Learning Noise-Induced Reward Functions for Surpassing Demonstrations in Imitation Learning

Liangyu Huo, Zulin Wang, Mai Xu*

School of Electronic and Information Engineering, Beihang University
37 Xueyuan Road, Haidian District, Beijing, P.R. China, 100191
{huoliangyu, wzulin, maixu}@buaa.edu.cn

Abstract

Imitation learning (IL) has recently shown impressive performance in training a reinforcement learning agent with human demonstrations, eliminating the difficulty of designing elaborate reward functions in complex environments. However, most IL methods work under the assumption of the optimality of the demonstrations and thus cannot learn policies to surpass the demonstrators. Some methods have been investigated to obtain better-than-demonstration (BD) performance with inner human feedback or preference labels. In this paper, we propose a method to learn rewards from suboptimal demonstrations via a weighted preference learning technique (LERP). Specifically, we first formulate the suboptimality of demonstrations as the inaccurate estimation of rewards. The inaccuracy is modeled with a reward noise random variable following the Gumbel distribution. Moreover, we derive an upper bound of the expected return with different noise coefficients and propose a theorem to surpass the demonstrations. Unlike existing literature, our analysis does not depend on the linear reward constraint. Consequently, we develop a BD model with a weighted preference learning technique. Experimental results on continuous control and high-dimensional discrete control tasks show the superiority of our LERP method over other state-of-the-art BD methods.

Introduction

For the past few years, imitation learning (IL) has shown excellent performance in various real-world applications, such as robot manipulation and autonomous control systems (Barakova and Vanderelst 2011). Unlike reinforcement learning (RL), IL methods train an agent from human demonstrations, neglecting the constraint of preprogrammed reward signals. The gamut of IL research can be roughly classified into two branches: behavioral cloning (BC) and inverse reinforcement learning (IRL). A fundamental assumption in most IL methods is that the demonstration is optimal (Ho and Ermon 2016; Peng et al. 2019). These methods aim to mimic the demonstrator's behavior and typically find policies whose performance is upper bounded by the demonstrator. However, access to high-quality expert data is expensive in real-world scenarios such as robotics learning, where the demonstrator is expected to have domain-specific knowledge. Previous methods for learning better-than-demonstration (BD) policies either require extra handcrafted reward signals or a human supervisor to score the agent during the training stage (Brown et al. 2019; Christiano et al. 2017). To overcome these issues, we propose an innovative BD method of imitation learning without additional information in this paper.

Recent research has advanced BD methods from the perspective of introducing perturbation in the action space (Brown, Goo, and Niekum 2020; Chen, Paleja, and Gombolay 2020a), matching demonstrations with different discount factors (Tao and Cao 2020), or learning intrinsic reward signals via curiosity-driven state exploration (Yu, Lyu, and Tsang 2020). Nevertheless, our method regards the suboptimality of demonstrations as an inaccurate estimation of the reward function due to the demonstrator's limited attention, inaccurate cognition, and observation error (Zheng, Liu, and Ni 2014). This starting point is inspired by the concept of bounded rationality in economics (McKelvey and Palfrey 1995), which has also been investigated in interactive RL research (Faulkner, Short, and Thomaz 2020). Moreover, most above BD methods are heuristic and lack theoretical discussion. In contrast, we theoretically analyze the rationale behind our method and propose a feasible training framework.

In this paper, we analyze the suboptimality of demonstrations from the perspective of perturbed reward functions and then propose a BD method called LEarning Rewards from suboptimal demonstrations with weighted Preference learning (LERP). First, we formulate the generation of suboptimal demonstrations as additive noise on the reward function. Then, an analytical expression of the suboptimal policy is derived under the Gumbel noise distribution. We further elaborate on an upper bound of the expected accumulated reward gap under different noise coefficients and propose a sufficient condition to achieve BD performance. Consequently, a modified preference learning algorithm is developed to infer the noise-induced reward, building on an existing BD method (Brown, Goo, and Niekum 2020). With the aforementioned upper bound interpretation, the difference of injected noise coefficients (Brown, Goo, and Niekum 2020) is quantitatively utilized to correct the preference objective function. The revised expression of the loss function can be considered as an importance sampling technique.

Our experiment comprehensively evaluates the proposed

*Corresponding author.

LERP method across several high-dimensional discrete Atari tasks and continuous virtual robotic MuJoCo tasks. The empirical results illustrate that our method outperforms other BD methods and demonstrations without ground-truth reward signals. To the best of our knowledge, our method is the first to explore exceeding the demonstrations in IL from the perspective of the noisy reward function. Our contribution is mainly three-fold: 1) we formulate suboptimality as reward noise and derive the analytical expression of the suboptimal policy based on Luce's choice axiom (LCA) (Pleskac 2015); 2) we discuss the BD's sufficient condition based on noisy rewards and conclude several principles to design BD frameworks; 3) we combine our theoretical findings with preference learning to develop a new BD method.

Related Work

Imitation learning. The BC method (Bain and Sammut 1995; Ross, Gordon, and Bagnell 2011; Torabi, Warnell, and Stone 2018) is the simplest way of imitation learning, which directly learns the policy mapping from expert demonstrations via supervised learning. From another perspective, the IRL method (Coates, Abbeel, and Ng 2008; Ng and Russell 2000; Abbeel and Ng 2004) infers the latent reward function from demonstrations and then trains the policy with forward RL frameworks. Recently, adversarial imitation learning was proposed to match state-action pairs between the agent and the expert by simultaneously learning the reward and policy. For instance, GAIL (Ho and Ermon 2016) introduces the generative adversarial network (GAN) (Goodfellow et al. 2014) to produce trajectories by a generator and infers the reward by a discriminator. However, GAIL is observed to perform well in low-dimensional tasks but fails to scale to high-dimensional pixel environments such as Atari tasks (Brown et al. 2019). Furthermore, Justin *et al.* (Fu, Luo, and Levine 2017) developed GAIL and proposed a method called AIRL to strengthen the correlation between the policy and the reward function. However, these methods commonly assume the optimality of the demonstrations. This assumption cannot hold in most practical tasks (Laidlaw and Russell 2021). By contrast, our method can learn from non-expert demonstrations, which may render the application of the above methods intractable or ineffective.

