

---

# No Experts, No Problem: Avoidance Learning from Bad Demonstrations

---

Anonymous Author(s)

Affiliation

Address

email

## Abstract

This paper addresses the problem of learning avoidance behavior within the context of offline imitation learning. In contrast to conventional methodologies that prioritize the replication of expert or near-expert demonstrations, our work investigates a setting where expert (or desirable) data is absent, and the objective is to learn to eschew undesirable actions by leveraging demonstrations of such behavior (i.e., learning from negative examples).

To address this challenge, we propose a novel training objective grounded in the maximum entropy principle. We further characterize the fundamental properties of this objective function, reformulating the learning process as a cooperative inverse Q-learning task. Moreover, we introduce an efficient strategy for the integration of unlabeled data (i.e., data of indeterminate quality) to facilitate unbiased and practical offline training. The efficacy of our method is evaluated across standard benchmark environments, where it consistently outperforms state-of-the-art baselines.

## 1 Introduction

Imitation learning [1, 47, 23] offers a feasible alternative to Reinforcement Learning (RL), enabling agents to learn directly from expert demonstrations without the need for explicit reward signals. It has proven effective in several tasks, even with limited expert data, and is particularly useful in capturing human preferences.

Most existing imitation learning approaches prioritize maximizing task performance (i.e., expected return) by closely mimicking expert demonstrations [18, 11, 28, 12]. However, in practice, expert or near-expert demonstrations may be unavailable or insufficient. First, acquiring expert demonstrations can be prohibitively expensive or time-consuming, as it often requires specialized skills or resources [40].

Second, in many domains, such as healthcare [15] or autonomous driving [6], expert-level performance may be rare or difficult to define, leading to a scarcity of high-quality demonstrations [30]. Third, expert demonstrations may not cover the full range of scenarios or edge cases, limiting their generalizability [43].

Instead of expert demonstrations, there may be collections of undesirable demonstrations that should be avoided and sub-optimal behaviors that may be partially imitated. In the development of self-

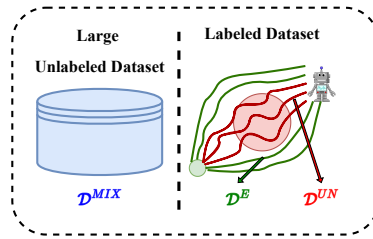


Figure 1: Datasets for offline imitation learning. In our setting, we only use Undesirable labeled dataset ( $\mathcal{D}^{UN}$ ) and Unlabeled dataset ( $\mathcal{D}^{MIX}$ ).

37 driving cars [5], while companies may collect user driving data to train their models, the system must  
 38 ensure it does not replicate faulty behavior, such as traffic violations or unsafe driving practices [6].  
 39 In the field of treatment optimization, data may include actions that led to bad patient outcomes, and  
 40 the system must learn to avoid such behaviors [15].

41 Although this is an important and interesting problem setting, there has been limited research address-  
 42 ing it effectively. Formally, given a dataset containing *undesired demonstrations* we wish to avoid,  
 43 alongside a much larger unlabeled dataset consisting of both desired and undesired demonstrations,  
 44 the goal is to learn a desirable policy that avoids undesired trajectories. To the best of our knowledge,  
 45 only SafeDICE [20] directly addresses this setting by mixing the undesirable and unlabeled datasets,  
 46 assigning negative weights to the former, and then mimicking the mixed policy. This approach, how-  
 47 ever, may suffer from the fact that *the mixed policy is not necessarily a desirable one* to follow, which  
 48 is often the case in practice. There are existing methods, such as preference-based RL [8, 32, 17]  
 49 and Discriminator-Weighted Behavioral Cloning (DWBC) [45] that, while not specifically designed  
 50 to address this problem, can be adapted to handle it.

51 Towards addressing this problem of avoiding undesirable trajectories, our key contributions are:

- 52 • First, we formulate the new learning problem as a cooperative training task to ensure  
 53 stable training, in contrast to prior imitation learning approaches where the objective is  
 54 adversarial [18, 11, 28].
- 55 • To efficiently solve the training problem using limited undesirable demonstrations, we  
 56 introduce our algorithm, *UNIQ*, which employs an occupancy correction mechanism to  
 57 recast the training objective, allowing the expectation over the undesirable policy to be  
 58 unbiasedly approximated using unlabeled trajectories.
- 59 • We evaluate our method on two popular benchmark environments, Mujoco and Safety-Gym,  
 60 using public datasets [38, 21]. Our experiments demonstrate superior performance compared  
 61 to several state-of-the-art baselines.

## 62 2 Related Work

63 **Imitation learning.** Imitation learning is a key technique for learning from demonstrations. Be-  
 64 havioral Cloning (BC) maximizes the likelihood of expert demonstrations but often struggles due  
 65 to distributional shift [40]. To improve, Generative Adversarial Imitation Learning [18, 11] aligns  
 66 the learner’s policy with the expert’s using GANs [14], while SQL [39] assigns simple rewards  
 67 to expert and non-expert demonstrations to learn a value function. PWIL [9] uses the Wasserstein  
 68 distance [44] to compute rewards. While these methods show promise, they rely on online interaction,  
 69 which can be impractical. For offline learning, AlgaeDICE [34] and ValueDICE [29] use Station-  
 70 ary Distribution Correction Estimation (DICE) but face stability issues. Inspired by ValueDICE,  
 71 O-NAIL [4] introduced an offline method without adversarial training. IQ-learn [12], a popular  
 72 approach, supports both online and offline learning and offers a state-of-the-art framework with  
 73 several variants developed based on it [2, 19]. Unlike the above mentioned works, our focus in this  
 74 paper is on offline **reverse** imitation learning that aims to avoid undesirable trajectories, as opposed  
 75 to imitating expert trajectories. As indicated earlier, this requires fundamentally different methods  
 76 due to the nature of the problem.

77 **Imitation Learning from Sub-optimal Demonstrations.** There are two main research directions in  
 78 this area. The first focuses on online and offline preference-based imitation learning methods. Online  
 79 approaches, such as T-REX [8], PrefPPO [31], and PEBBLE [32], leverage ranked sub-optimal  
 80 demonstrations to learn a preference-based reward function using the Bradley-Terry model [7].  
 81 While these methods achieve strong performance, they rely on interactions with the environment.  
 82 In contrast, offline methods, such as those proposed by [24, 22, 17], rely heavily on an extensive  
 83 dataset of pairwise trajectory comparisons. SPRINQL [19] addressed this reliance on a large offline  
 84 dataset, by utilizing demonstrations categorized into different levels of expertise, resulting in better  
 85 performance with fewer comparison data points. While these methods concentrate on imitating  
 86 preferred (or expert) trajectories, our focus is on avoiding non-preferred (or undesired) trajectories.  
 87 This important distinction necessitates solving a different optimization problem and consequently a  
 88 change in methodology.

89 The second direction focuses the use of additional unlabeled datasets to enhance learning from expert  
 90 data. Beginning with DemoDICE [26], several DICE-based methods [33, 25, 46] have been developed  
 91 to utilize small sets of expert demonstrations, supplemented by larger unlabeled datasets. In addition  
 92 to these DICE-based methods, DWBC [45] proposes a simple and efficient method based on training  
 93 a classifier using positive-unlabeled learning [27]. SafeDICE [20] presents a DICE-based framework  
 94 capable of learning from undesirable demonstrations. This method combines an undesirable policy  
 95 (represented by an undesirable dataset) with a random policy (represented by a larger unlabeled  
 96 dataset), assigns negative weights to the undesirable policy, and then applies a standard DICE-based  
 97 approach [26, 25] to mimic the combined policy by minimizing the KL divergence between the  
 98 learning policy and the mixed policy. The primary limitation of this approach is that, with the quality  
 99 of the unlabeled dataset being unknown, imitating the mixed policy may not lead to the desired  
 100 learning outcome.

### 101 3 Background

102 **Preliminaries.** We consider a MDP defined by the following tuple  $\mathcal{M} = \langle S, A, r, P, \gamma, s_0 \rangle$ , where  
 103  $S$  denotes the set of states,  $s_0$  represents the initial state set,  $A$  is the set of actions,  $r : S \times A \rightarrow \mathbb{R}$   
 104 defines the reward function for each state-action pair, and  $P : S \times A \rightarrow S$  is the transition function,  
 105 i.e.,  $P(s'|s, a)$  is the probability of reaching state  $s' \in S$  when action  $a \in A$  is made at state  $s \in S$ ,  
 106 and  $\gamma$  is the discount factor. In reinforcement learning (RL), the aim is to find a policy that maximizes  
 107 the expected long-term accumulated reward:  $\max_{\pi} \{ \mathbb{E}_{(s,a) \sim \rho_{\pi}} [r(s, a)] \}$ , where  $\rho_{\pi}$  is the occupancy  
 108 measure of policy  $\pi$ :  $\rho_{\pi}(s, a) = (1 - \gamma)\pi(a|s) \sum_{t=1}^{\infty} \gamma^t P(s_t = s|\pi)$ .

109 **Offline MaxEnt IRL** The goal of offline MaxEnt IRL is to derive a reward function  $r(s, a)$  based  
 110 on a set of expert demonstrations,  $\mathcal{D}^E$  without interacting with the environment. Let  $\rho^E$  denote the  
 111 occupancy measure of the expert policy. The MaxEnt IRL framework [47, 16], aims to recover the  
 112 expert’s reward function by maximizing the gap between the expected reward under the expert’s  
 113 policy and the maximum expected reward across all other policies (as determined by the inner  
 114 minimization):

$$\max_r \min_{\pi} \left\{ \mathbb{E}_{\rho^E} [r(s, a)] - \mathbb{E}_{\rho_{\pi}} [r(s, a)] - H(\pi, \mu) - \psi(r) \right\} \quad (1)$$

115 where  $H(\pi, \mu) = \mathbb{E}_{\rho^{\pi}} \left[ -\beta \log \frac{\pi(s, a)}{\mu(a|s)} \right]$  is the discounted causal entropy of the policy  $\pi$  given a  
 116 behavior policy  $\mu$ ,  $\beta$  is the regularization strength, and  $\psi(r) : \mathbb{R}^{S \times A} \rightarrow \mathbb{R}$  is a convex reward  
 117 regularizer. When  $\mu$  is uniform, the objective reduces to the original MaxEnt objective [47]. In the  
 118 offline RL setting,  $\mu$  is set to be the behavior policy that generated the offline dataset. As a result, this  
 119 objective imposes a conservative KL constraint on the learned policy, ensuring it stays close to the  
 120 behavior policy. This helps mitigate the out-of-distribution issues that commonly arise in offline RL  
 121 [36, 16].

**Inverse Q-learning (IQ-Learn) from expert demonstrations.** Given a reward function  $r$  and a  
 policy  $\pi$ , the soft Bellman equation is defined as

$$\mathcal{B}_{\pi}^{\pi}[Q](s, a) = r(s, a) + \gamma \mathbb{E}_{s'} [V^{\pi}(s')], \text{ where } V^{\pi}(s) = \mathbb{E}_{a \sim \pi(a|s)} \left[ Q(s, a) - \beta \log \frac{\pi(a|s)}{\mu(a|s)} \right].$$

122 The Bellman equation  $\mathcal{B}_{\pi}^{\pi}[Q] = Q$  is contractive and always yields a unique Q solution [12, 36]. In IQ-  
 123 Learn [12], they further define an inverse soft-Q Bellman operator  $\mathcal{T}^{\pi}[Q] = Q(s, a) - \gamma \mathbb{E}_{s'} [V^{\pi}(s')]$ .  
 124 [12] show that for any reward function  $r(a, s)$ , there is a unique  $Q^*$  function such that  $\mathcal{B}_{\pi}^{\pi}[Q^*] = Q^*$ ,  
 125 and for a  $Q^*$  function in the  $Q$ -space, there is a unique reward function  $r$  such that  $r = \mathcal{T}^{\pi}[Q^*]$ . This  
 126 result suggests that one can safely transform the objective function of Equation 1 from  $r$ -space to the  
 127  $Q$ -space as follows:

$$\max_Q \min_{\pi} \Phi(\pi, Q) = \mathbb{E}_{\rho^E} [\mathcal{T}^{\pi}[Q](s, a)] - \mathbb{E}_{\rho_{\pi}} [\mathcal{T}^{\pi}[Q](s, a)] - H(\pi, \mu) - \psi(\mathcal{T}^{\pi}[Q](s, a)) \quad (2)$$

128 which has several advantages, namely, the objective function  $\Phi(\pi, Q)$  is concave in  $\pi$  and  
 129 convex in  $Q$ . Moreover, the inner problem  $\min_{\pi} \Phi(\pi, Q)$  has a closed form solution as

130  $\exp(Q(s, a)/\beta)/\sum_a \exp(Q(s, a')/\beta)$  As a result, the maximin problem can be converted to a  
 131 *non-adversarial* problem in the Q-space as:

$$\max_Q \mathbb{E}_{\rho^E}[\mathcal{T}[Q](s, a)] - (1 - \gamma)\mathbb{E}_{s_0}[V^Q(s)] - \psi(\mathcal{T}[Q](s, a)) \quad (3)$$

132 where  $\mathcal{T}[Q](s, a) = Q(s, a) - \gamma\mathbb{E}_{s' \sim P(s'|s, a)}[V^Q(s')]$  and  $V^Q(s) =$   
 133  $\beta \log \left( \sum_a \mu(a|s) \exp(Q(s, a)/\beta) \right)$ , which is a softmax of the Q function. The reward  
 134 function can then be recovered as  $r^Q(s, a) = \mathcal{T}[Q](s, a)$ . Thus, in (3), the objective can be  
 135 interpreted as training a reward function (via a Q-function) that maximizes the expected reward  
 136 under the expert policy while minimizing the overall expected reward. An important and appealing  
 137 characteristic of the inverse Q-learning framework described above is that the learning objective  
 138 is concave in Q, theoretically guaranteeing convergence to a unique solution in the Q-space. This  
 139 property also ensures stable and robust training outcomes in practice. Later, we will show that this  
 140 property does not hold in our learning-from-mistakes setting, prompting us to adopt a more recent  
 141 framework, Extreme Q-learning [13], to reformulate the training process using updates based on  
 142 convex loss functions.

## 143 4 Undesired Demonstrations driven Inverse Q-learning - UNIQ

144 We now describe our approach for avoidance (and not imitation) learning of undesirable demonstra-  
 145 tions in the presence of an unlabeled (unknown quality) demonstrations dataset. First, we explain  
 146 our novel MaxEnt objective that facilitates this avoidance learning problem (Section 4.1). However,  
 147 directly solving this novel objective in reward space is inefficient, so we next provide a reformulation  
 148 in the the Q-space (Section 4.2). Third, we leverage the extreme Q-learning method (Section 4.3)  
 149 to solve this reformulated objective. Finally, we make our approach sample efficient by utilizing an  
 150 unlabeled dataset of trajectories (Section 4.4).

### 151 4.1 Novel MaxEnt Objective

152 In our setting, we have a set of undesired demonstrations, denoted as  $\mathcal{D}^{\text{UN}}$ , along with a supplementary  
 153 set of unlabeled demonstrations, denoted as  $\mathcal{D}^{\text{MIX}}$ . The unlabeled dataset  $\mathcal{D}^{\text{MIX}}$  may contain a mix of  
 154 random, undesired, and expert demonstrations, and it will be used to support offline learning. Let  
 155  $\rho^{\text{UN}}$  be the occupancy measure (or stationary distribution) of the undesired policy (represented by the  
 156 undesired dataset).

157 Adapting the MaxEnt RL framework, the goal here is to find a reward function and a policy that  
 158 minimizes: (a) *the expected reward obtained by undesirable state, action pairs (part of undesirable*  
 159 *trajectories); and (b) negative of the expected reward obtained by state, action pairs not in undesirable*  
 160 *trajectories*. To this end, we introduce the following new learning objective:

$$\min_r \min_{\pi} \left\{ L(\pi, r) = \mathbb{E}_{\rho^{\text{UN}}}[r(s, a)] - \mathbb{E}_{\rho^{\pi}}[r(s, a)] - H(\pi, \mu) + \psi(r) \right\} \quad (4)$$

161 In our context, where the objective contrasts with the  
 162 standard learning-from-expert-demonstration scheme, the  
 163 learning problem is *no longer adversarial* as in prior imi-  
 164 tation learning approaches [18, 28]. Instead, it can be  
 165 framed as a *cooperative learning problem*, where the ob-  
 166 jective is to jointly identify a policy and reward function  
 167 that minimize the objective function  $L(\pi, r)$ . Solving (4)  
 168 directly encourages the learning policy to deviate as much  
 169 as possible from the undesired policy, which is derived  
 170 from undesirable demonstrations (we give a detailed dis-  
 171 cussion in appendix). This approach contrasts with that  
 172 of SafeDICE [20], which minimizes the KL divergence  
 173 between the learning policy and the mixed policy, which  
 174 could suffer from low quality mixing datasets.

