Robust Imitation via Mirror Descent Inverse Reinforcement Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recently, adversarial imitation learning has shown a scalable reward acquisition method for inverse reinforcement learning (IRL) problems. However, estimated reward signals often become uncertain and fail to train a reliable statistical model since the existing methods tend to solve hard optimization problems directly. Inspired by a first-order optimization method called mirror descent, this paper proposes to predict a sequence of reward functions, which are iterative solutions for a constrained convex problem. IRL solutions derived by mirror descent are tolerant to the uncertainty incurred by target density estimation since the amount of reward learning is regulated with respect to local geometric constraints. We prove that the proposed mirror descent update rule ensures robust minimization of a Bregman divergence in terms of a rigorous regret bound of $\mathcal{O}(1/T)$ for step sizes $\{\eta_t\}_{t=1}^{T}$. Our IRL method was applied on top of an adversarial framework, and it outperformed existing adversarial methods in an extensive suite of benchmarks.

1 Introduction

One crucial requirement of practical imitation learning algorithms is *robustness*, often described as learning expert behavior from finite number of demonstrations, overcoming various realistic challenges [1]. In real-world problems such as motor control tasks, the demonstration size can be insufficient to create a precise model of an expert [2], and even in some cases, demonstrations can be noisy or suboptimal to solve the problem [3]. For such challenging scenarios, imitation learning algorithms inevitably struggle with unreliable statistical models; thus, the way of handling the uncertainty of estimated cost functions dramatically affects the imitation performance. Therefore, a thorough analysis of addressing these issues is required to construct a robust algorithm.

Inverse reinforcement learning (IRL) is an algorithm for learning ground-truth rewards from expert demonstrations where the expert acts optimally with respect to an unknown reward function [4]. Traditional IRL methods solve the imitation problem based on iterative algorithms [5, 6], alternating between the reward estimation process and a reinforcement learning (RL) [7] algorithm. Newer studies of adversarial imitation learning (AIL) [8, 9] rather suggest estimating the ground-truth reward function "directly," by using adversarial learning objectives [10] and nonlinear discriminative neural networks [9]. Compared to classical approaches, the AIL methods have shown great success on control benchmarks in terms of scalability for challenging control tasks [11].

Technically, it is well known that AIL formulates a divergence minimization problem with its discriminative signals, which incorporates fine-tuned estimations of the target densities [12]. Through the lens of differential geometries, the limitation of AIL naturally comes from the implication that minimizing the divergence does not guarantee unbiased progression due to constraints of the underlying space [13]. In order to ensure stability, we argue that an IRL algorithm's progress needs to be regulated, yielding gradual updates with respect to local geometries of policy distributions.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

We claim that there are two issues leading to unconstrained policy updates: **(1)** a statistical divergence often cannot be accurately obtained for challenging problems and **(2)** an immediate divergence between agent and expert densities does not guarantee unbiased learning directions. Our approach is connected to a collection of optimization processes called mirror descent (MD) [14]. For a sequence of parameters $\{w_t\}_{t=1}^{T}$ and a convex function Ω, an MD update for a cost function F_t is derived as

$$\nabla\Omega(w_{t+1}) = \nabla\Omega\big(w_t\big) - \eta_t \nabla F_t\big(w_t\big). \tag{1}$$

In the equation, the gradient $\nabla\Omega(\,\cdot\,)$ creates a transformation that links a parametric space to its dual space. Theoretically, MD is a first-order method for solving constrained problems which enjoys rigorous regret bounds for various geometries [15, 16], including probability spaces. Thus, applying MD to the reward estimation can be efficient in terms of the number of learning phases.

In this paper, we derive an MD update rule in IRL with a postulate of nonstationary estimations of the expert density, resulting to convergent reward acquisition even for challenging problems. Compared to MD algorithms in optimization studies, our methodology draws a sequence of functions on an alternative space induced by a reward operator Ψ_Ω (Definition 1). To this end, we propose an AIL algorithm called mirror descent adversarial inverse reinforcement learning (MD-AIRL). Our empirical evidence showed that MD-AIRL outperforms the regularized adversarial IRL (RAIRL) [17] methods. For example, MD-AIRL showed higher performance in 30 distinct cases among 32 different configurations in challenging MuJoCo [18] benchmarks, and it clearly showed higher tolerance to suboptimal data. All of these results are strongly aligned with our theoretical results.

Table 1: A technical overview. Traditional IRL methods lack scalability, and RAIRL does not guarantee convergence of its solution for realistic cases. MD-AIRL combines desirable properties.

Method	Reference	Scalability	Rewards	Bregman divergence	Iterative solutions	Rate of convergence
BC (1991)	[19]	✓	✗	✗	✗	✗
MM-IRL (2004)	[5]	✗	✓	✗	✓	✓
GAIL (2016)	[8]	✓	✓	✗	✗	✗
RAIRL (2021)	[17]	✓	✓	✓	✗	✗
MD-AIRL (ours)	–	✓	✓	✓	✓	✓

Our contributions. Our work is complementary to previous IRL studies; the theoretical and technical contributions are built upon a novel perspective of considering iterative RL and IRL algorithms as a combined optimization process with dual aspects. Comparing MD-AIRL with RAIRL, both are highly generalized algorithms in terms of a variety of choices of divergence functions. Tab. 1 shows that MD-AIRL brings beneficial results in realistic situations of limited time and data, since our approach is more aligned with earlier theoretical IRL studies providing formalized reward learning schemes and convergence guarantees. In summary, we list our main contributions below:

- Instead of a monolithic estimation process of a global solution in AIL, we derive a sequence of reward functions that provides iterative local objectives (Section 4).
- We formally prove that rewards derived by an MD update rule guarantee robust performance of divergence minimization along with a rigorous regret bound (Section 5).
- We propose a novel adversarial algorithm that is motivated by mirror descent, which is tolerant to unreliable discriminative signals of the AIL framework (Sections 6 and 7).

2 Related Works

Mirror descent. We are interested in a family of statistical divergence called the Bregman divergence [20]. The divergence generalizes constrained optimization problems such as least squares [21, 22], and it also has been applied in various subfields of machine learning [23, 24]. In differential geometries, the Bregman divergence is a first-order approximation for a metric tensor and satisfies metric-like properties such as the Pythagorean theorem [13]. MD is closely related to optimization methods regarding non-Euclidean geometries with a discretization of steps such as natural gradients [25, 26]. In the primal space, training with the infinitesimal limit of MD steps corresponds to a Riemannian gradient flow [27, 28]. In the RL domain, MD has been recently studied for policy optimization [29–31]. In this paper, we focus on learning with suboptimal representations of policy, and our distinct goal is to draw a robust reward learning scheme based on MD for the IRL problem.

Imitation learning. As a statistical model for the information geometry [32], energy-based policies (i.e., Boltzmann distributions) appeared in early IRL studies, such as Bayesian IRL, natural

gradient IRL, and maximum likelihood IRL [33–35] for modeling expert distribution to parameterized functions. Notably, MaxEnt IRL [6, 36] is one of the representative classical IRL algorithms based on an information-theoretic perspective toward IRL solutions. Also, discriminators of AIL are trained by logistic regression; thus, the logit score of the discriminator defines an energy function that approximates the truth data density for the expert distribution [37]. Other statistical entropies have also been applied to AIL, such as the Tsallis entropy [38]. On the one hand, our approach is closely related to RAIRL [17], which defined its objective as minimizing the Bregman divergence. On the other hand, this work further employs the divergence to derive iterative MD updates for reward functions, resulting to theoretically pleasing properties while retaining the scalability of AIL.

