IMITATION WITH NEURAL DENSITY MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose a new framework for Imitation Learning (IL) via density estimation of the expert's occupancy measure followed by Maximum Occupancy Entropy Reinforcement Learning (RL) using the density as a reward. Our approach maximizes a non-adversarial model-free RL objective that provably lower bounds reverse Kullback–Leibler divergence between occupancy measures of the expert and imitator. We present a practical IL algorithm, Neural Density Imitation (NDI), which obtains state-of-the-art demonstration efficiency on benchmark control tasks.

1 INTRODUCTION

Imitation Learning (IL) algorithms aim to learn optimal behavior by mimicking expert demonstrations. Perhaps the simplest IL method is Behavioral Cloning (BC) (Pomerleau, 1991) which ignores the dynamics of the underlying Markov Decision Process (MDP) that generated the demonstrations, and treats IL as a supervised learning problem of predicting optimal actions given states. Prior work showed that if the learned policy incurs a small BC loss, the worst case performance gap between the expert and imitator grows quadratically with the number of decision steps (Ross & Bagnell, 2010; Ross et al., 2011a). The crux of their argument is that policies that are "close" as measured by BC loss can induce disastrously different distributions over states when deployed in the environment. One family of solutions to mitigating such compounding errors is Interactive IL (Ross et al., 2011b; 2013; Guo et al., 2014), which involves running the imitator's policy and collecting corrective actions from an interactive expert. However, interactive expert queries can be expensive and are seldom available.

Another family of approaches (Ho & Ermon, 2016; Fu et al., 2017; Ke et al., 2020; Kostrikov et al., 2020; Kim & Park, 2018; Wang et al., 2017) that have gained much traction is to directly minimize a statistical distance between state-action distributions induced by policies of the expert and imitator, i.e the occupancy measures ρ_{π_E} and ρ_{π_θ}. As ρ_{π_θ} is an implicit distribution induced by the policy and environment[1], distribution matching with ρ_{π_θ} typically requires likelihood-free methods involving sampling. Sampling from ρ_{π_θ} entails running the imitator policy in the environment, which was not required by BC. While distribution matching IL requires additional access to an environment simulator, it has been shown to drastically improve demonstration efficiency, i.e the number of demonstrations needed to succeed at IL (Ho & Ermon, 2016). A wide suite of distribution matching IL algorithms use adversarial methods to match ρ_{π_θ} and ρ_{π_E}, which requires alternating between reward (discriminator) and policy (generator) updates (Ho & Ermon, 2016; Fu et al., 2017; Ke et al., 2020; Kostrikov et al., 2020; Kim et al., 2019). A key drawback to such Adversarial Imitation Learning (AIL) methods is that they inherit the instability of alternating min-max optimization (Salimans et al., 2016; Miyato et al., 2018) which is generally not guaranteed to converge (Jin et al., 2019). Furthermore, this instability is exacerbated in the IL setting where generator updates involve high-variance policy optimization and leads to sub-optimal demonstration efficiency. To alleviate this instability, (Wang et al., 2019; Brantley et al., 2020; Reddy et al., 2017) have proposed to do RL with fixed heuristic rewards. Wang et al. (2019), for example, uses a heuristic reward that estimates the support of ρ_{π_E} which discourages the imitator from visiting out-of-support states. While having the merit of simplicity, these approaches have no guarantee of recovering the true expert policy.

In this work, we propose a new framework for IL via obtaining a density estimate q_ϕ of the expert's occupancy measure ρ_{π_E} followed by Maximum Occupancy Entropy Reinforcement Learning (MaxOccEntRL) (Lee et al., 2019; Islam et al., 2019). In the MaxOccEntRL step, the density estimate q_ϕ is used as a *fixed reward* for RL and the occupancy entropy $\mathcal{H}(\rho_{\pi_\theta})$ is simultaneously maximized, leading to the objective $\max_\theta \mathbb{E}_{\rho_{\pi_\theta}}[q_\phi(s,a)] + \mathcal{H}(\rho_{\pi_\theta})$. Intuitively, our approach encourages the imitator to visit high density state-action pairs under ρ_{π_E} while maximally exploring the state-action space.

[1] we assume only samples can be taken from the environment dynamics and its density is unknown

There are two main challenges to this approach. First, we require accurate density estimation of ρ_{π_E}, which is particularly challenging when the state-action space is high dimensional and the number of expert demonstrations are limited. Second, in contrast to Maximum Entropy RL (MaxEntRL), MaxOccEntRL requires maximizing the entropy of an implicit density ρ_{π_θ}. We address the former challenge leveraging advances in density estimation (Germain et al., 2015; Du & Mordatch, 2018; Song et al., 2019). For the latter challenge, we derive a non-adversarial model-free RL objective that provably maximizes a lower bound to occupancy entropy. As a byproduct, we also obtain a model-free RL objective that lower bounds reverse Kullback-Lieber (KL) divergence between ρ_{π_θ} and ρ_{π_E}. We evaluate our method, named Neural Density Imitation (NDI), on high-dimensional benchmark robotics tasks, and show that it achieves state-of-the-art demonstration efficiency.

2 Imitation Learning via density estimation

We model an agent's decision making process as a discounted infinite-horizon Markov Decision Process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, P_0, r, \gamma)$. Here $\mathcal{S}, \mathcal{A}$ are state-action spaces, $P : \mathcal{S} \times \mathcal{A} \to \Omega(\mathcal{S})$ is a transition dynamics where $\Omega(\mathcal{S})$ is the set of probability measures on $\mathcal{S}$, $P_0 : \mathcal{S} \to \mathbb{R}$ is an initial state distribution, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is a reward function, and $\gamma \in [0, 1)$ is a discount factor. A parameterized policy $\pi_\theta : \mathcal{S} \to \Omega(\mathcal{A})$ distills the agent's decision making rule and $\{s_t, a_t\}_{t=0}^{\infty}$ is the stochastic process realized by sampling an initial state from $s_0 \sim P_0(s)$ then running π_θ in the environment, i.e $a_t \sim \pi_\theta(\cdot|s_t)$, $s_{t+1} \sim P(\cdot|s_t, a_t)$. We denote by $p_{\theta,t:t+k}$ the joint distribution of states $\{s_t, s_{t+1}, ..., s_{t+k}\}$, where setting $p_{\theta,t}$ recovers the marginal of s_t. The (unnormalized) occupancy measure of π_θ is defined as $\rho_{\pi_\theta}(s, a) = \sum_{t=0}^{\infty} \gamma^t p_{\theta,t}(s)\pi_\theta(a|s)$. Intuitively, $\rho_{\pi_\theta}(s, a)$ quantifies the frequency of visiting the state-action pair (s, a) when running π_θ for a long time, with more emphasis on earlier states.

