JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION
2024, VOL. 119, NO. 545, 232–245: Theory and Methods
https://doi.org/10.1080/01621459.2022.2106868

Taylor & Francis
Taylor & Francis Group

OPEN ACCESS

Check for updates

Statistically Efficient Advantage Learning for Offline Reinforcement Learning in Infinite Horizons

Chengchun Shi[a], Shikai Luo[b], Yuan Le[c], Hongtu Zhu[d], and Rui Song[e]

[a]London School of Economics and Political Science, London, UK; [b]ByteDance, Beijing, China; [c]Shanghai University of Finance and Economics, Shanghai, China; [d]University of North Carolina at Chapel Hill, Chapel Hill, NC; [e]North Carolina State University, Raleigh, NC

ABSTRACT

We consider reinforcement learning (RL) methods in offline domains without additional online data collection, such as mobile health applications. Most of existing policy optimization algorithms in the computer science literature are developed in online settings where data are easy to collect or simulate. Their generalizations to mobile health applications with a pre-collected offline dataset remain are less explored. The aim of this article is to develop a novel advantage learning framework in order to efficiently use pre-collected data for policy optimization. The proposed method takes an optimal Q-estimator computed by any existing state-of-the-art RL algorithms as input, and outputs a new policy whose value is guaranteed to converge at a faster rate than the policy derived based on the initial Q-estimator. Extensive numerical experiments are conducted to back up our theoretical findings. A Python implementation of our proposed method is available at *https://github.com/leyuanheart/SEAL*. Supplementary materials for this article are available online..

ARTICLE HISTORY
Received November 2021
Accepted July 2022

KEYWORDS
Advantage learning; Infinite horizons; Mobile health applications; Rate of convergence; Reinforcement learning

1. Introduction

Reinforcement learning (RL, see Sutton and Barto 2018, for an overview) is concerned with how intelligence agents learn and take actions in an unknown environment in order to maximize the cumulative reward that it receives. It has been arguably one of the most vibrant research frontiers in machine learning over the last few years. According to Google Scholar, over 40K scientific articles have been published in 2020 with the phrase "reinforcement learning." Over 100 papers on RL were accepted for presentation at ICML 2021, a premier conference in the machine learning area, accounting for more than 10% of the accepted papers in total. RL algorithms have been applied in a wide variety of real applications, including games (Silver et al. 2016), robotics (Kormushev, Calinon, and Caldwell 2013), healthcare (Komorowski et al. 2018), bidding (Jin et al. 2018), ridesharing (Xu et al. 2018) and automated driving (de Haan, Jayaraman, and Levine 2019), to name a few.

This article is partly motivated by developing statistical learning methodologies in offline RL domains such as mobile health (mHealth). mHealth technologies have recently emerged due to the use of mobile phones, tablets computers or wearable devices. They play an important role in precision medicine as they offer a means to monitor a patient's health status and deliver interventions in real-time. They also collect rich longitudinal data for optimal treatment decision making. One motivating example being considered in this article uses the OhioT1DM Dataset (Marling and Bunescu 2018). It contains 8 weeks of data for 6 patients with type 1 diabetes, an autoimmune disease wherein the pancreas produces insufficient levels of insulin. For those patients, their continuous glucose monitoring blood glucose levels, insulin doses being injected, self-reported times of meals and exercises are continually measured. Their outcomes have the potential to be improved by treatment policies tailored to the continually evolving health status of each patient (Luckett et al. 2020; Shi et al. 2020c).

Despite the popularity of developing various RL algorithms in the computer science literature, statistics as a field, has only recently begun to engage with RL both in depth and in breadth. Most works in the literature focused on developing data-driven methodologies for precision medicine with only a few treatment stages (see e.g., Murphy 2003; Robins 2004; Chakraborty, Murphy, and Strecher 2010; Qian and Murphy 2011; Zhang et al. 2013; Zhao et al. 2015; Wallace and Moodie 2015; Song et al. 2015; Luedtke and van der Laan 2016; Zhu et al. 2017; Shi et al. 2018b; Wang et al. 2018; Zhang et al. 2018; Qi et al. 2020; Nie, Brunskill, and Wager 2020). These methods require a large number of patients in the observed data to be consistent. They are not applicable to mHealth applications with only a few patients, which is the case in the OhioT1DM dataset. Nor are they applicable to many other sequential decision making problems where the number of decision stages is allowed to diverge to infinity, such as games or robotics. Recently, a few algorithms have been proposed in the statistics literature for policy optimization in mHealth applications (Ertefaie and Strawderman 2018; Luckett et al. 2020; Hu et al. 2020; Liao, Qi, and Murphy 2020; Zhou, Zhu, and Qu 2021).

CONTACT Chengchun Shi *C.Shi7@lse.ac.uk* London School of Economics and Political Science, London, UK.

Supplementary materials for this article are available online. Please go to *www.tandfonline.com/r/JASA*.

Among all existing methods in infinite horizons, Q-learning (Watkins and Dayan 1992) is arguably one of the most popular model-free RL algorithms. It derives the optimal policy by learning an optimal Q-function, without explicitly modeling the system dynamics. Variants of Q-learning include gradient Q-learning (Maei et al. 2010; Ertefaie and Strawderman 2018), fitted Q-iteration (Riedmiller 2005), deep Q-network (DQN, Mnih et al. 2015), double DQN (Van Hasselt, Guez, and Silver 2016) and quantile DQN (Dabney et al. 2018), among others. All these Q-learning type algorithms are primarily motivated by the application of developing artificial intelligence in online video games, so their generalization to offline applications with a pre-collected dataset remains unknown.

Different from online settings (e.g., video games) where data are easy to collect or simulate, the number of observations in many offline applications (e.g., healthcare) is limited. Take the OhioT1DM dataset as an example, only a few thousands observations are available (Shi et al. 2020b). With such limited data, it is critical to develop RL algorithms that are *statistically efficient*. Instead of proposing a specific algorithm for policy optimization, our work undertakes the ambitious task of devising an “efficiency enhancement” method that is generally applicable to any Q-learning type algorithms to improve their statistical efficiency. The input of our method is an optimal Q-estimator computed by existing state-of-the-art RL algorithms and the output is a new policy whose value converges at a faster rate than the policy derived based on the initial Q-estimator.

The proposed method is motivated by a line of research on developing A-learning type algorithms[1] to learn an optimal dynamic treatment regime (DTR) to implement precision medicine (see e.g., Murphy 2003; Robins 2004; Lu, Zhang, and Zeng 2013). These methods directly model the difference between two conditional mean functions (known as the contrast function). They are semi-parametrically efficient and outperform Q-learning [2] (see e.g., Chakraborty, Murphy, and Strecher 2010; Qian and Murphy 2011) in cases where the Q-function is misspecified (Shi et al. 2018a). In addition, A-learning has the so-called doubly robustness property, that is, the estimated optimal DTR is consistent when either the model for the conditional mean function or the treatment assignment mechanism is correctly specified.

The contributions of our article are summarized as follows. Methodologically, we propose a statistically efficient advantage learning procedure to estimate the optimal policy in offline infinite horizon settings. Our proposal integrates existing policy optimization and policy evaluation algorithms in RL. Specifically, we start with applying existing Q-learning type algorithms to compute an initial estimator for the optimal Q-function. Based on these Q-estimators, we leverage ideas from the off-policy evaluation literature (OPE, see e.g., Jiang and Li 2016; Thomas and Brunskill 2016; Liu et al. 2018; Kallus and Uehara 2019, 2020; Shi et al. 2021) to construct pseudo outcomes that are asymptotically unbiased to the optimal contrast function (see Section 2.2 for the detailed definition). With these pseudo outcomes as the prediction target, we can directly apply existing state-of-the-art supervised learning algorithms to derive the optimal policy. The use of OPE effectively alleviates the bias of the estimated contrast function resulting from the potential model misspecification of the optimal Q-function, which in turn improves the statistical efficiency over Q-learning. In that sense, our proposal shares similar spirits with the A-learning type methods to learn DTRs in finite horizons.

Theoretically, we show our estimated contrast function converges at a faster rate than the Q-function computed by existing state-of-the-art Q-learning type algorithms (Theorem 2). This in turn implies that our estimated policy achieves a larger value function (Theorem 3). All the error bounds derived in this article converge to zero when either the number of trajectories N or the number of decision stages per trajectory T to approach infinity. This guarantees the consistency of our method when applied to a wide range of real-world problems, ranging from the OhioT1DM Dataset that contains eight weeks’ data for 6 patients to the 2018 Intern Health Study with over 1000 subjects (see e.g., NeCamp et al. 2020). It is also applicable to data generated from online video games where both N and T are allowed to grow to infinity.

Empirically, we show that our procedure outperforms existing learning algorithms using both synthetic datasets and a real dataset from the mobile health application. We remark that most papers in the existing literature use synthetic datasets to evaluate the performance of different RL algorithms. Results in our article offer a useful evaluation tool for assessing these algorithms in real applications.

The rest of this article is organized as follows. In Section 2, we introduce some basic concepts in RL, describe the data generating process and formulate the problem. In Section 3, we demonstrate the advantage of A-learning over Q-learning by comparing their rate of convergence. The proposed algorithm is formally presented in Section 4. In Section 5, we study the statistical properties of our algorithm, proving that our estimated policy achieves a faster rate of convergence than existing Q-learning type algorithms. In Section 6, we investigate the finite sample performance of the proposed algorithm using Monte Carlo simulations. In Section 7, we use the OhioT1DM Dataset to further demonstrate the empirical advantage of the proposed algorithm over other baseline algorithms. Proofs of our major theorems are presented in Appendix B of the supplementary materials.

2. Preliminaries

We first formulate the policy optimization problem in infinite horizon settings. We next briefly review Q-learning.

2.1. Problem Formulation

RL is concerned with solving sequential decision making problems in an unknown environment. The observed data can be summarized into a sequence of state-action-reward triplets over

[1] Similar algorithms are developed in the causal inference literature for heterogeneous treatment effects estimation (see e.g., Tian et al. 2014; Nie and Wager 2017; Kennedy 2020; Li, Wang, and Tu 2021b).

[2] Q-learning here is different from those Q-learning type algorithms in RL, due to different data structures and model setups. It relies on a backward induction algorithm to identify the optimal DTR in finite horizon settings with only a few treatment stages. In contrast, Q-learning type algorithms in RL usually rely on a Markov assumption to derive the optimal policy in infinite horizons.

time. At each time $t \geq 0$, the decision maker observes some features from the environment, summarized into a *state* vector $S_t \in \mathbb{S}$ where the state space $\mathbb{S}$ is assumed to be a subset of $\mathbb{R}^d$. The decision maker then selects an *action* A_t from the action space $\mathbb{A}$. The environment responds by providing the decision maker with an immediate *reward* $R_t \in \mathbb{R}$ and moving to the next state S_{t+1}. In this paper, we focus on the setting where $\mathbb{A}$ is discrete. Extensions to the continuous action space are discussed in Appendices A.1 and A.2 of the supplementary article. The state space $\mathbb{S}$ can be either continuous or discrete.