Learning from suboptimal demonstrations. Some studies slightly relax the optimality assumption. For instance, Ratliff *et al.* (Ratliff, Bagnell, and Zinkevich 2006; Ratliff, Silver, and Bagnell 2009) introduced slack variables in the optimization objective. Bayesian IRL (Ramachandran and Amir 2007; Zheng, Liu, and Ni 2014) and maximum entropy IRL methods (Ziebart et al. 2008) utilize a probabilistic framework to account for perturbed demonstrations. However, these methods are susceptible to the selected features, and it is challenging to achieve BD performance. Some work has attempted to learn a BD policy with the extra demonstrators' information or hand-craft labels (Shiarlis, Messias, and Whiteson 2016; Syed and Schapire 2007). Other methods (Burchfiel, Tomasi, and Parr 2016; Brown et al. 2019; Laidlaw and Russell 2021; Wu et al. 2019; Zhang et al. 2021) were designed to utilize ranking information among demonstrated trajectories to train the reward function by preference learning. However, these methods are impractical when human ranking or querying is inaccessible. Research (Kaiser, Friedrich, and Dillmann 1995; Laidlaw and Russell 2021; Nussenbaum and Hartley 2019) have shown that the main reasons for the suboptimality of demonstrations can be summarized into the following three aspects: (1) perturbation on actions; (2) generalization of the demonstrations; and (3) uncertain intention and inaccurate reward estimation. Most recent BD methods without added supervised labels fall into the first aspect. Especially, Brown *et al.* (Brown, Goo, and Niekum 2020) proposed a state-of-the-art BD method called DREX, which uses preference learning on synthetic trajectories by injecting noise into a learned BC agent. Moreover, Yu *et al.* (Yu, Lyu, and Tsang 2020) developed a curiosity-driven method, referred to as GIRIL, to create an intrinsic reward signal via a generative model. These methods are heuristic without theoretical clarification or work under the linear reward constraint. In contrast, our method learns BD policies by modeling the inaccuracy in the reward space, and we elucidate the theoretical fundamentals without any constraints on the form of reward functions.

Method

In this section, the fundamental of IL and BD objectives is first presented. Then, we propose a noisy reward assumption to formulate suboptimal demonstrations. Suboptimal policies and BD conditions are further derived with a specific noise distribution. Finally, we combine the theoretical conclusions with preference learning to develop our method.

Problem Formulation

Markov decision process (MDP) models an agent's sequential decision-making process, which can be formulated as $\mathcal{M} := (\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}, \gamma)$. $\mathcal{S}$ is a set of states, and $\mathcal{A}$ is a set of actions. Mapping $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ defines the environment transition dynamics to the next state, which is unknown in the model-free setting. The reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ defines the reward signal of each state-action pair, and γ is a discount factor. Without loss of generality, for any reward function r, we denote $R_{\max}$ as the bounded reward, i.e., $|r(s, a)| \leq R_{\max}, \forall (s, a) \in \mathcal{S} \times \mathcal{A}$. A policy π is a probability distribution over actions given a particular state. The value function V^π represents the expected discount return, defined as $V^\pi(s) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t), |s_0 = s\right]$. The fundamental goal of RL is to find the optimal policy $\pi^* = \arg\max_\pi V^\pi(s), s \sim d_0$, where d_0 is denoted as the initial state distribution. In IL, the actual reward function of the MDP is not available. Let $D = \{\tau^1, \tau^2, ..., \tau^M\}$ denote a set of M demonstrations, where each trajectory $\tau^i = (s_0, a_0, s_1, a_1, ...)$ consists of the state and the action at each timestep t. For a specific trajectory τ, we can define the discount return under the ground-truth reward function r as $J(\tau|r) = \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)$, s_t, $a_t \in \tau$. BD methods aim to learn a policy exceeding the average return $V^D = \frac{1}{M} \sum_{\tau \in D} J(\tau|r)$. As DREX (Brown, Goo, and Niekum 2020), we assume a demonstrator follows a specific policy that optimizes what he/she thinks is the reward function $\tilde{r}_\epsilon$, defined as an inaccurate personal estimation of r.

Fundamental of Surpassing Demonstrations

Inspired by the quantal response equilibrium (QRE) (McKelvey and Palfrey 1995), we first formulate the noisy reward model of the suboptimal demonstrator. Constrained by limited cognitive abilities, a nonspecialist demonstrator cannot correctly assess the reward function of each state-action pair. This inaccuracy is modeled by additive noise in the reward space in this paper. Formally, the personal noisy reward function $\tilde{r}_\epsilon(s_t, a_t)$ can be expressed as follows:

$$\tilde{r}_\epsilon(s_t, a_t) = r(s_t, a_t) + \beta\epsilon. \tag{1}$$

Here, $r(s_t, a_t)$ represents the ground-truth reward function given s_t and a_t. Referring to LCA, ϵ is a zero-mean random variable introduced by demonstrator-free human cognition, following a distribution function $f(\epsilon)$. β is defined as a demonstrator-specific noise coefficient representing the intensity of reward noise. According to QRE, ϵ is considered independent of the state-action pair and it is sampled at every time when a demonstrator selects an action. We denote the demonstrator's person utility function (action value function) as $\tilde{Q}^\pi_\epsilon$, which can be defined as follows:

$$\begin{aligned}\tilde{Q}^\pi_\epsilon(s,a) &= \mathbb{E}_\pi\left[\sum_{t=0}^{\infty}\gamma^t\tilde{r}_\epsilon(s_t,a_t), |s_0=s, a_0=a\right] \\ &= \tilde{r}_\epsilon(s,a) + \gamma\mathbb{E}_{s',a'\sim\pi}\left[\tilde{Q}^\pi_\epsilon(s',a')\right]\end{aligned} \tag{2}$$

Accordingly, $\tilde{r}_\epsilon(s,a)$ contribute to a new MDP with noise variable ϵ. Assume that the demonstrator's learning ability is sufficiently good. For an optimal personal policy $\tilde{\pi}^*_\epsilon$, the following lemma indicates that the corresponding personal utility function $\tilde{Q}^{\tilde{\pi}^*_\epsilon}_\epsilon$ also follows the form of additive noise.

Lemma 1 *Given a ground-truth optimal action value function $Q^*(s,a)$, the personal optimal action value function $\tilde{Q}^{\tilde{\pi}^*_\epsilon}_\epsilon(s,a)$ can be approximated as follows:*

$$\tilde{Q}^{\tilde{\pi}^*_\epsilon}_\epsilon(s,a) \approx Q^*(s,a) + \hat{\beta}\epsilon, \tag{3}$$

where $\hat{\beta} = \beta/(1-\gamma)$ is a normalized noise coefficient.

