Identifiability in inverse reinforcement learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Inverse reinforcement learning attempts to reconstruct the reward function in a Markov decision problem, using observations of agent actions. As already observed in Russell [1998] the problem is ill-posed, and the reward function is not identifiable, even under the presence of perfect information about optimal behavior. We provide a resolution to this non-identifiability for problems with entropy regularization. For a given environment, we fully characterize the reward functions leading to a given policy and demonstrate that, given demonstrations of actions for the same reward under two distinct discount factors, or under sufficiently different environments, the unobserved reward can be recovered up to a constant. Through a simple numerical experiment, we demonstrate the accurate reconstruction of the reward function through our proposed resolution.

1 Introduction

Inverse reinforcement learning aims to use observations of agents' actions to determine their reward function. The problem has roots in the very early stages of optimal control theory; Kalman [1964] raised the question of whether, by observation of optimal policies, one can recover coefficients of a quadratic cost function (see also Boyd et al. [1994]). This question naturally generalizes to the generic framework of Markov decision process and stochastic control.

In the 1970s, these questions were taken up within economics, as a way of determining utility functions from observations. For instance, Keeney and Raiffa [1976] set out to determine a proper ordering of all possible states which are deterministic functions of actions. In this setup, the problem is static and the outcome of an action is immediate. Later in Sargent [1978], a dynamic version of a utility assessment problem was studied, under the context of finding the proper wage through observing dynamic labor demand.

As exemplified by Lucas' critique[1], in many applications it is not enough to find *some* pattern of rewards corresponding to observed policies; instead we may need to identify *the specific* rewards agents face, as it is only with this information that we can make valid predictions for their actions in a changed environment. In other words, we do not simply wish to learn a reward which allows us to imitate agents in the current environment, but which allows us to predict their actions in other settings.

In this paper, we give a precise characterization of the range of rewards which yield a particular policy for a regularized Markov decision problem. This separates the main task of estimation (of the optimal policy from observed actions) from the inverse problem (of inferring rewards from a given policy). We find that even with perfect knowledge of the optimal policy, the corresponding rewards are not

[1] The critique is best summarized by the quotation: "Given that the structure of an econometric model consists of optimal decision rules of economic agents, and that optimal decision rules vary systematically with changes in the structure of series relevant to the decision maker, it follows that any change in [regulatory] policy will systematically alter the structure of econometric models." (Lucas [1976])

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

fully identifiable; nevertheless, the space of consistent rewards is parameterized by the value function of the control problem. In other words, the reward can be fully determined given the optimal policy and the value function, but the optimal policy gives us no direct information about the value function.

We further show that, given knowledge of the optimal policy under two different discount rates, or sufficiently different transition laws, we can uniquely identify the rewards (up to a constant shift). This demonstrates the fundamental challenge of inverse reinforcement learning, which is to disentangle immediate rewards from future rewards (as captured through preferences over future states).

2 Background on reinforcement learning

The motivation behind inverse reinforcement learning is to use observed agent behavior to identify the rewards motivating agents. Given these rewards, one can forecast future behavior, possibly under a different environment. In a typical reinforcement learning[2] (RL) problem, an agent learns an optimal policy to maximize her total reward by interacting with the environment.

In order to analyse the inverse reinforcement learning problem, we begin with an overview of the 'primal' problem, that is, how to determine optimal policies given rewards. We particularly highlight a entropy regularized version of the Markov decision process (MDP), which provides a better-posed setting for inverse reinforcement learning. For mathematical simplicity, we focus on discrete-time problems with finitely many states and actions; our results can all be transferred to continuous settings, however some technical care is needed.

2.1 Discrete Markov decision processes with entropy regularization

The environment. We consider a simple Markov decision process (MDP) on an infinite horizon. The MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, f, \gamma)$ is described by: a finite state space $\mathcal{S}$; a finite set of actions $\mathcal{A}$; a (Markov) transition kernel $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$, that is, a function $\mathcal{T}$ such that $\mathcal{T}(s, a)$ gives probabilities[3] of each value of S_{t+1}, given the state $S_t = s$ and action $A_t = a$ at time t; and a reward function $f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ with discount factor $\gamma \in [0, 1)$.

An agent aims to choose a sequence of actions $\{A_0, A_1, ...\}$ from $\mathcal{A}$ in order to to maximize the expected value of total reward

$$\textstyle\sum_{t=0}^{\infty} \gamma^t f(S_t, A_t).$$

It will prove convenient for us to allow randomized policies π, that is, functions $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$, where $\pi(\cdot|s)$ is the distribution of actions the agent takes when in state s. For a given randomized policy π, we define $\mathcal{T}_\pi \in \mathcal{P}(\mathcal{S})$, the distribution of S_{t+1} given state S_t, by

$$\mathcal{T}_\pi(S_{t+1} = s'|S_t = s) = \textstyle\sum_{a \in \mathcal{A}} \mathcal{T}(s'|s, a)\pi(a|s).$$

Given an initial distribution $\rho \in \mathcal{P}(\mathcal{S})$ for S_0 and a policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$, we obtain a (unique) probability measure $\mathbb{P}^\pi_\rho$ such that, and any $a \in \mathcal{A}, s, s' \in \mathcal{S}$, $\mathbb{P}^\pi_\rho(S_0 = s) = \rho(s)$ and

$$\mathbb{P}^\pi_\rho(A_t = a|S_t = s) = \pi(a|s), \quad \mathbb{P}^\pi_\rho(S_{t+1} = s'|S_t = s, A_t = a) = \mathcal{T}(s'|s, a), \text{ for all } t.$$