Learning theory. There have been considerable achievements in dealing with temporal costs $\{F_t\}_{t=1}^{\infty}$, often referred to as *online learning* [39]. The most ordinary approach is stochastic gradient descent (SGD): $w_{t+1} = w_t - \eta_t \nabla F_t(w_t)$. In particular, SGD is a desirable algorithm when the parameter w_t resides in the Euclidean space since it ensures unbiased minimization of the expected cost. Apparently, policies appear in geometries of probabilities; thus, an incurred gradient may not be the direction of the steepest descent due to geometric constraints [25, 32]. An online form of MD (Eq. (1)) is analogous to SGD for non-Euclidean spaces, where each local metric is specified by a Bregman divergence [28]. Our theoretical findings and proofs follow the results of online mirror descent (OMD) that appeared in previous literature for general aspects [14, 15, 40, 26, 16, 28]. Our analyses extend existing theoretical results to IRL; at the same time, they are also highly general to cover various online imitation learning problems which require making decision sequentially.

3 Background

For sets X and Y, let Y^X be a set of functions from X to Y and Δ_X (Δ_X^Y) be a set of (conditional) probabilities over X (conditioned on Y). We consider an MDP defined as a tuple $(\mathcal{S}, \mathcal{A}, P, r, \gamma)$ with the state space $\mathcal{S}$, the action space $\mathcal{A}$, the Markovian transition kernel $P \in \Delta_{\mathcal{S}}^{\mathcal{S}\times\mathcal{A}}$, the reward function $r \in \mathbb{R}^{S\times A}$ and the discount factor $\gamma \in [0, 1)$. Let a function $\Omega : \Delta_{\mathcal{A}} \to \mathbb{R}$ be strongly convex. Using Ω, the Bregman divergence is defined as

$$D_\Omega(\pi^s \| \hat{\pi}^s) \coloneqq \Omega(\pi^s) - \Omega(\hat{\pi}^s) - \big\langle \nabla\Omega(\hat{\pi}^s),\ \pi^s - \hat{\pi}^s \big\rangle_{\mathcal{A}},$$

where π^s and $\hat{\pi}$ denote arbitrary policies for a given state s. For a representative divergence, one can consider the popular KL divergence. The KL divergence is a Bregman divergence when Ω is specified as the negative Shannon entropy: $\Omega(\pi^s) = \sum_a \pi^s(a) \ln \pi^s(a)$.

Regularization of policy distribution with respect to convex Ω brings distinct properties for learning agent [41, 42]. The objective of regularized RL is to find $\pi \in \Pi$ that maximizes the expected value of discounted cumulative returns along with a causal convex regularizer $\Omega\colon \Delta_{\mathcal{A}} \to \mathbb{R}$, i.e.

$$J_\Omega(\pi, r) \coloneqq \mathbb{E}_\pi\Big[\sum\nolimits_{i=0}^{\infty} \gamma^i \Big\{ r(s_i, a_i) - \Omega\big(\pi(\,\cdot\,|s_i)\big)\Big\}\Big], \tag{2}$$

where the subscript π on the expectation indicates that each action is sampled by $\pi(\cdot|s_i)$ for the given MDP. In this setup, a trainable policy space is a subset of the entire conditional probability space denoted as $\Pi \coloneqq [\Pi^s]_{s\in\mathcal{S}} \subset \Delta_{\mathcal{A}}^{\mathcal{S}}$, which is constrained by the parameterization of a policy.

The objective of IRL is to find a function r_E that rationalizes the expert behavior π_E. For an inner product $\langle \cdot, \cdot \rangle_{\mathcal{A}}$, consider Ω^*, the Legendre-Fenchel transform (convex conjugate) of Ω:

$$\forall\, q^s \in \mathbb{R}^{\mathcal{A}}, \quad \Omega^*\big(q^s\big) = \max_{\pi^s \in \Delta_{\mathcal{A}}} \big\langle \pi^s,\ q^s \big\rangle_{\mathcal{A}} - \Omega\big(\pi^s\big), \tag{3}$$

where q^s and π^s denote the shorthand notation of $q(s, \cdot\,)$ and $\pi(\cdot|s)$. Differentiating both sides with respect to q^s, the gradient of conjugate $\nabla\Omega^*$ maps q^s to a policy distribution. One fundamental property in *regularized* IRL [41] is that π_E is the maximizing argument of Ω^* for q_E, where q_E is the regularized state-action value function $q_E(s, a) = \mathbb{E}_{\pi_E}[\sum_{i=0}^{\infty} \gamma^i \{r_E(s, a) - \Omega(\pi_E^s)\} | s_0 = s, a_0 = a]$. Note that the problem is ill-posed, and every $\hat{r}_E$ that makes its value function $\hat{q}_E$ satisfy $\pi_E^s = \nabla\Omega^*(\hat{q}_E^s)\ \forall s \in \mathcal{S}$ is a valid solution. Addressing this issue, Jeon et al. [17] proposed a reward operator $\Psi_\Omega : \Delta_{\mathcal{A}}^{\mathcal{S}} \to \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$, providing a unique IRL solution by $\Psi_\Omega(\pi_E)$.

Definition 1 (Regularized reward operators)**.** Define the regularized reward operator Ψ_Ω as $\psi_\pi(s, a) \coloneqq \Omega'(s, a; \pi) - \big\langle \pi^s,\ \nabla\Omega(\pi^s) \big\rangle_{\mathcal{A}} + \Omega(\pi^s)$, for $\Omega'(s, \cdot\,; \pi) \coloneqq \nabla\Omega(\pi^s) = \big[\nabla_p \Omega(p)\big]_{p=\pi(\cdot|s)}$.

The reward function $\psi_E \coloneqq \Psi_\Omega(\pi_E)$ replaces its state-action value function, since the sum of composite Bregman divergences derived from Eq. (2) allows reward learning in a greedy manner [17].

4 RL-IRL as a Proximal Method

Associated reward functions. We consider the RL-IRL processes as a sequential algorithm with local constraints and define sequences $\{\pi_t\}_{t=1}^{\infty}$ and $\{\psi_t\}_{t=1}^{\infty}$ that denote policies and associated reward functions, respectively. The associated reward functions is in a space $\Psi_\Omega(\Pi)$, which is an alternative space of the dual space, defined by the regularized reward operator Ψ_Ω. Formally, we provide Lemma 1 which shows a bijective relation between the operators $\nabla\Omega^*$ and Ψ_Ω in the set Π. The proof is in Appendix A.

Lemma 1 (Natural isomorphism)**.** *Let* $\psi \in \Psi_\Omega(\Pi)$ *for* $\Psi_\Omega(X) := \{\, \psi \mid \psi(s,a) = \psi_\pi(s,a),\ \forall s \in \mathcal{S}, a \in \mathcal{A}, \pi \in X\}$. *Then,* $\nabla\Omega^*(\psi)$ *is unique and for every* $\pi = \nabla\Omega^*(\psi)$, $\pi \in \Pi$.

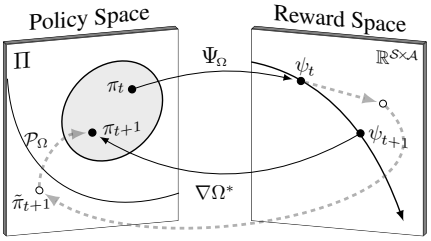

Figure 1: A schematic illustration. MD is locally constrained by a divergence (gray area), i.e., $D_\Omega(\,\cdot\,\|\pi_t)$. An MD update is performed for the reward function ψ_t in an associated reward space of defined by Ψ_Ω, and π_{t+1} is achieved in the desired space of Π by applying $\nabla\Omega^*$ for the function ψ_{t+1}. The gray dashed lines provide another interpretation of MD with $\tilde{\pi}_{t+1}$ and the projection operator $\mathcal{P}_\Omega$.