Refer to the full version of our paper for all the proofs. Briefly, *Lemma 1* holds under the mathematical expectation of ϵ. Adding a noise variable into the action value function to represent the randomization has shown feasibility in transfer learning (Cheng et al. 2022). Here, despite various choices of the noise distribution $f(\epsilon)$, they are required to satisfy certain properties to ensure a minor error of (3) during sampling. Following Chebyshev's inequality, for any given positive error constant δ_0, we can compute the error as follows:

$$P\left\{|\tilde{Q}^{\tilde{\pi}^*_\epsilon}_\epsilon - Q^*| \geq \delta_0\right\} \leq \frac{D(\hat{\beta}\epsilon)}{\delta_0^2} = \frac{\hat{\beta}^2 D(\epsilon)}{\delta_0^2}, \tag{4}$$

where $D(\epsilon)$ represents the variance of ϵ. For a demonstrator that is not optimal but rational enough, the noise coefficient $\hat{\beta}$ is not significant. In addition, for a zero-mean ϵ, the tail of the distribution function $f(\epsilon)$ has to be sufficiently small. Gaussian distribution (Kingma and Welling 2014) and Laplacian distribution (Zheng, Liu, and Ni 2014) are commonly used to model the noise variable. However, inspired by QRE (McKelvey and Palfrey 1995), the reward noise in this paper is formulated as an additive random variable with a Gumbel distribution, which is an extreme value distribution used to model the maximum of random variables. Following LCA, a demonstrator takes the maximum of the utility function to compute decision variables, which is consistent with Gumbel's idea. As shown later in the paper, this choice leads to a broadly used expression of the suboptimal policy in Bayesian IL (Ramachandran and Amir 2007).

Here, we focus on the analytical expression of the suboptimal policy with Gumbel reward noise. With a deterministic policy, a well-trained demonstrator selects the action with the maximal personal optimal $\tilde{Q}^{\tilde{\pi}^*_\epsilon}_\epsilon$ given the current state. Moreover, for a given state s, the demonstrator's reward error vector $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, ..., \epsilon_{|\mathcal{A}|})$ follows a joint distribution with density function $f_{\boldsymbol{\epsilon}}(\boldsymbol{\epsilon})$. From the behavioral decision rule, the demonstrator selects action a_k if and only if $\tilde{Q}^{\tilde{\pi}^*_\epsilon}_\epsilon(s,a_k) \geq \tilde{Q}^{\tilde{\pi}^*_\epsilon}_\epsilon(s,a_l), \forall l = 1,2,...,|\mathcal{A}|$. Therefore, we formally define the set of error vectors $\boldsymbol{\epsilon} \in \mathbb{R}^{|\mathcal{A}|}$ with which the demonstrator selects a_k under state s as follows:

$$\left\{\boldsymbol{\epsilon} \mid Q^*(s,a_k) + \hat{\beta}\epsilon_k \geq Q^*(s,a_l) + \hat{\beta}\epsilon_l, \forall l = 1,\cdots,|\mathcal{A}|\right\}. \tag{5}$$

Here, we use bold $\boldsymbol{\epsilon}$ to denote the vector consisting of the error components over all dimensions of the action space. We assume that the marginal distribution of the noise variable in each action dimension follows the Gumbel distribution with the same scalar parameter β, i.e., $\epsilon_k \sim \text{Gumbel}(0,\beta)$ (we abbreviate any variable ϵ with the probability density distribution $f(\epsilon) = \frac{1}{\beta}\exp\left(-\frac{\epsilon-\mu}{\beta} - \exp\left(-\frac{\epsilon-\mu}{\beta}\right)\right)$ as $\epsilon \sim \text{Gumbel}(\mu,\beta)$ in this paper). According to LCA, when a demonstrator selects actions with utility functions, the noise variables are independently sampled for different actions. Thus, any two components of $\boldsymbol{\epsilon}$ in different action dimensions are assumed to be independent. Then, we can derive the suboptimal policy with the following lemma.

Lemma 2 *For any normalized noise coefficient $\hat{\beta} > 0$ and reward noise $\epsilon \sim$ Gumbel$(0,\hat{\beta})$, the suboptimal policy can be expressed as follows:*

$$\pi(a_k|s) = \frac{\exp\left(\lambda Q^*(s,a_k)\right)}{\sum_{l=1}^{|\mathcal{A}|}\exp\left(\lambda Q^*(s,a_l)\right)}, \forall a_k \in \mathcal{A}, \tag{6}$$

where $\lambda = 1/\hat{\beta}$ is a temperature parameter.

Lemma 2 implies that introducing randomness into the reward space followed by training with RL methods can be transformed to introduce randomness into the action space directly. Moreover, the softmax-like policy in (6) follows the same form as in Bayesian IL, where λ suggests the demonstrator's optimality or cognitive ability. However, we propose a different method to utilize (6) in this paper. Our motivation can be divided into two categories. First, most of the traditional methods in Bayesian IL consider a fixed or prior known λ. As shown in reference (Nussenbaum and Hartley 2019), λ is influenced by personal factors and varies in

different demonstrators. Some recently developed methods have attempted to infer λ with sampling techniques. However, directly inferring λ in the inner loop of IRL involves computing Q^*, which renders these methods intractable in large-scale environments. Besides, despite similar expressions, most Bayesian IL methods do not leverage the characteristic of the Gumbel noise distribution. As shown in the following sections, our method takes advantage of this point.

Recall that the essential purpose of BD methods is to learn a policy π, with which the value function is more significant than the average discount return of demonstrations, i.e.,

$$V^{\pi} > V^{D}. \tag{7}$$

We omit the state s for brevity when there is no ambiguity. Here, we provide a sufficient condition under which it is possible to arrive at (7) in a standard IRL framework. Reference (Brown, Goo, and Niekum 2020) proposed this condition with the linear reward constraint. Contrarily, we remove this restriction and expand it in a more general case. For a given state s, let a^* denote the optimal response action with Q^*, i.e., $a^* = \arg\max_a Q^*(s,a)$. Here, for illustration, we expand the action space to a continuous space and regard valid actions as discrete values. Accordingly, the optimal policy is written in Dirac function δ as $\pi^*(a \mid s) = \delta(a - a^*)$. As the Dirac function can be regarded as a limiting form of the Gaussian function, we rewrite the optimal policy as follows:

$$\pi^*(a \mid s) = \lim_{\sigma \to +0} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(a-a^*)^2}{2\sigma^2}\right). \tag{8}$$

Here, $\sigma > 0$ is denoted as the standard deviation of the Gaussian distribution. Consequently, the BD condition can be described by the following theorem.