A policy defines the agent's way of behaving. A *history-dependent* policy π is a sequence of decision rules $\{\pi_t\}_{t\geq 0}$ where each π_t is a function that maps the observed data history to a probability distribution function on the action space at time t. When these decision rules are time-homogeneous (i.e., $\pi_1 = \pi_2 = \cdots = \pi_t = \cdots$) and depend on the past data history only through the current state vector, the resulting policy is referred to as a *stationary* policy. Following π, the discounted cumulative reward that the decision maker receives is referred to as the *value* function,

$$V^{\pi}(s) = \sum_{t=0}^{+\infty} \gamma^t \mathrm{E}^{\pi}(R_t | S_0 = s),$$

where the expectation E^{π} is taken by assuming that actions are assigned according to π and $0 \leq \gamma < 1$ is a discounted factor that balances the long-term and short-term rewards. The objective of policy optimization is to identify an optimal policy π^{opt} that maximizes the value, that is, $\pi^{\text{opt}} = \arg\max_{\pi} \mathrm{E}V^{\pi}(S_0)$.

We model the data generating process by a Markov decision process (MDP, Puterman 1994). Specifically, we impose the following Markov assumption (MA) and conditional mean independence assumption (CMIA).

(MA) There exists some function q such that for any $t \geq 0$, $\mathcal{S} \in \mathbb{S}$, we have

$$\Pr(S_{t+1} \in \mathcal{S} | \{S_j, A_j, R_j\}_{0\leq j\leq t}) = \int_{\mathcal{S}} q(s; A_t, S_t) ds.$$

(CMIA) There exists some reward function r such that for any $t \geq 0$, we have

$$\mathrm{E}(R_t | S_t, A_t, \{S_j, A_j, R_j\}_{0\leq j<t}) = r(A_t, S_t).$$

We make a few remarks. First, MA requires the future state to be conditional independent of the past data history given the current state-action pair. The function q corresponds to the Markov transition density function that characterizes the state transitions. This assumption is testable from the observed data (see e.g., Shi et al. 2020b). Second, under MA, CMIA is automatically satisfied when R_t is a deterministic function of S_t, A_t, and S_{t+1}. The latter assumption is commonly imposed in the literature (Ertefaie and Strawderman 2018; Luckett et al. 2020). CMIA is weaker than this assumption.

Second, these two assumptions lay the foundations of the existing state-of-the-art RL algorithms (e.g., DQN). Specifically, they guarantee the existence of an optimal stationary policy that is no worse than any history-dependent policies (see e.g., Puterman 1994). It allows us to restrict our attentions to the class of stationary policies. For any such policy π, we use $\pi(\bullet|s)$ to denote the probability mass function that the decision maker will follow to select actions given that the environment is in the state s.

The observed data consist of N trajectories. Specifically, let $\{(S_{i,t}, A_{i,t}, R_{i,t}, S_{i,t+1})\}_{0\leq t<T}$ be the data collected from the ith trajectory where T is the termination time. We assume these trajectories are independent copies of $\{(S_t, A_t, R_t)\}_{t\geq 0}$. Our objective is to learn π^{opt} based on this offline dataset.

2.2. Q-learning

For a given policy π, we define the state-action value function (better known as the Q-function) under π as

$$Q^{\pi}(a,s) = \sum_{t\geq 0} \gamma^t \mathrm{E}^{\pi}(R_t | A_0 = a, S_0 = s).$$

It represents the average cumulative reward that the decision maker will receive if they select the action a initially and follow π afterwards. In addition, notice that

$$\begin{aligned}
Q^{\pi}(a,s) &= \mathrm{E}(R_0 | A_0 = a, S_0 = s) \\
&\quad + \gamma \left\{ \sum_{t\geq 0} \gamma^t \mathrm{E}^{\pi}(R_{t+1} | A_0 = a, S_0 = s) \right\} \\
&= r(a,s) + \gamma \left\{ \sum_{t\geq 0} \gamma^t \mathrm{E}^{\pi}[\mathrm{E}^{\pi}(R_{t+1} | A_1, S_1, \right. \\
&\qquad \left. A_0 = a, S_0 = s) | A_0, S_0] \right\} \\
&= r(a,s) + \gamma \mathrm{E}^{\pi}\{Q^{\pi}(A_1, S_1) | A_0 = a, S_0 = s\} \\
&= r(a,s) + \gamma \int_{s'} \sum_{a'} \pi(a'|s') Q(a', s') q(s'; a, s) ds', \quad (1)
\end{aligned}$$

where the third equation follows from CMIA and the definition of Q^{π} and the last equation follows from MA. The above equation is referred to as the Bellman equation for Q^{π}.

Define optimal Q-function Q^{opt} as $Q^{\text{opt}}(a,s) = \max_{\pi} Q^{\pi}(a,s)$ for any state-action pair (a,s). Under MA and CMIA, it can be shown that π^{opt} satisfies

$$\pi^{\text{opt}}(a|s) = \mathbb{I}\left\{a = \arg\max_{a'} Q^{\text{opt}}(a', s)\right\}, \quad \forall a, s, \qquad (2)$$

where $\mathbb{I}\{\cdot\}$ denotes the indicator function. In addition, we have $Q^{\text{opt}} = Q^{\pi^{\text{opt}}}$. Similar to (1), one can show that Q^{opt} satisfies the following Bellman optimality equation,

$$Q^{\text{opt}}(a,s) = r(a,s) + \gamma \int_{s'} \max_{a'} Q^{\text{opt}}(s', a') q(s'; a, s) ds', \qquad (3)$$

or equivalently,

$$Q^{\text{opt}}(A_t, S_t) = \mathrm{E}\left\{R_t + \gamma \max_{a} Q^{\text{opt}}(a, S_{t+1}) | A_t, S_t\right\}. \qquad (4)$$

Equations (2) and (4) form the basis for all Q-learning type algorithms. Specifically, these algorithms first estimate the optimal Q-function by solving (4) and then derive the estimated optimal policy based on (2). Take the fitted Q-iteration algorithm as an example. It iteratively updates the optimal Q-function using supervised learning. At each iteration, the input includes (A_t, S_t)

that serves as the "predictors" and $R_t + \gamma \max_a \widetilde{Q}(a, S_{t+1})$ that serves as the "response" where $\widetilde{Q}$ denotes the current estimate of the optimal Q-function.

Finally, we introduce the contrast function. For a given π, define the contrast function associated with π as $\tau^{\pi}(a,s) = Q^{\pi}(a,s) - Q^{\pi}(a_0,s)$[3] for some $a_0 \in \mathbb{A}$. In practice, the control arm a_0 could be set to the action that occurs the most in the data. This is because the baseline Q-function $Q^{\pi}(a_0,s)$ needs to be accurately estimated in order to consistently estimate the contrast function. Hence, it is natural to consider the most frequently selected arm, which has the largest number of observations to learn the baseline Q-function. Let $\tau^{\text{opt}}(a,s) = \tau^{\pi^{\text{opt}}}(a,s)$ be the optimal contrast function. Similar to (2), we obtain that

$$\pi^{\text{opt}}(a|s) = \mathbb{I}\{a = \arg\max_{a'} \tau^{\text{opt}}(a',s)\},$$

for any a and s. Consequently, to estimate the optimal policy, it suffices to estimate τ^{opt}. This observation motivates the proposed advantage learning method.

3. Q- versus A-learning

This section is organized as follows. We first introduce the minimax-optimal statistical convergence rate in supervised learning, which serves as an evaluation metric to compare various supervised learning algorithms. We next demonstrate the advantage of A-learning over Q-learning by comparing the worst-case convergence rates of the estimated optimal contrast and Q-functions. Finally, we discuss the challenge of developing statistically efficient A-learning algorithms.

3.1. Minimax Optimal Statistical Convergence Rate

Consider a supervised learning setup where we have given iid random vectors $\{(X_i, Y_i) : 1 \le i \le n\}$. Our objective is to predict the value of the response Y from the value of the feature $X \in \mathbb{S}$. The aim is to construct a best predictor to approximate the conditional mean function $m(X) = \mathrm{E}(Y|X)$. For any such predictor $\widehat{m}$, its prediction accuracy is measured by the root mean square error,

$$\sqrt{\mathrm{E}|\widehat{m}(X) - m(X)|^2}. \tag{5}$$

Suppose m belongs to the class of p-smooth (also known as Hölder smooth with exponent p) functions. When p is an integer, this condition essentially requires m to have bounded derivatives up to the pth order. Formally speaking, for a J-tuple $\alpha = (\alpha_1, \ldots, \alpha_J)^\top$ of nonnegative integers and a given function h on $\mathbb{S}$, let D^α denote the differential operator:

$$D^\alpha h(s) = \frac{\partial^{\|\alpha\|_1} h(s)}{\partial s_1^{\alpha_1} \cdots \partial s_J^{\alpha_J}}.$$

Here, s_j denotes the jth element of s. For any $p > 0$, let $\lfloor p \rfloor$ denote the largest integer that is smaller than p. The class of p-smooth functions is defined as follows:

$$\Lambda(p,c) = \left\{ h : \sup_{\|\alpha\|_1 \le \lfloor p \rfloor} \sup_{s \in \mathbb{S}} |D^\alpha h(s)| \right.$$
$$\left. \le c, \sup_{\|\alpha\|_1 = \lfloor p \rfloor} \sup_{\substack{s_1, s_2 \in \mathbb{S} \\ s_1 \ne s_2}} \frac{|D^\alpha h(s_1) - D^\alpha h(s_2)|}{\|s_1 - s_2\|_2^{p - \lfloor p \rfloor}} \le c \right\},$$

for some constant $c > 0$. When $0 < p \le 1$, we have $\lfloor p \rfloor = 0$. It is equivalent to require h to satisfy $\sup_{s_1,s_2} |h(s_1) - h(s_2)| / \|s_1 - s_2\|_2^p \le c$. The notion of p-smoothness is thus reduced to the Hölder continuity.

Stone (1982) showed that the optimal minimax rate of convergence for $\widehat{m}$ is given by

$$n^{-p/(2p+d)}, \tag{6}$$

where d denotes the dimension of $\mathbb{S}$. In other words, for any data-dependent predictor $\widehat{m}$, there exists some p-smooth function m such that (5) decays at a rate of (6). This rate cannot be improved unless imposing certain parametric model assumptions on m. Notice that (6) increases with the smoothness parameter p. In other words, the smoother the underlying regression function, the faster worst-case rate of convergence a supervised learner could achieve.

Finally, we remark that we focus on the class of Hölder smooth functions throughout this paper. Alternatively, one may consider the Sobolev space. Discussion of Sobolev and Hölder spaces can be found in Giné and Nickl (2021).

3.2. Modelling Contrast or Q-function?

We assume the state space $\mathbb{S}$ is continuous and both the transition function $q(s'; a, \bullet)$ and reward function $r(a, \bullet)$ belong to the class of p-smooth functions on $\mathbb{S}$ for some $p > 0$. The p-smoothness assumption is likely to hold in many mobile health applications and we delegate the related discussions in Appendix A.3., supplementary materials. Under this condition, the optimal Q-function is p-smooth as well (see sec. 4, Fan et al. 2020). Fan et al. (2020) proved that the Q-function computed by DQN achieves a rate of $(NT)^{-p/(2p+d)}$ up to some logarithmic factors. As they commented, this rate achieves the minimax-optimal statistical convergence rate in (6) within the class of p-smooth functions and cannot be further improved.