Fig. 1 illustrates that there is unique ψ_t for π_t in every time step. Note that $\Psi_\Omega(\pi_t)$ is different from $\nabla\Omega(\pi_t)$; it is shifted by a vector $\mathbf{1}c$ with a constant $c = \Omega(\pi_t^s) - \langle \pi_t^s, \nabla\Omega(\pi_t^s)\rangle_{\mathcal{A}}$. Since the underlying space is a probability simplex, the operator $\nabla\Omega^*$ reconstructs the original point for both Ψ_Ω and $\nabla\Omega$, as the distributivity [41] $\Omega^*(y+\mathbf{1}c) = \Omega^*(y)+c$ holds (so $\nabla\Omega^*(y+\mathbf{1}c) = \nabla\Omega^*(y)$). An alternative interpretation is of considering a projection (gray dashed line in Fig. 1). Suppose that a policy π_t is updated to $\tilde{\pi}_{t+1} \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$. The Bregman projection operator $\mathcal{P}_\Omega$ is applied that locates the subsequent update π_{t+1} to the "feasible" region, i.e., $\mathcal{P}_\Omega(\tilde{\pi}_{t+1}) := \operatorname{argmin}_{\pi\in\Pi}[D_\Omega(\pi^s\|\tilde{\pi}_{t+1}^s)]_{s\in\mathcal{S}}$.

Consequently, one can consider an updated reward function ψ_{t+1} as a projected target of MD associated with an alternative parameterization of Π. For instance, the parameters of ψ_t can construct a softmax policy for a discrete space, or a Gaussian policy for a continuous space. Using the reward function ψ_{t+1}, an arbitrary regularized RL process maximizing Eq. (2) at the t-th step [17]

$$J_\Omega(\pi, \psi_{t+1}) = -\mathbb{E}_\pi\Big[\sum\nolimits_{i=0}^{\infty} \gamma^i D_\Omega\Big(\pi(\,\cdot\,|s_i)\Big\|\pi_{t+1}(\,\cdot\,|s_i)\Big)\Big] \tag{4}$$

becomes finding the next iteration $\pi_{t+1} = \nabla\Omega^*(\psi_{t+1})$ by maximizing the RL objective. The equation shows that a regularized RL algorithm with the regularizer Ω forms a cumulative sum of Bregman divergences; thus, the policy π_{t+1} is uniquely achieved by the property of divergence.

Online imitation learning. Our setup comes from an observation that imitation learning do not retain the global target π_E during training. That is, it is fundamentally uncertain to model global objectives (such as $J_\Omega(\pi, \psi_E)$) which are not attainable for both RL and IRL. Instead, we hypothesize on existence of a random process $\{\bar{\pi}_{E,t}\}_{t=1}^{\infty}$ where each estimation $\bar{\pi}_{E,t}$ resides in a closed, convex neighborhood of π_E, generated by an arbitrary estimation algorithm. Substituting ψ_E to $\psi_{\bar{\pi}_{E,t}} = \Psi_\Omega(\bar{\pi}_{E,t})$, $J_\Omega(\pi, \psi_{\bar{\pi}_{E,t}})$ forms a temporal costs F_t:

$$F_t(\pi) = \mathbb{E}_\pi\Big[\sum\nolimits_{i=0}^{\infty} \gamma^i D_\Omega\big(\pi(\,\cdot\,|s_i)\big\|\,\bar{\pi}_{E,t}(\,\cdot\,|s_i)\big)\Big]. \tag{5}$$

For a better understanding, we consider an actual experiment in Fig. 2. Suppose that policies of the learning agent and the expert follow multivariate Gaussian distributions at $\mathcal{N}([0,0]^\intercal, \mathrm{I})$ and

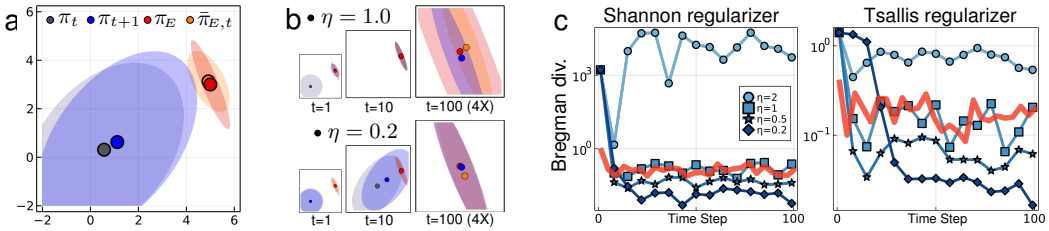

Figure 2: (**a**) A policy π_t learns from MD updates for temporal costs $D_\Omega(\,\cdot\,\|\bar{\pi}_{E,t})$. (**b**) The updates of π_t vary by η, and the distance between π_t and π_E can be closer than the distance between $\bar{\pi}_{E,t}$ and π_E when t is sufficiently large and the η is effectively low. (**c**) Two plots show $D_\Omega(\pi_t, \|\pi_E)$ associated with entropic regularizers for four different η (10 trials), with the red baselines $D_\Omega(\bar{\pi}_{E,t}\|\pi_E)$.

$\mathcal{N}([5,3]^\intercal, \Sigma_E)$ with $|\Sigma_E| < 1$. Suppose a (suboptimal) reference policy $\bar{\pi}_{E,t}$, independently fitted with a maximum likelihood estimator with a relatively high learning rate, starting from $\bar{\pi}_{E,1} = \pi_1$. The policy π_t was trained by a cost function $D_\Omega(\cdot \| \bar{\pi}_{E,t})$ using the MD update rule in Eq. (1). In Fig. 2, we first observed that choosing high step size constant η accelerates the training speed mostly in the early phase. The results also showed that the performance of MD ($D_\Omega(\pi_t \| \pi_E)$) outperformed that of referenced maximum likelihood estimation ($D_\Omega(\bar{\pi}_{E,t} \| \pi_E)$) by choosing an effectively low step size. This suggests that there are clear advantages in scheduling step sizes by time steps especially for unreliable $\bar{\pi}_{E,t}$.

MD update rules. As a result, we formulate subsequent MD steps for a policy dstistribution. Let w_t be a parameter on a set $\mathcal{W}$ and $F_t : \mathcal{W} \to \mathbb{R}$ be a convex cost function from a class of functions $\mathcal{F}$ at the t-th step. Replacing the L2 proximity term of proximal gradient descent with a Bregman divergence, the proximal form of the MD update for Eq. (1) is written as [43]

$$\underset{w \in \mathcal{W}}{\text{minimize}} \left\langle \nabla F_t(w_t),\, w - w_t \right\rangle_{\mathcal{W}} + \alpha_t D_\Omega\big(w \,\big\|\, w_t\big), \tag{6}$$

where $\alpha_t := 1/\eta_t$ denotes an inverse of the current step size η_t [44]. Plugging each divergences of the cumulative cost F_t to Eq. (6), the MD update for the subsequent reward function $\psi_{t+1} = \Psi_\Omega(\pi_{t+1})$ is derived by solving a problem