We write $\mathbb{E}^\pi_\rho$ for the corresponding expectation and $\mathbb{E}^\pi_s$ when the initial state is given by $s \in \mathcal{S}$. The classic objective in a MDP is to maximize the expected value $\mathbb{E}^\pi_s[\sum_{t=0}^{\infty} \gamma^t f(S_t, A_t)]$. With this objective, one can show (for example, see Bertsekas and Shreve [2004], Puterman [2014]) that there is an optimal deterministic feedback control (i.e. a policy π, taking values zero and one, which maximizes the expected value). This implies that, typically, an optimal agent will only make use of a single action for each state, and the choice of this action will not vary smoothly with changes in the reward, discount rate, or transition kernel.

[2]Reinforcement learning and optimal control are closely related problems, where reinforcement learning typically focuses on the challenge of numerically learning a good control policy, while optimal control focuses on the description of the optimizer. In the context of the inverse problem we consider they are effectively equivalent and we will use the terms interchangeably.

[3]Here, and elsewhere, we write $\mathcal{P}(X)$ for the set of all probability distributions on a set X.

Entropy regularised MDP. Given the lack of smoothness in the classical MDP, and to encourage exploration, a well-known variation on the classic MDP introduces a regularization term based on the Shannon entropy. Given a policy π and regularization coefficient $\lambda \geq 0$, the entropy regularized value of a policy π, when starting in state s, is defined by

$$V_\lambda^\pi(s) := \mathbb{E}_s^\pi\left[\sum_{t=0}^{\infty}\left(\gamma^t f(s_t,a_t) - \lambda\,\log\left(\pi(a_t|s_t)\right)\right)\right] = \mathbb{E}_s^\pi\left[\sum_{t=0}^{\infty}\left(\gamma^t f(s_t,a_t) + \lambda\mathcal{H}\big(\pi(\cdot|s_t)\big)\right)\right]. \tag{1}$$

Here $\mathcal{H}(\pi) = -\sum_{a\in\mathcal{A}}\pi(a)\log(\pi(a))$ is the entropy of π. We call this setting the regularised MDP $\mathcal{M}_\lambda = (\mathcal{S},\mathcal{A},\mathcal{T},f,\gamma,\lambda)$. The optimal value is given by $V_\lambda^*(s) := \max_\pi V_\lambda^\pi(s)$, where the maximum is taken over all (randomized feedback[4]) policies $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$.

We define the state-action value of π at $(s,a) \in \mathcal{S}\times\mathcal{A}$ by

$$Q_\lambda^\pi(s,a) = f(s,a) + \gamma\sum_{s'\in\mathcal{S}}\mathcal{T}(s'|s,a)V_\lambda^\pi(s'). \tag{2}$$

The dynamic programming principle (e.g Haarnoja et al. [2017, Theorem 2]) gives

$$\begin{aligned} V_\lambda^*(s) &= \max_{m\in\mathcal{P}(\mathcal{A})}\left[\sum_{a\in\mathcal{A}}\left(f(s,a) - \lambda\log\Big(m(a)\Big) + \gamma\mathbb{E}_{s_1\sim\mathcal{T}(\cdot|s,a)}\Big[V_\lambda^*(s_1)\Big]\right)m(a)\right] \\ &= \lambda\max_{m\in\mathcal{P}(\mathcal{A})}\left[\frac{1}{\lambda}\sum_{a\in\mathcal{A}}\left(f(s,a) + \gamma\mathbb{E}_{s_1\sim\mathcal{T}(\cdot|s,a)}\Big[V_\lambda^*(s_1)\Big]\right)m(a) + \mathcal{H}(m)\right]. \end{aligned} \tag{3}$$

Observing that on the right hand side we are maximizing over a linear function in m, plus an entropy term, and applying [Dupuis and Ellis, 2011, Proposition 1.4.2], we have that for any $s \in S$

$$V_\lambda^*(s) = V_\lambda^{\pi_\lambda^*}(s) = \lambda\log\sum_{a\in\mathcal{A}} e^{\frac{1}{\lambda}\left(f(s,a)+\gamma\mathbb{E}_{s_1\sim\mathcal{T}(\cdot|s,a)}[V_\lambda^{\pi_\lambda^*}(s_1)]\right)} = \lambda\log\sum_{a\in\mathcal{A}}\exp\left(\frac{1}{\lambda}Q_\lambda^{\pi_\lambda^*}(s,a)\right), \tag{4}$$

and the maximum in (3) is achieved by the randomized policy $m(a) = \pi_\lambda^*(a|s)$, where

$$\pi_\lambda^*(a|s) = \frac{\exp\left(\frac{1}{\lambda}Q_\lambda^{\pi_\lambda^*}(s,a)\right)}{\sum_{a'\in\mathcal{A}}\exp\left(\frac{1}{\lambda}Q_\lambda^{\pi_\lambda^*}(s,a')\right)} \qquad \text{for } a\in\mathcal{A}. \tag{5}$$