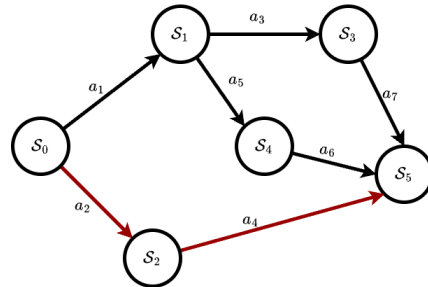


Figure 2: An illustrative MDP.

**Illustrative Example** We provide an illustrative example to demonstrate how the proposed objective guides the learning of a good policy from undesirable demonstrations. Consider the simple MDP shown in Figure 2. There are three trajectories from  $S_0$  to  $S_5$ :  $p_1 = [S_0, S_2, S_5]$ ,  $p_2 = [S_0, S_1, S_4, S_5]$ , and  $p_3 = [S_0, S_1, S_3, S_5]$ . We treat  $p_1$  as an undesirable trajectory, while  $p_2$  and  $p_3$  are considered optimal. The undesirable dataset is  $\mathcal{D}_{\text{und}} = \{p_1\}$ , and the unlabeled dataset includes all trajectories:  $\mathcal{D}_{\text{unlabeled}} = \{p_1, p_2, p_3\}$ . Under Equation (4), the training objective becomes:

$$\min_r \min_{\pi} \{R(p_1) - (P^{\pi}(p_1)R(p_1) + P^{\pi}(p_2)R(p_2) + P^{\pi}(p_3)R(p_3)) + \mathcal{H}(\pi)\},$$

where  $R(p)$  denotes the total accumulated reward along trajectory  $p$ , e.g.,  $R(p_1) = r(S_0, a_1) + r(S_1, a_3) + r(S_3, a_7)$  (ignoring discount factor  $\gamma$  for simplicity),  $P^{\pi}(p)$  is the probability of trajectory  $p$  under policy  $\pi$ , and  $\mathcal{H}(\pi)$  is the entropy regularization term. Simplifying the objective yields:

$$-P^{\pi}(p_2)R(p_2) - P^{\pi}(p_3)R(p_3) + (1 - P^{\pi}(p_1))R(p_1).$$

To minimize this expression, the algorithm is incentivized to assign higher rewards to  $p_2$  and  $p_3$ , and a lower reward to  $p_1$ . Consequently, to minimize the expected reward (note the negative coefficients), the optimal policy  $\pi$  should assign high probabilities to  $p_2$  and  $p_3$ , and a low probability to  $p_1$ . This encourages the policy to avoid undesirable behaviors and prefer optimal ones—even in the absence of direct expert demonstrations.

## 4.2 Learning in the Q-space

Directly solving Eq. 4 is not efficient. Instead, prior research indicates that transforming the reward learning problem into the Q-space and simplifying it can improve efficiency [12, 17]. As discussed in Section 3, there is a one-to-one mapping between any reward function  $r$  and a corresponding function  $Q$  in the Q-space. Therefore, the minimization problem in Eq. 4 can equivalently be transformed as:

$$\min_Q \min_{\pi} \left\{ L(\pi, Q) = \mathbb{E}_{\rho^{\text{UN}}}[\mathcal{T}^{\pi}[Q](s, a)] - \mathbb{E}_{\rho^{\pi}}[\mathcal{T}^{\pi}[Q](s, a)] - H(\pi) + \psi(\mathcal{T}^{\pi}[Q](s, a)) \right\} \quad (5)$$

where  $\mathcal{T}^{\pi}[Q](s, a) = Q(s, a) - \gamma \mathbb{E}_{s'}[V^{\pi}(s')]$  and  $V^{\pi}(s) = \mathbb{E}_{a \sim \pi(a|s)}[Q(s, a) - \beta \log \frac{\pi(a|s)}{\mu(a|s)}]$ . Compared to the primary objective of the standard MaxEnt framework [18, 12], our objective function in Eq. 5 is no longer adversarial with respect to  $Q$  and  $\pi$ .

Before we describe ways of simplifying objective in (5), we wish to highlight the properties that make IQ-learn and its variants efficient and effective:

**Property 1:** Closed-form optimization over  $\pi$ .

**Property 2:** Concavity of the simplified objective in the Q-space.

In comparison to imitation learning problem in the MaxEnt framework, since we negate the objective function, it alters its nature, turning a concave function into a convex one. We show that despite this change, property 1 is still preserved in the new objective. However, the simplified objective obtained by using property 1 is neither convex nor concave.

We first focus on **Property 1**, where we provide a closed form expression for policy,  $\pi^Q$ . If the regularizer function  $\psi(\cdot)$  is non-decreasing, then the objective function  $L(\pi, Q)$  is convex in  $\pi$ . Furthermore, the minimization problem  $\min_{\pi} L(\pi, Q)$  retains a closed-form solution, thereby simplifying the learning objective as shown in Proposition 4.1.

**Proposition 4.1.** *The function  $L(\pi, Q)$  is convex in  $\pi$  and the problem  $\min_{\pi} L(\pi, Q)$  has a unique optimal solution at  $\pi^Q(s, a) = \frac{\exp(Q(s, a)/\beta)}{\sum_{a'} \exp(Q(s, a')/\beta)}$ . Moreover, the learning objective function can be simplified as:  $\min_Q \min_{\pi} \{L(\pi, Q)\} = \min_Q \mathcal{F}(Q)$ , where*

$$\mathcal{F}(Q) = \mathbb{E}_{\rho^{\text{UN}}}[r^Q(s, a)] - (1 - \gamma) \mathbb{E}_{s_0}[V^Q(s_0)] + \psi(r^Q),$$

$$V^Q(s) = \beta \log(\sum_a \mu(a|s) \exp(Q(s, a)/\beta)) \text{ and } r^Q(s, a) = Q(s, a) - \gamma \mathbb{E}_{s' \sim P(\cdot|s, a)} V^Q(s').$$

Proposition 4.1 shows that, similar to standard inverse Q-learning algorithms, our training objective can be framed as an optimization problem over the Q-space, where the optimal policy can be computed as the soft-max of the Q-function. Thereby preserving **Property 1**.

However with regards to **Property 2**, while in inverse Q-learning with expert demonstrations, the training objective is convex within the Q-space, this is not the case in our context, where the objective  $\mathcal{F}(Q)$  is neither convex nor concave in  $Q$ . We state this observation as follows:

220 **Proposition 4.2.** *The learning objective function  $F(Q)$  is not convex in  $Q$ .*

221 This non-convexity arises from the presence of the term  $-V^Q$  in the objective function, where  $V^Q$  is  
 222 a log-sum-exp of  $Q$ , which is itself convex in  $Q$ . This directly implies that  $F(Q)$  is not convex in  $Q$ .  
 223 Such non-convexity is a significant disadvantage compared to prior inverse Q-learning approaches  
 224 that learn from expert demonstrations. In the following, we leverage the Extreme Q-learning method  
 225 [13] to reformulate the training as updates based on convex loss functions, effectively mitigating the  
 226 challenges associated with non-convexity.

### 227 4.3 Extreme Q-learning and Convexity

228 In the above formulations, the optimal value function  $V^Q$  can be expressed as a log-sum-exp of the Q-  
 229 function, i.e.,  $V^Q(s) = \beta \log(\sum_a \mu(a|s) \exp(Q(s, a)/\beta))$ . This log-sum-exp computation is often  
 230 impractical as  $\mu(a|s)$  is not explicitly known and the action space would be large. Furthermore, as  
 231 discussed earlier, it is the primary cause of the non-convexity in our learning objective function. To ad-  
 232 dress this issue, we leverage the Extreme Q-Learning (XQL) method [13] to update  $V$  simultaneously  
 233 with  $Q$ -function. Specifically, we define the Extreme-V function as:  $J(V|Q) = \mathbb{E}_{(s,a) \sim \mu} e^t - (t) - 1$ ,  
 234 where  $t = \frac{Q(s,a) - V(s)}{\beta}$  where it can be shown that minimizing  $J(V|Q)$  over  $V$  yields an optimal  
 235 solution  $V^*(s)$  such that:  $V^*(s) = \beta \log(\sum_a \mu(a|s) e^{Q(s,a)/\beta})$ . We then define the Q-learning  
 236 function conditional on  $V$  as  $\tilde{F}(Q|V)$ , which is defined as

$$\tilde{F}(Q|V) = \mathbb{E}_{\rho^{\text{UN}}} [Q(s, a) - \gamma \mathbb{E}_{s'} [V(s')]] - (1 - \gamma) \mathbb{E}_{s_0} [V(s_0)] + \psi(Q(s, a) - \gamma \mathbb{E}_{s'} [V(s')]).$$

237 The minimization of our learning objective  $F(Q)$  can be achieved through the following alternating  
 238 updates: (i) **Update  $Q$** : Minimize the objective  $\tilde{F}(Q|V)$ , and (ii) **Update  $V$** : Minimize the Extreme-  
 239 V function  $J(V|Q)$ . This procedure is particularly convenient as both updates can be shown to have  
 240 favorable convexity properties. Specifically, we demonstrate that the loss functions are convex in  
 241 their respective spaces:

242 **Proposition 4.3.** *Under any convex regularizer  $\psi$ , both  $\tilde{F}(Q|V)$  and  $J(V|Q)$  are **convex** in  $Q$  and  
 243  $V$ , respectively.*

### 244 4.4 Unbiased Q-Learning with Unlabeled Data

245 We now discuss the optimization in the Q-space. Directly solving  $\min_Q \tilde{F}(Q|V)$  using only the  
 246 undesirable dataset would be inefficient due to its small size (as shown in Appendix E.5). Furthermore,  
 247 directly combining the undesirable and unlabeled datasets for training is risky, as the unknown quality  
 248 of the unlabeled dataset could lead to learning in the wrong direction. For instance, the model could  
 249 unintentionally mimic low-quality demonstrations, as might occur in methods like SafeDICE. To  
 250 overcome this challenge, we address the following critical question: **How can we unbiasedly estimate**  
 251 **the objective  $\min_Q \tilde{F}(Q|V)$  while effectively utilizing the unlabeled dataset?**

252 To address this question, we leverage the larger set of unlabeled data  $\mathcal{D}^{\text{MIX}}$  to enhance the offline  
 253 training. To achieve this, we first let  $\rho^{\text{MIX}}$  be the occupancy measure (or stationary distribution) of the  
 254 policy represented by unlabeled dataset. We rewrite the expectation over  $\rho^{\text{UN}}$  as:  $\mathbb{E}_{\rho^{\text{UN}}} [r^Q(s, a)] =$   
 255  $\mathbb{E}_{\rho^{\text{MIX}}} [\tau(s, a) r^Q(s, a)]$ , where  $\tau(s, a) = \frac{\rho^{\text{UN}}(s, a)}{\rho^{\text{MIX}}(s, a)}$  represents the occupancy ratio between  $\rho^{\text{UN}}$  and  
 256  $\rho^{\text{MIX}}$ , we then rewrite the learning objective as follows:

$$\min_Q \left\{ \tilde{F}(Q|V) = \mathbb{E}_{\rho^{\text{MIX}}} [\tau(s, a) r^Q(s, a)] - (1 - \gamma) \mathbb{E}_{s_0} [V(s_0)] + \psi(r^Q) \right\} \quad (6)$$

257 where  $r^Q(s, a) = Q(s, a) - \gamma \mathbb{E}_{s'} [V(s')]$ . In this approach, the expectation  $\mathbb{E}_{\rho^{\text{MIX}}} [\tau(s, a) r^Q(s, a)]$   
 258 can be empirically approximated using the unlabeled samples from  $\mathcal{D}^{\text{MIX}}$ , where  $\tau(s, a)$  acts as an  
 259 occupancy correction. This correction allows us to leverage samples from the unlabeled dataset to  
 260 unbiasedly estimate the expectation over the undesirable policy. A key challenge here is that the  
 261 occupancy ratio  $\tau(s, a)$  is unknown. To address this, we propose estimating the ratio by solving the  
 262 following implicit maximization problem:

$$\max_{\vartheta: S \times A \rightarrow [0,1]} \{g(\vartheta)\} \quad (7)$$

where  $g(\vartheta) = \mathbb{E}_{\rho^{\text{MIX}}} [\log(1 - \vartheta(s, a))] + \mathbb{E}_{\rho^{\text{UN}}} [\log(\vartheta(s, a))]$ . The above formulation is similar to the discriminator-based formulations widely used in prior adversarial imitation learning work [18, 23]. In fact, the objective function  $g(\vartheta)$  is strictly concave in  $\vartheta$ . Let  $\vartheta^*$  denote the unique optimal solution to (7), then the occupancy ratio can be estimated as  $\tau(s, a) = \frac{\vartheta^*(s, a)}{1 - \vartheta^*(s, a)}$ .

So, our learning process can be broken down into two steps. In the first step, we learn the occupancy ratios by solving the maximization problems presented in 7. Following this, we optimize the following problem to learn  $Q$  and  $V$  functions.

$$\min_Q \left\{ \tilde{F}(Q|V) = \mathbb{E}_{\rho^{\text{MIX}}} \left[ \frac{\vartheta^*(s, a)}{1 - \vartheta^*(s, a)} r^Q(s, a) \right] - (1 - \gamma) \mathbb{E}_{s_0} [V(s_0)] + \psi(r^Q) \right\} \quad (8)$$

It is important to note that we utilize the stationary distribution  $\rho^{\text{MIX}}$  (represented by trajectories in the unlabeled dataset) in the objective function in (8). However, thanks to the occupancy correction  $\frac{\vartheta^*(s, a)}{1 - \vartheta^*(s, a)}$ , the outcome of the training is theoretically independent of the quality of the unlabeled policy  $\rho^{\text{MIX}}$ . This distinguishes our approach from SafeDICE [20], where the performance heavily relies on the quality of the unlabeled data.

#### 4.5 Practical Implementation

Our algorithm consists of two main steps. In the first step, we construct a network,  $\vartheta_\phi$ , where  $\phi$  are learnable parameters. We then use samples from  $\mathcal{D}^{\text{UN}}$  and  $\mathcal{D}^{\text{MIX}}$  to estimate the objective function  $g(\vartheta)$ . In the second step, after obtaining  $\phi$  from the first step, we calculate the occupancy correction  $\tau^*$  and use it to update the  $Q$  and  $V$  functions as described in Section 4.3. For the  $Q$ -updates, we utilize the following empirical objective:  $\tilde{F}(Q|V) = \sum_{(s, a, s') \sim \mathcal{D}^{\text{MIX}}} [\tau^*(s, a) r^Q(s, a, s') + \psi(r^Q(s, a, s'))] - (1 - \gamma) \sum_{s^0 \sim \mathcal{D}^{\text{MIX}}} V(s)$ , where the reward function is computed as  $r^Q(s, a, s') = Q(s, a) - \gamma V^Q(s')$ . We choose the  $\chi^2$ -divergence for the reward regularizer  $\psi(t) = 0.1t^2 - t$ , a popular choice in inverse Q-learning algorithm. For policy extraction we utilize the following practical weighted behavior cloning (WBC) with the objective:  $\max_\pi \left\{ \sum_{(s, a) \sim \mu} \exp(A(s, a)) \log \pi(a|s) \right\}$ , where  $A(s, a)$  is the advantage function defined as  $A(s, a) = (Q(s, a) - V(s))/\beta$  [37, 35].

### 5 Experiments

We evaluate our algorithm in the context of unconstrained RL and safe RL (i.e., constrained RL) using the Mujoco [42] and Safety-Gym [38, 21] domains, respectively. In the unconstrained task, we aim to test whether our algorithm can identify undesirable behaviors (from random or low-reward demonstrations) and avoid them. In the safety learning task, we focus on assessing whether our algorithm can satisfy safety constraints while maintaining reasonable rewards. Safe RL is also the main focus of SafeDICE in their experiments.

