Relaxed Stationary Distribution Correction Estimation for Improved Offline Policy Optimization

Woosung Kim, Donghyeon Ki, Byung-Jun Lee

Department of Artificial Intelligence
Korea University, Seoul, Republic of Korea
{wsk208, peop1e1n, byungjunlee}@korea.ac.kr

Abstract

One of the major challenges of offline reinforcement learning (RL) is dealing with distribution shifts that stem from the mismatch between the trained policy and the data collection policy. Stationary distribution correction estimation algorithms (DICE) have addressed this issue by regularizing the policy optimization with f-divergence between the state-action visitation distributions of the data collection policy and the optimized policy. While such regularization naturally integrates to derive an objective to get optimal state-action visitation, such an implicit policy optimization framework has shown limited performance in practice. We observe that the reduced performance is attributed to the biased estimate and the properties of conjugate functions of f-divergence regularization. In this paper, we improve the regularized implicit policy optimization framework by relieving the bias and reshaping the conjugate function by relaxing the constraints. We show that the relaxation adjusts the degree of involvement of the sub-optimal samples in optimization, and we derive a new offline RL algorithm that benefits from the relaxed framework, improving from a previous implicit policy optimization algorithm by a large margin.

Introduction

Reinforcement learning (RL) focuses on training agents to make desirable decisions in complex environments through trial and error. While RL has shown remarkable successes in a number of domains with favorable properties, its online learning nature makes it expensive and inefficient in practice, especially in real-world settings with high interaction costs. To this end, offline RL framework was proposed to obtain policies without any online interaction, using only a fixed dataset. However, offline RL turns out to face its own challenge called distribution shift, where the error accumulates as the policy is attempted to improve, as it deviates from the data collection policy. Numerous offline RL algorithms have been proposed to limit the distribution shift and result in the policy that actually improves from the data collection policy.

Our main focus is on implicit policy optimization methods for offline RL, where we optimize stationary distribution correction estimation (DICE) that corrects the data distribution to the stationary distribution induced by the policy being trained. In this case, it is common to impose a f-divergence regularization between the stationary distributions of the data collection policy and the trained policy (Wu, Tucker, and Nachum 2019; Lee et al. 2021) to mitigate the aforementioned distributional shift problem. Here, f-divergence plays a significant role, enabling safe estimation of the RL objective without sampling from the target policy and establishing approximate lower bounds on performance.

Despite their favorable theoretical properties, however, these implicit policy optimization algorithms, e.g. AlgaeDICE (Nachum et al. 2019b) and OptiDICE (Lee et al. 2021), have shown limited performance in practice, compared to other offline RL algorithms based on the conventional actor-critic framework. In particular, implicit policy optimization algorithms tend to show weak performance in large environments, and in datasets with low optimality (i.e., data collection policy being closer to random policy).

In this paper, we identify the main cause that weakens f-divergence regularized implicit policy optimization methods and propose *Offline **P**olicy **O**ptimization via **Rel**axed Stationary **DI**stribution **C**orrection **E**stimation* (PORelDICE), a refined algorithm that addresses the raised issues by relaxing the constraint. We first reformulate the objective in a unified convex conjugate form for better analysis and show how the bias of the estimator and the shape of the unified convex conjugate function contribute to the performance degradation. While the use of a biased estimator is common in recent deep RL algorithms (Dai et al. 2018; Lee et al. 2021), we show that the bias in implicit policy optimization method may result in significantly different policy, similar to the phenomenon known as *optimistic transition* (Levine et al. 2020). On the other hand, if we adopt an additional function approximator to alleviate the bias, the resultant algorithm tends to diverge due to the non-decreasing property of the conjugate function.

Our proposed algorithm, PORelDICE, relaxes the positivity constraint of the visitation distribution, which result in different shape of the conjugate function and adoption of samples with low advantage during optimization. We show that the relaxation corresponds to the minimization of the upper bound of the original objective, and leads to improved stability. We also show that the proposed relaxation can be applied to other offline RL frameworks, and the practical implementation of Sparse Q-learning (Xu et al. 2023) can

be interpreted using the constraint relaxation. We demonstrate that PORelDICE shows strong performance over various D4RL (Fu et al. 2020) offline benchmarks, improving from the prior DICE-based implicit policy optimization algorithm by a large margin.

Preliminaries

Markov Decision Process (MDP) We assume the reinforcement learning problem under an infinite-horizon discounted Markov Decision Process (MDP) framework. MDP can be represented as a tuple $\mathcal{M} = \langle S, A, T, R, \mu_0, \gamma \rangle$, where S is a set of states s and A is a set of actions a. $T(s'|s,a) : S \times A \to \Delta(S)$ is a transition probability from the state-action pair (s,a) to the next state s'. $R(s,a) : S \times A \to [0, r_{\max}]$ is a reward function of the state-action pair (s,a). $\mu_0 \in \Delta(S)$ is an initial state distribution and $\gamma \in [0,1)$ is the discount factor.

A policy $\pi(a|s) : S \to \Delta(A)$ gives a distribution of actions a of the agent given the state s. Stationary distribution, or state-action visitation probability, of π is defined as $d^\pi(s,a) := (1-\gamma)\sum_{t=0}^{\infty} \gamma^t \Pr(s_t = s, a_t = a)$. $d^\pi(s,a)$ represents discounted sum of probabilities of the agent with policy π visiting (s,a). The stationary distribution satisfies Bellman flow constraint $d^\pi(s,a) = \pi(a|s)((1-\gamma)\mu_0(s) + \gamma \sum_{\bar{s},\bar{a}} T(s|\bar{s},\bar{a}) d^\pi(\bar{s},\bar{a}))$, which implies that the count of state-action pairs exiting a state must match the sum of state-action pairs entering the state and the frequency of the state being the initial state.

