Of Moments and Matching:
A Game-Theoretic Framework for Closing the Imitation Gap

Gokul Swamy [1] **Sanjiban Choudhury** [2] **J. Andrew Bagnell** [1 2] **Zhiwei Steven Wu** [3]

Abstract

We provide a unifying view of a large family of previous imitation learning algorithms through the lens of *moment matching*. At its core, our classification scheme is based on whether the learner attempts to match (1) *reward* or (2) *action-value* moments of the expert's behavior, with each option leading to differing algorithmic approaches. By considering adversarially chosen divergences between learner and expert behavior, we are able to derive bounds on policy performance that apply for all algorithms in each of these classes, the first to our knowledge. We also introduce the notion of *moment recoverability*, implicit in many previous analyses of imitation learning, which allows us to cleanly delineate how well each algorithmic family is able to mitigate compounding errors. We derive three novel algorithm templates (`AdVIL`, `AdRIL`, and `DAeQuIL`) with strong guarantees, simple implementation, and competitive empirical performance.

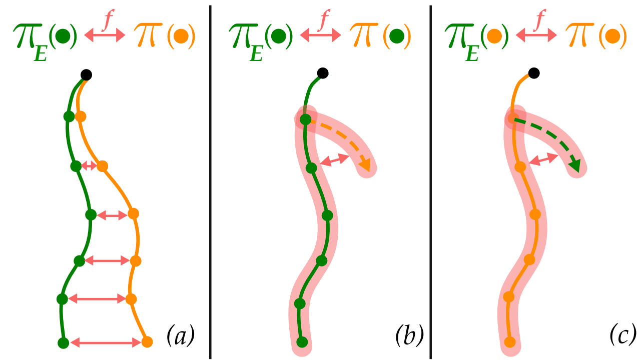

Figure 1. We consider three classes of imitation learning algorithms. **(a)** On-policy reward moment-matching algorithms require access to the environment to generate learner trajectories. **(b)** Off-policy Q-value moment-matching algorithms run completely offline but can produce policies with quadratically compounding errors. **(c)** On-policy Q-value moment-matching algorithms require access to the environment and a queryable expert but can produce strong policies in *recoverable* MDPs.

1. Introduction

When formulated as a statistical learning problem, imitation learning is fundamentally concerned with finding policies that minimize some notion of divergence between learner behavior and that of an expert demonstrator. Existing work has explored various types of divergences including KL (Pomerleau 1989, Bojarski et al. 2016), Jensen-Shannon (Rhinehart et al. 2018), Reverse KL (Kostrikov et al. 2019), f (Ke et al. 2019), and Wasserstein (Dadashi et al. 2020).

At heart, though, we care about the performance of the learned policy under an objective function that is not known to the learner. As argued by (Abbeel and Ng 2004) and (Ziebart et al. 2008), this goal is most cleanly formulated as a problem of *moment matching*, or, equivalently, optimizing Integral Probability Metrics (Sun et al. 2019) (IPMs). This is because a learner that in expectation matches the expert on all the basis functions of a class that includes the expert's objective function, or *matches moments*, must achieve the same performance and will thus be indistinguishable in terms of quality. Additionally, in sharp contrast to recently proposed approaches (Ke et al. 2019, Kostrikov et al. 2019, Jarrett et al. 2020, Rhinehart et al. 2018), moments, due to their simple forms as expectations of basis functions, can be effectively estimated via demonstrator samples and the uncertainty in these estimates can often be quantified to regularize the matching objective (Dudik et al. 2004). In short: these moment matching procedures are simple, effective, and provide the strongest policy performance guarantees we are aware of for imitation learning.

As illustrated in Fig. 1, there are three classes of moments a learner can focus on matching: (a) *on-policy reward moments*, (b) *off-policy Q-value moments*, and (c) *on-policy Q-value moments*, each of which have different requirements on the environment and on the expert. We abbreviate them as **reward**, **off-Q**, and **on-Q** moments, respectively.

[1]Robotics Institute, Carnegie Mellon University [2]Aurora Innovation [3]Institute for Software Research, Carnegie Mellon University. Correspondence to: Gokul Swamy <gswamy@cmu.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

Our key insight is that reward moments have more *discriminative power* because they can pick up on differences in induced state visitation distributions rather than just action conditionals. Thus, *reward moment matching is a harder problem with stronger guarantees than off-Q and on-Q moment matching*.

Our work makes the following three contributions:

1. We present a unifying framework for moment matching in imitation learning. Our framework captures a wide range of prior approaches and allows us to construct, to our knowledge, the first formal lower bounds demonstrating that the choice between matching "reward", "off-Q", or "on-Q" moments is fundamental to the problem of imitation learning rather than an artifact of a particular algorithm or analysis.

2. We clarify the dependence of imitation learning bounds on problem structure. We introduce a joint property of an expert policy and moment class, *moment recoverability*, that helps us characterize the problems for which compounding errors are likely to occur, regardless of the kind of feedback the learner is exposed to.

3. We provide three novel algorithms with strong performance guarantees. We derive idealized algorithms that match each class of moments. We also provide practical instantiations of these ideas, `AdVIL`, `AdRIL`, and `DAeQuIL`. These algorithms have significantly different practical performance as well as theoretical guarantees in terms of compounding of errors over time steps of the problem.

2. Related Work

Imitation Learning. Imitation learning has been shown to be an effective method of solving a variety of problems, from getting cars to drive themselves (Pomerleau 1989), to achieving superhuman performance in games like Go (Silver et al. 2016, Sun et al. 2018), to sample-efficient learning of control policies for high DoF robots (Levine and Koltun 2013), to allowing human operators to effectively supervise and teach robot fleets (Swamy et al. 2020). The issue of compounding errors in imitation learning was first formalized by (Ross et al. 2011), with the authors proving that an interactive expert that can suggest actions in states generated via learner policy rollouts will be able to teach the learner to recover from mistakes.

Adversarial Imitation Learning. Starting with the seminal work of (Ho and Ermon 2016), numerous proposed approaches have framed imitation learning as a game between a learner's policy and another network that attempts to discriminate between learner rollouts and expert demonstrations (Fu et al. 2018, Song et al. 2018). We build upon this work by elucidating the properties that result from the kind of *feedback* the learner is exposed to – whether they are able to see the consequences of their own actions via rollouts or if they are only able to propose actions in states from expert trajectories. Our proposed approaches also have stronger guarantees and less brittle performance than the popular GAIL (Ho and Ermon 2016).

Mathematical Tools. Our algorithmic approach combines two tools that have enjoyed success in imitation learning: *functional gradients* (Ratliff et al. 2009) and the *Integral Probability Metric* (Sun et al. 2019). We define two algorithms, `AdRIL` and `AdVIL` that are based on optimizing the value-directed IPM, with `AdRIL` having the discriminative player perform updates via functional gradient descent. The IPM is linear in the discriminative function, unlike other proposed metrics like the Donsker-Varadhan bound on KL divergence. Specifically, the Donsker-Varadhan bound includes an expectation of the exponentiated discriminative function, which makes estimation difficult with a few samples (McAllester and Stratos 2020). Our analysis makes repeated use of the *Performance Difference Lemma* (Kakade and Langford 2002, Bagnell et al. 2003) or PDL, which allows us to bound the suboptimality of the learner's policy.

