ROIL: Robust Offline Imitation Learning

Gersi Doko
Gersi.Doko@unh.edu
University of New Hampshire

Guang Yang
u132644@utah.edu
University of Utah

Daniel Brown
daniel.s.brown@utah.edu
University of Utah

Marek Petrik
mpetrik@cs.unh.edu
University of New Hampshire

Abstract

We study the problem of imitation learning via inverse reinforcement learning where the agent attempts to learn an expert's policy from a dataset of collected state, action tuples. We derive a new Robust model-based Offline Imitation Learning method (ROIL) that mitigates covariate shift by avoiding estimating the expert's occupancy frequency. Frequently in offline settings, there is insufficient data to reliably estimate the expert's occupancy frequency and this leads to models that do not generalize well. Our proposed approach, ROIL, is a method that is guaranteed to recover the expert's occupancy frequency and is efficiently solvable as an LP. We demonstrate ROIL's ability to achieve minimal regret in large environments under covariate shift, such as when the state visitation frequency of the demonstrations does not come from the expert.

1 Introduction

Imitation learning seeks to compute an optimal policy in a Markov decision process (MDP) without knowing the reward function. Instead, one only has access to a set of demonstrations performed by a domain expert (Chang et al., 2021; Panaganti et al., 2023; Spencer et al., 2021; Rashidinejad et al., 2022). Imitation learning promises techniques that can learn to act well in environments where describing an appropriate reward function may be challenging or impractical. Robotics, medicine, and autonomous driving are examples of problem domains that can benefit greatly from more reliable imitation learning algorithms.

Inverse Reinforcement Learning (IRL), or apprenticeship learning, is a common approach to imitation learning (Abbeel and Ng, 2004; Ziebart et al., 2008; Fu et al., 2018). IRL often leverages the environment's dynamics, modeled as an MDP, to efficiently mimic the observed policy of the expert (Arora and Doshi, 2021). The environment's dynamics may be known a priori (Syed et al., 2008; Lacotte et al., 2019) or estimated from data (Finn et al., 2016; Ho and Ermon, 2016; Chang et al., 2021). An important strength of IRL algorithms is that they can learn to mimic experts quite well even with remarkably little data. However, most IRL algorithms can be very sensitive to the state distribution in the training data. If the distribution of states present in the dataset does not follow the expert's occupancy frequency—a phenomenon known as *covariate shift*—the IRL algorithm may compute a policy that is much worse than the expert's policy.

In this paper, we propose ROIL, a new approach to IRL that is particularly resistant to any covariate shift. In particular, ROIL allows for data with a state distribution that does not follow the expert's occupancy frequency. Most existing algorithms are sensitive to covariate shifts because, in some form, they reduce to matching the expert's state occupancy frequency. In comparison, ROIL attempts to recover the set of plausible expert policies from the training data and compute a policy that minimizes the regret with respect to this set of experts. With an appropriate choice of modeling

assumption, we show that ROIL can be formulated as a convex optimization problem and solved using mature solvers.

There are several reasons why the expert demonstration data may not be sampled according to the true occupancy frequency. First, the expert's initial state distribution may differ from the initial state distribution when the learned policy is deployed. Second, the expert may focus on providing pedagogic demonstrations that focus on the most challenging parts of the state space (Cakmak and Lopes, 2012; Hadfield-Menell et al., 2016; Brown and Niekum, 2018). Third, the demonstrations may not even form a trajectory but instead consist of disconnected state-action pairs. Finally, the state distribution of the demonstrations may differ simply due to sampling and model errors and inconsistencies. Thus, the demonstrations may not be representative of the expert's true policy. As a result, one must be careful in formalizing the IRL problem to make it tractable.

To better illustrate the importance of covariate shift, consider the following extreme example. In an MDP with a small state space and the ability to jump between states, the expert provides a single demonstration for each state showing the optimal actions. Behavior cloning algorithms, which reduce imitation learning to a classification problem, will recover the optimal policy given that the classification bias is general enough. Yet, surprisingly, common IRL algorithms based on the same scheme as LPAL (Linear Programming Apprenticeship Learning) (Syed et al., 2008) or GAIL (Generative Adversarial Imitation Learning) (Ho and Ermon, 2016) can fail to recover a good policy, as we show below. ROIL, on the other hand, recovers the optimal policy even in this extreme setting while preserving most of the benefits of the low sample complexity of IRL.

As with most IRL algorithms, ROIL seeks to minimize the regret given the expert's demonstrations for the worst-case plausible reward function. However, ROIL departs significantly from existing IRL algorithms in that it does not directly use the estimate of the expert's occupancy frequency. As a result, ROIL cannot be seen as matching the expert's feature frequencies, which is a popular view of existing IRL techniques (Abbeel and Ng, 2004; Syed et al., 2008; Ho and Ermon, 2016). In contrast, ROIL uses the training data to construct a robust set of plausible expert policies and minimizes the regret of the computed policy in the context of this set.

The remainder of the paper is organized as follows. Section 2 describes the background in MDPs and IRL necessary to introduce ROIL. Then, in Section 3, we describe our general framework, analyze its basic properties, propose an optimization algorithm, and discuss several practical extensions. Section 4 analyzes ROIL's guarantees and limitations theoretically and compares them with prior work. Finally, in Section 5, we analyze ROIL numerically and compare it with relevant algorithms.

2 Preliminaries

Before describing the underlying MDP framework and formally defining the IRL objective, we define the basic notation we use in the paper. We use calligraphic letters to denote sets and a tilde to denote random variables. We also adopt the standard convention that $\mathcal{A}^{\mathcal{B}}$ represents the set of all functions from a set $\mathcal{B}$ to a set $\mathcal{A}$ and treat vectors as a function from indexes to real numbers. Finally, the sets $\mathbb{R}$ and $\mathbb{R}_+$ represent real and non-negative numbers respectively.

We assume that the domain can be modeled as a *Markov Decision Process* (Puterman, 1994) with a finite number of states $\mathcal{S} = \{1, \dots, S\}$ and a finite number of actions $\mathcal{A} = \{1, \dots, A\}$. The transition probability function $p\colon \mathcal{S}\times\mathcal{A} \to \Delta^{\mathcal{S}}$, where $\Delta^{\mathcal{S}} = \left\{x \in \mathbb{R}_+^S \mid \sum_{s\in\mathcal{S}} x_s = 1\right\}$ is the probability simplex over the elements of the set $\mathcal{S}$. The reward function $r^\star\colon \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ represents the reward obtained in each transition. We assume that the initial distribution over $p_0 \in \Delta^S$ satisfies that $p_0 > 0$.