Reinforcement learning aims to learn a policy that maximizes expected return $J(\pi) = \mathbb{E}_{(s,a)\sim d^\pi(s,a)}[R(s,a)]$, which represents an expected sum of discounted rewards.

Linear Programming form By using linear programming (LP) characterization of V-function (V-LP) (Puterman 1994; Bertsekas 1995; Bertsekas and Tsitsiklis 1996; Nachum and Dai 2020), policy optimization can be seen as solving the following optimization problem:

$$\min_V \ (1-\gamma)\mathbb{E}_{s\sim\mu_0}[V(s)] \tag{1}$$
$$\text{s.t. } V(s) \geq R(s,a) + \gamma \mathbb{E}_{s'\sim T(s'|s,a)}[V(s')] \quad \forall s, a.$$

The solution to this optimization problem is the state value of the corresponding MDP. The dual of V-LP gives an equivalent optimization problem with respect to state-action visitation distribution (Nachum and Dai 2020). Using LP duality, the dual of V-LP is defined as

$$\max_{d\geq 0} \ \mathbb{E}_{(s,a)\sim d(s,a)}[R(s,a)] \tag{2}$$
$$\text{s.t. } \sum_a d(s,a) = (1-\gamma)\mu_0(s) + \gamma \sum_{\bar{s},\bar{a}} T(s|\bar{s},\bar{a}) d(\bar{s},\bar{a}) \ \forall s$$

where Bellman flow constraints are relaxed to be only on states. Using Lagrangian duality, the optimization problem (2) can be further reformulated to an unconstrained optimization problem.

Regularized objective in offline RL In this paper, we focus on offline reinforcement learning (RL) setting, where the interaction between the agent and the environment is not allowed. Instead, the dataset $D = \{(s_i, a_i, r_i, s'_i)\}_i^N$ consisting of N single-step transitions is given to optimize policy π. We denote the empirical distribution of dataset D as d^D. In offline RL, the main purpose is to obtain a policy that can perform better than behavior patterns observed in the dataset D. Such an attempt can cause distribution shift, where the offline RL agent is trained under one distribution, while being evaluated on a different distribution to optimize its behavior (Levine et al. 2020).

One way to mitigate the distribution shift is to regularize the f-divergence between state-action visitation distribution d^π and empirical distribution d^D. It corresponds to additionally minimizing the f-divergence term with the original return maximization problem, resulting in the following regularized policy optimization:

$$\max_\pi \mathbb{E}_{(s,a)\sim d^\pi}[R(s,a)] - \alpha D_f(d^\pi \| d^D), \tag{3}$$

where f is continuously differentiable and strictly convex function and α determines the amount of regulation to impose, and D_f denotes the f-divergence $D_f(d^\pi \| d^D) = \mathbb{E}_{(s,a)\sim d^D}\left[f\left(\frac{d^\pi(s,a)}{d^D(s,a)}\right)\right]$. By choosing appropriate α, this regularization encourages conservative behavior, penalizing rare or nonexistent state-action pairs in the dataset.

Related works Algorithms based on stationary distribution correction estimation (DICE) reformulate (3) into a more tractable form by utilizing LP characterization of value functions and Lagrangian duality (Nachum et al. 2019a,b; Lee et al. 2021). AlgaeDICE (Nachum et al. 2019b), which first proposed the regularized policy optimization framework, derived unconstrained objective from the dual of the regularized Q-LP. The regularization enabled the objective to be evaluated with samples only from the given dataset (Nachum and Dai 2020). However, AlgaeDICE can suffer from numerical instability, since it optimizes over a max-min-max problem using out-of-distribution actions. OptiDICE (Lee et al. 2021) solves this issue by using V-LP with relaxed Bellman flow constraints. OptiDICE reformulates the same optimization objective (3) to a simple minimization problem by deriving the closed-form solution of the inner maximization problem.

Pathologies in Regularized Implicit Policy Optimization Framework

We begin with introducing regularized policy optimization framework proposed by Lee et al. (2021). Equivalent loss function is derived in a unified convex conjugate form to analyze how Lagrange multipliers are optimized. Based on the analysis, we find the causes of performance degradation. In particular, we focus on the bias of the derived algorithm and the conjugate function, to see how it affects the optimization.

Regularized Implicit Policy Optimization

Regularized implicit policy optimization framework (Lee et al. 2021) consists of two stages: value optimization and policy extraction. In value optimization stage, optimization

on visitation distribution is conducted to obtain the optimal ratio between state-action visitation distribution. In policy extraction stage, a policy is updated with the optimal ratio to induce optimal state-action visitation.

Value optimization Value optimization can be expressed as a convex optimization problem where f-divergence is added to the dual representation of V-LP as a convex regularization.

$$\max_{d\geq 0} \mathbb{E}_{(s,a)\sim d}[R(s,a)] - \alpha D_f(d(s,a)\|d^D(s,a)) \tag{4}$$
$$\text{s.t.} \sum_{\bar{a}} d(s,\bar{a}) = (1-\gamma)\mu_0(s) + \gamma \sum_{\bar{s},\bar{a}} T(s|\bar{s},\bar{a})d(\bar{s},\bar{a}) \; \forall s$$

As proposed by Lee et al. (2021), the Bellman flow constraints only on states enable implicit optimization of the policy by optimizing the induced state-action visitation distribution d instead. The optimal d^* that solves the problem will be a valid state-action visitation distribution due to the constraints while maximizing the objective.