$$\begin{aligned}
&\underset{\pi^s \in \Pi^s}{\text{minimize}} \Big\langle \underbrace{\nabla D_\Omega\big(\pi_t^s \,\big\|\, \bar{\pi}_{E,t}^s\big)}_{\nabla\Omega(\pi_t^s) - \nabla\Omega(\bar{\pi}_{E,t}^s)},\, \pi^s - \pi_t^s \Big\rangle_{\mathcal{A}} + \alpha_t D_\Omega\big(\pi^s \,\big\|\, \pi_t^s\big) \\
\iff \quad & \underset{\pi^s \in \Pi^s}{\text{minimize}}\, D_\Omega\big(\pi^s \big\| \bar{\pi}_{E,t}^s\big) - D_\Omega\big(\pi^s \big\| \pi_t^s\big) + \alpha_t D_\Omega\big(\pi^s \big\| \pi_t^s\big) \\
\iff \quad & \underset{\pi^s \in \Pi^s}{\text{minimize}}\, \eta_t \underbrace{D_\Omega\big(\pi^s \big\| \bar{\pi}_{E,t}^s\big)}_{\text{estimated expert}} + (1 - \eta_t) \underbrace{D_\Omega\big(\pi^s \big\| \pi_t^s\big)}_{\text{learning agent}} \qquad \forall s \in \mathcal{S},
\end{aligned} \tag{7}$$

where the gradient of D_Ω is taken with respect to its first argument π_t^s. Note that solving the optimization Eq. (7) requires interaction between π_t and the dynamics of the given environment in order to minimize F_t; thus, the corresponding RL process plays an essential role in sequential learning by the induced the value measures. At a glance the objective is analogous to finding an interpolation at each iteration where the point is controlled η_t. Fig. 3 shows that the uncertainty of π_t (blue region) gets minimal regardless of persisting uncertainty of $\bar{\pi}_{E,t}$ (red region).

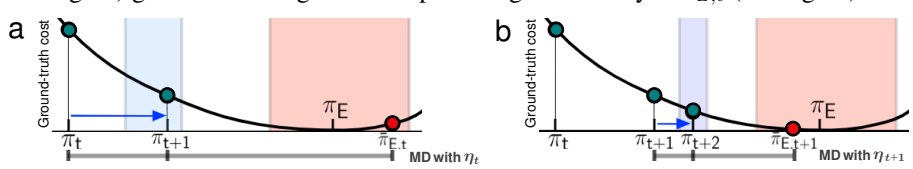

Figure 3: Illustrations of MD at the (**a**) t-th iteration and (**b**) $(t+1)$-th iteration where $\eta_t > \eta_{t+1}$. $\{\bar{\pi}_{E,t}\}_{t=1}^\infty$ is an arbitrary estimation process attained from a neighborhood of π_E with respect to a norm. The MD update is taken inside the interval of π_t and $\bar{\pi}_{E,t}$ using Eq. (7).

5 Convergence Analyses

Suppose that states at the t-th step $s_i^{(t)} \in \tau_t$ covers $\mathcal{S}$ by executing the policy π_t in an infinite horizon. From this assumption, we define a temporal cost function at the time step t as

$$f(\pi_t, \tau_t) := \textstyle\sum_{i=0}^\infty \gamma^i D_\Omega\big(\pi_t(\,\cdot \mid s_i^{(t)}) \,\big\|\, \bar{\pi}_{E,t}(\,\cdot \mid s_i^{(t)})\big), \tag{8}$$

that additionally involves a trajectory τ_t as an input. We refer to the global objective as finding a unique fixed point $\pi_* \in \Pi$ that minimizes a total cost $F(\pi) := \mathbb{E}[f(\pi, \tau_t)]$, where the expectation is taken over trajectories of entire steps, i.e., $\lim_{t\to\infty} \mathbb{E}_{\tau_{1:t}}[f(\pi, \tau_t)]$. Taking the (stepwise) gradient for each $\pi(\cdot|s)$, an optimal policy π_* is found by $\mathbb{E}[\nabla\Omega(\pi_*(\cdot|s)) - \nabla\Omega(\bar{\pi}_{E,t}(\cdot|s))] = 0$ when $t \to \infty$; hence, $\nabla\Omega(\pi_*(\cdot|s)) = \lim_{t\to\infty} \mathbb{E}[\nabla\Omega(\bar{\pi}_{E,t}(\cdot|s))]$. Introducing the optimal policy π_* allows not only the specific situation when **(1)** $\pi_E = \pi_* \in \Pi$ and the estimation algorithm of $\bar{\pi}_{E,t}$ is actually convergent with $t \to \infty$, but also more general situations where **(2)** $\pi_E \notin \Pi$ or the estimated expert policy $\bar{\pi}_{E,t}$ does not converge; the algorithm finds convergence to a fixed point by scheduling updates.

We state two conditions of $\{\eta_t\}_{t=1}^\infty$ to guarantee convergence justified in Theorems 1 and 2.

- Convergent sequence & divergent series:

$$\lim_{t\to\infty} \eta_t = 0 \qquad \text{and} \qquad \textstyle\sum_{t=1}^\infty \eta_t = \infty. \tag{9}$$

- Divergent series & convergent series of squared terms:

$$\sum_{t=1}^{\infty} \eta_t = \infty \qquad \text{and} \qquad \sum_{t=1}^{\infty} \eta_t^2 < \infty. \tag{10}$$

We assume Lipschitz continuity of $\nabla\Omega$ and boundedness of D_Ω in a Banach space. In some Ω, these two assumptions do not necessarily hold for extreme cases in $\Delta_{\mathcal{A}}^{\mathcal{S}}$, e.g., a distribution that $\pi(a|s) = 0$ for some entries. Nevertheless, these outliers can be leaved out if the parametrization is constrained to satisfy the assumptions. For example, one can either **(1)** prevent a policy to have non-zero entries of probabilities for a discrete policy by a constraint on its parameters, or **(2)** prevent having too low value of entropy for a continuous policy, typically by a constraint on the covariance functions.

Theorem 1 argues that the sequence $\{\eta_t\}_{t=1}^{\infty}$ shall diverge for its series; therefore, Eq. (9) is satisfied.

Theorem 1 (Stepsize considerations)**.** *Let Ω be strongly convex, $\nabla\Omega$ be Lipschitz continuous, and the associated Bregman divergence D_Ω is bounded. Assume a general condition of the problem that $\inf_{\pi\in\Pi}\mathbb{E}\big[f(\pi,\tau_t)\big]>0$. Then we get $\lim_{T\to\infty}\mathbb{E}_{\tau_{1:T}}\big[\sum_{i=0}^{\infty} D_\Omega\big(\pi_*(\cdot|s_i)\big\|\pi_T(\cdot|s_i)\big)\big] = 0$ if and only if Eq. (9) is satisfied.*

(a) If $\lim_{t\to\infty}\eta_t = 0$, then $T\in\mathbb{N}$, $n<T$, and $c>0$ exist such that $\mathbb{E}_{\tau_{1:T}}\big[f_T(\pi_T,\tau_T)\big] \geq \frac{c}{T-n}$.

(b) If the step size takes the form of $\eta_t = \frac{4}{t+1}$, then $\mathbb{E}_{\tau_{1:T}}\big[\sum_{i=0}^{\infty} D_\Omega\big(\pi_*(\cdot|s_i)\big\|\pi_T(\cdot|s_i)\big)\big] = \mathcal{O}(1/T)$.

Next, we provide Theorem 2, which addresses the convergence in a specific case when π_E resides in Π. Additionally, the theorem addresses the bounds of the performance for fixed size update $\eta_t \equiv \eta_1$.