Our proposed algorithms bear some resemblance to previously proposed methods, with `AdRIL` resembling SQIL (Reddy et al. 2019) and `AdVIL` resembling ValueDICE (Kostrikov et al. 2019). We note that `AdVIL`, while cleanly derived from the PDL, can also be derived from an IPM by using a telescoping substitution similar to the ValueDICE derivation. Notably, because `AdVIL` is linear in the discriminator, it does not suffer from ValueDICE's difficulty in estimating the expectation of an exponential. This difficulty might help explain why ValueDICE can underperform the behavioral cloning baseline on several benchmark tasks (Jarrett et al. 2020). Similarly, `AdRIL` can avoid the sharp degradation in policy performance that SQIL demonstrates (Barde et al. 2020). This is because SQIL hard-codes the discriminator while `AdRIL` adaptively updates the discriminator to account for changes in the policy's trajectory distribution. `DAeQuIL` can be seen as the natural extension of DAgger (Ross et al. 2011) to the adversarial loss setting.

3. Moment Matching Imitation Learning

We begin by formalizing our setup and objective.

3.1. Problem Definition

Let $\Delta(\mathcal{X})$ denote the space of all probability distribution over a set $\mathcal{X}$. Consider an MDP parameterized by $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, r, T, P_0 \rangle$[1], where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition operator, $r : \mathcal{S} \times \mathcal{A} \to [-1, 1]$ is a reward function, T is the hori-

[1] We ignore the discount factor for simplicity.

zon, and P_0 is the initial state distribution. Let π denote the learner's policy and let π_E denote the demonstrator's policy. A trajectory $\tau \sim \pi = \{s_t, a_t\}_{t=1 \dots T}$ refers to a sequence of state-action pairs generated by first sampling a state s_1 from P_0 and then repeatedly sampling actions a_t and next states s_{t+1} from π and $\mathcal{T}$ for $T-1$ timesteps. We also define our value and Q-value functions as $V_t^\pi(s) = \mathbb{E}_{\tau \sim \pi | s_t = s}[\sum_{t'=t}^T r(s_{t'}, a_{t'})]$, $Q_t^\pi(s,a) = \mathbb{E}_{\tau \sim \pi | s_t = s, a_t = a}[\sum_{t'=t}^T r(s_{t'}, a_{t'})]$. We also define the advantage function as $A_t^\pi(s,a) = Q_t^\pi(s,a) - V_t^\pi(s)$. Lastly, let performance be $J(\pi) = \mathbb{E}_{\tau \sim \pi}[\sum_{t=1}^T r(s_t, a_t)]$. Then,

Definition 1. *We define the **imitation gap** as:*

$$J(\pi_E) - J(\pi) \tag{1}$$

The goal of the learner is to minimize (1) and *close the imitation gap*. The unique challenge in imitation learning is that the reward function r is unknown and the learner must rely on demonstrations from the expert π_E to minimize the gap. A natural way to solve this problem is to empirically match *all* moments of $J(\pi_E)$. If all the moments are perfectly matched, regardless of the unknown reward function, the imitation gap must go to zero. We now delve into the various types of moments we can match.

3.2. Moment Taxonomy

Broadly speaking, a learner can focus on matching per-timestep *reward* or over-the-horizon *Q-value* moments of expert behavior. We use $\mathcal{F}_r : \mathcal{S} \times \mathcal{A} \to [-1, 1]$ to denote a class of reward functions, $\mathcal{F}_Q : \mathcal{S} \times \mathcal{A} \to [-T, T]$ to denote the set of Q functions induced by sampling actions from some $\pi \in \Pi$, and $\mathcal{F}_{Q_E} : \mathcal{S} \times \mathcal{A} \to [-\overline{Q}, \overline{Q}]$ to denote the set of Q functions induced by sampling actions from π_E. We assume all three function classes are closed under negation. Lastly, we refer to $H \in [0, 2\overline{Q}]$ as the *recoverability constant* of the problem and define it as follows:

Definition 2. *A pair $(\pi_E, \mathcal{F}_{Q_E})$ of an expert policy and set of expert Q-functions is said to be* $\mathbf{H}$-***recoverable*** *if $\forall s \in \mathcal{S}$, $\forall a \in \mathcal{A}$, $\forall f \in \mathcal{F}_{Q_E}$, $|f(s,a) - \mathbb{E}_{a' \sim \pi_E(s)}[f(s,a')]| < H$.*

H is an upper bound on all possible advantages that could be obtained by the expert under a Q-function in $\mathcal{F}_{Q_E}$. Intuitively, H tells us how many time-steps it takes the expert to recover from an arbitrary mistake. We defer a more in-depth discussion of the implications of this concept to Section 4.5.

Reward. Matching reward moments entails minimizing the following expansion of the imitation gap:

$$\begin{aligned}
&J(\pi_E) - J(\pi) \\
&= \mathop{\mathbb{E}}_{\tau \sim \pi_E} \sum_{t=1}^T r(s_t, a_t) - \mathop{\mathbb{E}}_{\tau \sim \pi} \sum_{t=1}^T r(s_t, a_t) \\
&= \mathop{\mathbb{E}}_{\tau \sim \pi} \sum_{t=1}^T -r(s_t, a_t) - \mathop{\mathbb{E}}_{\tau \sim \pi_E} \sum_{t=1}^T -r(s_t, a_t) \\
&\leq \sup_{f \in \mathcal{F}_r} \mathop{\mathbb{E}}_{\tau \sim \pi} \sum_{t=1}^T f(s_t, a_t) - \mathop{\mathbb{E}}_{\tau \sim \pi_E} \sum_{t=1}^T f(s_t, a_t)
\end{aligned}$$

Moment	Class	Env.	π_E Queries
Reward	$\mathcal{F}_r : [-1, 1]$	✓	✗
Off-Policy Q	$\mathcal{F}_Q : [-T, T]$	✗	✗
On-Policy Q	$\mathcal{F}_{Q_E} : [-\overline{Q}, \overline{Q}]$	✓	✓

Table 1. An overview of the requirements for the three classes of moment matching.

In the last step, we use the fact that $-r(s,a) \in \mathcal{F}_r$. Crucially, reward moment-matching demands on-policy rollouts $\tau \sim \pi$ for the learner to calculate per-timestep divergences.

Instead of matching moments of the reward function, we can consider matching moments of the action-value function. We can apply the Performance Difference Lemma (PDL) to expand the imitation gap (1) into either on-policy or off-policy expressions.

Off-Policy Q. Starting from the PDL:

$$\begin{aligned}
&J(\pi_E) - J(\pi) \\
&= \mathop{\mathbb{E}}_{\tau \sim \pi_E} [\sum_{t=1}^T Q_t^\pi(s_t, a_t) - \mathop{\mathbb{E}}_{a \sim \pi(s_t)}[Q_t^\pi(s_t, a)]] \\
&\leq \sup_{f \in \mathcal{F}_Q} \mathop{\mathbb{E}}_{\tau \sim \pi_E} [\sum_{t=1}^T \mathop{\mathbb{E}}_{a \sim \pi(s_t)} [f(s_t, a)] - f(s_t, a_t)]
\end{aligned}$$

In the last step, we use the fact that $Q_t^\pi(s,a) \in \mathcal{F}_Q$ for all $\pi \in \Pi$ and $r \in \mathcal{F}_r$. The above expression is off-policy – it only requires a collected dataset of expert trajectories to be evaluated and minimized. In general though, $\mathcal{F}_Q$ can be a far more complex class than $\mathcal{F}_r$ because it has to capture both the dynamics of the MDP and the choices of *any* policy.