A solution to an MDP is a *policy*. In this work, we restrict our attention to *stationary randomized* and *deterministic* policies. The set of deterministic policies is $\Pi_{\text{D}} = \mathcal{A}^{\mathcal{S}}$ and the set of randomized policies is $\Pi_{\text{R}} = \left(\Delta^{\mathcal{A}}\right)^{\mathcal{S}}$. Note that deterministic policies are a special case of randomized policies.

The objective in this work is the γ-discounted infinite horizon objective with $\gamma \in [0, 1)$. We denote the infinite-horizon discounted *return* of a policy $\pi \in \Pi$ and a reward $r \in \mathcal{R}$ is denoted by

$$\rho(\pi, r) \;=\; \lim_{T \to \infty} \mathbb{E}^{\pi, p_0} \left[\sum_{t=0}^{T} \gamma^t r(\tilde{s}_t, \tilde{a}_t) \right],$$

where the superscript on the expectation indicates that $\tilde{s}_0 \sim p_0$ and $\tilde{s}_{t+1} \sim p(\cdot|\tilde{s}_t, \tilde{a}_t)$, and $\tilde{a}_t \sim \pi(\cdot|\tilde{s}_t)$. The return ρ is parameterized by the reward because, in the IRL setting, the reward is uncertain.

It will be convenient to treat functions that map states and actions to real numbers as vectors, such as the reward function $r^\star \in \mathbb{R}^{SA}$. We also use $P_\pi \in \mathbb{R}_+^{S \times S}$ where $P_\pi(s, \cdot) = \sum_{a \in \mathcal{A}} p(\cdot|s, a)\pi(a|s)$ and $r_\pi \in \mathbb{R}^S = \sum_{a \in \mathcal{A}} r(s, a)\pi(a|s)$ to represent the transition probability matrix and reward vector respectively for each policy $\pi \in \Pi$. Similarly, P_a and r_a represent the transition probability matrix and a reward vector respectively for each action $a \in \mathcal{A}$.

An important and well-known fact that we use is the relation between the occupancy frequencies and policies. In particular, for each policy $\pi \in \Pi_{\mathrm{R}}$ there exists an occupancy frequency $u^\pi \in \mathbb{R}^{\mathcal{S} \times \mathcal{A}}$ such that $\rho(\pi, r) = r_\pi^\mathsf{T} u^\pi$. The space of occupancy frequencies for all $\pi \in \Pi_{\mathrm{R}}$ is denoted as $\mathcal{U}$ and satisfies (Puterman, 1994, Section 6.9):

$$\mathcal{U} \;=\; \{u^\pi \mid \pi \in \Pi\} \;=\; \Big\{ u \in \mathbb{R}_+^{SA} \mid \sum_{a \in \mathcal{A}} (I - \gamma \cdot P_a^\mathsf{T}) \cdot u(\cdot, a) = p_0 \Big\}. \tag{1}$$

Finally, for each $u \in \mathcal{U}$, one can construct a policy π_u such that $u^\pi = u$ (Puterman, 1994, Theorem 6.9.1) as

$$\pi_u(a|s) \;=\; \frac{u(s, a)}{\sum_{a' \in \mathcal{A}} u(s, a')}, \qquad \forall s \in \mathcal{S}, a \in \mathcal{A}. \tag{2}$$

The policy π_{u} is well-defined because $p_0 > 0$ guarantees that $\sum_{a \in \mathcal{A}} u(s, a) > 0$ for each $s \in \mathcal{S}$.

With the definitions above, we are now ready to describe the general IRL framework (Abbeel and Ng, 2004; Syed et al., 2008; Ho and Ermon, 2016). Recall that the main goal is to learn to act in an environment without knowing the true reward function $r^\star$. Instead, we have access to transition data generated from an expert's policy $\pi_{\mathrm{e}} \in \Pi_{\mathrm{D}}$. To simplify the exposition, we assume that the expert follows a deterministic policy and we discuss generalizations to randomized policies in Section 3. The IRL algorithm has access to a dataset $\mathcal{D} = (t_i, s_i, \pi_{\mathrm{e}}(s_i))_{i=1}^D$, where the states may or may not be selected sequentially from state trajectories.

To generalize from a small set of demonstrations, IRL algorithms typically rely on a feature function $\phi \colon \mathcal{S} \times \mathcal{A} \to \mathbb{R}^k$ that assigns k features to each state and action (Abbeel and Ng, 2004; Syed et al., 2008; Lacotte et al., 2019; Chang et al., 2021; Jonnavittula and Losey, 2021; Arora and Doshi, 2021; Javed et al., 2021; Ghosal et al., 2023). The features ϕ represent important characteristics of the state-action pairs that may be part of the demonstrator's reward function. We can represent our features with a feature matrix $\Phi \in \mathbb{R}^{SA \times k}$ where each row represents the features of a specific state and action. Linear IRL algorithms assume that rewards can be expressed as a linear combination of state and action features. Formally, the set $\mathcal{R} \subseteq \mathbb{R}^{SA}$ of *feasible rewards* is defined as

$$\mathcal{R} = \{\Phi w \mid w \in \mathcal{W}\}, \quad \text{where} \quad \mathcal{W} = \left\{ w \in \mathbb{R}^k \mid \|w\|_1 \leq 1 \right\}. \tag{3}$$

The L_1 norm in the definition of $\mathcal{W}$ serves to normalize w because optimal policies are invariant to the scale of the rewards (Abbeel and Ng, 2004; Syed et al., 2008).

Most IRL algorithms adopt the following scheme. The true reward $r^\star$ is unknown but is assumed to satisfy that $r^\star \in \mathcal{R}$. Algorithms as varied as LPAL (Syed et al., 2008), GAIL (Ho and Ermon, 2016), and MILO (Chang et al., 2021) seek to compute a policy that minimizes the worst-case regret with respect the expert's policy. In its essence the regret minimization problem is usually formalized as

$$\min_{\pi \in \Pi} \max_{r \in \mathcal{R}} \Big(\rho(\hat{\pi}_{\mathrm{e}}, r) - \rho(\pi, r) \Big). \tag{4}$$

Here, $\hat{\pi}_{\mathrm{e}}$ is the empirical estimate of the expert policy π_{e} constructed from the dataset $\mathcal{D}$.

