are described in Appendix B.1. Figure 3.3 visualizes the ground-truth rewards and recovered rewards, where each row corresponds to an action. The subfigures in the odd columns depict the ground-truth rewards for each strategy, while the subfigures in the even columns represent the recovered rewards. The plots indicate that our D-IQRL is able to capture the patterns of the ground-truth reward for each strategy.

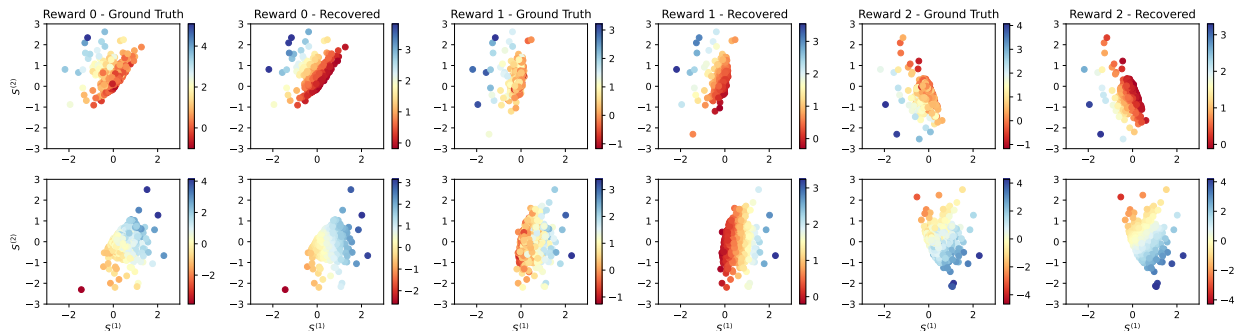


Figure 3.3: The recovered rewards and ground-truth rewards for each strategy. In each subplot, the x-axis and y-axis represent the two dimensions of the state variable. The top row corresponds to the reward function for action 0, and the bottom row corresponds to that of action 1.

To further demonstrate that the proposed algorithm can accommodate heterogeneity in expert demonstration while leveraging common task information, we compare it with two alternative training approaches: (a) local IQRL, which trains IQRL for each strategy without sharing information, and (b) centralized IQRL, which pools the data from all strategies and learns a single reward function. We compare the three training approaches based on three criteria: (a) the correlation between the true rewards and recovered rewards, (b) the accuracy of action prediction, and (c) the obtained values under the learned policies. Table

3.2 summarizes the results for three training approaches. Based on the overall performance evaluated on the pooled data, local IQRL shows superior performance compared to centralized IQRL, even with a smaller sample size due to stratification. This highlights the benefit of taking heterogeneity into account. Furthermore, our proposed D-IQRL outperforms local IQRL in all three criteria, demonstrating the benefit of knowledge sharing among strategies. To gain a deeper understanding of its performance, we examine each individual strategy. For strategy 1, which lies closer to the centroid of all strategy rewards, the results from all three approaches are quite similar. However, for strategies 0 and 2, which deviate more from the task reward, both local and distilled training outperform centralized training in terms of action matching rate and policy value, with the distilled approach slightly outperforming local training. It is worth noting that for strategy 2, while local training and distilled training exhibit similar action matching rates, the distilled training achieves a significantly higher reward correlation compared to local training. This indicates the advantage of leveraging common knowledge among strategies.

Table 3.2: Comparison of the three training approaches. The correlation between true rewards and recovered rewards, the accuracy (ACC) of action prediction, and the obtained values under learned policies are reported (with standard errors in parentheses).

	Strategy	Centralized IQRL	Local IQRL	<b>Distilled IQRL</b>
$Corr(R, \hat{R})$	0	0.552 (0.031)	0.793 (0.019)	0.807 (0.016)
	1	0.687 (0.023)	0.668 (0.049)	0.662 (0.018)
	2	0.577 (0.027)	0.516 (0.158)	0.816 (0.010)
	pooled	0.594 (0.025)	0.659 (0.072)	0.772 (0.012)
ACC	0	0.871 (0.012)	0.956 (0.005)	0.955 (0.005)
	1	0.956 (0.010)	0.970 (0.005)	0.972 (0.005)
	2	0.825 (0.014)	0.971 (0.003)	0.971 (0.004)
	pooled	0.884 (0.006)	0.965 (0.003)	0.966 (0.003)
$V^{\hat{\pi}}$	0	9.747 (0.188)	10.364 (0.108)	10.370 (0.109)
	1	8.869 (0.127)	8.942 (0.124)	8.456 (0.122)
	2	7.552 (0.200)	9.212 (0.108)	9.716 (0.107)
	pooled	8.723 (0.125)	9.506 (0.111)	9.514 (0.029)

### 3.5 Real Data Application

We apply the proposed IQRL algorithm to the Medical Information Mart for Intensive Care (MIMIC-III) database (Johnson et al. 2016), which includes patient trajectories in intensive care units. The target dataset extracted from the database contains the physiological features and therapeutic interventions at one-day intervals. We assess the performance of our method in both the two- and four-action settings, focusing on the choice of whether to put the patient on a ventilator, and whether to combine ventilation with antibiotic therapy. Since there is no recorded notion of reward, we evaluate the performance in terms of action matching against a held-out test set of demonstrations with cross-validation. Action matching serves as a primary quantitative performance metric for assessing whether the algorithm is able to recommend treatments that align with clinical practice. We compare our proposed method with several baseline IRL/IL methods used in the numerical experiments. Further details regarding the dataset and implementation can be found in Appendix B.2. Table 3.3 summarizes the action-matching metrics for optimal policies obtained through different methods. Our IQRL algorithm outperforms other baseline methods across all three metrics, especially in the more challenging 4-action setting.

Table 3.3: Comparison of policy performance of different imitation learning algorithms on the MIMIC-III dataset, evaluated by the quality of action matching against a held-out test set of demonstrations with cross-validation. The accuracy (ACC), the area under the receiving operator characteristic curve (AUC), and the average precision score (APS) of action prediction are reported (with standard error in parenthesis).

Metric	Ventilator			Ventilator + Antibiotics		
	ACC	AUC	APS	ACC	AUC	APS
VDICE	0.669 (0.018)	0.728 (0.024)	0.620 (0.030)	0.411 (0.011)	0.658 (0.009)	0.409 (0.008)
BC	0.785 (0.008)	0.863 (0.006)	0.797 (0.013)	0.525 (0.010)	0.772 (0.006)	0.551 (0.006)
AVRIL	0.774 (0.009)	0.859 (0.004)	0.798 (0.007)	0.517 (0.006)	0.767 (0.004)	0.542 (0.005)
IQ-Learn	0.786 (0.007)	0.869 (0.004)	0.808 (0.008)	0.531 (0.007)	0.774 (0.003)	0.552 (0.004)
<b>IQRL</b>	0.798 (0.003)	0.876 (0.004)	0.817 (0.008)	0.541 (0.002)	0.782 (0.003)	0.567 (0.003)

Our proposed method offers the advantage of learning an explicit reward that is portable and flexible in specifying the expression for the reward function, allowing us to better understand the intent of the experts. To gain further insights into the learned reward function for this dataset, we delve into how the learned reward varies with respect to a specific state feature for an otherwise average patient and also investigate how different actions affect the

reward. For illustration, we focus on two state features: blood pH value and partial pressure of carbon dioxide (PaCO<sub>2</sub>). The corresponding reward functions are visualized in Figure 3.4. For blood pH, the normal range is between 7.35 and 7.45. When the pH value deviates from this range, the use of mechanical ventilation becomes beneficial in maintaining appropriate levels of carbon dioxide and oxygen, leading to higher rewards than without ventilator. Similarly, a PaCO<sub>2</sub> level over 50 mm Hg is considered a concerning condition, and the initiation of mechanical ventilation is often recommended. The learned reward function also indicates a preference for managing mechanical ventilation when PaCO<sub>2</sub> exceeds this threshold. Therefore, the reward function depicted in Figure 3.4 aligns with clinical practice.

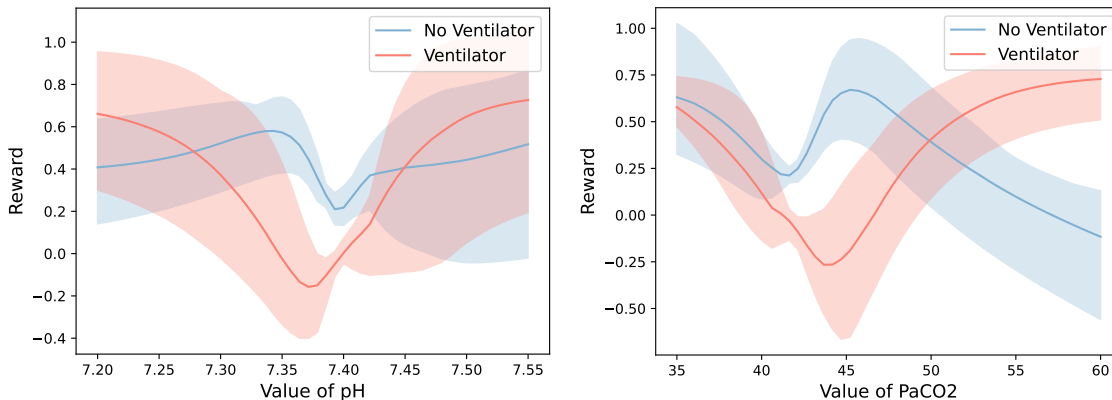


Figure 3.4: The state-action reward for an otherwise average patient as their blood pH (left) or PaCO<sub>2</sub> level (right) varies.

To better interpret the reward function, we also consider the reward as a linear function of pre-specified reward components and use the proposed algorithm to learn the coefficients. We adopt the seven reward components proposed by Prasad et al. (2017) with slight modifications.

Specifically, the reward components are given by

$$\begin{aligned}
r_1(s_t, a_t, s_{t+1}) &= \frac{1}{2} \sum_{v \in \mathcal{V}} \mathbb{1}\{v_{\min} \leq s_{t+1}^v \leq v_{\max}\}, \\
r_2(s_t, a_t, s_{t+1}) &= - \max \left( 0, \max_{v \in \mathcal{V}} \left( \frac{|s_{t+1}^v - s_t^v|}{s_t^v} - 0.2 \right) \right) \\
r_3(s_t, a_t, s_{t+1}) &= \mathbb{1}\{s_{t+1}^{\text{vent on}} = 0\} \mathbb{1}\{s_t^{\text{vent on}} = 1\}, \\
r_4(s_t, a_t, s_{t+1}) &= \mathbb{1}\{s_{t+1}^{\text{vent on}} = 0\} \mathbb{1}\{s_t^{\text{vent on}} = 0\}, \\
r_5(s_t, a_t, s_{t+1}) &= - \mathbb{1}\{s_{t+1}^{\text{vent on}} = 0\} \cdot \sum_{v \in \mathcal{V}^{exp}} (1 - \mathbb{1}\{v_{\min} \leq s_{t+1}^v \leq v_{\max}\}), \\
r_6(s_t, a_t, s_{t+1}) &= - \mathbb{1}\{s_{t+1}^{\text{vent on}} = 1\} \mathbb{1}\{s_t^{\text{vent on}} = 1\}, \\
r_7(s_t, a_t, s_{t+1}) &= - \mathbb{1}\{s_{t+1}^{\text{vent on}} = 1\} \mathbb{1}\{s_t^{\text{vent on}} = 0\}.
\end{aligned} \tag{3.12}$$

Here,  $\mathcal{V}$  is a collection of the three vital signs: heart rate, respiratory rate, and arterial pH.  $\mathcal{V}^{exp}$  corresponds to features related to extubation conditions (i.e., FiO2, SpO2, and PEEP). These reward components capture various aspects of ventilator management:  $r_1$  measures physiological stability and yields a higher value when the vital measurements are within the expected ranges;  $r_2$  is a negative feedback signal that penalizes sharp changes in consecutive vital measurements;  $r_3$  yields a positive reward at the time of successful extubation;  $r_4$  assigns a positive reward for each additional day without a ventilator;  $r_5$  acts as a negative feedback signal when the criteria on extubation-related features are not met during extubation;  $r_6$  penalizes additional days spent on the ventilator;  $r_7$  imposes a penalty due to reintubation. The reward function is constructed as a linear combination of the seven reward components:  $r(s_t, a_t, s_{t+1}) = \sum_{j=1}^7 \omega_j \cdot r_j(s_t, a_t, s_{t+1})$ . To construct a reward function for policy learning, instead of specifying the weights by hand, one can apply IRL techniques to derive the reward weights. By applying the proposed algorithm with the constraint that the reward coefficients sum up to 1, we obtained the following weights: [0.205, 0, 0, 0.206, 0.015, 0, 0.574]. The learned reward indicates that the penalty for reintubation plays an important role in guiding ventilator decision-making. Additionally, maintaining the vital signs within a normal range and reducing patients' reliance on mechanical ventilation are also crucial factors to consider.

The previous results are obtained by pooling all expert demonstrations from different types of care units together and learning a single reward function. However, patients admitted to different care units often exhibit different conditions. Previous studies have recognized the heterogeneity among patients across different ICU types (Alves et al. 2018; Suresh et al. 2018). To account for potential heterogeneity in treatment decision-making, we apply the proposed D-IQRL to learn the reward function for each specific ICU type, and compare the performance

with local and centralized training to demonstrate the benefit of sharing information across strategies. Following the grouping criteria used in Qian et al. (2023), we consider three types of ICUs: (a) MICU (Medical ICU), a specialized ICU for critically ill patients with medical conditions; (b) SICU/TSICU (Surgical / Trauma Surgical ICU), specialized ICUs for critically ill patients who had major surgeries or traumatic injuries; and (c) CCU/CSRU (Cardiac Care Unit / Cardiovascular Surgical Recovery Unit), specialized ICUs for critically ill patients with cardiovascular conditions. We focus on ventilator management and evaluate the performance on different ICU types. The results are presented in Table 3.4. For simplicity, we only report the accuracy of action matching. Overall, the results obtained from the three training approaches show close performance, indicating a relatively low level of heterogeneity in ventilator management based on the observed data. Despite this, we still observe that the D-IQRL method slightly outperforms localized training in most cases, demonstrating the benefit of information sharing. Furthermore, the distilled approach not only achieves comparable performance to centralized training for MICU and SICU/TSICU but also yields higher accuracy for CCU/CSRU. These results demonstrate the effectiveness of our distilled IQRL in capturing the patterns of heterogeneous strategies in the expert demonstrations.

Table 3.4: Comparison of policy performance of the three training approaches on the MIMIC-III dataset, evaluated by the site-wise accuracy (ACC) of action prediction against a held-out test set of demonstrations with cross-validation.

Site	Centralized IQRL	Local IQRL	<b>Distilled IQRL</b>
MICU	0.798 (0.009)	0.795 (0.013)	0.796 (0.011)
CCU/CSRU	0.817 (0.010)	0.822 (0.006)	0.820 (0.003)
SICU/TSICU	0.782 (0.005)	0.778 (0.005)	0.781 (0.008)
pooled	0.798 (0.003)	0.796 (0.004)	0.797 (0.005)

To gain a better understanding of the variations in the learned reward functions, we again consider a linear reward model that encompasses the seven reward components defined in (3.12). In Figure 3.5, we present the learned reward weights, with each ICU group represented by a different color. The plot reveals a certain level of heterogeneity in the learned weights, with slight differences observed between CCU/CSRU and MICU/SICU/TSICU. Specifically, for CCU/CSRU, the learned reward function assigns higher weights to  $r_1$  and  $r_2$  and a lower weight to  $r_7$ . This implies that the experts’ decision-making pays more attention to the stability of vital signs for patients with cardiovascular conditions.

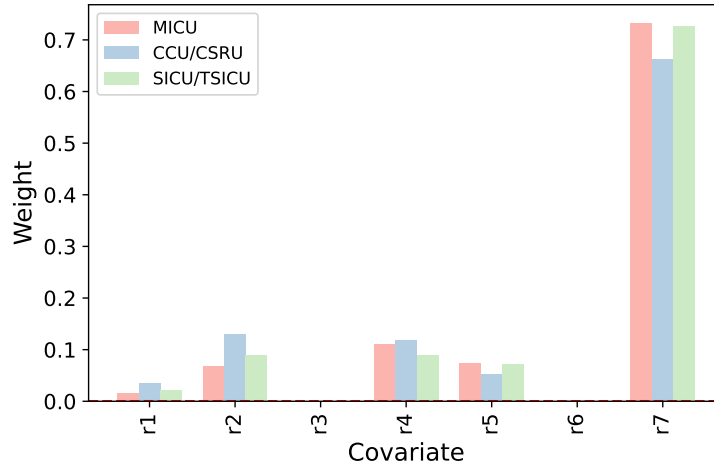


Figure 3.5: The average rewards weights for different care unit types obtained from D-IQRL.

In summary, the above results together demonstrate the effectiveness of our method in accommodating the specific characteristics of different ICU types while leveraging shared knowledge.

### 3.6 Discussion and Future Work

In this work, we present a novel offline IRL method called IQRL, which provides a stable and efficient solution to recovering the reward function for imitation learning. Our proposed method achieves appealing performance and also offers a reliable estimate of the reward function. Furthermore, we propose an extension called D-IQRL to handle potentially heterogeneous demonstrations. This method incorporates reward distillation to facilitate information sharing among different strategies. One limitation of our method is its reliance on known strategy labels, which may not always be available. In light of this, one direction for future work is to incorporate cohort discovery methods or latent class models to identify subgroups within heterogeneous demonstrations. Besides, we believe delving into the theoretical property of the proposed algorithms to better understand their performance can also be an interesting direction worthy of future work.

## REFERENCES

- Alves, T., Laender, A., Veloso, A., and Ziviani, N. (2018). Dynamic prediction of icu mortality risk using domain adaptation. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 1328–1336. IEEE.
- Barto, A. G., Sutton, R. S., and Anderson, C. W. (1983). Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(5):834–846.
- Bellman, R. (1957). A markovian decision process. *Journal of Mathematics and Mechanics*, pages 679–684.
- Boyd, S., Boyd, S. P., and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge university press.
- Bradley, R. C. (2005). Basic properties of strong mixing conditions. a survey and some open questions. *Probability Surveys*, 2:107–144.
- Bradtke, S. J. and Barto, A. G. (1996). Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 22(1):33–57.
- Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. (2016). Openai gym. *arXiv preprint arXiv:1606.01540*.
- Chan, A. J. and van der Schaar, M. (2021). Scalable bayesian inverse reinforcement learning. *International Conference on Learning Representations*.
- Che, Z., Purushotham, S., Cho, K., Sontag, D., and Liu, Y. (2018). Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1):6085.
- Chen, L., Paleja, R., Ghuy, M., and Gombolay, M. (2020). Joint goal and strategy inference across heterogeneous demonstrators via reward network distillation. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, pages 659–668.
- Coronato, A., Naeem, M., De Pietro, G., and Paragliola, G. (2020). Reinforcement learning for intelligent healthcare applications: A survey. *Artificial Intelligence in Medicine*, 109:101964.
- Dai, B., Nachum, O., Chow, Y., Li, L., Szepesvári, C., and Schuurmans, D. (2020). Coindice: Off-policy confidence interval estimation. *Advances in Neural Information Processing Systems*, 33:9398–9411.
- De Boor, C. (1976). *Splines as linear combinations of B-splines: A survey*. University of Wisconsin-Madison. Mathematics Research Center.
- Diggle, P. and Kenward, M. G. (1994). Informative drop-out in longitudinal data analysis. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 43(1):49–73.