#### 5.1 Experiment Setting

**Baselines.** We compare our algorithm against several baseline methods: (i) **BC**, which learns from the entire unlabeled dataset (BC-mix); (ii) **IPL** [17], a state-of-the-art algorithm in preference based RL; (iii) **DWBC** [45] and (iv) **LS-IQ** [3], two leading IL algorithms for learning from expert demonstrations, adapted to avoid undesirable demonstrations; and (v) **SafeDICE** [20], specifically designed for learning from undesirable demonstrations. We also include comparisons with modified baselines (similar to DWBC); however, due to space constraints, these are presented in Appendix E.7. Detailed implementation of these baselines is provided in Appendix D.3.

**Environments and Dataset Generation.** For our experiments, we define the components of the undesirable dataset  $\mathcal{D}^{\text{UN}}$  and the unlabeled dataset  $\mathcal{D}^{\text{MIX}}$ . In the Mujoco experiments, we use the official D4RL dataset [10], which consists of three performance levels: random, medium, and expert. We combine the random and medium levels to create the undesirable dataset  $\mathcal{D}^{\text{UN}}$  and use all three performance levels to construct the unlabeled dataset  $\mathcal{D}^{\text{MIX}}$ . In the safety learning experiments, we train two policies: a constrained policy (using PPO-Lagrangian [38] to achieve low cost performance) and an unconstrained policy (using PPO [41] maximizes the return while ignoring the cost signals). Data is collected from both policies, with the unconstrained policy is used to generate high-cost

demonstrations for undesirable dataset  $\mathcal{D}^{\text{UN}}$ , and both datasets are combined to form the unlabeled dataset  $\mathcal{D}^{\text{MIX}}$ . More details about the datasets can be found in Appendix D.2.

**Evaluation Metrics.** For the Mujoco domain, a higher return indicates better performance. In the Safety-Gym domain, the ideal outcome is to achieve the lowest cost possible while not significantly sacrificing the return. All experiments are run with at least 5 training seeds.

**Experiment Concerns.** Throughout the experiments, we aim to address several key questions: **(Q1)** *How does UNIQ perform compared to other baselines in the context of unconstrained and safe RL?* **(Q2)** *How does the presence of an undesirable dataset contribute to the performance of UNIQ and other baselines?* **(Q3)** *How does the quality of unlabeled dataset affect the performance of all the algorithms?* **(Q4)** *What happens if we do not use the unlabeled dataset?* **(Q5)** *We use WBC for the policy extraction; what if we directly extract the policy from the Q-function?* In the main paper, we provide experiments for **(Q1)**, **(Q2)**, and **(Q3)**, while the other questions and some additional experiments are addressed in the appendix.

## 5.2 Main Comparison

		BC-mix	IPL	LS-IQ	DWBC	SafeDICE	UNIQ	Expert
Halfcheetah	Return $\uparrow$	17.9 $\pm$ 6.3	3.2 $\pm$ 0.7	-2.8 $\pm$ 0.3	2.2 $\pm$ 0.0	3.1 $\pm$ 0.9	<b>75.7</b> $\pm$ 6.8	84.6 $\pm$ 7.5
Ant	Return $\uparrow$	86.3 $\pm$ 11.8	84.5 $\pm$ 13.1	1.8 $\pm$ 5.6	84.6 $\pm$ 12.4	4.4 $\pm$ 2.6	<b>104.4</b> $\pm$ 10.5	117.5 $\pm$ 11.0
Hopper	Return $\uparrow$	3.7 $\pm$ 2.9	11.3 $\pm$ 14.3	1.0 $\pm$ 0.3	14.8 $\pm$ 15.7	51.0 $\pm$ 3.1	<b>73.5</b> $\pm$ 20.6	110.8 $\pm$ 1.2
Walker2d	Return $\uparrow$	10.7 $\pm$ 8.7	11.8 $\pm$ 24.2	-0.1 $\pm$ 0.1	38.8 $\pm$ 24.7	41.7 $\pm$ 4.1	<b>105.9</b> $\pm$ 4.0	107.9 $\pm$ 0.9
Point-Goal	Return $\uparrow$	27.1 $\pm$ 0.1	26.9 $\pm$ 0.1	-6.8 $\pm$ 4.4	26.9 $\pm$ 0.1	27.0 $\pm$ 0.1	23.4 $\pm$ 0.4	25.9 $\pm$ 0.2
	Cost $\downarrow$	48.8 $\pm$ 2.9	52.7 $\pm$ 3.4	18.0 $\pm$ 29.3	45.8 $\pm$ 3.4	46.8 $\pm$ 3.1	<b>27.1</b> $\pm$ 3.0	26.0 $\pm$ 2.6
Car-Goal	Return $\uparrow$	34.1 $\pm$ 0.5	34.7 $\pm$ 0.3	-0.6 $\pm$ 2.5	32.8 $\pm$ 0.7	33.5 $\pm$ 0.7	27.9 $\pm$ 0.8	26.2 $\pm$ 0.7
	Cost $\downarrow$	52.0 $\pm$ 4.2	54.4 $\pm$ 3.7	58.5 $\pm$ 41.7	47.4 $\pm$ 3.8	50.5 $\pm$ 4.0	<b>31.0</b> $\pm$ 2.8	23.6 $\pm$ 2.8
Point-Button	Return $\uparrow$	17.6 $\pm$ 0.7	16.9 $\pm$ 0.9	-13.3 $\pm$ 6.7	17.2 $\pm$ 0.9	15.1 $\pm$ 0.5	12.6 $\pm$ 1.4	14.4 $\pm$ 1.0
	Cost $\downarrow$	120.2 $\pm$ 10.0	124.8 $\pm$ 11.3	11.0 $\pm$ 10.2	123.5 $\pm$ 14.4	91.0 $\pm$ 6.4	<b>23.0</b> $\pm$ 4.7	30.6 $\pm$ 3.3
Car-Button	Return $\uparrow$	17.6 $\pm$ 0.7	17.2 $\pm$ 0.8	-8.6 $\pm$ 4.8	17.1 $\pm$ 1.0	17.4 $\pm$ 0.6	12.7 $\pm$ 1.1	14.0 $\pm$ 0.8
	Cost $\downarrow$	241.6 $\pm$ 15.3	257 $\pm$ 12.6	28.0 $\pm$ 27.0	249.2 $\pm$ 20.9	201.3 $\pm$ 10.8	<b>148.6</b> $\pm$ 18.7	107.6 $\pm$ 8.9

Table 1: Comparison results for Mujoco (i.e. unconstrained RL) and Safety-gym (i.e. constrained RL) tasks.

In this section, we aim to answer question **(Q1)**, comparing UNIQ with the mentioned baselines across four Mujoco tasks and four Safety-Gym tasks. For the Mujoco tasks, we use 5 trajectories each from the random and medium datasets to construct the undesirable dataset  $\mathcal{D}^{\text{UN}}$ , and 500, 500, and 100 trajectories from the random, medium, and expert datasets, respectively, to construct the unlabeled dataset  $\mathcal{D}^{\text{MIX}}$ . For the Safety-Gym tasks, we use 100 high-cost trajectories for the undesirable dataset  $\mathcal{D}^{\text{UN}}$  and combine 1600 high-cost and 400 low-cost trajectories for the unlabeled dataset  $\mathcal{D}^{\text{MIX}}$ . To facilitate comparison, we provide the performance of policies learned from desirable data (the expert dataset in Mujoco and the constrained dataset in Safety-Gym), which serve as the baseline for the highest performance. We note that our experiments require more samples than standard learning-from-expert experiments. This is intuitive, as *learning from mistakes* typically requires more examples than learning from good demonstrations.

The experimental results are shown in Table 1. Overall, BC-mix, LS-IQ, and IPL fail to meet our objectives (except in the Ant environment, where BC using all datasets seems to achieve good performance). These experiment results also show that the state-of-the-art algorithm in imitation learning, LS-IQ, does not perform well in the “avoid bad” scenario. DWBC outperforms BC-mix in Mujoco by using a discriminator to understand the undesirable dataset but struggles in the Safety-Gym tasks. Meanwhile, SafeDICE slightly outperforms DWBC in the Safety-Gym context. In general, UNIQ is the highest-performing algorithm in the Mujoco tasks and achieves closest (both in terms of return and costs) performance to policies learned from only the expert demonstrations (reported in “Expert” Column).



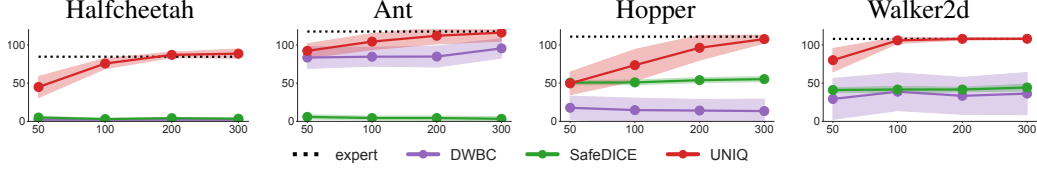


Figure 3: Comparison of results (MuJoCo tasks) for different sizes of expert demonstrations included in the unlabeled dataset, shaded by standard deviation.

### 5.3 Ablation Studies

**Impact of the Quality of the Unlabeled Dataset.** To address **Question (Q3)**, we modify the quality of the unlabeled dataset  $\mathcal{D}^{\text{Mix}}$  by adjusting the number of desirable demonstrations while keeping the same  $\mathcal{D}^{\text{UN}}$  as Section 5.2. The detailed results are shown in Figure 3. Overall, UNIQ consistently achieves the highest performance and adapts effectively to variations in dataset quality. The corresponding training curves for Mujoco tasks and Safety-gym tasks can be found in Appendix E.1.

**Impact of the Size of the Undesirable Dataset.** We address **(Q2)**: *the impact of the undesirable dataset on the final performance*. Detailed results are presented in Figure 4, where the  $x$ -axis represents the number of undesirable trajectories in  $\mathcal{D}^{\text{UN}}$  and keeping the same  $\mathcal{D}^{\text{Mix}}$  as Section 5.2. Overall, the size of the dataset significantly influences the final cost across all algorithms. As the size of the undesirable dataset increases, UNIQ achieves the lowest cost, albeit with a trade-off in return. Detailed learning curves for Mujoco and Safety-Gym can be found in the Appendix E.3. Furthermore, we discuss methods to control the conservativeness (trade-off between return and cost) of our approach in the Appendix E.10.

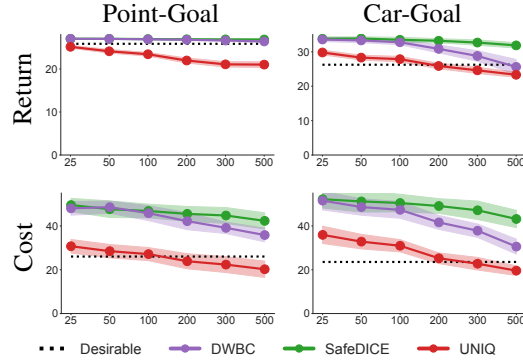


Figure 4: Comparison results for different sizes of the undesirable dataset in Safety gym, shaded by standard deviation.

## 6 Conclusion, Future Work

**Conclusion.** We have developed UNIQ, a principled framework designed to facilitate learning from undesirable demonstrations. UNIQ formulates the learning process as updates on  $Q$ - and  $V$ -functions based on convex loss functions, enabling unbiased training using unlabeled datasets of unknown quality. Additionally, UNIQ requires minimal hyper-parameter tuning, as it does not introduce any additional hyperparameters beyond those typically used in inverse  $Q$ -learning algorithms. Furthermore, UNIQ demonstrates superior performance in generating safe policies across several safe reinforcement learning experiments, outperforming other baseline methods.

**Limitations and Future work.** There are some aspects that have not been addressed in this paper, as they are too significant to be fully explored here. For instance, we assume the presence of only one set of undesirable demonstrations, whereas multiple datasets of varying quality could be leveraged to enhance the training. Additionally, each undesirable trajectory may not be undesirable in its entirety, as it could contain some good actions. Extracting the good parts from undesirable demonstrations could improve sample efficiency but introduces new challenges that warrant further investigation.

## References

- [1] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1, 2004.
- [2] Firas Al-Hafez, Davide Tateo, Oleg Arenz, Guoping Zhao, and Jan Peters. Ls-iq: Implicit reward regularization for inverse reinforcement learning. In *Eleventh International Conference on Learning Representations (ICLR)*, 2023.
- [3] Firas Al-Hafez, Davide Tateo, Oleg Arenz, Guoping Zhao, and Jan Peters. Ls-iq: Implicit reward regularization for inverse reinforcement learning. In *ICLR*, 2023.
- [4] Oleg Arenz and Gerhard Neumann. Non-adversarial imitation learning and its connections to adversarial methods. *arXiv preprint arXiv:2008.03525*, 2020.
- [5] Claudine Badue, R nik Guidolini, Raphael Vivacqua Carneiro, Pedro Azevedo, Vinicius B Cardoso, Avelino Forechi, Luan Jesus, Rodrigo Berriel, Thiago M Paixao, Filipe Mutz, et al. Self-driving cars: A survey. *Expert systems with applications*, 165:113816, 2021.
- [6] Mayank Bansal, Alex Krizhevsky, and Abhijit Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *arXiv preprint arXiv:1812.03079*, 2018.
- [7] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.
- [8] Daniel Brown, Wonjoon Goo, Prabhat Nagarajan, and Scott Niekum. Extrapolating beyond sub-optimal demonstrations via inverse reinforcement learning from observations. In *International conference on machine learning*, pages 783–792. PMLR, 2019.
- [9] Robert Dadashi, Leonard Hussenot, Matthieu Geist, and Olivier Pietquin. Primal wasserstein imitation learning. In *International Conference on Learning Representations*, 2021.
- [10] Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning, 2020.
- [11] Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adversarial inverse reinforcement learning. In *International Conference on Learning Representations*, 2018.
- [12] Divyansh Garg, Shuvam Chakraborty, Chris Cundy, Jiaming Song, and Stefano Ermon. Iq-learn: Inverse soft-q learning for imitation. *Advances in Neural Information Processing Systems*, 34:4028–4039, 2021.
- [13] Divyansh Garg, Joey Hejna, Matthieu Geist, and Stefano Ermon. Extreme q-learning: Maxent rl without entropy. In *International Conference on Learning Representations (ICLR)*, 2023.
- [14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.
- [15] Omer Gottesman, Fredrik Johansson, Matthieu Komorowski, Aldo Faisal, David Sontag, Finale Doshi-Velez, and Leo Anthony Celi. Guidelines for reinforcement learning in healthcare. *Nature medicine*, 25(1):16–18, 2019.
- [16] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.
- [17] Joey Hejna and Dorsa Sadigh. Inverse preference learning: Preference-based rl without a reward function. *Advances in Neural Information Processing Systems*, 36, 2024.
- [18] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29, 2016.
- [19] Huy Hoang, Tien Mai, and Pradeep Varakantham. Springl: Sub-optimal demonstrations driven offline imitation learning, 2024.