We derive Lagrangian dual to solve the constrained optimization problem (4). Ratio between state-action visitation distribution $d(s,a)/d^D(s,a)$ is replaced with $w(s,a)$ for simpler notation.

$$\max_{w\geq 0}\min_{\nu} (1-\gamma)\mathbb{E}_{s\sim\mu_0}[\nu(s)] - \mathbb{E}_{(s,a)\sim d^D}\left[\alpha f\left(w(s,a)\right) - w(s,a)e_\nu(s,a)\right] \tag{5}$$

where $\nu(s)$ is introduced as the Lagrangian multiplier for the Bellman flow constraints and $e_\nu(s,a) = R(s,a) + \gamma\mathbb{E}_{s'\sim T(s'|s,a)}\nu(s') - \nu(s)$. Noting that $\nu(s)$ becomes a state-value function when $\alpha \to 0$ as shown in the duality of V-LP, e_ν can be understood as an advantage function.

The order of optimization on w and ν is switched based on the strong duality of (5) and Karush-Kuhn-Tucker (KKT) conditions are applied to optimize $w(s,a)$ first. Loss function for ν and optimal visitation distribution ratio $w^*(s,a)$ are given below. Details of the entire derivation are given in the appendix.

$$\min_{\nu} (1-\gamma)\mathbb{E}_{s\sim\mu_0}[\nu(s)] - \mathbb{E}_{(s,a)\sim d^D}\left[\alpha f\left(w^*(s,a)\right) - w^*(s,a)e_\nu(s,a)\right] \tag{6}$$
$$w^*(s,a) = \max\left(0, (f')^{-1}\left(\frac{e_\nu(s,a)}{\alpha}\right)\right) \tag{7}$$

After the minimization on $\nu(s)$ is complete, $w^*(s,a)$ with $\nu^*(s)$ is considered as the optimal visitation distribution ratio.

Policy extraction In policy extraction stage, optimal state-action visitation distribution ratio $w^*(s,a)$ from the previous stage is used to obtain policy π that induces d^*. In tabular domain, the policy can be easily obtained by performing marginalization $\pi^*(a|s) = \frac{d^D(s,a)w^*(s,a)}{\sum_{\bar{a}} d^D(s,\bar{a})w^*(s,\bar{a})}$.

For continuous domain, we use weighted behavior cloning:

$$\max_{\psi} \mathbb{E}_{d^*}[\log \pi_\psi(a|s)] = \mathbb{E}_{d^D}[w^*(s,a)\log\pi_\psi(a|s)]$$

Based on the derivation above, OptiDICE (Lee et al. 2021) shows decent performance when trained in small environments or with datasets collected using near-optimal policies. However, the algorithm shows limited performance compared to other offline RL algorithms when the environment gets larger, or the quality of the data collection policy is low. In the following subsections, we focus on how value is optimized in value optimization stage and show how it affects the performance of the framework.

Unified Convex Conjugate Function

In this sub, we derive the equivalent loss function to (6) with the unified convex conjugates to better understand how ν is optimized. From (5), we start by replacing the positivity constraint on state-action visitation distribution $w \geq 0$ with indicator function $\delta_+(w(s,a))$.

$$\min_{\nu}\max_{w} (1-\gamma)\mathbb{E}_{s\sim\mu_0}[\nu(s)] - \mathbb{E}_{d^D}\left[\alpha f\left(w(s,a)\right) + \alpha\delta_+\left(w(s,a)\right) - w(s,a)e_\nu(s,a)\right] \tag{8}$$

where the convex indicator function $\delta_+(x) = 0$ when $x \geq 0$ and $\delta_+(x) = \infty$ otherwise. The equivalence can be seen from the fact that maximization on w fails with negative infinity when $w \geq 0$ is violated.

The reformulated objective provides a new interpretation of the framework where policy optimization is regularized with the sum of two convex functions: f from f-divergence and indicator function δ_+. We denote the sum of two functions as $f_+(x) := f(x) + \delta_+(x)$. We also denote convex conjugate of $f(x)$ as $f^*(y)$, which is defined as a function of the optimal value of convex optimization problem $f^*(y) = \max_x xy - f(x)$. Using the defined notations, we can rewrite the objective (6) as:

$$\min_{\nu} (1-\gamma)\mathbb{E}_{s\sim\mu_0}[\nu(s)] + \mathbb{E}_{(s,a)\sim d^D}\left[\alpha f_+^*\left(\frac{e_\nu(s,a)}{\alpha}\right)\right],$$
$$f_+^*(y) = \begin{cases} \frac{1}{2}y^2 + y & \text{if } y \geq -1 \\ -\frac{1}{2} & \text{otherwise} \end{cases}. \tag{9}$$

Here, we instantiated f using Pearson χ^2-divergence: $f(x) = \frac{1}{2}(x-1)^2$. Note that this choice is the most widely adopted default choice for f-divergence regularized RL algorithms other than KL divergence (Nachum et al. 2019b; Lee et al. 2021; Xu et al. 2023).

We now describe two characteristics of f_+^* by computing its gradient: **non-decreasing** and **zero-gradient region**. We use the property of convex conjugate that the gradient of $f^*(y)$ is equivalent to the optimal x that maximizes the optimization problem used to define the conjugate function.

Proposition 1 (Gradient of $f_+^*(y)$)**.** *For convex function $f_+(x) = f(x) + \delta_+(x)$, which is sum of the convex function $f(x)$ and the indicator function $\delta_+(x)$, its convex conjugate $f_+^*(y)$ is non-decreasing:*

$$\frac{\partial f_+^*(y)}{\partial y} = x^\star = \max(0, (f')^{-1}(y)) \geq 0 \tag{10}$$

where $x^\star = \arg\max_x xy - f_+(x)$ is the solution that gives f_+^*. Furthermore, $f_+^*(y)$ has flat region with zero-gradient where $y < f'(0)$.

Proof. From the definition of convex conjugate, we have $f_+^*(y) = \max_x xy - f_+(y) = x^\star y - f_+(x^\star)$. Taking derivative, we get $\frac{\partial f_+^*(y)}{\partial y} = x^\star$. □

The proposition implies that imposing the constraints in a convex optimization problem leads to constraining the gradient of its conjugate function. This particular shape of f_+^* results in bad optimization properties of the objective (6), as we will see in the following subs.