We denote policy performance as $J(\pi_\theta, \bar{r}) = \mathbb{E}_{\pi_\theta}[\sum_{t=0}^{\infty} \gamma^t \bar{r}(s_t, a_t)] = \mathbb{E}_{(s,a)\sim\rho_{\pi_\theta}}[\bar{r}(s, a)]$ where $\bar{r}$ is a (potentially) augmented reward function and $\mathbb{E}$ denotes the generalized expectation operator extended to non-normalized densities $\hat{p} : \mathcal{X} \to \mathbb{R}^+$ and functions $f : \mathcal{X} \to \mathcal{Y}$ so that $\mathbb{E}_{\hat{p}}[f(x)] = \sum_x \hat{p}(x)f(x)$. The choice of $\bar{r}$ depends on the RL framework. In standard RL, we simply have $\bar{r} = r$, while in Maximum Entropy RL (MaxEntRL) (Haarnoja et al., 2017), we have $\bar{r}(s, a) = r(s, a) - \log \pi_\theta(a|s)$. We denote the entropy of $\rho_{\pi_\theta}(s, a)$ as $\mathcal{H}(\rho_{\pi_\theta}) = \mathbb{E}_{\rho_{\pi_\theta}}[-\log \rho_{\pi_\theta}(s, a)]$ and overload notation to denote the γ-discounted causal entropy of policy π_θ as $\mathcal{H}(\pi_\theta) = \mathbb{E}_{\pi_\theta}[\sum_{t=0}^{\infty} -\gamma^t \log \pi_\theta(a_t|s_t)] = \mathbb{E}_{\rho_{\pi_\theta}}[-\log \pi_\theta(a|s)]$. Note that we use a generalized notion of entropy where the domain is extended to non-normalized densities. We can then define the Maximum Occupancy Entropy RL (MaxOccEntRL) (Lee et al., 2019; Islam et al., 2019) objective as $J(\pi_\theta, \bar{r} = r) + \mathcal{H}(\rho_{\pi_\theta})$. Note the key difference between MaxOccEntRL and MaxEntRL: entropy regularization is on the occupancy measure instead of the policy, i.e seeks state diversity instead of action diversity. We will later show in section 2.2, that a lower bound on this objective reduces to a complete model-free RL objective with an augmented reward $\bar{r}$.

Let π_E, π_θ denote an expert and imitator policy, respectively. Given only demonstrations $\mathcal{D} = \{(s, a)_i\}_{i=1}^{k} \sim \pi_E$ of state-action pairs sampled from the expert, Imitation Learning (IL) aims to learn a policy π_θ which matches the expert, i.e $\pi_\theta = \pi_E$. Formally, IL can be recast as a distribution matching problem (Ho & Ermon, 2016; Ke et al., 2020) between occupancy measures ρ_{π_θ} and ρ_{π_E}:

$$\text{maximize}_\theta \; -d(\rho_{\pi_\theta}, \rho_{\pi_E}) \tag{1}$$

where $d(\hat{p}, \hat{q})$ is a generalized statistical distance defined on the extended domain of (potentially) non-normalized probability densities $\hat{p}(x), \hat{q}(x)$ with the same normalization factor $Z > 0$, i.e $\int_x \hat{p}(x)/Z = \int_x \hat{q}(x)/Z = 1$. For ρ_π and ρ_{π_E}, we have $Z = \frac{1}{1-\gamma}$. As we are only able to take samples from the transition kernel and its density is unknown, ρ_{π_θ} is an implicit distribution[2]. Thus, optimizing Eq. 1 typically requires likelihood-free approaches leveraging samples from ρ_{π_θ}, i.e running π_θ in the environment. Current state-of-the-art IL approaches use likelihood-free adversarial methods to approximately optimize Eq. 1 for various choices of d such as reverse Kullback-Liebler (KL) divergence (Fu et al., 2017; Kostrikov et al., 2020) and Jensen-Shannon (JS) divergence (Ho & Ermon, 2016). However, adversarial methods are known to suffer from optimization instability which is exacerbated in the IL setting where one step in the alternating optimization involves RL.

[2] probability models that have potentially intractable density functions, but can be sampled from to estimate expectations and gradients of expectations with respect to model parameters (Huszár, 2017).

We instead derive a non-adversarial objective for IL. In this work, we choose d to be (generalized) reverse-KL divergence and leave derivations for alternate choices of d to future work.

$$-D_{\mathrm{KL}}(\rho_{\pi_\theta}||\rho_{\pi_E}) = \mathbb{E}_{\rho_{\pi_\theta}}[\log \rho_{\pi_E}(s,a) - \log \rho_{\pi_\theta}(s,a)] = J(\pi_\theta, \bar{r} = \log \rho_{\pi_E}) + \mathcal{H}(\rho_{\pi_\theta}) \quad (2)$$

We see that maximizing negative reverse-KL with respect to π_θ is equivalent to Maximum Occupancy Entropy RL (MaxOccEntRL) with $\log \rho_{\pi_E}$ as the fixed reward. Intuitively, this objective drives π_θ to visit states that are most likely under ρ_{π_E} while maximally spreading out probability mass so that if two state-action pairs are equally likely, the policy visits both. There are two main challenges associated with this approach which we address in the following sections.