- Dong, L., Laber, E., Goldberg, Y., Song, R., and Yang, S. (2020). Ascertaining properties of weighting in the estimation of optimal treatment regimes under monotone missingness. *Statistics in Medicine*, 39(25):3503–3520.
- Finn, C., Levine, S., and Abbeel, P. (2016). Guided cost learning: Deep inverse optimal control via policy optimization. In *International Conference on Machine Learning*, pages 49–58.
- Fu, J., Luo, K., and Levine, S. (2017). Learning robust rewards with adversarial inverse reinforcement learning. In *International Conference on Learning Representations*.
- Fujimoto, S., Meger, D., and Precup, D. (2019). Off-policy deep reinforcement learning without exploration. In *International Conference on Machine Learning*, pages 2052–2062. PMLR.
- Garg, D., Chakraborty, S., Cundy, C., Song, J., and Ermon, S. (2021). Iq-learn: Inverse soft-q learning for imitation. *Advances in Neural Information Processing Systems*, 34:4028–4039.
- Ghasemipour, S. K. S., Zemel, R., and Gu, S. (2020). A divergence minimization perspective on imitation learning methods. In *Conference on Robot Learning*, pages 1259–1277. PMLR.
- Gleave, A. and Habryka, O. (2018). Multi-task maximum entropy inverse reinforcement learning. *arXiv preprint arXiv:1805.08882*.
- Gleave, A. and Toyer, S. (2022). A primer on maximum causal entropy inverse reinforcement learning. *arXiv preprint arXiv:2203.11409*.
- Goldberg, Y. and Kosorok, M. R. (2012). Q-learning with censored data. *Annals of Statistics*, 40(1):529.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11):139–144.
- Gottesman, O., Johansson, F., Komorowski, M., Faisal, A., Sontag, D., Doshi-Velez, F., and Celi, L. A. (2019). Guidelines for reinforcement learning in healthcare. *Nature Medicine*, 25(1):16–18.
- Gotts, J. E. and Matthay, M. A. (2016). Sepsis: pathophysiology and clinical management. *BMJ*, 353.
- Graham, J. W. et al. (2009). Missing data analysis: Making it work in the real world. *Annual Review of Psychology*, 60(1):549–576.
- Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. (2017). Reinforcement learning with deep energy-based policies. *arXiv preprint arXiv:1702.08165*.
- Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning*, pages 1861–1870. PMLR.

- Hansen, L. P. (1982). Large sample properties of generalized method of moments estimators. *Econometrica: Journal of the Econometric Society*, pages 1029–1054.
- Harutyunyan, H., Khachatryan, H., Kale, D. C., Ver Steeg, G., and Galstyan, A. (2019). Multitask learning and benchmarking with clinical time series data. *Scientific Data*, 6(1):96.
- Hesterberg, T. C. (1988). *Advances in importance sampling*. Stanford University.
- Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.
- Ho, J. and Ermon, S. (2016). Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, pages 4565–4573.
- Huang, J. Z. et al. (1998). Projection estimation in multiple regression with application to functional anova models. *Annals of Statistics*, 26(1):242–272.
- Jabaley, C. S., Groff, R. F., Sharifpour, M., Raikhelkar, J. K., and Blum, J. M. (2018). Modes of mechanical ventilation vary between hospitals and intensive care units within a university healthcare system: a retrospective observational study. *BMC Research Notes*, 11:1–8.
- Jarrett, D., Yoon, J., Bica, I., Qian, Z., Ercole, A., and van der Schaar, M. (2021). Clairvoyance: A pipeline toolkit for medical time series. In *International Conference on Learning Representations*.
- Jiang, N. and Li, L. (2016). Doubly robust off-policy value evaluation for reinforcement learning. In *International Conference on Machine Learning*, pages 652–661. PMLR.
- Johnson, A. E., Pollard, T. J., Shen, L., Lehman, L.-w. H., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Anthony Celi, L., and Mark, R. G. (2016). Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3(1):1–9.
- Kallus, N. and Uehara, M. (2022). Efficiently breaking the curse of horizon in off-policy evaluation with double reinforcement learning. *Operations Research*.
- Kam, H. J. and Kim, H. Y. (2017). Learning representations for the early detection of sepsis with deep neural networks. *Computers in Biology and Medicine*, 89:248–255.
- Kim, J. K. and Yu, C. L. (2011). A semiparametric estimation of mean functionals with nonignorable missing data. *Journal of the American Statistical Association*, 106(493):157–165.
- Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Kober, J., Bagnell, J. A., and Peters, J. (2013). Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274.

- Komorowski, M., Celi, L. A., Badawi, O., Gordon, A. C., and Faisal, A. A. (2018). The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. *Nature Medicine*, 24(11):1716–1720.
- Kostrikov, I., Nachum, O., and Tompson, J. (2019). Imitation learning via off-policy distribution matching. In *International Conference on Learning Representations*.
- Lagoudakis, M. G. and Parr, R. (2003). Least-squares policy iteration. *The Journal of Machine Learning Research*, 4:1107–1149.
- Le, H., Voloshin, C., and Yue, Y. (2019). Batch policy learning under constraints. In *International Conference on Machine Learning*, pages 3703–3712. PMLR.
- 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*.
- Liu, D. C. and Nocedal, J. (1989). On the limited memory bfgs method for large scale optimization. *Mathematical Programming*, 45(1):503–528.
- Liu, Q., Li, L., Tang, Z., and Zhou, D. (2018). Breaking the curse of horizon: Infinite-horizon off-policy estimation. *Advances in Neural Information Processing Systems*, 31.
- Luckett, D. J., Laber, E. B., Kahkoska, A. R., Maahs, D. M., Mayer-Davis, E., and Kosorok, M. R. (2019). Estimating dynamic treatment regimes in mobile health using v-learning. *Journal of the American Statistical Association*.
- McLeish, D. L. (1974). Dependent central limit theorems and invariance principles. *The Annals of Probability*, 2(4):620–628.
- 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*.
- Murphy, S. A. (2003). Optimal dynamic treatment regimes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(2):331–355.
- Nachum, O., Chow, Y., Dai, B., and Li, L. (2019). Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections. *Advances in Neural Information Processing Systems*, 32.
- Newey, W. K. (1994). Kernel estimation of partial means and a general variance estimator. *Econometric Theory*, 10(2):1–21.
- Newey, W. K. and McFadden, D. (1994). Large sample estimation and hypothesis testing. *Handbook of Econometrics*, 4:2111–2245.
- Peng, X., Ding, Y., Wihl, D., Gottesman, O., Komorowski, M., Lehman, L.-w. H., Ross, A., Faisal, A., and Doshi-Velez, F. (2018). Improving sepsis treatment strategies by combining deep and kernel-based reinforcement learning. In *AMIA Annual Symposium Proceedings*, volume 2018, page 887. American Medical Informatics Association.

- Pirracchio, R., Petersen, M. L., Carone, M., Rigon, M. R., Chevret, S., and van der Laan, M. J. (2015). Mortality prediction in intensive care units with the super icu learner algorithm (sicula): a population-based study. *The Lancet Respiratory Medicine*, 3(1):42–52.
- Prasad, N., Cheng, L.-F., Chivers, C., Draugelis, M., and Engelhardt, B. E. (2017). A reinforcement learning approach to weaning of mechanical ventilation in intensive care units. In *33rd Conference on Uncertainty in Artificial Intelligence*.
- Precup, D. (2000). Eligibility traces for off-policy policy evaluation. *Computer Science Department Faculty Publication Series*, page 80.
- Purushotham, S., Meng, C., Che, Z., and Liu, Y. (2018). Benchmarking deep learning models on large healthcare datasets. *Journal of Biomedical Informatics*, 83:112–134.
- Puterman, M. L. (1994). Markov decision processes: Discrete stochastic dynamic programming.
- Qian, J., Qin, R., Hong, L., Shi, Y., Yuan, H., Zhang, B., Nie, W., Li, Y., and Han, B. (2023). Characteristics and clinical outcomes of patients with lung cancer requiring icu admission: a retrospective analysis based on the mimic-iii database. *Emergency Cancer Care*, 2(1):1.
- Raffin, A. (2020). Rl baselines3 zoo. <https://github.com/DLR-RM/rl-baselines3-zoo>.
- Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., and Dormann, N. (2021). Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8.
- Raghu, A., Komorowski, M., Celi, L. A., Szolovits, P., and Ghassemi, M. (2017). Continuous state-space models for optimal sepsis treatment: a deep reinforcement learning approach. In *Machine Learning for Healthcare Conference*, pages 147–163. PMLR.
- Rotnitzky, A. and Robins, J. (1997). Analysis of semi-parametric regression models with non-ignorable non-response. *Statistics in Medicine*, 16(1):81–102.
- Rubinstein, R. Y. (1981). *Simulation and the Monte Carlo Method*. John Wiley & Sons, Inc.
- Rusu, A. A., Colmenarejo, S. G., Gulcehre, C., Desjardins, G., Kirkpatrick, J., Pascanu, R., Mnih, V., Kavukcuoglu, K., and Hadsell, R. (2015). Policy distillation. *arXiv preprint arXiv:1511.06295*.
- Sallab, A. E., Abdou, M., Perot, E., and Yogamani, S. (2017). Deep reinforcement learning framework for autonomous driving. *Electronic Imaging*, 2017(19):70–76.
- Scherpf, M., Gräßer, F., Malberg, H., and Zaunseder, S. (2019). Predicting sepsis with a recurrent neural network using the mimic iii database. *Computers in Biology and Medicine*, 113:103395.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

- 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.
- Shao, J. and Wang, L. (2016). Semiparametric inverse propensity weighting for nonignorable missing data. *Biometrika*, 103(1):175–187.
- Shi, C., Wan, R., Chernozhukov, V., and Song, R. (2021a). Deeply-debiased off-policy interval estimation. In *International Conference on Machine Learning*, pages 9580–9591. PMLR.
- Shi, C., Wan, R., Song, R., Lu, W., and Leng, L. (2020). Does the markov decision process fit the data: Testing for the markov property in sequential decision making. In *International Conference on Machine Learning*, pages 8807–8817. PMLR.
- Shi, C., Zhang, S., Lu, W., and Song, R. (2021b). Statistical inference of the value function for reinforcement learning in infinite-horizon settings. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*.
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489.
- Singer, M., Deutschman, C. S., Seymour, C. W., Shankar-Hari, M., Annane, D., Bauer, M., Bellomo, R., Bernard, G. R., Chiche, J.-D., Coopersmith, C. M., et al. (2016). The third international consensus definitions for sepsis and septic shock (sepsis-3). *JAMA*, 315(8):801–810.
- Sonabend, A., Lu, J., Celi, L. A., Cai, T., and Szolovits, P. (2020). Expert-supervised reinforcement learning for offline policy learning and evaluation. In *Advances in Neural Information Processing Systems*, volume 33, pages 18967–18977.
- Sun, M., Baron, J., Dighe, A., Szolovits, P., Wunderink, R. G., Isakova, T., and Luo, Y. (2019). Early prediction of acute kidney injury in critical care setting using clinical notes and structured multivariate physiological measurements. *MedInfo*, 264:368–72.
- Sun, X., Panda, R., Feris, R., and Saenko, K. (2020). Adashare: Learning what to share for efficient deep multi-task learning. *Advances in Neural Information Processing Systems*, 33:8728–8740.
- Suresh, H., Gong, J. J., and Guttag, J. V. (2018). Learning tasks for multitask learning: Heterogenous patient populations in the icu. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 802–810.
- Syed, U., Bowling, M., and Schapire, R. E. (2008). Apprenticeship learning using linear programming. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1032–1039.

- Teh, Y., Bapst, V., Czarnecki, W. M., Quan, J., Kirkpatrick, J., Hadsell, R., Heess, N., and Pascanu, R. (2017). Distral: Robust multitask reinforcement learning. *Advances in Neural Information Processing Systems*, 30.
- Uehara, M., Huang, J., and Jiang, N. (2020). Minimax weight and q-function learning for off-policy evaluation. In *International Conference on Machine Learning*, pages 9659–9668. PMLR.
- Vincent, J. L., Moreno, R., Takala, J., Willatts, S., De Mendonça, A., Bruining, H., Reinhart, C. K., Suter, P. M., and Thijs, L. G. (1996). The sofa (sepsis-related organ failure assessment) score to describe organ dysfunction/failure. *Intensive Care Medicine*, 22(7):707–710.
- Wang, L., Zhang, W., He, X., and Zha, H. (2018). Supervised reinforcement learning with recurrent neural network for dynamic treatment recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2447–2456.
- Wang, Z., Schaul, T., Hessel, M., Hasselt, H., Lanctot, M., and Freitas, N. (2016). Dueling network architectures for deep reinforcement learning. In *International Conference on Machine Learning*, pages 1995–2003. PMLR.
- Wulfmeier, M., Ondruska, P., and Posner, I. (2015). Maximum entropy deep inverse reinforcement learning. *arXiv preprint arXiv:1507.04888*.
- Zeng, S., Li, C., Garcia, A., and Hong, M. (2022). Maximum-likelihood inverse reinforcement learning with finite-time guarantees. *Advances in Neural Information Processing Systems*.
- Zhang, R., Dai, B., Li, L., and Schuurmans, D. (2020a). Gendice: Generalized offline estimation of stationary values. *arXiv preprint arXiv:2002.09072*.
- Zhang, Z., Zheng, B., and Liu, N. (2020b). Individualized fluid administration for critically ill patients with sepsis with an interpretable dynamic treatment regimen model. *Scientific Reports*, 10(1):1–9.
- 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.
- Ziebart, B. D., Bagnell, J. A., and Dey, A. K. (2010). Modeling interaction via the principle of maximum causal entropy. In *International Conference on Machine Learning*.
- Ziebart, B. D., Maas, A. L., Bagnell, J. A., Dey, A. K., et al. (2008). Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pages 1433–1438. Chicago, IL, USA.
- Zimmerman, L. P., Reyfman, P. A., Smith, A. D., Zeng, Z., Kho, A., Sanchez-Pinto, L. N., and Luo, Y. (2019). Early prediction of acute kidney injury following icu admission using a multivariate panel of physiological measurements. *BMC Medical Informatics and Decision Making*, 19(1):1–12.

## APPENDICES

# Appendix A

## Off-Policy Evaluation with Nonignorable Missing Data

### A.1 Sensitivity Analysis

We also conducted a sensitivity analysis to evaluate the robustness of the IPW adjustment method under model misspecification. Specifically, we aimed to assess the performance of the proposed approach in the presence of (a) a misspecified instrumental variable and (b) a misspecified missing propensity model. The following propensity model is considered

$$\lambda_3(S_t, A_t, R_{t+1}, S_{t+1}) = \{1 + \exp(8.5 + 0.5S_t^{(1)} - R_{t+1} - 0.1R_{t+1}^2)\}^{-1}.$$

In this scenario, the variable  $S_t^{(2)}$  still serves as an instrumental variable, while the logit of missing propensity becomes nonlinear in outcome variables  $(R_{t+1}, S_{t+1})$ . We utilize this model to assess the robustness of our proposed method to potential misspecifications. Additionally, we investigate the impact of misspecifying the instrumental variable. Specifically, we examine the case where we use  $S_t^{(1)}$  instead of  $S_t^{(2)}$  as the instrumental variable when fitting the dropout model. Here we focus on the scenario of  $T = 25$  and target policy of  $\pi_1$ . The results are summarized in Table A.1, which indicate that the IPW estimator can still provide more reliable value estimation than the complete-case estimator even under slight model misspecification. However, if the instrumental variable is misspecified, the proposed method may fail. Based on this observation, we suggest careful selection of the instrumental variable based on subject matter knowledge or through ad hoc data diagnosis.

Table A.1: Results of value estimates and 95% confidence intervals for policy  $\pi_1$  under dropout propensity  $\lambda_3$ . The average bias, MSE values, ECP, and AL are reported for each estimator (with standard error in parenthesis). The suffix (IV $\times$ ) indicates misspecification of instrumental variable, and the suffix (mod $\times$ ) indicates misspecification of the parametric model for outcome variables.

$n$	DROPOUT	METHOD	BIAS	MSE	ECP	AL
500	NO DROPOUT	CC	-0.004 (1.236)	1.525	0.932	4.740 (0.150)
	MNAR	CC	-2.340 (1.693)	8.330	0.736	7.008 (0.707)
		IPW	0.201 (2.104)	4.451	0.948	8.619 (2.236)
		IPW(IV $\times$ )	-2.441 (1.845)	9.348	0.699	7.058 (1.224)
		IPW(MOD $\times$ )	0.031 (2.126)	4.503	0.932	8.693 (2.416)
1000	NO DROPOUT	CC	-0.011 (0.728)	0.528	0.976	3.344 (0.078)
	MNAR	CC	-2.339 (1.226)	6.971	0.552	4.893 (0.299)
		IPW	0.255 (1.566)	2.507	0.928	6.032 (0.932)
		IPW(IV $\times$ )	-2.413 (1.405)	7.787	0.472	4.927 (0.670)
		IPW(MOD $\times$ )	0.040 (1.736)	3.004	0.908	6.041 (1.064)

## A.2 Additional Experimental Details

In this section, we provide more details on the experiments and our implementation.

**Data generation** To generate complete data with  $n$  trajectories, we first sample  $n$  initial states from the reference distribution  $\mathbb{G}$ , and then generate the action, next state, and reward following the generative model introduced in Section 2.5. This process is repeated until reaching the maximum horizon  $T$ . To generate incomplete data, we first calculate the dropout probability  $\lambda_{i,t}$  at each step using the dropout model  $\lambda_j(\cdot)$  defined in Section 2.5, this corresponds to the probability of subject  $i$  dropping out after taking action  $A_t$ . Given the dropout probability, we sample the response indicator  $\eta_{i,t+1}$  from a Bernoulli distribution with mean  $(1 - \lambda_{i,t})$ . To control the overall missing rate, we also set a no-dropout period of two steps, i.e.,  $\eta_{i,0} = \eta_{i,1} = 1$ . After the second step, the dropout probability is applied and a trajectory will terminate when the response indicator  $\eta_{i,t}$  turns 0.

**Complete-Case (CC) estimator** The OPE step of Shi et al. (2021b) approximates the Q-function with linear sieves,  $Q^\pi(s, a) \approx \Phi_L^\top(s)\beta_{\pi,a}$ , where  $\Phi_L(\cdot) = \{\phi_{L,1}(\cdot), \dots, \phi_{L,L}(\cdot)\}^\top$  is a vector consisting of  $L$  spline bases. In our implementation, we first scale the state variables onto  $[0, 1]$  and then construct 6 cubic B-spline bases for each dimension, where the knots are

placed at equally-spaced quantiles of the transformed state variables. To avoid extrapolation of the basis function, three repeated knots are placed on the boundary. The tensor product of the basis for each dimension is used to construct the final basis, hence  $L = 36$ . The number of basis functions  $L$  is allowed to grow with the sample size to reduce the approximation error. For a fair comparison, here we fix  $L = 36$  throughout the experiments despite the sample sizes. The CC estimator of the Q-function parameter  $\beta_\pi^*$  is given in (2.7). The matrix inversion of  $\widehat{\Sigma}_\pi \in \mathbb{R}^{mL \times mL}$  tends to be unstable when  $mL$  is large, so we add a small ridge penalty with weight  $10^{-5}$  to improve the stability. Given  $\widehat{\beta}_\pi$ , the value function can be calculated as  $\widehat{V}^\pi(s) = \mathbf{U}_\pi^\top(s)\widehat{\beta}_\pi$ . We approximate the integrated value  $\widehat{V}^\pi(\mathbb{G}) = \int_{\mathbf{s} \in \mathcal{S}} \widehat{V}^\pi(\mathbf{s})\mathbb{G}(d\mathbf{s})$  by sampling 10,000 states from the reference distribution  $\mathbb{G}$  and take the average of the estimated value for each state.

**Fitting dropout propensity model** To calculate the IPW estimator, we need to estimate the dropout probability from the data first. For ignorable missingness (MAR), we fit a logistic regression with the correctly specified model to predict the dropout probability. For nonignorable missingness (MNAR), we adopt the semiparametric method proposed by Shao and Wang (2016) and treat  $S_2$  as the instrumental variable. To construct multiple estimating equations, the instrumental variable  $S_2$  is first discretized into 4 bins based on the quartiles. The nonparametric part is approximated using Gaussian kernel with bandwidth  $h_l = c \cdot \sigma_l n_l^{-1/3}$ , where  $\sigma_l$ 's and  $n_l$ 's are the estimated standard deviation and the sample size for samples with  $S_2 = l \in \{1, 2, 3, 4\}$ . We pick  $c = 7.5$  in the bandwidth formula based on an inspection of the objective function curve. In the minimization step of GMM, we use the limited-memory BFGS algorithm (Liu and Nocedal 1989) with several initial values to avoid local minimum.

**Inverse Probability Weighted (IPW) estimator** After getting an estimate of the parameter  $\widehat{\psi}_{nT}$  for the dropout model, we plug in the estimated probability to calculate  $\widehat{\beta}_{\pi, \text{IPW}}$ , which is given in (2.9). To avoid extremely large inverse weights, we bound the missing propensity below 0.01. After obtaining  $\widehat{\beta}_{\pi, \text{IPW}}$ , we calculate the integrated value  $\widehat{V}_{\text{IPW}}^\pi(\mathbb{G})$  in a similar way to the CC estimator.

We can also construct confidence intervals for the proposed IPW estimator. The theoretical form of the asymptotic variance is very complicated. For ease of computation, we suggest using an approximation of  $\widehat{\sigma}_{\pi, \text{IPW}}^2(\mathbb{G})$  given by (2.10). Based on our empirical experiments, this approximation is very close to the result of bootstrapping, so we deploy it in our implementation.