- [20] Youngsoo Jang, Geon-Hyeong Kim, Jongmin Lee, Sungryull Sohn, Byoungjip Kim, Honglak Lee, and Moontae Lee. Safedice: offline safe imitation learning with non-preferred demonstrations. *Advances in Neural Information Processing Systems*, 36, 2024.
- [21] Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Josef Dai, and Yaodong Yang. Safety gymnasium: A unified safe reinforcement learning benchmark. *Advances in Neural Information Processing Systems*, 36, 2023.
- [22] Yachen Kang, Diyuang Shi, Jinxin Liu, Li He, and Donglin Wang. Beyond reward: Offline preference-guided policy optimization. In *International Conference on Machine Learning*, pages 15753–15768. PMLR, 2023.
- [23] Michael Kelly, Chelsea Sidrane, Katherine Driggs-Campbell, and Mykel J Kochenderfer. Hg-dagger: Interactive imitation learning with human experts. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8077–8083. IEEE, 2019.
- [24] Changyeon Kim, Jongjin Park, Jinwoo Shin, Honglak Lee, Pieter Abbeel, and Kimin Lee. Preference transformer: Modeling human preferences using transformers for rl. In *The Eleventh International Conference on Learning Representations*, 2023.
- [25] Geon-Hyeong Kim, Jongmin Lee, Youngsoo Jang, Hongseok Yang, and Kee-Eung Kim. Lobsdice: Offline learning from observation via stationary distribution correction estimation. *Advances in Neural Information Processing Systems*, 35:8252–8264, 2022.
- [26] Geon-Hyeong Kim, Seokin Seo, Jongmin Lee, Wonseok Jeon, HyeongJoo Hwang, Hongseok Yang, and Kee-Eung Kim. Demodice: Offline imitation learning with supplementary imperfect demonstrations. In *International Conference on Learning Representations*, 2021.
- [27] Ryuichi Kiryo, Gang Niu, Marthinus C Du Plessis, and Masashi Sugiyama. Positive-unlabeled learning with non-negative risk estimator. *Advances in neural information processing systems*, 30, 2017.
- [28] Ilya Kostrikov, Ofir Nachum, and Jonathan Tompson. Imitation learning via off-policy distribution matching. In *International Conference on Learning Representations*, 2019.
- [29] Ilya Kostrikov, Ofir Nachum, and Jonathan Tompson. Imitation learning via off-policy distribution matching. In *International Conference on Learning Representations*, 2020.
- [30] Hoang Le, Nan Jiang, Alekh Agarwal, Miroslav Dudík, Yisong Yue, and Hal Daumé III. Hierarchical imitation and reinforcement learning. In *International conference on machine learning*, pages 2917–2926. PMLR, 2018.
- [31] K Lee, L Smith, A Dragan, and P Abbeel. B-pref: Benchmarking preference-based reinforcement learning. *Neural Information Processing Systems (NeurIPS)*, 2021.
- [32] Kimin Lee, Laura M Smith, and Pieter Abbeel. Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. In *International Conference on Machine Learning*, pages 6152–6163. PMLR, 2021.
- [33] Yecheng Ma, Andrew Shen, Dinesh Jayaraman, and Osbert Bastani. Versatile offline imitation from observations and examples via regularized state-occupancy matching. In *International Conference on Machine Learning*, pages 14639–14663. PMLR, 2022.
- [34] Ofir Nachum, Bo Dai, Ilya Kostrikov, Yinlam Chow, Lihong Li, and Dale Schuurmans. Algaedice: Policy gradient from arbitrary experience. *arXiv preprint arXiv:1912.02074*, 2019.
- [35] Ashvin Nair, Abhishek Gupta, Murtaza Dalal, and Sergey Levine. Awac: Accelerating online reinforcement learning with offline datasets. *arXiv preprint arXiv:2006.09359*, 2020.
- [36] Gergely Neu, Anders Jonsson, and Vicenç Gómez. A unified view of entropy-regularized markov decision processes. *arXiv preprint arXiv:1705.07798*, 2017.
- [37] Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019.

- 472 [38] Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep rein-  
473 forcement learning. *arXiv preprint arXiv:1910.01708*, 7:1, 2019.
- 474 [39] Siddharth Reddy, Anca D Dragan, and Sergey Levine. Sqil: Imitation learning via reinforcement  
475 learning with sparse rewards. *arXiv preprint arXiv:1905.11108*, 2019.
- 476 [40] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and  
477 structured prediction to no-regret online learning. In *Proceedings of the fourteenth interna-*  
478 *tional conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and  
479 Conference Proceedings, 2011.
- 480 [41] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal  
481 policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- 482 [42] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based  
483 control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages  
484 5026–5033. IEEE, 2012.
- 485 [43] Faraz Torabi, Garrett Warnell, and Peter Stone. Behavioral cloning from observation. *arXiv*  
486 *preprint arXiv:1805.01954*, 2018.
- 487 [44] Leonid Nisonovich Vaserstein. Markov processes over denumerable products of spaces, de-  
488 scribing large systems of automata. *Problemy Peredachi Informatsii*, 5(3):64–72, 1969.
- 489 [45] Haoran Xu, Xianyuan Zhan, Honglei Yin, and Huiling Qin. Discriminator-weighted offline  
490 imitation learning from suboptimal demonstrations. In *Proceedings of the 39th International*  
491 *Conference on Machine Learning*, pages 24725–24742, 2022.
- 492 [46] Lantao Yu, Tianhe Yu, Jiaming Song, Willie Neiswanger, and Stefano Ermon. Offline imitation  
493 learning with suboptimal demonstrations via relaxed distribution matching. In *Proceedings of*  
494 *the AAAI conference on artificial intelligence*, volume 37, pages 11016–11024, 2023.
- 495 [47] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, et al. Maximum entropy  
496 inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

497 **Appendix**

498 The appendix contains the following details:

499 **Missing Proofs:** Refer to Appendix B for proofs omitted from the main paper.

500 **Experimental Details:** We provide information on:

- 501 • Baseline implementation (Appendix D.3)
- 502 • Hyper-parameter selection for each task domain (Appendix D.4)
- 503 • Task descriptions (Appendix D.1)
- 504 • Generation of undesirable and unlabeled datasets (Appendix D.2)

505 **Additional Experiments:** We address the following remaining questions:

- 506 • **(Q2)** How does the presence of an undesirable dataset contribute to the performance of  
507 UNIQ and other baselines? (Appendix E.1)
- 508 • **(Q3)** How does the quality of unlabeled dataset affect the performance of all the algorithms?  
509 (Appendix E.2 and Appendix E.3)
- 510 • **(Q4)** What happens if we do not use the unlabeled dataset (the training is solely based on  
511 the undesirable dataset)? (Appendix E.5)
- 512 • **(Q5)** We use WBC for the policy extraction; what if we directly extract the policy from the  
513 Q-function? (Appendix E.4)

514 Additionally, we present experiments demonstrating the performance of UNIQ and other baseline  
515 methods on the dataset from the safeDICE paper (see Appendix E.12), performance in the Mujoco-  
516 velocity benchmark(see Appendix E.11), comparison results using CVaR costs (see Appendix E.9),  
517 How does  $\tau$  effect to the overall results? (Appendix E.6), What if we want to avoid all possible  
518 undesirable performance in Safetygym (low-reward low-cost, low-reward high-cost, high-reward  
519 high-cost)? (Appendix E.8), and how to control the conservativeness in the Safety-gym tasks(see  
520 Appendix E.10).

## 521 A Pseudo Code of UNIQ

Below we present a pseudo code of our UNIQ algorithm.

---

### Algorithm 1 UNIQ: UNdesired Demonstrations driven Inverse Q-Learning

---

**Require:**  $\mathcal{D}^{\text{UN}}, \mathcal{D}^{\text{MIX}}, \vartheta_\phi, \pi_\theta, N_\mu, N, Q_{w_q}$  and  $V_{w_v}$  networks.  
1: **# Estimating the occupancy correction  $\tau^*$**   
2: **for** certain number of iterations:  $i = 1 \dots N_\mu$  **do**  
3:   Update  $(\phi)$  to maximize  $g(\vartheta_\phi)$ .  
4: **end for**  
5: **# Train Q and V, and policy functions**  
6: **for** certain number of iterations  $i = 1 \dots N$  **do**  
7:   Update  $w_q$  to minimize  $\tilde{F}(Q_{w_q} | V_{w_v})$   
8:   Update  $w_v$  to minimize the Extreme-V function:  $J(V_{w_v} | Q_{w_q})$   
9:   Update  $\theta$  via the WBC:  $\max_\pi \left\{ \sum_{(s,a) \sim \mathcal{D}^{\text{MIX}}} \exp(A(s,a)) \log \pi(a|s) \right\}$   
10: **end for**

---

522

## 523 B Missing Proofs

524 We provide proofs that are omitted in the main paper.

### 525 B.1 Proof of Proposition 4.1

526 **Proposition.** *The following statements hold:*

- 527 (i) *The function  $L(\pi, Q)$  is convex in  $\pi$  and the problem  $\min_\pi L(\pi, Q)$  has a unique optimal*  
528 *solution at  $\pi^Q(s, a) = \frac{\exp(Q(s,a)/\beta)}{\sum_{a'} \exp(Q(s,a')/\beta)}$ .*  
529 (ii) *The learning objective function can be simplified as:*

$$\min_Q \min_\pi \{L(\pi, Q)\} = \min_Q \{\mathcal{F}(Q) = \mathbb{E}_{\rho^{\text{UN}}}[r^Q(s, a)] - (1 - \gamma)\mathbb{E}_{s_0}[V^Q(s_0)] + \psi(r^Q)\}$$

530 where  $V^Q(s) = \beta \log(\sum_a \mu(a|s) \exp(Q(s, a)/\beta))$  and  $r^Q(s, a) = Q(s, a) -$   
531  $\gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} V^Q(s')$ .

532 *Proof.* We first express the second and third terms of the objective function  $L(\pi, Q)$  in 5 as:

$$\mathbb{E}_{\rho_\pi}[\mathcal{T}^\pi[Q](s, a)] + H(\pi) = \mathbb{E}_{\rho_\pi}[Q(s, a) - \gamma \mathbb{E}_{s'}[V^\pi(s')]] - \beta \mathbb{E}_{\rho_\pi}[\log \frac{\pi(s, a)}{\mu(a|s)}]$$

533

$$= \mathbb{E}_{\rho_\pi}[Q(s, a) - \beta \log \frac{\pi(s, a)}{\mu(a|s)} - \gamma \mathbb{E}_{s'}[V^\pi(s')]] = \mathbb{E}_{\rho_\pi}[V(s) - \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}[V^\pi(s')]]$$

534

$$= (1 - \gamma)\mathbb{E}_{s_0 \sim P_0}[V^\pi(s_0)].$$

535 Thus, the objective function becomes:

$$L(\pi, Q) = \mathbb{E}_{\rho^{\text{UN}}}[Q(s, a) - \gamma \mathbb{E}_{s'}[V^\pi(s')]] - (1 - \gamma)\mathbb{E}_{s_0 \sim P_0}[V^\pi(s_0)] + \sum_{s,a} \psi(Q(s, a) - \gamma \mathbb{E}_{s'}[V^\pi(s')]).$$

536 We now observe that  $V^\pi(s) = \mathbb{E}_{a \sim \pi(a|s)}[Q(s, a) - \beta \log \frac{\pi(a|s)}{\mu(a|s)}]$  is concave in  $\pi$ . Therefore, both  
537 terms  $\mathbb{E}_{\rho^{\text{UN}}}[Q(s, a) - \gamma \mathbb{E}_{s'}[V^\pi(s')]]$  and  $-(1 - \gamma)\mathbb{E}_{s_0 \sim P_0}[V^\pi(s_0)]$  are convex in  $\pi$ . Additionally,  
538 since  $\psi(t)$  is convex and non-increasing in  $t$ , and  $Q(s, a) - \gamma \mathbb{E}_{s'}[V^\pi(s')]$  is convex in  $\pi$ , each  
539 function  $\psi(Q(s, a) - \gamma \mathbb{E}_{s'}[V^\pi(s')])$  is convex in  $\pi$ . Thus, combining all terms, we conclude that  
540  $L(\pi, Q)$  is convex in  $\pi$ .

541 Furthermore, each term  $Q(s, a) - \gamma \mathbb{E}_{s'}[V^\pi(s')]$ ,  $-(1 - \gamma)\mathbb{E}_{s_0 \sim P_0}[V^\pi(s_0)]$ , and  $\psi(Q(s, a) - \gamma \mathbb{E}_{s'}[V^\pi(s')])$   
542 strictly decreases in  $V^\pi$ , implying that the minimization of  $L(\pi, Q)$  over  $\pi$  is achieved when  $V^\pi(s)$

is maximized for all  $s$ . Since  $V^\pi(s)$  is strictly concave in  $\pi$ , maximizing  $V^\pi(s)$  over  $\pi$  has a unique optimal solution:

$$\pi^Q(a|s) = \frac{\exp(Q(s, a)/\beta)}{\sum_a \exp(Q(s, a)/\beta)}.$$

This validates part (i) of the theorem.

For part (ii), we observe that the problem  $\max_\pi V^\pi(s)$  has the optimal solution  $\pi^Q$  as shown above, and the optimal value is:

$$\begin{aligned} \max_\pi V^\pi(s) &= \max_\pi \left\{ \sum_a \pi(a|s) Q(s, a) - \beta \pi(a|s) \log \frac{\pi(a|s)}{\mu(a|s)} \right\} \\ &= \beta \log \left( \sum_a \mu(a|s) \exp(Q(s, a)/\beta) \right) \stackrel{\text{def}}{=} V^Q(s). \end{aligned}$$

This directly leads to:

$$\min_Q \min_\pi \{L(\pi, Q)\} = \min_Q \{F(Q) = \mathbb{E}_{\rho^{\text{UN}}} [r^Q(s, a)] - (1 - \gamma) \mathbb{E}_{s_0} [V^Q(s_0)] + \psi(r^Q)\},$$

as required. □

## B.2 Proof of Proposition 4.2

**Proposition:** *The learning objective function  $F(Q)$  is not convex in  $Q$ , even without any regularizer.*

*Proof.* We rewrite the function  $F(Q)$  (without the regularizer  $\psi(\cdot)$ ):

$$F(Q) = \mathbb{E}_{\rho^{\text{UN}}} [Q(s, a) - \gamma \mathbb{E}_{s'} [V^Q(s')]] - (1 - \gamma) \mathbb{E}_{s_0} [V^Q(s_0)],$$

where  $V^Q(s) = \beta \log (\sum_a \mu(a|s) e^{Q(s, a)/\beta})$  is a log-sum-exp function, which is convex in  $Q$ . Since  $V^Q(s)$  is convex in  $Q$ , all the components associated with  $V^Q(s)$  in  $F(Q)$  are concave in  $Q$ . This directly implies the non-convexity of  $F(Q)$ .

When the regularizer is included, it can be observed that if  $\psi(\cdot)$  is convex in  $Q$ , then  $\psi(r^Q(s, a))$  is also convex in  $Q$ . As a result, the objective function  $F(Q)$  takes the form of a difference-of-convex program. □

## B.3 Proof of Proposition 4.3

**Proposition:** *Under any convex regularizer  $\psi$ , both  $\tilde{F}(Q|V)$  and  $J(V|Q)$  are convex in  $Q$  and  $V$ , respectively.*

*Proof.* We first write the loss function for the  $Q$ -updates as:

$$\tilde{F}(Q|V) = \mathbb{E}_{\rho^{\text{UN}}} [Q(s, a) - \gamma \mathbb{E}_{s'} [V(s')]] - (1 - \gamma) \mathbb{E}_{s_0} [V(s_0)] + \psi(Q(s, a) - \gamma \mathbb{E}_{s'} [V(s')]),$$

where:

- The first term is linear in  $Q$ ,
- The second term is a constant (since the  $V$ -function is fixed when updating the  $Q$ -function), and
- The last term involves a convex function of  $Q$ , making it convex.

Thus,  $\tilde{F}(Q|V)$  is convex in  $Q$ .

For the convexity of  $J(V|Q)$ , we first recall its formulation:

$$J(V|Q) = \mathbb{E}_{(s, a) \sim \mu} \left[ e^{\frac{Q(s, a) - V(s)}{\beta}} - \left( \frac{Q(s, a) - V(s)}{\beta} \right) - 1 \right].$$

The convexity of  $J(V|Q)$  can be observed as follows:

- The first term is an exponential function of  $V$ , which is convex in  $V$ ,
- The remaining terms are either linear in  $V$  or constants.