Since the optimal contrast function corresponds to the difference between two optimal Q-functions, τ^{opt} is at least at smooth as Q^{opt}. On the other hand, there are cases where τ^{opt} is strictly "smoother" than Q^{opt}, leading to a possibly faster worst-case rate of convergence according to the minimax-optimal rate formula. We consider two examples to elaborate.

Example 1 (Independent Transitions). Consider the setting where the state transitions are independent, that is, $q(s'; a, s) = q(s')$ is independent of (a, s). Then $Q^{\text{opt}}(a,s) = r(a,s) + C$ for some constant $C > 0$ that is independent of s and a. Suppose the reward function has the following decomposition

$$r(a,s) = r^*(a,s) + r_0(s),$$

for some p-smooth baseline reward function r_0 and p^*-smooth function r^* with $p^* > p$. It follows that $Q^{\text{opt}}(a, \bullet)$ is p-smooth whereas $\tau^{\text{opt}}(a, \bullet) = r^*(a, \bullet) - r^*(a_0, \bullet)$ is p^*-smooth.

[3] Here, we define the contrast function as the difference between two Q-functions. Alternatively, one may define τ^π to be the advantage function, that is, the difference between Q^π and V^π.

Example 2 (*Dependent Transitions*). Suppose q has the following decomposition

$$q(s'; a, s) = q^*(s'; a, s) + q_0(s'; s), \tag{7}$$

where $q^*(s'; \bullet, a)$ has derivatives up to the p^*th order whereas $q_0(s'; \bullet)$ has derivatives up to the pth order with $p < p^*$. By changing the order of integration and differentiation with respect to s, we can show that the second term on the right-hand-side (RHS) of (3) is p-smooth. Suppose $r(a, \bullet)$ has derivatives of all orders. It follows from (3) that Q^{opt} is p-smooth.

On the contrary, by (3) and (7), we have that

$$\tau^{\text{opt}}(a, s) = r(a, s) - r(a_0, s) \\ + \gamma \int_{s'} \max_{a'} Q^{\text{opt}}(a', s') q^*(s'; a, s) ds'.$$

Using similar arguments, we can show that the last term on the RHS is p^*-smooth. This in turn implies that τ^{opt} is p^*-smooth as well.

To conclude this section, we remark that the minimax rate for the contrast function has been recently established in single-stage decision making (Kennedy, Balakrishnan, and Wasserman 2022). In infinite horizon settings with tabular models, several papers have investigated the minimax-optimality of the Q-learning estimator (see e.g., Wainwright 2019; Li et al. 2020, 2021a). In settings with continuous state space, a recent proposal of Chen and Qi (2022) derived a minimax lower bound for the Q-function estimator under a fixed target policy and found that the rate matches those for nonparametric regression (Stone 1982). We expect that similar arguments can be applied to formally obtain the minimax lower bounds for the estimated optimal Q- or contrast function.

3.3. The Challenge

So far we have shown that the worse-case convergence rate of the estimated optimal contrast function is faster than that of the estimated optimal Q-function. However, it remains challenging to devise an advantage learning algorithm that achieves such a rate of convergence. To elaborate, let us revisit the Bellman optimality equation in (4). By the definition of the optimal contrast function, it follows that

$$\tau^{\text{opt}}(A_t, S_t) = \text{E}\left\{ R_t + \gamma \max_a \tau^{\text{opt}}(a, S_{t+1}) \right. \\ \left. + \gamma Q^{\text{opt}}(a_0, S_{t+1}) \middle| A_t, S_t \right\} - Q^{\text{opt}}(a_0, S_t). \tag{8}$$

The presence of the nuisance function $Q^{\text{opt}}(a_0, \bullet)$ in the above equation poses a serious challenge to efficient estimation of τ^{opt}. A simple solution is to apply Q-learning type algorithms to learn the nuisance function, plug in this estimator in (8) and update τ^{opt} using for example, fitted Q-iteration. However, such an approach would yield a sub-optimal solution. This is because the estimation error of the initial Q-estimator would directly affect that of the estimated contrast function. As a result, the estimated contrast would have the same convergence rate as the Q-estimator.

4. Statistically Efficient A-Learning

We first present the motivation of our algorithm. We next formally introduce our proposal.

4.1. A Thought Experiment

To illustrate the idea, in this section, let us consider a simplified model where the discounted factor $\gamma = 0$ and the transitions are independent (see Example 1). In that case, we are interested in learning an optimal myopic policy the maximizes the short-term reward on average, which is essentially a single-stage decision making problem. By definition, the Q-function Q^{π} and the contrast τ^{π} are independent of the policy π. Equation (8) can be rewritten as

$$\tau(A_t, S_t) = \text{E}(R_t | A_t, S_t) - Q(a_0, S_t) \tag{9}$$

where $Q(a_0, s) = \text{E}(R_t | A_t = a_0, S_t = s)$.

A-learning algorithms developed in the statistics literature can be employed to learn the contract function in this setting. They are motivated by the following identity,

$$\sum_{a \neq a_0} \text{E}[\{\mathbb{I}(A_t = a) - \Pr(A_t = a | S_t)\} \\ \{R_t - \tau(A_t, S_t) - Q(a_0, S_t)\} | S_t] = 0. \tag{10}$$

Unlike Equation (9), the above equation is doubly-robust. It holds when either the propensity score $\Pr(A_t = \bullet | S_t)$ or the Q-function $Q(a_0, \bullet)$ is correctly specified. This motives the following two-step procedure. In the first step, we first estimate the propensity score and the Q-function from the observed data. In the second step, we plug in these estimates in (10) to estimate the contrast function. Such a two-step method guarantees the estimated contrast to be robust to the potential model misspecification of the Q-function.

When linear sieves are used to approximate τ, that is, $\tau(a, s) = \phi(a, s)^{\top} \beta_0$ for some basis function ϕ, an estimating equation for β_0 can be constructed based on (10). A Dantzig selector-type regularization can be applied when the number of basis functions is large Shi et al. (2018a). To employ more flexible machine learning methods, we can consider the following least-square objective function,

$$\sum_{\substack{i,t \\ a \neq a_0}} \Bigg[\underbrace{\left\{ \frac{\mathbb{I}(A_{i,t} = a)}{\Pr(A_{i,t} = a | S_{i,t})} - \frac{\mathbb{I}(A_{i,t} = a_0)}{\Pr(A_{i,t} = a_0 | S_t)} \right\} \{R_{i,t} - Q(A_{i,t}, S_{i,t})\} + Q(a, S_{i,t}) - Q(a_0, S_{i,t})}_{\psi(S_{i,t}, A_{i,t}, R_{i,t}, a)} \\ - \tau(a, S_{i,t}) \Bigg]^2.$$

Here, $\psi(S_{i,t}, A_{i,t}, R_{i,t}, a)$ serves as a pseudo outcome for $\tau(a, S_{i,t})$. It is derived based on augmented inverse probability weighting (AIPW, see e.g., Bang and Robins 2005). One can similarly show that $\mathrm{E}\{\psi(S_{i,t}, A_{i,t}, R_{i,t}, a)|S_{i,t}\}$ is unbiased to $\tau(a, S_{i,t})$ when either the propensity score or the Q-function is correctly specified. A by-product of the doubly-robustness property is that when both nuisance functions are estimated from the data, the bias of the pseudo outcome will converge at a faster rate than these estimated nuisance functions. This in turn allows the resulting estimated contrast to converge at a faster rate than the Q-function. See for example, Section 5 for details.

Although the above solution is developed in single-stage decision making, the same principle can be applied to general sequential decision making problems in infinite horizons, as we detail in the next section.

4.2. The Complete Algorithm

Our proposal involves two key components. First, we apply existing off-policy evaluation methods to construct pseudo outcomes for the optimal contrast function. This effectively reduces the bias of the initial Q-estimators, as we show in Theorem 1 that the bias of our pseudo outcomes decays at a much faster rate than initial Q-estimators. It in turn ensures that the estimated contrast is robust to the model misspecification of the Q-function, improving its rate of convergence. Second, we learn τ^{opt} by directly minimizing the least square loss between the pseudo outcomes and the estimated contrast. This allows us to borrow the strength of supervised learning to improve the statistical efficiency for RL. We call this set of method SEAL—short for *statistically efficient advantage learning*.

Our proposal consists of five steps, including data splitting, policy optimization, estimation of the density ratio, construction of pseudo outcomes, and supervised learning. We next discuss each step in detail.

4.2.1. Step 1. Data Splitting

We randomly divide the indices of all trajectories $\{1, \ldots, N\}$ into $\mathbb{L}$ subsets $\cup_{\ell=1}^{\mathbb{L}} \mathcal{I}_\ell$ with equal size, for some fixed integer $\mathbb{L} > 0$. Let $\mathcal{I}_\ell^c$ be the complement of $\mathcal{I}_\ell$. Data splitting allows us to use one part of the data ($\mathcal{I}_\ell^c$) to train RL models and the remaining part ($\mathcal{I}_\ell$) to construct the pseudo outcomes. We could aggregate the estimate over different ℓ to get full efficiency. This allows the bias of the constructed pseudo outcomes to decay to zero under minimal conditions on the estimated RL models. We remark that data splitting has been commonly used in the statistics and machine learning literature (see e.g., Chernozhukov et al. 2018; Romano and DiCiccio 2019; Kallus and Uehara 2019).

4.2.2. Step 2. Policy Optimization

For $\ell = 1, \ldots, \mathbb{L}$, we apply existing state-of-the-art Q-learning type algorithms to the data subset in $\mathcal{I}_\ell^c$ to compute an initial Q-estimator $\widehat{Q}^{(\ell)}$ for Q^{opt}. Several algorithms can be applied here, as we elaborate below.

Example 3 (DQN). The deep Q-network algorithm is a Q-learning type method that uses a neural network Q-function approximator and several tricks to mitigate instability. It was developed in online settings and shown to yield superior performance to previously known methods for playing Atari 2600 games. To handle offline data, at each time step, we sample a minibatch of transitions $\{(S_{i,t}, A_{i,t}, R_{i,t}, S_{i,t+1})\}_{(i,t)\in\mathcal{M}}$ and update the parameter θ of the Q-network by the gradient of

$$\sum_{(i,t)\in\mathcal{M}} \{R_{i,t} + \max_a \gamma Q_{\theta^*}(a, S_{i,t+1}) - Q_\theta(A_{i,t}, S_{i,t})\}^2, \quad (11)$$

where Q_{θ^*} is the target network whose parameter θ^* is updated every T_{target} steps by letting $\theta^* = \theta$. In Mnih et al. (2015), T_{target} is set to 10,000. As T_{target} grows to infinity, performing T_{target} stochastic gradient steps is equivalent to solve

$$\arg\min_\theta \sum_{i=1}^{N} \sum_{t=0}^{T-1} \{R_{i,t} + \max_a \gamma Q_{\theta^*}(a, S_{i,t+1}) - Q_\theta(A_{i,t}, S_{i,t})\}^2.$$

In that sense, DQN shares similar spirits with the fitted Q-iteration algorithm (Fan et al. 2020).