On-Policy Q. Expanding in the reverse direction:

$$\begin{aligned}
&J(\pi_E) - J(\pi) \\
&= - \mathop{\mathbb{E}}_{\tau \sim \pi} [\sum_{t=1}^T Q_t^{\pi_E}(s_t, a_t) - \mathop{\mathbb{E}}_{a \sim \pi_E(s_t)}[Q_t^{\pi_E}(s_t, a)]] \\
&\leq \sup_{f \in \mathcal{F}_{Q_E}} \mathop{\mathbb{E}}_{\tau \sim \pi} [\sum_{t=1}^T f(s_t, a_t) - \mathop{\mathbb{E}}_{a \sim \pi_E(s_t)}[f(s_t, a)]]
\end{aligned}$$

In the last step, we use the fact that $Q_t^{\pi_E}(s,a) \in \mathcal{F}_{Q_E}$ for all $r \in \mathcal{F}_r$. In the realizable setting, $\pi_E \in \Pi$, $\mathcal{F}_{Q_E} \subseteq \mathcal{F}_Q$. While $\mathcal{F}_{Q_E}$ is a smaller class, to actually evaluate this expression, we require an *interactive* expert that can tell us what action they would take in any state visited by the learner as well as on-policy samples from the learner's current policy $\tau \sim \pi$.

With this taxonomy in mind, we now turn our attention to deriving policy performance bounds. [2]

[2]See Sup. E for mixed moments and an alternative Q-moment scheme that can be extended to the IL from observation alone setting.

3.3. Moment Matching Games

A unifying perspective on the three moment matching variants can be achieved by viewing the learner as solving a game. More specifically, we consider variants of a two-player minimax game between a learner and a discriminator. The learner selects a policy $\pi \in \Pi$, where $\Pi \triangleq \{\pi : \mathcal{S} \rightarrow \Delta(\mathcal{A})\}$. We assume Π is convex, compact and that $\pi_E \in \Pi$.[3] The discriminator (adversarially) selects a function $f \in \mathcal{F}$, where $\mathcal{F} \triangleq \{f : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}\}$. We assume that $\mathcal{F}$ is convex, compact, closed under negation, and finite dimensional.[4] Depending on the class of moments being matched, we assume that $\mathcal{F}$ is spanned by convex combinations of the elements of $\mathcal{F}_r/2$, $\mathcal{F}_Q/2T$, or $\mathcal{F}_{Q_E}/2T$. Lastly, we set the learner as the *minimization player* and the discriminator as the *maximization player*.

Definition 3. *The* ***on-policy reward***, ***off-policy* Q**, *and* ***on-policy* Q** *payoff functions are, respectively:*

$$U_1(\pi, f) = \frac{1}{T}\Big(\mathop{\mathbb{E}}_{\tau\sim\pi}\Big[\sum_{t=1}^{T} f(s_t, a_t)\Big] - \mathop{\mathbb{E}}_{\tau\sim\pi_E}\Big[\sum_{t=1}^{T} f(s_t, a_t)\Big]\Big)$$

$$U_2(\pi, f) = \frac{1}{T}\Big(\mathop{\mathbb{E}}_{\substack{\tau\sim\pi_E \\ a\sim\pi(s_t)}}\Big[\sum_{t=1}^{T} f(s_t, a)\Big] - \mathop{\mathbb{E}}_{\tau\sim\pi_E}\Big[\sum_{t=1}^{T} f(s_t, a_t)\Big]\Big)$$

$$U_3(\pi, f) = \frac{1}{T}\Big(\mathop{\mathbb{E}}_{\tau\sim\pi}\Big[\sum_{t=1}^{T} f(s_t, a_t)\Big] - \mathop{\mathbb{E}}_{\substack{\tau\sim\pi \\ a\sim\pi_E(s_t)}}\Big[\sum_{t=1}^{T} f(s_t, a)\Big]\Big)$$

When optimizing over the policy class Π which contains π_E, we have a minimax value of 0: for $j \in \{1, 2, 3\}$,

$$\min_{\pi\in\Pi} \max_{f\in\mathcal{F}} U_j(\pi, f) = 0$$

Furthermore, for certain representations of the policy,[5] strong duality holds: for $j \in \{1, 2, 3\}$,

$$\min_{\pi\in\Pi} \max_{f\in\mathcal{F}} U_j(\pi, f) = \max_{f\in\mathcal{F}} \min_{\pi\in\Pi} U_j(\pi, f)$$

We now study the properties that result from achieving an approximate equilibrium for each imitation game.

4. From Approximate Equilibria to Bounded Regret

A learner computing an equilibrium policy for any of the moment matching games will be imperfect due to many sources including restricted policy class, optimization error, or imperfect estimation of expert moments. More formally, in a game with payoff U_j, a pair $(\hat{\pi} \in \Pi, \hat{f} \in \mathcal{F})$ is a *δ-approximate equilibrium solution* if the following holds:

$$\sup_{f\in\mathcal{F}} U_j(f, \hat{\pi}) - \frac{\delta}{2} \leq U_j(\hat{f}, \hat{\pi}) \leq \inf_{\pi\in\Pi} U_j(\hat{f}, \pi) + \frac{\delta}{2}$$

We assume access to an *algorithmic primitive* capable of finding such strategies:

Definition 4. *An* ***imitation game δ-oracle*** $\mathbf{\Psi}\{\delta\}(\cdot)$ *takes payoff function $U : \Pi \times \mathcal{F} \rightarrow [-k, k]$ and returns a $(k\delta)$-approximate equilibrium strategy for the policy player.*

We now bound the imitation gap of solutions returned by such an oracle.

Moment Matched	Upper Bound	Lower Bound
Reward	$O(\epsilon T)$	$\Omega(\epsilon T)$
Off-Policy Q	$O(\epsilon T^2)$	$\Omega(\epsilon T^2)$
On-Policy Q	$O(\epsilon H T)$	$\Omega(\epsilon T)$

Table 2. An overview of the difference in bounds between the three types of moment matching. All bounds are on imitation gap (1).

4.1. Example MDPs

For use in our analysis, we first introduce two MDPs, Loop and Cliff. As seen in Fig. 2, Loop is an MDP where a learner can enter a state where it has seen no expert demonstrations (s_2) and make errors for the rest of the horizon. Cliff is an MDP where a single mistake can result in the learner being stuck in an absorbing state.

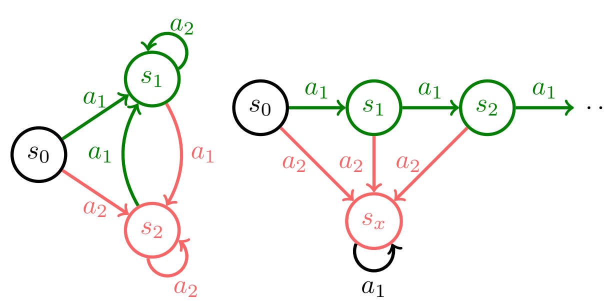

Figure 2. Left: Borrowed from (Ross et al. 2011), the goal of Loop is to spend time in s_1. Right: a folklore MDP Cliff, where the goal is to not "fall off the cliff" and end up in s_x evermore.

4.2. Reward Moment Performance Bounds

Let us first consider the reward moment-matching game.