Therefore,  $J(V|Q)$  is convex in  $V$ .  $\square$

## C Connection between UNIQ and Statistical Distance Maximization

In this section, we connect the new MaxEnt objective function of UNIQ with the concept of statistical distance. Specifically, we show that solving the new MaxEnt objective  $\min_r \min_\pi L(\pi, r)$  in (4) is equivalent to maximizing a statistical distance between the occupancy measures of the learning policy and the undesirable policy. In particular, if the feasible set of the reward function is unrestricted, the following equality holds:

$$\min_r \min_\pi \{L(\pi, r)\} = -\max_\pi \{d_\psi(\rho^\pi, \rho^{\text{UN}}) - H(\pi)\}, \quad (9)$$

where  $d_\psi(\rho^\pi, \rho^{\text{UN}}) = \psi^*(\rho^\pi - \rho^{\text{UN}})$ , and  $\psi^*$  is the convex conjugate of the convex function  $\psi$ , i.e.,

$$\psi^*(t) = \sup_z (tz - \psi(z)).$$

To verify (9), we write the objective function as:

$$L(\pi, r) = \sum_{s,a} r(s, a) (\rho^{\text{UN}}(s, a) - \rho^\pi(s, a)) + \psi(r) - H(\pi).$$

Thus, the minimization of  $L(\pi, r)$  over  $r$  can be expressed as:

$$\begin{aligned} \min_r L(\pi, r) &= H(\pi) + \min_r \left\{ \sum_{s,a} r(s, a) (\rho^{\text{UN}}(s, a) - \rho^\pi(s, a)) + \psi(r) \right\} \\ &= H(\pi) - \max_r \left\{ \sum_{s,a} r(s, a) (\rho^\pi(s, a) - \rho^{\text{UN}}(s, a)) - \psi(r) \right\}. \end{aligned}$$

Since the feasible set of the reward function  $r$  is unrestricted, we have:

$$\max_r \{ \langle r, \rho^\pi - \rho^{\text{UN}} \rangle - \psi(r) \} = \psi^*(\rho^\pi - \rho^{\text{UN}}).$$

This allows us to rewrite the training problem as:

$$\begin{aligned} \min_r \min_\pi \{L(\pi, r)\} &= \min_\pi \{H(\pi) - \psi^*(\rho^\pi - \rho^{\text{UN}})\} \\ &= -\max_\pi \{ \psi^*(\rho^\pi - \rho^{\text{UN}}) - H(\pi) \} \\ &= -\max_\pi \{ d_\psi(\rho^\pi, \rho^{\text{UN}}) - H(\pi) \}. \end{aligned}$$

We thus obtain the desired equation in (9), proving the equivalence.

## D Experimental settings

### D.1 Environmental Details

#### D.1.1 Mujoco

The MuJoCo locomotion benchmark focuses on a set of environments designed to evaluate the ability of reinforcement learning algorithms to control agents with complex, multi-joint dynamics. These tasks involve training agents to move efficiently and stably in simulated environments. The illustrations are shown in Figure 5.



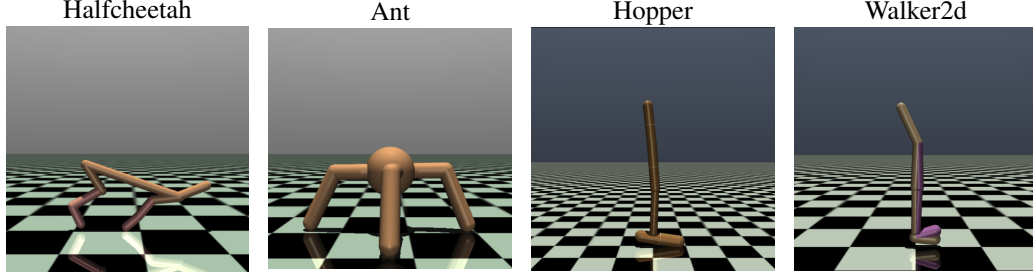


Figure 5: Mujoco environments.

### D.1.2 Safe-Gym

Safe-Gym is a collection of reinforcement learning environments designed with a focus on safety, built on top of the OpenAI Gym framework. It introduces constraints that simulate safety-critical scenarios commonly encountered in real-world applications. In Safe-Gym, agents are rewarded for completing task-specific objectives but face penalties for violating safety constraints, such as surpassing speed limits, colliding with obstacles, or entering restricted zones. These constraints enable Safe-Gym to replicate environments where safety is paramount, including robotic navigation in congested areas, autonomous vehicle control, and industrial automation. The environment features two types of agents: Point (an easy agent) and Car (a more challenging agent), as well as two types of tasks: Goal (easy) and Button (hard). Additionally, the environment dynamics change with each new episode, introducing variability and increasing the complexity of the tasks. Illustrations of these tasks are shown in Figure 6.

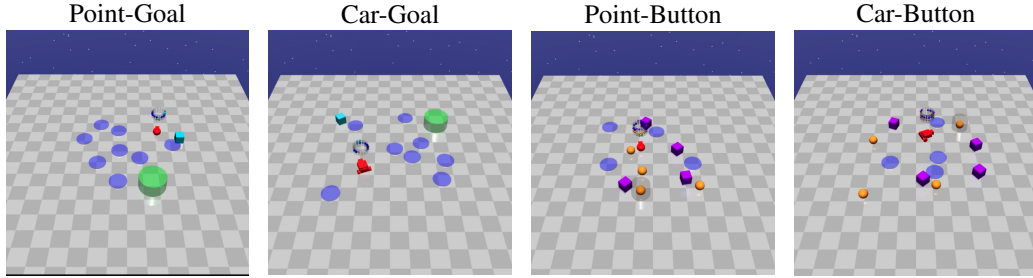


Figure 6: Safety-gym environments.

### D.1.3 Mujoco-velocity

From original Mujoco, Mujoco-Velocity is a specialized environment within the Mujoco physics simulation suite, focusing on controlling the velocity of two specific agents: Cheetah and Ant. These agents must complete locomotion tasks while adhering to safety constraints on their speed. The goal is to balance task performance with maintaining safe velocity limits. For instance, Cheetah must run as fast as possible while staying within predefined speed bounds to avoid penalties, mimicking real-world scenarios where exceeding speed limits can cause system failure or unsafe operations. Similarly, Ant must navigate through its environment without violating velocity constraints, ensuring stability and safety.

## D.2 Dataset generation details

In this paper, we tackle the problem of **Offline Learning with Undesirable Demonstrations** by utilizing two datasets:

- Unlabeled dataset  $\mathcal{D}^{\text{MIX}}$ : A large dataset comprising both desired and undesired demonstrations, reflecting real-world data (e.g., chat conversations, driving behaviors, treatment decisions, etc.).

- Undesired dataset  $\mathcal{D}^{\text{UN}}$ : A smaller, accurate dataset containing demonstrations that exhibit behaviors we aim to avoid.

### D.2.1 Mujoco dataset

We use the official D4RL dataset with three different performance: random, medium (sub-optimal), and expert. We then combine them to collect the dataset as follow:

- Unlabeled dataset  $\mathcal{D}^{\text{MIX}}$ : We mix all three dataset into a large unlabeled dataset with a specific ratio.
- Undesired dataset  $\mathcal{D}^{\text{UN}}$ : We mix random and medium dataset with the same ratio (1 : 1).

### D.2.2 Safety-gym and Mujoco-velocity dataset

We simulate this scenario using the Safety-gym and Mujoco-velocity environments. First, we train both unconstrained and constrained policies using PPO [41] and PPO-lagrangian [38]. We then collect the training datasets as follows:

- Unlabeled dataset  $\mathcal{D}^{\text{MIX}}$ : We roll out the constrained and unconstrained policies, mixing them at ratios of (1 : 4).
- Undesired dataset  $\mathcal{D}^{\text{UN}}$ : We roll out the unconstrained policy, gathering trajectories that violate the constraint.

The quality of Safety-gym dataset are shown in Table 2 while details information of Mujoco-velocity datasets are shown in Table 3.

	Point-Goal	Car-Goal	Point-Button	Car-Button
Mean Unconstrained return	$26.8 \pm 1.1$	$35.2 \pm 2.1$	$21.49 \pm 4.87$	$17.82 \pm 5.57$
Mean Constrained return	$25.3 \pm 2.1$	$25.7 \pm 6.6$	$13.71 \pm 6.44$	$14.99 \pm 8.45$
Mean Unconstrained cost	$57.8 \pm 38.9$	$61.9 \pm 48.1$	$138.9 \pm 77.3$	$236.5 \pm 115.9$
Mean Constrained cost	$22.3 \pm 28.0$	$21.0 \pm 31.0$	$27.4 \pm 34.1$	$113.5 \pm 33.2$
Cost Threshold	25.0	25.0	25.0	100.0

Table 2: Safety-Gym expert policies performance.

	Cheetah	Ant
Mean Unconstrained return	$3027.4 \pm 400.6$	$2972.2 \pm 1020.0$
Mean Constrained return	$2751.2 \pm 11.6$	$2830.0 \pm 145.5$
Mean Unconstrained cost	$626.7 \pm 95.0$	$624.0 \pm 231.0$
Mean Constrained cost	$14.1 \pm 4.5$	$18.7 \pm 4.4$
Cost Threshold	25.0	25.0

Table 3: Mujoco-velocity expert policies performance.

## D.3 Baseline implementation details

### D.3.1 BC

We use the original BC objective:

$$\min_{\pi} -\mathbb{E}_{s,a \sim \mathcal{D}} \log \pi(a|s) \quad (10)$$

### 642 D.3.2 LS-IQ

We use the official implementation of LS-IQ in [this link](#). We modify the objective to fit with our scenario when undesirable dataset available:

$$\min_Q \mathbb{E}_{s,a \sim \mathcal{D}^{\text{UN}}} [(Q(s,a) - Q_{\min})^2] + \mathbb{E}_{s,a,s' \sim \mathcal{D}^{\text{MIX}}} [(Q(s,a) - r_{\max} - \gamma V(s'))^2],$$

643 where  $V(s) = Q(s, \pi(a|s)) + H(\pi)$ ;  $Q_{\min} = r_{\min}/(1 - \gamma)$ ; and  $r_{\min}$  and  $r_{\max}$  are predefined.

### 644 D.3.3 IPL

We use the official implementation of IPL [17] from [this link](#). The only difference between our setting and IPL is the pairwise dataset. We create pairwise comparisons from the unlabeled dataset and the undesired dataset and train the Q-function with the following new loss function:

$$P_{Q^\pi}[\sigma^{\text{MIX}} > \sigma^{\text{UN}}] = \frac{\exp \sum_t (\mathcal{T}^\pi Q)(s_t^{\text{MIX}}, a_t^{\text{MIX}})}{\exp \sum_t (\mathcal{T}^\pi Q)(s_t^{\text{MIX}}, a_t^{\text{MIX}}) + \exp \sum_t (\mathcal{T}^\pi Q)(s_t^{\text{UN}}, a_t^{\text{UN}})},$$

where:

$$(\mathcal{T}^\pi Q)(s, a) = Q(s, a) - \gamma \mathbb{E}_{s'} [V^\pi(s')].$$

### 645 D.3.4 DWBC

We use the official implementation of DWBC [45] from [this link](#). We modify the algorithm to train a discriminator that assigns 0 to the undesired dataset  $\mathcal{D}^{\text{UN}}$  and 1 to the unlabeled dataset  $\mathcal{D}^{\text{MIX}}$  while keeping the Positive Unlabeled learning technique from the paper:

$$\begin{aligned} \min_{\theta} \eta \mathbb{E}_{(s,a) \sim \mathcal{D}^{\text{MIX}}} [-\log d_\theta(s, a, \log \pi)] \\ + \mathbb{E}_{(s,a) \sim \mathcal{D}^{\text{UN}}} [-\log (1 - d_\theta(s, a, \log \pi))] \\ - \eta \mathbb{E}_{(s,a) \sim \mathcal{D}^{\text{MIX}}} [-\log (1 - d_\theta(s, a, \log \pi))] . \end{aligned}$$

### 646 D.3.5 SafeDICE

647 We use the official implementation of SafeDICE [20] from [this link](#). As the algorithm has been  
648 designed to solve this problem, we do not make any further modifications.

## 649 D.4 Hyper-parameter selection

650 For fair comparison, we keep the same basic hyper-parameters across all the baselines which are  
651 detailed as follow for Mujoco, Safety-gym and Mujoco-velocity tasks as Table 4:

HYPER PARAMETER	MUJOCO	SAFETY-GYM	MUJOCO-VELOCITY
ACTOR NETWORK	256x3	256x3	256x3
CRITIC NETWORK	256x3	256x3	256x3
VALUE NETWORK	256x3	256x3	256x3
TRAINING STEP	1,000,000	1,000,000	1,000,000
GAMMA	0.99	0.99	0.99
LR ACTOR	0.0003	0.0001	0.0001
LR CRITIC	{0.0001, 0.0003}	0.0003	0.0003
LR VALUE	0.0003	0.0003	0.0003
LR DISCRIMINATOR	0.0001	0.0001	0.0001
BATCH SIZE	256	256	256
SOFT UPDATE CRITIC FACTOR	[0.001, 0.01]	0.005	0.005

Table 4: Hyper parameters.

In UNIQ, the  $\alpha$  and  $\beta$  parameters are selected in range  $[0.5, 10.0]$ . Lastly, we apply state normalization for Mujoco and Mujoco-velocity datasets as follow:

$$s_{\text{normalized}} = \frac{s - \mu}{\sigma}$$

Where:

$$\mu = \frac{1}{|\mathcal{D}|} \sum_{s' \in \mathcal{D}} s'$$
$$\sigma = \sqrt{\frac{1}{|\mathcal{D}|} \sum_{s' \in \mathcal{D}} (s' - \mu)^2}$$

## 652 **D.5 Computational Resource**

653 Our experiments were carried out on a GPU cluster equipped with 8 NVIDIA RTX 3090 GPUs. Each  
654 experimental setup was run with five distinct training seeds in parallel, sharing a single GPU, eight  
655 CPU cores, and 64 GB of RAM. Within this shared configuration, completing one million training  
656 steps across all five seeds took approximately 30 minutes. The software environment was built using  
657 JAX version 0.4.28, with support for CUDA 12—specifically, CUDA version 12.3.2 and cuDNN  
658 version 8.9.7.29.

659 It is important to note that in the Safety-Gym domain, evaluation is relatively slow due to the use of  
660 50 different environment random seeds for each (we have 100 evaluations in total). While the training  
661 phase takes around 30 minutes, the extensive evaluation increases the total runtime per experiment to  
662 approximately 2 hours.

## 663 E Additional Experiments

### 664 E.1 Performance with Different Sizes of undesirable Dataset

665 In this section, we provide a complete experiment to evaluate the contribution of the undesirable  
 666 dataset to our algorithm. We test our method with an increasing size of the undesirable dataset,  
 667  $\mathcal{D}^{\text{UN}} = \{5, 10, 20, 50, 100, 200\}$  for the Mujoco tasks and  $\mathcal{D}^{\text{UN}} = \{25, 50, 100, 200, 300, 500\}$ . The  
 668 full experiment results are shown in Figure 7 (Mujoco) and Figure 8 (Safety-Gym).

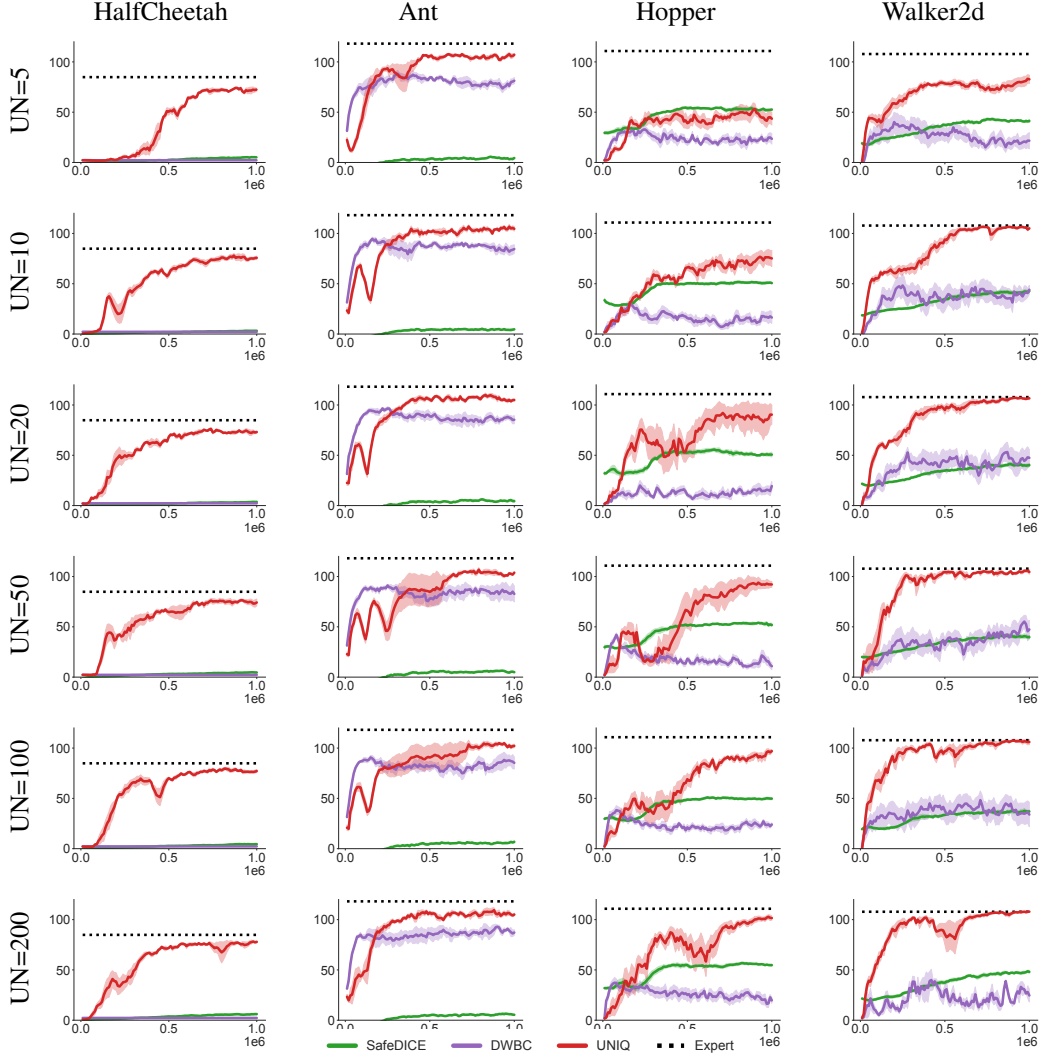


Figure 7: Comparison with different size of Undesirable dataset in Mujoco environments.