Example 4 (Double DQN). The double DQN algorithm is very similar to DQN. It is developed to alleviate the overestimation bias of the learned Q-function. DQN is likely to overestimate the Q-function under certain conditions, due to the biased resulting from the maximization step $\max_a Q_{\theta^*}(a, S_{i,t+1})$ in (11). See for example, Sutton and Barto (2018) for a detailed explanation of the maximization bias. To reduce this bias, it replaces the target $R_{i,t} + \max_a \gamma Q_{\theta^*}(a, S_{i,t+1})$ by

$$R_{i,t} + \gamma Q_{\theta^*}(\arg\max_a Q_\theta(a, S_{i,t+1}), S_{i,t+1}).$$

In other words, it decomposes the maximization operation into action selection and state-action value evaluation, uses the Q-network for action selection and the target network for value evaluation. It was shown in Van Hasselt, Guez, and Silver (2016) that such a trick leads to much better performance on several games empirically.

Example 5 (Quantile DQN). The quantile DQN algorithm can be viewed as a distributional version of DQN with quantile regression. Instead of directly learning Q^{opt}, the expected return given the initial state-action pair, it learns quantiles of the return based on the distributional analogue of Bellman's optimality equation (4) and averages the learned quantiles to estimate Q^{opt}. Please refer to Dabney et al. (2018) for details.

Given the Q-estimator $\widehat{Q}^{(\ell)}$, we denote the derived optimal policy (see Equation (2)) by $\widehat{\pi}^{(\ell)}$, for $\ell = 1, \ldots, \mathbb{L}$.

4.2.3. Step 3. Estimation of the Density Ratio

The purpose of this step is to learn a density ratio estimator based on each data subset. These estimators are further employed in the subsequent step to construct the pseudo outcomes for the optimal contrast function.

We first define the density ratio. For a given policy π, let $p_t^\pi(\bullet, \bullet|a, s)$ denote the conditional probability density function of (A_t, S_t) given the initial state-action pair (a, s) assuming that the decision maker follows π at time $1, 2, \ldots, t$. We define the γ-discounted average visitation density as follows,

$$p_\gamma^\pi(\bullet, \bullet|a, s) = (1-\gamma) \sum_{t\geq 1} \gamma^{t-1} p_t^\pi(\bullet, \bullet|a, s).$$

Let $p_\infty(\bullet,\bullet)$ denote the density function of the limiting distribution of the stochastic process $\{(A_t,S_t)\}_{t\ge 0}$. We define the density ratio as

$$\omega^\pi(a',s'|a,s)=\frac{p_\gamma^\pi(a',s'|a,s)}{p_\infty(a',s')},$$

for any s,a,s',a'. Such a density ratio plays an important role in breaking the curse of horizon in off-policy evaluation[4] (Liu et al. 2018).

In this step, we learn the density ratio $\omega^{\widehat{\pi}^{(\ell)}}$ based on each data subset in $\mathcal{I}_\ell^c$, for $\ell=1,\ldots,\mathbb{L}$, where $\widehat{\pi}^{(\ell)}$ is the initial optimal policy computed in Step 2. Several methods can be used here, for example, Liu et al. (2018), Uehara, Huang, and Jiang (2019), and Kallus and Uehara (2019). In our implementation, we adopt the proposal in Liu et al. (2018) to construct a minimax loss function to estimate $\omega^{\widehat{\pi}^{(\ell)}}$. We use $\widehat{\omega}^{(\ell)}$ to denote the corresponding estimator. Additional details are given in Appendix C, supplementary materials to save space.

4.2.4. Step 4. Construction of Pseudo Outcomes

For $\ell=1,\ldots,\mathbb{L}$, consider a pair of indices (i,t) with $i\in\mathcal{I}_\ell, 0\le t<T$. In this step, we focus on constructing a pseudo outcome $\widetilde{Q}_{i,t,a}$ for $Q^{\rm opt}(a,S_{i,t})$ for any $a\in\mathbb{A}$, based on the Q- and density ratio estimators computed in Steps 2 and 3. The corresponding pseudo outcome for $\tau^{\rm opt}(a,S_{i,t})$ is given by $\widetilde{\tau}_{i,t,a}=\widetilde{Q}_{i,t,a}-\widetilde{Q}_{i,t,a_0}$.

To motivate our method, notice that by the Bellman equation,

$$Q^{\rm opt}(a,S_{i,t})=r(a,S_{i,t})+\gamma{\rm E}\{V^{\pi^{\rm opt}}(S_{i,t+1})|A_{i,t}=a,S_{i,t}\},$$

it suffices to construct pseudo outcomes for $r(a,S_{i,t})$ and ${\rm E}\{V^{\pi^{\rm opt}}(S_{i,t+1})|A_{i,t}=a,S_{i,t}\}$. Pseudo outcomes for $r(a,S_{i,t})$ can be derived based on augmented inverse propensity-score weighting, as in Section 4.1,

$$\widetilde{r}_{i,t,a}=\widehat{r}^{(\ell)}(a,S_{i,t})+\frac{\mathbb{I}(A_{i,t}=a)}{\Pr(A_{i,t}=a|S_{i,t})}\{R_{i,t}-\widehat{r}^{(\ell)}(a,S_{i,t})\},$$

where $\widehat{r}^{(\ell)}$ denotes some estimator for the reward function r computed using the data subset in $\mathcal{I}_\ell^c$. As we have commented, the use of AIPW ensures the unbiasedness of the pseudo outcome, regardless of whether $\widehat{r}^{(\ell)}$ is consistent to r or not.

As for ${\rm E}\{V^{\pi^{\rm opt}}(a',S_{i,t+1})|A_{i,t}=a,S_{i,t}\}$, since $\pi^{\rm opt}$ is unknown, we consider approximating it by

$$\nu^{(\ell)}(a,S_{i,t})={\rm E}\{V^{\widehat{\pi}^{(\ell)}}(S_{i,t+1})|S_{i,t},A_{i,t}=a\},\qquad(12)$$

using the estimated optimal policy $\widehat{\pi}^{(\ell)}$.

Suppose for now, the Markov transition density function q is known. Then $\nu^{(\ell)}(S_{i,t},a)$ can be estimated using the existing policy evaluation methods. Here, we consider the doubly reinforcement learning method proposed by Kallus and Uehara (2019),

$$\int_{s'}\max_{a'}\widehat{Q}^{(\ell)}(a',s')q(s';a,S_{i,t})ds'+\frac{1}{1-\gamma}\eta_{i,t,a},\qquad(13)$$

where $\eta_{i,t,a}$ is an augmentation term, defined as

$$\frac{1}{|\mathcal{I}_\ell|T-1}\sum_{\substack{i'\in\mathcal{I}_\ell\\(i',t')\neq(i,t)}}\widehat{\omega}^{(\ell)}(A_{i',t'},S_{i',t'}|a,S_{i,t})$$
$$\{R_{i',t'}+\gamma\max_{a'}\widehat{Q}^{(\ell)}(a',S_{i',t'+1})-\widehat{Q}^{(\ell)}(A_{i',t'},S_{i',t'})\}.$$

The second term $R_{i',t'}+\gamma\max_{a'}\widehat{Q}^{(\ell)}(a',S_{i',t'+1})-\widehat{Q}^{(\ell)}(A_{i',t'},S_{i',t'})$ denotes the Bellman residual constructed based on the initial Q-estimator. When the initial Q-estimator is consistent, it follows from the Bellman optimality equation that the mean of $\eta_{i,t,a}$ is asymptotically zero. The purpose of adding $\eta_{i,t,a}$ in (13) is to offer additional protection against potential model misspecification of the initial Q-estimator. Specifically, it ensures that (13) is unbiased to $\nu^{(\ell)}(S_{i,t},a)$ when either $\widehat{Q}^{(\ell)}$ or $\widehat{\omega}^{(\ell)}$ is consistent (see e.g., Kallus and Uehara 2019). In addition, when the estimated ratio is consistent, it allows the bias of (13) to decay to zero at a rate faster than $\widehat{Q}^{(\ell)}$. See Theorem 1 for a formal statement.

However, the pseudo outcome outlined in (13) suffers from two major limitations. The first one is that the transition density q is in general unknown in practice. The second one is that the calculation of $\eta_{i,t,a}$ requires $O(NT)$ number of flops, which is computationally intensive to implement on large datasets.

Let $\widehat{\nu}^{(\ell)}$ be some estimator for $\nu^{(\ell)}$ (see Equation (12)) computed using $\{O_{i,t}\}_{0\le t<T_i,i\in\mathcal{I}_\ell^c}$. To address the first limitation, we again use augmented inverse probability weighting and replace the first term in (13) by

$$\widetilde{\nu}_{i,t,a}=\widehat{\nu}^{(\ell)}(a,S_{i,t})+\frac{\mathbb{I}(A_{i,t}=a)}{\Pr(A_{i,t}=a|S_{i,t})}$$
$$\{\max_{a'}\widehat{Q}^{(\ell)}(a',S_{i,t+1})-\widehat{\nu}^{(\ell)}(a,S_{i,t})\}.$$

Similar to $\widetilde{r}_{i,t,a}$, one can easily verify that $\widetilde{\nu}_{i,t,a}$ is unbiased to $\nu(a,S_{i,t})$ regardless of whether $\widehat{\nu}^{(\ell)}$ is consistent or not. To address the second limitation, we randomly sample a minibatch $\mathcal{M}_{i,t}$ from the set $\{(i',t'):(i',t')\neq(i,t),i'\in\mathcal{I}_\ell,0\le t'<T\}$ to approximate $\eta_{i,t,a}$ by $\widetilde{\eta}_{i,t,a}$, constructed based on the observations in $\mathcal{M}_{i,t}$ only. Specifically, we define $\widetilde{\eta}_{i,t,a}$ by

$$\frac{1}{|\mathcal{M}_{i,t}|}\sum_{(i',t')\in\mathcal{M}_{i,t}}\widehat{\omega}^{(\ell)}(A_{i',t'},S_{i',t'}|a,S_{i,t})$$
$$\{R_{i',t'}+\gamma\max_{a'}\widehat{Q}^{(\ell)}(a',S_{i',t'+1})-\widehat{Q}^{(\ell)}(A_{i',t'},S_{i',t'})\},$$

When $|\bullet|$ denotes the cardinality of a set. When $|\mathcal{M}_{i,t}|$ is much smaller than NT, it largely facilitates the computation.

Combining both parts yields the following,

$$\begin{aligned}&\widetilde{r}_{i,t,a}+\gamma\widetilde{\nu}_{i,t,a}+\gamma(1-\gamma)^{-1}\widetilde{\eta}_{i,t,a}\\&\quad=\widehat{r}^{(\ell)}(a,S_{i,t})+\widehat{\nu}^{(\ell)}(a,S_{i,t})\\&\qquad+\frac{\mathbb{I}(A_{i,t}=a)}{\Pr(A_{i,t}=a|S_{i,t})}\{R_{i,t}+\max_{a'}\widehat{Q}^{(\ell)}(a',S_{i,t+1})\\&\qquad-\widehat{r}^{(\ell)}(a,S_{i,t})-\widehat{\nu}^{(\ell)}(a,S_{i,t})\}+\frac{\gamma}{1-\gamma}\widetilde{\eta}_{i,t,a}.\end{aligned}$$

Notice that $r(a,S_{i,t})+\gamma\nu^{(\ell)}(a,S_{i,t})=Q^{\widehat{\pi}^{(\ell)}}(a,S_{i,t})$ can be estimated by $\widehat{Q}^{(\ell)}(a,S_{i,t})$. Putting all the pieces together, we

[4] Notice that our defined density ratio is slightly different from those in the existing OPE literature in that it involves an initial state-action pair.

obtain the following pseudo outcome for $\widetilde{Q}_{i,t,a}$, defined by

$$\widehat{Q}^{(\ell)}(a, S_{i,t}) + \frac{\mathbb{I}(A_{i,t} = a)}{\Pr(A_{i,t} = a|S_{i,t})}\{R_{i,t} + \gamma \max_{a'} \widehat{Q}^{(\ell)}(a', S_{i,t+1}) \\ -\widehat{Q}^{(\ell)}(A_{i,t}, S_{i,t})\} + \frac{\gamma}{(1-\gamma)}\widetilde{\eta}_{i,t,a}.$$

As we have discussed, the pseudo outcome for the optimal contrast is obtained by $\widetilde{\tau}_{i,t,a} = \widetilde{Q}_{i,t,a} - \widetilde{Q}_{i,t,a_0}$.