The conceptual optimization in (4) is impractical because the optimization over π is non-convex and computationally challenging. Instead, using the correspondence between policies and occupancy frequencies in (2), LPAL and related algorithms solve the following surrogate optimization problem:

$$\min_{u \in \mathcal{U}} \max_{r \in \mathcal{R}} \left(\hat{u}_{\mathrm{e}}^{\mathsf{T}} r - u^{\mathsf{T}} r \right) \;=\; \min_{u \in \mathcal{U}} \left\| (\hat{u}_{\mathrm{e}} - u)^{\mathsf{T}} \Phi \right\|_{\infty}, \tag{5}$$

where the equality follows because L_∞ is the dual norm to the L_1 norm used in the definition of $\mathcal{W}$.

The value $\hat{u}_{\mathrm{e}}$ in (4) represents the *empirical* estimate of $u_{\mathrm{e}}^{\star}$ constructed as

$$\hat{u}_{\mathrm{e}}(s,a) \;=\; \chi \cdot \sum_{(t,s',a') \in \mathcal{D}} \gamma^t \cdot \mathbb{1}\left\{ s = s' \wedge \; a = a' \right\}, \tag{6}$$

where χ is a normalization factor chosen to guarantee that $1^{\mathsf{T}} \hat{u}_{\mathrm{e}} = (1-\gamma)^{-1}$. In practice, it is common to estimate the *feature counts* $\hat{u}_{\mathrm{e}}^{\mathsf{T}} \Phi$ directly rather than estimating $\hat{u}_{\mathrm{e}}$.

Some IRL algorithms, like GAIL, add other regularization terms to the scheme in (5) and substitute different definitions for the reward set $\mathcal{W}$ (Ho and Ermon, 2016). In this work, we focus on the fundamental properties and trade-offs of this formulation and leave more complex extensions for future work.

An important limitation of the formulation in (5) is that it relies on estimating the expert's occupancy frequency $\hat{u}_{\mathrm{e}}$ well. Because the occupancy frequency represents the frequency of *both* states and actions it is very sensitive to the initial distribution and covariate shifts in state distributions which may often arise in imitation learning settings. As discussed in the introduction, the expert may focus on difficult states when performing the demonstrations or have a behavioral state visitation policy that dictates what states to visit. In the remainder of the paper, we build on (5) to address its sensitivity to the initial distribution and state distribution of the provided dataset, thereby achieving better off-policy performance.

3 ROIL Formulation

In this section, we describe and justify ROIL and study its computational properties; we defer the analysis of its approximation errors to Section 4. First, we describe the foundations of the approach in Section 3.1 and then outline several modifications that reduce ROIL's conservativeness and improve its performance in Section 3.2. We conclude the section with a visualization of ROIL as a Chebyshev center problem, which offers additional insights into its performance in Section 3.3.

3.1 Basic Formulation

Similar to the standard IRL schema outlined in (5), ROIL also adopts a principled robust optimization perspective and minimizes the worst-case regret. The main idea is to compute a policy $\pi \in \Pi$ that minimizes regret with respect to the worst-case plausible expert's policy $\pi_{\mathrm{e}} \in \Pi_{\mathrm{R}}(\mathcal{D})$ and a reward function $r \in \mathcal{R}$. Formally, the basic ROIL optimization problem is as follows:

$$\min_{\pi \in \Pi_{\mathrm{R}}(\mathcal{D})} \max_{\pi_{\mathrm{e}} \in \Pi_{\mathrm{R}}(\mathcal{D})} \max_{r \in \mathcal{R}} \left(\rho(\pi_{\mathrm{e}}, r) - \rho(\pi, r) \right). \tag{7}$$

Here, $\Pi_{\mathrm{R}}(\mathcal{D}) \subseteq \Pi_{\mathrm{R}}$ represents the set of all policies consistent with $\mathcal{D}$ and are defined as

$$\Pi_{\mathrm{R}}(\mathcal{D}) = \left\{ \pi \in \Pi_{\mathrm{R}} \mid \pi(a|s) = 1, \, \forall (s,a) \in \mathcal{D} \right\}. \tag{8}$$

If the expert demonstrations in $\mathcal{D}$ are constructed from a deterministic policy, then that policy must be contained in (8). However, when the expert's policy is randomized, the construction in (8) may exclude the expert policy from $\Pi_{\mathrm{R}}(\mathcal{D})$. We discuss how the definition can be extended to account for randomized policies in Section 3.2.

Before describing an efficient formulation for solving ROIL, we discuss its benefits compared with the generic IRL scheme in (4). Recall that $\hat{u}_{\mathrm{e}}$, constructed in (6), depends on the initial state distribution which may lead to large errors when the demonstration and execution state distributions differ. Instead, ROIL uses the training data to construct the set $\Pi_{\mathrm{R}}(\mathcal{D})$—which is independent of the state distribution—and minimizes regret to all consistent expert policies.

Next, we show that the ROIL optimization problem in (7) can be reduced to a linear program with a polynomial size. It may be surprising that such a reduction is possible since ROIL's objective involves maximizing a non-concave bilinear function. We derive this reduction using the occupancy-based formulation, similar to existing IRL algorithms. A key part of the formulation is a set of occupancy frequencies Υ that are consistent with expert demonstrations defined for $c \in \mathbb{R}^{SA}$ as

$$\Upsilon = \left\{ u \in \mathcal{U} \mid c^{\mathsf{T}} u = 0 \right\}, \quad \text{where} \quad c(s,a) = \begin{cases} 1 & \text{if } (s,a) \notin \mathcal{D} \wedge \exists a' \in \mathcal{A}, (s,a') \in \mathcal{D}, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

The following lemma shows that the set of occupancy frequencies constructed in (9) is exactly the set of frequencies of policies that are consistent with the dataset.

Lemma 1. *If $\mathcal{D}$ is generated by a deterministic policy $\pi \in \Pi_{\mathrm{D}}$. Then*

$$u \in \Upsilon \qquad \Leftrightarrow \qquad \left(u = u^{\pi},\ \exists \pi \in \Pi_{\mathrm{R}}(\mathcal{D})\right).$$

Please see the proof in Appendix A.