From (4) we see that $\exp\left(V_\lambda^*(s)/\lambda\right) = \sum_{a\in\mathcal{A}}\exp\left(Q_\lambda^{\pi_\lambda^*}(s,a)/\lambda\right)$ and so we can write the optimal policy and value as

$$\begin{aligned} \pi_\lambda^*(a|s) &= \exp\left(\left(Q_\lambda^{\pi_\lambda^*}(s,a) - V_\lambda^*(s)\right)/\lambda\right) \\ &= \exp\left(\left(f(s,a) + \mathbb{E}_{s_1\sim\mathcal{T}(\cdot|s,a)}[\gamma V_\lambda^*(s_1) - V_\lambda^*(s)]\right)\right)/\lambda\Big), \end{aligned} \tag{6}$$

$$V_\lambda^*(s) = \lambda\log\sum_{a\in\mathcal{A}}\exp\left(Q_\lambda^{\pi_\lambda^*}(s,a)/\lambda\right). \tag{7}$$

From this analysis, we make the following observations regarding the regularized MDP:

- The optimal policy will select all actions in $\mathcal{A}$ with some positive probabilities.
- If λ is increased, this has the effect of 'flattening out' the choice of actions, as seen in the softmax function in (5). Conversely, sending $\lambda \to 0$ will result in a true maximizer being chosen, and the regularized problem degenerates to the classical MDP.
- Adding a constant to the reward does not change the policy.

In many modern approaches, one replaces dependence on the state with dependence on a space of 'features'. This has benefits when fitting a model, but does not significantly change the problem considered.

[4]Given the Markov structure there is no loss of generality when working with policies of feedback form.

2.2 Literature review

One of the earliest discussions of inverse reinforcement learning (IRL) in the context of machine learning can be found in Ng and Russell [2000]. Their method is to first identify a class of reward functions, for an IRL problem with finitely many states and actions, a deterministic optimal strategy, and the assumption that the reward function depends only on the state variable. Then, assuming the reward function is expressable in terms of some known basis functions in the state, a linear programming formulation for the IRL problem is presented, to pick the reward function that maximally differentiates optimal policy from the other policies. This characterization of reward functions demonstrates the general non-uniqueness of solutions to IRL problems.

In past two decades, there have been many algorithms tackling IRL problems. One significant category of algorithms arises from the maximum entropy approach to optimal control. In Ziebart [2010], IRL problems were linked with maximum causal entropy problems with statistical matching constraints. Similar models can be found in Abbeel and Ng [2004]; Ziebart et al. [2008], Levine et al. [2011] and Boularias et al. [2011]. A connection between maximum entropy IRL and GANs has been established in Finn et al. [2016a].

Fu et al. [2017] introduce an adversarial inverse RL algorithm that is able to recover the policy up to a constant, assuming the reward is only a function of the state and a certain decomposability condition on the dynamics holds, see Theorem C.1 in their paper. Ng et al. [1999] give a characterization of transformations of rewards which do not affect optimal policies for unregularized problems, closely related to our main result, we discuss this further in Remark 2 below. A key distinction with our approach is that we focus on the case when the transition probabilities are fixed, and we work with the entropy regularized MDP $\mathcal{M}_\lambda$

3 Analysis of inverse reinforcement learning

We now shift our focus to 'inverse' reinforcement learning, that is, the problem of inferring the reward function given observation of agents' actions.

Consider a discrete time, finite-state and finite-action MDP $\mathcal{M}_\lambda$, as described in Section 2. Suppose a 'demonstrator' agent acts optimally, and hence generates a *trajectory* of states and actions for the system

$$\tau = (s_1, a_1, s_2, a_2, ...).$$

We assume that it is possible for us to observe τ (over a long period), and seek to infer the reward f which the agent faces.

A first observation is that, assuming each state $s \in \mathcal{S}$ appears infinitely often in the sequence τ, and the agent uses a randomized feedback control $\pi_\lambda(a|s)$, it is possible to infer this control. A simple consistent estimator for the control is

$$(\pi_\lambda)_N(a|s) = \frac{\#\{a_t = a \text{ and } s_t = s; \quad t \leq N\}}{\#\{s_t = s; \quad t \leq N\}} \to \pi_\lambda(a|s) \text{ a.s. as } N \to \infty.$$

Similarly, assuming each state-action pair (s, a) appears infinitely often in τ, we can infer the controlled transition probabilities $\mathcal{T}(s'|s, a)$. A simple consistent estimator is given by

$$\mathcal{T}_N(s'|s, a) = \frac{\#\{s_t = s, a_t = a \text{ and } s_{t+1} = s'; \quad t \leq N\}}{\#\{s_t = s \text{ and } a_t = a; \quad t \leq N\}} \to \mathcal{T}(s'|s, a) \text{ a.s. as } N \to \infty.$$

If our agent is known to follow a regularized optimal strategy, as in (6), and we have a simple accessibility condition[5] on the underlying states, then every state-action pair will occur infinitely often in the resulting trajectory. Therefore, given sufficiently long observations, we will know the values of $\pi(a|s)$ and $\mathcal{T}(s'|s, a)$ for all $s, s' \in \mathcal{S}$ and $a \in \mathcal{A}$.