1. $\log \rho_{\pi_E}$ is unknown and must be estimated from the demonstrations $\mathcal{D}$. Density estimation remains a challenging problem, especially when there are a limited number of samples and the data is high dimensional (Liu et al., 2007). Note that simply extracting the conditional $\pi(a|s)$ from an estimate of the joint $\rho_{\pi_E}(s,a)$ is an alternate way to do BC and does not resolve the compounding error problem (Ross et al., 2011a).
2. $\mathcal{H}(\rho_{\pi_\theta})$ is hard to maximize as ρ_{π_θ} is an implicit density. This challenge is similar to the difficulty of entropy regularizing generators (Mohamed & Lakshminarayanan, 2016; Belghazi et al., 2018; Dieng et al., 2019) for Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), and most existing approaches (Dieng et al., 2019; Lee et al., 2019) use adversarial optimization.

2.1 Estimating the expert occupancy measure

We seek to learn a parameterized density model $q_\phi(s,a)$ of ρ_{π_E} from samples. We consider two canonical families of density models: Autoregressive models and Energy-based models (EBMs).

Autoregressive Models (Germain et al., 2015; Papamakarios et al., 2017): An autoregressive model $q_\phi(x)$ for $x = (x_1, ..., x_{\dim(\mathcal{S})+\dim(\mathcal{A})}) = (s,a)$ learns a factorized distribution of the form:

$$q_\phi(x) = \Pi_i q_{\phi_i}(x_i|x_{<i}) \quad (3)$$

For instance, each factor q_{ϕ_i} could be a mapping from $x_{<i}$ to a Gaussian density over x_i. When given a prior over the true dependency structure of $\{x_i\}$, this can be incorporated by refactoring Eq. 3. Autoregressive models are typically trained via Maximum Likelihood Estimation (MLE).

$$\max_\phi \mathbb{E}_{\rho_{\pi_E}}[\log q_\phi(s,a)] \quad (4)$$

Energy-based Models (EBM) (Du & Mordatch, 2018; Song et al., 2019): Let $\mathcal{E}_\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ be an energy function. An energy based model is a parameterized Boltzman distribution of the form:

$$q_\phi(s,a) = \frac{1}{Z(\phi)} e^{-\mathcal{E}_\phi(s,a)} \quad (5)$$

where $Z(\phi) = \int_{\mathcal{S}\times\mathcal{A}} e^{-\mathcal{E}_\phi(s,a)}$ denotes the partition function. Energy-based models are desirable for high dimensional density estimation due to their expressivity, but are typically difficult to train due to the intractability of computing the partition function. However, our IL objective in Eq. 1 conveniently only requires a non-normalized density estimate as policy optimality is invariant to constant shifts in the reward. Thus, we opted to perform non-normalized density estimation with EBMs using score matching which allows us to directly learn $\mathcal{E}_\phi$ without having to estimate $Z(\phi)$. Standard score matching (Hyvärinen, 2005) minimizes the following objective.

$$\min_\phi \mathbb{E}_{\rho_{\pi_E}}[\mathrm{tr}(\nabla^2_{s,a}\mathcal{E}_\phi(s,a)) + \frac{1}{2}\|\nabla_{s,a}\mathcal{E}_\phi(s,a)\|_2^2] \quad (6)$$

2.2 Maximum Occupancy Entropy Reinforcement Learning

In general maximizing the entropy of implicit distributions is challenging due to the fact that there is no analytic form for the density function. Prior works have proposed using adversarial methods involving noise injection (Dieng et al., 2019) and fictitious play (Brown, 1951; Lee et al., 2019). We instead propose to maximize a novel lower bound to the occupancy entropy which we prove is equivalent to maximizing a non-adversarial model-free RL objective. We first introduce a crucial ingredient in deriving our occupancy entropy lower bound, which is a tractable lower bound to Mutual Information (MI) first proposed by Nguyen, Wainright, and Jordan (Nguyen et al., 2010), also

known as the f-GAN KL (Nowozin et al., 2016) and MINE-f (Belghazi et al., 2018). For random variables X, Y distributed according to $p_{\theta_{xy}}(x,y), p_{\theta_x}(x), p_{\theta_y}(y)$ where $\theta = (\theta_{xy}, \theta_x, \theta_y)$, and any critic function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, it holds that $I(X;Y|\theta) \geq I^f_{\text{NWJ}}(X;Y|\theta)$ where,

$$I^f_{\text{NWJ}}(X;Y|\theta) := \mathbb{E}_{p_{\theta_{xy}}}[f(x,y)] - e^{-1}\mathbb{E}_{p_{\theta_x}}[\mathbb{E}_{p_{\theta_y}}[e^{f(x,y)}]] \tag{7}$$

This bound is tight when f is chosen to be the optimal critic $f^*(x,y) = \log \frac{p_{\theta_{xy}}(x,y)}{p_{\theta_x}(x)p_{\theta_y}(y)} + 1$. We are now ready to state a lower bound to the occupancy entropy.

Theorem 1 *Let MDP $\mathcal{M}$ satisfy assumption 1 (App. A). For any critic $f : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$, it holds that*

$$\mathcal{H}(\rho_{\pi_\theta}) \geq \mathcal{H}^f(\rho_{\pi_\theta}) \tag{8}$$

where

$$\mathcal{H}^f(\rho_{\pi_\theta}) := \mathcal{H}(s_0) + (1+\gamma)\mathcal{H}(\pi_\theta) + \gamma \sum_{t=0}^{\infty} \gamma^t I^f_{\text{NWJ}}(s_{t+1}; s_t|\theta) \tag{9}$$

See Appendix A.1 for the proof and a discussion of the bound tightness. Here onwards, we refer to $\mathcal{H}^f(\rho_{\pi_\theta})$ from Theorem 1 as the State-Action Entropy Lower Bound (SAELBO). The SAELBO mainly decomposes into policy entropy and mutual information between consecutive states. Intuitively, the policy entropy term $\mathcal{H}(\pi_\theta)$ pushes the policy to be as uniformly random as possible. The MI term $I_{\text{NWJ}}(s_{t+1}; s_t|\theta)$ encourages the agent to seek out states where it is currently behaving least randomly as the next state is more predictable from the current state (i.e shares more information) if the policy is less random. Together these objectives will encourage the agent to continuously seek out states where its policy is currently least random and update the policy to be more random in those states! We note an important distinction between solely maximizing the policy entropy term versus our bound. Consider a video game where the agent must make a sequence of purposeful decisions to unlock new levels, e.g Montezuma's revenge. Assume that your policy is a conditional gaussian with the initial variance set to be small. Soley optimizing policy entropy will yield a uniform random policy which has very low probability of exploring any of the higher levels. On the otherhand, optimizing our bound will encourage the agent to continuously seek out novel states, including new levels, since the policy variance will be small there. Next, we show that the gradient of the SAELBO is equivalent to the gradient of a model-free RL objective.