Theorem 1 *Let notations V^* and d^* denote the optimal value function and the corresponding state distribution. The BD condition, i.e., $V^{\pi} > V^{D}$ is guaranteed if:*

$$V^*(s) - V^D(s) > \frac{2\sqrt{2}R_{\max}}{(1-\gamma)^2}\sqrt{\mathbb{E}_{s\sim d^*}\Big[\Phi^{\sigma}_{Q^*}(s) - \lambda\Psi_{Q^*}(s)\Big]}, \tag{9}$$

where $\Phi^{\sigma}_{Q^*}(s) > 0$ and $\Psi_{Q^*}(s) > 0$ are two polynomials depending on the state s and the optimal action value Q^*.

Intuitively, *Theorem 1* illustrates three following conclusions that render learning a BD policy easier. First, the demonstration is sufficiently suboptimal, which is also indicated in DREX (Brown, Goo, and Niekum 2020). Second, the factor $1/(1-\gamma)$ represents the effective length of an infinite horizon MDP, which leads to the compounding error in IL. Third, the learned agent's temperature parameter λ is sufficiently large, suggesting that the learned reward function is sufficiently close to the ground-truth reward. In other words, if we accurately recover the reward function, then an RL method can be employed to ensure the BD condition.

Weighted Preference Learning Objective

To utilize the above conclusions, we further propose a feasible BD method based on preference learning. Recently, rank-based learning techniques have shown promising results in recovering rewards (Brown et al. 2019; Brown, Goo, and Niekum 2020; Hester et al. 2018). These methods explore fitting human rankings of demonstrations to learn the reward indirectly. Since collecting the ranked training set needs extra labels or prior knowledge, some studies have attempted to introduce perturbation to create ranked demonstrations (Laskey et al. 2017) automatically. For example, Brown *et al.* (Brown, Goo, and Niekum 2020) proposed the DREX method via automatically ranked demonstrations. To train the model, DREX first uses BC to learn an initial policy π_0 from the original demonstration D. Then, it synthesizes trajectories τ_η (we use the subscript to represent it is a synthetic rather than a natural trajectory when it is necessary to distinguish) by injecting different levels of noise to π_0 as follows:

$$\tau_\eta \sim \pi_\eta(a \mid s) = \eta U(a) + (1-\eta)\pi_0(a \mid s), \tag{10}$$

where η represents the noise level, and U represents a uniform distribution. Based on the assumption (Brown, Goo, and Niekum 2020), a synthetic trajectory with higher η suggests a lower discount return, i.e., $J(\tau_i|r) \leq J(\tau_j|r)$, $\forall \eta_i \geq \eta_j$, $\tau_i \sim \pi_{\eta_i}$, $\tau_j \sim \pi_{\eta_j}$. Finally, DREX designs a parameterized reward function $r_\theta(s)$ and maximizes a pairwise preference learning objective J_{D} based on the Luce-Shepard rule (Luce 2012) as follows:

$$J_{\mathrm{D}}(\theta) = \sum_{(\tau_i,\tau_j)\in\mathcal{P}} \log \frac{\exp\left(\sum_{s\in\tau_i} r_\theta(s)\right)}{\exp\left(\sum_{s\in\tau_i} r_\theta(s)\right) + \exp\left(\sum_{s\in\tau_j} r_\theta(s)\right)}, \tag{11}$$

where $\mathcal{P} = \{(i,j) : \eta_i < \eta_j\}$ is the set of ranked synthetic trajectories.

Similar to DREX, most above methods start from the perspective of the randomness in the action space. By contrast, we turn to the perspective of the reward space based on our noisy reward formulation. Here, we first analyze the validity of our motivation, i.e., whether different levels of noise in the reward space results in automatically ranked trajectories.

We denote β_0 as the initial noise coefficient of the demonstrator, which is unknown to the IL method. We can consider an additive random noise variable that also follows a Gumbel distribution. For convenience, we analyze the effectiveness of two different noise variables. Nevertheless, the conclusion can be extended to general cases. Precisely, let β_i and β_j denote two introduced noise coefficients and $\beta_i \geq \beta_j$. According to *Lemma 3*, the optimal value gap of π_i is more significant than π_j, meaning that π_j is more likely to obtain BD performance than π_i. This conclusion is consistent with DREX (Brown, Goo, and Niekum 2020). Thus, it ensures the effectiveness of introducing noise into the reward space to synthesize ranked demonstrations automatically. However, another problem of DREX is that it ignores the quantitative relationship between noise levels, which is elaborated in reference (Chen, Paleja, and Gombolay 2020a). To alleviate this problem, researchers (Chen, Paleja, and Gombolay 2020a) designed a sigmoid low-pass filter to force the reward function to connect with the noise level. It is worth noting that this filter needs recomputing

for each environment, and it lacks a theoretical interpretation. This paper theoretically discusses the effect of different noise coefficients on the value function in the following lemma and then modifies the preference learning objective of (11) to infer rewards more accurately.

Lemma 3 *For any state s and two positive noise coefficients β_i and β_j, if $\beta_i \geq \beta_j$, the upper bound of the value gap is proportional to $\sqrt{\beta_i - \beta_j}$, i.e.,*

$$\sup_s V^{\pi_j}(s) - V^{\pi_i}(s) \propto \sqrt{\beta_i - \beta_j}. \tag{12}$$

Here, *Lemma 3* indicates the quantitative relationship between the value gap and the difference of reward noise coefficients. Because of the convexity of the square root function, the value gap grows fast when the noise difference is significant. Along with the policies being optimized, the influence of the noise difference gradually decreases. Hence, this relationship can be utilized in preference learning. However, two difficulties make it impractical to explicitly use (12) in the reward space. First, β_0 of the demonstrator is unknown, leading to the imprecise injection of temperature parameters. Second, the ground-truth reward function is not accessible during the IL training stage. Nevertheless, *Lemma 2* has shown the equivalency between introducing randomness into the reward and action spaces. Thus, we extend *Lemma 3* with the noise injection of DREX to conclude the following corollary.