Lemma 1. ***Reward Upper Bound:*** *If $\mathcal{F}_r$ spans $\mathcal{F}$, then for all MDPs, π_E, and $\pi \leftarrow \Psi\{\epsilon\}(U_1)$, $J(\pi_E) - J(\pi) \leq O(\epsilon T)$.*

[3] The full policy class satisfies all these assumptions.

[4] Our results extend to infinite-dimensional Reproducing Kernel Hilbert Spaces.

[5] One option is a mixture distribution over class Π' where optimization is now performed over the mixture weights. This distribution can then be collapsed to a single policy (Syed et al. 2008). A second option is to optimize over the causal polytope $P(\mathbf{A}^T||\mathbf{S}^T)$, as defined in (Ziebart et al. 2010).

Proof. We start by expanding the imitation gap:

$$\begin{aligned}
&J(\pi_E) - J(\pi) \\
&\leq \sup_{f \in \mathcal{F}_r} \mathop{\mathbb{E}}_{\tau \sim \pi} \sum_{t=1}^{T} f(s_t, a_t) - \mathop{\mathbb{E}}_{\tau \sim \pi_E} \sum_{t=1}^{T} f(s_t, a_t) \\
&\leq \sup_{f \in \mathcal{F}} \mathop{\mathbb{E}}_{\tau \sim \pi} \sum_{t=1}^{T} 2f(s_t, a_t) - \mathop{\mathbb{E}}_{\tau \sim \pi_E} \sum_{t=1}^{T} 2f(s_t, a_t) \\
&= 2T \sup_{f \in \mathcal{F}} U_1(\pi, f) \leq 2T\epsilon
\end{aligned}$$

The first line follows from the closure of $\mathcal{F}_r$ under negation. The last line follows from the definition of an ϵ-approximate equilibrium. □

In words, this bound means that in the worst case, we have an imitation gap that is $O(\epsilon T)$ rather than an imitation gap that compounds quadratically in time.

Lemma 2. ***Reward Lower Bound:*** *There exists an MDP, π_E, and $\pi \leftarrow \Psi\{\epsilon\}(U_1)$ such that $J(\pi_E) - J(\pi) \geq \Omega(\epsilon T)$.*

Proof. Consider CLIFF with a reward function composed of two indicators: $r(s, a) = -\mathbb{1}_{s_x} - \mathbb{1}_{a_2}$ and a perfect expert that never takes a_2. If with probability ϵ the learner's policy takes action a_2 only in s_0, the optimal discriminator would not only be able to penalize the learner for taking a_2 but also for the next $T-1$ timesteps for being in s_x. Together, this would lead to an average cost of ϵ per timestep. Under r, this would make the learner ϵT worse than the expert, giving us $J(\pi_E) - J(\pi) = \epsilon T \geq \Omega(\epsilon T)$. □

Notably, both of these bounds are purely a function of the *game*, not the policy search algorithm and therefore apply for *all* algorithms that can be written in the form of a reward moment-matching imitation game. Our bounds do not depend on the size of the state space and therefore apply to continuous spaces, unlike those presented in (Rajaraman et al. 2020). Several recently proposed algorithms (Ho and Ermon 2016, Brantley et al. 2020, Spencer et al. 2021, Yang et al. 2020) including GAIL and SQIL can be understood as also solving this or a related game.

4.3. Off-Q Moment Performance Bounds

We contrast the preceding guarantees with those based on matching off-Q moments.

Lemma 3. ***Off-*Q *Upper Bound:*** *If $\mathcal{F}_Q$ spans $\mathcal{F}$, then for all MDPs, π_E, and $\pi \leftarrow \Psi\{\epsilon\}(U_2)$, $J(\pi_E) - J(\pi) \leq O(\epsilon T^2)$.*

Proof. Starting from the PDL:

$$\begin{aligned}
&J(\pi_E) - J(\pi) \\
&\leq \sup_{f \in \mathcal{F}_Q} \mathop{\mathbb{E}}_{\tau \sim \pi_E} [\sum_{t=1}^{T} \mathop{\mathbb{E}}_{a \sim \pi(s_t)} [f(s_t, a)] - f(s_t, a_t)] \\
&\leq \sup_{f \in \mathcal{F}} \mathop{\mathbb{E}}_{\tau \sim \pi_E} [\sum_{t=1}^{T} \mathop{\mathbb{E}}_{a \sim \pi(s_t)} [2Tf(s_t, a)] - 2Tf(s_t, a_t)] \\
&= 2T^2 \sup_{f \in \mathcal{F}} U_2(\pi, f) \leq 2\epsilon T^2
\end{aligned}$$

The T in the second to last line comes from the fact that $\mathcal{F}_Q/2T \subseteq \mathcal{F}$. Thus, a policy π returned by $\Psi\{\epsilon\}(U_2)$ must satisfy $J(\pi_E) - J(\pi) \leq O(\epsilon T^2)$ – that is, it can do up to $O(\epsilon T^2)$ worse than the expert. □

Lemma 4. ***Off-*Q *Lower Bound:*** *There exists an MDP, π_E, and $\pi \leftarrow \Psi\{\epsilon\}(U_2)$ such that $J(\pi_E) - J(\pi) \geq \Omega(\epsilon T^2)$.*

Proof. Once again, consider CLIFF. If the learner policy instead takes a_2 with probability ϵT in s_0, the optimal discriminator would be able to penalize the learner up to ϵT for that timestep and ϵ on average. However, on rollouts, the learner would have an ϵT chance of paying a cost of 1 for the rest of the horizon, leading to a lower bound of $J(\pi_E) - J(\pi) = \epsilon T^2 \geq \Omega(\epsilon T^2)$. □

These bounds apply for *all* algorithms that can be written in the form of an off-Q imitation game, including behavioral cloning (Pomerleau 1989) and ValueDICE (Kostrikov et al. 2019).

4.4. On-Q Moment Performance Bounds

We now derive performance bounds for on-Q algorithms with interactive experts.

Lemma 5. ***On-*Q *Upper Bound:*** *If $\mathcal{F}_{Q_E}$ spans $\mathcal{F}$, then for all MDPs with H-recoverable $(\mathcal{F}_{Q_E}, \pi_E)$, and $\pi \leftarrow \Psi\{\epsilon\}(U_3)$, $J(\pi_E) - J(\pi) \leq O(\epsilon HT)$.*

Proof. Starting from the PDL:

$$\begin{aligned}
&J(\pi_E) - J(\pi) \\
&\leq \sup_{f \in \mathcal{F}_{Q_E}} \mathop{\mathbb{E}}_{\tau \sim \pi} [\sum_{t=1}^{T} f(s_t, a_t) - \mathop{\mathbb{E}}_{a \sim \pi_E(s_t)} [f(s_t, a_t)]] \\
&\leq \sup_{f \in \mathcal{F}} \mathop{\mathbb{E}}_{\tau \sim \pi_E} [\sum_{t=1}^{T} 2Tf(s_t, a_t) - \mathop{\mathbb{E}}_{a \sim \pi_E(s_t)} [2Tf(s_t, a)]] \\
&= 2T^2 \sup_{f \in \mathcal{F}} U_3(\pi, f) \leq (\epsilon \frac{H}{2T}) 2T^2 = \epsilon HT
\end{aligned}$$

As before, the T in the second to last line comes from the fact that $\mathcal{F}_{Q_E}/2T \subseteq \mathcal{F}$. The $H/2T$ factor comes from the scale of the payoff. Thus, a policy π returned by $\Psi\{\epsilon\}(U_3)$ must satisfy $J(\pi_E) - J(\pi) \leq O(\epsilon HT)$ – that is, it can do up to $O(\epsilon HT)$ worse than the expert. □

Lemma 6. ***On-*Q *Lower Bound:*** *There exists an MDP, π_E, and $\pi \leftarrow \Psi\{\epsilon\}(U_3)$ such that $J(\pi_E) - J(\pi) \geq \Omega(\epsilon T)$.*

Proof. The proof of the *reward lower bound* holds verbatim because every policy, including the previously considered π_E will be stuck in s_x after it falls in. □

As before, these bounds apply for all algorithms that can be written in the form of an on-Q imitation game, including DAgger (Ross et al. 2011) and AggreVaTe (Ross and Bagnell 2014). For example, in the bounds for AggreVaTe, Q_{max} is equivalent to the recoverability constant H.