We now outline the main step in constructing a linear program formulation for solving ROIL. As Lemma 1 shows, maximizing over the policy space is equivalent to maximizing over the occupancy frequency space. Then, using the fact that $\rho(\pi, r) = r^{\mathsf{T}} u^{\pi}$ and the representation of $\mathcal{R}$ in (3), we can reformulate (7) to

$$\min_{u \in \Upsilon} \max_{r \in \mathcal{R}} \max_{v \in \Upsilon} (v-u)^{\mathsf{T}} r \ = \ \min_{u \in \Upsilon} \max_{w \in \mathcal{W}} \max_{v \in \Upsilon} (v-u)^{\mathsf{T}} \Phi w. \tag{10}$$

Solving the formulation in (10) directly is challenging because it involves maximizing a non-concave bilinear function in both w and v. To turn this optimization into a tractable convex optimization problem, we take the following steps. The maximization over w maximizes a convex function $w \mapsto \max_{v \in \Upsilon} (v-u)^{T} \Phi w$. Therefore, there exists an optimal w in one of the extreme points of $\mathcal{W}$, leading to the following equivalent formulation:

$$\min_{u \in \Upsilon} \max_{w \in \mathrm{ext}(\mathcal{W})} \left(-u^{\mathsf{T}} \Phi w + \max_{v \in \Upsilon} v^{\mathsf{T}} \Phi w \right). \tag{11}$$

The set $\mathcal{W}$ is an L_1-norm ball. The number of its extreme points is linear in the number of features, and we can enumerate them to obtain the following linear program:

$$\begin{array}{ll} \underset{t \in \mathbb{R}, u \in \mathbb{R}^{SA}}{\text{minimize}} & t \\ \text{subject to} & t \geq -u^{\mathsf{T}} \Phi w + b(w), \quad \forall w \in \mathrm{ext}(\mathcal{W}), \\ & u \in \Upsilon, \end{array} \tag{12}$$

where $b(w) = \max_{v \in \Upsilon} v^{\mathsf{T}} \Phi w$. Note that the constraints $u \in \mathcal{U}$ defined in (1) are linear and $b(w)$ can be computed by solving a linear program or an MDP.

From the discussion above and the epigraph formulation (Boyd and Vandenberghe, 2004), we get the following theorem that states the correctness of the linear program formulation.

Proposition 1. *Let $u^{\star}$ be optimal in (12). Then, $\pi_{u^{\star}}$ constructed in (2) is an optimal π in (7).*

3.2 Extensions

An important strength of ROIL is its flexibility. The basic ROIL formulation makes no assumptions except that actions in $\mathcal{D}$ are distributed according to the expert's policy. It also does not make any assumptions about the state distribution or the optimality of the expert's policy. In this section, we discuss several simple techniques that can be used to incorporate additional assumptions about the data in $\mathcal{D}$ that help to compute less conservative solutions.

First, we can make ROIL less conservative by restricting the generic reward set $\mathcal{R}$ when computing the regret. If the expert's policy π_{e} is close to optimal, it is sufficient to consider only a subset of $\mathcal{R}$ restricted to rewards that are consistent with the near-optimality of π_{e}. That is, we can solve (10) with $\mathcal{W}_{\mathrm{e}}^{\tau} \subseteq \mathcal{W}$ defined as

$$\mathcal{W}_{\mathrm{e}}^{\tau} \;:=\; \left\{ w \in \mathcal{W} \mid \hat{u}_{\mathrm{e}}^{\mathsf{T}} \Phi w + \tau \geq \max_{u \in \mathcal{U}} u^{\mathsf{T}} \Phi w \right\}, \tag{13}$$

where $\tau \geq 0$ represents the allowed sub-optimality of the expert's policy π_{e}. It is important to emphasize that the optimality of a policy is insensitive to the choice of the initial distribution. In practice, we adapt (12) to solve

$$\begin{array}{ll} \underset{t \in \mathbb{R}, u \in \mathbb{R}^{SA}}{\text{minimize}} & t \\ \text{subject to} & t \geq -u^{\mathsf{T}} \Phi w + b(w), \quad \forall w \in \mathcal{W}_{\mathrm{e}}^{\tau}, \\ & u \in \Upsilon. \end{array} \tag{14}$$

We solve this optimization problem by first collecting m samples $w \in \mathcal{W}_{\mathrm{e}}^{\tau}$ from a uniform distribution and then choose τ in (13) appropriately to retain 10% of the samples.

A second option for reducing the conservativeness of ROIL is to suppose that the expert's demonstrations are close to being on-policy. Then, one can use $\mathcal{D}$ to estimate $\hat{u}_{\mathrm{e}} \approx u_{\mathrm{e}}^{\star}$ and consider the set $\hat{\Upsilon}_{\epsilon}$ defined as

$$\hat{\Upsilon}_{\epsilon} \;=\; \left\{ u \in \Upsilon \mid \|(\hat{u}_{\mathrm{e}} - u)^{\mathsf{T}} \Phi\|_{\infty} \leq \varepsilon \right\}. \tag{15}$$

This set represents all policies that are consistent with the expert's demonstrations and also have occupancy frequencies that are close to the observed expert data. One can estimate ϵ by solving

$$\epsilon \;=\; \eta \cdot \min_{u \in \mathcal{U}} \|(\hat{u}_{\mathrm{e}} - u)^{\mathsf{T}} \Phi\|_{\infty}, \tag{16}$$

where $\eta \geq 1$ is some constant. The linear program in (12) can be easily adapted to handle the restricted set $\hat{\Upsilon}_{\mathrm{e}}$ by redefining $b(w) = \max_{v \in \hat{\Upsilon}_{\mathrm{e}}} v^{\mathsf{T}} \Phi w$.

Finally, we discuss how to extend ROIL to account for an expert policy which randomizes between actions. In such a scenario, the constraint $c^{\mathsf{T}} u = 0$ must be replaced by a constraint $u_{s,a} / \sum_{a' \in \mathcal{A}} |u(s,a') - \hat{\pi}_{\mathrm{e}}(s,a)| \leq \epsilon, \forall s \in \mathcal{S}, a \in \mathcal{A}$ for some appropriately chosen ϵ and an estimate $\hat{\pi}_{\mathrm{e}}$ of expert's policy. Using perspective functions, one can readily see that this constraint is convex and does not increase the computational complexity of this formulation.

3.3 Discussion and Visualization

We now discuss a connection between ROIL and the geometric problem of computing the Chebyshev center of a convex set. This connection helps to elucidate what conditions make ROIL tractable and offers an intuitive way of visualizing ROIL and its relationship with other IRL algorithms.