This leads, naturally, to an abstract version of the inverse reinforcement learning problem: Given knowledge of $\pi(a|s)$ and $\mathcal{T}(s'|s, a)$ for all $s, s' \in \mathcal{S}$ and $a \in \mathcal{A}$, and assuming π is generated by an

[5] In particular, for every pair of states s, s', there needs to exist a finite sequence $s = s_1, s_2, ..., s_n = s'$ of states and $a_1, ..., a_{n-1}$ of actions such that $\prod_{k=1}^{n-1} \mathcal{T}(s_{k+1}|s_k, a_k) > 0$. This is certainly the case, for example, if we assume $\mathcal{T}(s'|s, a) > 0$ for all $s, s' \in \mathcal{S}$ and $a \in \mathcal{A}$.

agent following an entropy-regularized MDP $\mathcal{M}_\lambda$, can we determine the initial reward function f that the agent faces?

As observed by Kalman [1964], for an unregularized controller the only thing we can say is that the observed controls are maximizers of the state-action value function, and not even that these maximizers are unique. Therefore, very little can be said about the underlying reward in the unregularized setting. Indeed, as already observed in Russell [1998] the problem of constructing a reward using state-action data is fundamentally ill-posed. One pathological case is to simply take f constant, so all actions are optimal. Alternatively, if we infer a unique optimal action $a^\star(s)$ for each s, we then could take $f(s, a^\star(s)) \in (0, \infty]$ and $f(s, a) = 0$ for $a \neq a^\star(s)$.

A popular approach to inverse RL problem follows the principle of Maximum Entropy introduced in Ziebart [2010] and further studied in Boularias et al. [2011], Abbeel and Ng [2004]. In MaxEntIRL, one assumes that trajectories are generated[6] with a law

$$\mathbb{P}(\tau) = \frac{\rho(s_0)}{Z} \prod_t \mathcal{T}(s_{t+1}|s_t, a_t) e^{\sum_t f(a_t, s_t)}$$

for a constant $Z > 0$. One then considers a parametric approximation of the reward, which can be inferred e.g. via likelihood estimation. Comparing with the distribution of trajectories from an optimal regularized agent, this approach implicitly assumes that $\pi(a|s) \propto \exp\{f(a, s)\}$. Comparing with (6), this is analogous to assuming the value function is a constant (from which we can compute Z) and $\lambda = 1$. As we shall see in Theorem 1, for inverse RL the value function can be chosen arbitrarily, demonstrating the consistency of this approach with our entropy-regularized agents.

3.1 Inverse Markov decision problem with finite state-action space

We consider a Markov decision problem as in Section 2. As discussed above, we assume that we have full knowledge of $\mathcal{S}, \mathcal{A}, \mathcal{T}, \gamma$, and of the regularization parameter λ and the entropy-regularized optimal control π_λ in (6), but not the reward function f.

Our first theorem characterizes the set of all reward functions f which generate a given control policy.

Theorem 1. *For a fixed policy $\bar{\pi}(a|s) > 0$, discount factor $\gamma < 1$, and an arbitrary choice of function $v : \mathcal{S} \to \mathbb{R}$, there is a unique corresponding reward function*

$$f(s, a) = \lambda \log \bar{\pi}(a|s) - \gamma \textstyle\sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) v(s') + v(s)$$

such that the MDP with reward f yields a value function $V_\lambda^{\pi_\lambda^*} = v$ and entropy-regularized optimal policy $\pi_\lambda^* = \bar{\pi}$.

As a consequence of this theorem, we observe that the choice of value function is not determined by the observed optimal policy, but can be chosen arbitrarily. We also see that the space of reward functions f consistent with a given policy can be parameterized by the set of value functions.

Remark 1. *A simple degrees-of-freedom argument gives this result intuitively. There are $n = |\mathcal{S}|$ possible states and $k = |\mathcal{A}|$ possible actions in each state, so the reward function can be described by a vector in $\mathbb{R}^{n \times k}$. From the policy, which satisfies $\sum_{a \in \mathcal{A}} \pi(a|s) = 1$ for all s, we observe $n \times (k-1)$ linearly independent values. Therefore, the space of consistent rewards has $n \times k - n \times (k-1) = n$ free variables, which we identify with the n values $\{v(s)\}_{s \in \mathcal{S}}$.*

Given Theorem 1, we see that it is not possible to fully identify the reward faced by a single agent, given only observations of their policy. Fundamentally, the issue is that the state-action value function Q combines both immediate rewards f with preferences v over the future state. If we provide data which allows us to disentangle these two effects, for example by considering agents with different discount rates or transition functions, then the true reward can be determined up to a constant, as shown by our next result. In order to clearly state the result, we give the following definition.

Definition 1. *Consider a pair of Markov decision problems on the same state and action spaces, but with respective discount rates $\gamma, \tilde{\gamma}$ and transition probabilities $\mathcal{T}, \tilde{\mathcal{T}}$. We say that this pair is*

[6] As discussed by Levine [2018], for deterministic problems this simplifies to $P(\tau) \propto e^{\sum_t f(a_t, s_t)}$, which is often taken as a starting point.

value-distinguishing *if, for functions* $w, \tilde{w} : \mathcal{S} \to \mathbb{R}$*, the statement*

$$w(s) - \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s,a) w(s') = \tilde{w}(s) - \tilde{\gamma} \sum_{s' \in \mathcal{S}} \tilde{\mathcal{T}}(s'|s,a) \tilde{w}(s') \text{ for all } a \in \mathcal{A}, s \in \mathcal{S} \tag{8}$$

implies at least one of w *and* $\tilde{w}$ *is a constant function.*

In this definition, note that constant functions $w, \tilde{w}$ are always solutions to (8), in particular for $c \in \mathbb{R}$ we can set $w \equiv c$ and $\tilde{w} \equiv (1-\gamma)c/(1-\tilde{\gamma})$. At the same time, this is a system of $|\mathcal{A}| \times |\mathcal{S}|$ equations in $2 \times |\mathcal{S}|$ unknowns, so the definition will hold provided our agents differ, and their actions have sufficiently varied impact on the resulting transition probabilities. In a linear-quadratic context, we will see that it is always enough to vary the discount rates (Corollary 1).