Corollary 1 *For any state s and two positive noise levels η_i and η_j, if $\eta_i \geq \eta_j$, the upper bound of the value gap is proportional to $\sqrt{\log \eta_i - \log \eta_j}$, i.e.,*

$$\sup_s V^{\pi_j}(s) - V^{\pi_i}(s) \propto \sqrt{\log \eta_i - \log \eta_j}. \tag{13}$$

Inspired by reference (Kovalchik 2020), we modify the objective in (11) to adjust to the noise difference. Specifically, for any two synthetic trajectories τ_i, τ_j in ranked demonstration P, which are caused by injecting noise η_i, η_j and $\eta_i > \eta_j$, our model outputs a reward $r_\theta(s,a)$ to optimize the objective function J_{L} as follows:

$$J_{\mathrm{L}}(\theta) = \sum_{(\tau_i,\tau_j)\in\mathcal{P}} \log \frac{1}{1+\exp\left(\frac{-\left(\sum_{s,a\in\tau_i} r_\theta(s,a) - \sum_{s,a\in\tau_j} r_\theta(s,a)\right)}{\sqrt{\log \eta_i - \log \eta_j}}\right)}. \tag{14}$$

Like other pairwise ranking learning, this objective function operates in a supervised learning framework and trains a binary classifier that predicts whether one trajectory is better than the other based on the predicted returns. The weighted factor $1/\sqrt{\log \eta_i - \log \eta_j}$ can be regarded as introducing an importance sampling technique (Tokdar and Kass 2010) to the original preference objective. As pointed out in reference (Bobu et al. 2020), the objective needs to be more refined with the similarity of trajectories. When the two trajectories are closer, the factor becomes more noteworthy, indicating that this pair of training demonstrations has a considerable weight factor. This idea intuitively coincides with how humans learn and keep improving.

Experiment

Experimental Setup

In the experiment, we evaluated the proposed LERP method on several continuous MuJoCo tasks (Todorov, Erez, and Tassa 2012) and discrete Atari tasks (Bellemare et al. 2013) within OpenAI Gym (Brockman et al. 2016), which are chosen in related studies (Brown, Goo, and Niekum 2020; Yu, Lyu, and Tsang 2020; Brown et al. 2019). To create initial demonstrations, we used a partially trained Proximal Policy Optimization (PPO) (Schulman et al. 2017) agent with ground-truth rewards for several simulation steps.

For MuJoCo tasks, the under-trained checkpoints were used to generate demonstrations with the length of 1000 timesteps. Then, BC was employed to learn the initial policy π_0 with the early stop trick to prevent overfitting. Similar to DREX, we injected 20 levels of noise into π_0, i.e., $\eta = \{0.00, 0.05, 0.10, ..., 0.95\}$, and collected 10 interacted trajectories for each level. In this way, we built a synthetic ranked demonstration of 5000 paired trajectory snippets, which were sampled at fixed intervals with the auxiliary label η to compute the weighted factor. For Atari tasks, we generated 20 initial demonstrations of each task. The noise level η was sampled from $\{0.05, 0.25, 0.50, 0.75, 0.95\}$, and we collected 20 trajectories of each level and synthesized 15000 ranked demonstrations.

Performance of Surpassing Demonstrations

First, we evaluated the performance of the agent trained by the learned reward function. We compared our LERP method with behavioral cloning (BC) (Bain and Sammut 1995), several state-of-the-art IRL methods, i.e., GAIL (Ho and Ermon 2016), LESS (Bobu et al. 2020) (for MuJoCo tasks), SSRR (Chen, Paleja, and Gombolay 2020b) (for MuJoCo tasks), DREX (Brown, Goo, and Niekum 2020), and GIRIL (Yu, Lyu, and Tsang 2020) (for Atari tasks). The last two methods are also designed for learning from suboptimal demonstrations. LESS is a modified version of LERP where the weighted factor is computed by the similarity of state-action pairs (Bobu et al. 2020). For a fair comparison, we used the official implementations of these methods with the same network architecture as our method. During the RL training stage, we blocked the ground-truth reward signal from the environment. Then, we optimized several agents with the learned reward functions of LERP and other IRL methods by PPO. Table 1 shows the final performance in terms of average returns on five MuJoCo tasks and seven Atari tasks. As shown in Table 1, LERP performs the best on 10 out of 12 tasks among the IL methods, particularly on Hopper, Kong Fu Master, Humanoid, and Space Invaders. In comparison, DREX achieves the highest return on Q*bert. However, as DREX neglects the quantitative ranking relation in preference learning, it performs mediocrely in most Atari tasks and even fails on some tasks such as Walker2d and Pong. In contrast to DREX, our LERP method utilizes the difference of noise levels quantitatively, which is significant in improving stability. For example, LERP achieves almost twice as many returns as DREX on Ant, Hopper and Kong Fu Master. On the other side, since LESS also refines

	BC		Random		DREX		GIRIL		GAIL		LESS		LERP	
Environment	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.
HalfCheetah	393.4	48.8	-291.6	82.7	908.0	461.5	-	-	-11.8	22.5	356.3	482.5	**1106.8**	26.3
Hopper	973.2	11.6	15.9	8.9	1541.1	716.9	-	-	1086.2	569.0	2347.0	930.5	**2808.6**	471.4
Ant	697.4	24.6	-57.3	98.7	394.8	173.6	-	-	**738.9**	65.0	684.1	25.6	612.7	26.2
Walker2d	232.5	42.8	1.7	5.4	1493.9	316.3	-	-	296.4	88.1	-7.1	2.5	**1825.7**	234.8
Humanoid	201.2	14.1	117.7	26.9	436.8	201.8	-	-	161.0	1.9	443.8	83.2	**633.7**	107.3
Enduro	113.5	77.2	0.0	0.0	565.0	137.7	0.5	0.2	52.8	5.8	-	-	**704.4**	211.8
Kung Fu Master	8785.0	4875.4	450.0	261.3	9570.0	3501.2	18200.0	4281.6	1211.2	196.5	-	-	**19520.0**	5625.4
Pong	-13.0	9.2	-20.2	0.9	-20.6	0.4	-15.5	7.4	-20.6	0.8	-	-	**-8.2**	9.8
Q*bert	560.0	69.5	137.5	121.6	**27967.5**	12629.1	2370.0	1496.9	532.1	24.0	-	-	9622.5	3790.3
Seaquest	240.0	70.2	96.0	60.9	670.0	51.5	672.0	65.8	114.2	9.7	-	-	**762.0**	26.0
Space Invaders	187.5	84.9	134.0	94.3	618.5	126.8	705.5	286.6	179.1	69.5	-	-	**936.0**	530.1
Ms Pacman	300.0	154.0	150.0	209.5	690.0	122.1	552.0	215.7	172.4	9.6	-	-	**758.0**	259.8

Table 1: Performance of each method on MuJoCo and Atari tasks regarding the average return (Avg.) and the standard deviation (Stdev.) over nine trials. Bold and underline indicate the best and second-best performance on each task throughout this paper.