Theorem 2 *Let $q_\pi(a|s)$ and $\{q_t(s)\}_{t\geq 0}$ be probability densities such that $\forall s, a \in \mathcal{S} \times \mathcal{A}$ satisfy $q_\pi(a|s) = \pi_\theta(a|s)$ and $q_t(s) = p_{\theta,t}(s)$. Then for all $f : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$,*

$$\nabla_\theta \mathcal{H}^f(\rho_{\pi_\theta}) = \nabla_\theta J(\pi_\theta, \bar{r} = r_\pi + r_f) \tag{10}$$

where

$$r_\pi(s_t, a_t) = -(1+\gamma)\log q_\pi(a_t|s_t) \tag{11}$$

$$r_f(s_t, a_t, s_{t+1}) = \gamma f(s_t, s_{t+1}) - \frac{\gamma}{e}\mathbb{E}_{\tilde{s}_t \sim q_t, \tilde{s}_{t+1} \sim q_{t+1}}[e^{f(\tilde{s}_t, s_{t+1})} + e^{f(s_t, \tilde{s}_{t+1})}] \tag{12}$$

See Appendix A.2 for the proof. Theorem 2 shows that maximizing the SAELBO is equivalent to maximizing a discounted model-free RL objective with the reward $r_\pi + r_f$, where r_π contributes to maximizing $\mathcal{H}(\pi_\theta)$ and r_f contributes to maximizing $\sum_{t=0}^{\infty} \gamma^t I^f_{\text{NWJ}}(s_{t+1}; s_t|\theta)$. Note that evaluating r_f entails estimating expectations with respect to q_t, q_{t+1}. This can be accomplished by rolling out multiple trajectories with the current policy and collecting the states from time-step $t, t+1$. Alternatively, if we assume that the policy is changing slowly, we can simply take samples of states from time-step $t, t+1$ from the replay buffer. Combining the results of Theorem 1, 2, we end the section with a lower bound on the original distribution matching objective from Eq. 1 and show that maximizing this lower bound is again, equivalent to maximizing a model-free RL objective.

Corollary 1 *Let MDP $\mathcal{M}$ satisfy assumption 1 (App. A). For any critic $f : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$, it holds that*

$$-D_{\text{KL}}(\rho_{\pi_\theta} || \rho_{\pi_E}) \geq J(\pi_\theta, \bar{r} = \log \rho_{\pi_E}) + \mathcal{H}^f(\rho_{\pi_\theta}) \tag{13}$$

Furthermore, let r_π, r_f be defined as in Theorem 2. Then,

$$\nabla_\theta\big(J(\pi_\theta, \bar{r} = \log \rho_{\pi_E}) + \mathcal{H}^f(\rho_{\pi_\theta})\big) = \nabla_\theta J(\pi_\theta, \bar{r} = \log \rho_{\pi_E} + r_\pi + r_f) \tag{14}$$

In the following section we derive a practical distribution matching IL algorithm combining all the ingredients from this section.

Algorithm 1: Neural Density Imitation (NDI)

Require: Demonstrations $\mathcal{D} \sim \pi_E$, Reward weights λ_π, λ_f, Learning rates η_θ, η_ϕ, Critic f

Phase 1. *Density estimation*:

Learn $q_\phi(s, a)$ from $\mathcal{D}$ using MADE or EBMs by optimizing Eq. 4 or 15 with learning rate η_ϕ.

Phase 2. *MaxOccEntRL*:

for $k = 1, 2, ...$ **do**

Collect $(s_t, a_t, s_{t+1}, \bar{r}) \sim \pi_\theta$ and add to replay buffer $\mathcal{B}$, where $\bar{r} = \log q_\phi + \lambda_\pi r_\pi + \lambda_f r_f$,

$$r_\pi(s_t, a_t) = -\log \pi_\theta(a_t|s_t)$$

$$r_f(s_t, a_t, s_{t+1}) = f(s_t, s_{t+1}) - \frac{\gamma}{e}\mathbb{E}_{\tilde{s}_t \sim \mathcal{B}_t, \tilde{s}_{t+1} \sim \mathcal{B}_{t+1}}[e^{f(s_{t+1}, \tilde{s}_t)} + e^{f(\tilde{s}_{t+1}, s_t)}]$$

Update π_θ using Soft Actor-Critic (SAC) (Haarnoja et al., 2018):

$$\theta_{k+1} \leftarrow \theta_k + \eta_\theta \nabla_\theta J(\theta, \bar{r})|_{\theta=\theta_k}$$

end

3 Neural Density Imitation (NDI)

From previous section's results, we propose Neural Density Imitation (NDI) that works in two phases:

Phase 1: Density estimation: As described in Section 2, we seek to leverage Autoregressive models and EBMs for density estimation of the expert's occupancy measure ρ_{π_E} from samples. As in (Ho & Ermon, 2016; Fu et al., 2017), we take the state-action pairs in the demonstration set $\mathcal{D} = \{(s, a)_i\}_{i=1}^N \sim \pi_E$ to approximate samples from ρ_{π_E} and fit q_ϕ on $\mathcal{D}$. For Autoregressive models, we use Masked Autoencoders for Density Estimation (MADE) (Germain et al., 2015) where the entire collection of conditional density models $\{q_{\phi_i}\}$ is parameterized by a single masked autoencoder network. Specifically, we use a gaussian mixture variant (Papamakarios et al., 2017) of MADE where each of the conditionals q_{ϕ_i} map inputs $x_{<i}$ to the mean and covariance of a gaussian mixture distribution over x_i. The autoencoder is trained via MLE as in Eq. 4. With EBMs, we perform non-normalized log density estimation and thus directly parameterize the energy function $\mathcal{E}_\phi$ with neural networks since $\log q_\phi = \mathcal{E}_\phi + \log Z(\phi)$. One hurdle to learning $\mathcal{E}_\phi$ with the standard score matching objective of Eq. 6 is the computational expense of estimating $\mathrm{tr}(\nabla^2_{s,a}\mathcal{E}_\phi(s, a))$. To reduce this expense, we use Sliced Score Matching (Song et al., 2019) which leverages random projections $v \sim p_v$ to approximate the trace term as in the following equation.