In (3), we define the set $\mathcal{W}$ in terms of an L_1 ball. However, this set could be defined in terms of any norm $\|\cdot\|$ as $\mathcal{W} = \left\{ w \in \mathbb{R}^k \mid \|w\| \leq 1 \right\}$. For any norm, there exists a *dual norm* $\|\cdot\|_{\star}$ defined as $\|x\|_{\star} \;=\; \sup_{y \neq 0} {}^{y^{\mathsf{T}} x} / \|y\|$. (Horn and Johnson, 2013). The dual to an L_p norm ($p \geq 1$) is an L_q norm such that ${}^1/_p + {}^1/_q = 1$. Using the definition of a dual norm, the ROIL optimization problem in (10) can be represented as

$$\min_{u \in \mathcal{U}} \max_{v \in \Upsilon} \|(v - u)^{\mathsf{T}} \Phi\|_{\star}. \tag{17}$$

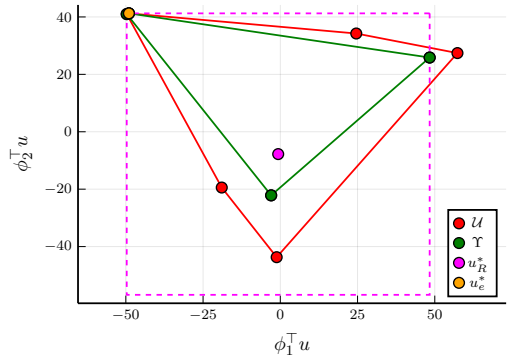

Figure 1: Depiction of ROIL solution (u_R^*) as a Chebyshev center of the set Υ. The dashed line shows the minimum circumscribed L_∞ ball.

The following proposition states the correctness of (17) and follows from the discussion above.

Proposition 2. *Suppose that $u^\star$ is optimal in (17) with $\|z\|_\star = \|z\|_\infty$ for $z \in \mathbb{R}^k$. Then $\pi_{u^\star}$, as constructed in (2), is an optimal π in (7).*

The optimization problem in (17) is equivalent to computing the *Chebyshev center* of the set Υ with respect to the norm defined by $\|\cdot\|_\infty$. The Chebyshev center is a point that minimizes the distance to the most distant point in the set Υ. Figure 1 visualizes the Chebyshev center for an MDP with two features. The red polygon represents the set $\mathcal{U}$ and the green polygon represents the set Υ; the points correspond to deterministic policies.

The relationship to the Chebyshev center problem also offers additional computational insights regarding the choice of $\mathcal{W}$. It is known that popular choices of the distance metric $\|\cdot\|_\star$ in computing the Chebyshev center of a polyhedron are NP-hard. One notable exception is when the distance metric $\|\cdot\|_\star$ corresponds to the L_∞ norm. Since L_1 is the dual norm to the L_∞ norm, the choice of L_1 in the definition of $\mathcal{W}$ is crucial to obtaining a tractable optimization problem (Wu et al., 2013; Eldar et al., 2008).

4 Theoretical Analysis

In this section, we turn to a theoretical analysis of ROIL. We study the theoretical guarantees of the quality of the solutions computed by ROIL. In particular, we show that, unlike other popular IRL algorithms, ROIL guarantees to recover the expert's policy when demonstrations for all states are available. We also discuss the limitations that arise from the assumption inherent in ROIL formulations and give an approximation error bound in terms of the approximation error bounds.

First, we show that LPAL and GAIL, popular IRL algorithms, suffer from a surprising weakness. The algorithms may not recover the expert's policy even when given demonstrations of deterministic actions for *every* state in a tabular MDP. While it is not a prevalent scenario in practice, it points out that simply adding more demonstrations is insufficient for these methods. We consider LPAL and GAIL, which can be stated for tabular features $\Phi = I$ as the following optimization problems:

$$\min_{u \in \mathcal{U}} \|u - \hat{u}_e\|_\infty, \qquad \text{and} \qquad \min_{u \in \mathcal{U}} D_{\mathrm{JS}}(u, \hat{u}_e) - \lambda H(\pi^u). \tag{18}$$

Here, the first optimization problem represents LPAL (Syed et al., 2008) and the second optimization represents GAIL (Ho and Ermon, 2016, eq. (15)). The distance metric D_{JS} represents the Jensen-Shannon entropy, and $\lambda \geq 0$ is a regularization parameter. The LPAL optimization problem in (18) follows immediately from (5) by optimizing over the set of occupancy frequencies. For the sake of consistency with ROIL, we assume that $\mathcal{W}$ is chosen as in (3). This is a superficial

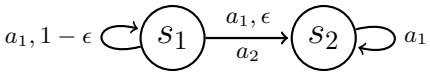

Figure 2: MDP used in Example 1. The edge labels denote the actions and the corresponding transition probabilities (if less than 1).

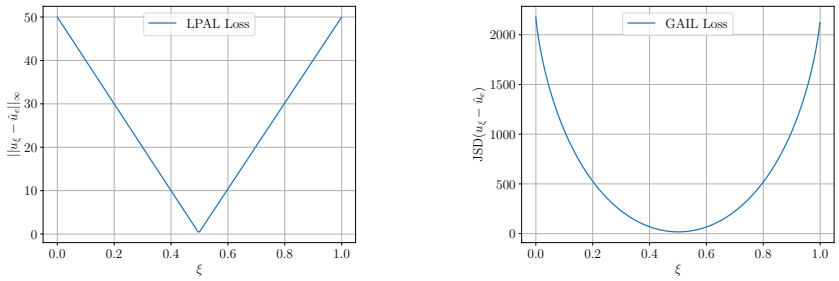

Figure 3: The loss functions of LPAL and GAIL. Here JSD refers to the Jensen-Shannon Divergence which is the loss function minimized by GAIL when the coefficient of the causal entropy term H is zero (Ho and Ermon, 2016).

difference from the original LPAL derivations that assume that feature weights are non-negative: $\mathcal{W} = \{w \in \mathbb{R}^k_+ \mid \|w\|_1 \leq 1\}$.

Proposition 3. *LPAL and GAIL as defined in* (18) *may not recover* π_e *even when the demonstrations cover the entire state space:* $\{s \in \mathcal{S} \mid \exists\, a \in \mathcal{A},\ (s,a) \in \mathcal{D}\} = \mathcal{S}$.

We show the proposition by constructing the following example.

