# Inverse Reinforcement Learning in General Environments

Michael Cohen

July 2019

## 1 Introduction

The eventual aim of IRL is to understand human goals. However, typical algorithms for IRL assume the environment is finite-state Markov, and it is often left unspecified how raw observational data would be converted into a record of human actions, alongside the space of actions available. For IRL to learn human goals, the AI has to consider general environments, and it has to have a way of identifying human actions. Lest these extensions appear trivial, I consider one of the simplest proposals, and discuss some difficulties that might arise.

## 2 Dedicated Channel for Human Actions

In this proposed setup, we sit the human demonstrator at a computer, and at certain timesteps, she takes over for the AI. She sees the observations that the AI sees. She picks an action from the action space of the AI, and the action is implemented in the real world in the same way that it would be if the AI had picked it. Later, this action is annotated for the AI as having originated from the human demonstrator. This takes care of the problem of recognizing a human action from observational data.

Formally, I consider an infinite interaction history, with an action and an observation at each timestep from a finite action space $\mathcal{A}$ and a finite observation space $\mathcal{O}$. The action and observation at timestep $t$ are denoted $a_t$ and $o_t$. The indicator for whether $a_t$ was selected by the human demonstrator takes a value of 0 or 1, and is denoted $d_t$ ($d$ for demonstrator).

## 3 Beyond Finite-State Markov

An "environment" samples each successive observation, conditionally on all previous actions and observations. Of course, it may not depend on all such actions and observations. Thus, an environment $\nu$ maps $(\mathcal{A} \times \mathcal{O})^* \times \mathcal{A}$ to a probability distribution over $\mathcal{O}$. ($\mathcal{X}^* = \bigcup_{i=0}^{\infty} \mathcal{X}^i$). If this probability distribution is computable, I call it a computable environment.

Typically, in IRL, the demonstrator's utility function is simply a function of the state. Since my abstraction does not straightforwardly include a state, I instead say it is a function over the entire interaction history–that is the utility at time $t$ may depend on all actions and observations $a_k$, $o_k$ for $k \leq t$. Thus, $U : (\mathcal{A} \times \mathcal{O})^* \to [0, 1]$. I restrict the utility to $[0, 1]$ to keep expectations from diverging, as is typical in reinforcement learning. I also restrict consideration to computable utilty functions. Let $h_{\leq t}$ denote the interaction history up to and including timestep $t$; that is: $a_0 o_0 a_1 o_1 ... a_t o_t$. $h_{<t} a_t o_t$ denotes the same.

This completes the extension from finite-state Markov environments with utility functions dependant on the state to general computable environments with utility functions dependant on all available information.

# 4 General Inverse Reinforcement Learning

The AI begins with a prior over the set of all computable environments and the set of all computable utility functions $\mathcal{M} \times \mathcal{U}$. Let $(\mu, U)$ be the true environment and the true utility function, respectively.

Let $Q^*_\mu(h_{<t} a_t) = \mathbb{E}_{o_t \sim \mu(\cdot | h_{<t} a_t)}[U(h_{<t} a_t o_t) + \gamma V^*_\mu(h_{<t} a_t o_t)]$, and let $V^*_\mu(h_{\leq t}) = \max_{a \in \mathcal{A}} Q^*_\mu(h_{\leq t} a)$. The AI supposes that all observations are sampled from $\mu$, and that when $d_t = 1$, $a_t \in \operatorname{argmax}_{a \in \mathcal{A}} Q^*_\mu(h_{<t} a)$, where the argmax returns the set of elements tied for best. (It is now common in IRL to suppose instead that $a_t \sim \operatorname{softmax}^{(\beta)}_{a \in \mathcal{A}} Q^*_\mu(h_{<t} a)$,[1] but I set this relatively simple extension aside for now). Let $d_t \sim \text{Bernoulli}(\varepsilon)$. Since $\mu$ and $U$ (alongside $\gamma$ and the AI's own policy) induce a probability measure over an infinite interaction history, the AI can perform Bayesian updates to generate posterior beliefs about $\mu$ and $U$, as the observations come in. Combined with some discount rate of its own, the AI can straightforwardly maximize expected utility.

# 5 A Dangerous Failure Mode

Here's one hypothesis about the utility function that the evidence will support. Let's call the human demonstrator Eva. The utility is 1 if the button Eva would have pressed is pressed, and the utility is 0 otherwise. (For simplicity, I assume Eva is a deterministic system; if Eva is stochastic, similar examples get more complicated, but the problems they pose don't go away). How might the AI behave in light of this hypothesis? It might take over the world, kill everyone, and send in a robot to the room where Eva was, to press the button Eva would have pressed until the end of time. (Note that doing nothing isn't good enough: Eva doesn't press the button in non-Eva-controlled timesteps, Eva might fall ill or be replaced, etc. In these cases, the button Eva would have pressed doesn't get pressed.)

[1] $\operatorname{softmax}^{(\beta)}_{a \in \mathcal{A}} f(a)$ is a probability distribution over $\mathcal{A}$ with $p(a) \propto e^{\beta f(a)}$.

## 6 Take-Aways

I won't go into more detail trying to analyze the likelihood that dangerous hypotheses about the true utility function like the one above will dominate the benign hypotheses which we would like the AI to learn. I won't try to enumerate other dangerous hypotheses that observational evidence would never discredit, or search for the least esoteric such one. My main point is that IRL, as it is typically described, *feels* nearly complete: just throw in a more advanced RL algorithm as a subroutine and some narrow-AI-type add-on for identifying human actions from a video feed, and voilá, we have a superhuman human-helper. The sort of failure mode described here is only conceivable once we try to fill in the last few details of the proposal concretely.

In fact, the failure mode seems to be exactly the same as the wireheading failure mode in straight reinforcement learning; another way of understanding wireheading is that from all possible utility functions which could be computing the reward, the AI hypothesizes that it's just a function of what the human operator types in, so it sends in a robot to replace the human operator. If IRL doesn't even replace the central failure mode of RL with a new failure mode, that might lead us to wonder whether the IRL proposal accomplishes anything from a safety perspective.

Of the many proposals for safe AGI which are in a similar state of completion as IRL, my intuition is that most fall prey to similarly pernicious failure modes. I don't want to dismiss partially complete proposals for safe AGI–they are necessary inspiration for complete proposals. But maybe we could be spending more effort trying to follow through to fully specified proposals which we can properly put through the gauntlet.