Having a zero-gradient region can be problematic and can significantly degrade learning efficiency, as previously studied with the dying ReLU problems (Lu et al. 2019). However, while all f_+^* is non-decreasing, the existence of zero-gradient region is determined by the choice of f-divergence. For example, KL divergence with $f'(0) = -\infty$ has no zero-gradient region with all $w^*(s,a)$ positive. Previously, Lee et al. (2021) has pointed out that Pearson χ^2-divergence can suffer from the dying gradient and used a soft version of χ^2 to prevent loss of efficiency:

$$f_{\text{soft}-\chi^2}(x) = \begin{cases} x\log x + x - 1 & \text{if } 0 < x < 1 \\ \frac{1}{2}(x-1)^2 & \text{if } x \geq 1 \end{cases}. \quad (11)$$

However, as we will see, soft χ^2 alone is not able to address the other issues to be raised in the following subs.

Bias in Surrogate Objective and Its Implication

The previous work uses single-sample estimate of advantage $\hat{e}_\nu(s,a,s') = r(s,a) + \gamma\nu(s') - \nu(s)$ to approximate $e_\nu(s,a)$, i.e. uses the surrogate objective

$$\min_\nu \ (1-\gamma)\mathbb{E}_{s\sim\mu_0}[\nu(s)] + \mathbb{E}_{d^D}\left[\alpha f_+^*\left(\frac{\hat{e}_\nu(s,a,s')}{\alpha}\right)\right]. \quad (12)$$

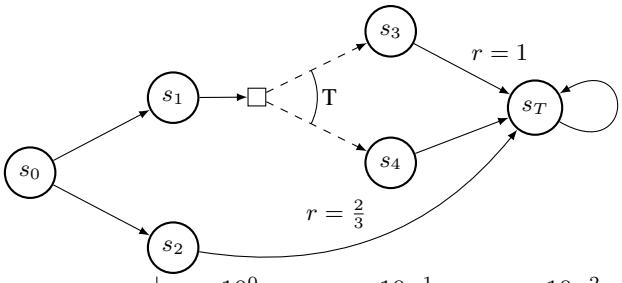
However, the approximation leads to a biased estimate of the latter term. The bias occurs as the expectation over transition probability $T(s'|s,a)$ is moved outside the non-linear function f_+^*, noting that $e_\nu(s,a) = \mathbb{E}_{s'}[\hat{e}_\nu(s,a,s')]$ but $f_+^*(e_\nu(s,a)) \leq \mathbb{E}_{s'}[f_+^*(\hat{e}_\nu(s,a,s'))]$ where equality holds only when the transition is deterministic.

While the surrogate objective serves as an upper bound of the original objective, the bias may result in serious implications when combined with the property of f_+^*. Due to the zero-gradient property of f_+^*, ν is not updated by the samples with $\hat{e}_\nu(s,a,s') < \alpha f'(0)$. In other words, we are optimizing ν only with *good* transition samples that give $\hat{e}_\nu(s,a,s') > \alpha f'(0)$, similar to optimizing ν under original objective and the modified transition probability, biased toward *good* next states (i.e., the *optimistic transition*). Even if f-divergence with no zero gradient region is adopted, unless f_+^* is linear, the surrogate objective will optimize ν by weighting *good* transition samples more than the others.

It is also worthwhile to note that $w^*(s,a)$ optimized based on the surrogate objective is not even a valid stationary distribution correction.

Proposition 2 (Validity of $\hat{\nu}^*$)**.** *For $\hat{\nu}^*$ that minimizes the surrogate objective* (12)*, the corresponding stationary distribution correction*

$$\hat{w}^*(s,a) = \max\left(0, (f')^{-1}\left(\frac{e_{\hat{\nu}^*}(s,a)}{\alpha}\right)\right)$$



α	10^0	10^{-1}	10^{-2}
$\pi_\nu(a_{s_1}\|s_0)$	0.483	0.359	0.0
$\pi_{\hat{\nu}}(a_{s_1}\|s_0)$	0.483	0.589	1.0
$\hat{d}^*(s_3, a_{s_T})$	2.96×10^{-3}	5.77×10^{-3}	9.80×10^{-3}
$\hat{d}^*(s_4, a_{s_T})$	1.41×10^{-3}	0.0	0.0

Figure 1: We illustrate the toy example for a better understanding of the implication of bias in the surrogate objective. Each bold and dashed line represents action and transition, respectively. All rewards except the one shown in the figure are 0. Surrogate objective results in a policy optimized under optimistic transition, which prefers s_1 over s_2.

is not a valid stationary distribution correction, i.e., π such that $\hat{d}^*(s,a) = \hat{w}^*(s,a)d^D(s,a)$ satisfying Bellman flow constraints for π does not exist in general. The exception is when the transition is deterministic.

The proof is given in the appendix. It is noteworthy that the biased objective is equivalent to the unbiased objective when the transition is deterministic, and has no negative implication in those cases. However, the adoption of a function approximator for estimating ν aggregates nearby states, and the deterministic transition functions may effectively work as stochastic transition functions under function approximations.

Illustrative example We illustrate the implication of the biased objective (12) through a toy example given in Figure 1. In the example, s_1 has a 0.5 probability of transitioning to s_3 which gives a reward of 1. On the other hand, s_2 always gives a reward of $2/3$, and we need to choose the state to go from s_0. Normally, we should prefer s_2 over s_1 since it has a higher expected return, as π_ν shows in the figure (the policy optimized under unbiased objective). However, as shown with $\pi_{\hat{\nu}}$ in the figure, regardless of the choice of α, the policy optimized under the surrogate objective prefers s_1 over s_2, due to the ignored samples that transit to s_4. Furthermore, it turns out that $\hat{d}^*(s_4, a_{s_T}) = 0 \neq \hat{d}^*(s_3, a_{s_T})$ while they should be the same due to 0.5 transition probabilities from s_1, implying that these $\hat{d}^*$s optimized under biased objective are not valid stationary distributions. Details of the example can be found in the appendix.