Theorem 2 (Optimal cases)**.** *Let Ω be strongly convex, $\nabla\Omega$ be Lipschitz continuous, and the associated Bregman divergences be bounded. Assume $\pi_1 \neq \pi_E$ and $\inf_{\pi\in\Pi}\mathbb{E}[f(\pi,\tau_t)] = 0$. Then, $\mathbb{E}\big[f(\pi_t,\tau_t)\big] = 0$ if and only if $\sum_{t=1}^{\infty}\eta_t = \infty$. If $\eta_t \equiv \eta_1$, then there exist $c_1, c_2 \in (0,1)$ such that $c_1^{T-1}\cdot A_1 \leq A_T \leq c_2^{T-1}\cdot A_1$, for $A_t = \mathbb{E}_{\tau_{1:t}}\big[D_\Omega(\pi_E\|\pi_t)\big]$.*

Lastly, Proposition 1 provides the sufficient condition for the almost certain convergence of the algorithm by imposing the stronger condition of step size in Eq. (10). The proofs are in the appendix.

Proposition 1 (General cases)**.** *Assume that $\pi_E \notin \Pi$, hence $\inf_{\pi\in\Pi}\mathbb{E}[f(\pi,\tau_t)] > 0$. If the step sizes satisfies Eq. (10), then $\lim_{t\to\infty}\sum_{i=0}^{\infty}\gamma^i D_\Omega\big(\pi_*(\cdot|s_i)\big\|\pi_t(\cdot|s_i)\big)$ converges to 0 almost surely.*

Regrets. For a sequence of state trajectories $\{\tau_t\}_{t\in\mathbb{N}}$, let us define a regret at the t-th iteration as

$$\tfrac{1}{t}\textstyle\sum_{i=1}^{t} f(\pi_i,\tau_i) - \inf_{\pi\in\Pi}\big\{\tfrac{1}{t}\sum_{j=1}^{t} f(\pi,\tau_j)\big\}. \tag{11}$$

In the optimal case of $\inf_{\pi\in\Pi}\mathbb{E}[f(\pi,\tau_t)] = 0$, the cost f inherits the property of Bregman divergence so that the infimum is achieved by 0 at π_E. In this case, the regret is bounded to $\mathcal{O}(1/T)$ by the theorems. By Proposition 1, the MD updates converge for the case of $\inf_{\pi\in\Pi}\mathbb{E}[f(\pi,\tau_t)] > 0$ when the step sizes abide by Eq. (10). Thus, the regret is bounded to $\mathcal{O}(1/T)$ even for the general case.

6 Algorithm: MD-IRL on an Adversarial Framework

In this section, we propose MD-AIRL, a novel AIL algorithm which trains a parameterized reward function with adversarial learning and the MD update rule. Here, the parameters θ, ϕ, and ν are newly presented representing agent policy, reward, and expert policy functions respectively.

Dual discriminators. Motivated by studies regarding multiple discriminators [45, 46], we propose a dual discriminator architecture to clearly separate two concepts in AIL: matching overall state densities and imitating specific behavior. Given a learning agent policy π_θ, an estimation policy π_ν, and a discriminative neural network for states $d_\xi : \mathcal{S} \to \mathbb{R}$, two discriminators

$$D_\nu(s,a;\,\theta,\xi) = \sigma\big(\log\big\{\pi_\nu(a|s)\big/\pi_\theta(a|s)\big\} + d_\xi(s)\big) \quad \text{and} \quad D_\xi(s) = \sigma\big(d_\xi(s)\big), \quad \forall\, s\in\mathcal{S},\ a\in\mathcal{A},$$

where $\sigma(\cdot)$ denotes the sigmoid function. The discriminators are trained using binary logistic regression losses with respect to mini-batch adversarial samples:

$$\underset{d_\xi}{\text{maximize}}\, \mathcal{J}_{d_\xi} = \mathbb{E}_{\pi_E}\big[\log D_\xi(s)\big] + \mathbb{E}_{\pi_\phi}\big[\log(1 - D_\xi(s))\big], \tag{12}$$

$$\underset{\pi_\nu}{\text{maximize}}\, \mathcal{J}_{\pi_\nu} = \mathbb{E}_{\pi_E}\big[\log D_\nu(s,a)\big] + \mathbb{E}_{\pi_\phi}\big[\log(1 - D_\nu(s,a))\big], \tag{13}$$

Algorithm 1 Mirror Descent Adversarial Inverse Reinforcement Learning.

Input: trajectories $\{\tau_t^*\}_{t=1}^T$, an agent π_θ, a reference policy π_ν, a neural network $d_\xi : \mathcal{S} \to \mathbb{R}$, a regularized reward function $\psi_\phi \in \Psi_\Omega(\Pi)$, α_1, α_T, and λ.
for $t \leftarrow 1$ to T **do**
  $\alpha_t \leftarrow \alpha_1 + \frac{\alpha_T - \alpha_1}{T-1}(t-1)$ and then $\eta_t \leftarrow 1/\alpha_t$.
  Optimize d_ξ and π_ν via binary logistic regression for D_ξ and D_ν.
  Optimize ψ_ϕ with the objective in Eq. (14) using both τ_t^* and τ_t.
  Train π_θ via RL to maximize $\psi_\phi^\lambda(s,a)$ with regularizer $\lambda\Omega(\cdot)$.
Output: π_θ, ψ_ϕ^λ.

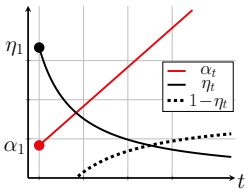

Figure 4: A sequence example of (α_t, η_t).

where d_ξ and π_θ are not trained for learning D_ν. Let $\rho_\pi \in \Delta_\mathcal{S}$ denote the state visitation density of π, which is defined as $\rho_\pi(s) := (1-\gamma)\,\mathbb{E}_\pi\big[\sum_{i=0}^\infty \gamma^i\, \mathbb{I}\{s_i = s\}\big]$ where $\mathbb{I}\{\cdot\}$ is an indicator function. Since the function $x \mapsto \alpha \log \sigma(x) + \beta \log(1-\sigma(x))$ attains its maximum at $\sigma(x) = \frac{\alpha}{\alpha+\beta}$ [10], the convergence in an optimal case is found at $\pi_\nu = \pi_E$ and $d_\xi(s) = \log\{\rho_{\pi_E}(s)/\rho_\pi(s)\}$.

Learning with MD-based rewards. Let $\psi_\phi \in \Psi(\Pi)$ denote a parameterized regularized reward function and π_ϕ denotes a corresponding policy from ϕ. Using a step size η_t, the RL agent π_θ, and the estimated expert policy π_ν, the objective of ϕ is defined according to the update rule in Eq. (7):

$$\underset{\psi_\phi}{\text{minimize}}\, \mathcal{L}_{\psi_\phi} = \mathbb{E}_{s\sim\bar{\tau}_t}\big[\eta_t\, D_\Omega\big(\pi_\phi(\,\cdot\,|\,s)\big\|\pi_\nu(\,\cdot\,|\,s)\big) + (1-\eta_t) D_\Omega\big(\pi_\phi(\,\cdot\,|\,s)\big\|\pi_\theta(\,\cdot\,|\,s)\big)\big], \tag{14}$$

The trajectory $\bar{\tau}_t$ denotes sample states using both agent and expert trajectories. The step size is adjusted with by linearly increasing α_t throughout traning, originated from the analyses in Section 5.