Theorem 2. *Suppose we observe the policies of two agents solving entropy-regularized MDPs, who face the same reward function, but whose discount rates or transition probabilities vary, such that their MDPs are value-distinguishing. Then the reward function consistent with both agents' actions either does not exist, or is identified up to addition of a constant.*

Given the addition of a constant to f does not affect the resulting policy (it simply increases the value function by a corresponding quantity), we cannot expect to do better than Theorem 2 without direct observation of the agent's rewards or value function in at least one state.

Remark 2. *Comparing with Ng et al. [1999] provides a useful insight to our result. In Ng et al. [1999] it is assumed that the rewards are of the form* $F(S_t, A_t, S_{t+1})$*; for a fixed MDP, this adds no generality, as we can write* $f(s,a) = \mathbb{E}[F(s,a,S_{t+1})|S_t = s, A_t = a]$*. Ng et al. [1999] show that, for any 'shaping potential'* $\Upsilon : \mathcal{S} \to \mathbb{R}$*, the reward* $\tilde{F} = F + \gamma\Upsilon(S_{t+1}) - \Upsilon(S_t)$ *yields the same optimal policies for* every *(unregularized) MDP, and these are the only perturbations with this property (cf. the role of* V *in* (6)*).*

In our results, we instead parameterize a family of costs f *in terms of the value function (Theorem 1), and show these are the only costs which lead to the given optimal policy* for a fixed (regularized) MDP. *Furthermore, we show that when we consider two value-distinguishing MDPs (Theorem 2), there is a unique reward (independent of* S_{t+1}*) which yields the correct optimal policies under both problems.*

3.2 A linear-quadratic-Gaussian problem

In this section, we will present the corresponding results for a class of one-dimensional linear-quadratic problems with Gaussian noise, ultimately inspired by Kalman [1964]. This simplified framework allows us to explicitly observe the degeneracy of inverse reinforcement learning, even if we add restrictions on the choice of value functions.

Optimal LQG control Suppose our agent seeks to control, using a real-valued process A_t a discrete time process with dynamics

$$S_{t+1} = (\bar{\mu} + \mu_s S_t + \mu_a A_t) + (\bar{\sigma} + \sigma_s S_t + \sigma_a A_t) Z_{t+1}$$

for constants $\bar{\mu}, \mu_s, \mu_a, \bar{\sigma}, \sigma_s, \sigma_a$. The innovations process Z is a Gaussian white noise with unit variance. Our agent uses a randomized strategy $\pi(a|s)$ to maximize the expectation of the entropy-regularized infinite-horizon discounted linear-quadratic reward:

$$\mathbb{E}\Big[\sum_{t=1}^{\infty} \gamma^t \Big(\int_{\mathbb{R}} f(S_t, a)\pi(a|s)da + \lambda \mathcal{H}(\pi(\cdot|S_t)) \Big)\Big]$$

where $f(s,a) = \alpha_{20}s^2 + \alpha_{11}sa + \alpha_{02}a^2 + \alpha_{10}s + \alpha_{01}a + \alpha_{00}$ and $\mathcal{H}(\pi) = -\int_{\mathbb{R}} \pi(a)\log(\pi(a)da$ is the Shannon entropy of π. We assume the coefficients of f are such that the problem is well posed (i.e. it is not possible to obtain an infinite expected reward).

Just as in the discrete state and action space setting, we can write down the state-action value function

$$Q^{\pi}_{\lambda}(s,a) = f(s,a) + \gamma \int_{\mathbb{R}} V^{\pi}_{\lambda}(s') \frac{1}{\sqrt{2\pi(\bar{\sigma} + \sigma_s s + \sigma_a a)^2}} \exp\Big(- \frac{(s' - \mu_s s - \mu_a a)^2}{2(\bar{\sigma} + \sigma_s s + \sigma_a a)^2} \Big) ds'. \tag{9}$$

Using this, the optimal policy and value function are given by

$$\pi^*_\lambda(a|s) = \exp\Big(\big(Q^{\pi^*_\lambda}_\lambda(s,a) - V^{\pi^*_\lambda}_\lambda(s)\big)/\lambda\Big), \tag{10}$$

$$V^*_\lambda(s) = V^{\pi^*_\lambda}_\lambda(s) = \lambda \log \int_{\mathbb{R}} \exp\Big(Q^{\pi^*_\lambda}_\lambda(s,a)/\lambda\Big) da\,. \tag{11}$$

What is particularly convenient about this setting is that Q is a quadratic in (s,a), V_λ is a quadratic in s, and $\pi^*_\lambda(\cdot|s)$ is a Gaussian density. In particular, the optimal policy is of the form

$$\begin{aligned}\pi^*_\lambda(a|s) &= \frac{1}{\sqrt{2\pi\lambda k_3}} \exp\left\{-\frac{(a-k_1 s - k_2)^2}{2\lambda k_3}\right\} \\ &= \exp\left\{-\frac{1}{\lambda}\left[\frac{a^2 - 2(k_1 s + k_2)a}{2k_3} + \frac{(k_1 s + k_2)^2}{2k_3} + \frac{\lambda}{2}\log\left(2\pi k_3 \lambda\right)\right]\right\}\end{aligned} \tag{12}$$

for some constants $k_1, k_2 \in \mathbb{R}, k_3 > 0$, which can be determined[7] in terms of the known parameters $\mu_a, \mu_s, \sigma, \lambda$ and the parameters of the reward function $\{\alpha_{ij}\}_{i+j\leq 2}$.