Additional function approximator to alleviate bias We alleviate the bias that incurs optimistic transition by additionally adopting a state-action value like function $U(s,a)$ to estimate $R(s,a) + \gamma\mathbb{E}_{s'\sim T(s'|s,a)}\nu(s')$. Given that $U(s,a)$ accurately approximates as intended, the expectation on transition probability $T(s'|s,a)$ is no longer affected by f_+^*

and *bad* transitions are not excluded from the training. New loss functions on ν and U are given as

$$\begin{aligned}\min_{\nu}\ & (1-\gamma)\mathbb{E}_{s\sim\mu_0}[\nu(s)] \\ & + \mathbb{E}_{(s,a)\sim d^D}\left[\alpha f_+^*\left(\frac{U(s,a)-\nu(s)}{\alpha}\right)\right] \\ \min_{U}\ & \mathbb{E}_{(s,a,s')\sim d^D}\left[\left(R(s,a)+\gamma\nu(s')-U(s,a)\right)^2\right]\end{aligned} \tag{13}$$

However, we empirically found that directly optimizing (13) leads to the divergence of ν and U. The divergence stems from non-decreasing f_+^*, which causes ν to only increase. Unlike the biased objective (12) that balances the increment of $\nu(s)$ by the decrement of $\nu(s')$, we now no longer have the decrement of $\nu(s')$ since we adopted $U(s,a)$ instead of it. As ν continues to increase, U is optimized to match ν and triggers a further increase in ν. Due to insufficient coverage of initial state distribution and generally small coefficient $(1-\gamma)$, the minimization of the first term $(1-\gamma)\mathbb{E}_{s\sim\mu_0}[\nu(s)]$ is not usually sufficient to prevent the divergence.

A similar phenomenon can be found when we set $\tau = 1$ in Implicit Q-learning (IQL) (Kostrikov, Nair, and Levine 2021), which uses an asymmetric loss function for the optimization of the state value function. In IQL, expectile regression is used to estimate the upper expectile of the state-action value function with $\tau \in [0,1]$. When $\tau = 1$, expectile regression is also characterized as a non-decreasing function with the zero-gradient region, and it approximates the Bellman optimality operator. These similar characteristics of f_+^* causes IQL to diverge when $\tau = 1$. As IQL successfully operates under $\tau \neq 1$, it motivates the framework based on f_+^* without the non-decreasing property.

Relaxed Implicit Policy Optimization via Mirror Descent

In this section, we aim to modify non-decreasing f_+^* to stabilize the framework. We show that by temporarily relaxing the positivity constraint $w \geq 0$, we can naturally get f_+^* without non-decreasing property, by relaxing the lower bound of the gradient of f_+^*. We derive a practical algorithm from the relaxed framework and discuss the benefits of relaxation from two perspectives. We also show that the practical detail of policy loss used in Sparse Q-learning (Xu et al. 2023) can be also interpreted as an adaptation of the proposed relaxation technique.

To relax the positivity constraint, we add another indicator function to f_+^* and divide the function into two convex functions:

$$\begin{aligned}f_+(x) &:= f(x) + \delta_+(x) \\ &= f(x) + \delta_+(x-\epsilon) + \delta_+(x) = f_\epsilon(x) + \delta_+(x).\end{aligned}$$

where $\epsilon \in (-\infty, 0)$ and $f_\epsilon(x) := f(x) + \delta_+(x-\epsilon)$. Additional indicator function $\delta_+(x-\epsilon)$ does not affect $f_+(x)$ as $x < \epsilon$ is already punished by $\delta_+(x)$. We use $f_\epsilon(x)$ and $\delta_+(x)$ to reformulate the optimization problem as,

$$\begin{aligned}&\max_{w,w_\epsilon}\min_{\nu}\ (1-\gamma)\mathbb{E}_{s\sim\mu_0}[\nu(s)] \\ &- \mathbb{E}_{d^D}\left[\alpha f_\epsilon\left(w_\epsilon(s,a)\right) + \alpha\delta_+(w(s,a)) - w_\epsilon(s,a)e_\nu(s,a)\right] \\ &\text{s.t. } w(s,a) = w_\epsilon(s,a)\ \forall s,a\end{aligned} \tag{14}$$

where the equality condition is added to maintain the equivalence to the original problem.

We obtain the relaxed optimization problem by swapping the order of optimization from $\max_{w,w_\epsilon}\min_\nu$ to $\max_w \min_\nu \max_{w_\epsilon}$. We consider the inner optimization problem $\min_\nu \max_{w_\epsilon}$ as our relaxed optimization problem.

$$\begin{aligned}&\min_{\nu}\max_{w_\epsilon}\ (1-\gamma)\mathbb{E}_{s\sim\mu_0}[\nu(s)] \\ &- \mathbb{E}_{(s,a)\sim d^D}\left[\alpha f_\epsilon\left(w_\epsilon(s,a)\right) - w_\epsilon(s,a)e_\nu(s,a)\right]\end{aligned}$$

where f_ϵ indicates that the constraint is relaxed to $w \geq \epsilon$.