In order to mitigate covariance shifts, we define the MD-AIRL reward function as linear combinations of the regularized reward function and the state density discriminative signal:

$$\psi_\phi^\lambda(s,a) = \lambda\psi_\phi(s,a) + d_\xi(s), \tag{15}$$

for a coefficient $\lambda \in \mathbb{R}^+$. Utilizing an arbitrary regularized RL algorithm which uses $\lambda\Omega(\cdot)$ as its regularizer, the reward learning regarding agent policy π_θ is decomposed into the following:

$$\begin{aligned}\mathbb{E}_{\pi_\theta}\Big[\psi_\phi^\lambda(s,a) - \lambda\Omega\big(\pi_\theta(\,\cdot\,|\,s)\big)\Big] &= \lambda\mathbb{E}_{\pi_\theta}\Big[\psi_\phi(s,a) - \Omega\big(\pi_\theta(\,\cdot\,|\,s)\big)\Big] - D_{\mathrm{KL}}\big(\rho_{\pi_\theta}\big\|\,\rho_{\pi_E}\big)\\ &= -\lambda\mathbb{E}_{\pi_\theta}\Big[D_\Omega\big(\pi_\theta(\,\cdot\,|s)\big\|\pi_\phi(\,\cdot\,|s)\big)\Big] - D_{\mathrm{KL}}\big(\rho_{\pi_\theta}\big\|\,\rho_{\pi_E}\big).\end{aligned}$$

Minimizing the first term of $\mathbb{E}_{\pi_\theta}\big[D_\Omega(\pi_\theta^s \| \pi_\phi^s)\big]$ represents learning with our MD formulation. However, ψ_ϕ that is trained on limited samples from $\bar{\tau}_t$ might not cover the entire reachable states because of misaligned densities in challenging problems. Therefore, the second term $D_{\mathrm{KL}}(\rho_{\pi_\theta} \| \rho_{\pi_E})$ plays an auxiliary role in facilitating the supports of state visitation densities to be matched properly. Algorithm 1 summarizes the entire procedure. We defer additional details to the appendix.

7 Experimental Results

The aim of our experiments was to identify whether MD-AIRL facilitates robustness while retaining scalability of AIL. The comparative method was RAIRL with density based models (RAIRL-DBM) which contained comparable expressiveness as MD-AIRL For RL, we used RAC algorithm [42] that is a generalization of the SAC algorithm [47]. We considered a class of regularizers $\Omega(p) = -\mathbb{E}_{x\sim p}[\varphi(p(x))]$ with **(1)** Shannon $(\varphi(x) = \log(x))$, **(2)** Tsallis $(\varphi(x;q) = \frac{1}{q-1}(x^{q-1}-1)$, $q = 2$ by default), **(3)** exp $(\varphi(x) = e - e^x)$, **(4)** cos $(\varphi(x) = \cos(\frac{\pi}{2}x))$, and **(5)** sin $(\varphi(x) = 1 - \sin(\frac{\pi}{2}x))$.

7.1 Large scale multiarmed bandits

We first considered multiarmed bandit problems, where the cardinality of action spaces varies largely. Learning the optimal distribution of π_E becomes challenging as the size of action space $|\mathcal{A}|$ increases, because the frequency of each sample becomes sparse due to the curse of dimensionality. The stateless expert distribution π_E was generated by the parameters of softmax distribution $\pi_E(a) = {}^{\exp(z_a)}/_{\sum_i \exp(z_i)}$ where the logits z_i were randomly initialized. We set the action size to $|\mathcal{A}| = 10^2, 10^3, 10^4$ and restricted each sample size of adversarial learning to 16.

Table 2: The training results of $|\mathcal{A}| \cdot D_\Omega(\pi_T \| \pi_E)$ with five types of regularization (five runs with different seeds).

	$\lvert\mathcal{A}\rvert = 10^2$		$\lvert\mathcal{A}\rvert = 10^3$		$\lvert\mathcal{A}\rvert = 10^4$	
Method	RAIRL	MD-AIRL	RAIRL	MD-AIRL	RAIRL	MD-AIRL
Shannon	2.55 ± 1.59	$\mathbf{2.28 \pm 1.20}$	140.3 ± 87.5	$\mathbf{125.3 \pm 61}$	-	-
Tsallis	0.21 ± 0.13	$\mathbf{0.11 \pm 0.04}$	0.55 ± 0.13	$\mathbf{0.24 \pm 0.03}$	4.95 ± 2.3	$\mathbf{4.21 \pm 0.2}$
exp	0.27 ± 0.17	$\mathbf{0.13 \pm 0.06}$	0.55 ± 0.12	$\mathbf{0.23 \pm 0.03}$	5.06 ± 2.4	$\mathbf{4.97 \pm 0.7}$
cos	0.05 ± 0.04	$\mathbf{0.02 \pm 0.01}$	0.03 ± 0.02	$\mathbf{0.01 \pm 0.01}$	0.21 ± 0.6	$\mathbf{0.05 \pm 0.1}$
sin	0.34 ± 0.25	$\mathbf{0.12 \pm 0.04}$	3.82 ± 3.46	$\mathbf{1.07 \pm 0.75}$	8.12 ± 3.8	$\mathbf{7.59 \pm 1.0}$

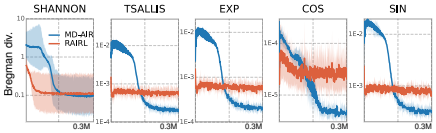

Figure 5: The cost $D_\Omega(\pi_T \| \pi_E)$ on the log-scale at $|\mathcal{A}| = 10^3$. The shade represents 95% confidence interval.

Tab. 2 shows that MD-AIRL achieved overall lower Bregman divergence on average when three different cardinalities and five regularizers were considered. Fig. 5 shows that the Bregman divergence was large for MD-AIRL at the early training phase, because we chose the initial step size η_1 to be greater than 1 ($\alpha_1 = 0.5$). MD-AIRL exceeded the discriminative performance of RAIRL after certain steps, while the progression of RAIRL mostly stopped at local minima. MD-AIRL outperformed RAIRL in four cases by an choosing effectively low step size at η_T to be less than 1 ($\alpha_T = 2$). These results match the properties of MD algorithms and our convergence analyses. Therefore, we argue that a constrained update rule with appropriate step sizes is necessary for robust reward acquisition and imitation for situations when the total number of data samples is limited.

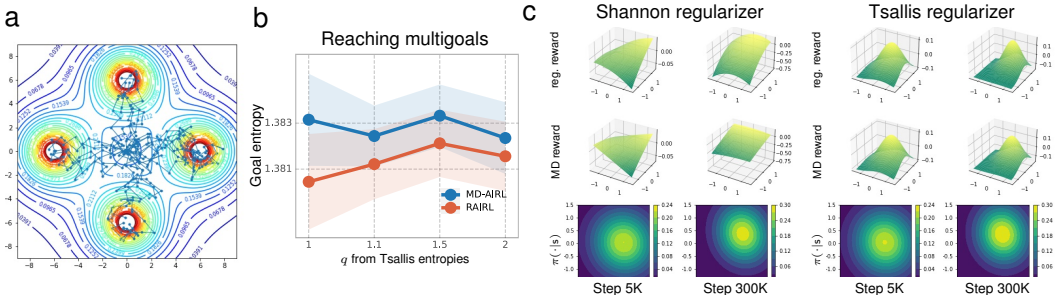

Figure 6: (**a**) Visualisation of trajectories trained by MD-AIRL, and the ground-truth reward surface. (**b**) The entropies for the probabilities of achieving four goals. The x-axis indicates the q value from the Tsallis regularizers (the Shannon regularizer is considered by $q = 1$ [48]). (**c**) The top and middle of each column show induced reward surfaces. The bottom shows the agent policy.

7.2 A continuous multigoal environment

We then considered a multigoal environment. In this environment, an agent is a two-dimensional point mass initialized at the origin, and the four goals are located in the four cardinal directions. The objective of imitation learning is to go toward each direction evenly as possible where the expert model was trained by an SAC algorithm. To draw informative reward surfaces regarding stochastic actions, we considered the multivariate Gaussian distribution policies parameterized with full covariance matrices instead of conventional diagonal Gaussian policies (see Appendices 2 and 3).