Example 1. Consider an MDP with two states and transition probabilities depicted in Figure 2. Suppose that $\pi_e(s) = a_1$ for each $s \in \mathcal{S}$. The occupancy frequency for this policy is $u_e^\star = \left[\frac{\epsilon+\gamma-1}{\epsilon(1-\gamma)}, 0, \frac{1}{\epsilon}\right]$. Assume that the dataset $\mathcal{D} = ((s_1,a_1),(s_2,a_1))$ represents the demonstrations; note that the state distribution needs to respect the state distribution of $u_e^\star$. The estimated occupancy frequency from this dataset will be $\hat{u}_e = \left[\frac{1}{2(1-\gamma)}, 0, \frac{1}{2(1-\gamma)}\right]$ where the elements correspond to $(s_1,a_1),(s_1,a_2),(s_2,a_1)$. The set of occupancy frequencies in this MDP is

$$\mathcal{U}_\xi = \left\{ \xi \cdot \left[\frac{\epsilon+\gamma-1}{\epsilon(1-\gamma)}, 0, \frac{1}{\epsilon}\right] + (1-\xi)\cdot\left[0, 1-\epsilon, \frac{\gamma}{1-\gamma}+\epsilon\right] \mid \xi \in [0,1] \right\}, \tag{19}$$

because the set of occupancy frequencies of randomized policies can be represented as a convex hull of the frequencies of deterministic policies. One can then readily verify that $u_e^\star$ does not minimize either one of the objectives in (18) when $\lambda = 0$. Specifically, choosing $\xi = 0.5$ in (19) achieves minimal loss however one can easily verify $u_e^\star = u_\xi$ when $\xi = 1$. Figure 3 depicts the objective functions in (18) as a function of ξ in (19).

In contrast with LPAL and GAIL, ROIL is guaranteed to recover the expert's policy when provided with demonstrations for all states as the following proposition states.

Proposition 4. *Suppose that* $\{s \in \mathcal{S} \mid (\cdot, s, \cdot) \in \mathcal{D}\} = \mathcal{S}$. *Then* $u_e^\star$ *is the unique minimizer to* (12).

Proof. When $\mathcal{D}$ completely covers the states, $\Pi_R(\mathcal{D}) = \{\pi_e\}$ and $\Upsilon = \{u_e^\star\}$ by Lemma 1. One can readily see that $u_e^\star$ attains 0 objective in (12), which is optimal because the regret is lower-bounded by 0. The uniqueness is immediate because Υ is a singleton. □

Recall that ROIL dispenses with the assumption that the states in the demonstrations $\mathcal{D}$ are distributed according to the occupancy frequency. This assumption makes ROIL appropriate in a

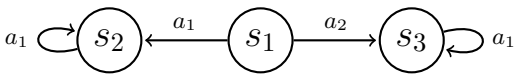

Figure 4: An MDP used in Example 2.

broader range of off-policy scenarios than competing IRL algorithms. However, the following result shows the limitations arising from ignoring the state distribution assumption in $\mathcal{D}$. The following example demonstrates that even when all but one state is covered by $\mathcal{D}$, ROIL cannot give any guarantees on the regret of the computed policy.

Example 2. Consider the deterministic MDP depicted in Figure 4 where $\mathcal{S} = \{s_1, s_2, s_3\}$, $\mathcal{A} = \{a_1, a_2\}$, $p_0 = [1, 0, 0]$, and $\Phi = r^\star = [0, 0, 1, 1, -1, -1]$. Here, $r^\star$ is the true reward and vectors are ordered as $(s_1, a_1), (s_1, a_2), (s_2, a_1), \ldots$. Assume that the expert follows the optimal policy $\pi_{\mathrm{e}}(s) = a_1, \forall s \in \mathcal{S}$ with an occupancy frequency $u_{\mathrm{e}}^\star = [1, 0, \gamma/1-\gamma, 0, 0, 0]$. However, ROIL fails to find this solution even when demonstrations cover all but one state. Consider the dataset $\mathcal{D} = ((s_2, a_1), \ldots, (s_2, a_1))$. The optimal solution to ROIL is $u = [1/2, 1/2, \gamma/2(1-\gamma), 0, \gamma/2(1-\gamma), 0]$ which is sub-optimal regardless of how well the estimated $\hat{u}_{\mathrm{e}}$ approximated $u_{\mathrm{e}}^\star$. Using the observed data, ROIL has no evidence supporting taking actions a_1 or a_2 in the initial state s_1.

Example 2 exposes a limitation of ROIL but also hints at how to overcome it. Note that occupancy frequency matching methods, like LPAL (Syed et al., 2008), may do well in Example 2. This is because LPAL will use the prevalence of the state s_2 in $\mathcal{D}$ to deduce that taking action a_1 in s_1 is preferable to a_2. As we discuss in Section 3.2, it is easy to extend ROIL to benefit from similar distributional assumptions. The following theorem establishes approximation bounds for ROIL with this assumption.

Theorem 1. *Suppose that $u_{\mathrm{r}}^\star$ is an optimal solution to (12) with $\hat{\Upsilon}_\epsilon$ some $\epsilon > 0$ such that $\hat{\Upsilon}_\epsilon \neq \emptyset$ and an occupancy frequency estimate $\hat{u}_{\mathrm{e}}$. Then the regret of $\pi_{\mathrm{r}}^\star = \pi^{u_{\mathrm{r}}^\star}$ is bounded as*

$$\rho(\pi_{\mathrm{e}}, r) - \rho(\pi_{\mathrm{r}}^\star, r) \;\leq\; \|(u_{\mathrm{e}}^\star - \hat{u}_{\mathrm{e}})^\mathsf{T}\Phi\|_\infty + \varepsilon, \qquad \forall r \in \mathcal{R}.$$

Moreover, one can choose ϵ such that $\epsilon = \|(u_{\mathrm{e}}^\star - \hat{u}_{\mathrm{e}})^\mathsf{T}\Phi\|_\infty$.