We obtain optimal solution of the inner maximization problem $w_\epsilon^*(s,a) = \max\left(\epsilon, (f')^{-1}\left(e_\nu(s,a)/\alpha\right)\right)$ and loss function on ν using convex conjugate.

$$\begin{aligned}\min_{\nu}\ & (1-\gamma)\mathbb{E}_{s\sim\mu_0}[\nu(s)] \\ & + \mathbb{E}_{(s,a)\sim d^D}\left[\alpha f_\epsilon^*\left(\frac{U(s,a)-\nu(s)}{\alpha}\right)\right]\end{aligned}$$

$$f_\epsilon^*(y) = \begin{cases}\frac{1}{2}y^2 + y & \text{if } y \geq \epsilon - 1 \\ \epsilon y - \frac{1}{2}\epsilon^2 + \epsilon - \frac{1}{2} & \text{otherwise}\end{cases},$$

where we provide an example of f_ϵ^* when Pearson χ^2-divergence is used with the relaxation. We compute the gradient of f_ϵ^* to show the relaxation removes the non-decreasing property and zero-gradient region of f_+^*.

Proposition 3 (Gradient of f_ϵ^*)**.** *Given convex conjugate $f^*(y) := \max_x xy - f(x)$ of $f(x)$ and $x^\star = (f')^{-1}(y)$ that maximizes $xy - f(x)$, convex conjugate $f_\epsilon^*(y)$ and its gradient is given as,*

$$\begin{aligned}f_\epsilon^*(y) :&= \max_{x\geq\epsilon}\ xy - f(x) \\ &= \begin{cases}f^*(y) & \text{if } y \geq f'(\epsilon) \\ \epsilon(y - f'(\epsilon)) + f^*(f'(\epsilon)) & \text{otherwise}\end{cases} \\ \frac{\partial f_\epsilon^*(y)}{\partial y} &= x_\epsilon^\star = \max(\epsilon, x^\star)\end{aligned} \tag{15}$$

where constraint $x \geq \epsilon$ decides the lower bound of the gradient of $f_\epsilon^*(y)$.

We replace maximization on w in (14) with the projection of negative $w_\epsilon^*(s,a) \in [\epsilon, 0)$ to zero to obtain the optimal ratio $w^*(s,a)$ for policy extraction. We now present the learning objective of the relaxed framework:

PORelDICE The learning objective of ν, U, π are:

$$\min_{\nu}\ (1-\gamma)\mathbb{E}_{\mu_0}[\nu(s)] + \mathbb{E}_{d^D}\left[\alpha f_\epsilon^*\left(\frac{U(s,a)-\nu(s)}{\alpha}\right)\right] \tag{16}$$

$$\min_{U}\ \mathbb{E}_{(s,a,s')\sim d^D}\left[\left(R(s,a)+\gamma\nu(s')-U(s,a)\right)^2\right] \tag{17}$$

$$\min_{\pi}\ \mathbb{E}_{d^D}\left[w^*(s,a)\log(\pi(a|s))\right], \tag{18}$$

$$\text{where } w^*(s,a) = \max\left(0, (f')^{-1}\left(\frac{U(s,a)-\nu(s)}{\alpha}\right)\right)$$

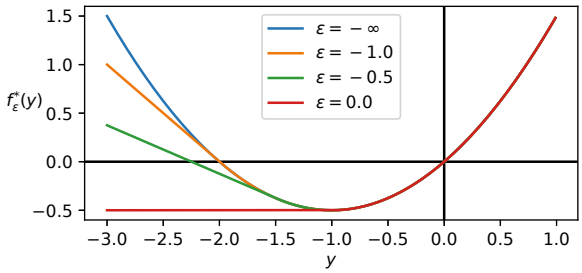


Figure 2: $f^*_\epsilon(y)$ when $f(x) = \frac{1}{2}(x-1)^2$

The detailed pseudo-code is provided in the appendix.

Benefits of Relaxed Policy Optimization Framework

We show ν loss (16) derived from the relaxed policy optimization framework is an upper bound of ν loss (13). The relationship between the losses is easily shown by comparing two conjugate functions f^*_ϵ with different ϵ.

$$f^*_+(y) \leq f^*_\epsilon(y), \quad \forall y$$

where $f^*_+(y)$ corresponds to $f^*_\epsilon(y)$ with $\epsilon = 0$. The inequality can be inferred from $f^*_\epsilon(y)$ described in (15) and Figure 2, where the left side of $f^*(y)$ is replaced with tangent line whose gradient is ϵ. We note that f^*_ϵ does not upper bound f^*_+ when $\epsilon \geq 0$ and is not valid.

Divergence of ν As previously mentioned, the divergence of ν in (13) is caused by non-decreasing property of f^*_+. The gradient of ν in $f^*_+((U(s,a) - \nu(s))/\alpha)$ is always negative, and ν is updated to only increase. However, ν does not diverge with relaxed ν loss (16), since the gradient of $f^*_\epsilon((U(s,a) - \nu(s))/\alpha)$ can be both positive and negative. This is because the relaxation lowered the lower bound of the gradient in f^*_ϵ from zero to ϵ.

Setting the lower bound of the gradient as ϵ has another advantage in that the samples with $U(s,a) - \nu(s) < \alpha f'(\epsilon)$ have the same gradient when updating ν. This lowers the impact of negative outliers on $U(s,a) - \nu(s)$ and enhances the robustness of the framework, similar to Huber loss.

Data samples with low advantage When optimizing ν with (16), contribution to the loss of a sample $(s,a) \sim d^D$ is determined by the optimal ratio of state-action visitation distributions: $w^*(s,a) = \max(0, (f')^{-1}(e_\nu(s,a)/\alpha))$ and $w^*_\epsilon(s,a) = \max(\epsilon, (f')^{-1}(e_\nu(s,a)/\alpha))$. In the original objective (6), samples with low advantage e_ν leads to $w^*(s,a) = 0$, being excluded from optimization of ν. This corresponds to e_ν/α of the samples being located in the zero-gradient region of f^*_+.[1]

This has two implications: 1) in the early stage of optimization, samples are excluded based on premature estimate of $e_\nu(s,a)$, lowering the efficiency of the training. 2) it effectively leads to smaller state-action coverage of the dataset, resulting in ν with poor generalization ability. We conjecture the latter as one of the reasons why the original objective performed less competitively in large environments (requires better generalization) or low-quality datasets (samples being more excluded).