Theorem 3. *Consider an agent with a policy of the form* (12)*. Suppose we also know that the value function V is a quadratic (or, equivalently, that the reward function is a quadratic in (s,a)). The space of rewards consistent with this policy is given by:*

$$\begin{aligned}\mathbb{F} = \Bigg\{ & f(s,a) = a_{20}s^2 + a_{11}sa + a_{02}a^2 + a_{10}s + a_{01}a + a_{00} \Bigg| \\ & (a_{20},\, a_{11},\, a_{02}) = \Big(\frac{-k_1^2}{2k_3},\, \frac{k_1}{k_3},\, \frac{-1}{2k_3}\Big) - \beta_2\Big(\gamma(\mu_s^2+\sigma_s^2) - 1,\, 2\gamma(\mu_s\mu_a + \sigma_s\sigma_a),\, \gamma(\mu_a^2+\sigma_a^2)\Big), \\ & (a_{10},\, a_{01}) = \Big(\frac{-k_1 k_2}{k_3},\, \frac{k_2}{k_3}\Big) - \beta_2\Big(2\gamma(\bar\mu\mu_s + \bar\sigma\sigma_s), 2\gamma(\bar\mu\mu_a + \bar\sigma\sigma_a)\Big) - \beta_1\Big(\gamma\mu_s + 1,\, \gamma\mu_a\Big), \\ & a_{00}, \beta_2, \beta_1 \in \mathbb{R}\Bigg\}.\end{aligned} \tag{13}$$

As in Theorem 1, we see that the inverse reinforcement learning problem only defines the rewards up to the choice of value function, which is arbitrary; the restriction to quadratic rewards or values simply reduces our problem to the smaller range of rewards determined by the three coefficients in the quadratic V.

The following theorem gives the linear-quadratic version of Theorem 2. As our agents' actions have a linear effect on the state variable, this leads to a particularly simple set of conditions for identifiability of the reward, given observation of two agents' policies.

Theorem 4. *Suppose we now have two agents, who are both following their respective optimal controls of the form* (12)*, for the same reward function, but disagree on some combination of the dynamics and discount rate. We write*

$$x_1 = \Big(\gamma\mu_s + 1,\, \gamma\mu_a\Big) \qquad \textit{and} \qquad x_2 = \Big(\gamma(\mu_s^2+\sigma_s^2) - 1,\, 2\gamma(\mu_s\mu_a + \sigma_s\sigma_a),\, \gamma(\mu_a^2+\sigma_a^2)\Big),$$

giving us two pairs of vectors (x_1, x_2) (for the first agent) and $(\tilde x_1, \tilde x_2)$ (for the second agent). We assume we know these vectors for each agent. The quadratic reward function f consistent with both agents' policies, if it exists, is uniquely identified up to the addition of a constant shift, if (and only if)

$$\frac{x_1}{\|x_1\|} \neq \frac{\tilde x_1}{\|\tilde x_1\|} \qquad \textit{and} \qquad \frac{x_2}{\|x_2\|} \neq \frac{\tilde x_2}{\|\tilde x_2\|}.$$

Due to the simplicity of the characterization in Theorem 4, we can easily see that it is enough to observe two agents using different discount rates.

[7] The explicit formulae for k_1, k_2 and k_3, and the coefficients of the value function, can be obtained by equating the coefficients of f with the values obtained in (16). Under the assumption that the optimal control problem is well posed, this has a solution with $k_3 > 0$.

Corollary 1. *Suppose we observe two agents, each using optimal policies of the form* (12)*, for the same dynamics and rewards, but different discount rates. Then the underlying quadratic reward consistent with both agents' policies is identifiable up to a constant.*

Proof. Simply observe that the value of γ introduces a non-scaling change in the vectors x_1, x_2 defined in Theorem 4. □

4 Numerical examples of inverse reinforcement learning

In this section, we present a regularized MDP as in Section 3.1 to illustrate numerically the identifiability issue associated with inverse RL. In particular, we consider a state space $\mathcal{S}$ with 10 states and an action space $\mathcal{A}$ with 5 actions, with $\lambda = 1$. We compute optimal policies as in Section 2 and reconstruct the underlying rewards. As discussed in Section 3 the optimal policies and the transition kernel can be inferred from state-action trajectories, so will assumed known. We identify the state and action spaces with the basis vectors in $\mathbb{R}^{10}$ and $\mathbb{R}^5$ respectively, so can write $f(a, s) = a^\top Rs$ for the reward function, and $\mathcal{T}(s'|s, a) = s^\top P_a(s')$ for the transition function. The true reward R_{tr} and transition matrices $\{P_a\}_{a\in\mathcal{A}}$ are randomly generated and fixed; see Figures 4 and 5.