We again make some remarks. First, we employ cross-fitting to construct $\widetilde{\tau}_{i,t,a}$. That is, $\widehat{Q}^{(\ell)}$ and $\widehat{\omega}^{(\ell)}$ are computed by observations that are independent of $(S_{i,t}, A_{i,t}, R_{i,t}, S_{i,t+1})$. This helps avoid overfitting which can easily result from the estimation of the Q-function and density ratio. Second, to simplify the presentation, we assume the propensity score is known. In practice, it can be estimated from the observed data and our theoretical results will be the same when the estimated propensity score satisfies certain rate of convergence.

4.2.5. Supervised Learning

In the final step, we factorize the contrast function τ^{opt} by some models $\tau \in \mathcal{T}$ and estimate the model parameter by minimizing the following objective function,

$$\widehat{\tau} = \arg\min_{\tau \in \mathcal{T}} \sum_{i=1}^{N} \sum_{t=0}^{T-1} \sum_{a \neq a_0} \{\widetilde{\tau}_{i,t,a} - \tau(a, S_{i,t})\}^2. \tag{14}$$

The corresponding estimated optimal policy is given by $\mathbb{I}\{a = \arg\max_a^* \widehat{\tau}(a^*, s)\}$ for any a and s.

To solve (14), it is equivalent to solve

$$\arg\min_{\tau \in \mathcal{T}_a} \sum_{i=1}^{N} \sum_{t=0}^{T-1} \{\widetilde{\tau}_{i,t,a} - \tau(a, S_{i,t})\}^2, \tag{15}$$

for each $a \neq a_0$. Many methods are available to solve (15), as it is essentially a nonparametric regression problem. In our implementation, we set $\mathcal{T}_a$ to the class of deep neural networks (DNNs), so as to capture the complex dependence between the reward and the state-action pair. The input of the network is a d-dimensional vector, corresponding to the state (colored in blue in Figure 1). The hidden units (colored in green) are grouped in a sequence of L_a layers. Each unit in the hidden layer is determined as a nonlinear transformation of a linear combination of the nodes from the previous layer. We use W_a to denote the total number of parameters. These parameters are updated by the Adam algorithm (Kingma and Ba 2015).

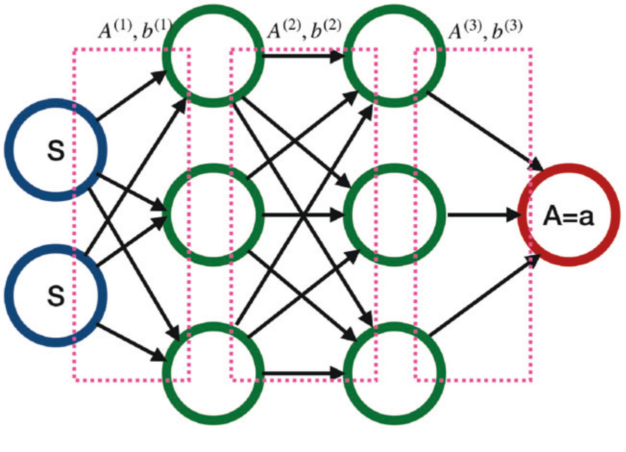

Figure 1. Illustration of a two-layer, fully connected DNN. The state is two-dimensional.

5. Theoretical Findings

We first summarize our theoretical findings. In Theorem 1, we provide a finite sample bias analysis of the pseudo outcome, proving its bias decays at a faster rate than the initial Q-estimator. In Theorem 2, we show our estimator for the optimal contrast achieves a faster rate of the convergence than the Q-estimator. In Theorem 3, we show the resulting optimal policy achieves a larger value than those computed by Q-learning type algorithms. Finally, we discuss a potential limitation of the proposed method. All the error bounds derived in this paper converge to zero when either N or T diverges to infinity. As commented in the introduction, this ensure our method is valid when applied to a wide range of real problems.

5.1. Finite Sample Bias Analysis

In this section, we focus on deriving an error bound on the bias $\mathrm{E}(\widetilde{Q}_{i,t,a}|S_{i,t}) - Q^{\text{opt}}(S_{i,t}, a)$ as a function of the total number of observations NT. We introduce the following conditions.

(A1) The state space $\mathbb{S}$ is compact. There exists some constant $\alpha > 0$ such that

$$\lambda \left\{ s \in \mathbb{S} : \max_a Q^{\text{opt}}(a, s) - \max_{a' \in \mathbb{A} - \arg\max_a Q^{\text{opt}}(a,s)} Q^{\text{opt}}(a', s) \leq \varepsilon \right\} \\ = O(\varepsilon^{\alpha}), \tag{16}$$

where λ denotes the Lebesgue measure and the big-O term in (16) is uniform in $0 < \varepsilon < \delta$ for some sufficiently small $\delta > 0$. By convention, $\max_{a' \in \mathbb{A} - \arg\max_a Q^{\text{opt}}(a,s)} Q^{\text{opt}}(a', s) = -\infty$ if the set $\mathbb{A} - \arg\max_a Q^{\text{opt}}(a, s)$ is empty.

(A2) $Q^{\text{opt}}(a, \cdot)$ is p-smooth and $\tau^{\text{opt}}(a, \cdot)$ is p^*-smooth for some $p^* < p$ and any a.

(A3) There exists some constant $0 < c_0 \leq 1$ such that the followings hold for any a and ℓ, with probability approaching 1,

$$\mathrm{E}_{(a,s)\sim p_\infty} |\widehat{Q}^{(\ell)}(a, s) - Q^{\text{opt}}(a, s)|^2 = O\{(NT)^{-2p/(2p+d)}\},$$
$$\mathrm{E}_{(a,s)\sim p_\infty, (a',s')\sim p_\infty} |\widehat{\omega}^{(\ell)}(a', s'|a, s) - \omega^{\widehat{\pi}^{(\ell)}}(a', s'|a, s)|^2 \\ = O\{(NT)^{-c_0}\}.$$

(A4) The process $\{(S_t, A_t, R_t)\}_{t \geq 0}$ is stationary and exponentially β-mixing (see e.g., Bradley 2005, for detailed definitions).

(A5) The probability density function p_∞ is uniformly bounded away from zero.

In (A1), we require the state space to be continuous. When it is discrete, we can replace the Lebesgue measure with the counting measure. Our theories are equally applicable. we refer to the quantity $\max_a Q^{\text{opt}}(a, s) - \max_{a' \in \mathcal{A} - \arg\max_a Q^{\text{opt}}(a,s)} Q^{\text{opt}}(a', s)$ as the “margin” of the optimal Q-function. It measures the difference in value between π^{opt} and the policy that assigns the best suboptimal treatment(s) at the first decision point and follows π^{opt} subsequently. Such a margin-type condition is commonly used to bound the excess misclassification error (Tsybakov 2004; Audibert and Tsybakov 2007) and the regret of estimated optimal treatment regime (Qian and Murphy 2011; Luedtke and van der Laan 2016; Shi, Lu, and Song 2020a). Here, we impose Condition (A1) to bound the difference between $Q^{\widehat{\pi}^{(\ell)}}(S_{i,t}, a)$ and $Q^{\mathrm{opt}}(S_{i,t}, a)$. This condition is mild. To elaborate, we consider a simple scenario where $\mathbb{A} = \{0, 1\}$ and $a_0 = 0$. It follows that the margin equals $|\tau^{\mathrm{opt}}(1, s)|$ if $\tau^{\mathrm{opt}}(1, s)$ is nonzero and $+\infty$ otherwise. (16) is thus equivalent to the following,

$$\lambda\{s \in \mathbb{S} : 0 < |\tau^{\mathrm{opt}}(1, s)| \le \varepsilon\} = O(\varepsilon^{\alpha}) \tag{17}$$

The above condition can be satisfied in a wide range of settings. We consider three examples to illustrate.

Example 6. Suppose $\tau^{\mathrm{opt}}(1, s) = 0$ for any s. Then (17) is automatically satisfied. In this example, the two actions have the same effects. Any policy would achieve the same value.

Example 7. Suppose $\inf_s |\tau^{\mathrm{opt}}(1, s)| > 0$. Then (17) is automatically satisfied for any sufficiently small $\varepsilon > 0$. When the optimal contrast function is continuous, it requires $\tau^{\mathrm{opt}}(1, s)$ to be always positive or negative as a function of s. As such, the optimal policy is nondynamic and will assign the same action at each time.

Example 8. Consider the case where the state is one-dimensional. Suppose

$$\tau^{\mathrm{opt}}(1, s) = \begin{cases} s^{1/\alpha}, & \text{if } s > 0; \\ 0, & \text{otherwise}, \end{cases}$$

we have $\lambda\{s \in \mathbb{S} : 0 < |\tau^{\mathrm{opt}}(1, s)| \le \varepsilon\} = \lambda\{s \in \mathbb{S} : 0 < s \le \varepsilon^{\alpha}\} = \varepsilon^{\alpha}$. (17) is thus satisfied.

In (A2), we assume the optimal contrast function is strictly "smoother" than the optimal Q-function. As we have discussed, this assumption holds under several cases. See Examples 1 and 2 in Section 3.2 for details.

In the first part of (A3), we assume the squared prediction loss of the estimated Q-function achieves a rate of $(NT)^{-2p/(2p+d)}$. As we have commented, this condition automatically holds when deep-Q network is used to fit the initial Q-estimator. The second part of (A3) is mild as the constant c_0 could be arbitrarily small. Suppose some parametric model (e.g., linear) is imposed to learn $\widehat{\omega}^{(\ell)}$. When the model is correctly specified, then we have $c_0 = 1$. When kernels are used for function approximation, the rate c_0 can be established in a similar manner as in Theorem 5.4 of Liao, Qi, and Murphy (2020).

(A4) requires the β-mixing coefficients of the process $\{(S_t, A_t, R_t)\}_{t\ge 0}$ to decay to zero at an exponential rate. These coefficients characterize the temporal dependence of the observations and are equal to zero when the data are independent. The smaller the coefficients, the weaker the dependence. When the propensity score is stationary over time, $\{(S_t, A_t, R_t)\}_{t\ge 0}$ forms a time-homogeneous Markov chain. (A4) is automatically satisfied when the Markov chain is geometrically ergodic (see Theorem 3.7 of Bradley 2005). Geometric ergodicity is less restrictive than those imposed in the existing reinforcement learning literature that requires observations to be independent (see e.g., Degris, White, and Sutton 2012) or to follow a uniform-ergodic Markov chain (see e.g., Zou, Xu, and Liang 2019). We also remark that the stationarity assumption in (A1) is assumed only to simplify the technical proof. Our theoretical results are equally applicable even without this condition (see e.g., the proof of Lemma 3 of Shi et al. 2020c).