Proof. The result follows by replacing the worst-case over the L_1 ball by its dual norm (L_∞), and from the construction of $\mathcal{W}$, and the triangle inequality:

$$\begin{aligned}\rho(\pi_{\mathrm{e}}, r) - \rho(\pi_{\mathrm{r}}^\star, r) &\leq \max_{r \in \mathcal{R}} (u_{\mathrm{e}}^\star - u_{\mathrm{r}}^\star)^\mathsf{T} r = \|(u_{\mathrm{e}}^\star - u_{\mathrm{r}}^\star)^\mathsf{T}\Phi\|_\infty \leq \|(u_{\mathrm{e}}^\star - \hat{u}_{\mathrm{e}} + \hat{u}_{\mathrm{e}} - u_{\mathrm{r}}^\star)^\mathsf{T}\Phi\|_\infty \\ &\leq \|(u_{\mathrm{e}}^\star - \hat{u}_{\mathrm{e}})^\mathsf{T}\Phi\|_\infty + \|(\hat{u}_{\mathrm{e}} - u_{\mathrm{r}}^\star)^\mathsf{T}\Phi\|_\infty \leq \|(u_{\mathrm{e}}^\star - \hat{u}_{\mathrm{e}})^\mathsf{T}\Phi\|_\infty + \varepsilon.\end{aligned}$$

The last inequality follows from the fact that the $u_{\mathrm{r}}^\star \in \hat{\Upsilon}_\epsilon$. □

Theorem 1 shows that when $\|(u_{\mathrm{e}}^\star - \hat{u}_{\mathrm{e}})^\mathsf{T}\Phi\|_\infty$ is small, then ROIL with the extensions is guaranteed to find a policy that has a small regret to the expert's policy. We also note that Theorem 1 essentially matches the error bounds derived for LPAL (Syed et al., 2008).

5 Experimental Results

In this section, we study ROIL's behavior numerically on common benchmark problems. We study its performance both on-policy (states distributed according to the true expert policy) and off-policy (states distributed arbitrarily) and compare it with closely related IRL algorithms.

The first domain we use is an instance of the standard grid world problem (Abbeel and Ng, 2004) in which each square is designated a color that represents the feature that is active for the state. The reward is some linear combination of the features for each state. That is, the matrix Φ represents the state colors and $r^\star = \Phi^\mathsf{T} w$ for some $w \in \mathcal{W}$. The features for each action in the state are identical.

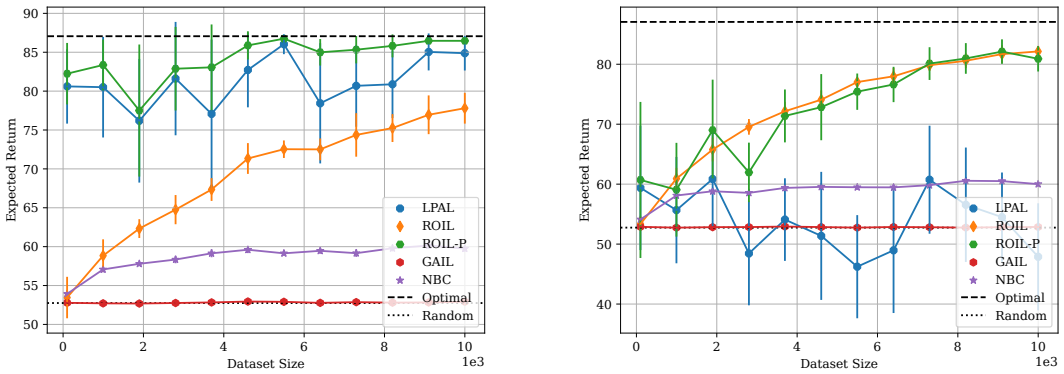

Figure 5: On-policy (left) and off-policy (right) expected return of imitation learning methods on 40x40 grid world for a fixed reward. Higher is better.

The agent has a choice of up, down, left, and right as their actions with small noise that takes it to a different neighboring state. The rewards in this domain are generated randomly for each run by sampling w uniformly from $\mathcal{W}$. The expert policy is computed and chosen as the optimal policy for the true (unobserved) reward.

The second domain we use is a driving simulator inspired by prior work (Abbeel and Ng, 2004; Syed et al., 2008; Brown et al., 2018; Trinh et al., 2024) where the agent begins in the bottom row and can go straight up, up and to the left, or up and to the right. At the first row, the actor loops back to the bottom row to simulate a continuous environment. Similarly to the grid world, the driving simulator has some small noise in the transitions. The driving simulator has some motorists on the road, which the actor must avoid, and the left-most and right-most columns are designated as "offroad" where the actor receives negative rewards.

To generate on-policy data, we use the standard protocol in which expert demonstrations are trajectories of a policy. To generate off-policy data, we collect states according to a uniform behavior policy. That is, the expert follows a uniform behavior policy $\pi_b(a|s) = 1/|\mathcal{A}|$, which controls the transition dynamics. The uniform policy π_b is only used to generate the states in $\mathcal{D}$ and the actions are chosen by the true expert π_e.

We evaluate two versions of ROIL: The basic ROIL makes no assumptions on $\hat{u}_\text{e}$ and solves (12). ROIL-P solves (14), pruning away reward functions that make $\hat{u}_\text{e}$ perform sub-optimally see Equation (14). We compare these algorithms with two IRL algorithms: LPAL and GAIL. For consistency with our results, we do not impose the constraint $w \geq 0$ used in the original LPAL formulation (Syed et al., 2008). For the GAIL implementation, we use the original formulation with $\lambda = 0$; we did not find that λ had a significant effect on our results. We also compare it with Naive Behavioral Cloning (NBC). NBC follows the expert's policy in states that are visited but takes a random action in states that have not been visited.

Figures 5 and 6 depict the performance of multiple IRL methods as a function of the number of samples in the demonstrations. The samples are constructed from trajectories sampled from the domain. Each data point is computed as an average of 10 seeds, and standard error bars are displayed; see the appendix for more details. We do not provide timing data because most of the algorithms are implemented in Python, and the main focus of our methods is for a setting where sample complexity and not computation time are the limiting factors.

LPAL performs very well on policy in our experiments. This is unsurprising because it matches $\hat{u}_\text{e}$, and its estimate improves with increasing samples. However, in the off-policy regime, LPAL and other occupancy frequency matching methods fail to work well because the estimate $\hat{u}_\text{e}$ does not

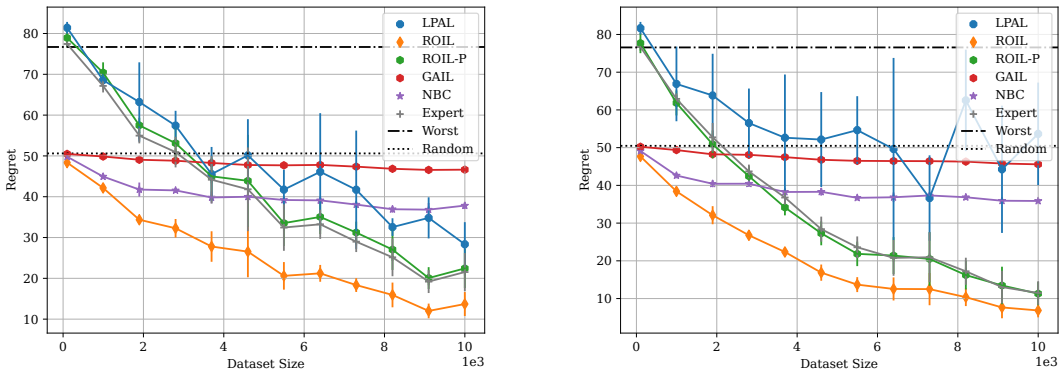

Figure 6: Robust regret for on-policy (left) and off-policy (right) of imitation learning methods on 40x40 Grid World (lower is better).

necessarily improve with increasing numbers of samples. ROIL, on the other hand, works well in both settings.