## A.3 Additional Details for Real Data Application

**Data overview** The sepsis dataset is extracted from the MIMIC-III v1.4 database (Johnson et al. 2016). We follow the data processing procedure described in Komorowski et al. (2018) and use a pure-python re-implementation available at [https://github.com/microsoft/mimic\\_sepsis](https://github.com/microsoft/mimic_sepsis). The trajectories start from the onset of sepsis or transfer into ICU, whichever came later, and are collected until 48 hours afterward. We focus on this time window to capture treatment management in the early phase. There are 16,735 trajectories in the dataset. After examining the dataset, we found that some patients were discharged early due to recovery or other reasons. However, we did not observe any significant difference in critical physiological features between these patients and those who stayed for the entire duration of the ICU stay. Therefore, we believe that adjustment for such dropout is not necessary, which is also supported by our numerical results. On the other hand, around 5% of trajectories are incomplete due to mortality within the time window. As seen in Figure 2.1, patients who died during their ICU stay were generally in a worse state, and thus, adjusting for missing data in such cases is expected to yield more reliable results. Although previous works have applied offline RL or OPE to this sepsis dataset (Raghu et al. 2017; Peng et al. 2018; Sonabend et al. 2020), to the best of our knowledge, none of these works considered the aforementioned monotone missingness issue.

**State, action, and reward** Given the physiological features available in the dataset, we construct a 15-dimensional state feature vector to represent important features that clinicians would examine when deciding treatment and dosage for patients. The following features are used in our model:

- Demographics: Age
- Lab values: Arterial pH, Chloride, Hemoglobin, INR-International Normalized Ratio, PT-Prothrombin Time, Arterial Blood Gas, Ionised Calcium, Calcium, Arterial Lactate
- Vital signs: Saturation of Peripheral Oxygen (SpO<sub>2</sub>), Temperature, Heart Rate (HR), Respiratory Rate (RR)
- Other: Sequential Organ Failure Assessment (SOFA) score

The state features are aggregated over a time resolution of 4 hours, and in the absence of data in the current time window, the last recorded value is carried forward. The action space consists of three actions: no intravenous fluids and no vasopressors, intravenous fluids only, and vasopressors. The reward is constructed as  $R_{t+1} = 1 - 2 \cdot \mathbb{1}(S_{t+1}^{\text{SOFA}} \geq 12)$ , which penalizes high SOFA scores over a threshold.

**Target policies** In our experiments, we evaluate three target policies: a fitted behavior policy, and optimal policies learned via Dueling Double Deep Q-Network (Wang et al. 2016) and a discrete version of Batch-Constrained Deep Q-Learning (BCQ) (Fujimoto et al. 2019). The behavior policy is fitted with a random forest with 250 trees. For the other types of Q-learning algorithms, we run for  $2 \times 10^5$  iterations with mini-batch size 256 and learning rate 0.001. The dataset is split into two parts, the first part is used for learning the optimal policy, and the second part is used for policy evaluation.

**More details on implementation** Similar to the synthetic environment, we first scale the state variables onto  $[0, 1]$  and construct 4 cubic B-spline basis for each dimension, so there are  $L = 60$  basis functions in total. We do not use tensor product here due to high-dimensionality concerns. The discount factor  $\gamma$  is set to 0.8. To handle nonignorable missingness, we incorporate a dropout propensity model into the value estimator. To fit the dropout propensity model, we use the Ionised Calcium level as an instrument variable and discretize it into 4 bins based on quartiles. We consider a dropout propensity model in (2.8) with  $\mathcal{U}_t = (S_t^{\text{SpO}_2}, S_t^{\text{HR}}, S_t^{\text{RR}})$ ,  $Z_{t+1} = S_t^{\text{SOFA}}$ , and a bandwidth of  $h_l = 7.5 \cdot \sigma_l n_l^{-1/3}$ . For such a large dataset, the kernel estimator used in the semiparametric method can be a bottleneck in computation. To accelerate the computation, we apply the downsampling technique, where we repeatedly sample random subsets from the whole dataset and aggregate the value estimation results by taking the average.

## A.4 Assumptions

In this section, we provide the assumptions for the theoretical results. The following assumption is introduced by Shi et al. (2021b) to ensure the consistency and asymptotic distribution of value estimation when there is no missing data.

**Assumption 4.** *The following conditions hold.*

- (a) *The transition kernel  $p(\cdot|s, a)$  is absolutely continuous with respect to the Lebesgue measure, then there exists some transition density function  $q$  such that  $p(ds'|s, a) = q(s'|s, a) ds'$ . Let  $\Lambda(p, c)$  denotes the class of  $p$ -smooth functions as follows*

$$\Lambda(p, c) = \left\{ h : \sup_{\|\alpha\|_1 \leq \lfloor p \rfloor} \sup_{s \in \mathcal{S}} |D^\alpha h(s)| \leq c, \sup_{\|\alpha\|_1 = \lfloor p \rfloor} \sup_{\substack{s, s' \in \mathcal{S} \\ s \neq s'}} \frac{|D^\alpha h(s) - D^\alpha h(s')|}{\|s - s'\|_2^{p - \lfloor p \rfloor}} \leq c \right\},$$

where  $D^\alpha$  denotes the differential operator  $D^\alpha h(s) = \frac{\partial^{\|\alpha\|_1} h(s)}{\partial s_1^{\alpha_1} \dots \partial s_d^{\alpha_d}}$ ,  $s_j$  denotes the  $j$ -th element of  $s$ ,  $\lfloor p \rfloor$  denote the largest integer that is smaller than  $p$ . Assume there exist

some  $p, c > 0$  such that  $r(\cdot, a), q(s'|\cdot, a) \in \Lambda(p, c)$  for any  $a \in \mathcal{A}, s' \in \mathcal{S}$ .

(b) Let  $\text{BSpl}(L, r)$  denote a tensor-product B-spline basis of degree  $r$  and dimension  $L$  on  $[0, 1]^d$ , and let  $\text{Wav}(L, r)$  denote a tensor-product Wavelet basis of regularity  $r$  and dimension  $L$  on  $[0, 1]^d$ . The sieve  $\Phi_L$  is either  $\text{BSpl}(L, r)$  or  $\text{Wav}(L, r)$  with  $r > \max(p, 1)$ .

(c) Assume the Markov chain has a unique invariant distribution with some density function  $\mu(\cdot)$  on  $\mathcal{S}$ , the probability density function of  $S_0$  is denoted as  $\nu_0$ . The density functions  $\mu$  and  $\nu_0$  are uniformly bounded away from 0 and  $\infty$  on  $\mathcal{S}$ .

(d) Suppose (i) and (ii) hold when  $T \rightarrow \infty$  and (iii) holds when  $T$  is bounded.

(i)  $\lambda_{\min} \left[ \int_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \{ \boldsymbol{\xi}(s, a) \boldsymbol{\xi}^\top(s, a) - \gamma^2 \mathbf{u}_\pi(s, a) \mathbf{u}_\pi^\top(s, a) \} b(a|s) \mu(s) ds \right] \geq \bar{c}$  for some constant  $\bar{c} > 0$ , where  $\mathbf{u}_\pi(s, a) = \mathbb{E} \{ \mathbf{U}_\pi(S_1) | S_0 = s, A_0 = a \}$  and  $\lambda_{\min}(K)$  denotes the minimum eigenvalue of a matrix  $K$ .

(ii) The Markov chain  $\{S_t\}_{t \geq 0}$  is geometrically ergodic, i.e., there exists some function  $M(\cdot)$  on  $\mathcal{S}$  and some constant  $\rho < 1$  such that  $\int_{s \in \mathcal{S}} M(s) \mu(s) ds < +\infty$  and  $\|p_S^t(\cdot | s) - \mu(\cdot)\|_{TV} \leq M(s) \rho^t$  for any  $t \geq 0$ , where  $\|\cdot\|_{TV}$  denotes the total variation norm,  $p_S^t(\mathcal{B}|s) = P(S_t \in \mathcal{B} | S_0 = s)$  is the  $t$ -step transition kernel.

(iii)  $\lambda_{\min} \left[ \sum_{t=0}^{T-1} \mathbb{E} \{ \boldsymbol{\xi}_t \boldsymbol{\xi}_t^\top - \gamma^2 \mathbf{u}_\pi(S_t, A_t) \mathbf{u}_\pi^\top(S_t, A_t) \} \right] \geq T \bar{c}$  for some constant  $\bar{c} > 0$ .

(e) The number of basis  $L$  satisfies  $L = o\{\sqrt{nT}/\log(nT)\}$ ,  $L^{2p/d} \gg nT \{1 + \|\int_s \Phi_L(s) \mathbb{G}(ds)\|_2^{-2}\}$ .

(f) There exists some constant  $c_0 \geq 1$  such that

$$\delta_\pi(s, a) = \mathbb{E} \left[ \left\{ R_1 + \gamma \sum_{a \in \mathcal{A}} \pi(a|S_1) Q^\pi(S_1, a) - Q^\pi(S_0, A_0) \right\}^2 \middle| S_0 = s, A_0 = a \right] \geq c_0^{-1}$$

for any  $s \in \mathcal{S}, a \in \mathcal{A}$ , and  $P(\max_t |R_t| \leq c_0) = 1$ .

These assumptions together guarantee consistent value estimation under complete data. Assumption 4(a) basically assumes the smoothness of the reward function  $r$  and the transition density function  $q$  with respect to the current state  $s$ , this allows us to establish the smoothness of the Q-function. Assumption 4(b) specifies the types of sieve  $\Phi_L$  to approximate the Q-function. Assumption 4(c) is a mild condition on the marginal distribution over states, when  $\nu_0 = \mu$ ,  $\{S_t\}_{t \geq 0}$  is stationary. Assumption 4(d) is imposed to guarantee the invertibility of  $\widehat{\Sigma}_\pi$ . The geometric ergodicity condition in Assumption 4(d)(ii) ensures that  $\{S_t\}_{t \geq 0}$  is exponentially  $\beta$ -mixing (see Theorem 3.7 of Bradley (2005)). We remark that the geometric

ergodicity condition is less restrictive than the independence assumption imposed in some existing reinforcement learning literature (e.g., Dai et al. (2020)). For Assumption 4(e), the constraint on the number of basis functions is to bound the approximation error for the Q-function. Assumption 4(f) is a mild condition on the randomness of observed the reward  $R_{t+1}$  around  $r(S_t, A_t)$  and the uniform boundedness of the observed reward.

Besides the assumptions on Q-function approximation, we also make the following assumption regarding the dropout propensity. Recall that the dropout propensity model takes the form

$$\lambda(S_t, A_t, R_{t+1}, S_{t+1}; \boldsymbol{\psi}) = \{1 + \exp[g(\mathcal{U}_t) + \boldsymbol{\psi}^\top Z_{t+1}]\}^{-1},$$

where  $\boldsymbol{\psi} \in \mathbb{R}^q$  is an unknown tilting parameter,  $Z_{t+1} \in \mathbb{R}^q$  are features mapped from  $(R_{t+1}, S_{t+1})$ ,  $\mathcal{U}_t$  is the non-instrumental part of  $(S_t, A_t)$ , and  $g(\cdot)$  is a non-parametric function of  $\mathcal{U}_t$ .

The following assumption is a boundedness assumption imposed on the dropout propensities, which ensures that the inverse weights are bounded above.

**Assumption 5.** *There exist some  $c_\lambda > 0$  such that  $1 - \lambda(S_t, A_t, R_{t+1}, S_{t+1}) \geq c_\lambda$ .*

To estimate the parameter of dropout propensity model  $\boldsymbol{\psi}^*$ , the  $\tilde{L}$  estimating equations are constructed as follows

$$\begin{aligned} \mathbb{E}_{nT} \{m_l(\mathcal{U}_t, \mathcal{V}_t, R_{t+1}, S_{t+1}, \eta_{t+1}; g, \boldsymbol{\psi})\} &= 0, \text{ for } l = 1, \dots, \tilde{L}, \\ \text{where } m_l(\mathcal{U}_t, \mathcal{V}_t, R_{t+1}, S_{t+1}, \eta_{t+1}; g, \boldsymbol{\psi}) &= \mathbb{1}(\mathcal{V}_t = l) \left( \frac{\eta_{t+1}}{1 - \lambda(\mathcal{U}_t, R_{t+1}, S_{t+1}; g, \boldsymbol{\psi})} - 1 \right). \end{aligned} \quad (\text{A.1})$$

For succinctness, suppress the data arguments in  $m_l(\mathcal{U}_t, \mathcal{V}_t, R_{t+1}, S_{t+1}, \eta_{t+1}; g, \boldsymbol{\psi})$  and write it as  $m_{l,t}(g, \boldsymbol{\psi})$ . Stack  $m_{l,t}(g, \boldsymbol{\psi})$  together and denote the vector as  $\mathbf{m}_t(g, \boldsymbol{\psi})$ . The estimating equation becomes

$$\mathbb{E}_{nT} \{\mathbf{m}_t(g, \boldsymbol{\psi})\} = \mathbf{0}.$$

To estimate  $\boldsymbol{\psi}$ , the non-parametric part  $g$  is first profiled based on the following relationship

$$\exp\{g_\psi(\mathcal{U}_t)\} = \frac{\mathbb{E}\{1 - \eta_{t+1} \mid \mathcal{U}_t\}}{\mathbb{E}\{\eta_{t+1} \exp(\boldsymbol{\psi}^\top Z_{t+1}) \mid \mathcal{U}_t\}}. \quad (\text{A.2})$$

The corresponding kernel estimator is given by

$$\exp\{\hat{g}_\psi(\mathcal{U}_t)\} = \frac{\sum_{i=1}^n \sum_{t=0}^{T-1} (1 - \eta_{i,t+1}) K_h(\mathcal{U}_t - \mathcal{U}_{i,t})}{\sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \exp(\boldsymbol{\psi}^\top Z_{i,t+1}) K_h(\mathcal{U}_t - \mathcal{U}_{i,t})},$$

where  $K_h(\cdot) = h^{-1}K(\cdot/h)$ ,  $K(\cdot)$  is symmetric kernel function,  $h$  is the bandwidth. By

Assumption 5(a), the true function of  $g(u)$  can be expressed as  $g_0(u) = g_{\psi^*}(u)$ .

To guarantee the consistency and asymptotic normality of the dropout propensity estimator, the following assumption is imposed; see Section 3 of Shao and Wang (2016).

**Assumption 6.** *The following conditions hold.*

- (a) *The kernel  $K(u)$  has bounded derivatives of order  $d'$ , satisfies  $\int K(u)du = 1$ , and has zero moments of order up to  $m' - 1$  and nonzero  $m'$ -th order moment.*
- (b) *The true function of  $g(u)$  is continuously differentiable to order  $d'$  and bounded on an open set containing the support of  $u$ .*
- (c) *The moment  $\mathbb{E}\{\exp(4\boldsymbol{\psi}^\top Z)\}$  is finite and the function  $\mathbb{E}\{\exp(4\boldsymbol{\psi}^\top Z)\}f(u)$  is bounded, where  $f(u)$  is the marginal density of  $u$ .*
- (d) *The bandwidth of the kernel,  $h = h_{nT}$ , is such that  $h_{nT} \rightarrow 0$ ,  $nT \cdot h_{nT}^{p'} \rightarrow \infty$ ,  $(nT)^{1/2}h_{nT}^{p'+2d'}/\log(nT) \rightarrow \infty$  and  $nT \cdot h_{nT}^{2m'} \rightarrow 0$  as  $nT \rightarrow \infty$ , where  $p'$  is the dimension of  $u$ .*
- (e) *Let  $\mathbf{m}_0(g_\psi, \boldsymbol{\psi}) = \mathbb{E}\{\mathbf{m}_t(g_\psi, \boldsymbol{\psi})\}$ .  $\boldsymbol{\psi}^*$  is the unique solution to  $\mathbf{m}_0(g_{\boldsymbol{\psi}^*}, \boldsymbol{\psi}^*) = \mathbf{0}$ , and that  $\sup_{\boldsymbol{\psi}} \|\mathbf{m}_0(g_\psi, \boldsymbol{\psi})\| < \infty$ .*
- (f) *Let*

$$\boldsymbol{\omega} = \begin{pmatrix} \mathbb{E}\{1 - \eta \mid u\} \\ \mathbb{E}\{\eta \exp(\boldsymbol{\psi}^\top Z) \mid u\} \end{pmatrix} := \begin{pmatrix} \omega_1(u) \\ \omega_2(u) \end{pmatrix},$$

$$\boldsymbol{\omega}_0 = \begin{pmatrix} \mathbb{E}\{1 - \eta \mid u\} \\ \mathbb{E}\{\eta \exp(\boldsymbol{\psi}^{*\top} Z) \mid u\} \end{pmatrix},$$

$$\tilde{m}_l(v, u, z, \eta, \boldsymbol{\omega}, \boldsymbol{\psi}) = \mathbb{1}(v = l) \left[ \eta \left\{ 1 + \exp(\boldsymbol{\psi}^\top z) \frac{\omega_1(u)}{\omega_2(u)} \right\} - 1 \right].$$

There exists a vector of the functional  $G(z, u, \eta, \boldsymbol{\omega})$  which is linear in  $\boldsymbol{\omega}$  such that

(i) For small enough  $\|\boldsymbol{\omega} - \boldsymbol{\omega}_0\|$ ,

$$\begin{aligned} & \|\tilde{\mathbf{m}}(v, u, z, \eta, \boldsymbol{\omega}, \boldsymbol{\psi}^*) - \tilde{\mathbf{m}}(v, u, z, \eta, \boldsymbol{\omega}_0, \boldsymbol{\psi}^*) - G(v, u, z, \eta, \boldsymbol{\omega} - \boldsymbol{\omega}_0)\| \\ & \leq b(v, u, z, \eta) \|\boldsymbol{\omega} - \boldsymbol{\omega}_0\|^2, \end{aligned}$$

where  $\mathbb{E}\{b(v, u, z, \eta)\} < \infty$ , and  $\tilde{\mathbf{m}}(\cdot)$  is the  $L$ -dimensional vector with  $l$ -th component as  $\tilde{m}_l(\cdot)$ .

(ii)  $\|G(v, u, z, \eta, \boldsymbol{\omega})\| \leq c(v, u, z, \eta) \|\boldsymbol{\omega}\|$  and  $\mathbb{E}\{c(v, u, z, \eta)^2\} < \infty$ .

(iii) There exists an almost everywhere continuous function  $v(u)$  with  $\int \|v(u)\|du < \infty$ , such that  $\mathbb{E}\{G(v, u, z, \eta, \boldsymbol{\omega})\} = \int v(u)\boldsymbol{\omega}(u)du$  for all  $\|\boldsymbol{\omega}\| \leq \infty$ , and also  $\mathbb{E}\{\sup_{\|\zeta\| \leq \varepsilon} \|v(u + \zeta)\|^4\} < \infty$  for some  $\varepsilon > 0$ .

(g) For small enough  $\|\boldsymbol{\omega} - \boldsymbol{\omega}_0\|$ ,  $\tilde{\mathbf{m}}(v, u, z, \eta, \boldsymbol{\omega}, \boldsymbol{\psi})$  is continuously differentiable in  $\boldsymbol{\psi}$  in a neighbourhood of  $\boldsymbol{\psi}^*$ , and there is  $k(v, u, z, \eta)$  with  $\mathbb{E}\{k(v, u, z, \eta)\} < \infty$  such that

$$\begin{aligned} & \|\nabla_{\boldsymbol{\psi}} \tilde{\mathbf{m}}(v, u, z, \eta, \boldsymbol{\omega}, \boldsymbol{\psi}) - \nabla_{\boldsymbol{\psi}} \tilde{\mathbf{m}}(v, u, z, \eta, \boldsymbol{\omega}_0, \boldsymbol{\psi}^*)\| \\ & \leq k(v, u, z, \eta) (\|\boldsymbol{\psi} - \boldsymbol{\psi}^*\|^\varepsilon + \|\boldsymbol{\omega} - \boldsymbol{\omega}_0\|^\varepsilon) \end{aligned}$$

for an  $\varepsilon > 0$ , and  $\boldsymbol{\Gamma} = \mathbb{E}\{\nabla_{\boldsymbol{\psi}} \tilde{\mathbf{m}}(v, u, z, \eta, \boldsymbol{\omega}_0, \boldsymbol{\psi}^*)\}$  exists and is of full rank.