Fig. 6 (a) shows trajectories generated by the trained agent. Fig. 6 (b) shows that MD-AIRL achieved higher entropy for reaching the multiple goals. Fig. 6 (c) shows reward surfaces with regularizers, which were calculated by $\psi_\phi(s, a) + \varphi(\pi_\theta(a|s))$ for each point of $a \in \mathcal{A}$ and $s = (5, -1)$. During the training, the MD reward was similar to the estimated ground truth using adversarial training. However, the surface of MD-AIRL became flatter than the ground-truth estimation when π_t was sufficiently close to the expert behavior. As a result, we claim that a drastic change in the target distribution, which is one of the typical characteristics of adversarial frameworks, are prevented. We argue that these characteristics mitigate overfitting caused by unreliable discriminative signals.

7.3 A continuous control benchmark: MuJoCo

Lastly, we validated MD-AIRL on the MuJoCo continuous control suite. We assumed full covariance Gaussian policies for both learner's policy π and expert policy π_E. We used the hyperbolized environment assumption [17] where the action constraint is incorporated to the dynamics as a part of the environment using hyperbolic tangent activation.

Sample efficiency. For each task, we considered two different numbers of episodes collected by an expert policy. In Fig. 7, the performance of MD-AIRL, RAIRL, and behavior cloning (bc) algorithms [19] is shown with the expert and random agent performance. MD-AIRL was able to achieve

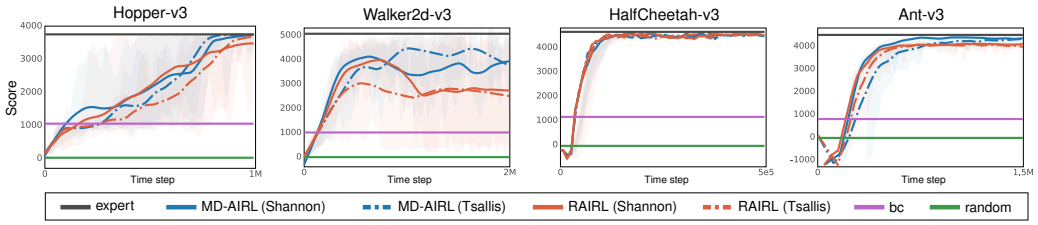

Figure 7: Average scores for 5 runs with two different regularizers (Shannon and Tsallis regularizer). The agent and IRL reward functions were trained with 4 episodes of expert demonstrations.

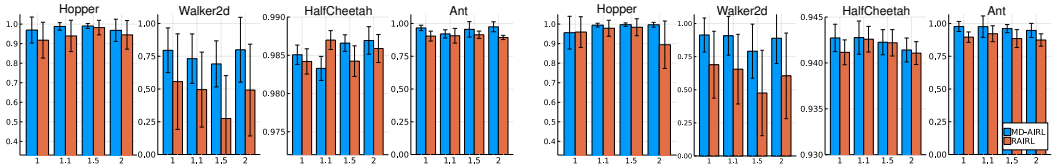

Figure 8: Scores on the last 10^5 steps in a total of 32 different settings. The x-axis indicates the q value of the Tsallis regularizers. The scores are rescaled by considering the expert performance as 1, and the error bars represent standard deviations. Left: 4 demonstrations. Right: 100 demonstrations.

consistent performance throughout the tasks and demonstration size. On the training curves, MD-AIRL showed high tolerance to the scarcity of data compared to RAIRL for 4 expert demonstrations. The plots in Fig. 8 indicate that MD-AIRL showed higher average scores with lower variance compared to RAIRL, across 30 distinct cases among 32 different configurations we have tested. MD-AIRL inherits the scalability of AIL, and it is highly stable with respect to limited sample sizes.

Table 3: Scores on noisy demonstrations. The values of ε represents scales of the Gaussian noises.

	Method	$\varepsilon = 0.01$	$\varepsilon = 0.5$
Hopper	RAIRL (Shannon)	3636.03 ± 391.09	3573.74 ± 508.14
	MD-AIRL (Shannon)	$\mathbf{3669.25 \pm 177.78}$	$\mathbf{3653.31 \pm 267.87}$
	RAIRL (Tsallis)	3671.12 ± 322.32	3576.17 ± 515.75
	MD-AIRL (Tsallis)	$\mathbf{3730.14 \pm 63.09}$	$\mathbf{3701.24 \pm 205.68}$
Walker2d	RAIRL (Shannon)	2856.56 ± 939.9	2451.00 ± 1392.6
	MD-AIRL (Shannon)	$\mathbf{3386.38 \pm 953.59}$	$\mathbf{3252.65 \pm 1395.7}$
	RAIRL (Tsallis)	2731.84 ± 1058.7	2435.10 ± 1555.2
	MD-AIRL (Tsallis)	$\mathbf{3624.00 \pm 992.63}$	$\mathbf{3093.54 \pm 963.96}$

	Method	$\varepsilon = 0.01$	$\varepsilon = 0.5$
HalfCheetah	RAIRL (Shannon)	4354.15 ± 63.83	4216.99 ± 661.17
	MD-AIRL (Shannon)	$\mathbf{4373.17 \pm 68.12}$	$\mathbf{4337.18 \pm 106.40}$
	RAIRL (Tsallis)	4364.13 ± 68.09	4216.67 ± 248.08
	MD-AIRL (Tsallis)	$\mathbf{4388.87 \pm 73.19}$	$\mathbf{4247.44 \pm 266.73}$
Ant	RAIRL (Shannon)	4493.74 ± 383.04	3777.78 ± 505.78
	MD-AIRL (Shannon)	$\mathbf{4658.29 \pm 201.37}$	$\mathbf{4284.38 \pm 329.79}$
	RAIRL (Tsallis)	4359.62 ± 168.46	3660.22 ± 508.54
	MD-AIRL (Tsallis)	$\mathbf{4705.25 \pm 130.53}$	$\mathbf{4127.37 \pm 457.25}$

Noisy demonstrations. Tab. 3 shows the results of imitation learning experiments for 100 expert demonstrations with two different levels of Gaussian additive noises, resulting to suboptimal demonstrations. MD-AIRL is highly tolerant to noisy data, consistently achieving higher performance. The experiment is closely related to the general case in the theory the results suggest that the characteristics of MD-AIRL are in alignment with our analyses of the MD reward learning scheme.

8 Conclusions and Discussion

In this paper, we presented MD-AIRL, a practical AIL algorithm designed to solve the imitation learning problem in the real world. We proved that the proposed method has clear advantages over previous AIL methods in terms of robustness. We verified MD-AIRL on a variety of situations, including high-dimensional spaces, limited samples, and imperfect demonstrations. The empirical evidence showed that MD-AIRL outperforms previous methods on various benchmarks. We conclude that the rich foundation of optimization theories shows a promising direction to AIL studies.

Considering RL and its inverse problem IRL with geometric perspectives is vital for solving real-world problems. Although our work covers various imitation learning problems with the Bregman divergence, this does not include some other problems when the proximity term is of other statistical divergence families such as the f-divergence. The assumptions on Ω in our analyses are usually justified by enforcing a specific policy space, but some outliers might have substantial meaning for certain control tasks. Therefore, extensive analyses on parameterizations when applying MD remain as future works. This work can be applied to multiple domains, including autonomous driving and robotics. The proposed method is expected to ease the unsafe behavior of RL in real-world problems. On the other side, this work has the potential of automation of labor, particularly routine manual jobs.