4.1 Non-uniqueness of infinite-sample IRL

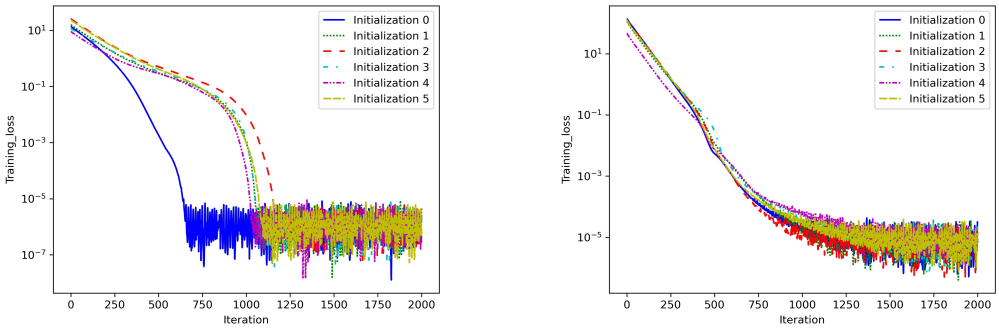
We first look at inverse RL starting from a single optimal policy π_1 with discount factor $\gamma_1 = 0.95$. We represent π_1 as a matrix Π_1 in $\mathbb{R}^{5\times 10}$, where each column gives the probabilities of each action when in the corresponding state.

To solve the inverse RL problem, we numerically find R, v to minimize the loss

$$L_{\text{sing}}(R, v) = \sum_{a\in\mathcal{A}} \sum_{s\in\mathcal{S}} \left[a^\top \Pi_1 s - \exp\left\{a^\top Rs + \gamma_1 s^\top P_a v - v^\top s\right\}\right]^2 .$$

An Adam optimizer is adopted with $\alpha = 0.002$, $(\beta_1, \beta_2) = (0.5, 0.9)$ with overall 2000 minimization steps. The experiments are conducted over 6 different random initializations, sampled from the same distribution as was used to construct the ground truth model. The training loss L_{sing} decays rapidly to close to 0, as shown in Figure 1a. This indicates that, after the minimization procedure comes to an end, the learnt reward matrix $\hat{R}$ reveals a corresponding optimal policy $\hat{\Pi}_1$ close to the true optimal policy Π_1; see also Figure 8 for a direct comparison.

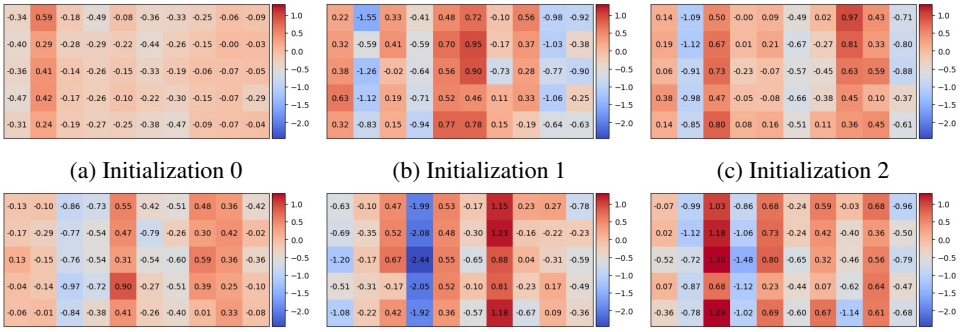
However, when comparing the learnt reward $\hat{R}$ and the underlying reward R_{tr}, as in Figures 2 and 6a, as well as the comparison between the corresponding value vectors as in Figure 6b, we can see that the true reward function R_{tr} has not been correctly inferred. Here this is not an issue of statistical error, as we assume full information on the optimal policy and the Markov transition kernel.



(a) Loss with one optimal policy, under γ_1

(b) Loss with two optimal policies, under γ_1 and γ_2

Figure 1: Training Losses



(a) Initialization 0 (b) Initialization 1 (c) Initialization 2

(d) Initialization 3 (e) Initialization 4 (f) Initialization 5

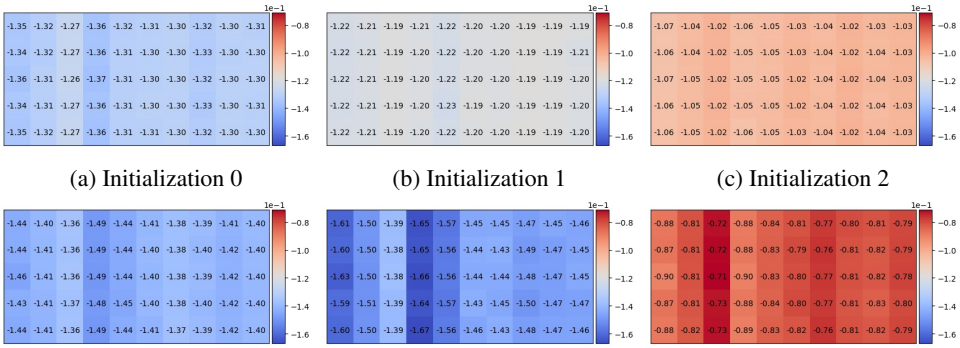
Figure 2: Learning from one optimal policy, under γ_1: difference $\hat{R} - R_{\text{tr}}$ between learnt and true reward matrices