Assumption 6(a) imposes restrictions on the kernel, the boundedness condition of the  $d'$ -th derivative ensures that  $\|g\|$  is well defined. Here,  $\|g\|$  denotes the Sobolev norm of  $g$  for some nonnegative integer  $d'$ , which is defined by  $\|g\| = \max_{\ell \leq d'} \sup_{u \in \mathcal{U}} \left\| \frac{\partial^\ell g(u)}{\partial u^\ell} \right\|$ . The other two conditions control the bias of  $\hat{g}(u)$ , a larger  $m'$  will lead to a faster convergence rate of  $\mathbb{E}[\hat{g}(u)]$  to  $g_0(u)$ , where  $g_0(u)$  represents the true function for  $g(u)$ . Assumption 6(b) imposes smoothness on  $g_0(u)$  in order for the bias-reducing kernels to have the desired effect. The fourth-moment condition in Assumption 6(c) is useful for obtaining optimal convergence rates for  $g$  (Newey 1994). Assumption 6(d) imposes restrictions on the bandwidth. These four assumptions together guarantee the consistency of the kernel estimator, combining with Assumption 6(e) leads to the consistency of  $\boldsymbol{\psi}$  estimator (see Theorem 1 of Shao and Wang (2016)). Assumption 6(f)(g) are required to prove the asymptotic normality of  $\hat{\boldsymbol{\psi}}$  (see Theorem 2 of Shao and Wang (2016) and Theorem 8.12 of Newey and McFadden (1994)). More discussions on these assumptions for semiparametric two-step estimators can be found in Section 8.3 of Newey and McFadden (1994).

## A.5 Proof of Main Results

In this section, we provide the proofs for Theorem 1, 2 and 3. For simplicity, we will omit the subscript  $\pi$  in  $\boldsymbol{\Sigma}_\pi, \hat{\boldsymbol{\Sigma}}_\pi, \boldsymbol{\beta}_\pi, \hat{\boldsymbol{\beta}}_\pi, \sigma_\pi, \hat{\sigma}_\pi$ . We first introduce the following lemmas from Shi et al. (2021b), the proofs can be found in Section E of their paper.

**Lemma 1.** *Under Assumption 4(a), there exists some constant  $c' > 0$  such that  $Q^\pi(s, a) \in \Lambda(p, c')$  for any policy  $\pi$  and  $a \in \mathcal{A}$ .*

**Lemma 2.** *Under Assumption 4(b), there exists some constant  $c^* \geq 1$  such that*

$$(c^*)^{-1} \leq \lambda_{\min} \left\{ \int_{s \in \mathcal{S}} \Phi_L(s) \Phi_L^\top(s) ds \right\} \leq \lambda_{\max} \left\{ \int_{s \in \mathcal{S}} \Phi_L(s) \Phi_L^\top(s) ds \right\} \leq c^*$$

and  $\sup_{s \in \mathcal{S}} \|\Phi_L(s)\|_2 \leq c^* \sqrt{L}$ .

**Lemma 3.** *Suppose Assumption 4 holds. Define  $\Sigma = \mathbb{E}\widehat{\Sigma}$ , we have  $\|\Sigma^{-1}\|_2 \leq 3\bar{c}^{-1}$ ,  $\|\Sigma\|_2 = O(1)$ ,  $\|\widehat{\Sigma} - \Sigma\|_2 = O_p\{L^{1/2}(nT)^{-1/2} \log(nT)\}$ ,  $\|\widehat{\Sigma}^{-1} - \Sigma^{-1}\|_2 = O_p\{L^{1/2}(nT)^{-1/2} \log(nT)\}$  and  $\|\widehat{\Sigma}^{-1}\|_2 \leq 6\bar{c}^{-1}$  with probability approaching 1, as either  $n \rightarrow \infty$  or  $T \rightarrow \infty$ .*

**Lemma 4.** *Suppose Assumption 4 holds. As either  $n \rightarrow \infty$  or  $T \rightarrow \infty$ , we have*

$$\begin{aligned} \lambda_{\max}(T^{-1} \sum_{t=0}^{T-1} \mathbb{E} \xi_t \xi_t^\top) &= O_p(1), & \lambda_{\max}\{(nT)^{-1} \sum_{i=1}^n \sum_{t=0}^{T-1} \xi_{i,t} \xi_{i,t}^\top\} &= O_p(1), \\ \lambda_{\min}(T^{-1} \sum_{t=0}^{T-1} \mathbb{E} \xi_t \xi_t^\top) &\geq \bar{c}/2, & \lambda_{\min}\{(nT)^{-1} \sum_{i=1}^n \sum_{t=0}^{T-1} \xi_{i,t} \xi_{i,t}^\top\} &\geq \bar{c}/3 \end{aligned}$$

with probability approaching 1.

**Lemma 5.**  $\|\int_s \mathbf{U}(s) \mathbb{G}(ds)\|_2 \geq m^{-1/2} \|\int_s \Phi_L(s) \mathbb{G}(ds)\|_2$ , where  $m$  is the number of actions in the action space.

**Lemma 6.** *Define  $\Sigma^* = \int_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \xi(s, a) \{\xi(s, a) - \gamma \mathbf{u}(s, a)\}^\top b(a | s) \mu(s) ds$ . Suppose  $T \rightarrow \infty$ . Under the given conditions in Lemma 3, we have  $\|\Sigma - \Sigma^*\|_2 \preceq T^{-1/2}$ .*

**Remark 6.** *The notation  $a_n \preceq b_n$  means that there exists some constant  $C > 0$  such that  $a_n \leq C \cdot b_n$  for any  $n$ . The notation  $a_n \preceq 1$  means  $a_n = O_p(1)$ .*

Next, we will go through the proof for the consistency and asymptotic result for the proposed IPW estimator. The big idea is similar to the proof of Theorem 1 in Shi et al. (2021b) but with additional components to handle inverse weights and associated uncertainty.

### A.5.1 Proof of Theorem 1

*Proof.* We first provide a sketch of the big idea. Assume the true Q-function is  $Q^\pi(s, a) = \Phi_L^\top(s) \beta_\pi^*$  and the true parameter  $\beta^*$  satisfies

$$\mathbb{E}\{\mathbf{M}_t(\beta_\pi^*)\} = \mathbf{0}, \text{ where } \mathbf{M}_t(\beta_\pi) = \xi_t \{R_{t+1} - (\xi_t - \gamma \mathbf{U}_{\pi, t+1})^\top \beta_\pi\}.$$

Under incomplete data, the equation becomes  $\mathbb{E}\{\eta_{t+1} \mathbf{M}_t(\beta_\pi)\} = \mathbf{0}$ . Using the condition of MAR (Definition 1), we apply the conditional independence between  $\eta_{t+1}$  and  $(S_{t+1}, R_{t+1})$  to separate the  $\eta_{t+1}$  and  $\mathbf{M}_t$  term as follows

$$\begin{aligned} \mathbb{E}\{\eta_{t+1} \mathbf{M}_t(\beta_\pi)\} &= \mathbb{E}\{\mathbb{E}(\eta_{t+1} \mathbf{M}_t(\beta_\pi) | S_t, A_t, \eta_t)\} \\ &= \mathbb{E}\{\mathbb{E}(\eta_{t+1} | S_t, A_t, \eta_t) \mathbb{E}(\mathbf{M}_t(\beta_\pi) | S_t, A_t)\}. \end{aligned}$$

It follows from  $\mathbb{E}\{R_{t+1} + \gamma \sum_{a' \in \mathcal{A}} Q^\pi(S_{t+1}, a') \pi(a' | S_{t+1}) - Q^\pi(S_t, A_t) | S_t, A_t\} = 0$  that

$$\mathbb{E}\{\mathbf{M}_t(\boldsymbol{\beta}_\pi^*) | S_t, A_t\} = \mathbf{0}.$$

Therefore,  $\mathbb{E}\{\eta_{t+1} \mathbf{M}_t(\boldsymbol{\beta}_\pi^*)\} = \mathbf{0}$ , then  $\boldsymbol{\beta}_\pi^*$  is still the solution to  $\mathbb{E}\{\eta_{t+1} \mathbf{M}_t(\boldsymbol{\beta}_\pi)\} = \mathbf{0}$ . As a result, the corresponding value estimator is still unbiased under some regularity conditions. However, for nonignorable missingness (MNAR),  $\mathbb{E}\{\eta_{t+1} \mathbf{M}_t(\boldsymbol{\beta}_\pi^*)\} = \mathbf{0}$  no longer holds because  $\eta_{t+1}$  and  $\mathbf{M}_t$  cannot be separated using the conditional independence. Thus the complete-case estimator  $\widehat{\boldsymbol{\beta}}_{\pi, \text{CC}}$  will be biased from  $\boldsymbol{\beta}_\pi^*$  unless the probability  $P(\eta_{t+1} = 1 | S_t, A_t, R_{t+1}, S_{t+1}, \eta_t)$  is a constant. Next, we provide a more rigorous proof that takes into account the approximation error.

By Condition 4(a)(b)(e), the number of basis  $L$  for the Q-function satisfies  $L^{2p/d} \gg nT \{1 + \|\int_s \Phi_L(s) \mathbb{G}(ds)\|_2^{-2}\}$ . It follows from Lemma 5 that

$$L^{2p/d} \gg nT \left\{ 1 + \left\| \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\|_2^{-2} \right\}.$$

By Lemma 1 and Condition 4(b), there exist a set of vectors  $\{\beta_a^*\}$  that satisfy

$$\sup_{s \in \mathcal{S}, a \in \mathcal{A}} |Q^\pi(s, a) - \Phi_L^\top(s) \beta_a^*| \leq CL^{-p/d}, \quad (\text{A.3})$$

for some constant  $C > 0$  (Huang et al. 1998). Let  $\boldsymbol{\beta}^* = (\beta_1^{*\top}, \dots, \beta_m^{*\top})^\top$ , define

$$\begin{aligned} r_{i,t} &= \gamma \sum_{a \in \mathcal{A}} \{ \Phi_L^\top(S_{i,t+1}) \beta_a^* - Q^\pi(S_{i,t+1}, a) \} \pi(a | S_{i,t+1}) - \{ \Phi_L^\top(S_{i,t}) \beta_{A_{i,t}}^* - Q^\pi(S_{i,t}, A_{i,t}) \}, \\ \varepsilon_{i,t} &= R_{i,t+1} + \gamma \sum_{a \in \mathcal{A}} Q^\pi(S_{i,t+1}, a) \pi(a | S_{i,t+1}) - Q^\pi(S_{i,t}, A_{i,t}). \end{aligned} \quad (\text{A.4})$$

The condition  $P(\max_t |R_t| \leq c_0) = 1$  in Assumption 4(f) implies that  $R_{i,t} \leq c_0, \forall i, t$ , almost surely. By Lemma 1, we have  $|Q^\pi(s, a)| \leq c'$  for any  $\pi, s, a$ . Therefore, the error term  $\varepsilon_{i,t}$  can be bounded as follows

$$\max_{0 \leq t < T, 1 \leq i \leq n} |\varepsilon_{i,t}| \leq c_0 + (\gamma + 1)c' \leq c_0 + 2c', \text{ almost surely.} \quad (\text{A.5})$$

In addition, it follows from (A.3) that

$$\max_{0 \leq t < T, 1 \leq i \leq n} |r_{i,t}| \leq 2 \sup_{s \in \mathcal{S}, a \in \mathcal{A}} |Q^\pi(s, a) - \Phi_L^\top(s) \beta_a^*| \leq 2CL^{-p/d}. \quad (\text{A.6})$$

For incomplete data, we can only leverage the observed samples for inference. With the response indicator  $\eta_t$  defined in Section 2.3, the estimating equations can be written as  $\mathbb{E}\{\mathbf{M}_t(\boldsymbol{\beta}^*)|\eta_{t+1}=1\}=\mathbf{0}$ , or equivalently,  $\mathbb{E}\{\eta_{t+1}\mathbf{M}_t(\boldsymbol{\beta}^*)\}=\mathbf{0}$ . The estimator for  $\boldsymbol{\beta}^*$  is given by

$$\widehat{\boldsymbol{\beta}}_{\text{CC}} = \underbrace{\left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{i,t+1})^\top \right\}^{-1}}_{\widehat{\boldsymbol{\Sigma}}_{\text{CC}}} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} R_{i,t+1} \right).$$

Let  $\boldsymbol{\Sigma}_{\text{CC}} = \mathbb{E}\widehat{\boldsymbol{\Sigma}}_{\text{CC}}$ . By definition,

$$\begin{aligned} \widehat{\boldsymbol{\beta}}_{\text{CC}} - \boldsymbol{\beta}^* &= \widehat{\boldsymbol{\Sigma}}_{\text{CC}}^{-1} \left[ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} \left\{ R_{i,t+1} - (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{i,t+1})^\top \boldsymbol{\beta}^* \right\} \right] \\ &= \widehat{\boldsymbol{\Sigma}}_{\text{CC}}^{-1} \left[ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} \times \right. \\ &\quad \left. \left\{ R_{i,t+1} - \Phi_L^\top(S_{i,t}) \boldsymbol{\beta}_{A_{i,t}}^* + \gamma \sum_{a \in \mathcal{A}} \Phi_L^\top(S_{i,t+1}) \boldsymbol{\beta}_a^* \pi(a | S_{i,t+1}) \right\} \right] \\ &= \widehat{\boldsymbol{\Sigma}}_{\text{CC}}^{-1} \left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} (\varepsilon_{i,t} + r_{i,t}) \right\} \\ &= \underbrace{\boldsymbol{\Sigma}_{\text{CC}}^{-1} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} \right)}_{\zeta_1} + \underbrace{\left( \widehat{\boldsymbol{\Sigma}}_{\text{CC}}^{-1} - \boldsymbol{\Sigma}_{\text{CC}}^{-1} \right) \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} \right)}_{\zeta_2} \\ &\quad + \underbrace{\widehat{\boldsymbol{\Sigma}}_{\text{CC}}^{-1} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} r_{i,t} \right)}_{\zeta_3}. \end{aligned}$$

It suffices to derive the error bounds for  $\|\zeta_1\|_2$ ,  $\|\zeta_2\|_2$ , and  $\|\zeta_3\|_2$ .

**Error bound for  $\|\zeta_3\|_2$ .** For any  $\mathbf{a} \in \mathbb{R}^{mL}$ ,

$$\begin{aligned}
& \left| \mathbf{a}^\top \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} r_{i,t} \right) \right| \\
& \leq \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} |\mathbf{a}^\top \boldsymbol{\xi}_{i,t}| |r_{i,t} \eta_{i,t+1}| \leq \max_{i,t} |r_{i,t}| \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} |\mathbf{a}^\top \boldsymbol{\xi}_{i,t}| \right) \\
& \leq 2CL^{-p/d} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} |\mathbf{a}^\top \boldsymbol{\xi}_{i,t}| \right) \leq 2CL^{-p/d} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \mathbf{a}^\top \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \mathbf{a} \right)^{1/2}.
\end{aligned} \tag{A.7}$$

The second inequality uses the bound of binary  $\eta_t$  that  $|\eta_t| \leq 1$ , the third inequality follows from (A.6), and the fourth inequality applies the Cauchy-Schwarz inequality. Then we obtain

$$\left\| \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} r_{i,t} \right\|_2 \leq 2CL^{-p/d} \lambda_{\max}^{1/2} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \right).$$

By Lemma 3 and Lemma 4, we have

$$\|\zeta_3\|_2 \leq \left\| \widehat{\boldsymbol{\Sigma}}_{\text{CC}}^{-1} \right\|_2 \left\| \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} r_{i,t} \right\|_2 = O_p(1) O_p(L^{-p/d}) = O_p(L^{-p/d}), \tag{A.8}$$

which indicates that  $\zeta_3$  is driven by the approximation error of the Q-function, and can be controlled by increasing the number of basis functions.

The main difference between ignorable missingness (MAR) and nonignorable missingness (MNAR) lies in  $(\zeta_1 + \zeta_2)$ . In the following steps, we will show that the complete-case value estimator  $\widehat{V}_{\text{CC}}^\pi(\mathbb{G})$  is still consistent under ignorable missingness (MAR) but becomes biased under nonignorable missingness (MNAR).

- **MAR**

To show  $\|\widehat{\boldsymbol{\beta}}_{\text{CC}} - \boldsymbol{\beta}^*\|_2 = o_p(1)$ , we need to derive the error bound for  $\|\zeta_1\|_2$ ,  $\|\zeta_2\|_2$  and show they are  $o_p(1)$ .

**Error bound for  $\|\zeta_2\|_2$ .** We first derive the error bound for  $\|\zeta_2\|_2$ . By Markov Assumption, Conditional Mean Independence Assumption and Bellman equation,  $\mathbb{E}(\varepsilon_t | \mathcal{F}_t) = \mathbb{E}(\varepsilon_t | S_t, A_t) = 0$ , where  $\mathcal{F}_t = \{(S_j, A_j, R_{j+1})\}_{0 \leq j < t} \cup \{S_t, A_t\}$  denotes the past information up to time  $t$ . Together with the conditional independence of  $\eta_{t+1}$  and  $\varepsilon_t$  based on the definition of MAR, we have  $\mathbb{E}\{\eta_{t+1} \boldsymbol{\xi}_t \varepsilon_t\} = \mathbb{E}\{\mathbb{E}(\eta_{t+1} \boldsymbol{\xi}_t \varepsilon_t | \mathcal{F}_t, \eta_t)\} = \mathbb{E}\{\boldsymbol{\xi}_t \mathbb{E}(\eta_{t+1} | \mathcal{F}_t, \eta_t) \mathbb{E}(\varepsilon_t | \mathcal{F}_t)\} = \mathbf{0}$ . Similarly, for any  $0 \leq t_1 < t_2 < T$ , we obtain  $\mathbb{E}\{\eta_{t_1+1} \eta_{t_2+1} \varepsilon_{t_1} \varepsilon_{t_2} \boldsymbol{\xi}_{t_1}^\top \boldsymbol{\xi}_{t_2}\} = 0$ . In addition, by

the independence assumption among trajectories, we have

$$\mathbb{E} \left\{ \eta_{i_1, t_1+1} \eta_{i_2, t_2+1} \varepsilon_{i_1, t_1} \varepsilon_{i_2, t_2} \boldsymbol{\xi}_{i_1, t_1}^\top \boldsymbol{\xi}_{i_2, t_2} \right\} = 0.$$

It follows that

$$\mathbb{E} \left\| \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i, t+1} \boldsymbol{\xi}_{i, t} \varepsilon_{i, t} \right\|_2^2 = \sum_{i=1}^n \sum_{t=0}^{T-1} \mathbb{E} \left\{ \eta_{i, t+1}^2 \varepsilon_{i, t}^2 \boldsymbol{\xi}_{i, t}^\top \boldsymbol{\xi}_{i, t} \right\} = n \sum_{t=0}^{T-1} \mathbb{E} \left\{ \eta_{t+1}^2 \varepsilon_t^2 \boldsymbol{\xi}_t^\top \boldsymbol{\xi}_t \right\}.$$

Together with (A.5) and Lemma 2, we obtain

$$\mathbb{E} \left\| \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i, t+1} \boldsymbol{\xi}_{i, t} \varepsilon_{i, t} \right\|_2^2 \leq (c_0 + 2c')^2 n \sum_{t=0}^{T-1} \mathbb{E} \boldsymbol{\xi}_t^\top \boldsymbol{\xi}_t \leq (c_0 + 2c')^2 nT \sup_{s \in \mathcal{S}} \|\Phi_L(s)\|_2^2 \preceq nTL.$$

By Markov inequality,

$$\frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i, t+1} \boldsymbol{\xi}_{i, t} \varepsilon_{i, t} = O_p \left\{ \sqrt{L/(nT)} \right\}.$$

Combine with Lemma 3 yields

$$\zeta_2 = O_p \left\{ \sqrt{L/nT} \log(nT) \right\} O_p \left\{ \sqrt{L/(nT)} \right\} = O_p \left\{ L(nT)^{-1} \log(nT) \right\}. \quad (\text{A.9})$$

**Error bound for  $\|\zeta_1\|_2$ .** Using similar arguments as bounding  $\|\zeta_2\|_2$ , we obtain

$$\zeta_1 = O_p \left\{ L^{1/2} (nT)^{-1/2} \right\}. \quad (\text{A.10})$$

Combining (A.8), (A.9), and (A.10), we have

$$\widehat{\boldsymbol{\beta}}_{\text{CC}} - \boldsymbol{\beta}^* = O_p \left\{ L^{1/2} (nT)^{-1/2} \right\} + O_p \left\{ L(nT)^{-1} \log(nT) \right\} + O_p \left\{ L^{-p/d} \right\}.$$

It follows from Condition 4(e) that

$$\|\widehat{\boldsymbol{\beta}}_{\text{CC}} - \boldsymbol{\beta}^*\|_2 = O_p \left( L^{-p/d} \right) + O_p \left\{ L^{1/2} (nT)^{-1/2} \right\} = o_p(1).$$

Recall that  $\widehat{V}_{\text{CC}}^\pi(\mathbb{G}) = \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \widehat{\boldsymbol{\beta}}_{\text{CC}}$ , thus,

$$\left| \widehat{V}_{\text{CC}}^\pi(\mathbb{G}) - V^\pi(\mathbb{G}) \right| \leq \left\| \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\|_2 \left\| \widehat{\boldsymbol{\beta}}_{\text{CC}} - \boldsymbol{\beta}^* \right\|_2 = o_p(1), \quad (\text{A.11})$$

that is,  $\widehat{V}_{CC}^\pi(\mathbb{G}) \xrightarrow{p} V^\pi(\mathbb{G})$  as  $nT \rightarrow \infty$ . Therefore, the value estimator  $\widehat{V}_{CC}^\pi(\mathbb{G})$  is still consistent when the dropout mechanism is MAR.