(A5) is very similar to the positivity assumption imposed in single-stage decision making. These assumptions enable us to derive the following theorem.

Theorem 1. Assume (A1)–(A5) hold. $\widehat{Q}^{(\ell)}$, $\widehat{\omega}^{(\ell)}$ and the rewards are uniformly bounded. Then there exists some constant $\bar{c} > p/(2p + d)$ such that for any $a \in \mathbb{A}$,

$$\frac{1}{NT}\sum_{i,t} \mathrm{E}|\mathrm{E}(\widetilde{Q}_{i,t,a}|S_{i,t}) - Q^{\mathrm{opt}}(S_{i,t}, a)| = O\{(NT)^{-\bar{c}}\}.$$

Theorem 1 states that the conditional bias of $\widetilde{Q}_{i,t,a}$ decays at a rate of $(NT)^{-\bar{c}}$ on average. In comparison, under (A3), the squared prediction loss of the initial Q-estimator decays at a rate of $(NT)^{-2p/(2p+d)}$. Suppose the square bias and variance of $\widehat{Q}^{(\ell)}$ are of the same order. Then we expect $\mathrm{E}\{\widehat{Q}^{(\ell)}(a, S_{i,t})|S_{i,t}\}$ to approach $Q^{\mathrm{opt}}(S_{i,t}, a)$ at a rate of $(NT)^{-p/(2p+d)}$. Since $\bar{c} > p/(2p + d)$, biases of our pseudo outcomes are much smaller than the initial Q-estimators.

5.2. Efficiency Enhancement

In this section, we establish the convergence rates of the estimated contrast function and the derived optimal policy. Without loss of generality, we assume the state space $\mathbb{S} = [0, 1]^d$. We write $a_n \asymp b_n$ for two sequences $\{a_n\}_n$ and $\{b_n\}_n$ if there exists some universal constant $c \ge 1$ such that $c^{-1}a_n \le b_n \le ca_n$ for all n.

Theorem 2. Assume the conditions in Theorem 1 hold. Then there exists DNN class $\{\mathcal{T}_a\}_a$ with $L_a \asymp \log(NT)$ and $W_a \asymp (NT)^C \log(NT)$ for some $C > d/(2p + d)$ and any $a \ne a_0$ such that with probability approaching 1,

$$\mathrm{E}_{s\sim p_\infty}|\widehat{\tau}(a, s) - \tau^{\mathrm{opt}}(a, s)|^2 = O\{(NT)^{-c_1}\},$$

for some constant $2p/(2p + d) < c_1 \le 2p^*/(2p^* + d)$, where the expectation is taken with respect to the stationary state distribution.

Theorem 2 formally shows that our estimated contrast function converges at a faster rate than the Q-function computed by Q-learning type-estimators, leading to the desired efficiency enhancement property. To illustrate why the estimated contrast converges faster, suppose we have access to some unbiased pseudo outcome for $\tau(a, S_{i,t})$. Then under (A2), the estimated contrast function would converge at a minimax optimal rate of $(NT)^{-p^*/(2p^*+d)}$, which is much faster than that of the Q-estimator. In practice, we do not have access to unbiased pseudo outcomes. As such, the rate would depend on the bias of the pseudo outcome $\widetilde{Q}_{i,t,a} - \widetilde{Q}_{i,t,a_0}$. Nonetheless, the efficiency enhancement property holds as long as the bias decays faster than the convergence rate of the Q-estimator. The latter assertion is confirmed in Theorem 1.

We next show this in turn leads to an improvement in the value. More specifically, for any policy π, define the integrated value function $\mathcal{V}(\pi) = \int_{\mathbb{S}} V^{\pi}(s)\nu_0(s)ds$ where ν_0 denotes the density function of S_0. Let $\widehat{\pi}^{\tau}$ denote the derived policies based on the estimated contrast function $\widehat{\tau}$.

Theorem 3. Assume the conditions in Theorem 1 hold and ν_0 is uniformly bounded from above. Then

$$\mathcal{V}(\pi^{\text{opt}}) - \mathrm{E}\mathcal{V}(\widehat{\pi}^{\tau}) = O\{(NT)^{-\alpha_0 c_1/2}\},$$

where $\alpha_0 = (2+2\alpha)/(\alpha+2) > 1$ and α is defined in (A1).

Let $\widehat{\mathrm{Q}}$ denotes a Q-learning type estimator that satisfies

$$\mathrm{E}_{(a,s)\sim p_\infty} |\widehat{\mathrm{Q}}(a,s) - Q^{\text{opt}}(a,s)|^2 = O\{(NT)^{-2p/(2p+d)}\}, \quad (18)$$

and $\widehat{\pi}^{\mathrm{Q}}$ be the derived policy based on $\widehat{\mathrm{Q}}$. Similar to Theorem 3, we can show that $\mathrm{E}\mathcal{V}(\widehat{\pi}^{\mathrm{Q}})$ converges at a rate of $\alpha_0 p/(2p+d)$. Based on the fact that $c_1 > 2p/(2p+d)$, it is clear that the value of our estimated policy converges to the optimal value at a faster rate than those of policies computed by Q-learning type algorithms.

The convergence rates in Theorems 2 and 3 relies crucially on the exponent α in the margin condition (A1) and the convergence rate of the estimated density ratio in (A3), that is, $(NT)^{-c_0}$. The following corollary shows that under certain conditions on α and c_0, the exponent c_1 in both theorems achieve a maximum value of $2p^*/(2p^*+d)$.

Corollary 1. Suppose the conditions in Theorems 2 and 3 hold. Suppose $c_0 \geq 2p^*/(2p^*+d) - 2p/(2p+d)$ and $\alpha \geq 2[2 - \{p^*/(2p^*+d)\}/\{p/(2p+d)\}]^{-1} - 2$. Then with proper choice of the DNN class $\{\mathcal{T}_a\}_a$, we have for any $a \neq a_0$ that

$$\mathrm{E}_{s\sim p_\infty} |\widehat{\tau}(a,s) - \tau^{\text{opt}}(a,s)|^2 = O\{(NT)^{-2p^*/(2p^*+d)}\},$$

with probability approaching 1, and that $\mathcal{V}(\pi^{\text{opt}}) - \mathrm{E}\mathcal{V}(\widehat{\pi}^{\tau}) = O\{(NT)^{-\alpha_0 p^*/(2p^*+d)}\}$.

Notice that we do not require the optimal policy to be unique in order to establish the regret bound of the estimated optimal policy. This is because our proposal is value-based which derives the optimal policy using the estimated advantage function. The advantage function is well-defined despite that the optimal policy might not be unique, and the regret bound decays to zero as long as the estimated advantage function is consistent. To elaborate, let us revisit Example 8. By definition, when the state is nonpositive, both actions are optimal. The uniqueness assumption is thus violated. Nonetheless, the regret is zero since choosing either action is optimal.

Finally, we remark that although the proposed contrast function estimator converges at a faster rate than Q-learning type estimators, these rates are asymptotic. A potential limitation of the proposed method is that our estimated contrast function might have larger variance than Q-learning type estimators in finite samples, due to the use of importance sampling in constructing the pseudo outcomes. This reflects a bias-variance tradeoff. The proposed A-learning method might suffer from a larger variance whereas Q-learning type methods might suffer from a larger bias. This observation is consistent with the findings in the literature on learning DTRs (see e.g., Schulte et al. 2014).

6. Simulations

We evaluate the performance of our method using two synthetic datasets generated by the Open AI Gym environment (see *https://gym.openai.com/*) in this section. We consider the following Q-learning type baseline methods: (a) DQN; (b) double DQN (DDQN); (c) quantile DQN (QR-DQN). See Examples 3–5 for detailed discussion of these algorithms. As we have commented in Section 4, our policy optimization procedure at Step 2 is generally applicable to any Q-learning type algorithms. To validate this claim, for each of these Q-learning type methods in (a)–(c), we couple it with sample splitting to compute the initial Q-estimator in Step 2 based on each half of the data, and apply our proposal in Steps 3-5 to learn an optimal policy. This yields three estimated optimal policies. We denote them by (d) SEAL-DQN, (e) SEAL-DDQN and (f) SEAL-QR-DQN, respectively. Then we contrast them with the corresponding Q-learning type algorithms in (a)–(c) fitted based on the entire offline data. In addition to these baseline methods, we also consider three recently developed offline policy optimization methods in the computer science literature, including (g) batch-constrained deep Q-learning (BCQ, Fujimoto, Meger, and Precup 2019), (h) random ensemble mixture (REM, Agarwal, Schuurmans, and Norouzi 2020) and (i) bootstrapping error accumulation reduction (BEAR, Kumar et al. 2019). We compare them with the proposed procedure based on QR-DQN, which yields the best performance among (d)–(f).

6.1. *LunarLander-v2*

We conduct experiments in an OpenAI Gym environment, LunarLander-v2. Detailed description about this environment can be found at LunarLander-v2. To generate the data, we train a QR-DQN agent 500K time steps, with learning rate 0.0005. The estimated policy after 500K time steps is near optimal and solves the environment (e.g., achieves a score of 200 on average). The state-of-the-art optimal average reward is over 250[5]. We then terminate the training process, store all the generated trajectories encountered during the online training process and use them as the offline data. The behavior policy corresponds to the ϵ-greedy policy used to train the online QR-DQN agent with $\epsilon = 0.1$. The offline dataset consists of 1089 trajectories. Each trajectory lasts for 459 time steps on average. The average immediate reward equals 118.

The training data consist of 200 trajectories randomly sampled out of the 1089 trajectories. For each of the estimated optimal policy learned based on (a)–(i), we evaluate its value by computing the mean reward of 100 trajectories generated in the environment under this policy. We repeat the entire data generating process, the training and evaluation procedures 10 times with different random seeds. We also vary the number of training steps for the initial Q-estimator and apply the proposed method to each of the estimated Q-functions. For fair comparison, we use the same number of training steps (i.e., 20K, 30K, 40K, or 50K) to train the baseline policy.

[5] *https://medium.datadriveninvestor.com/training-the-lunar-lander-agent-with-deep-q-learning-and-its-variants-2f7ba63e822c*

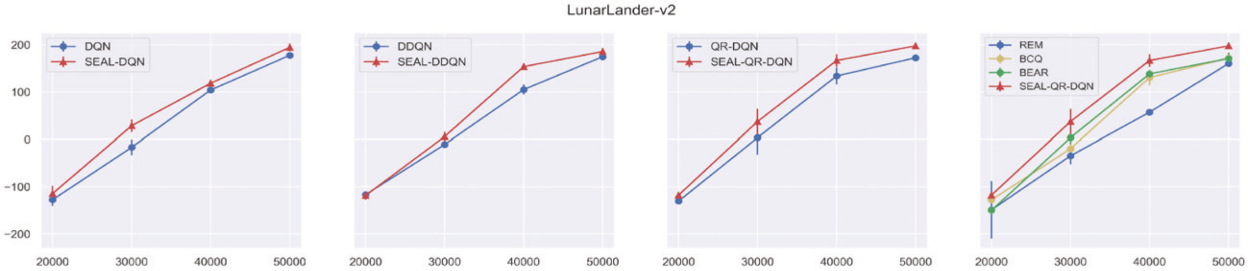

Figure 2. Synthetic data analysis results for LunarLander-v2. Horizontal axis represents the number of training steps used to train the initial Q-estimator based on half the data as well as the baseline method based on the entire dataset. Vertical axis represents the average reward of 100 evaluations. The error bar corresponds to the 95% confidence interval for the value, constructed based on 10 replications. The first three panels compare one baseline Q-learning algorithm (DQN, DDQN, QR-DQN) with our method that uses such a baseline to compute the initial Q-estimator. The last panel compares our algorithm based on QR-DQN against REM, BCQ, and BEAR.