	Synthetic Demonstrations				Initial Demonstrations				Unseen Demonstrations			
Environment	DREX	ME-IRL	LERP-L	LERP	DREX	ME-IRL	LERP-L	LERP	DREX	ME-IRL	LERP-L	LERP
HalfCheetah	0.900	0.820	**0.908**	0.907	0.748	0.728	**0.946**	0.778	0.791	-0.372	0.830	**0.899**
Hopper	0.987	0.866	**0.994**	0.990	0.991	0.121	**0.998**	0.993	-0.909	-0.893	-0.830	**-0.624**
Ant	0.950	0.830	0.933	**0.954**	0.733	0.514	**0.852**	0.826	0.952	0.954	0.913	**0.963**
Walker2d	0.944	0.912	0.925	**0.949**	0.832	-0.237	0.840	**0.877**	-0.069	**0.529**	0.421	-0.122
Humanoid	0.449	0.639	**0.816**	0.473	0.140	0.573	0.106	**0.581**	0.877	-0.634	0.608	**0.964**

Table 2: Average return correlation coefficients with the ground-truth return on five MuJoCo tasks.

the objective of DREX with state-action similarity, it can improve the performance of DREX on most tasks, but LERP still outperforms LESS.

For a clear comparison, Figure 1 shows the smoothed last-100-episode return curves of these IRL methods on all tasks. We can observe that in most tasks, the learned reward function with LERP leads to a faster convergence than DREX. For example, LERP trains a competitive agent on Hopper after approximately 5 million timesteps, whereas DREX begins to converge at 8 million timesteps. Referring to the ability to exceed the demonstrations, the learned return of LERP outperforms the best demonstrations on nine tasks. Moreover, LERP often surpasses the average return of the demonstrations by a significant margin, for example, by 5.9 times better on HalfCheetah, 3.3 times better on Walker2d, and 2.1 times better on Space Invaders. Overall, LERP results in an average performance increase of 117% and 86% across all tasks, compared with the average and best demonstrations. The aforementioned results illustrate the effectiveness of learning BD policies with the proposed LERP method.

Analysis of Reward Inference

To verify whether the quantitative usage of noise difference can contribute to more accurate reward functions, we evaluated the preference accuracy of the learned reward model on unseen trajectory pairs. We divided the results by the difference of the corresponding noise levels. Here, we refer to the top sixth of the test pairs ranked by noise similarity as "similar pairs" and the complete set as "entire pairs". We added a linear modified version of LERP (denoted by LERP-L) where the weighted factor $1/\sqrt{\log \eta_i - \log \eta_j}$ was replaced by $|\eta_i - \eta_j|$. The results are shown in Figure 2. It turns out that LERP obtains the average accuracy of 82.0% (92.7%) in the similar (entire) pairs, versus 76.3% (89.9%) for DREX. These suggest that the noise weight factor can improve preference accuracy, especially when the trajectory pairs are similar. Moreover, we studied the reconstruction and the extrapolation ability of the learned reward model. For this purpose, we evaluated the learned reward on the following different kinds of trajectories: (1) sampled in initial demonstrations D; (2) sampled in synthetic demonstrations P; (3) sampled in unseen demonstrations generated by a better PPO agent. These unseen trajectories suggested how well the learned reward function can extrapolate beyond the training set. Then, we compared the correlation between the ground-truth returns of these trajectories and the predicted returns by DREX, LERP, LERP-L with another IRL baseline ME-IRL (Ziebart et al. 2008). As shown in Table 2, LERP and LERP-L recover the most accurate rewards in the five methods on 5/5, 5/5, and 4/5 tasks in synthetic, initial, and unseen demonstrations, respectively. A particularly notable case is HalfCheetah, where LERP achieves a correlation of 0.899 with the ground-truth reward, versus only -0.372 and 0.791 for ME-IRL and DREX in unseen demonstrations. Moreover, LERP-L yields similar correlation coefficients to LERP in synthetic and initial demonstrations on most tasks. However, LERP performs better than LERP-L in unseen demonstrations, especially on Humanoid. These