- MNAR

**Error bounds for  $\|\zeta_1 + \zeta_2\|_2$ .** Under nonignorable missingness, the conditional independence of  $\eta_{t+1}$  and  $\varepsilon_t$  no longer holds, so

$$\begin{aligned} \mathbb{E}\{\eta_{t+1}\boldsymbol{\xi}_t\varepsilon_t\} &= \mathbb{E}\{\mathbb{E}\{\eta_{t+1}\boldsymbol{\xi}_t\varepsilon_t \mid \mathcal{F}_t, S_{t+1}, R_{t+1}, \eta_t\}\} = \mathbb{E}\{\boldsymbol{\xi}_t\varepsilon_t\mathbb{E}\{\eta_{t+1} \mid \mathcal{F}_t, S_{t+1}, R_{t+1}, \eta_t\}\} \\ &= \mathbb{E}\{\boldsymbol{\xi}_t\varepsilon_t\eta_t(1 - \lambda(S_t, A_t, S_{t+1}, R_{t+1}))\} \neq \mathbf{0}. \end{aligned}$$

We cannot bound the term  $\frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1}\boldsymbol{\xi}_{i,t}\varepsilon_{i,t}$  as in the MAR case. As a result,  $\|\zeta_1 + \zeta_2\| = o_p(1)$  may no longer hold. Therefore,  $\widehat{V}^\pi(\mathbb{G})$  can be biased when the dropout mechanism is MNAR.  $\square$

## A.5.2 Proof of Theorem 2

*Proof.* The steps in this proof will be very similar to the proof of Theorem 1, but now we incorporate the inverse weights. For succinctness, we use the notation

$$\omega_{i,t+1}(\boldsymbol{\psi}) := \eta_{i,t+1}\{1 - \lambda(S_{i,t}, A_{i,t}, R_{i,t+1}, S_{i,t+1}; \boldsymbol{\psi})\}^{-1}$$

to represent the weighting term, where  $\boldsymbol{\psi} \in \mathbb{R}^k$  is the parameter of the dropout propensity model. Under assumption 5(a) on the correct specification of the dropout propensity, there exists some  $\boldsymbol{\psi}^*$  such that  $\lambda(S_t, A_t, R_{t+1}, S_{t+1}) = \lambda(S_t, A_t, R_{t+1}, S_{t+1}; \boldsymbol{\psi}^*)$ . It follows from the definition of dropout propensity that  $\mathbb{E}(\eta_{t+1} \mid \mathcal{F}_t, R_{t+1}, S_{t+1}, \eta_t = 1) = 1 - \lambda(S_t, A_t, R_{t+1}, S_{t+1}; \boldsymbol{\psi}^*)$ , therefore,  $\mathbb{E}\{\omega_{t+1}(\boldsymbol{\psi}^*) \mid \mathcal{F}_t, R_{t+1}, S_{t+1}, \eta_t = 1\} = 1$ .

Similar to the previous proof,  $\widehat{\boldsymbol{\beta}}_{IPW} - \boldsymbol{\beta}^*$  can be decomposed as

$$\begin{aligned} \widehat{\boldsymbol{\beta}}_{IPW} - \boldsymbol{\beta}^* &= \underbrace{\boldsymbol{\Sigma}^{-1} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} \right)}_{\zeta_1} \\ &\quad + \underbrace{\left( \widehat{\boldsymbol{\Sigma}}_{IPW}^{-1} - \boldsymbol{\Sigma}^{-1} \right) \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} \right)}_{\zeta_2} \\ &\quad + \underbrace{\widehat{\boldsymbol{\Sigma}}_{IPW}^{-1} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) \boldsymbol{\xi}_{i,t} r_{i,t} \right)}_{\zeta_3}. \end{aligned}$$

To prove its consistency, it suffices to show  $\|\zeta_1\|_2$ ,  $\|\zeta_2\|_2$ , and  $\|\zeta_3\|_2$  are  $o_p(1)$ . To bound these terms, we first introduce the following lemma. This lemma is similar to Lemma 3, but it is with respect to  $\widehat{\Sigma}_{IPW}$  instead of  $\widehat{\Sigma}$ . The proof can be found in Appendix A.5.5.

**Lemma 7.** *Suppose Assumption 4-6 holds. We have  $\|\widehat{\Sigma}_{IPW} - \Sigma\|_2 = O_p\{L^{1/2}(nT)^{-1/2} \log(nT)\}$ ,  $\|\widehat{\Sigma}_{IPW}^{-1} - \Sigma^{-1}\|_2 = O_p\{L^{1/2}(nT)^{-1/2} \log(nT)\}$  and  $\|\widehat{\Sigma}_{IPW}^{-1}\|_2 \leq 6\bar{c}^{-1}$  with probability approaching 1, as either  $n \rightarrow \infty$  or  $T \rightarrow \infty$ .*

Next, we will use Lemma 7 to derive the error bounds for  $\|\zeta_1\|_2$ ,  $\|\zeta_2\|_2$ , and  $\|\zeta_3\|_2$ .

**Error bound for  $\|\zeta_3\|_2$ .** It follows from condition 5(b) that  $\max_{1 \leq i \leq n, 0 \leq t < T} |\omega_{i,t}(\boldsymbol{\psi})| \leq c_\lambda^{-1}$ . Using similar arguments in (A.7), we have

$$\left| \mathbf{a}^\top \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) \boldsymbol{\xi}_{i,t} r_{i,t} \right) \right| \leq \frac{2CL^{-p/d}}{c_\lambda} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \mathbf{a}^\top \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \mathbf{a} \right)^{1/2}$$

for any  $\mathbf{a} \in \mathbb{R}^{mL}$ . Thus,

$$\left\| \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) \boldsymbol{\xi}_{i,t} r_{i,t} \right\|_2 \leq \frac{2CL^{-p/d}}{c_\lambda} \lambda_{\max}^{1/2} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \right) = O_p(L^{-p/d}).$$

By Lemma 7, we obtain

$$\zeta_3 = O_p(L^{-p/d}). \quad (\text{A.12})$$

**Error bounds for  $\|\zeta_2\|_2$ .** The RHS of  $\zeta_1$  and  $\zeta_2$  can be decomposed as follows

$$\begin{aligned} \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} &= \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\boldsymbol{\psi}^*) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} \\ &+ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \left( \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) - \omega_{i,t+1}(\boldsymbol{\psi}^*) \right) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t}. \end{aligned} \quad (\text{A.13})$$

We first show that

$$\frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\boldsymbol{\psi}^*) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} = O_p\{\sqrt{L/(nT)}\}.$$

By the property of conditional expectation, we have

$$\mathbb{E} \{ \omega_{t+1}(\boldsymbol{\psi}^*) \boldsymbol{\xi}_t \varepsilon_t \} = \mathbb{E} \{ \mathbb{E}(\omega_{t+1}(\boldsymbol{\psi}^*) \mid \mathcal{F}_t, R_{t+1}, S_{t+1}, \eta_t) \boldsymbol{\xi}_t \varepsilon_t \} = \mathbb{E} \{ \eta_t \boldsymbol{\xi}_t \varepsilon_t \} = \mathbf{0}. \quad (\text{A.14})$$

Using similar arguments in deriving  $\zeta_2$ 's error bound in Proof A.5.1, we show

$$\begin{aligned} & \mathbb{E} \left\| \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\boldsymbol{\psi}^*) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} \right\|_2^2 = n \sum_{t=0}^{T-1} \mathbb{E} \{ \omega_{t+1}^2(\boldsymbol{\psi}^*) \varepsilon_t^2 \boldsymbol{\xi}_t^\top \boldsymbol{\xi}_t \} \\ & \leq \frac{(c_0 + 2c')^2}{c_\lambda^2} n \sum_{t=0}^{T-1} \mathbb{E} \boldsymbol{\xi}_t^\top \boldsymbol{\xi}_t \leq \frac{(c_0 + 2c')^2}{c_\lambda^2} nT \sup_{s \in \mathcal{S}} \|\Phi_L(s)\|_2^2 \preceq nTL. \end{aligned}$$

Then by the Markov inequality,

$$\frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\boldsymbol{\psi}^*) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} = O_p\{\sqrt{L/(nT)}\}. \quad (\text{A.15})$$

Next, we show that

$$\frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \left( \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) - \omega_{i,t+1}(\boldsymbol{\psi}^*) \right) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} = O_p\{(nT)^{-1/2}\}.$$

A mean value expansion of the LHS around  $\boldsymbol{\psi}^*$  and multiply by  $\sqrt{nT}$  yields

$$\frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} = \frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\boldsymbol{\psi}^*) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} + \sqrt{nT} \mathbf{H}_1 (\widehat{\boldsymbol{\psi}} - \boldsymbol{\psi}^*) + o_p(1), \quad (\text{A.16})$$

where  $\mathbf{H}_1 = \mathbb{E} \{ \boldsymbol{\xi}_t \varepsilon_t \nabla_{\boldsymbol{\psi}} \omega_{t+1}(\boldsymbol{\psi}^*)^\top \}$ . Similarly, we obtain

$$\frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) = \frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\boldsymbol{\psi}^*) + \sqrt{nT} \mathbb{E} \{ \nabla_{\boldsymbol{\psi}} \omega_{t+1}(\boldsymbol{\psi}^*)^\top \} (\widehat{\boldsymbol{\psi}} - \boldsymbol{\psi}^*) + o_p(1),$$

Here, we apply the semiparametric IPW approach (Shao and Wang 2016) to estimate the dropout probability. The nonparametric part  $g(\cdot)$  is first profiled as a function of  $\boldsymbol{\psi}$  using (A.2), and then the parameter of interest  $\boldsymbol{\psi}$  is estimated via GMM (Hansen 1982). According to the proof of Theorem 2 in Shao and Wang (2016),  $\sqrt{nT}(\widehat{\boldsymbol{\psi}} - \boldsymbol{\psi}^*)$  can be written as

$$\sqrt{nT}(\widehat{\boldsymbol{\psi}} - \boldsymbol{\psi}^*) = \frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} \boldsymbol{\phi}_{i,t+1} + o_p(1).$$

Here  $\boldsymbol{\phi}_{i,t+1} = (\boldsymbol{\Gamma}^\top \mathbf{W} \boldsymbol{\Gamma})^{-1} \boldsymbol{\Gamma}^\top \mathbf{W} \mathbf{m}_{i,t}(\widehat{g}_{\boldsymbol{\psi}^*}, \boldsymbol{\psi}^*)$ , where  $\mathbf{m}_t(\widehat{g}_{\boldsymbol{\psi}}, \boldsymbol{\psi})$  is a concatenation of  $m_{l,t}(\widehat{g}_{\boldsymbol{\psi}}, \boldsymbol{\psi})$  defined in (A.1),  $\boldsymbol{\Gamma}$  is defined in Assumption 6(g),  $\mathbf{W}$  is the inverse of positive definite  $\widetilde{L} \times \widetilde{L}$  matrix with element at  $(l, l')$  equals to  $\mathbb{E} \{ m_{l,t}(g_{\boldsymbol{\psi}^*}, \boldsymbol{\psi}^*) m_{l',t}(g_{\boldsymbol{\psi}^*}, \boldsymbol{\psi}^*) \}$ . Under Assumption

6(a)-(f), we have

$$\sqrt{nT} \left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} (\mathbf{m}_{i,t}(\widehat{g}_{\psi^*}, \boldsymbol{\psi}^*) - \mathbf{m}_{i,t}(g_{\psi^*}, \boldsymbol{\psi}^*) - \boldsymbol{\tau}_{i,t}(\boldsymbol{\psi}^*)) \right\} \xrightarrow{p} 0,$$

where  $\boldsymbol{\tau}_{i,t}(\boldsymbol{\psi}^*) = v(\mathcal{U}_{i,t}) \begin{pmatrix} 1 - \eta_{i,t+1} \\ \eta_{i,t+1} \exp(\boldsymbol{\psi}^{*\top} Z_{i,t+1}) \end{pmatrix} - \mathbb{E} \left[ v(\mathcal{U}_{i,t}) \begin{pmatrix} 1 - \eta_{i,t+1} \\ \eta_{i,t+1} \exp(\boldsymbol{\psi}^{*\top} Z_{i,t+1}) \end{pmatrix} \right],$  (A.17)

where the function  $v(\cdot)$  is defined in Assumption 6(f)(iii). By Theorem 8.11 of Newey and McFadden (1994), we have

$$\sqrt{nT} \left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \mathbf{m}_{i,t}(\widehat{g}_{\psi^*}, \boldsymbol{\psi}^*) \right\} \xrightarrow{d} \mathcal{N}(\mathbf{0}, \boldsymbol{\Omega}_{\psi^*}),$$

where  $\boldsymbol{\Omega}_{\psi^*} = \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \{ \mathbf{m}_t(g_{\psi^*}, \boldsymbol{\psi}^*) + \boldsymbol{\tau}_t(\boldsymbol{\psi}^*) \} \{ \mathbf{m}_t(g_{\psi^*}, \boldsymbol{\psi}^*) + \boldsymbol{\tau}_t(\boldsymbol{\psi}^*) \}^\top.$

Together with Assumption 6(g), we have

$$\sqrt{nT}(\widehat{\boldsymbol{\psi}} - \boldsymbol{\psi}^*) \xrightarrow{d} \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\psi^*}), \text{ where } \boldsymbol{\Sigma}_{\psi^*} = (\boldsymbol{\Gamma}^\top \mathbf{W} \boldsymbol{\Gamma})^{-1} \boldsymbol{\Gamma}^\top \mathbf{W} \boldsymbol{\Omega}_{\psi^*} \mathbf{W} \boldsymbol{\Gamma} (\boldsymbol{\Gamma}^\top \mathbf{W} \boldsymbol{\Gamma})^{-1}. \quad (\text{A.18})$$

Plug (A.18) into (A.16) yields

$$\frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} (\omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) - \omega_{i,t+1}(\boldsymbol{\psi}^*)) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} = \sqrt{nT} \mathbf{H}_1(\widehat{\boldsymbol{\psi}} - \boldsymbol{\psi}^*) + o_p(1) = O_p\{(nT)^{-1/2}\} \quad (\text{A.19})$$

Combining (A.15) and (A.19) yields

$$\frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}(\widehat{\boldsymbol{\psi}}) \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} = O_p\{\sqrt{L/(nT)}\}.$$

Together with Lemma 7, we obtain

$$\zeta_2 = O_p\{\sqrt{L/nT} \log(nT)\} O_p\{\sqrt{L/(nT)}\} = O_p\{L(nT)^{-1} \log(nT)\}. \quad (\text{A.20})$$

**Error bounds for  $\|\zeta_1\|_2$ .** Similarly, we obtain the error bound for  $\zeta_1$  as follows

$$\zeta_1 = O_p\{L^{1/2}(nT)^{-1/2}\}. \quad (\text{A.21})$$

Combining (A.12), (A.20) and (A.21), we have

$$\widehat{\boldsymbol{\beta}}_{\text{IPW}} - \boldsymbol{\beta}^* = O_p \{L^{1/2}(nT)^{-1/2}\} + O_p \{L^{-p/d}\} + O_p \{L(nT)^{-1} \log(nT)\} = o_p(1). \quad (\text{A.22})$$

Following similar arguments as (A.11), we show  $|\widehat{V}_{\text{IPW}}^\pi(\mathbb{G}) - V^\pi(\mathbb{G})| = o_p(1)$ , therefore  $\widehat{V}_{\text{IPW}}^\pi(\mathbb{G})$  is a consistent value estimator.  $\square$

### A.5.3 Proof of Theorem 3

*Proof.* We first provide an outline for the proof, which consists of four steps. In the first step, we give the form of  $\widehat{\sigma}_{\text{IPW}}^2(s)$  and  $\sigma_{\text{IPW}}^2(s)$ . In the second step, we show the linear representation  $\sqrt{nT} \left\{ \widehat{V}_{\text{IPW}}^\pi(\mathbb{G}) - V^\pi(\mathbb{G}) \right\} / \sigma_{\text{IPW}}(\mathbb{G}) = \sqrt{nT} \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \zeta_1 / \sigma_{\text{IPW}}(\mathbb{G}) + o_p(1)$  holds. In the third step, we show  $\sqrt{nT} \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \zeta_1 / \sigma_{\text{IPW}}(\mathbb{G}) \xrightarrow{d} \mathcal{N}(0, 1)$  based on the martingale central limit theorem. In the last step, we show  $\widehat{\sigma}_{\text{IPW}}(\mathbb{G}) / \sigma_{\text{IPW}}(\mathbb{G}) \xrightarrow{p} 1$ . A detailed proof is presented as follows. For succinctness, we will write  $\omega_{t+1}(\widehat{\boldsymbol{\psi}})$  and  $\omega_{t+1}(\boldsymbol{\psi}^*)$  as  $\widehat{\omega}_{t+1}$  and  $\omega_{t+1}^*$  respectively for the rest of the derivation, and use notation  $\omega_{t+1, \boldsymbol{\psi}}$  to replace  $\omega_{t+1}(\boldsymbol{\psi})$ . Also, we will use  $\mathbf{m}_t^*$ ,  $\widehat{\mathbf{m}}_t$  to represent  $\mathbf{m}_t(g_{\boldsymbol{\psi}^*}, \boldsymbol{\psi}^*)$ ,  $\mathbf{m}_t(g_{\widehat{\boldsymbol{\psi}}}, \widehat{\boldsymbol{\psi}})$ , and use  $\boldsymbol{\tau}_t^*$ ,  $\widehat{\boldsymbol{\tau}}_t$  to represent  $\boldsymbol{\tau}_t(\boldsymbol{\psi}^*)$ ,  $\boldsymbol{\tau}_t(\widehat{\boldsymbol{\psi}})$ .

**Step 1.** Derive  $\widehat{\sigma}_{\text{IPW}}^2(s)$  and  $\sigma_{\text{IPW}}^2(s)$ .

In the previous proof, we have already shown that  $\widehat{\boldsymbol{\beta}}_{\text{IPW}} - \boldsymbol{\beta}^* = \zeta_1 + \zeta_2 + \zeta_3$ , where

$$\begin{aligned} \zeta_1 &= \boldsymbol{\Sigma}^{-1} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \widehat{\omega}_{i,t+1} \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} \right) = O_p \{L^{1/2}(nT)^{-1/2}\} \\ \zeta_2 &= \left( \widehat{\boldsymbol{\Sigma}}_{\text{IPW}}^{-1} - \boldsymbol{\Sigma}^{-1} \right) \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \widehat{\omega}_{i,t+1} \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} \right) = O_p \{L(nT)^{-1} \log(nT)\} \\ \zeta_3 &= \widehat{\boldsymbol{\Sigma}}_{\text{IPW}}^{-1} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \widehat{\omega}_{i,t+1} \boldsymbol{\xi}_{i,t} r_{i,t} \right) = O_p \{L^{-p/d}\} \end{aligned}$$

As long as the number of basis  $L$  satisfies Assumption 4(e), we have

$$\sqrt{nT}(\widehat{\boldsymbol{\beta}}_{\text{IPW}} - \boldsymbol{\beta}^*) = \sqrt{nT} \zeta_1 + o_p(1) = \boldsymbol{\Sigma}^{-1} \left( \frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} \widehat{\omega}_{i,t+1} \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} \right) + o_p(1).$$

Plug (A.18) into the mean expansion in (A.16) and define  $\mathbf{H}_2 = \mathbf{H}_1(\boldsymbol{\Gamma}^\top \mathbf{W} \boldsymbol{\Gamma})^{-1} \boldsymbol{\Gamma}^\top \mathbf{W}$ , we

can express  $\sqrt{nT}\zeta_1$  as follows

$$\begin{aligned}
\sqrt{nT}\zeta_1 &= \Sigma^{-1} \left\{ \frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} (\omega_{i,t+1}^* \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} + \mathbf{H}_1 \phi_{i,t+1}) \right\} + o_p(1) \\
&= \Sigma^{-1} \left\{ \frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} (\omega_{i,t+1}^* \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} + \mathbf{H}_2 \mathbf{m}_{i,t}(\hat{g}_{\psi^*}, \psi^*)) \right\} + o_p(1) \\
&= \Sigma^{-1} \left\{ \frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} (\omega_{i,t+1}^* \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} + \mathbf{H}_2 (\mathbf{m}_{i,t}^* + \boldsymbol{\tau}_{i,t}^*)) \right\} + o_p(1).
\end{aligned}$$

The last line applies the result of (A.17).