4.5. Recoverability in Imitation Learning

The bounds we presented above beg the question of when on-Q moment matching has error properties similar to those of reward moment matching versus those of off-Q moment matching. Recoverability allows us to cleanly answer this question and others. We begin by providing more intuition for said concept.

Concretely, in Fig. 2, LOOP is 1-recoverable for the expert policy that always moves towards s_1. CLIFF is not H-recoverable for any $H < T$ if the expert never ends up in s_x. A sufficient condition for H-recoverability is that the state occupancy distribution that results from taking an arbitrary action and then $H-1$ actions according to π_E is the same as that of taking H actions according to π_E. We emphasize that recoverability is a property of the *set of moments* matched and the expert, not just of the expert, as has been previously considered (Pan et al. 2019).

Bound Clarification. Our previously derived upper bound for on-Q moment matching ($J(\pi_E) - J(\pi) \leq O(\epsilon HT)$) tells us that for $O(1)$-recoverable MDPs, on-Q moment matching behaves like reward moment matching while for $O(T)$-recoverable MDPs, it instead behaves like off-Q moment matching and has an $O(\epsilon T^2)$ upper bound. Thus, $O(1)$-recoverability is in a certain sense necessary for achieving $O(\epsilon T)$ error with on-Q moment matching.

Another perspective on recoverability is that it helps us delineate problems where compounding errors are hard to avoid for both on-Q and reward moment matching. Let $l(s) = \sum_{a' \in \mathcal{A}} |\mathbb{E}_{a \sim \pi_E(s)}[\mathbb{1}_{a'}(a)] - \mathbb{E}_{a \sim \pi(s)}[\mathbb{1}_{a'}(a)]|$ be the classification error of a state s. We prove the following lemma in Supplementary Material A.1:

Lemma 7. *Let $\kappa > 0$. There exists a $(\pi_E, \{r, -r\})$ pair that for any $H < T$ is not H-recoverable such that $l(s) = \kappa$ on any state leads to $J(\pi_E) - J(\pi) \geq \Omega(\kappa T^2)$.*

Because some states might not appear on expert rollouts, evaluating $l(s)$ can require an interactive expert. However, even with this strong form of feedback, a classification error of κ on a single state can lead to $\Omega(\kappa T^2)$ imitation gap for $O(T)$-recoverable MDPs. This lemma also implies that in such an MDP, achieving $O(\kappa T)$ imitation gap via on-policy moment matching would require the learner to have a classification error $\propto \kappa/T$, or to make vanishingly rare errors as we increase the horizon. We note that this does not conflict with our previously stated bounds but reveals that achieving a moment-matching error of (time-independent) ϵ might require achieving a classification error that scales inversely with time for $O(T)$-recoverable MDPs. Practically, this can be rather challenging. Thus, neither on-Q nor reward moment matching is a silver bullet for getting $O(\epsilon T)$ error for $O(T)$-recoverable problems.

5. Finding Equilibria: Idealized Algorithms

We now provide reduction-based methods for computing (approximate) equilibria for these moment matching games which can be seen as blueprints for constructing our previously described oracle (Def. 4). We study, in particular, finite state problems and a complete policy class. We analyze an approach to equilibria finding where an outer player follows a *no-regret* strategy and the inner player follow a (modified) *best response* strategy, by which we can efficiently find policies with strong performance guarantees.

5.1. Preliminaries

An *efficient no-regret algorithm* over a class $\mathcal{X}$ produces iterates $x^1, \ldots, x^H \in \mathcal{X}$ that satisfy the following property for any sequence of loss functions $l^1, \ldots, l^H$:

$$\text{Regret}(H) = \sum_t^H l^t(x^t) - \min_{x \in \mathcal{X}} \sum_t^H l^t(x) \leq \beta_{\mathcal{X}}(H)$$

where $\beta_{\mathcal{X}}(H)/H \leq \epsilon$ holds for H that are $O(\text{poly}(\frac{1}{\epsilon}))$.

5.2. Theoretical Guarantees

We are interested in obtaining a policy efficiently that is a near-equilibrium solution to the game. We consider two general strategies to do so:

Primal. We execute a no-regret algorithm on the policy representation, while a maximization oracle over the space $\mathcal{F}$ computes the best response to those policies.

Dual. We execute a no-regret algorithm on the space $\mathcal{F}$, while a minimization oracle over policies computes *entropy regularized* best response policies.

The asymmetry in the above is driven by the need to recover the equilibrium strategy for the policy player and the fact that a dual approach on the original, unregularized objective $U_j(\cdot, f)$ will typical not converge to a single policy but rather shift rapidly as f changes.[6]

By choosing the the policy representation to be a causally conditioned probability distribution over actions, $P(\mathbf{A}^T||\mathbf{S}^T) = \prod_{t=1}^T P(A_t|\mathbf{S}_{1:t}, \mathbf{A}_{1:t-1})$, we find each of the imitation games is *bilinear* in both policy and discriminator f and strongly dual .[7] Thus, we can efficiently compute a near-equilibrium assuming access to the optimization oracles in either *primal* or *dual* above:

[6]The average over iterations of the policies generated in an unregularized dual will also be near-equilibrium but can be inconvenient. Entropy regularization provides a convenient way to extract a single policy and meshes well with empirical practice.

[7]Following (Ziebart et al. 2010) we can represent the policy as an element of the causal conditioned polytope and regularize with the causal entropy $H(P(\mathbf{A^T}||\mathbf{S^T}))$ to denote the causal Shannon entropy of a policy. An equivalent result can be proved for optimizing over *occupancy measures* (Ho and Ermon 2016).