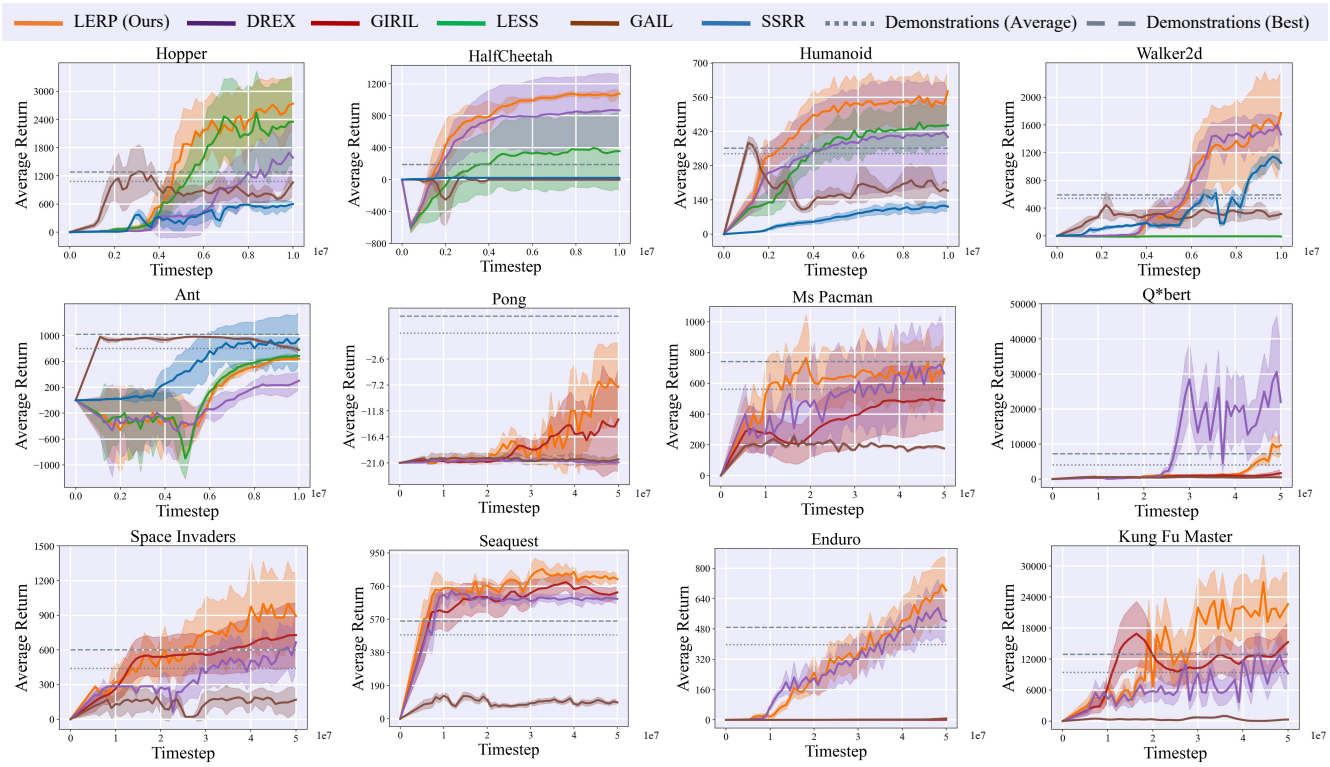

Figure 1: Average return with simulation timesteps on MuJoCo and Atari tasks. The solid lines show the average performance over nine trials and the shaded area represents the standard deviation.

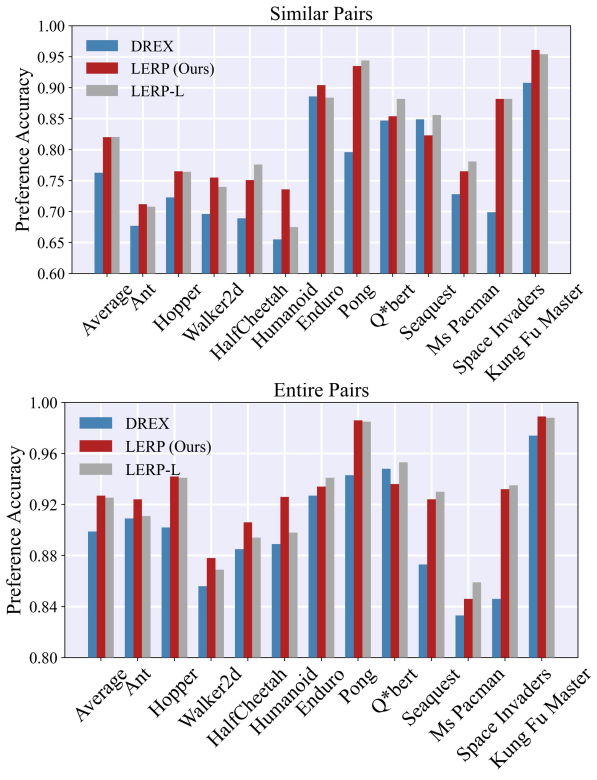

Figure 2: Preference accuracy on similar and entire pairs.

suggest that the weighted factor $1/\sqrt{\log \eta_i - \log \eta_j}$ can further improve the generalization of the learned reward model compared with the linear weight.

Conclusion

In this paper, we proposed a LERP method for surpassing demonstrations in IL. First, we formulated the suboptimality of demonstrations as an inaccurate estimation of the reward function, which is modeled by additive noise. Then, under the Gumbel noise distribution, we derived an upper bound of the expected accumulated reward gap with different noise parameters. Moreover, we discussed the theoretical condition of surpassing demonstrations and concluded several principles for achieving BD performance. According to our theoretical findings, we developed a new method for noise-induced reward inference based on a weighted preference learning technique. Empirical results show that our LERP method outperforms both the demonstrations and other state-of-the-art BD methods. One limitation is that the quantitative suboptimal degree of the demonstrations with which our method shows more effectiveness than the standard IL methods is not well investigated in this paper, which we regard as future work. Moreover, we only applied the proposed imperfect reward modeling with the concept of bounded rationality in preference learning based IL methods. However, we believe it can be inspiring to other RL and IL domains.

Acknowledgments

This work was supported by the NSFC projects 62250001, 62231002, 61922009, 61876013 and 61925105.

References

Abbeel, P.; and Ng, A. Y. 2004. Apprenticeship learning via inverse reinforcement learning. In *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*, volume 69 of *ACM International Conference Proceeding Series*.

Bain, M.; and Sammut, C. 1995. A Framework for Behavioural Cloning. In *Machine Intelligence 15*, 103–129.

Barakova, E. I.; and Vanderelst, D. 2011. From Spreading of Behavior to Dyadic Interaction-A Robot Learns What to Imitate. *International Journal of Intelligent Systems*, 26(3): 228–245.

Bellemare, M. G.; Naddaf, Y.; Veness, J.; and Bowling, M. 2013. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47: 253–279.

Bobu, A.; Scobee, D. R.; Fisac, J. F.; Sastry, S. S.; and Dragan, A. D. 2020. Less is more: Rethinking probabilistic models of human behavior. In *Proceedings of the 2020 acm/ieee international conference on human-robot interaction*, 429–437.

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; and Zaremba, W. 2016. Openai gym. *ArXiv preprint*, abs/1606.01540.

Brown, D.; Goo, W.; Nagarajan, P.; and Niekum, S. 2019. Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations. In *International conference on machine learning*, 783–792. PMLR.

Brown, D. S.; Goo, W.; and Niekum, S. 2020. Better-than-demonstrator imitation learning via automatically-ranked demonstrations. In *Conference on robot learning*, 330–359. PMLR.