4.2 Uniqueness of IRL with multiple discount rates

We now demonstrate that the issue of identifiability can be resolved if there is additional information on an optimal policy under the same reward matrix R_{tr} but different environment. Here, we assume we are given the policy Π_1 optimal with discount factor $\gamma_1 = 0.95$, and the policy Π_2 optimal with discount factor $\gamma_2 = 0.25$. Correspondingly, the loss function for the minimization is adjusted to

$$L_{\text{doub}}(R, v_1, v_2) = \frac{1}{2} \sum_{a \in \mathcal{A}} \sum_{s \in \mathcal{S}} \left[a^\top \Pi_1 s - \exp\left\{ a^\top R s + \gamma_1 s^\top P_a v_1 - v_1^\top s \right\} \right]^2 + \frac{1}{2} \sum_{a \in \mathcal{A}} \sum_{s \in \mathcal{S}} \left[a^\top \Pi_2 s - \exp\left\{ a^\top R s + \gamma_2 s^\top P_a v_2 - v_2^\top s \right\} \right]^2 .$$

An Adam optimizer is adopted with $\alpha = 0.005$, $(\beta_1, \beta_2) = (0.5, 0.9)$ with overall 2000 minimization steps. With the same set of 6 random initializations for the minimization procedure, the training loss L_{doub} also decays rapidly to close to 0. This again suggests that the learnt reward matrix $\tilde{R}$ can lead to policies $\tilde{\Pi}_1$ and $\tilde{\Pi}_2$, each optimal when using the corresponding discount factor γ_1 and γ_2, that are close to the given policies Π_1 and Π_2; see Figures 9 and 10. What differs from the single optimal policy case is that, with the additional information Π_2, we are able to consistently recover R_{tr} up to a constant shift; see Figures 3 and 7. Some numerical error remains, due to the optimization algorithm used, as seen by the fact the graphs in Figure 3 do still vary, and the error in the value function v_1 in 7(a). Nevertheless, the errors are an order of magnitude less than was observed in Figure 6 when using observations under a single discount rate.



(a) Initialization 0 (b) Initialization 1 (c) Initialization 2

(d) Initialization 3 (e) Initialization 4 (f) Initialization 5

Figure 3: Learning from two optimal policies, under γ_1 and γ_2: difference $\tilde{R} - R_{\text{tr}}$ between learnt and true R matrices. Note scale of 10^{-1}.

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.

Dimitir P Bertsekas and Steven Shreve. *Stochastic optimal control: the discrete-time case*. 2004.

Abdeslam Boularias, Jens Kober, and Jan Peters. Relative entropy inverse reinforcement learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 182–189. JMLR Workshop and Conference Proceedings, 2011.

Stephen Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. *Linear matrix inequalities in system and control theory*. SIAM, 1994.

Paul Dupuis and Richard S Ellis. *A weak convergence approach to the theory of large deviations*, volume 902. John Wiley & Sons, 2011.

Chelsea Finn, Paul Christiano, Pieter Abbeel, and Sergey Levine. A connection between generative adversarial networks, inverse reinforcement learning, and energy-based models. *arXiv preprint arXiv:1611.03852*, 2016a.

Chelsea Finn, Sergey Levine, and Pieter Abbeel. Guided cost learning: Deep inverse optimal control via policy optimization. In *International conference on machine learning*, pages 49–58, 2016b.

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

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*, pages 1352–1361. PMLR, 2017.

R. E. Kalman. When Is a Linear Control System Optimal? *Journal of Basic Engineering*, 86(1): 51–60, 03 1964.

Ralph L. Keeney and Howard Raiffa. *Decisions with Multiple Objectives: Preferences and Value Trade-Offs*. Wiley, 1976.

Sergey Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*, 2018.

Sergey Levine, Zoran Popovic, and Vladlen Koltun. Nonlinear inverse reinforcement learning with gaussian processes. *Advances in neural information processing systems*, 24:19–27, 2011.

Robert E. Lucas. Econometric policy evaluation: A critique. *Carnegie-Rochester Conference Series on Public Policy*, 1:19–46, 1976.

Andrew Y Ng and Stuart Russell. Algorithms for inverse reinforcement learning. In *Proceedings of Seventeenth International Conference on Machine Learning*. Citeseer, 2000.

Andrew Y. Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *In Proceedings of the Sixteenth International Conference on Machine Learning*, pages 278–287. Morgan Kaufmann, 1999.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Stuart Russell. Learning agents for uncertain environments. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pages 101–103, 1998.

Thomas J Sargent. Estimation of dynamic labor demand schedules under rational expectations. *Journal of Political Economy*, 86(6):1009–1044, 1978.

Brian D Ziebart. *Modeling Purposeful Adaptive Behavior with the Principle of Maximum Causal Entropy*. PhD thesis, Carnegie Mellon University, 2010.

Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] This work connects with well known concerns in economics, which we mention. As a work in the theory of RL, it has limited potential for direct negative societal impacts.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] All results are rigorously stated
   (b) Did you include complete proofs of all theoretical results? [Yes] The main results' proofs are given in appendix A due to length.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] The code and instructions was included in the supplemental material. We did not work with any particular dataset.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] We specified training details in the main text.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] For the purpose of demonstration, instead of reporting error bars, we included training and test errors with respect to each random seed.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] We did not include this information in the main text or the supplemental material. All the experiments have been conducted on Google Research Colaboratory with GPU as hardware accelerator. The experiment in Section 4 took 2 minutes and 14 seconds in total for all 6 random initializations; for each random initialization in the guided cost learning it took 7 hours and 32 minutes to finish and the experiment was repeated 6 times with different initializations.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]