Algorithm 1 `AdVIL`

Input: Expert demonstrations $\mathcal{D}_E$, Policy class Π, Discriminator class $\mathcal{F}$, Performance threshold δ, Learning rates $\eta_f > \eta_\pi$
Output: Trained policy π
Set $\pi \in \Pi$, $f \in \mathcal{F}$, $L(\pi, f) = 2\delta$
while $L(\pi, f) > \delta$ **do**
  $L(\pi, f) = \mathbb{E}_{(s,a)\sim\mathcal{D}_E}[\mathbb{E}_{x\sim\pi(s)}[f(s,x)] - f(s,a)]$
  $f \leftarrow f + \eta_f \nabla_f L(\pi, f)$
  $\pi \leftarrow \pi - \eta_\pi \nabla_\pi L(\pi, f)$
end while

Algorithm 2 `AdRIL`

Input: Expert demonstrations $\mathcal{D}_E$, Policy class Π, Dynamics $\mathcal{T}$, Kernel K, Performance threshold δ
Output: Trained policy π
Set $\pi \in \Pi$, $f = 0$, $\mathcal{D}_\pi = \{\}$, $\mathcal{D}' = \{\}$, $L(\pi, f) = 2\delta$
while $L(\pi, f) > \delta$ **do**
  $f \leftarrow \mathbb{E}_{\tau\sim D_\pi}[\sum_t K(sa, \cdot)] - \mathbb{E}_{\tau\sim D_E}[\sum_t K(sa, \cdot)]$
  $\pi, \mathcal{D}' \leftarrow \mathtt{MaxEntRL}(\mathtt{T} = \mathcal{T}, \mathtt{r} = -f)$
  $\mathcal{D}_\pi \leftarrow \mathcal{D}_\pi \cup \mathcal{D}'$
  $L(\pi, f) = \mathbb{E}_{\tau\sim D'}[\sum_t f(s,a)] - \mathbb{E}_{\tau\sim D_E}[\sum_t f(s,a)]$
end while

Theorem 1. *Given access to the no-regret and maximization oracles in either **primal** or **dual** above, for all three imitation games we are able to compute a δ-approximate equilibrium strategy for the policy player in poly$(\frac{1}{\delta}, T, \ln|\mathcal{S}|, \ln|\mathcal{A}|)$ iterations of the outer player optimization.*

This result relies on a general lemma that establishes we can recover the equilibrium inner player by appropriately entropy regularizing that inner player's decisions. We prove both in Sup. A.

By substituting $\delta = \epsilon$ in Theorem 1 and combining with our previously derived bounds, we can cleanly show that these theorems can also be viewed as certificates of policy performance. Thus, our framework enables us to efficiently *close the imitation gap*.

6. Practical Moment Matching Algorithms

We now present three practical algorithms that match reward moments (`AdRIL`), off-Q moments (`AdVIL`), or on-Q moments (`DAeQuIL`). They are specific, implementable versions of our preceding abstract procedures. At their core, all three algorithms optimize an IPM. IPMs are a distance between two probability distributions (Müller et al. 1997):

$$\sup_{f\in\mathcal{F}} \mathop{\mathbb{E}}_{x\sim P_1}[f(x)] - \mathop{\mathbb{E}}_{x\sim P_2}[f(x)]$$

Plugging in the learner and expert trajectory distributions, we end up with our IPM-based objective:

$$\sup_{f\in\mathcal{F}} \sum_{t=1}^{T} \mathop{\mathbb{E}}_{\tau\sim\pi}[f(s_t, a_t)] - \mathop{\mathbb{E}}_{\tau\sim\pi_E}[f(s_t, a_t)] \tag{2}$$

As noted previously, this objective is equivalent to and inherits the strong guarantees of moment matching and allows our analysis to easily transfer.

6.1. `AdVIL`: Adversarial Value-moment Imitation Learning

We can transform our IPM-based objective (2) into an expression only over (s, a) pairs from expert data to fit into the off-Q moment matching framework by performing a series of substitutions. We refer interested readers to Supplementary Material B.1. We arrive at the following expression:

$$\sup_{v\in\mathcal{F}} \mathop{\mathbb{E}}_{\tau\sim\pi_E}[\sum_{t=1}^{T} \mathop{\mathbb{E}}_{a\sim\pi(s_t)}[v(s_t, a)] - v(s_t, a_t)] \tag{3}$$

Intuitively, by minimizing (3) over policies $\pi \in \Pi$, one is driving down the extra cost the learner has over the expert. We term this approach *Adversarial Value-moment Imitation Learning* (`AdVIL`) because if we view π as optimizing cumulative reward, $-v$ could be viewed as a value function. We set the learning rate for f to be greater than that for π, making `AdVIL` a primal algorithm.

Practically, `AdVIL` bears similarity to the Wasserstein GAN (WGAN) framework (Arjovsky et al. 2017), though we consider an IPM rather than the more restricted Wasserstein distances. However, the overlap is enough that techniques from the WGAN literature including gradient penalties on the discriminator (Gulrajani et al. 2017) and orthogonal regularization of the policy player (Brock et al. 2018) help.

6.2. `AdRIL`: Adversarial Reward-moment Imitation Learning

We now present our dual reward moment matching algorithm and refer interested readers to Supplementary Material B.2 for the full derivation. In brief, we solve for the discriminator in closed form via functional gradient descent in Reproducing Kernel Hilbert Space (RKHS) and have the policy player compute a best response via maximum entropy reinforcement learning. Let $\overline{\mathcal{D}_k}$ denote the aggregated dataset of policy rollouts. Assuming a constant number of training steps at each iteration and averaging functional gradients over k iterations of the algorithm, we get the cost function for the policy and round k:

$$\sum_{t=1}^{T} \frac{1}{|\overline{\mathcal{D}_k}|} \sum_{\tau}^{\overline{\mathcal{D}_k}} K([s_t, a_t], \cdot) - \frac{1}{|\mathcal{D}_E|} \sum_{\tau}^{\mathcal{D}_E} K([s_t, a_t], \cdot)$$

For an indicator kernel and under the assumption we never see the same state twice, this is equivalent to maximizing a reward function that is 1 at each expert datapoint,

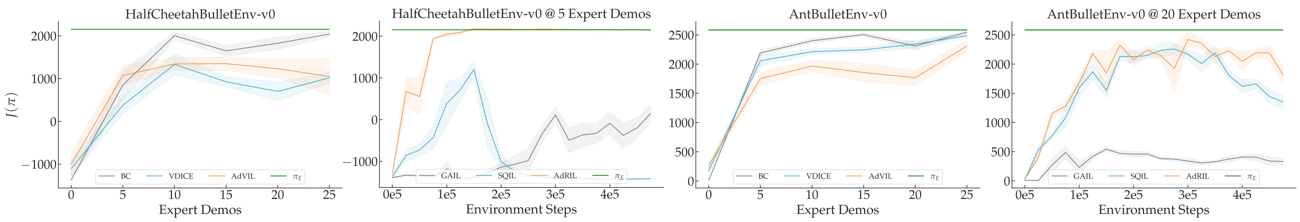

Figure 3. Our proposed methods (in orange) are able to match or out-perform the baselines across a variety of continuous control tasks. $J(\pi)$s are averaged across 10 evaluation episodes. Standard errors are across 5 runs of each algorithm.

$\propto \frac{-1}{k}$ along previous rollouts that do not perfectly match the expert, and 0 everywhere else. We term this approach *Adversarial Reward-moment Imitation Learning* (`AdRIL`).

We note that under these assumption, our objective resembles that of SQIL (Reddy et al. 2019). SQIL can be seen as a degenerate case of `AdRIL` that never updates the discriminator function. This oddity removes solution quality guarantees while introducing the need for early stopping (Arenz and Neumann 2020).

6.3. `DAeQuIL`: DAgger-esque Qu-moment Imitation Learning

We present the natural extension of DAgger (Ross et al. 2011) to the space of moments: `DAeQuIL` (DAgger-esque Qu-moment Imitation Learning) in Algorithm 3. Like DAgger, one can view `DAeQuIL` as a primal algorithm that uses Follow the Regularized Leader as the no-regret algorithm for the policy player. Per-round losses are adversarially chosen though. It is a subtle point but as written, `DAeQuIL` is technically not solving the on-Q game directly because it optimizes over a history of learner state distributions instead of the current learner's state distribution. However, it retains strong performance guarantees – see Sup. B.3 for more.