Let  $\zeta_{i,t} := \omega_{i,t+1}^* \boldsymbol{\xi}_{i,t} \varepsilon_{i,t} + \mathbf{H}_2 (\mathbf{m}_{i,t}^* + \boldsymbol{\tau}_{i,t}^*)$ , the expression for  $\sqrt{nT}\zeta_1$  can be simplified as

$$\sqrt{nT}\zeta_1 = \Sigma^{-1} \left\{ \frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} \zeta_{i,t} \right\} + o_p(1). \tag{A.23}$$

The asymptotic variance of  $\zeta_1$  is given by

$$\begin{aligned}
\boldsymbol{\Omega}_{\text{IPW}} &= \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} (\zeta_{i,t} \zeta_{i,t}^\top) \\
&= \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \{ \omega_{i,t+1}^{*2} \varepsilon_{i,t}^2 \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \} + \\
&\quad + \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left\{ \omega_{i,t+1}^* \varepsilon_{i,t} \boldsymbol{\xi}_{i,t} (\mathbf{m}_{i,t}^* + \boldsymbol{\tau}_{i,t}^*)^\top \right\} \mathbf{H}_2^\top \\
&\quad + \frac{1}{T} \sum_{t=0}^{T-1} \mathbf{H}_2 \mathbb{E} \left\{ \omega_{i,t+1}^* \varepsilon_{i,t} (\mathbf{m}_{i,t}^* + \boldsymbol{\tau}_{i,t}^*) \boldsymbol{\xi}_{i,t}^\top \right\} \\
&\quad + \frac{1}{T} \sum_{t=0}^{T-1} \mathbf{H}_2 \mathbb{E} \left\{ (\mathbf{m}_{i,t}^* + \boldsymbol{\tau}_{i,t}^*) (\mathbf{m}_{i,t}^* + \boldsymbol{\tau}_{i,t}^*)^\top \right\} \mathbf{H}_2^\top.
\end{aligned}$$

**Remark 7.** Its estimator  $\widehat{\boldsymbol{\Omega}}_{\text{IPW}}$  can be calculated using the empirical form. In our implementation, since  $\widehat{\boldsymbol{\Omega}}_{\text{IPW}}$  has a complicated form, we ignore the uncertainty associated with dropout

propensity estimation and only keep the first term. The approximation  $\tilde{\Omega}_{IPW}$  is given by

$$\begin{aligned}\tilde{\Omega}_{IPW} &= \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \hat{\omega}_{i,t+1}^2 \hat{\varepsilon}_{i,t}^2 \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \\ &= \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \left\{ \frac{\eta_{i,t+1}}{1 - \lambda(S_{i,t}, A_{i,t}, R_{i,t+1}, S_{i,t+1}; \hat{\boldsymbol{\psi}}_{nT})} \times \right. \\ &\quad \left. (R_{i,t+1} + \gamma \sum_{a \in \mathcal{A}} \Phi_L^\top(S_{i,t+1}) \hat{\boldsymbol{\beta}}_a \pi(a|S_{i,t+1}) - \Phi_L^\top(S_{i,t}) \hat{\boldsymbol{\beta}}_{A_{i,t}}) \right\}^2.\end{aligned}$$

Despite of the proposed approximation for computational purpose, we still use the theoretical version for the rest of our derivation.

The asymptotic variance of  $\hat{V}_{IPW}^\pi(s)$  and its estimator are given by

$$\begin{aligned}\sigma_{IPW}^2(s) &= \mathbf{U}_\pi^\top(s) \boldsymbol{\Sigma}^{-1} \boldsymbol{\Omega}_{IPW} (\boldsymbol{\Sigma}^\top)^{-1} \mathbf{U}_\pi(s), \\ \hat{\sigma}_{IPW}^2(s) &= \mathbf{U}_\pi^\top(s) \hat{\boldsymbol{\Sigma}}_{IPW}^{-1} \hat{\boldsymbol{\Omega}}_{IPW} (\hat{\boldsymbol{\Sigma}}_{IPW}^\top)^{-1} \mathbf{U}_\pi(s).\end{aligned}$$

It follows that

$$\begin{aligned}\sigma_{IPW}^2(\mathbb{G}) &= \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\Omega}_{IPW} (\boldsymbol{\Sigma}^\top)^{-1} \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}, \\ \hat{\sigma}_{IPW}^2(\mathbb{G}) &= \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \hat{\boldsymbol{\Sigma}}_{IPW}^{-1} \hat{\boldsymbol{\Omega}}_{IPW} (\hat{\boldsymbol{\Sigma}}_{IPW}^\top)^{-1} \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}\end{aligned}\tag{A.24}$$

**Step 2.** Show the following linear representation holds

$$\frac{\sqrt{nT} \left\{ \hat{V}_{IPW}^\pi(\mathbb{G}) - V^\pi(\mathbb{G}) \right\}}{\sigma_{IPW}(\mathbb{G})} = \frac{\sqrt{nT} \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \zeta_1}{\sigma_{IPW}(\mathbb{G})} + o_p(1).\tag{A.25}$$

Using arguments similar to step 2 of Theorem 1's proof in Shi et al. (2021b), we have

$$\left| \hat{V}_{IPW}^\pi(\mathbb{G}) - V^\pi(\mathbb{G}) - \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \zeta_1 \right| \leq \left\| \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\|_2 \left\| \hat{\boldsymbol{\beta}}_{IPW} - \boldsymbol{\beta}^* - \zeta_1 \right\|_2 + CL^{-p/d}.\tag{A.26}$$

Here we introduce the following lemma.

**Lemma 8.** *Suppose Assumption 4-6 holds. Then there exist  $C_{\Omega,1}$  such that  $\lambda_{\min}(\boldsymbol{\Omega}_{IPW}) \geq C_{\Omega,1}$  with probability approaching 1. Besides,  $\lambda_{\max}(\boldsymbol{\Omega}_{IPW}) = O_p(1)$ .*

Lemma 8 can be shown by noting that

$$\begin{aligned}
& \lambda_{\min} \left( \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left\{ \omega_{i,t+1}^* \varepsilon_{i,t}^2 \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \right\} \right) = \lambda_{\min} \left( \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left\{ \frac{1}{1 - \lambda_{i,t}^*} \varepsilon_{i,t}^2 \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \right\} \right) \\
& \geq \lambda_{\min} \left( \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left\{ \varepsilon_{i,t}^2 \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \right\} \right) \geq c_0^{-1} \lambda_{\min} \left( \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left\{ \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \right\} \right) \geq \frac{\bar{c}}{3c_0} := C_{\Omega,1}, \\
& \lambda_{\max}(\mathbf{A}) \leq c_\lambda^{-1} (c_0 + 2c')^2 \lambda_{\max} \left( \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left\{ \boldsymbol{\xi}_{i,t} \boldsymbol{\xi}_{i,t}^\top \right\} \right) = O_p(1).
\end{aligned}$$

By Lemma 8, the lower bound of  $\sigma_{\text{IPW}}^2(\mathbb{G})$  satisfies

$$\sigma_{\text{IPW}}^2(\mathbb{G}) \geq C_{\Omega,1} \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{\Sigma}^\top)^{-1} \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}. \quad (\text{A.27})$$

According to Lemma 3, we have  $\lambda_{\max}(\boldsymbol{\Sigma}^\top \boldsymbol{\Sigma}) = O(1)$ . This implies that  $\lambda_{\min}\{\boldsymbol{\Sigma}^{-1}(\boldsymbol{\Sigma}^\top)^{-1}\} \geq \bar{C}$  for some constant  $\bar{C} > 0$ , hence

$$\sigma_{\text{IPW}}^2(\mathbb{G}) \geq C_{\Omega,1} \bar{C} \left\| \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\|_2^2 \quad (\text{A.28})$$

Combining Equation (A.28) together with Equation (A.26) yields that

$$\begin{aligned}
& \frac{1}{\sigma_{\text{IPW}}(\mathbb{G})} \left| \widehat{V}_{\text{IPW}}^\pi(\mathbb{G}) - V^\pi(\mathbb{G}) - \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \zeta_1 \right| \\
& \leq \frac{1}{\sqrt{C_{\Omega,1} \bar{C}}} \left\| \widehat{\boldsymbol{\beta}}_{\text{IPW}} - \boldsymbol{\beta}^* - \zeta_1 \right\|_2 + \frac{CL^{-p/d}}{\sqrt{C_{\Omega,1} \bar{C}} \left\| \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\|_2}.
\end{aligned}$$

According to the previous proof, we have

$$\widehat{\boldsymbol{\beta}}_{\text{IPW}} - \boldsymbol{\beta}^* = \zeta_1 + O_p \left\{ L(nT)^{-1} \log(nT) \right\} + O_p(L^{-p/d})$$

Together with the conditions that  $L \ll \sqrt{nT}/\log(nT)$ ,  $L^{2p/d} \gg nT \left\{ 1 + \left\| \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\|_2^{-2} \right\}$ , we obtain

$$\frac{\sqrt{nT} \{ \widehat{V}_{\text{IPW}}^\pi(\mathbb{G}) - V^\pi(\mathbb{G}) \}}{\sigma_{\text{IPW}}(\mathbb{G})} = \frac{\sqrt{nT} \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \zeta_1}{\sigma_{\text{IPW}}(\mathbb{G})} + o_p(1). \quad (\text{A.29})$$

This completes the second step of the proof.

**Step 3.** Show

$$\frac{\sqrt{nT} \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \zeta_1}{\sigma_{\text{IPW}}(\mathbb{G})} \xrightarrow{d} \mathcal{N}(0, 1).$$

In this step, we first construct a martingale and then apply the martingale central limit theorem. For any integer  $1 \leq g \leq nT$ , let  $i(g)$  and  $t(g)$  be the quotient and the remainder of  $g + T - 1$  divided by  $T$ , that is,  $g = \{i(g) - 1\} \cdot T + t(g) + 1$ ,  $1 \leq i(g) \leq n$ ,  $0 \leq t(g) < T$ . Let  $\mathcal{F}^{(0)} = \{S_{1,0}, A_{1,0}\}$ , then iteratively define  $\{\mathcal{F}^{(g)}\}_{1 \leq g \leq nT}$  as follows:

$$\begin{aligned} \mathcal{F}^{(g)} &= \mathcal{F}^{(g-1)} \cup \left\{ R_{i(g), t(g)+1}, \eta_{i(g), t(g)+1}, S_{i(g), t(g)+1}, A_{i(g), t(g)+1} \right\}, & \text{if } t(g) < T - 1 \\ \mathcal{F}^{(g)} &= \mathcal{F}^{(g-1)} \cup \left\{ R_{i(g), T}, \eta_{i(g), T}, S_{i(g), T}, S_{i(g)+1, 0}, A_{i(g)+1, 0} \right\}, & \text{otherwise.} \end{aligned}$$

Use  $\boldsymbol{\xi}^{(g)}, \mathbf{m}^{(g)}, \boldsymbol{\tau}^{(g)}, \varepsilon^{(g)}, \omega_\psi^{(g)}$  to represent  $\boldsymbol{\xi}_{i(g), t(g)}, \mathbf{m}_{i(g), t(g)}, \boldsymbol{\tau}_{i(g), t(g)}, \varepsilon_{i(g), t(g)}$ , and  $\omega_{i(g), t(g)+1, \psi}$ , respectively. It follows from (A.23) that

$$\sqrt{nT} \frac{\left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \zeta_1}{\sigma_{\text{IPW}}(\mathbb{G})} = \sum_{g=1}^{nT} \frac{\left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} \zeta^{(g)}}{\sqrt{nT} \sigma_{\text{IPW}}(\mathbb{G})} + o_p(1), \quad (\text{A.30})$$

where  $\zeta^{(g)} = \omega^{*(g)} \boldsymbol{\xi}^{(g)} \varepsilon^{(g)} + \mathbf{H}_2(\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})$ . Using similar arguments as (A.14), we can show  $\mathbb{E}\{\omega^{*(g)} \boldsymbol{\xi}^{(g)} \varepsilon^{(g)} \mid \mathcal{F}^{(g-1)}\} = \mathbf{0}$ . Meanwhile,  $\mathbb{E}\{\mathbf{m}^{*(g)} \mid \mathcal{F}^{(g-1)}\} = \mathbf{0}$  holds as a result of  $\mathbb{E}\{\omega^{*(g)} \mid \mathcal{F}^{(g-1)}\} = 1$ , and  $\mathbb{E}\{\boldsymbol{\tau}^{*(g)} \mid \mathcal{F}^{(g-1)}\} = \mathbf{0}$  holds by construction. Therefore,  $\mathbb{E}\{\zeta^{(g)} \mid \mathcal{F}^{(g-1)}\} = \mathbf{0}$ , the first term of the RHS of (A.30) forms a martingale with respect to the filtration  $\{\sigma(\mathcal{F}^{(g)})\}_{g \geq 0}$ , where  $\sigma(\mathcal{F}^{(g)})$  stands for the  $\sigma$ -algebra generated by  $\mathcal{F}^{(g)}$ .

We can then use a martingale central limit theorem for triangular arrays (Corollary 2.8 of McLeish (1974)) to show the asymptotic normality. This requires to verify the following two conditions:

- (a)  $\max_{1 \leq g \leq nT} \left| \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} \zeta^{(g)} \right| / \left\{ \sqrt{nT} \sigma_{\text{IPW}}(s) \right\} \xrightarrow{p} 0$ .
- (b)  $(nT)^{-1} \sum_{g=1}^{nT} \left| \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} \zeta^{(g)} \right|^2 / \left\{ \sigma_{\text{IPW}}^2(s) \right\} \xrightarrow{p} 1$ .

First, we verify condition (a). It follows from Cauchy-Schwarz inequality that

$$\left| \frac{\left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} \zeta^{(g)}}{\sqrt{nT} \sigma_{\text{IPW}}(s)} \right| \leq \frac{\left\| \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} \right\|_2 \|\zeta^{(g)}\|_2}{\sqrt{nT} \sigma_{\text{IPW}}(s)}.$$

Notice that

$$\begin{aligned}
\|\zeta^{(g)}\|_2 &= \|\omega^{*(g)}\boldsymbol{\xi}^{(g)}\varepsilon^{(g)} + \mathbf{H}_2(\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})\|_2 \leq \|\omega^{*(g)}\boldsymbol{\xi}^{(g)}\varepsilon^{(g)}\|_2 + \|\mathbf{H}_2(\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})\|_2 \\
&\leq |\omega^{*(g)}| \|\boldsymbol{\xi}^{(g)}\|_2 |\varepsilon^{(g)}| + \|\mathbf{H}_2\|_2 \|\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}\|_2 \\
&\leq \frac{(c_0 + 2c')}{c_\lambda} \sup_s \|\Phi_L(s)\|_2 + (\|\mathbf{m}^{*(g)}\|_2 + \|\boldsymbol{\tau}^{*(g)}\|_2) \|\mathbf{H}_2\|_2 \\
&\leq \frac{(c_0 + 2c')c^*}{c_\lambda} \sqrt{L} + \left(\frac{1}{c_\lambda} - 1 + \|\boldsymbol{\tau}^{*(g)}\|_2\right) \|\mathbf{H}_2\|_2 \leq C_\zeta \sqrt{L}, \text{ for some constant } C_\zeta.
\end{aligned}$$

Together with Equation (A.28), we have

$$\left| \frac{\left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} \zeta^{(g)}}{\sqrt{nT} \sigma_{\text{IPW}}(s)} \right| \leq \frac{C_\zeta}{\sqrt{C_{\Omega,1} \bar{C}}} \frac{\sqrt{L}}{\sqrt{nT}}.$$

Since  $L \ll \sqrt{nT}/\log(nT)$ , condition (a) is proven.

Next, we verify condition (b). Notice that

$$\begin{aligned}
&\left| \frac{1}{nT} \sum_{g=1}^{nT} \frac{\left| \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} \zeta^{(g)} \right|^2}{\sigma_{\text{IPW}}^2(s)} - 1 \right| \\
&= \frac{1}{\sigma_{\text{IPW}}^2(s)} \times \left| \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} \left( \frac{1}{nT} \sum_{g=1}^{nT} \zeta^{(g)} \zeta^{(g)\top} - \boldsymbol{\Omega}_{\text{IPW}} \right) (\boldsymbol{\Sigma}^\top)^{-1} \left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\} \right|,
\end{aligned}$$

where

$$\boldsymbol{\Omega}_{\text{IPW}} = \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \{ \zeta^{(g)} \zeta^{(g)\top} \}$$

In view of Equation (A.27), it suffices to show

$$\left\| \frac{1}{nT} \sum_{g=1}^{nT} \zeta^{(g)} \zeta^{(g)\top} - \boldsymbol{\Omega}_{\text{IPW}} \right\|_2 = o_p(1). \tag{A.31}$$

This can be proven using similar arguments in bounding  $\|\widehat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}\|_2$  in the proof of Lemma 3. Therefore,

$$\sqrt{nT} \frac{\left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \zeta_1}{\sigma_{\text{IPW}}(\mathbb{G})} = \sum_{g=1}^{nT} \frac{\left\{ \int_s \mathbf{U}(s) \mathbb{G}(ds) \right\}^\top \boldsymbol{\Sigma}^{-1} \zeta^{(g)}}{\sqrt{nT} \sigma_{\text{IPW}}(\mathbb{G})} \xrightarrow{d} \mathcal{N}(0, 1).$$

It follows from (A.29) and Slutsky's theorem that,

$$\frac{\sqrt{nT}\{\widehat{V}^\pi(\mathbb{G}) - V^\pi(\mathbb{G})\}}{\sigma_{\text{IPW}}(\mathbb{G})} \xrightarrow{d} \mathcal{N}(0, 1).$$

**Step 4.** Show  $\widehat{\sigma}_\pi(\mathbb{G})/\sigma_{\text{IPW}}(\mathbb{G}) \xrightarrow{p} 1$ .

Using similar arguments in verifying condition (b), it suffices to show

$$\left\| \widehat{\Sigma}_{\text{IPW}}^{-1} \widehat{\Omega}_{\text{IPW}} (\widehat{\Sigma}_{\text{IPW}}^\top)^{-1} - \Sigma^{-1} \Omega_{\text{IPW}} (\Sigma^\top)^{-1} \right\|_2 = o_p(1).$$

Lemma 8 indicates that  $\|\Omega_{\text{IPW}}\|_2 = O_p(1)$ . This together with Lemma 7 and the condition  $L \ll \sqrt{nT}/\log(nT)$  yields that

$$\begin{aligned} \left\| \widehat{\Sigma}_{\text{IPW}}^{-1} \Omega_{\text{IPW}} (\widehat{\Sigma}_{\text{IPW}}^\top)^{-1} - \Sigma^{-1} \Omega_{\text{IPW}} (\Sigma^\top)^{-1} \right\|_2 &\leq \left\| \widehat{\Sigma}_{\text{IPW}}^{-1} - \Sigma^{-1} \right\|_2 \|\Omega_{\text{IPW}}\|_2 \|\widehat{\Sigma}_{\text{IPW}}^{-1}\|_2 \\ &\quad + \|\Sigma^{-1}\|_2 \|\Omega_{\text{IPW}}\|_2 \|\widehat{\Sigma}_{\text{IPW}}^{-1} - \Sigma^{-1}\|_2 \\ &= O_p\{L^{1/2}(nT)^{-1/2} \log(nT)\} = o_p(1). \end{aligned}$$