On the other hand, in the relaxed framework, samples with low advantage have $w^*_\epsilon(s,a) = \epsilon$ and affect the optimization on ν as a penalty with negative ϵ.

As ϵ gets smaller, ν is more affected by samples with low advantage. $\nu(s)$ approaches $E_{a\sim\pi_D(a|s)}[U(s,a)]$ when $\epsilon = -\infty$ and approaches $\max_a U(s,a)$ when $\epsilon = 0$. This is similar to adjusting τ in IQL, but the difference is that the impact of negative advantage on ν is limited by ϵ.

Interpretation of modified SQL policy objective We show that by applying the proposed relaxation technique on f-divergence between policies, we can derive the practical policy objective used in Sparse Q-learning (Xu et al. 2023). Starting from the behavior-regularized MDP problem:

$$\max_\pi \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t,a_t) - \alpha f_+ \left(\frac{\pi(a_t|s_t)}{\pi_D(a_t|s_t)} \right) \right) \right]$$

where the indicator function corresponds to positivity constraint of π and $f(x) = \frac{1}{2}(x-1)$ corresponds to Neyman χ^2-divergence. By temporarily relaxing the positivity constant, it results in an SQL-like algorithm that has similar V and Q loss but different actor loss to the original SQL (Proof in Appendix):

$$\min_\pi \mathbb{E}_{d^D} \left[\max\left(0, \left(\frac{Q(s,a) - V(s)}{2\alpha} \right) \right) \log(\pi(a|s)) \right]$$

SQL uses the same actor loss by removing the "1+" term from its original actor loss as shown in Appendix D in Xu et al. (2023), but the intuition behind the modification is not fully explained. We emphasize that our relaxation technique can be one of the interpretations of such modification. In the experiment section, we denote this as *SQL trick*, and show that *SQL (w/ trick)* generally shows better performance than *SQL (w/o trick)*.

Experiments

In this section, we present PoRelDICE and compare it to previous offline reinforcement learning algorithms utilized in Deep Data-Driven Reinforcement Learning, i.e, D4RL benchmark (Fu et al. 2020). In D4RL benchmarks, we use 9 tasks of Gym-MuJoCo, 6 tasks of Antmaze and 3 tasks of Kitchen, which are the frequently adopted benchmarks in recent studies. Details of the algorithm and the experiment are given in the appendix.

Gym-Mujoco, Antmaze, Kitchen Datasets

We evaluate our algorithm in various D4RL environments and its normalized scores are given in Table 1. We compare PORelDICE with BC, CQL (Kumar et al. 2020), IQL (Kostrikov, Nair, and Levine 2021), SQL (w/ and w/o trick) (Xu et al. 2023) and OptiDICE (Lee et al. 2021). Table 1 shows that PORelDICE has reached state-of-the-art performance on multiple tasks. By addressing the

[1]While the use of soft χ^2 prevents the complete exclusion of samples, the exponentially decreasing f^* result in limited contribution from samples with low advantage as well.

Env	BC	CQL	IQL	SQL (w/ trick)	SQL (w/o trick)	OptiDICE (wbc)	PORelDICE
Hopper-m	52.9	58.5	66.3	67.3±6.9	63.8±7.3	57.0±5.8	**74.9**±10.2
Hopper-m-r	18.1	95.0	95.2	90.4±6.3	65.6±2.8	63.3±4.3	**95.4**±2.0
Hopper-m-e	52.5	105.4	101.5	108.3±8.0	**109.4**±4.2	80.6±16.0	101.3±9.6
Walker2d-m	75.3	72.5	72.5	77.6±5.7	77.6±6.3	64.6±14.3	**82.5**±0.6
Walker2d-m-r	26.0	77.2	76.1	66.2±12.0	66.6±8.1	44.7±4.0	**80.4**±4.4
Walker2d-m-e	107.5	109.6	110.6	111.2±0.2	109.1±0.1	108.1±0.7	**111.3**±0.5
Halfcheetah-m	42.6	44.0	47.4	47.5±0.3	43.9±0.2	42.9±0.2	**49.7**±0.15
Halfcheetah-m-r	36.6	**45.5**	44.2	43.7±0.5	41.3±2.8	38.5±2.3	45.0±0.5
Halfcheetah-m-e	55.2	90.7	86.7	91.0±5.4	91.5±1.2	91.9±0.7	**93.3**±0.3
Gym-mujoco-total	466.7	698.4	700.5	703.2	668.8	600.4	**733.8**
Antmaze-u	54.6	84.8	85.5	**91.2**±2.5	90.8±3.6	62.0±5.9	86.4±3.3
Antmaze-u-d	45.6	43.4	66.7	56.8±2.2	53.0±5.5	**71.2**±6.8	55.4±1.9
Antmaze-m-p	0	65.2	72.2	71.8±2.7	71.2±7.3	1.0±0.7	**74.8**±3.9
Antmaze-m-d	0	54.0	**71.0**	57.2±18.8	57.6±18.7	9.2±0.9	49.4±20.1
Antmaze-l-p	0	38.4	39.6	39.6±2.4	41.6±4.8	0 ± 0.0	**50.4**±6.1
Antmaze-l-d	0	31.6	**47.5**	37.2±1.2	37.2±6.0	0 ± 0.0	43.6±5.9
Antmaze-total	100.2	317.4	**382.5**	353.8	351.4	144.5	360.0
Kitchen-c	33.8	43.8	61.4	60.0±5.4	65.0±7.6	63.0±8.5	**73.0**±7.3
Kitchen-p	33.9	49.8	46.1	64.0±10.0	71.0±4.8	54.0±3.7	**72.0**±3.9
Kitchen-m	47.5	51.0	52.8	**57.0**±10.0	42.0±10.4	49.0±5.7	43.0±8.5
Kitchen-total	115.2	144.6	160.3	181.0	178.0	166.0	**188.0**

Table 1: Normalized scores of PORelDICE compared with model-free offline reinforcement learning algorithms. Our algorithms attain SOTA performance on multiple tasks. We average the score and get a 95% confidence interval with 5 seeds.