Our experiments demonstrate that ROIL and ROIL-P perform well both on-policy and off-policy. Because ROIL minimizes the worst-case regret, it can be quite conservative when the dataset is small. In comparison, our results confirm that pruning the reward vectors in ROIL-P makes it less conservative and improves its performance significantly. Additional empirical evaluation and discussion of methods described in Section 3.2 can be found in Appendix B.

6 Conclusion

We presented a new algorithm for IRL that can handle expert demonstrations gathered from off-policy (or offline) state distributions which may not form a trajectory. This is an important topic that, to the best of our knowledge, has not received sufficient attention in prior work. We proposed ROIL, a principled and flexible framework for this problem. ROIL minimizes the regret concerning the expert's policy and makes minimal assumptions about the data and the expert. However, the framework can be easily extended to a setting in which one makes more assumptions about the expert and the demonstrations generated. We address a surprising weakness with other IRL methods like LPAL and GAIL and provide guarantees on our convergence to the expert policy when all states are observed while the existing algorithms may not.

There are many avenues for future work. ROIL builds on the same ideas as most modern IRL algorithms and can be readily integrated with the improvements developed in recent years (Arora and Doshi, 2021). It is important to study whether there are possible refinements of ROIL along the lines described in Section 3.2 that would significantly impact its performance. We also studied ROIL in a simple tabular setting. Future work should study the best ways to generalize these ideas to large problems with continuous states and actions and non-linear function approximators.

Acknowledgments

This work was supported, in part, by NSF grants 2144601 and 2218063.

References

P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2004.

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

Boyd and Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

D. S. Brown and S. Niekum. Machine teaching for inverse reinforcement learning: Algorithms and applications. In *AAAI Conference on Artificial Intelligence*, 2018.

D. S. Brown, Y. Cui, and S. Niekum. Risk-aware active inverse reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2018.

M. Cakmak and M. Lopes. Algorithmic and human teaching of sequential decision tasks. In *AAAI Conference on Artificial Intelligence*, pages 1536–1542, 2012.

J. D. Chang, M. Uehara, D. Sreenivas, R. Kidambi, and W. Sun. Mitigating covariate shift in imitation learning via offline data without great coverage, 2021.

Y. C. Eldar, A. Beck, and M. Teboulle. A minimax Chebyshev estimator for bounded error estimation. *IEEE Transactions on Signal Processing*, 56(4):1388–1397, 2008.

C. Finn, S. Levine, and P. Abbeel. Guided cost learning: Deep inverse optimal control via policy optimization. In *International conference on machine learning*, pages 49–58. PMLR, 2016.

J. Fu, K. Luo, and S. Levine. Learning robust rewards with adverserial inverse reinforcement learning. In *International Conference on Learning Representations*, 2018.

G. R. Ghosal, M. Zurek, D. S. Brown, and A. D. Dragan. The effect of modeling human rationality level on learning rewards from multiple feedback types. In *AAAI Conference on Artificial Intelligence*, volume 37, pages 5983–5992, 2023.

Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2023. URL https://www.gurobi.com.

D. Hadfield-Menell, S. J. Russell, P. Abbeel, and A. Dragan. Cooperative inverse reinforcement learning. *Advances in neural information processing systems*, 29, 2016.

J. Ho and S. Ermon. Generative adversarial imitation learning. *Advances in Neural Information Processing Systems*, 2016.

R. A. Horn and C. A. Johnson. *Matrix Analysis*. Cambridge University Press, 2nd edition, 2013.

Z. Javed, D. S. Brown, S. Sharma, J. Zhu, A. Balakrishna, M. Petrik, A. Dragan, and K. Goldberg. Policy gradient bayesian robust optimization for imitation learning. In *International Conference on Machine Learning*, pages 4785–4796. PMLR, 2021.

A. Jonnavittula and D. P. Losey. I know what you meant: Learning human objectives by (under) estimating their choice set. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2747–2753, 2021.

J. Lacotte, M. Ghavamzadeh, Y. Chow, and M. Pavone. Risk-sensitive generative adversarial imitation learning. In *Artificial Intelligence and Statistics (AISTATS)*, 2019.

K. Panaganti, Z. Xu, D. Kalathil, and M. Ghavamzadeh. Distributionally robust behavioral cloning for robust imitation learning. In *Conference on Decision and Control*, 2023.

M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley; Sons, Inc., 1st edition, 1994.

P. Rashidinejad, B. Zhu, C. Ma, J. Jiao, and S. Russell. Bridging offline reinforcement learning and imitation learning: A tale of pessimism. *IEEE Transactions on Information Theory*, 68(12): 8156–8196, 2022.

J. Spencer, S. Choudhury, A. Venkatraman, B. Ziebart, and J. A. Bagnell. Feedback in imitation learning: The three regimes of covariate shift. *arXiv:2102.02872 [cs, stat]*, 2021.

U. Syed, M. Bowling, and R. E. Schapire. Apprenticeship learning using linear programming. In *International Conference on Machine Learning*, pages 1032–1039, 2008.

T. Trinh, H. Chen, and D. S. Brown. Autonomous assessment of demonstration sufficiency via bayesian inverse reinforcement learning. In *ACM/IEEE International Conference on Human-Robot Interaction*, pages 725–733, 2024.

D. Wu, J. Zhou, and A. Hu. A new approximate algorithm for the Chebyshev center. *Automatica*, 49(8):2483–2488, 2013.

B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey, et al. Maximum entropy inverse reinforcement learning. In *AAAI Conference on Artificial Intelligence*, pages 1433–1438. Chicago, IL, USA, 2008.