Thus, it remains to show  $\|\widehat{\Sigma}_{\text{IPW}}^{-1} \widehat{\Omega}_{\text{IPW}} (\widehat{\Sigma}_{\text{IPW}}^\top)^{-1} - \widehat{\Sigma}_{\text{IPW}}^{-1} \Omega_{\text{IPW}} (\widehat{\Sigma}_{\text{IPW}}^\top)^{-1}\|_2 = o_p(1)$ , or

$$\|\widehat{\Omega}_{\text{IPW}} - \Omega_{\text{IPW}}\|_2 = o_p(1). \quad (\text{A.32})$$

In view of (A.31), we only need to show  $\|\widehat{\Omega}_{\text{IPW}} - \frac{1}{nT} \sum_{g=1}^{nT} \zeta^{(g)} \zeta^{(g)\top}\|_2 = o_p(1)$ . Notice that

$$\begin{aligned} \widehat{\Omega}_{\text{IPW}} - \frac{1}{nT} \sum_{g=1}^{nT} \zeta^{(g)} \zeta^{(g)\top} &= \frac{1}{nT} \sum_{g=1}^{nT} \left\{ \widehat{\zeta}^{(g)} \widehat{\zeta}^{(g)\top} - \zeta^{(g)} \zeta^{(g)\top} \right\} \\ &= \frac{1}{nT} \sum_{g=1}^{nT} \left\{ \left( \widehat{\zeta}^{(g)} - \zeta^{(g)} \right) \widehat{\zeta}^{(g)\top} + \zeta^{(g)} \left( \widehat{\zeta}^{(g)} - \zeta^{(g)} \right)^\top \right\}. \end{aligned}$$

By the triangle inequality, we have

$$\left\| \widehat{\Omega}_{\text{IPW}} - \frac{1}{nT} \sum_{g=1}^{nT} \zeta^{(g)} \zeta^{(g)\top} \right\|_2 \leq \left\| \frac{1}{nT} \sum_{g=1}^{nT} \left( \widehat{\zeta}^{(g)} - \zeta^{(g)} \right) \widehat{\zeta}^{(g)\top} \right\|_2 + \left\| \frac{1}{nT} \sum_{g=1}^{nT} \zeta^{(g)} \left( \widehat{\zeta}^{(g)} - \zeta^{(g)} \right)^\top \right\|_2. \quad (\text{A.33})$$

It suffices to show

$$\left\| \frac{1}{nT} \sum_{g=1}^{nT} \left( \widehat{\zeta}^{(g)} - \zeta^{(g)} \right) \widehat{\zeta}^{(g)\top} \right\|_2 = o_p(1) \quad \text{and} \quad \left\| \frac{1}{nT} \sum_{g=1}^{nT} \zeta^{(g)} \left( \widehat{\zeta}^{(g)} - \zeta^{(g)} \right)^\top \right\|_2 = o_p(1).$$

Recall that  $\widehat{\zeta}^{(g)} = \widehat{\omega}^{(g)}\widehat{\varepsilon}^{(g)}\boldsymbol{\xi}^{(g)} + \widehat{\mathbf{H}}_2(\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)})$ , where

$$\widehat{\varepsilon}^{(g)} = R_{i^{(g)}, t^{(g)}+1} + \gamma \sum_{a \in \mathcal{A}} \pi(a | S_{i^{(g)}, t^{(g)}+1}) \Phi_L^\top(S_{i^{(g)}, t^{(g)}+1}) \widehat{\boldsymbol{\beta}}_a - \Phi_L^\top(S_{i^{(g)}, t^{(g)}}) \widehat{\boldsymbol{\beta}}_{A_{i^{(g)}, t^{(g)}}},$$

and  $\widehat{\omega}^{(g)}, \widehat{\mathbf{H}}_2, \widehat{\mathbf{m}}^{(g)}, \widehat{\boldsymbol{\tau}}^{(g)}$  are obtained by plugging in  $\widehat{\boldsymbol{\psi}}$ .

We first show  $\|\frac{1}{nT} \sum_{g=1}^{nT} \zeta^{(g)}(\widehat{\zeta}^{(g)} - \zeta^{(g)})^\top\|_2 = \|\frac{1}{nT} \sum_{g=1}^{nT} (\widehat{\zeta}^{(g)} - \zeta^{(g)})\zeta^{(g)\top}\|_2 = o_p(1)$ , the other statement can be shown using similar arguments. By definition,  $\widehat{\zeta}^{(g)} - \zeta^{(g)}$  can be expressed as

$$\begin{aligned} \widehat{\zeta}^{(g)} - \zeta^{(g)} &= \widehat{\omega}^{(g)}\widehat{\varepsilon}^{(g)}\boldsymbol{\xi}^{(g)} + \widehat{\mathbf{H}}_2(\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)}) - \omega^{*(g)}\varepsilon^{(g)}\boldsymbol{\xi}^{(g)} - \mathbf{H}_2(\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}) \\ &= \underbrace{(\widehat{\omega}^{(g)}\widehat{\varepsilon}^{(g)} - \omega^{*(g)}\varepsilon^{(g)})\boldsymbol{\xi}^{(g)}}_{\mathbf{E}_1^{(g)}} + \underbrace{\widehat{\mathbf{H}}_2(\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)}) - \mathbf{H}_2(\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})}_{\mathbf{E}_2^{(g)}}, \end{aligned}$$

and  $\zeta^{(g)}$  can be expressed as

$$\zeta^{(g)} = \underbrace{\omega^{*(g)}\varepsilon^{(g)}\boldsymbol{\xi}^{(g)}}_{\mathbf{E}_3^{(g)}} + \underbrace{\mathbf{H}_2(\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})}_{\mathbf{E}_4^{(g)}}.$$

By the triangle inequality,

$$\begin{aligned} \left\| \frac{1}{nT} \sum_{g=1}^{nT} (\widehat{\zeta}^{(g)} - \zeta^{(g)})\zeta^{(g)\top} \right\|_2 &\leq \left\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_1^{(g)} \mathbf{E}_3^{(g)\top} \right\|_2 + \left\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_1^{(g)} \mathbf{E}_4^{(g)\top} \right\|_2 \\ &\quad + \left\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_2^{(g)} \mathbf{E}_3^{(g)\top} \right\|_2 + \left\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_2^{(g)} \mathbf{E}_4^{(g)\top} \right\|_2. \end{aligned}$$

Thus, it suffices to show  $\|\frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_i^{(g)} \mathbf{E}_j^{(g)\top}\|_2 = o_p(1)$  for all  $i \in \{1, 2\}, j \in \{3, 4\}$ .

We first show  $\|\frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_1^{(g)} \mathbf{E}_3^{(g)\top}\|_2 = o_p(1)$ . It is equivalent to showing

$$\sup_{\mathbf{a} \in \mathbb{S}^{mL-1}} \left| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{a}^\top \boldsymbol{\xi}^{(g)} \boldsymbol{\xi}^{(g)\top} \mathbf{a} (\widehat{\omega}^{(g)}\widehat{\varepsilon}^{(g)} - \omega^{*(g)}\varepsilon^{(g)}) \omega^{*(g)}\varepsilon^{(g)} \right| = o_p(1),$$

where  $\mathbb{S}^{mL-1}$  denotes the unit sphere  $\{\mathbf{a} \in \mathbb{R}^{mL} : \|\mathbf{a}\|_2 = 1\}$ . According to Lemma 4,  $\sup_{\mathbf{a} \in \mathbb{S}^{mL-1}} \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{a}^\top \boldsymbol{\xi}^{(g)} \boldsymbol{\xi}^{(g)\top} \mathbf{a} = O_p(1)$ , hence we only need to show  $\max_{1 \leq g \leq nT} |(\widehat{\omega}^{(g)}\widehat{\varepsilon}^{(g)} - \omega^{*(g)}\varepsilon^{(g)})\widehat{\omega}^{(g)}\widehat{\varepsilon}^{(g)}| = o_p(1)$ . The bound in (A.5) indicates that  $\varepsilon^{(g)}$ 's are uniformly bounded, together with the bound for  $\omega^{*(g)}$  given in condition 5(b), we obtain  $\max_{1 \leq g \leq nT} |\omega^{*(g)}\varepsilon^{(g)}| =$

$O_p(1)$ . Therefore, it remains to show

$$\max_{1 \leq g \leq nT} |\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)}| = o_p(1).$$

Note that the term can be decomposed as  $\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)} = (\widehat{\omega}^{(g)} - \omega^{*(g)}) \widehat{\varepsilon}^{(g)} + \omega^{*(g)} (\widehat{\varepsilon}^{(g)} - \varepsilon^{(g)})$ . Using similar arguments in showing (E.50) in Shi et al. (2021b), we have  $\max_{1 \leq g \leq nT} |\varepsilon^{(g)} - \widehat{\varepsilon}^{(g)}| = o_p(1)$ . On the other hand, the consistency of dropout propensity model,  $\widehat{\boldsymbol{\psi}} \xrightarrow{p} \boldsymbol{\psi}^*$ , indicates that  $\widehat{\omega}^{(g)} \xrightarrow{p} \omega^{*(g)}$  for any  $g$ , thus  $\max_{1 \leq g \leq nT} |\omega^{(g)} - \widehat{\omega}^{(g)}| = o_p(1)$ . Combine them together yields

$$\begin{aligned} \max_{1 \leq g \leq nT} |\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)}| &= \max_{1 \leq g \leq nT} |(\widehat{\omega}^{(g)} - \omega^{*(g)}) \widehat{\varepsilon}^{(g)} + \omega^{*(g)} (\widehat{\varepsilon}^{(g)} - \varepsilon^{(g)})| \\ &\leq \max_{1 \leq g \leq nT} |\widehat{\omega}^{(g)} - \omega^{*(g)}| |\widehat{\varepsilon}^{(g)}| + \max_{1 \leq g \leq nT} |\omega^{*(g)}| |\widehat{\varepsilon}^{(g)} - \varepsilon^{(g)}| = o_p(1). \end{aligned} \tag{A.34}$$

This completes the proof for  $\|\frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_1^{(g)} \mathbf{E}_3^{(g)\top}\|_2 = o_p(1)$ .

Next, we show  $\|\frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_2^{(g)} \mathbf{E}_4^{(g)\top}\|_2 = o_p(1)$ . Using similar arguments in bounding  $\|\widehat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}\|_2$  in the proof of Lemma 3, we obtain

$$\|\widehat{\mathbf{H}}_2 - \mathbf{H}_2\|_2 = o_p(1), \quad \left\| \frac{1}{nT} \sum_{g=1}^{nT} (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top - \boldsymbol{\Omega}_{\boldsymbol{\psi}^*} \right\|_2 = o_p(1),$$

where

$$\boldsymbol{\Omega}_{\boldsymbol{\psi}^*} = \frac{1}{T} \sum_{t=0}^T \mathbb{E} \{ \mathbf{m}_t^* + \boldsymbol{\tau}_t^* \} \{ \mathbf{m}_t^* + \boldsymbol{\tau}_t^* \}^\top$$

It follows from  $\widehat{\boldsymbol{\psi}} \xrightarrow{p} \boldsymbol{\psi}^*$  that  $\frac{1}{nT} \sum_{g=1}^{nT} \|\widehat{\mathbf{m}}^{(g)} - \mathbf{m}^{*(g)}\|_2^2 \xrightarrow{p} 0$ ,  $\frac{1}{nT} \sum_{g=1}^{nT} \|\widehat{\boldsymbol{\tau}}^{(g)} - \boldsymbol{\tau}^{*(g)}\|_2^2 \xrightarrow{p} 0$ . By

the triangle inequality, we have  $\frac{1}{nT} \sum_{g=1}^{nT} \|(\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)}) - (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})\|_2^2 \xrightarrow{p} 0$ . Therefore,

$$\begin{aligned}
& \left\| \frac{1}{nT} \sum_{g=1}^{nT} (\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top - \frac{1}{nT} \sum_{g=1}^{nT} (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top \right\|_2 \\
& \leq \frac{1}{nT} \sum_{g=1}^{nT} \|(\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top - (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top\|_2 \\
& \leq \frac{1}{nT} \sum_{g=1}^{nT} \|(\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)}) - (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})\|_2 \| \mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)} \|_2 \\
& \leq \left( \frac{1}{nT} \sum_{g=1}^{nT} \|(\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)}) - (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})\|_2^2 \right) \left( \frac{1}{nT} \sum_{g=1}^{nT} \| \mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)} \|_2^2 \right) \xrightarrow{p} 0,
\end{aligned}$$

the last step is because the convergence of the diagonal elements of  $\frac{1}{nT} \sum_{g=1}^{nT} (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top$  implies that  $\frac{1}{nT} \sum_{g=1}^{nT} \| \mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)} \|_2^2$  is bounded in probability. Based on the aforementioned results, we can show

$$\begin{aligned}
& \left\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_2^{(g)} \mathbf{E}_4^{(g)\top} \right\|_2 \\
& = \left\| \frac{1}{nT} \sum_{g=1}^{nT} \left\{ \widehat{\mathbf{H}}_2 (\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top \mathbf{H}_2 - \mathbf{H}_2 (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top \mathbf{H}_2 \right\} \right\|_2 \\
& \leq \left\| (\widehat{\mathbf{H}}_2 - \mathbf{H}_2) \left\{ \frac{1}{nT} \sum_{g=1}^{nT} (\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top \right\} \mathbf{H}_2 \right\|_2 \\
& + \left\| \mathbf{H}_2 \left\{ \frac{1}{nT} \sum_{g=1}^{nT} (\widehat{\mathbf{m}}^{(g)} + \widehat{\boldsymbol{\tau}}^{(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top - (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}) (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top \right\} \mathbf{H}_2 \right\|_2 \\
& = o_p(1).
\end{aligned}$$

It remains to show  $\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_1^{(g)} \mathbf{E}_4^{(g)\top} \|_2 = o_p(1)$  and  $\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_2^{(g)} \mathbf{E}_3^{(g)\top} \|_2 = o_p(1)$ . They can be shown in similar ways, here we only prove  $\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_1^{(g)} \mathbf{E}_4^{(g)\top} \|_2 = o_p(1)$  for

brevity. Notice that

$$\begin{aligned}
& \left\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_1^{(g)} \mathbf{E}_4^{(g)\top} \right\|_2 = \left\| \left\{ \frac{1}{nT} \sum_{g=1}^{nT} (\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)}) \boldsymbol{\xi}^{(g)} (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top \right\} \mathbf{H}_2^\top \right\|_2 \\
& \leq \left\| \frac{1}{nT} \sum_{g=1}^{nT} (\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)}) \boldsymbol{\xi}^{(g)} (\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)})^\top \right\|_2 \|\mathbf{H}_2\|_2 \\
& \leq \frac{1}{nT} \sum_{g=1}^{nT} \|(\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)}) \boldsymbol{\xi}^{(g)}\|_2 \|\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}\|_2 \|\mathbf{H}_2\|_2 \\
& \leq \left( \frac{1}{nT} \sum_{g=1}^{nT} \|(\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)}) \boldsymbol{\xi}^{(g)}\|_2^2 \right) \left( \frac{1}{nT} \sum_{g=1}^{nT} \|\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}\|_2^2 \right) \|\mathbf{H}_2\|_2.
\end{aligned}$$

Since  $\frac{1}{nT} \sum_{g=1}^{nT} \|\mathbf{m}^{*(g)} + \boldsymbol{\tau}^{*(g)}\|_2^2$  and  $\|\mathbf{H}_2\|_2$  are bounded in probability, it suffices to show  $\frac{1}{nT} \sum_{g=1}^{nT} \|(\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)}) \boldsymbol{\xi}^{(g)}\|_2^2 = o_p(1)$ . We have already prove  $\max_{1 \leq g \leq nT} |\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)}| = o_p(1)$  in (A.34), hence  $\max_{1 \leq g \leq nT} (\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)})^2 = o_p(1)$ . Meanwhile, by Lemma 4, we have  $\sup_{\mathbf{a} \in \mathbb{S}^{mL-1}} \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{a}^\top \boldsymbol{\xi}^{(g)} \boldsymbol{\xi}^{(g)\top} \mathbf{a} = O_p(1)$ . Thus,

$$\sup_{\mathbf{a} \in \mathbb{S}^{mL-1}} \left| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{a}^\top \boldsymbol{\xi}^{(g)} \boldsymbol{\xi}^{(g)\top} \mathbf{a} (\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)})^2 \right| = o_p(1),$$

equivalently,  $\frac{1}{nT} \sum_{g=1}^{nT} \|(\widehat{\omega}^{(g)} \widehat{\varepsilon}^{(g)} - \omega^{*(g)} \varepsilon^{(g)}) \boldsymbol{\xi}^{(g)}\|_2^2 = o_p(1)$ . This completes the proof for  $\left\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_1^{(g)} \mathbf{E}_4^{(g)\top} \right\|_2 = o_p(1)$ . We can also show  $\left\| \frac{1}{nT} \sum_{g=1}^{nT} \mathbf{E}_2^{(g)} \mathbf{E}_3^{(g)\top} \right\|_2 = o_p(1)$  using similar steps.

Combine these results together yields  $\left\| \frac{1}{nT} \sum_{g=1}^{nT} \zeta^{(g)} (\widehat{\zeta}^{(g)} - \zeta^{(g)})^\top \right\|_2 = o_p(1)$ . Similarly we can show  $\left\| \frac{1}{nT} \sum_{g=1}^{nT} (\widehat{\zeta}^{(g)} - \zeta^{(g)}) \widehat{\zeta}^{(g)\top} \right\|_2 = o_p(1)$ . Therefore,  $\|\widehat{\boldsymbol{\Omega}}_{\text{IPW}} - \frac{1}{nT} \sum_{g=1}^{nT} \zeta^{(g)} \zeta^{(g)\top}\|_2 = o_p(1)$ , and hence (A.32) is proven.  $\square$

#### A.5.4 Proof of Theorem 4

*Proof.* Similar to the proof of Theorem 1, we define  $\varepsilon_{i,t}$  as follows:

$$\varepsilon_{i,t} = R_{i,t+1} + \gamma \sum_{a \in \mathcal{A}} Q^\pi(S_{i,t+1}, a) \pi(a|S_{i,t+1}) - Q^\pi(S_{i,t}, A_{i,t}),$$

and use  $\mathcal{F}_t = \{(S_j, A_j, R_{j+1})\}_{0 \leq j < t} \cup \{S_t, A_t\}$  to denote the past information up to time  $t$ . Based on Assumption 1, 2, and Bellman equation,  $\varepsilon_{i,t}$  satisfies  $\mathbb{E}(\varepsilon_t | \mathcal{F}_t) = \mathbb{E}(\varepsilon_t | S_t, A_t) = 0$ . For simplicity, we use  $\widehat{\omega}_{\pi,i,t}$  and  $\omega_{\pi,i,t}$  to represent  $\widehat{\omega}_{\pi,nT}(S_{i,t}, A_{i,t})$  and  $\omega_\pi(S_{i,t}, A_{i,t})$  respectively.