Algorithm 3 `DAeQuIL`

Input: Queryable expert π_E, Policy class Π, Discriminator class $\mathcal{F}$, Performance threshold δ, Behavioral cloning loss $\ell_{BC} : \Pi \to \mathbb{R}$, Strongly convex fn $R : \Pi \to \mathbb{R}$
Output: Trained policy π
Optimize: $\pi \leftarrow \arg\min_{\pi' \in \Pi} \ell_{BC}(\pi')$.
Set $L(\pi) = 2\delta$, $\mathcal{D} = []$, $F = []$, $t = 1$
while $L(\pi) > \delta$ **do**
  Rollout π to generate $\mathcal{D}_\pi \leftarrow [(s, a), \ldots]$.
  Relabel $\mathcal{D}_\pi$ to $\mathcal{D}_E \leftarrow [(s, a') | a' \sim \pi_E(s), \forall s \in \mathcal{D}_\pi]$
  $L(f) = \mathbb{E}_{(s,a)\sim\mathcal{D}_\pi}[f(s, a)] - \mathbb{E}_{(s,a)\sim\mathcal{D}_E}[f(s, a)]$
  Append: $F \leftarrow F \cup [\arg\max_{f' \in \mathcal{F}} L(f')]$.
  Append: $\mathcal{D} \leftarrow \mathcal{D} \cup [(s, t) \,|\, \forall s \in \mathcal{D}_\pi]$.
  $L(\pi) = \mathbb{E}_{(s,t)\in\mathcal{D}}[F[t](s, \pi(s))] + \ell_{BC}(\pi) + R(\pi)$
  Optimize: $\pi \leftarrow \arg\min_{\pi' \in \Pi} L(\pi')$.
  $t \leftarrow t + 1$
end while

7. Experiments

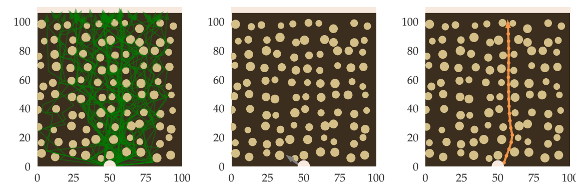

Figure 4. **Left:** The expert demonstrates many feasible trajectories, causing a learner that attempts to just match the mean action to crash directly into the tree the expert was avoiding. **Center:** On-policy corrections do not help DAgger as it still tries to match the mean action and crashes into the first tree it encounters. **Right:** `DAeQuIL`, when run with moments that allow the learner to focus on swerving out of the way of trees, regardless of direction, is able to produce policies that successfully navigate through the forest.

We test our algorithms against several baselines on several higher-dimensional continuous control tasks from the PyBullet suite (Coumans and Bai 2016–2019). We measure the performance of off-Q algorithms as a function of the amount of data provided with a fixed maximum computational budget and of reward moment-matching algorithms as a function of the amount of environment interactions. We see from Fig. 3 that `AdVIL` can match the performance of ValueDICE and Behavioral Cloning across most tasks. `AdRIL` performs better than GAIL across all environments and does not exhibit the catastrophic collapse in performance SQIL does on the tested environments. On both environments, behavioral cloning is able to recover the optimal policy with enough data, indicating there is little covariate shift (Spencer et al. 2021). However, on HalfCheetah, we see `AdRIL` recover a strong policy with far less data than it takes Behavioral Cloning to, showcasing the potential benefit of the learner observing the consequences of their own actions. We refer readers to Sup. C for a description of our hyparameters and setup. Notably, `AdVIL` is able to converge reliably to a policy of the same quality as that found by ValueDICE with an order of magnitude less compute. As seen in Fig. 4, `DAeQuIL` is able to significantly out-perform DAgger on a toy UAV navigation task – see Sup. D for full information. We release our code at **`https://github.com/gkswamy98/pillbox`**.

8. Discussion

8.1. A Unifying View of Moment Matching IL

We present a cohesive perspective of moment matching in imitation learning in Table 3. We note that reward moment-matching dual algorithms have been a repeated success story in imitation learning but that there has been comparatively less work done in off-Q and on-Q dual algorithms.

MOMENT	PRIMAL	DUAL
OFF-Q	VDICE, **ADVIL**	✗
REWARD	GAIL	MMP, LEARCH, MAXENT IOC, SQIL, **ADRIL**, A+N
ON-Q	DAGGER, GPS **IFAIL**, **DAEQUIL**	✗

Table 3. An taxonomy of moment matching algorithms. **Bold** text indicates algorithms that are IPM-based.

8.2. The Hidden Cost of Reward Moment Matching

At first glance, the reward moment matching bound might seem to good to be true – reward matching algorithms don't require a queryable expert like on-Q approaches yet their performance bound seems to be tighter. This better performance characteristic is a product of a potentially *exponentially* harder optimization problem for the learner. Consider the following tree-structured MDP with $|\mathcal{A}|$ actions at each step, each of which lead to a distinct state. Consider an

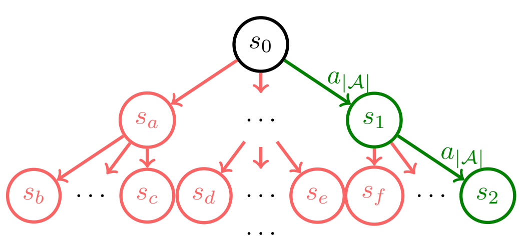

Figure 5. In TREE, the expert always takes the right-most action from the current node. The exponential number of T-length trajectories in this problem make it a challenge for reward moment matching approaches.

expert that takes $a_{|\mathcal{A}|}$ at each timestep. Solving the reward matching problem requires the learner to simultaneously optimize over all T timesteps of the problem while considering the effect of past actions on future states. If we set Π to be the class of deterministic policies, this is equivalent to optimizing over the set of all length T trajectories, of which there are $O(|\mathcal{A}|^T)$ in tree-structured problems. In contrast, for off-Q approaches, we attempt to match expert moments on a fixed expert state distribution. Similarly, we can optimize over a fixed history of past learner state distributions under a weak realizability assumption in the on-Q setting. In the preceding example, the Q-matching settings are like being handed a node at each level of the tree and being asked to choose between the $|\mathcal{A}|$ edges available, leading to a total of $O(|\mathcal{A}|T)$ options to choose between. As we saw in Sec. 5, the policy player sometimes needs to compute a best response over the entire set of choices it has available, which means these search space sizes directly affect the per-iteration complexity of the moment-matching algorithms. Concisely, the price we pay for solving an easier optimization problem is looser policy performance bounds.

8.3. Takeaways

In this work, we tease apart the differences in requirements and performance guarantees that come from matching reward, on-Q, and off-Q adversarially chosen moments. Reward moment matching has strong guarantees but requires access to an accurate simulator or the real world. Off-Q moment matching can be done purely on collected data but incurs an $O(\epsilon T^2)$ imitation gap.