References

[1] Ziyu Wang, Josh S Merel, Scott E Reed, Nando de Freitas, Gregory Wayne, and Nicolas Heess. Robust imitation of diverse behaviors. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, 2017.

[2] Yuke Zhu, Ziyu Wang, Josh Merel, Andrei Rusu, Tom Erez, Serkan Cabi, Saran Tunyasuvunakool, JÃ¡nos KramÃ¡r, Raia Hadsell, Nando de Freitas, and Nicolas Heess. Reinforcement and imitation learning for diverse visuomotor skills. In *Proceedings of Robotics: Science and Systems*, 2018.

[3] Beomjoon Kim, Amir-massoud Farahmand, Joelle Pineau, and Doina Precup. Learning from limited demonstrations. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

[4] Andrew Y Ng and Stuart J Russell. Algorithms for inverse reinforcement learning. In *Proceedings of the 17th International Conference on Machine Learning*, pages 663–670. Morgan Kaufmann Publishers Inc., 2000.

[5] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the 21st International Conference on Machine Learning*, page 1. ACM, 2004.

[6] Brian D Ziebart, Andrew Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *Proceedings of 23rd AAAI Conference on Artificial Intelligence*, volume 3, pages 1433–1438, 2008.

[7] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

[8] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, pages 4565–4573, 2016.

[9] Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv preprint arXiv:1710.11248*, 2017.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[11] Ilya Kostrikov, Kumar Krishna Agrawal, Debidatta Dwibedi, Sergey Levine, and Jonathan Tompson. Discriminator-actor-critic: Addressing sample inefficiency and reward bias in adversarial imitation learning. In *7th International Conference on Learning Representations*, 2019.

[12] Seyed Kamyar Seyed Ghasemipour, Richard Zemel, and Shixiang Gu. A divergence minimization perspective on imitation learning methods. In *Conference on Robot Learning*, pages 1259–1277. PMLR, 2020.

[13] Dan Butnariu and Elena Resmerita. Bregman distances, totally convex functions, and a method for solving operator equations in banach spaces. In *Abstract and Applied Analysis*, volume 2006, 2006.

[14] Arkadiĭ Semenovich Nemirovsky and David Borisovich Yudin. *Problem Complexity and Method Efficiency in Optimization*. A Wiley-Interscience publication. Wiley, 1983. ISBN 9780471103455.

[15] Nati Srebro, Karthik Sridharan, and Ambuj Tewari. On the universality of online mirror descent. In *Advances in Neural Information Processing Systems*, pages 2645–2653, 2011.

[16] Yunwen Lei and Ding-Xuan Zhou. Convergence of online mirror descent. *Applied and Computational Harmonic Analysis*, 48(1):343 – 373, 2020. ISSN 1063-5203.

[17] Wonseok Jeon, Chen-Yang Su, Paul Barde, Thang Doan, Derek Nowrouzezahrai, and Joelle Pineau. Regularized inverse reinforcement learning. In *9th International Conference on Learning Representations*, 2021.

[18] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012.

[19] Dean A Pomerleau. Efficient training of artificial neural networks for autonomous navigation. *Neural computation*, 3(1):88–97, 1991.

[20] Lev M Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR computational mathematics and mathematical physics*, 7(3):200–217, 1967.

[21] Stephen P. Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2014. ISBN 978-0-521-83378-3.

[22] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Fundamentals of convex analysis*. Springer Science & Business Media, 2004.

[23] Arindam Banerjee, Srujana Merugu, Inderjit S. Dhillon, and Joydeep Ghosh. Clustering with bregman divergences. *Journal of Machine Learning Research*, 6(58):1705–1749, 2005.

[24] Akash Srivastava, Kristjan H. Greenewald, and Farzaneh Mirzazadeh. Bregmn: scaled-bregman generative modeling networks. *CoRR*, abs/1906.00313, 2019.

[25] Shun-ichi Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

[26] G. Raskutti and S. Mukherjee. The information geometry of mirror descent. *IEEE Transactions on Information Theory*, 61(3):1451–1457, 2015.

[27] Manfredo P Do Carmo. *Differential geometry of curves and surfaces: revised and updated second edition*. Courier Dover Publications, 2016.

[28] Suriya Gunasekar, Blake E. Woodworth, and Nathan Srebro. Mirrorless mirror descent: A more natural discretization of riemannian gradient flow. *CoRR*, abs/2004.01025, 2020.

[29] Guanghui Lan. Policy mirror descent for reinforcement learning: Linear convergence, new sampling complexity, and generalized problem classes. *CoRR*, abs/2102.00135, 2021.

[30] Wenhao Zhan, Shicong Cen, Baihe Huang, Yuxin Chen, Jason D. Lee, and Yuejie Chi. Policy mirror descent for regularized reinforcement learning: A generalized framework with linear convergence. *CoRR*, abs/2105.11066, 2021.

[31] Manan Tomar, Lior Shani, Yonathan Efroni, and Mohammad Ghavamzadeh. Mirror descent policy optimization. *arXiv preprint arXiv:2005.09814*, 2020.

[32] Shun-ichi Amari. *Information geometry and its applications*, volume 194. Springer, 2016.

[33] Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning. In *Proceedings of International Joint Conference on Artificial Intelligence*, pages 2586–2591, 2007.

[34] Gergely Neu and Csaba Szepesvári. Apprenticeship learning using inverse reinforcement learning and gradient methods. In *Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence*, pages 295–302, 2007.

[35] Monica Babeş-Vroman, Vukosi Marivate, Kaushik Subramanian, and Michael Littman. Apprenticeship learning about multiple intentions. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, page 897–904, 2011. ISBN 9781450306195.

[36] Brian D Ziebart, J Andrew Bagnell, and Anind K Dey. Modeling interaction via the principle of maximum causal entropy. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 1255–1262, 2010.

[37] Junbo Jake Zhao, Michaël Mathieu, and Yann LeCun. Energy-based generative adversarial networks. In *5th International Conference on Learning Representations*, 2017.

[38] Kyungjae Lee, Sungjoon Choi, and Songhwai Oh. Maximum causal tsallis entropy imitation learning. In *Advances in Neural Information Processing Systems*, pages 4403–4413, 2018.

[39] Amos Fiat and Gerhard J Woeginger. *Online algorithms: The state of the art*, volume 1442. Springer, 1998.

[40] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

[41] Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized Markov decision processes. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2160–2169. PMLR, 2019.

[42] Wenhao Yang, Xiang Li, and Zhihua Zhang. A regularized approach to sparse optimal policy in reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 5938–5948, 2019.

[43] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003. ISSN 0167-6377.

[44] David H Gutman and Javier F Peña. A unified framework for bregman proximal methods: subgradient, gradient, and accelerated gradient schemes. *arXiv*, pages arXiv–1812, 2018.

[45] Tu Nguyen, Trung Le, Hung Vu, and Dinh Phung. Dual discriminator generative adversarial nets. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, 2017.

[46] LI Chongxuan, Taufik Xu, Jun Zhu, and Bo Zhang. Triple generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 4088–4098, 2017.

[47] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning*, pages 1856–1865, 2018.

[48] Constantino Tsallis. Possible generalization of boltzmann-gibbs statistics. *Journal of statistical physics*, 52(1-2):479–487, 1988.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 8.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 8.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 5.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix A.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A] Our emprical studies can be reproduced by from the detailed information in Appendices B & C. The code and data will be available in the final version.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix C.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Section 7.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A] In experiments, each algorithm was exectued in CPU (a single thread).
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A] The MuJoCo simulator used in our experiments is freely available to everyone. See the site (https://mujoco.org).
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] We are preparing to release the source code and data in the final version.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]