The value estimation error  $\widehat{V}_{\text{CC}}^\pi(\mathbb{G}) - V^\pi(\mathbb{G})$  can be decomposed as

$$\begin{aligned}
& \widehat{V}_{\text{CC}}^\pi(\mathbb{G}) - V^\pi(\mathbb{G}) \\
&= \frac{1}{1-\gamma} \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \widehat{\omega}_{\pi,i,t} R_{i,t+1} - \mathbb{E}_{S_0 \sim \mathbb{G}} \left\{ \sum_{a \in \mathcal{A}} \pi(a|S_0) Q^\pi(S_0, a) \right\} \\
&= \frac{1}{1-\gamma} \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \widehat{\omega}_{\pi,i,t} \left[ \varepsilon_{i,t+1} + Q^\pi(S_{i,t}, A_{i,t}) - \gamma \sum_{a \in \mathcal{A}} Q^\pi(S_{i,t+1}, a) \pi(a|S_{i,t+1}) \right] \\
&\quad - \mathbb{E}_{S_0 \sim \mathbb{G}} \left\{ \sum_{a \in \mathcal{A}} \pi(a|S_0) Q^\pi(S_0, a) \right\} \\
&= - \frac{1}{1-\gamma} \mathcal{L}_{nT}(\widehat{\omega}_\pi, Q^\pi) \cdots \cdots \text{(I)} \\
&\quad + \frac{1}{1-\gamma} \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \widehat{\omega}_{\pi,i,t} \varepsilon_{i,t+1} \cdots \cdots \text{(II)}
\end{aligned}$$

For (I), note that  $\mathcal{L}_{nT}(\widehat{\omega}_\pi, Q^\pi)$  captures the difference between two sides of equation (2.5) under the estimated density ratio and the true Q-function. This loss term is dependent on the specific algorithm, the choice of function class  $\mathcal{Q}$  and the computation procedure. Here we assume this term converges to 0. For (II), by applying the Cauchy inequality, we have

$$\begin{aligned}
& \mathbb{E} \left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \widehat{\omega}_{\pi,i,t} \varepsilon_{i,t} \right\}^2 \\
& \leq \mathbb{E} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \widehat{\omega}_{\pi,i,t}^2 \right) \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1}^2 \varepsilon_{i,t}^2 \right) \\
& \leq c_\omega^2 \cdot \mathbb{E} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1}^2 \varepsilon_{i,t}^2 \right)
\end{aligned} \tag{A.35}$$

The last inequality follows from the boundedness of  $\widehat{\omega}_{\pi,i,t}$ . Next, we derive the bound for  $\mathbb{E} \left( (nT)^{-1} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1}^2 \varepsilon_{i,t}^2 \right)$ . Under the MAR assumption,  $\eta_{t+1}$  and  $\varepsilon_t$  are conditionally independent, it follows that

$$\mathbb{E} \{ \eta_{t+1} \varepsilon_t \} = \mathbb{E} \{ \mathbb{E}(\eta_{t+1} \varepsilon_t | \mathcal{F}_t, \eta_t) \} = \mathbb{E} \{ \mathbb{E}(\eta_{t+1} | \mathcal{F}_t, \eta_t) \mathbb{E}(\varepsilon_t | \mathcal{F}_t) \} = \mathbf{0}.$$

Similarly, for any  $0 \leq t_1 < t_2 < T$ , we obtain  $\mathbb{E} \{ \eta_{t_1+1} \eta_{t_2+1} \varepsilon_{t_1} \varepsilon_{t_2} \} = 0$ . In addition, by the independence assumption among trajectories, we have

$$\mathbb{E} \{ \eta_{i_1, t_1+1} \eta_{i_2, t_2+1} \varepsilon_{i_1, t_1} \varepsilon_{i_2, t_2} \} = 0.$$

Applying the bound for  $\varepsilon_t$  derived from (A.5) yields

$$\begin{aligned} \mathbb{E} \left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \varepsilon_{i,t} \right\}^2 &= \frac{1}{(nT)^2} \sum_{i=1}^n \sum_{t=0}^{T-1} \mathbb{E} \{ \eta_{i,t+1}^2 \varepsilon_{i,t}^2 \} = \frac{1}{(nT)^2} \cdot n \sum_{t=0}^{T-1} \mathbb{E} \{ \eta_{t+1}^2 \varepsilon_t^2 \} \\ &\leq \frac{1}{nT} (c_0 + 2c')^2. \end{aligned} \quad (\text{A.36})$$

Combine (A.35) and (A.36), we have

$$\mathbb{E} \left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \widehat{\omega}_{\pi,i,t} \varepsilon_{i,t} \right\}^2 \leq \frac{1}{nT} (c_0 + 2c')^2 c_\omega^2.$$

By Markov inequality,

$$\frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \widehat{\omega}_{\pi,i,t} \varepsilon_{i,t} = O_p\{(nT)^{-1/2}\}.$$

As a result,  $\widehat{V}_{CC}^\pi(\mathbb{G}) \xrightarrow{p} V^\pi(\mathbb{G})$  as  $nT \rightarrow \infty$ , indicating that  $\widehat{V}_{CC}^\pi(\mathbb{G})$  is a consistent estimator of  $V^\pi(\mathbb{G})$  under ignorable missingness.

However, when the missingness is nonignorable, the conditional independence between  $\eta_{t+1}$  and  $\varepsilon_t$  no longer holds. As a result, the convergence of (II) to 0 is not guaranteed, and the complete-case value estimator  $\widehat{V}_{CC}^\pi(\mathbb{G})$  will be biased from  $V^\pi(\mathbb{G})$ .  $\square$

### A.5.5 Proof of Lemma 7

*Proof.* Recall that

$$\begin{aligned} \widehat{\Sigma}_{\text{IPW}} &= \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \widehat{\omega}_{i,t+1} \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top \\ \Sigma &= \mathbb{E} \left\{ \omega_{i,t+1}^* \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1}) \right\} = \mathbb{E} \left\{ \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1}) \right\}. \end{aligned}$$

It follows that

$$\begin{aligned}
\widehat{\Sigma}_{\text{IPW}} - \Sigma &= \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \widehat{\omega}_{i,t+1} \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top - \Sigma \\
&= \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} (\widehat{\omega}_{i,t+1} - \omega_{i,t+1}^*) \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top \right) \\
&\quad + \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}^* \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top - \Sigma \right)
\end{aligned}$$

Using similar arguments in proving Lemma 3, we obtain

$$\left\| \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}^* \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top - \Sigma \right\|_2 = O_p \{ L^{1/2} (nT)^{-1/2} \log(nT) \}.$$

On the other hand, using a similar technique as in (A.16), we have

$$\frac{1}{\sqrt{nT}} \sum_{i=1}^n \sum_{t=0}^{T-1} (\widehat{\omega}_{i,t+1} - \omega_{i,t+1}^*) \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top = \sqrt{nT} \mathbf{H}_3 (\widehat{\boldsymbol{\psi}} - \boldsymbol{\psi}^*) + o_p(1),$$

for some tensor  $\mathbf{H}_3$ . By convergence of  $\widehat{\boldsymbol{\psi}}$  given in (A.18), we have

$$\left\| \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} (\widehat{\omega}_{i,t+1} - \omega_{i,t+1}^*) \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top \right\|_2 = O_p \{ (nT)^{-1/2} \}.$$

Therefore,

$$\begin{aligned}
\left\| \widehat{\Sigma}_{\text{IPW}} - \Sigma \right\|_2 &\leq \left\| \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \omega_{i,t+1}^* \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top - \Sigma \right\|_2 \\
&\quad + \left\| \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} (\widehat{\omega}_{i,t+1} - \omega_{i,t+1}^*) \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi,i,t+1})^\top \right\|_2 \\
&= O_p \{ L^{1/2} (nT)^{-1/2} \log(nT) \} + O_p \{ (nT)^{-1/2} \} \\
&= O_p \{ L^{1/2} (nT)^{-1/2} \log(nT) \}.
\end{aligned}$$

Based on this result, we can follow similar steps in the proof for Lemma 3 to show  $\|\widehat{\Sigma}_{\text{IPW}}^{-1}\|_2 \leq 6\bar{c}^{-1}$  and  $\|\widehat{\Sigma}_{\text{IPW}}^{-1} - \Sigma^{-1}\|_2 = O_p \{ L^{1/2} (nT)^{-1/2} \log(nT) \}$ . The proof is hence completed.  $\square$

### A.5.6 Connection Between LSTDQ and MWL

Here we provide a detailed derivation to show that MWL is equivalent to LSTDQ when  $\omega_\pi(s, a)$  and  $Q^\pi(s, a)$  are modeled with the same set of basis functions, i.e.,  $\omega_\pi(s, a) = \Phi_L(s)^\top \boldsymbol{\alpha}_{\pi, a}$  and  $Q^\pi(s, a) = \Phi_L(s)^\top \boldsymbol{\beta}_{\pi, a}$ .

Let  $\boldsymbol{\alpha}_\pi = (\boldsymbol{\alpha}_{\pi, 1}^\top, \dots, \boldsymbol{\alpha}_{\pi, m}^\top)^\top$ ,  $\boldsymbol{\beta}_\pi = (\boldsymbol{\beta}_{\pi, 1}^\top, \dots, \boldsymbol{\beta}_{\pi, m}^\top)^\top$ , using the notation of  $\boldsymbol{\xi}(s, a)$  and  $\mathbf{U}_\pi(s)$  defined in Section 2.3.2,  $\omega_\pi$  and  $Q^\pi$  can be expressed as  $\omega_\pi(s, a) = \boldsymbol{\xi}(s, a)^\top \boldsymbol{\alpha}_\pi$ ,  $Q^\pi(s, a) = \boldsymbol{\xi}(s, a)^\top \boldsymbol{\beta}_\pi$ . Similarly, we denote  $\boldsymbol{\xi}(S_t, A_t)$  and  $\mathbf{U}_\pi(S_t)$  as  $\boldsymbol{\xi}_t$  and  $\mathbf{U}_{\pi, t}$  for simplicity. The objective function in (2.11) is equivalent to

$$\begin{aligned} \mathcal{L}_{nT}(\omega_\pi, Q^\pi) &= \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i, t+1} \omega_\pi(S_{i, t}, A_{i, t}) \left( \gamma \sum_{a' \in \mathcal{A}} \pi(a' | S_{i, t+1}) Q^\pi(S_{i, t+1}, a') - Q^\pi(S_{i, t}, A_{i, t}) \right) \\ &\quad + (1 - \gamma) \cdot \mathbb{E}_{S_0 \sim \mathbb{G}} \left\{ \sum_{a \in \mathcal{A}} \pi(a | S_0) Q^\pi(S_0, a) \right\} \\ &= \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i, t+1} \boldsymbol{\alpha}_\pi^\top \boldsymbol{\xi}_{i, t} (\gamma \mathbf{U}_{\pi, i, t+1}^\top - \boldsymbol{\xi}_{i, t}^\top) \boldsymbol{\beta}_\pi + (1 - \gamma) \cdot \left\{ \int_s \mathbf{U}_\pi^\top(s) \mathbb{G}(ds) \right\} \boldsymbol{\beta}_\pi \\ &= \left\{ \boldsymbol{\alpha}_\pi^\top \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i, t+1} \boldsymbol{\xi}_{i, t} (\gamma \mathbf{U}_{\pi, i, t+1}^\top - \boldsymbol{\xi}_{i, t}^\top) \right) + (1 - \gamma) \cdot \int_s \mathbf{U}_\pi^\top(s) \mathbb{G}(ds) \right\} \boldsymbol{\beta}_\pi \end{aligned}$$

Notice that  $\mathcal{L}_{nT}(\omega_\pi, Q^\pi)$  is linear in  $\boldsymbol{\beta}_\pi$ . To attain  $\max_{Q^\pi \in \mathcal{Q}} \mathcal{L}_{nT}(\omega_\pi, Q^\pi)^2 = 0$ , it suffices to satisfy

$$\boldsymbol{\alpha}_\pi^\top \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i, t+1} \boldsymbol{\xi}_{i, t} (\gamma \mathbf{U}_{\pi, i, t+1}^\top - \boldsymbol{\xi}_{i, t}^\top) \right) + (1 - \gamma) \cdot \int_s \mathbf{U}_\pi^\top(s) \mathbb{G}(ds) = \mathbf{0}.$$

Consequently, the estimation of  $\boldsymbol{\alpha}_\pi$  is given by

$$\hat{\boldsymbol{\alpha}}_\pi = (1 - \gamma) \left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i, t+1} (\boldsymbol{\xi}_{i, t} - \gamma \mathbf{U}_{\pi, i, t+1}) \boldsymbol{\xi}_{i, t}^\top \right\}^{-1} \left\{ \int_s \mathbf{U}_\pi(s) \mathbb{G}(ds) \right\}.$$

Finally, by substituting  $\widehat{\boldsymbol{\alpha}}_\pi$  into the value estimator, we arrive at the following estimator

$$\begin{aligned}
& \widehat{V}_{\text{CC}}^\pi(\mathbb{G}) \\
&= \frac{1}{1-\gamma} \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \widehat{\omega}_{\pi, nT}(S_{i,t}, A_{i,t}) R_{i,t+1} \\
&= \frac{1}{1-\gamma} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t}^\top R_{i,t+1} \right) \widehat{\boldsymbol{\alpha}}_\pi \\
&= \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t}^\top R_{i,t+1} \right) \left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi, i, t+1}) \boldsymbol{\xi}_{i,t}^\top \right\}^{-1} \left\{ \int_s \mathbf{U}_\pi(s) \mathbb{G}(ds) \right\} \\
&= \left\{ \int_s \mathbf{U}_\pi(s) \mathbb{G}(ds) \right\} \left\{ \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} (\boldsymbol{\xi}_{i,t} - \gamma \mathbf{U}_{\pi, i, t+1})^\top \right\}^{-1} \left( \frac{1}{nT} \sum_{i=1}^n \sum_{t=0}^{T-1} \eta_{i,t+1} \boldsymbol{\xi}_{i,t} R_{i,t+1} \right). \tag{A.37}
\end{aligned}$$

This estimator is identical to the complete-case LSTDQ estimator discussed in Section 2.3.2.

# Appendix B

## Offline Inverse Reinforcement Learning via Joint Soft-Q and Reward Learning

### B.1 Additional Experimental Details

**Gridworld Experiment** In this experiment, we evaluate the performance of IQRL in a tabular GridWorld environment <sup>1</sup>. The action space consists of five possible actions: (up, down, left, right, stay), with a 0.3 probability of random movement. Here we follow the steps of Chan and van der Schaar (2021) and Garg et al. (2021) for this experiment: the optimal policy is first obtained through value iteration under discount factor  $\gamma = 0.9$ , then the expert demonstrations are collected by generating  $n = 30$  trajectories under the learned optimal policy with a horizon of  $T = 20$  and random initial states. To accommodate the tabular environment, we modified our implementation to support tabular state space. Instead of utilizing function approximation, we directly learned the state-only reward  $r(s)$  for each state and the soft Q-function  $Q^\pi(s, a)$  for each state-action pair. The temperature parameter is set to  $\alpha = 1$  and the soft Bellman residual weight is set to  $\lambda = 0.05$ . The parameters are updated until convergence via Adam optimizer (Kingma and Ba 2014) with a batch size of 256 and a learning rate of  $5 \times 10^{-3}$ .

**2D-Linear Experiment** In this experiment, we evaluate the performance of IQRL in a continuous 2D-Linear environment. The transition dynamics and the ground-truth reward function are described in Section 3.4.1. We first learn the optimal policy using the Proximal Policy Optimization (PPO) algorithm (Schulman et al. 2017) via the Stable Baselines3 library (Raffin et al. 2021). We use the hyper-parameters given in Stable Baselines3 Zoo (Raffin 2020)

---

<sup>1</sup>Open-source implementation of GridWorld: <https://github.com/yrlu/irl-imitation>

for the CartPole environment, which also yields satisfactory performance in the 2D-Linear environment. The discount factor is set to be  $\gamma = 0.99$ , and the training of the RL agent was executed until convergence. The expert demonstrations are then collected by generating  $n$  synthetic expert trajectories with a horizon of  $T$  following the learned policy, where  $n = 100$ ,  $T = 10, 20$ . In our implementation of IQRL, the reward function is parameterized with a linear model such that the weights add up to 1. The soft Q-function is modeled with neural networks consisting of 2 hidden layers, each with 64 units. The hidden layers were connected by exponential linear unit (ELU) activation functions. The temperature parameter is set to  $\alpha = 0.1$ . We further tuned the soft bellman residual weight  $\lambda$  via grid-search based on the action matching rate. Specifically, for  $\omega$  values of 0.3, 0.6, and 0.8, the  $\lambda$  values that yield the best performance are 0.75, 15 and 0.75, respectively. The parameters for the reward and soft Q-function are updated until convergence via Adam optimizer with a batch size of 64 and a learning rate of  $3 \times 10^{-4}$ .

**Classic Control Experiment** For the three classic control tasks (i.e., CartPole, Acrobot, and LunarLander), we generate demonstrations by running pre-trained and hyperparameter-optimized agents from the RL Baselines Zoo (Raffin 2020) in OpenAI Stable Baselines (Raffin et al. 2021). Following the experimental setup of Chan and van der Schaar (2021), we randomly sample (1,3,7,10,15) trajectories from a pool of 1000 expert trajectories for each algorithm. These trajectories are then sub-sampled, taking every 10th step in Acrobot and CartPole, and every 5th step in LunarLander. For the baseline methods, we utilize a PyTorch reimplementaion that is based on the publicly available code of AVRIL<sup>2</sup>, IQ-Learn<sup>3</sup>, and ValueDICE<sup>4</sup>. Note that the original ValueDICE is designed for continuous environment, we adapt it for discrete environments by using an actor with a Gumbel-softmax distribution output. For IQRL, the penalty weight for soft bellman residual is fixed to  $\lambda = 0.1$ . To ensure a fair comparison, all algorithms use neural networks with 2 hidden layers of 64 units each, connected by ELU activation functions. We use a temperature of 0.1 for CartPole and Acrobot, and 0.01 for LunarLander. Through experimentation, we find that when the sample size is relatively small in terms of the complexity of the task, a smaller temperature tends to yield better results. As the sample size grows, a slightly larger temperature typically leads to improved performance. All algorithms are trained until convergence, and evaluated by the value of the learned policy, which is calculated by performing 300 live roll-outs in the simulation environment. This entire process is repeated 10 times with different seeds and

---

<sup>2</sup>AVRIL code: <https://github.com/XanderJC/scalable-birl>

<sup>3</sup>IQ-Learn code: <https://github.com/Div99/IQ-Learn>

<sup>4</sup>ValueDICE code: [https://github.com/google-research/google-research/tree/master/value\\_dice](https://github.com/google-research/google-research/tree/master/value_dice)

seen trajectories.

**Multi-Strategy 2D-Linear Experiment** In this experiment, we assess the performance of D-IQRL in the presence of heterogeneous expert demonstrations, and compare it with local IQRL and centralized IQRL. We consider  $K = 3$  strategies. For each strategy, we obtain the optimal policy using the PPO algorithm and generate  $n = 100$  trajectories for each strategy with a horizon of  $T = 20$ . There are a total of  $n \times K = 300$  trajectories. Based on a grid search, the temperature parameter is set to 0.1 and the soft bellman residual weight  $\lambda_1$  is set to 0.75 for all three training approaches. For D-IQRL, the regularization weight of the strategy-specific reward is set to  $\lambda_2 = 0.01$ . The rest of the specifications are the same as in the single-strategy experiment. We run the algorithms until converge to obtain a more reliable reward estimation. During the evaluation, the reward correlation is calculated between the learned strategy reward for observed state-action pairs and the ground-truth strategy reward obtained from the environment. Additionally, the policy value is estimated by performing 100 roll-outs in the environment following the learned policy.

**Automated Hyperparameter Tuning** In the aforementioned experiments, the reward and Q-function are simultaneously learned under a prespecified penalty weight  $\lambda$ . This hyperparameter needs to be carefully tuned for each task via a grid search. Alternatively, we can automate  $\lambda$  tuning by applying the dual ascent (Boyd et al. 2004). This approach entails alternating between the optimization of the Lagrangian with respect to the primal variables  $\omega$  and  $\beta$ , followed by a gradient step on the dual variable  $\lambda$ . Based on the constrained objective in (3.10), we can write the objective for optimizing  $\lambda$  as follows

$$\mathcal{J}(\lambda) = \lambda \cdot \{ \mathbb{E}_{\mathcal{D}} [r_{\omega}(S_{i,t}, A_{i,t}) + \gamma \cdot V_{\beta}(S_{i,t+1}) - Q_{\beta}(S_{i,t}, A_{i,t})]^2 - \delta \}. \quad (\text{B.1})$$

The parameter  $\lambda$  is optimized by maximizing the objective function  $\mathcal{J}(\lambda)$  via gradient ascent. A similar strategy is also used in the automatic temperature tuning for the Soft Actor-Critic (SAC) algorithm (Haarnoja et al. 2018).

## B.2 Additional Details for Real Data Application

We extract the mechanical ventilation and antibiotics trajectories from the MIMIC-III database (Johnson et al. 2016) following the steps of Jarrett et al. (2021)<sup>5</sup>. The data is aggregated into one-day intervals. Different from the original implementation that only

---

<sup>5</sup>Code for MIMIC-III data extraction: [https://github.com/vanderschaarlab/clairvoyance/tree/main/datasets/mimic\\_data\\_extraction](https://github.com/vanderschaarlab/clairvoyance/tree/main/datasets/mimic_data_extraction)

considered patients who received antibiotic therapy, we included all available patients to learn the optimal treatment decision for both mechanical ventilation and antibiotic therapy. The dataset comprises a total of 18,313 trajectories in total, with 7,683 from MICU, 5,107 from CCU/CSRU, and 5,523 from SICU/TSICU. We adopt 5-fold cross-validation to tune the parameter and evaluate the performance. The temperature parameter  $\alpha$  is set to 1 for all algorithms. Besides, we set  $\lambda = 2.5$  for IQRL, and  $\lambda_1 = 0.01, \lambda_2 = 2.5$  for distilled IQRL. The models are trained until convergence with a batch size of 64 and a learning rate of  $10^{-3}$ .