We formalize a notion of *recoverability* that is both necessary and sufficient to understand recovering from errors in imitation learning. If a problem (due to expert or the MDP itself) is $O(T)$-recoverable, there exist problems where no algorithm can escape an $O(T^2)$ compounding of errors; if it is $O(1)$-recoverable, we find on policy algorithms prevent compounding. Together, these constitute a cohesive picture of moment matching in imitation learning.

We derive idealized no-regret procedures and practical IPM-based algorithms that are conceptually elegant and correct for difficulties encountered by prior methods. While behavioral cloning equally weights action-conditional errors, `AdVIL` can prevent headaches with value moment-based weighting. `AdRIL` is simple to implement, does not require training a GAN, and enjoys strong performance both in theory and practice. `DAeQuIL`'s moment-based losses are able to help relieve hiccups from focusing on action-conditionals that can stymie DAgger.

Acknowledgments

We thank Siddharth Reddy for helpful discussions. We also thank Allie Del Giorno, Anirudh Vemula, and the members of the Social AI Group for their comments on drafts of this work. ZSW was supported in part by the NSF FAI Award #1939606, a Google Faculty Research Award, a J.P. Morgan Faculty Award, a Facebook Research Award, and a Mozilla Research Grant. Lastly, GS would like to thank John Steinbeck for inspiring the title of this work:

> *"As happens sometimes, a moment settled and hovered and remained for much more than a moment."* – John Steinbeck, Of Mice and Men.

References

Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1, 2004.

Oleg Arenz and Gerhard Neumann. Non-adversarial imitation learning and its connections to adversarial methods, 2020.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017.

James Bagnell, Sham M Kakade, Jeff Schneider, and Andrew Ng. Policy search by dynamic programming. *Advances in neural information processing systems*, 16:831–838, 2003.

Paul Barde, Julien Roy, Wonseok Jeon, Joelle Pineau, Christopher Pal, and Derek Nowrouzezahrai. Adversarial soft advantage fitting: Imitation learning without policy optimization, 2020.

Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, Xin Zhang, Jake Zhao, and Karol Zieba. End to end learning for self-driving cars. *CoRR*, abs/1604.07316, 2016. URL `http://arxiv.org/abs/1604.07316`.

Kiante Brantley, Wen Sun, and Mikael Henaff. Disagreement-regularized imitation learning. In *Eighth International Conference on Learning Representations (ICLR)*, April 2020.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. `http://pybullet.org`, 2016–2019.

Robert Dadashi, Léonard Hussenot, Matthieu Geist, and Olivier Pietquin. Primal wasserstein imitation learning. *arXiv preprint arXiv:2006.04678*, 2020.

Miroslav Dudik, Steven J Phillips, and Robert E Schapire. Performance guarantees for regularized maximum entropy density estimation. In *International Conference on Computational Learning Theory*, pages 472–486. Springer, 2004.

Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55 (1):119–139, 1997.

Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adversarial inverse reinforcement learning, 2018.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pages 5767–5777, 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*, 2018.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning, 2016.

Daniel Jarrett, Ioana Bica, and Mihaela van der Schaar. Strictly batch imitation learning by energy-based distribution matching, 2020.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *In Proc. 19th International Conference on Machine Learning*. Citeseer, 2002.

Liyiming Ke, Matt Barnes, Wen Sun, Gilwoo Lee, Sanjiban Choudhury, and Siddhartha S. Srinivasa. Imitation learning as f-divergence minimization. *CoRR*, abs/1905.12888, 2019. URL `http://arxiv.org/abs/1905.12888`.

Ilya Kostrikov, Ofir Nachum, and Jonathan Tompson. Imitation learning via off-policy distribution matching, 2019.

Sergey Levine and Vladlen Koltun. Guided policy search. In *International Conference on Machine Learning*, pages 1–9, 2013.

David McAllester and Karl Stratos. Formal limitations on the measurement of mutual information, 2020.

K-R Müller, Alexander J Smola, Gunnar Rätsch, Bernhard Schölkopf, Jens Kohlmorgen, and Vladimir Vapnik. Predicting time series with support vector machines. In *International Conference on Artificial Neural Networks*, pages 999–1004. Springer, 1997.

Yunpeng Pan, Ching-An Cheng, Kamil Saigol, Keuntaek Lee, Xinyan Yan, Evangelos Theodorou, and Byron Boots. Imitation learning for agile autonomous driving. In *The International Journal of Robotics Research (IJRR)*, 2019.

Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. In *Advances in neural information processing systems*, pages 305–313, 1989.

Antonin Raffin. Rl baselines3 zoo. https://github.com/DLR-RM/rl-baselines3-zoo, 2020.

Antonin Raffin, Ashley Hill, Maximilian Ernestus, Adam Gleave, Anssi Kanervisto, and Noah Dormann. Stable baselines3. https://github.com/DLR-RM/stable-baselines3, 2019.

Nived Rajaraman, Lin F. Yang, Jiantao Jiao, and Kannan Ramachandran. Toward the fundamental limits of imitation learning, 2020.

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

Siddharth Reddy, Anca D. Dragan, and Sergey Levine. Sqil: Imitation learning via reinforcement learning with sparse rewards, 2019.

Nicholas Rhinehart, Kris M Kitani, and Paul Vernaza. R2p2: A reparameterized pushforward policy for diverse, precise generative path forecasting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 772–788, 2018.

Stephane Ross and J. Andrew Bagnell. Reinforcement and imitation learning via interactive no-regret learning, 2014.

Stephane Ross, Geoffrey J. Gordon, and J. Andrew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning, 2011.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587): 484–489, 2016.

Jiaming Song, Hongyu Ren, Dorsa Sadigh, and Stefano Ermon. Multi-agent generative adversarial imitation learning. *CoRR*, abs/1807.09936, 2018. URL http://arxiv.org/abs/1807.09936.

Jonathan Spencer, Sanjiban Choudhury, Arun Venkatraman, Brian Ziebart, and J. Andrew Bagnell. Feedback in imitation learning: The three regimes of covariate shift, 2021.

Wen Sun, Geoffrey J Gordon, Byron Boots, and J Bagnell. Dual policy iteration. *Advances in Neural Information Processing Systems*, 31:7059–7069, 2018.

Wen Sun, Anirudh Vemula, Byron Boots, and J. Andrew Bagnell. Provably efficient imitation learning from observation alone, 2019.

Gokul Swamy, Siddharth Reddy, Sergey Levine, and Anca D Dragan. Scaled autonomy: Enabling human operators to control robot fleets. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5942–5948. IEEE, 2020.

Umar Syed, Michael Bowling, and Robert E Schapire. Apprenticeship learning using linear programming. In *Proceedings of the 25th international conference on Machine learning*, pages 1032–1039, 2008.

Gerald Tesauro. Temporal difference learning and td-gammon. *Communications of the ACM*, 38(3):58–68, 1995.

Steven Wang, Sam Toyer, Adam Gleave, and Scott Emmons. The imitation library for imitation learning and inverse reinforcement learning. https://github.com/HumanCompatibleAI/imitation, 2020.

Fan Yang, Alina Vereshchaka, Yufan Zhou, Changyou Chen, and Wen Dong. Variational adversarial kernel learned imitation learning. 34:6599–6606, Apr. 2020. doi: 10.1609/aaai.v34i04.6135. URL https://ojs.aaai.org/index.php/AAAI/article/view/6135.

Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, 2008.

Brian D Ziebart, J Andrew Bagnell, and Anind K Dey. Modeling interaction via the principle of maximum causal entropy. 2010.