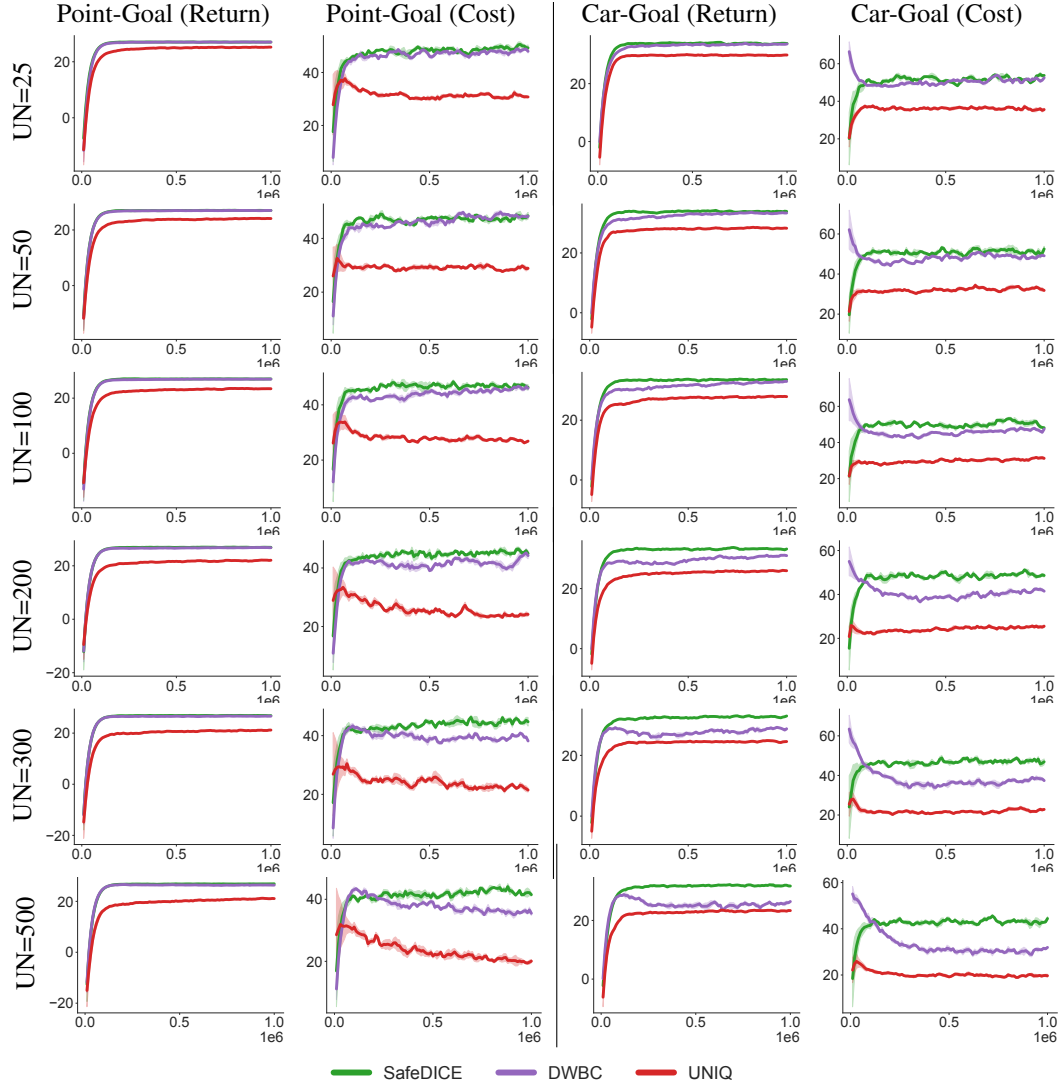


Figure 8: Comparison with different size of Undesirable dataset in Safety-gym environments.

669 **E.2 Performance with Different Number of Undesirable Demonstrations in the Unlabeled**  
670 **Dataset**

671 We further change the number of undesirable demonstrations in the unlabeled dataset  $\mathcal{D}^{\text{Mix}}$  to examine  
672 how it impacts the performance of the algorithm. The experimental results are shown in Figure 9.  
673 The results show that when the quality of the dataset decreases significantly, the performance of all  
674 algorithms worsens. However, UNIQ still achieves the highest performance across all environments.

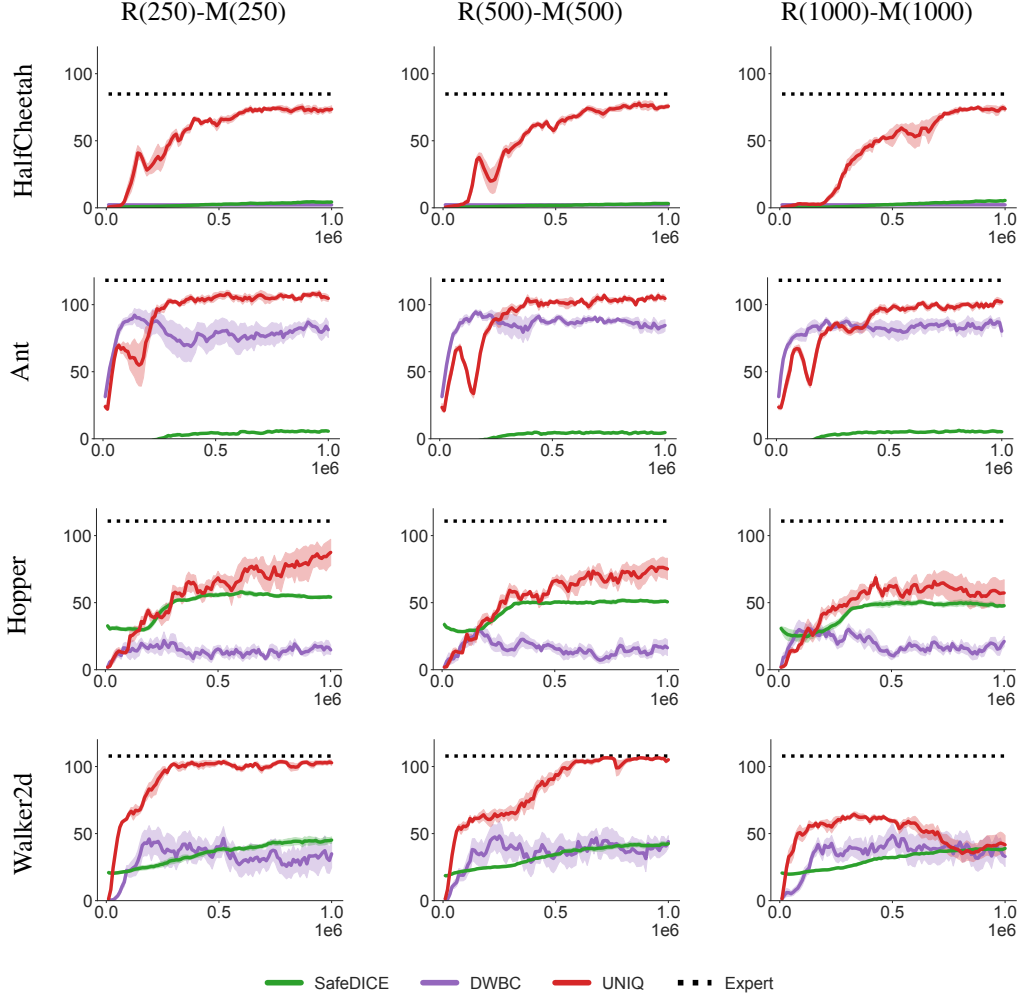


Figure 9: Comparison with different size of Undesirable dataset in Mujoco environments.

### 675 E.3 Performance with Different Number of Desirable Demonstrations in the Unlabeled 676 Dataset

677 We also test how changing the amount of desirable data in the unlabeled dataset affects the perfor-  
678 mance. The experimental results are shown in Figure 10 (Mujoco) and Figure 11 (Safety-Gym). We  
679 observe an increasing trend in performance as the size of the desirable data increases (higher returns  
680 in Mujoco and lower costs in Safety-Gym).

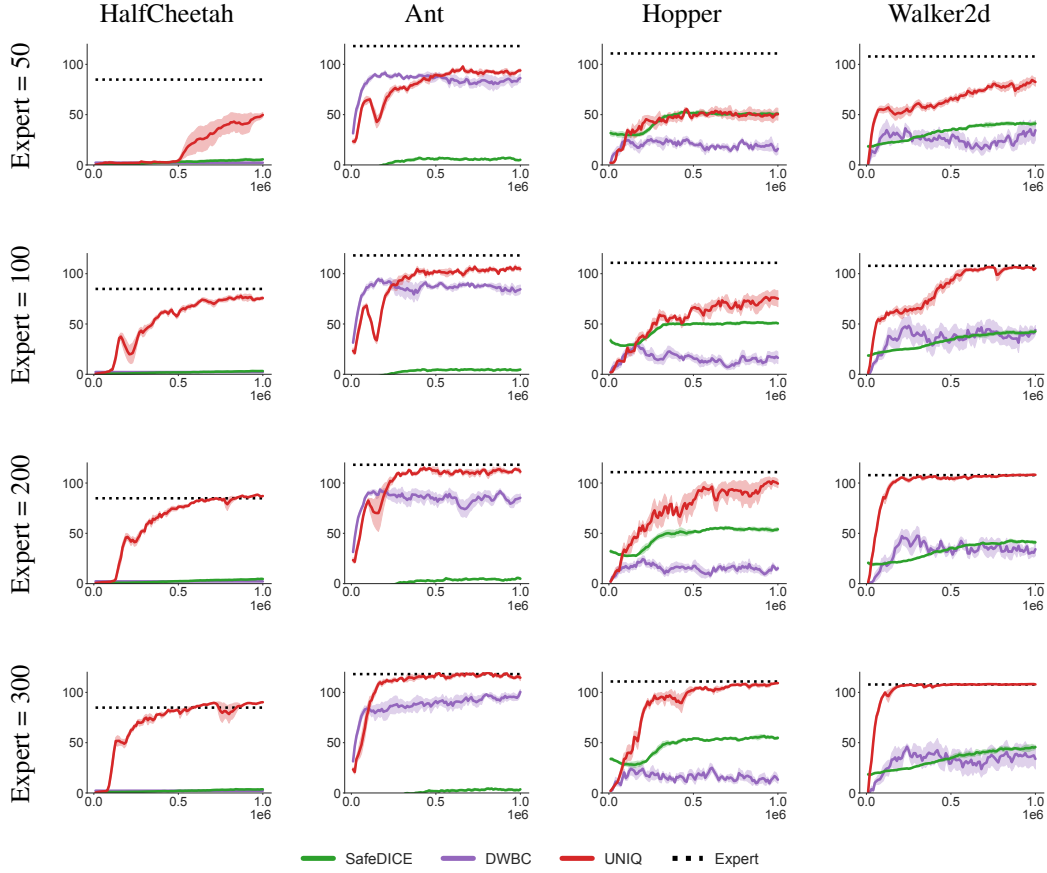


Figure 10: Comparison with different size of Undesirable dataset in Mujoco environments.



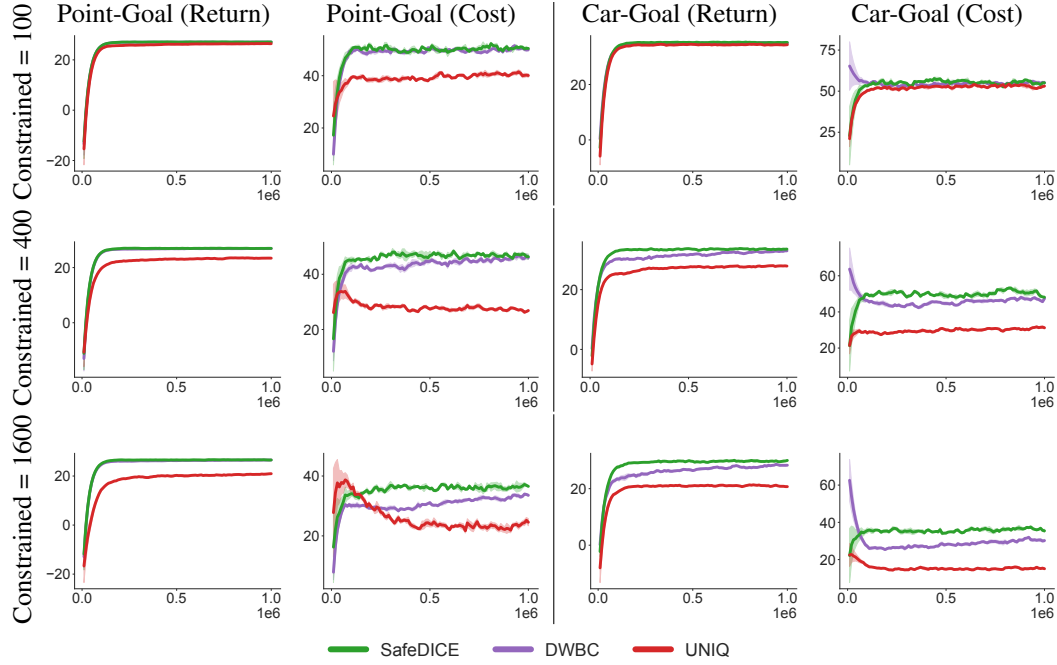


Figure 11: Comparison with different size of Undesirable dataset in Safety-gym environments.

#### 681 E.4 Direct Policy Extraction from Q-functions

682 As our actor update objective uses Weighted Behavioral Cloning, a question arises: what if we directly  
 683 extract the actor from the Q-function [12, 3]? We conduct experiments to evaluate the performance of  
 684 UNIQ when the actor is learned directly from the Q-function. The results are presented in Figure 12  
 685 and Figure 13..