$$\min_\phi \mathbb{E}_{p_v}[\mathbb{E}_{\rho_{\pi_E}}[v^T \nabla^2_{s,a}\mathcal{E}_\phi(s, a)v + \frac{1}{2}\|\mathcal{E}_\phi(s, a)\|_2^2]] \tag{15}$$

Phase 2: MaxOccEntRL After we've acquired a log density estimate $\log q_\phi$ from the previous phase, we perform RL with entropy regularization on the occupancy measure. Inspired by Corollary 1, we propose the following RL objective

$$\max_\theta J(\pi_\theta, \bar{r} = \log q_\phi + \lambda_\pi r_\pi + \lambda_f r_f) \tag{16}$$

where $\lambda_\pi, \lambda_f > 0$ are weights introduced to control the influence of the occupancy entropy regularization. In practice, Eq. 16 can be maximized using any RL algorithm by simply setting the reward function to be $\bar{r}$ from Eq. 16. In this work, we use Soft Actor-Critic (SAC) (Haarnoja et al., 2018). For our critic f, we *fix it* to be a normalized RBF kernel for simplicity, but future works could explore learning the critic to match the optimal critic:

$$f(s_{t+1}, s_t) = \log \frac{e^{-\|s_{t+1} - s_t\|_2^2}}{\mathbb{E}_{q_t, q_{t+1}}[e^{-\|s_{t+1} - s_t\|_2^2}]} + 1 \tag{17}$$

While simple, our choice of f emulates two important properties of the optimal critic $f^*(x, y) = \log \frac{p(x|y)}{p(x)} + 1$: (1). it follows the same "form" of a log-density ratio plus a constant (2). consecutively sampled states from the joint, i.e $s_t, s_{t+1} \sim p_{\theta, t:t+1}$ have high value under our f since they are likely to be close to each other under smooth dynamics, while samples from the marginals $s_t, s_{t+1} \sim q_t, q_{t+1}$ are likely to have lower value under f since they can be arbitrarily different states. To

estimate the expectations with respect to q_t, q_{t+1} in Eq. 12, we simply take samples of previously visited states at time $t, t+1$ from the replay buffer. Note that even if f is learned to be the optimal critic, NDI is still non-adversarial since both π and f are updated in the same direction of maximizing the objective, unlike GAIL which performs min-max optimization. (see Appendix A.3)

4 Related Works

Prior literature on Imitation learning (IL) in the absence of an interactive expert revolves around Behavioral Cloning (BC) (Pomerleau, 1991; Wu et al., 2019) and distribution matching IL (Ho & Ermon, 2016; Fu et al., 2017; Song et al., 2018; Ke et al., 2020; Kostrikov et al., 2020; Kim et al., 2019). Many approaches in the latter category minimize statistical divergences using adversarial methods to solve a min-max optimization problem, alternating between reward (discriminator) and policy (generator) updates. ValueDICE, a more recently proposed adversarial IL approach, formulates reverse KL divergence into a completely off-policy objective thereby greatly reducing the number of environment interactions. A key issue with such Adversarial Imitation Learning (AIL) approaches is optimization instability (Miyato et al., 2018; Jin et al., 2019). Recent works have sought to avoid adversarial optimization by instead performing RL with a heuristic reward function that estimates the support of the expert occupancy measure. Random Expert Distillation (RED) (Wang et al., 2019) and Disagreement-regularized IL (Brantley et al., 2020) are two representative approaches in this family. A key limitation of these approaches is that support estimation is insufficient to recover the expert policy and thus they require an additional behavioral cloning step. Unlike AIL, we maximize a non-adversarial RL objective and unlike heuristic reward approaches, our objective provably lower bounds reverse KL between occupancy measures of the expert and imitator. Density estimation with deep neural networks is an active research area, and much progress has been made towards modeling high-dimensional structured data like images and audio. Most successful approaches parameterize a normalized probability model and estimate it with maximum likelihood, e.g., autoregressive models (Uria et al., 2013; 2016; Germain et al., 2015; van den Oord et al., 2016) and normalizing flow models (Dinh et al., 2014; 2016; Kingma & Dhariwal, 2018). Some other methods explore estimating non-normalized probability models with MCMC (Du & Mordatch, 2019; Yu et al., 2020) or training with alternative statistical divergences such as score matching (Hyvärinen, 2005; Song et al., 2019; Song & Ermon, 2019) and noise contrastive estimation (Gutmann & Hyvärinen, 2010; Gao et al., 2019). Related to MaxOccEntRL, recent works (Lee et al., 2019; Elad Hazan, 2019; Islam et al., 2019) on exploration in RL have investigated state-marginal occupancy entropy maximization. To do so, (Elad Hazan, 2019) requires access to a robust planning oracle, while (Lee et al., 2019) uses fictitious play, an alternative adversarial algorithm that is guaranteed to converge. Unlike these works, our approach maximizes the SAELBO which requires no planning oracle nor min-max optimization, and is trivial to implement with existing RL algorithms.

5 Experiments

Environment: Following prior work, we run experiments on benchmark Mujoco (Todorov et al., 2012; Brockman et al., 2016) tasks: Hopper (11, 3), HalfCheetah (17, 6), Walker (17, 6), Ant (111, 8), and Humanoid (376, 17), where the (observation, action) dimensions are noted parentheses.

Pipeline: We train expert policies using SAC (Haarnoja et al., 2018). All of our results are averaged across five random seeds where for each seed we randomly sample a trajectory from an expert, perform density estimation, and then MaxOccEntRL. For each seed we save the best imitator as measured by our augmented reward $\bar{r}$ from Eq. 16 and report its performance with respect to the ground truth reward. We don't perform sparse subsampling on the data as in (Ho & Ermon, 2016) since real world demonstration data typically aren't subsampled to such an extent and using full trajectories was sufficient to compare performance.