Burchfiel, B.; Tomasi, C.; and Parr, R. 2016. Distance Minimization for Reward Learning from Scored Trajectories. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, 3330–3336.

Chen, L.; Paleja, R.; and Gombolay, M. 2020a. Learning from suboptimal demonstration via self-supervised reward regression. *ArXiv preprint*, abs/2010.11723.

Chen, L.; Paleja, R.; and Gombolay, M. 2020b. Learning from suboptimal demonstration via self-supervised reward regression. *ArXiv preprint*, abs/2010.11723.

Cheng, Z.; Ye, D.; Zhu, T.; Zhou, W.; Yu, P. S.; and Zhu, C. 2022. Multi-agent reinforcement learning via knowledge transfer with differentially private noise. *International Journal of Intelligent Systems*, 37(1): 799–828.

Christiano, P. F.; Leike, J.; Brown, T. B.; Martic, M.; Legg, S.; and Amodei, D. 2017. Deep Reinforcement Learning from Human Preferences. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 4299–4307.

Coates, A.; Abbeel, P.; and Ng, A. Y. 2008. Learning for control from multiple demonstrations. In *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, volume 307 of *ACM International Conference Proceeding Series*, 144–151.

Faulkner, T. A. K.; Short, E. S.; and Thomaz, A. L. 2020. Interactive reinforcement learning with inaccurate feedback. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 7498–7504. IEEE.

Fu, J.; Luo, K.; and Levine, S. 2017. Learning robust rewards with adversarial inverse reinforcement learning. *ArXiv preprint*, abs/1710.11248.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Hester, T.; Vecerik, M.; Pietquin, O.; Lanctot, M.; Schaul, T.; Piot, B.; Horgan, D.; Quan, J.; Sendonaris, A.; Osband, I.; et al. 2018. Deep q-learning from demonstrations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Ho, J.; and Ermon, S. 2016. Generative Adversarial Imitation Learning. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, 4565–4573.

Kaiser, M.; Friedrich, H.; and Dillmann, R. 1995. Obtaining good performance from a bad teacher. In *Programming by Demonstration vs. Learning from Examples Workshop at ML*, volume 95. Citeseer.

Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Kovalchik, S. 2020. Extension of the Elo rating system to margin of victory. *International Journal of Forecasting*, 36(4): 1329–1341.

Laidlaw, C.; and Russell, S. 2021. Learning the Preferences of Uncertain Humans with Inverse Decision Theory. *ArXiv preprint*, abs/2106.10394.

Laskey, M.; Lee, J.; Fox, R.; Dragan, A.; and Goldberg, K. 2017. Dart: Noise injection for robust imitation learning. In *Conference on robot learning*, 143–156. PMLR.

Luce, R. D. 2012. *Individual choice behavior: A theoretical analysis*.

McKelvey, R. D.; and Palfrey, T. R. 1995. Quantal response equilibria for normal form games. *Games and economic behavior*, 10(1): 6–38.

Ng, A. Y.; and Russell, S. J. 2000. Algorithms for Inverse Reinforcement Learning. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000), Stanford University, Stanford, CA, USA, June 29 - July 2, 2000*, 663–670.

Nussenbaum, K.; and Hartley, C. A. 2019. Reinforcement learning across development: What insights can we draw from a decade of research? *Developmental cognitive neuroscience*, 40: 100733.

Peng, X. B.; Kanazawa, A.; Toyer, S.; Abbeel, P.; and Levine, S. 2019. Variational Discriminator Bottleneck: Improving Imitation Learning, Inverse RL, and GANs by Constraining Information Flow. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Pleskac, T. J. 2015. Decision and choice: Luce's choice axiom. *International encyclopedia of the social & behavioral sciences*, 5: 895–900.

Ramachandran, D.; and Amir, E. 2007. Bayesian inverse reinforcement learning. In *Proceedings of the 20th international joint conference on Artifical intelligence*, 2586–2591.

Ratliff, N. D.; Bagnell, J. A.; and Zinkevich, M. 2006. Maximum margin planning. In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, volume 148 of *ACM International Conference Proceeding Series*, 729–736.

Ratliff, N. D.; Silver, D.; and Bagnell, J. A. 2009. Learning to search: Functional gradient techniques for imitation learning. *Autonomous Robots*, 27(1): 25–53.

Ross, S.; Gordon, G.; and Bagnell, D. 2011. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 627–635. JMLR Workshop and Conference Proceedings.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal policy optimization algorithms. *ArXiv preprint*, abs/1707.06347.

Shiarlis, K.; Messias, J.; and Whiteson, S. 2016. Inverse reinforcement learning from failure. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, 1060–1068.

Syed, U.; and Schapire, R. E. 2007. A Game-Theoretic Approach to Apprenticeship Learning. In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*, 1449–1456.

Tao, F.; and Cao, Y. 2020. Learn to Exceed: Stereo Inverse Reinforcement Learning with Concurrent Policy Optimization. *ArXiv preprint*, abs/2009.09577.

Todorov, E.; Erez, T.; and Tassa, Y. 2012. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 5026–5033. IEEE.

Tokdar, S. T.; and Kass, R. E. 2010. Importance sampling: a review. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(1): 54–60.

Torabi, F.; Warnell, G.; and Stone, P. 2018. Behavioral Cloning from Observation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, 4950–4957.

Wu, Y.-H.; Charoenphakdee, N.; Bao, H.; Tangkaratt, V.; and Sugiyama, M. 2019. Imitation learning from imperfect demonstration. In *International Conference on Machine Learning*, 6818–6827. PMLR.

Yu, X.; Lyu, Y.; and Tsang, I. W. 2020. Intrinsic Reward Driven Imitation Learning via Generative Model. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, 10925–10935.

Zhang, S.; Cao, Z.; Sadigh, D.; and Sui, Y. 2021. Confidence-Aware Imitation Learning from Demonstrations with Varying Optimality. *Advances in Neural Information Processing Systems*, 34: 12340–12350.

Zheng, J.; Liu, S.; and Ni, L. M. 2014. Robust Bayesian Inverse Reinforcement Learning with Sparse Behavior Noise. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada*, 2198–2205.

Ziebart, B. D.; Maas, A. L.; Bagnell, J. A.; and Dey, A. K. 2008. Maximum Entropy Inverse Reinforcement Learning. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008*.