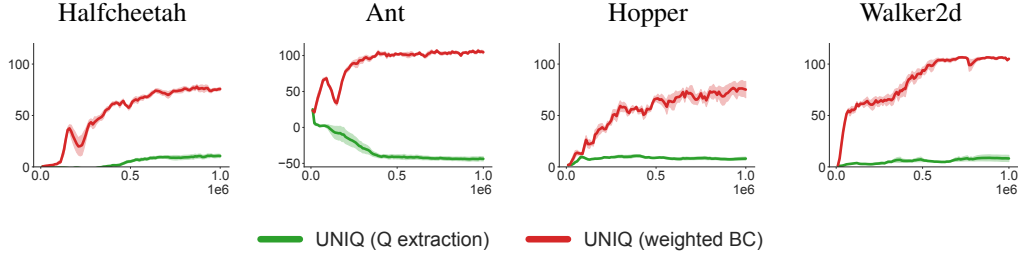


Figure 12: compare UNIQ BC update with Q extraction update in D4RL

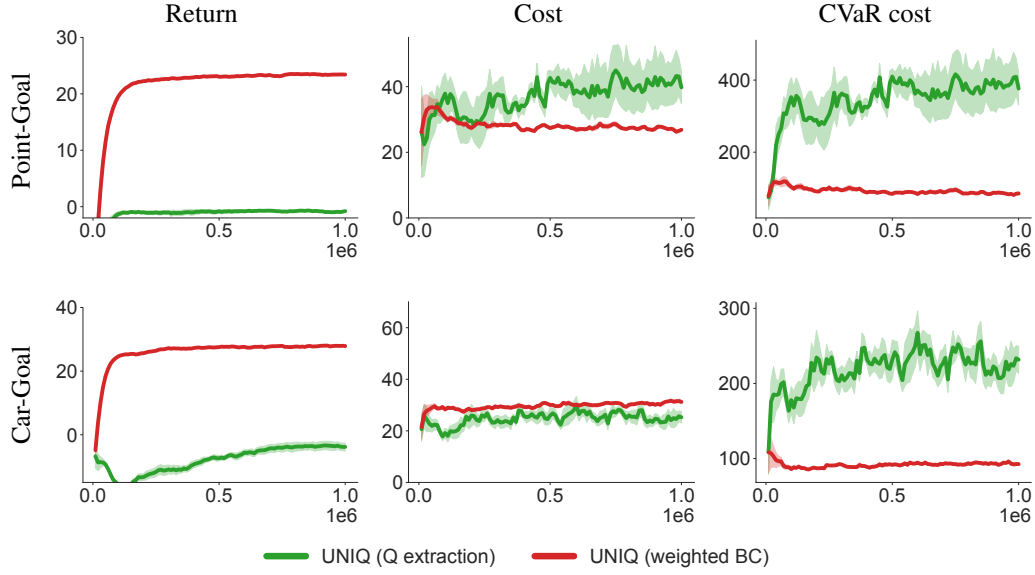


Figure 13: compare UNIQ BC update with Q extraction update in Safetygym

#### 686 E.5 BC without Unlabeled Dataset

687 As UNIQ uses an unlabeled dataset to learn by avoiding undesirable behaviors and learning from the  
 688 remaining parts of the dataset, an interesting question arises: what if we only avoid the undesirable  
 689 dataset (no unlabeled dataset is given)? To explore this, we train a Behavioral Cloning (BC) agent  
 690 that, instead of following the dataset, explicitly avoids the actions from the undesirable dataset by  
 691 minimizing the probability of those assigned actions, denoted as BC-UN. The detailed results are  
 692 shown in Figure 14 (Mujoco) and Figure 15 (Safety-Gym).

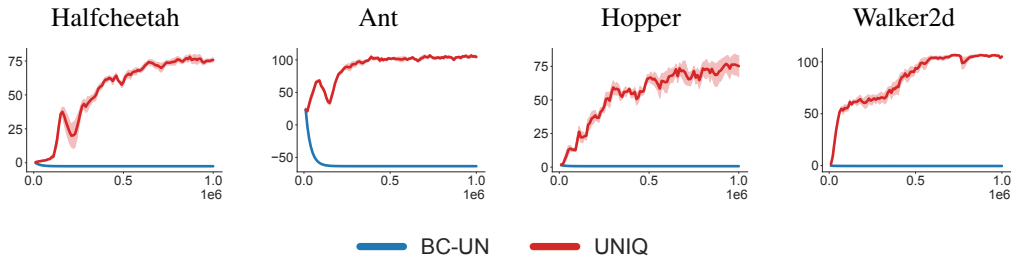


Figure 14: BC only avoid undesirable data in D4RL

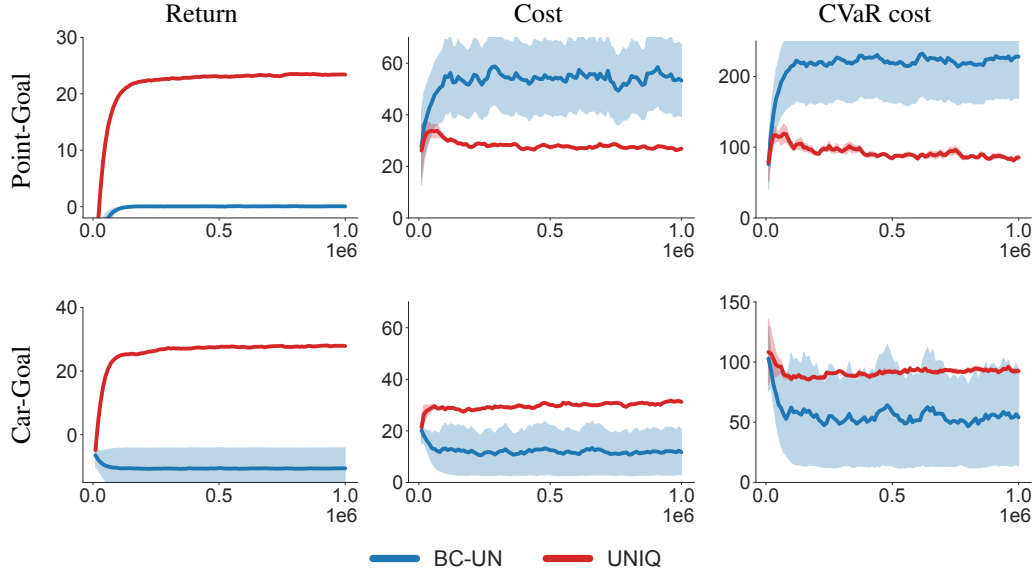


Figure 15: BC only avoid undesirable data in Safetygym

## E.6 ablation study with different $\tau$

Task	$\tau = 1$	Non-converged $\tau$	Converged $\tau$
HalfCheetah	$18.5 \pm 5.6$	$64.9 \pm 5.2$	<b><math>75.7 \pm 6.8</math></b>
Ant	$88.5 \pm 8.2$	$95.2 \pm 12.4$	<b><math>104.4 \pm 10.5</math></b>
Hopper	$57.4 \pm 12.4$	$58.0 \pm 14.6$	<b><math>73.5 \pm 20.6</math></b>
Walker2d	$12.8 \pm 7.7$	$72.0 \pm 6.1$	<b><math>105.9 \pm 4.0</math></b>

Table 5: Ablation study with different  $\tau$ : We maintain the same dataset construction as in the main experiment.

To assess the effect of occupancy ratio estimation on our algorithm, we conducted additional ablation studies using three configurations: (i) a uniform occupancy ratio ( $\tau = 1$ ), (ii) a non-converged estimation of  $\tau$ , and (iii) a properly converged  $\tau$ . The results of this ablation study (for the unconstrained RL setting) are shown in Table 5.

## E.7 Additional comparison with modified baselines for avoiding undesirable

Task	BC-remove-bad	IS-WBC	ILID	UNIQ
HalfCheetah	$2.3 \pm 0.0$	$2.2 \pm 0.0$	$2.1 \pm 0.0$	<b><math>75.7 \pm 6.8</math></b>
Ant	$36.2 \pm 19.6$	$73.4 \pm 12.3$	<b><math>106.44 \pm 4.3</math></b>	$104.4 \pm 10.5$
Hopper	$19.7 \pm 18.9$	$7.6 \pm 5.0$	$5.9 \pm 6.5$	<b><math>73.5 \pm 20.6</math></b>
Walker2d	$37.8 \pm 33.3$	$18.8 \pm 19.3$	$69.6 \pm 13.0$	<b><math>105.9 \pm 4.0</math></b>

Table 6: Additional comparison with BC-remove-bad, IS-WBC, and ILID: while keeping the dataset construction consistent with the main experiment.

Since our method is designed to learn from undesirable demonstrations, we evaluate it alongside several baseline algorithms that were originally developed for expert demonstrations but have been adapted for our setting:

- **BC-remove-bad:** We first train a classifier to detect and remove undesirable demonstrations from the unlabeled dataset. Behavior Cloning (BC) is then applied to the filtered data to learn the policy.
- **ISW-BC:** We modify the original discriminator objective to better suit the learning-from-undesirable-demonstrations framework. Specifically, we train the discriminator to differ-

entiate between undesirable and unlabeled demonstrations, rather than between expert and non-expert data as in the original method. This adaptation helps guide the policy away from undesirable behaviors without requiring expert data.

- **ILID**: We adapt the original ILID approach by removing all states in the unlabeled dataset that resemble those in the undesirable dataset. However, the original ILID also includes a regularization component that prevents the learned policy from deviating significantly from a BC-trained expert policy. Since expert demonstrations are unavailable in our setting, this regularization is excluded in our adaptation.

The comparative performance results are reported in Table 6. Overall, UNIQ yell the best performance in Mujoco tasks.

#### E.8 Performance with 4 category types (low-cost low-reward, low-cost high-reward, high-cost low-reward, high-cost high-reward) of dataset in Safety-gym

		DWBC	SafeDICE	UNIQ	Expert
Point-Goal	Return	$20.1 \pm 1.1$	$19.9 \pm 0.4$	<b><math>24.3 \pm 0.8</math></b>	$25.9 \pm 0.2$
	Cost	$47.0 \pm 4.5$	$41.7 \pm 5.0$	<b><math>26.0 \pm 4.4</math></b>	$26.0 \pm 2.6$
Car-Goal	Return	$20.9 \pm 1.5$	$22.0 \pm 3.6$	<b><math>22.5 \pm 2.2</math></b>	$26.2 \pm 0.7$
	Cost	$30.3 \pm 3.3$	$30.5 \pm 7.7$	<b><math>19.9 \pm 2.8</math></b>	$23.6 \pm 2.8$

Table 7: Performance of 4 different category of performance types on Point-Goal and Car-Goal Tasks.

The objective of this experiment is to evaluate whether the algorithm can successfully avoid all types of undesirable demonstrations in the Safety-Gym environment. While our main experiments (Table 1) primarily assess safety performance with minimal return degradation, this experiment focuses on the algorithm’s ability to comprehensively reject undesirable trajectories.

To achieve this, we construct a dataset based on four categories of Safety-Gym performance: high-return low-cost (H-L), high-return high-cost (H-H), low-return low-cost (L-L), and low-return high-cost (L-H). The dataset is divided into two subsets: the undesirable dataset  $\mathcal{D}^{UN}$ , which consists of 50 trajectories each from the H-H, L-L, and L-H categories (totaling 150 trajectories); and the unlabeled dataset  $\mathcal{D}^{MIX}$ , which contains 500 trajectories from each category, resulting in 2,000 trajectories in total.

The experimental results are reported in Table 7. may initially appear counterintuitive—SafeDICE and UNIQ exhibit reduced cost despite the more challenging data. However, this outcome is likely due to a trade-off inherent in the policy behavior: lower rewards are often correlated with lower costs. In particular, policies that prioritize minimizing cost may opt for conservative behaviors, such as remaining stationary, which avoids hazards but also leads to diminished task progress and, consequently, lower returns.

## 735 E.9 Comparisons with CVaR 10% Cost for Safety-gym Tasks

736 We also report the CVaR 10% cost for Safety-gym tasks, supporting the result of the Table 1 with  
 737 CVaR is the mean of 10% highest in cost trajectories during the evaluation process. The full results  
 738 are shown in Table 8.

		BC-mix	LS-IQ	IPL	DWBC	SafeDICE	UNIQ
Point-Goal	Return	27.1±0.1	−6.8 ± 4.4	26.9±0.1	26.9±0.1	27.0±0.1	23.4±0.4
	Cost	48.8±2.9	18.0 ± 29.3	52.7±3.4	45.8±3.4	46.8±3.1	27.1±3.0
	CVaR	115.4±7.7	133.3 ± 240.7	117.9±8.2	110.4±7.8	111.0±7.6	<b>85.9± 18.9</b>
Car-Goal	Return	34.1±0.5	−0.6 ± 2.5	34.7±0.3	32.8±0.7	33.5±0.7	27.9±0.8
	Cost	52.0±4.2	58.5 ± 41.7	54.4±3.7	47.4±3.8	50.5±4.0	31.0±2.8
	CVaR	132.8±10.6	311.1 ± 208.9	134.9±8.7	123.2±10.6	128.5±10.2	<b>93.3 ± 8.4</b>
Point-Button	Return	17.6 ± 0.7	−13.3 ± 6.7	16.9±0.9	17.2 ± 0.9	15.1 ± 0.5	12.6 ± 1.4
	Cost	120.2 ± 10.0	11.0 ± 10.2	124.8±11.3	123.5 ± 14.4	91.0 ± 6.4	23.0 ± 4.7
	CVaR	311.0 ± 47.8	58.2 ± 89.1	307.2 ± 47.3	309.6 ± 63.7	229.2 ± 21.2	<b>98.7 ± 33.1</b>
Car-Button	Return	17.6 ± 0.7	−8.6 ± 4.8	17.2±0.8	17.1 ± 1.0	17.4 ± 0.6	12.7 ± 1.1
	Cost	241.6 ± 15.3	28.0 ± 27.0	257±12.6	249.2 ± 20.9	201.3 ± 10.8	148.6 ± 18.7
	CVaR	545.3 ± 45.6	195.9 ± 204.1	553.2 ± 35.7	550.5 ± 66.3	<b>410.5 ± 34.3</b>	449.8 ± 63.4

Table 8: Full comparison results in Return, Cost, and CVaR 10%.

## 739 E.10 Controlling Conservativeness in UNIQ

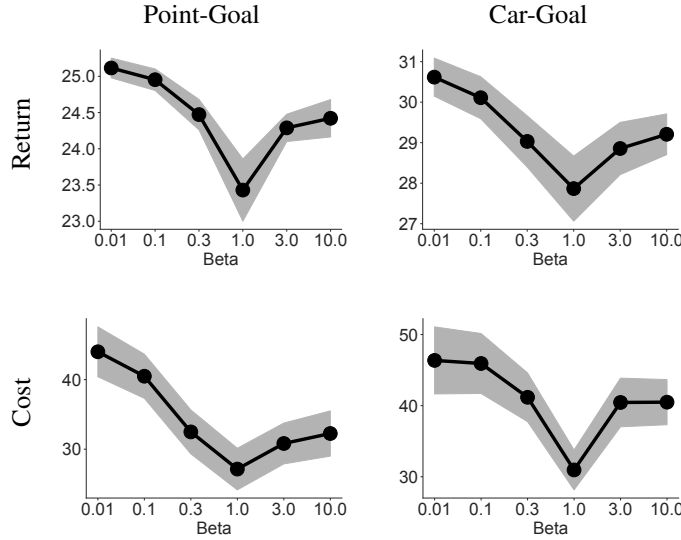


Figure 16: Comparison results of UNIQ with different  $\alpha$  selections.

740 The main paper demonstrates that the policy returned by UNIQ achieves significantly lower costs  
 741 (indicating safety) but, in some cases, also lower rewards compared to other imitation learning  
 742 baselines. While this aligns well with our objective of learning safe policies by avoiding unsafe  
 743 demonstrations, it also raises concerns about the algorithm’s conservativeness.

744 In this section, we show that the conservativeness of UNIQ can be effectively controlled by introducing  
 745 a parameter to the Weighted BC formulation. Specifically, we adjust the conservativeness of the  
 746 algorithm by adding a parameter  $\beta$  to the Weighted BC update:

$$\sum_{(s,a) \sim \mathcal{D}^{\text{Mix}}} \exp((Q_{w_q}(s,a) - V_{w_v}(s)) * \beta) \log \pi_{\theta}(a|s)$$

When  $\beta = 1$ , the Weighted BC theoretically returns the exact policy derived from Q-learning, as reported in the main paper. In contrast:

- As  $\beta \rightarrow 0$ , the Weighted BC returns a random policy.
- As  $\beta \rightarrow \infty$ , the resulting policy becomes deterministic, always selecting the best action with probability 1.

Thus, by varying  $\beta$ , we can deviate the outcome of the Weighted BC from the policy given by Q-learning, reducing the conservativeness of the learned policy.

To experimentally demonstrate this, we vary  $\beta$  and report the corresponding returns and costs on four MuJoCo environments. The results are presented in Figure 16 and Table 9, showing how different values of  $\beta$  impact the trade-off between safety and performance.

Figure 16 demonstrates that UNIQ achieves its safest (and most conservative) performance when  $\beta = 1$ . At this value, the policy prioritizes minimizing costs, making it the most risk-averse option. However, as  $\beta$  deviates from 1, both the cost and return increase. This indicates that the Weighted BC formulation produces less conservative policies that are less safe but capable of achieving higher rewards.