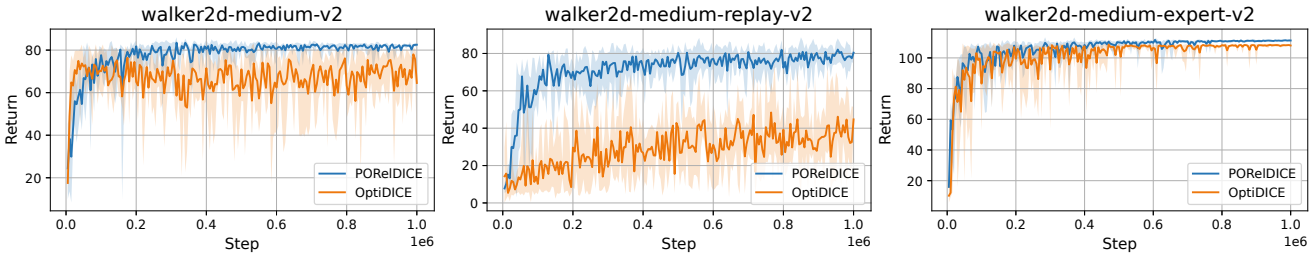


Figure 3: Learning curves of PORelDICE and OptiDICE on D4RL Walker2d datasets.

two issues of the implicit policy optimization framework, we achieve noticeable performance improvement from OptiDICE. In Mujoco domains, compared to other baselines, PORelDICE achieves the highest performance in 7 out of 9 tasks. In Antmaze and Kitchen domains, our algorithms show competitive results to the prior methods. We notice that while OptiDICE shows very low performance in extremely complex tasks such as Antmaze-medium and Antmaze-large, PORelDICE shows significantly improved performance compared to OptiDICE, even outperforming other baselines in some of the tasks.

We also report the learning curves of the algorithms in Figure 3. It can be found that PORelDICE improves both stability and final performance of OptiDICE. As shown in previous sections, the surrogate objective used in OptiDICE suffers from bias toward high advantage transitions. Under the function approximation, more diverse datasets effectively result in more stochastic transitions, and the bias will become more significant. PORelDICE resolves this issue, and is expected to improve more in diverse (sub-optimal) datasets. Such a trend can be observed in the results.

Conclusion

In this paper, we focus on improving offline reinforcement learning (RL) based on stationary distribution correction estimation (DICE). We identify two causes of the performance degradation in the existing framework proposed by Lee et al. (2021): the use of biased estimate and the shape of the conjugate function. To address these issues, we make two modifications of alleviating the bias with an additional function approximator and applying relaxation to positivity constraint on state-action visitation distribution. We also show the degree of relaxation is closely related to how samples with low advantage are treated. We demonstrate the efficacy of the proposed PORelDICE over various D4RL benchmarks, outperforming a number of well-known offline RL algorithms.

Acknowledgments

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2022-0-00311, Development of Goal-Oriented Reinforcement Learning Techniques for Contact-Rich Robotic Manipulation of Everyday Objects) and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2022R1F1A1074880)

References

Bertsekas, D.; and Tsitsiklis, J. N. 1996. *Neuro-dynamic programming*. Athena Scientific.

Bertsekas, D. P. 1995. Dynamic Programming and Optimal Control.

Dai, B.; Shaw, A.; Li, L.; Xiao, L.; He, N.; Liu, Z.; Chen, J.; and Song, L. 2018. SBEED: Convergent reinforcement learning with nonlinear function approximation. In *International Conference on Machine Learning*, 1125–1134. PMLR.

Fu, J.; Kumar, A.; Nachum, O.; Tucker, G.; and Levine, S. 2020. D4rl: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*.

Kostrikov, I.; Nair, A.; and Levine, S. 2021. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*.

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

Lee, J.; Jeon, W.; Lee, B.; Pineau, J.; and Kim, K.-E. 2021. Optidice: Offline policy optimization via stationary distribution correction estimation. In *International Conference on Machine Learning*, 6120–6130. PMLR.

Levine, S.; Kumar, A.; Tucker, G.; and Fu, J. 2020. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*.

Lu, L.; Shin, Y.; Su, Y.; and Karniadakis, G. E. 2019. Dying relu and initialization: Theory and numerical examples. *arXiv preprint arXiv:1903.06733*.

Nachum, O.; Chow, Y.; Dai, B.; and Li, L. 2019a. Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections. *Advances in Neural Information Processing Systems*, 32.

Nachum, O.; and Dai, B. 2020. Reinforcement learning via fenchel-rockafellar duality. *arXiv preprint arXiv:2001.01866*.

Nachum, O.; Dai, B.; Kostrikov, I.; Chow, Y.; Li, L.; and Schuurmans, D. 2019b. Algaedice: Policy gradient from arbitrary experience. *arXiv preprint arXiv:1912.02074*.

Puterman, M. L. 1994. Markov Decision Processes: Discrete Stochastic Dynamic Programming.

Wu, Y.; Tucker, G.; and Nachum, O. 2019. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361*.

Xu, H.; Jiang, L.; Li, J.; Yang, Z.; Wang, Z.; Chan, V. W. K.; and Zhan, X. 2023. Offline rl with no ood actions: In-sample learning via implicit value regularization. *arXiv preprint arXiv:2303.15810*.