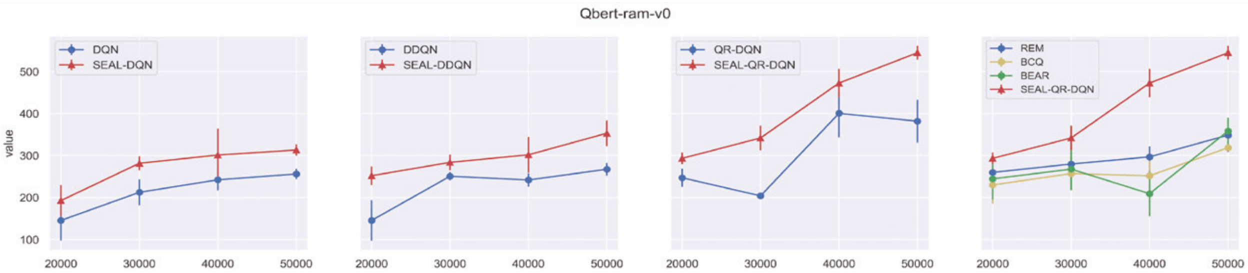

Figure 3. Synthetic data analysis results for Qbert-ram-v0. Same legend as Figure 2.

Reported in Figure 2 are the values of the estimated policies computed by (a)–(i) as well as the associated 95% confidence intervals, with different number of training steps. We summarize our findings as follows: (i) The proposed procedure achieves a larger value compared to the baseline methods in most cases; (ii) Our improvement is significant in many cases, as suggested by the error bar; (iii) All the methods get improved as the number of training steps increases.

6.2. *Qbert-ram-v0*

We next conduct experiments in another environment, Qbert-ram-v0. The best 100-episode average reward for Qbert-ram-v0 is 586.00 ± 12.16. We similarly train a Quantile DQN agent to generate 1373 trajectories. Each trajectory lasts for 364 time steps on average. The average return per trajectory equals 278. We similarly compare our procedures (d)–(f) with the baseline methods (a)–(c) and (g)–(i). Results are depicted in Figure 3. Overall, findings are very similar to those in Section 6.1. We notice that the performances of some deep Q-learning methods drop when the number of training step increases and cannot even improve after a few more iterations. We discuss this in detail in Appendix A.4, supplementary materials to save space.

Finally, it can very computationally expensive to implement deep RL algorithms in LunarLander-v2 and Qbert-ram-v0. For instance, in our implementation, it took a few hours to run one simulation. As such, our simulation results are aggregated over 10 runs only. We also remark that beginning with DQN, 5 or less runs are common in the existing RL literature, as it is often computationally prohibitive to evaluate more runs (Agarwal et al. 2021); see also the numerical studies in Mnih et al. (2015), Silver et al. (2016), Kumar et al. (2019), and Agarwal, Schuurmans, and Norouzi (2020).

7. The OhioT1DM Dataset

In this section, we use the OhioT1DM Dataset (Marling and Bunescu 2018) to illustrate the usefulness of our new method in mobile health applications. The data contains continuous measurements for six patients with type 1 diabetes over eight weeks. The objective is to learn an optimal policy that maps patients' time-varying covariates into the amount of insulin injected at each time to maximize patients' health status.

In our experiment, we divide each day of follow-up into 1 hr intervals and a treatment decision is made every hour. We consider three important time-varying state variables, including the average blood glucose level G_t during the 1 hr interval $(t-1, t]$, the carbohydrate estimate for the meal C_t during $(t-1, t]$ and Ex_t which measures exercise intensity during $(t-1, t]$. At time t, we define the action A_t by discretizing the amount of insulin In_t injected. The reward R_t is chosen according to the Index of Glycemic Control (Rodbard 2009) that is a deterministic function G_{t+1}. Detailed definitions of A_t and R_t are given in Appendix C, supplementary materials. We will receive a low reward if the patient's average blood glucoses level is outside the range $[80, 140]$. Let $X_t = (G_t, C_t, \mathrm{Ex}_t)$. We define the state S_t by concatenating measurements over the last four decision points, that is, $S_t = (X_{t-3}^T, A_{t-3}, \ldots, X_t)^\top$. This ensures the Markov assumption is satisfied (Shi et al. 2020b). The number of decision points for each patient in the OhioT1DM dataset ranges from 1119 to 1288. Transitions across different days are treated as independent trajectories. This yields 279 trajectories in total.

We use cross-validation to evaluate the performance of different algorithms. Specifically, we apply each of the method in (a)–(i) to the training dataset to learn an optimal policy. Then we apply the fitted Q-evaluation (FQE, Le, Voloshin, and Yue 2019) algorithm to the testing dataset to evaluate the values of these policies. FQE is very similar to FQI. It iteratively update

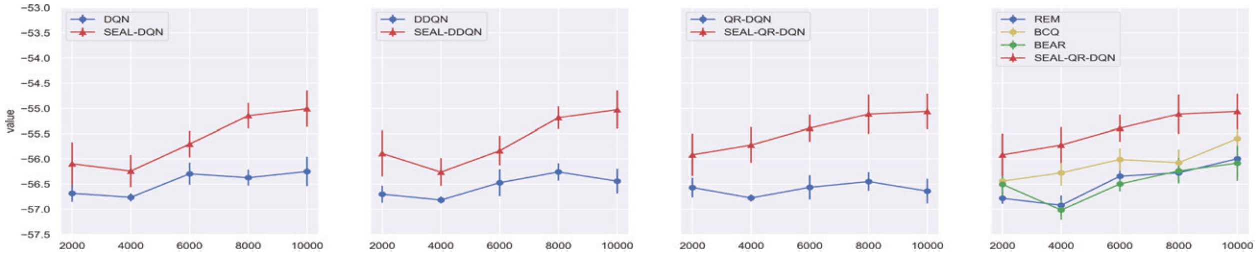

Figure 4. Real data analysis results. Same legend as Figure 2.

the state-action value using supervised learning algorithms. See Algorithm 1 in Appendix C, supplementary materials for details. In our implementation, we set the function approximator $\mathcal{F}$ to a class of DNN and apply deep learning to update the value. These estimated values are further aggregated over different training/testing combinations. Finally, we repeat this procedure 10 times with different random seeds to further aggregated the values. Results are reported in Figure 4. Our method performs significantly better than other baseline methods in most cases.

Supplementary Materials

The supplementary materials contain discussions of the p-smoothness assumption and the pessimistic principle, extensions of our proposal to the continuous action space, technical proofs and some additional implementation details.

Acknowledgments

The authors thank the AE, and the reviewers for their constructive comments and suggestions.

Disclosure Statement

The authors report there are no competing interests to declare.

Funding

CS' research is partly supported by the EPSRC grant EP/W014971/1. RS' research is partly supported by the NSF grants DMS-1555244 and DMS-2113637.

ORCID

Rui Song https://orcid.org/0000-0003-1875-2115

References

Agarwal, R., Schuurmans, D., and Norouzi, M. (2020), "An Optimistic Perspective on Offline Reinforcement Learning," in *International Conference on Machine Learning*, pp. 104–114. [241,242]

Agarwal, R., Schwarzer, M., Castro, P. S., Courville, A. C., and Bellemare, M. (2021), "Deep Reinforcement Learning at the Edge of the Statistical Precipice," in *Advances in Neural Information Processing Systems* (Vol. 34). [242]

Audibert, J.-Y., and Tsybakov, A. B. (2007), "Fast Learning Rates for Plug-in Classifiers," *The Annals of Statistics*, 35, 608–633. [240]

Bang, H., and Robins, J. M. (2005), "Doubly Robust Estimation in Missing Data and Causal Inference Models," *Biometrics*, 61, 962–973. [237]

Bradley, R. C. (2005), "Basic Properties of Strong Mixing Conditions. A Survey and some Open Questions," *Probability Surveys*, 2, 107–144. [239,240]

Chakraborty, B., Murphy, S., and Strecher, V. (2010), "Inference for Non-regular Parameters in Optimal Dynamic Treatment Regimes," *Statistical Methods in Medical Research*, 19, 317–343. [232,233]

Chen, X., and Qi, Z. (2022), "On Well-Posedness and Minimax Optimal Rates of Nonparametric q-function Estimation in Off-policy Evaluation," arXiv preprint arXiv:2201.06169. [236]

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., and Robins, J. (2018), "Double/Debiased Machine Learning for Treatment and Structural Parameters," *The Economic Journal*, 21, C1–C68. [237]

Dabney, W., Rowland, M., Bellemare, M. G., and Munos, R. (2018), "Distributional Reinforcement Learning with Quantile Regression," in *Thirty-Second AAAI Conference on Artificial Intelligence*. [233,237]

de Haan, P., Jayaraman, D., and Levine, S. (2019), "Causal Confusion in Imitation Learning," in *Proceedings of the NIPS*, pp. 11698–11709. [232]

Degris, T., White, M., and Sutton, R. S. (2012), "Off-policy Actor-Critic," in *Proceedings of the 29th International Conference on International Conference on Machine Learning*, pp. 179–186. [240]

Ertefaie, A., and Strawderman, R. L. (2018), "Constructing Dynamic Treatment Regimes over Indefinite Time Horizons," *Biometrika*, 105, 963–977. [232,233,234]

Fan, J., Wang, Z., Xie, Y., and Yang, Z. (2020), "A Theoretical Analysis of Deep Q-Learning," in *Learning for Dynamics and Control*, pp. 486–489. PMLR. [235,237]

Fujimoto, S., Meger, D., and Precup, D. (2019), "Off-policy Deep Reinforcement Learning without Exploration," in *International Conference on Machine Learning*, pp. 2052–2062. [241]

Giné, E., and Nickl, R. (2021), *Mathematical Foundations of Infinite-Dimensional Statistical Models*, Cambridge: Cambridge University Press. [235]

Hu, X., Qian, M., Cheng, B., and Cheung, Y. K. (2020), "Personalized Policy Learning using Longitudinal Mobile Health Data," *Journal of the American Statistical Association*, 116, 410–420. [232]

Jiang, N., and Li, L. (2016), "Doubly Robust Off-Policy Value Evaluation for Reinforcement Learning," in *International Conference on Machine Learning*, pp. 652–661. [233]

Jin, J., Song, C., Li, H., Gai, K., Wang, J., and Zhang, W. (2018), "Real-Time Bidding with Multi-Agent Reinforcement Learning in Display Advertising," in *Proceedings of the CIKM*, pp. 2193–2201. [232]

Kallus, N., and Uehara, M. (2019), "Efficiently Breaking the Curse of Horizon: Double Reinforcement Learning in Infinite-Horizon Processes," arXiv preprint arXiv:1909.05850. [233,237,238]