Architecture: We experiment with two variants of our method, NDI+MADE and NDI+EBM, where the only difference lies in the density model. Across all experiments, our density model q_ϕ is a two-layer MLP with 256 hidden units. For hyperparameters related to the MaxOccEntRL step, λ_π is tuned automatically in the stable-baselines implementation (Hill et al., 2018), and we set $\lambda_f = 0.005$ (see Section 5.3 for ablation studies). For full details on architecture see Appendix B.

Baselines: We compare our method against the following baselines: (1). *Behavioral Cloning (BC)* (Pomerleau, 1991): learns a policy via direct supervised learning on $\mathcal{D}$. (2). *Random Expert Distillation (RED)* (Wang et al., 2019): estimates the support of the expert policy using a predictor

Table 1: **Task Performance** when provided with *one* demonstration. NDI (orange rows) outperforms all baselines on all tasks. See Appendix C.1 for results with varying demonstrations.

	HOPPER	HALF-CHEETAH	WALKER	ANT	HUMANOID
RANDOM	14 ± 8	282 ± 80	1 ± 5	70 ± 111	123 ± 35
BC	1432 ± 382	2674 ± 633	1691 ± 1008	1425 ± 812	353 ± 171
RED	448 ± 516	383 ± 819	309 ± 193	910 ± 175	242 ± 67
GAIL	3261 ± 533	3017 ± 531	3957 ± 253	2299 ± 519	204 ± 67
VALUEDICE	2749 ± 571	3456 ± 401	3342 ± 1514	1016 ± 313	364 ± 50
NDI+MADE	3288 ± 94	4119 ± 71	4518 ± 127	555 ± 311	$\mathbf{6088 \pm 689}$
NDI+EBM	$\mathbf{3458 \pm 210}$	$\mathbf{4511 \pm 569}$	$\mathbf{5061 \pm 135}$	$\mathbf{4293 \pm 431}$	5305 ± 555
EXPERT	3567 ± 4	4142 ± 132	5006 ± 472	4362 ± 827	5417 ± 2286

and target network (Burda et al., 2018), followed by RL using this heuristic reward. (3). *Generative Adversarial Imitation Learning (GAIL)* (Ho & Ermon, 2016): on-policy adversarial IL method which alternates reward and policy updates. (4). *ValueDICE* (Kostrikov et al., 2020): current state-of-the-art adversarial IL method that works off-policy. See Appendix B for baseline implementation details.

5.1 TASK PERFORMANCE

Table 1 compares the ground truth reward acquired by agents trained with various IL algorithms when *one* demonstration is provided by the expert. (See Appendix C.1 for performance comparisons with varying demonstrations) *NDI+EBM achieves expert level performance on all mujoco benchmarks when provided one demonstration and outperforms all baselines on all mujoco benchmarks*. NDI+MADE achieves expert level performance on 4/5 tasks but fails on Ant. We found spurious modes in the density learned by MADE for Ant, and the RL algorithm was converging to these local maxima. We found that baselines are commonly unable to solve Humanoid with one demonstration (the most difficult task considered). RED is unable to perform well on all tasks without pretraining with BC as done in (Wang et al., 2019). For fair comparisons with methods that do not use pretraining, we also do not use pretraining for RED. See Appendix C.2 for results with a BC pretraining step added to all algorithms. GAIL and ValueDICE perform comparably with each other, both outperforming behavioral cloning. We note that these results are somewhat unsurprising given that ValueDICE (Kostrikov et al., 2020) did not claim to improve demonstration efficiency over GAIL (Ho & Ermon, 2016), but rather focused on reducing the number of environment interactions. Both methods notably under-perform the expert on Ant-v3 and Humanoid-v3 which have the largest state-action spaces. Although minimizing the number of environment interactions was not a targeted goal of this work, we found that NDI roughly requires an order of magnitude less environment interactions than GAIL. Please see Appendix C.3 for full environment sample complexity comparisons.

5.2 DENSITY EVALUATION

In this section, we examine the learned density model q_ϕ for NDI+EBM and show that it highly correlates with the true mujoco rewards which are linear functions of forward velocity. We randomly sample test states s and multiple test actions a_s per test state, both from a uniform distribution with boundaries at the minimum/maximum state-action values in the demonstration set. We then visualize the log marginal $\log q_\phi(s) = \log \sum_{a_s} q_\phi(s, a_s)$ projected on to two state dimensions: one corresponding to the forward velocity of the robot and the other a random selection, e.g the knee joint angle. Each point in Figure 1 corresponds to a projection of a sampled test state s, and the colors scale with the value of $\log q_\phi(s)$. For all environments besides Humanoid, we found that the density estimate positively correlates with velocity even on uniformly drawn state-actions which were not contained in the demonstrations. We found similar correlations for Humanoid on states in the demonstration set. Intuitively, a good density estimate should indeed have such correlations, since the true expert occupancy measure should positively correlate with forward velocity due to the expert attempting to consistently maintain high velocity.

5.3 ABLATION STUDIES

Table 2 shows the effect of the varying λ_f on task (reward) and imitation performance (KL), i.e similarities between π, π_E measured as $\mathbb{E}_{s\sim\pi}[D_{\mathrm{KL}}(\pi(\cdot|s)||\pi_E(\cdot|s))]$. Setting λ_f too large (≥ 0.1)

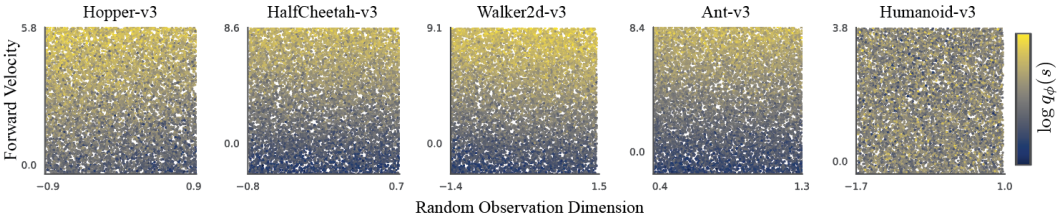

Figure 1: **Learned density visualization**. We randomly sample test states s and multiple test actions a_s per test state, both from a uniform distribution, then visualize the log marginal $\log q_\phi(s) = \log \sum_{a_s} q_\phi(s, a_s)$ projected onto two state dimensions: one corresponding to forward velocity and the other a random selection. Much like true reward function in Mujoco environments, we found that the log marginal positively correlates with forward velocity on 4/5 tasks.