Table 9 provides a more detailed breakdown of the costs and returns for different values of  $\beta$ . The results show that UNIQ can effectively balance safety and performance: by adjusting  $\beta$ , it is possible to achieve a safer policy (i.e., lower cost) while maintaining competitive returns (compared to other baselines). This adaptability highlights the flexibility of UNIQ.

When safety is critical, setting  $\beta = 1$  ensures the most conservative policy, aligning with the objective of avoiding unsafe demonstrations. On the other hand, by varying  $\beta$ , one can tune the trade-off to achieve policies that are less safe but yield higher rewards, making UNIQ suitable for a range of scenarios depending on the desired safety-performance balance. This versatility demonstrates its practicality across different applications with varying safety requirements.

		DWBC	SafeDICE	UNIQ (0.01)	UNIQ (0.1)	UNIQ (0.3)	UNIQ (1.0)	UNIQ (3.0)
Point-Goal	Return	26.9±0.1	27.0±0.1	25.1 ± 0.1	25.0 ± 0.1	24.5 ± 0.2	23.4±0.4	24.3 ± 0.2
	Cost	45.8±3.4	46.8±3.1	44.0 ± 3.6	40.5 ± 3.2	32.5 ± 3.2	<b>27.1±3.0</b>	30.8 ± 2.9
Car-Goal	Return	32.8±0.7	33.5±0.7	30.6 ± 0.5	30.1 ± 0.5	29.0 ± 0.6	27.9±0.8	28.9 ± 0.6
	Cost	47.4±3.8	50.5±4.0	46.4 ± 4.7	45.9 ± 4.2	41.2 ± 3.4	<b>31.0±2.8</b>	40.4 ± 3.4

Table 9: Comparison with different  $\alpha$

## E.11 Full Numerical Experiment for Mujoco Velocity Tasks

We evaluate our method on two MuJoCo velocity tasks: Cheetah and Ant. we test the method with varying sizes of the undesired dataset, annotated as "env-UN= {1, 5, 10}" while the unlabeled dataset  $\mathcal{D}^{\text{Mix}}$  is combined from 1600 high-cost and 400 low-cost trajectories. The detailed results are summarized in Table 10 and learning curves are shown in Figure 17 and Figure 18. Overall, increasing the size of the undesired dataset helps SafeDICE and DWBC achieve higher performance, while UNIQ reaches its peak performance with just a single undesired trajectory.

		DWBC	SafeDICE	UNIQ
Cheetah-UN=10	Return	3135.6 $\pm$ 127.4	2841.9 $\pm$ 56.1	<b>2662.0<math>\pm</math>33.1</b>
	Cost	311.0 $\pm$ 116.0	550.2 $\pm$ 13.5	<b>0.0<math>\pm</math>0.0</b>
	CVaR	897.7 $\pm$ 10.0	682.2 $\pm$ 14.4	<b>0.0<math>\pm</math>0.0</b>
Cheetah-UN=5	Return	3430.9 $\pm$ 107.5	2860.8 $\pm$ 57.8	<b>2661.2<math>\pm</math>29.7</b>
	Cost	578.8 $\pm$ 89.6	553.9 $\pm$ 25.3	<b>0.0<math>\pm</math>0.0</b>
	CVaR	909.2 $\pm$ 6.3	686.7 $\pm$ 17.3	<b>0.0<math>\pm</math>0.0</b>
Cheetah-UN=1	Return	3720.7 $\pm$ 39.2	2910.0 $\pm$ 61.8	<b>2755.3<math>\pm</math>23.8</b>
	Cost	823.0 $\pm$ 17.5	575.5 $\pm$ 23.0	<b>0.0<math>\pm</math>0.0</b>
	CVaR	916.4 $\pm$ 5.0	702.2 $\pm$ 20.0	<b>0.0<math>\pm</math>0.0</b>
Ant-UN=10	Return	2225.0 $\pm$ 759.3	2713.0 $\pm$ 56.2	<b>2850.5<math>\pm</math>177.5</b>
	Cost	470.5 $\pm$ 162.8	439.7 $\pm$ 57.7	<b>15.2<math>\pm</math>10.8</b>
	CVaR	795.0 $\pm$ 103.7	668.7 $\pm$ 14.6	<b>24.6<math>\pm</math>13.7</b>
Ant-UN=5	Return	2210.0 $\pm$ 655.7	2727.4 $\pm$ 49.8	<b>2838.2<math>\pm</math>177.9</b>
	Cost	494.5 $\pm$ 146.8	464.4 $\pm$ 35.3	<b>13.1<math>\pm</math>7.5</b>
	CVaR	805.7 $\pm$ 16.4	671.2 $\pm$ 16.0	<b>22.1<math>\pm</math>10.5</b>
Ant-UN=1	Return	2259.4 $\pm$ 653.8	2724.4 $\pm$ 90.7	<b>2841.4<math>\pm</math>214.9</b>
	Cost	507.5 $\pm$ 147.8	506.5 $\pm$ 40.3	<b>16.9<math>\pm</math>7.1</b>
	CVaR	789.3 $\pm$ 91.3	685.8 $\pm$ 16.4	<b>27.2<math>\pm</math>9.6</b>

Table 10: Full comparison between UNIQ and other baselines in Mujoco-velocity domain. With decreasing of undesirable dataset size, the performance of DWBC and SafeDICE become worse. In contrast, UNIQ able to achieve highest performance with just a single undesirable trajectory.

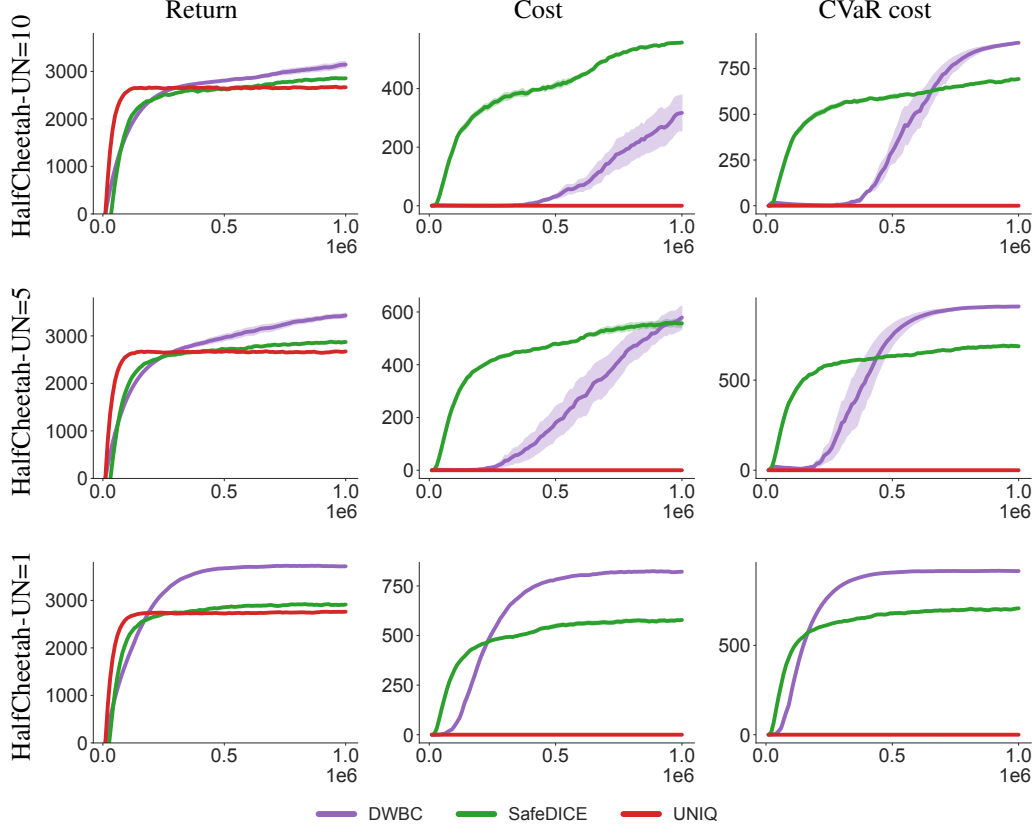


Figure 17: Cheetah task with unlabeled dataset(400-1600) and different undesired dataset.

#### 778 E.12 Performance with the Dataset Employed in the SafeDICE Paper

779 As we are using a different dataset from the SafeDICE dataset, we also provide a comparison with the  
 780 dataset from SafeDICE paper. The detailed performance of the expert dataset is shown in Table 11:

	Point-Goal	Point-Button
Mean non-preferred demonstrations cost	20.018	21.933
Mean preferred demonstrations return	19.911	8.286
Mean non-preferred demonstrations cost	107.977	166.099
Mean preferred demonstrations return	13.798	12.085

Table 11: SafeDICE dataset performance.

781 We mix 300 preferred demonstrations and 1200 non-preferred demonstrations for the unlabeled  
 782 dataset and use 100 non-preferred demonstrations for the undesired dataset. The performance is  
 783 shown in Figure 19. It is clearly that our method can achieve lower cost than SafeDICE.



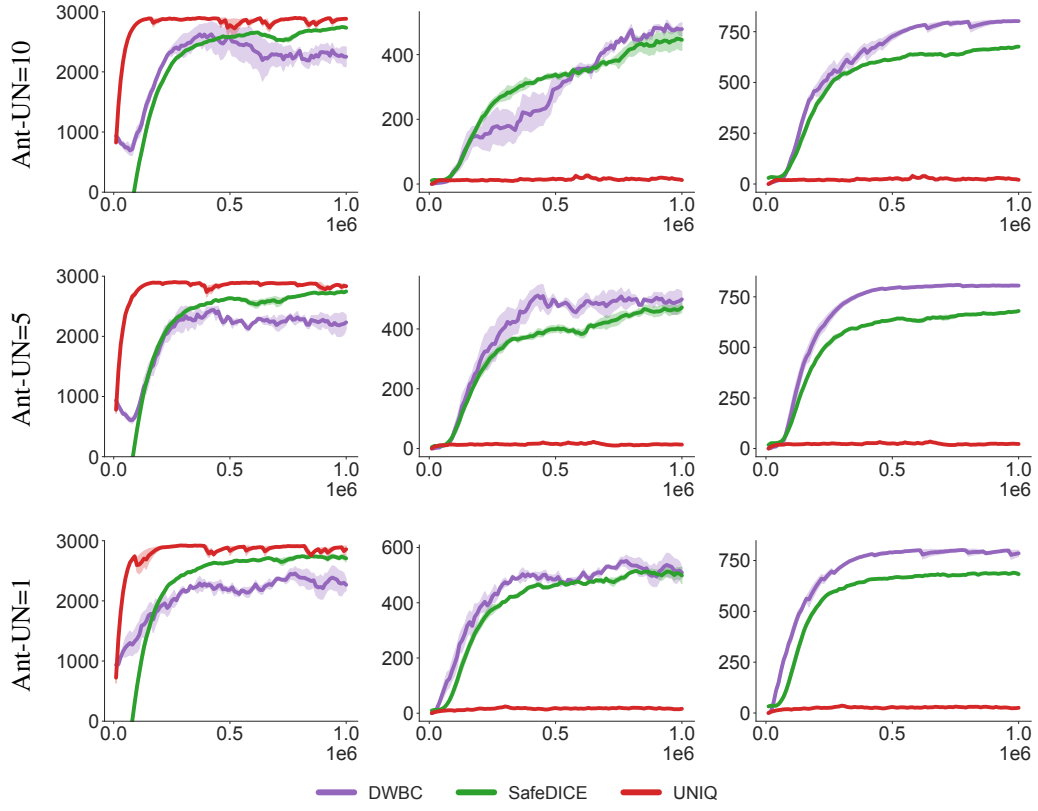


Figure 18: Ant task with unlabeled dataset(400-1600) and different undesired dataset.

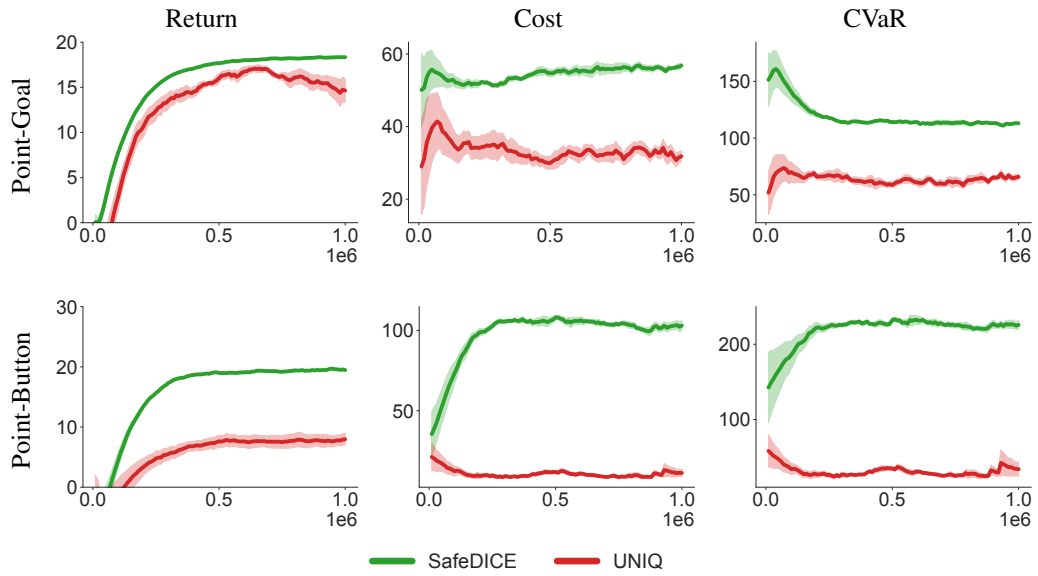


Figure 19: Comparison between UNIQ compared to SafeDICE in their dataset.

## NeurIPS Paper Checklist

### 1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: *Our abstract and introduction include our main claims reflecting our main contributions and finding.*

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

### 2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: *We have a discussion on the limitations of our work in the conclusion section.*

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

### 3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: *All the proofs of the theorems and propositions stated in the main paper are provided in the appendix with clear references.*

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

#### 4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: *We provide details on the environments and hyper-parameter settings in the appendix. We also uploaded our source code for re-productivity purposes.*

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

#### 5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: *We have describe how to generate our data as well as provide it along with our submitted source code with sufficient instructions for their use.*

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

## 6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: *We have detailed these information in the main paper and the appendix of our paper.*

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

## 7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: *We have reported the mean scores and standard deviations for the result tables. We have also shown training curves constructed from mean scores and shaded by standard error. All the experiments are reported with multiple training seeds as well as different datasets.*

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: *We have provided these information in the "Computational Resource" section in our appendix.*

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

## 9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification:

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

## 10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: *The paper provides a general offline imitation learning algorithm that only testing on the simulated environments. As such, we do not foresee any direct societal impact.*

Guidelines:

- The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

## 11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: *Our training data are from official datasets and official simulator which have no risk for misuse.*

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

## 12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: *We have provided clear citations to the source code and data we used in the paper.*

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets) has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

### 13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: *Our source code is submitted alongside the paper, accompanied by sufficient instructions. We will share the code publicly for re-productibility or benchmarking purposes.*

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

### 14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[NA\]](#)

Justification: *We have no crowdsourcing experiments.*

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

### 15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [\[NA\]](#)

Justification: *The paper is not research with human subjects.*

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- 1095 • Depending on the country in which research is conducted, IRB approval (or equivalent)  
1096 may be required for any human subjects research. If you obtained IRB approval, you  
1097 should clearly state this in the paper.
- 1098 • We recognize that the procedures for this may vary significantly between institutions  
1099 and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the  
1100 guidelines for their institution.
- 1101 • For initial submissions, do not include any information that would break anonymity (if  
1102 applicable), such as the institution conducting the review.

#### 1103 16. **Declaration of LLM usage**

1104 Question: Does the paper describe the usage of LLMs if it is an important, original, or  
1105 non-standard component of the core methods in this research? Note that if the LLM is used  
1106 only for writing, editing, or formatting purposes and does not impact the core methodology,  
1107 scientific rigorousness, or originality of the research, declaration is not required.

1108 Answer: [NA]

1109 Justification: *The core method development in this research does not involve LLMs as any*  
1110 *important, original, or non-standard components.*

1111 Guidelines:

- 1112 • The answer NA means that the core method development in this research does not  
1113 involve LLMs as any important, original, or non-standard components.
- 1114 • Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>)  
1115 for what should or should not be described.