——— (2020), "Double Reinforcement Learning for Efficient Off-policy Evaluation in Markov Decision Processes," *Journal of Machine Learning Research*, 21, 1–63. [233]

Kennedy, E. H. (2020), "Optimal Doubly Robust Estimation of Heterogeneous Causal Effects," arXiv preprint arXiv:2004.14497. [233]

Kennedy, E. H., Balakrishnan, S., and Wasserman, L. (2022), "Minimax Rates for Heterogeneous Causal Effect Estimation," arXiv preprint arXiv:2203.00837. [236]

Kingma, D. P., and Ba, J. (2015), "Adam: A Method for Stochastic Optimization," in *3rd International Conference on Learning Representations, ICLR 2015*, San Diego, CA, USA, May 7–9, 2015, eds. Y. Bengio and Y. LeCun, Conference Track Proceedings. [239]

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, 1716–1720. [232]

Kormushev, P., Calinon, S., and Caldwell, D. G. (2013), “Reinforcement Learning in Robotics: Applications and Real-World Challenges,” *Robotics*, 2, 122–148. [232]

Kumar, A., Fu, J., Soh, M., Tucker, G., and Levine, S. (2019), “Stabilizing Off-policy Q-Learning via Bootstrapping Error Reduction,” in *Advances in Neural Information Processing Systems*, pp. 11761–11771. [241,242]

Le, H., Voloshin, C., and Yue, Y. (2019), “Batch Policy Learning under Constraints,” in *International Conference on Machine Learning*, pp. 3703–3712. [242]

Li, G., Cai, C., Chen, Y., Gu, Y., Wei, Y., and Chi, Y. (2021a), “Is Q-Learning Minimax Optimal? A Tight Sample Complexity Analysis,” arXiv preprint arXiv:2102.06548. [236]

Li, G., Wei, Y., Chi, Y., Gu, Y., and Chen, Y. (2020), “Sample Complexity of Asynchronous Q-Learning: Sharper Analysis and Variance Reduction,” in *Advances in Neural Information Processing Systems* (Vol. 33), pp. 7031–7043. [236]

Li, R., Wang, H., and Tu, W. (2021b), “Robust Estimation of Heterogeneous Treatment Effects using Electronic Health Record Data,” *Statistics in Medicine*, 40, 2713–2752. [233]

Liao, P., Qi, Z., and Murphy, S. (2020), “Batch Policy Learning in Average Reward Markov Decision Processes,” arXiv preprint arXiv:2007.11771. [232,240]

Liu, Q., Li, L., Tang, Z., and Zhou, D. (2018), “Breaking the Curse of Horizon: Infinite-Horizon Off-policy Estimation,” in *Advances in Neural Information Processing Systems*, pp. 5356–5366. [233,238]

Lu, W., Zhang, H. H., and Zeng, D. (2013), “Variable Selection for Optimal Treatment Decision,” *Statistical Methods in Medical Research*, 22, 493–504. [233]

Luckett, D. J., Laber, E. B., Kahkoska, A. R., Maahs, D. M., Mayer-Davis, E., and Kosorok, M. R. (2020), “Estimating ‘Dynamic Treatment Regimes in Mobile Health using v-learning,” *Journal of the American Statistical Association*, 115, 692–706. [232,234]

Luedtke, A. R., and van der Laan, M. J. (2016), “Statistical Inference for the Mean Outcome under a Possibly non-unique Optimal Treatment Strategy,” *The Annals of Statistics*, 44, 713–742. [232,240]

Maei, H. R., Szepesvári, C., Bhatnagar, S., and Sutton, R. S. (2010), “Toward Off-policy Learning Control with Function Approximation,” in *ICML*, pp. 719–726. [233]

Marling, C., and Bunescu, R. C. (2018), “The ohiot1dm Dataset for Blood Glucose Level Prediction,” in *KHD@ IJCAI*, pp. 60–63. [232,242]

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. (2015), “Human-Level Control through Deep Reinforcement Learning,” *Nature*, 518, 529–533. [233,237,242]

Murphy, S. A. (2003), “Optimal Dynamic Treatment Regimes,” *Journal of the Royal Statistical Society*, Series B, 65, 331–366. [232,233]

NeCamp, T., Sen, S., Frank, E., Walton, M. A., Ionides, E. L., Fang, Y., Tewari, A., and Wu, Z. (2020), “Assessing Real-Time Moderation for Developing Adaptive Mobile Health Interventions for Medical Interns: Micro-Randomized Trial,” *Journal of Medical Internet Research*, 22, e15033. [233]

Nie, X., Brunskill, E., and Wager, S. (2020), “Learning When-to-Treat Policies,” *Journal of the American Statistical Association*, 116, 392–409. [232]

Nie, X., and Wager, S. (2017), “Quasi-Oracle Estimation of Heterogeneous Treatment Effects,” arXiv preprint arXiv:1712.04912. [233]

Puterman, M. L. (1994), *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics, New York: Wiley. [234]

Qi, Z., Liu, D., Fu, H., and Liu, Y. (2020), “Multi-Àrmed Angle-based Direct Learning for Estimating Optimal Individualized Treatment Rules with Various Outcomes,” *Journal of the American Statistical Association*, 115, 678–691. [232]

Qian, M., and Murphy, S. A. (2011), “Performance Guarantees for Individualized Treatment Rules,” *The Annals of Statistics*, 39, 1180–1210. [232,233,240]

Riedmiller, M. (2005), “Neural Fitted q Iteration–First Experiences with a Data Efficient Neural Reinforcement Learning Method,” in *European Conference on Machine Learning*, pp. 317–328, Springer. [233]

Robins, J. M. (2004), “Optimal Structural Nested Models for Optimal Sequential Decisions,” in *Proceedings of the Second Seattle Symposium in Biostatistics*, pp. 189–326, Springer. [232,233]

Rodbard, D. (2009), “Interpretation of Continuous Glucose Monitoring Data: Glycemic Variability and Quality of Glycemic Control,” *Diabetes Technology & Therapeutics*, 11, S55–S67. [242]

Romano, J., and DiCiccio, C. (2019), “Multiple Data Splitting for Testing,” Technical Report. [237]

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, 640–661. [241]

Shi, C., Fan, A., Song, R., and Lu, W. (2018a), “High-Dimensional *A*-Learning for Optimal Dynamic Treatment Regimes,” *The Annals of Statistics*, 46, 925–957. [233,236]

Shi, C., Lu, W., and Song, R. (2020a), “Breaking the Curse of Nonregularity with Subagging—Inference of the Mean Outcome under Optimal Treatment Regimes,” *Journal of Machine Learning Research*, 21, 1–67. [240]

Shi, C., Song, R., Lu, W., and Fu, B. (2018b), “Maximin Projection Learning for Optimal Treatment Decision with Heterogeneous Individualized Treatment Effects,” *Journal of the Royal Statistical Society*, Series B, 80, 681–702. [232]

Shi, C., Wan, R., Chernozhukov, V., and Song, R. (2021), “Deeply-Debiased Off-Policy Interval Estimation,” in *International Conference on Machine Learning*, pp. 9580–9591. PMLR. [233]

Shi, C., Wan, R., Song, R., Lu, W., and Leng, L. (2020b), “Does the Markov Decision Process Fit the Data: Testing for the Markov Property in Sequential Decision Making,” in *International Conference on Machine Learning*, pp. 8807–8817. PMLR. [233,234,242]

Shi, C., Zhang, S., Lu, W., and Song, R. (2020c), “Statistical Inference of the Value Function for Reinforcement Learning in Infinite Horizon Settings,” arXiv preprint arXiv:2001.04515. [232,240]

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. (2016), “Mastering the Game of Go with Deep Neural Networks and Tree Search,” *Nature*, 529, 484–489. [232,242]

Song, R., Wang, W., Zeng, D., and Kosorok, M. R. (2015), “Penalized Q-Learning for Dynamic Treatment Regimens,” *Statistica Sinica*, 25, 901–920. [232]

Stone, C. J. (1982), “Optimal Global Rates of Convergence for Nonparametric Regression,” *Annals of Statistics*, 10, 1040–1053. [235,236]

Sutton, R. S., and Barto, A. G. (2018), *Reinforcement Learning: An Introduction*. Adaptive Computation and Machine Learning (2nd ed.), Cambridge, MA: MIT Press. [232,237]

Thomas, P., and Brunskill, E. (2016), “Data-Efficient Off-Policy Policy Evaluation for Reinforcement Learning,” in *International Conference on Machine Learning*, pp. 2139–2148. PMLR. [233]

Tian, L., Alizadeh, A. A., Gentles, A. J., and Tibshirani, R. (2014), “A Simple Method for Estimating Interactions between a Treatment and a Large Number of Covariates,” *Journal of the American Statistical Association*, 109, 1517–1532. [233]

Tsybakov, A. B. (2004), “Optimal Aggregation of Classifiers in Statistical Learning,” *The Annals of Statistics* 32, 135–166. [240]

Uehara, M., Huang, J., and Jiang, N. (2019), “Minimax Weight and q-function Learning for Off-policy Evaluation,” arXiv preprint arXiv:1910.12809. [238]

Van Hasselt, H., Guez, A., and Silver, D. (2016), “Deep Reinforcement Learning with Double q-learning,” in *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 30). [233,237]

Wainwright, M. J. (2019), “Variance-Reduced *q*-Learning is Minimax Optimal,” arXiv preprint arXiv:1906.04697. [236]

Wallace, M. P., and Moodie, E. E. M. (2015), "Doubly-Robust Dynamic Treatment Regimen Estimation via Weighted Least Squares," *Biometrics*, 71, 636–644. [232]

Wang, L., Zhou, Y., Song, R., and Sherwood, B. (2018), "Quantile-Optimal Treatment Regimes," *Journal of the American Statistical Association*, 113, 1243–1254. [232]

Watkins, C. J., and Dayan, P. (1992), "Q-Learning," *Machine Learning*, 8, 279–292. [233]

Xu, Z., Li, Z., Guan, Q., Zhang, D., Li, Q., Nan, J., Liu, C., Bian, W., and Ye, J. (2018), "Large-Scale Order Dispatch in On-demand Ride-Hailing Platforms: A Learning and Planning Approach," in *Proceedings of the ACM KDD*, pp. 905–913. [232]

Zhang, B., Tsiatis, A. A., Laber, E. B., and Davidian, M. (2013), "Robust Estimation of Optimal Dynamic Treatment Regimes for Sequential Treatment Decisions," *Biometrika*, 100, 681–694. [232]

Zhang, Y., Laber, E. B., Davidian, M., and Tsiatis, A. A. (2018), "Estimation of Optimal Treatment Regimes using Lists," *Journal of the American Statistical Association*, 113, 1541–1549. [232]

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, 583–598. [232]

Zhou, W., Zhu, R., and Qu, A. (2021), "Estimating Optimal Infinite Horizon Dynamic Treatment Regimes via pt-learning," arXiv preprint arXiv:2110.10719. [232]

Zhu, R., Zhao, Y.-Q., Chen, G., Ma, S., and Zhao, H. (2017), "Greedy Outcome Weighted Tree Learning of Optimal Personalized Treatment Rules," *Biometrics*, 73, 391–400. [232]

Zou, S., Xu, T., and Liang, Y. (2019), "Finite-Sample Analysis for Sarsa with Linear Function Approximation," in *Advances in Neural Information Processing Systems*, pp. 8665–8675. [240]