Table 2: **Effect of varying MI reward weight** λ_f on (1). Task performance of NDI-EBM (top row) and (2). Imitation performance of NDI-EBM (bottom row) measured as the average KL divergence between π, π_E on states s sampled by running π in the true environment, i.e $\mathbb{E}_{s\sim\pi}[D_{\text{KL}}(\pi(\cdot|s)||\pi_E(\cdot|s))]$, normalized by the average D_{KL} between the random and expert policies. $D_{\text{KL}}(\pi||\pi_E)$ can be computed analytically since π, π_E are conditional gaussians. Setting λ_f too large hurts task performance while setting it too small is suboptimal for matching the expert occupancy. A middle point of $\lambda = 0.005$ achieves a balance between the two metrics.

		HOPPER	HALF-CHEETAH	WALKER	ANT	HUMANOID
$\lambda_f = 0.0001$	REWARD	$\mathbf{3506 \pm 188}$	$\mathbf{5697 \pm 805}$	$\mathbf{5171 \pm 157}$	4158 ± 523	$\mathbf{5752 \pm 632}$
	KL	0.15 ± 0.05	0.32 ± 0.15	0.25 ± 0.04	0.51 ± 0.05	0.41 ± 1.82
$\lambda_f = 0.005$	REWARD	3458 ± 210	4511 ± 569	5061 ± 135	$\mathbf{4293 \pm 431}$	5305 ± 555
	KL	$\mathbf{0.11 \pm 0.02}$	$\mathbf{0.17 \pm 0.09}$	$\mathbf{0.22 \pm 0.14}$	$\mathbf{0.32 \pm 0.12}$	$\mathbf{0.12 \pm 0.14}$
$\lambda_f = 0.1$	REWARD	1057 ± 29	103 ± 59	2710 ± 501	-1021 ± 21	142 ± 50
	KL	0.78 ± 0.13	1.41 ± 0.51	0.41 ± 0.11	2.41 ± 1.41	0.89 ± 0.21
EXPERT	REWARD	3567 ± 4	4142 ± 132	5006 ± 472	4362 ± 827	5417 ± 2286

hurts both task and imitation performance as the MI reward r_f dominates the RL objective leading to excessive state-action entropy maximization. Setting it too small (≤ 0.0001), i.e only maximizing policy entropy $\mathcal{H}(\pi_\theta)$, turns out to benefit task performance, sometimes enabling the imitator to outperform the expert by concentrating most of it's trajectory probability mass to the mode of the expert's trajectory distribution. However, the boosted task performance comes at the cost of suboptimal imitation performance, e.g imitator cheetah running faster than the expert. We found that a middle point of $\lambda_f = 0.005$ simultaneously achieves expert level task performance and good imitation performance. This phenomenon is analogous to how controlling the strength of entropy regularization on the generator can lead to either mode-seeking vs mode covering behavior in GANs. In summary, these results show that state-action entropy $\mathcal{H}^f$ maximization improves distribution matching between π, π_E over policy entropy $\mathcal{H}(\pi_\theta)$ maximization, but distribution matching may not be ideal for task performance maximization, e.g in apprenticeship learning settings.

6 DISCUSSION AND OUTLOOK

This work's main contribution is a new principled framework for IL and an algorithm that obtains state-of-the-art demonstration efficiency. One future direction is to apply NDI to harder visual IL tasks for which AIL is known perform poorly. While the focus of this work is to improve on demonstration efficiency, another important IL performance metric is environment sample complexity. Future works could explore combining off-policy RL or model-based RL with NDI to improve on this end. Finally, there is a rich space of questions to answer regarding the effectiveness of the SAELBO reward r_f. We posit that, for example, in video game environments r_f may be crucial for success since the expert spends most of its time in levels that are extremely hard to reach with just action entropy maximization. Furthermore, one could improve on the tightness of SAELBO by incorporating negative samples (Van Den Oord et al., 2018) and learning the critic function f so that it is close to the optimal critic.

REFERENCES

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. Mine: Mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

Kiante Brantley, Wen Sun, and Mikael Henaff. Disagreement-regularized imitation learning. 2020.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Zaremba Wojciech. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

George Brown. Iterative solution of games by fictitious play. *Activity Analysis of Production and Allocation*, 1951.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

Adji Dieng, Francisco Ruiz, David M. Blei, and Michalis K. Titsias. Prescribed generative adversarial networks. *arXiv preprint arXiv:1910.04302*, 2019.

L Dinh, D Krueger, and Y Bengio. NICE: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. *arXiv preprint arXiv:1605.08803*, May 2016.

Yilun Du and Igor Mordatch. Implicit generation and generalization in energy-based models. *arXiv preprint arXiv:1903.08689*, 2018.

Yilun Du and Igor Mordatch. Implicit generation and generalization in energy-based models. *arXiv preprint arXiv:1903.08689*, 2019.

Karan Singh Abby Van Soest Elad Hazan, Sham M. Kakade. Provably efficient maximum entropy exploration. *arXiv preprint arXiv:1812.02690*, 2019.

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

Ruiqi Gao, Erik Nijkamp, Diederik P Kingma, Zhen Xu, Andrew M Dai, and Ying Nian Wu. Flow contrastive estimation of energy-based models. *arXiv preprint arXiv:1912.00589*, 2019.

Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. Made: Masked autoencoder for distribution estimation. *arXiv preprint arXiv:1502.03509*, 2015.

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*, pp. 2672–2680, 2014.

Xiaoxiao Guo, Satinder Singh, Honglak Lee, Richard L. Lewis, and Xiaoshi Wang. Deep learning for real-time atari game play using offline monte-carlo tree search planning. *Advances in Neural Information Processing Systems 27 (NIPS 2014)*, 2014.

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 297–304, 2010.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. *arXiv preprint arXiv:1702.08165*, 2017.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft Actor-Critic: Off-Policy maximum entropy deep reinforcement learning with a stochastic actor. *arXiv preprint arXiv:1801.01290*, January 2018.

Ashley Hill, Antonin Raffin, Maximilian Ernestus, Adam Gleave, Anssi Kanervisto, Rene Traore, Prafulla Dhariwal, Christopher Hesse, Oleg Klimov, Alex Nichol, Matthias Plappert, Alec Radford, John Schulman, Szymon Sidor, and Yuhuai Wu. Stable baselines. https://github.com/hill-a/stable-baselines, 2018.

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

Ferenc Huszár. Variational inference using implicit distributions. *arXiv preprint arXiv:1702.08235*, 2017.

Aapo Hyvärinen. Estimation of Non-Normalized statistical models by score matching. *Journal of machine learning research: JMLR*, 6(Apr):695–709, 2005. ISSN 1532-4435, 1533-7928.

Riashat Islam, Raihan Seraj, Pierre-Luc Bacon, and Doina Precup. Entropy regularization with discounted future state distribution in policy gradient methods. *arXiv preprint arXiv:1912.05104*, 2019.

Chi Jin, Praneeth Netrapalli, and Michael I. Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? *arXiv preprint arXiv:1902.00618*, 2019.

Liyiming Ke, Matt Barnes, Wen Sun, Gilwoo Lee, Sanjiban Choudhury, and Siddhartha Srinivasa. Imitation learning as f-divergence minimization. *arXiv preprint arXiv:1905.12888*, 2020.

Kee-Eung Kim and Hyun Soo Park. Imitation learning via kernel mean embedding. *AAAI*, 2018.

Kuno Kim, Yihong Gu, Jiaming Song, Shengjia Zhao, and Stefano Ermon. Domain adaptive imitation learning. *arXiv preprint arXiv:1910.00105*, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, December 2014.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems*, pp. 10215–10224, 2018.

Ilya Kostrikov, Ofir Nachum, and Jonathan Tompson. Imitation learning via off-policy distribution matching. 2020.

Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric Xing, Sergey Levine, and Ruslan Salakhutdinov. Efficient exploration via state marginal matching. *arXiv preprint arXiv:1906.05274*, 2019.

Han Liu, John Lafferty, and Larry Wasserman. Sparse nonparametric density estimation in high dimensions using the rodeo. volume 2 of *Proceedings of Machine Learning Research*, pp. 283–290, San Juan, Puerto Rico, 21–24 Mar 2007. PMLR.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

Shakir Mohamed and Balaji Lakshminarayanan. Learning in implicit generative models. *arXiv preprint arXiv:1610.03483*, 2016.

XuanLong Nguyen, Martin J. Wainwright, and Michael I. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *arXiv preprint arXiv:0809.0853*, 2010.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training generative neural samplers using variational divergence minimization. *arXiv preprint arXiv:1606.00709*, June 2016.

George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. *arXiv preprint arXiv:1705.07057*, 2017.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. *arXiv preprint arXiv:1705.05363*, 2017.

Dean A Pomerleau. Efficient training of artificial neural networks for autonomous navigation. *Neural computation*, 3(1):88–97, 1991. ISSN 0899-7667.

Siddharth Reddy, Anca D. Dragan, and Sergey Levine. Sqil: Imitation learning via reinforcement learning with sparse rewards. *arXiv preprint arXiv:1905.11108*, 2017.

Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 661–668, 2010.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. 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*, pp. 627–635, 2011a.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. 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*, pp. 627–635, 2011b.

Stephane Ross, Narek Melik-Barkhudarov, Kumar Shaurya Shankar, Andreas Wendel, Debadeepta Dey, Andrew J. Bagnell, and Martial Hebert. Learning monocular reactive uav control in cluttered natural environments. *International Conference on Robotics and Automation (ICRA)*, 2013.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *arXiv:1606.03498*, 2016.

Jiaming Song, Hongyu Ren, Dorsa Sadigh, and Stefano Ermon. Multi-agent generative adversarial imitation learning. 2018.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *arXiv preprint arXiv:1907.05600*, 2019.

Yang Song, Sahaj Garg, Jiaxin Shi, and Stefano Ermon. Sliced score matching: A scalable approach to density and score estimation. *arXiv preprint arXiv:1905.07088*, 2019.

E Todorov, T Erez, and Y Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, October 2012. doi: 10.1109/IROS.2012.6386109.

Emanuel Todorov. Convex and analytically-invertible dynamics with contacts and constraints: Theory and implementation in mujoco. *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

Benigno Uria, Iain Murray, and Hugo Larochelle. RNADE: The real-valued neural autoregressive density-estimator. In C J C Burges, L Bottou, M Welling, Z Ghahramani, and K Q Weinberger (eds.), *Advances in Neural Information Processing Systems 26*, pp. 2175–2183. Curran Associates, Inc., 2013.

Benigno Uria, Marc-Alexandre Côté, Karol Gregor, Iain Murray, and Hugo Larochelle. Neural autoregressive distribution estimation. *The Journal of Machine Learning Research*, 17(1):7184–7220, 2016.

Aaron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759*, January 2016.

Aaron Van Den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Ruohan Wang, Carlo Ciliberto, Pierluigi Amadori, and Yiannis Demiris. Random expert distillation: Imitation learning via expert policy support estimation. 2019.

Ziyu Wang, Josh Merel, Scott Reed, Greg Wayne, Nando de Freitas, and Nicolas Heess. Robust imitation of diverse behaviors. *arXiv preprint arXiv:1707.02747*, 2017.

Alan Wu, AJ Piergiovanni, and Michael S. Ryoo. Model-based behavioral cloning with future image similarity learning. *arXiv preprint arXiv:1910.03157*, 2019.

Lantao Yu, Yang Song, Jiaming Song, and Stefano Ermon. Training deep energy-based models with f-divergence minimization. *arXiv preprint arXiv:2003.03463*, 2020.