Inverse Reinforcement Learning Under Noisy Observations (Extended Abstract)

Shervin Shahryari
Institute for Artificial Intelligence
University of Georgia
Athens, GA 30602
shervinf66@gmail.com

Prashant Doshi
THINC Lab, Dept. of Computer Science
University of Georgia
Athens, GA 30602
pdoshi@cs.uga.edu

ABSTRACT

We consider the problem of performing inverse reinforcement learning when the trajectory of the expert is not perfectly observed by the learner. Instead, noisy observations of the trajectory are available. We generalize the previous method of expectation-maximization for inverse reinforcement learning, which allows the trajectory of the expert to be partially hidden from the learner, to incorporate an observation model.

1. INTRODUCTION

Inverse reinforcement learning (IRL) [1, 2] seeks to find the observed expert's rewards from passive observations, and usually models the expert as a Markov decision process (MDP). However, past methods predominantly assume that the learner has perfect observability of the expert's trajectory consisting of a sequence of state and actions pairs [2, 3]. In this paper, we relax this assumption – the learner is not able to observe the expert's states and actions directly. Consider the scenario introduced by Bogert and Doshi [4], in which a hidden robot seeks to learn a patroller's behavior in order to penetrate the patrol without being spotted. Because the learner is hidden, it's field of view is limited and it cannot see the patroller during much of the patrol. But, it may hear the patroller's movement sound at all times, which is useful for estimating the patroller's states and actions using an observation model.

Consequently, a sequence of observations is provided to the learner. Clearly, the fact that sensors tend to be noisy motivates the need for IRL under noisy observations. We generalize the previous method of expectation-maximization for IRL [5], which allows the trajectory of the expert to be partially hidden from the learner, to operate in situations where observations by the learner are noisy. The generalization incorporates an observation function into IRL, which models observations as a function of both state and action. Importantly, this generalization can enable the learner to fuse data from different sensors with different levels of noise.

2. ROBUST IRL

Because sensors tend to be noisy, a robotic learner may not perceive the expert's state-action pairs perfectly. We consider the problem where sensory information is noisy or information comes from different sensors with differing levels of noise, and a model of the observation noise is available. A straightforward way to continue using previous methods such as maximizing entropy [6] is to simply pick the trajectory that is most likely given the sequence of observations. However, it is easy to see that this method may not be appropriate when noise levels are considerable. Thus, we need a principled way of performing IRL with noise.

2.1 Hidden MDP

As a first step, we generalize the model that the learner ascribes to the expert in order to include observations. Kitani et al. [7] introduced the hidden MDP (hMDP) model, as depicted in Fig. 1 using an influence diagram, which is appropriate here. While observations in the previous hMDP were influenced by the state only, we extend the model so that both state and action at a time step affect the observation in that step. The conditional probability table of the observation node in each time slice of the network is the same, and it represents the observation model.

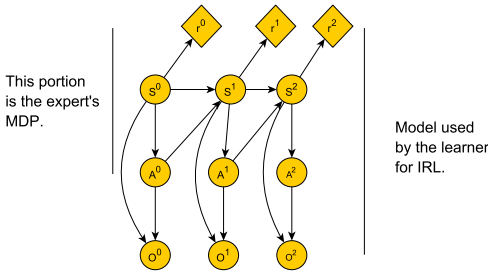

Figure 1: In hMDP, the state and action are hidden from the learner but an observation of the state and action is available to the learner at each time step.

2.2 Formulation

Let the learner receive a sequence of observations of length N, $\vec{\omega} \coloneqq (o^1, o^2, ..., o^N)$, instead of the expert's trajectory τ. A stochastic observation model, $Pr(o|s,a)$ (all occurring in the same decision epoch), that captures its sensor noise is available to the learner. Note that $Pr(\vec{\omega}|\tau) = \prod_{i=1}^{N} Pr(o^i|\langle s,a\rangle^i)$ using chain rule and the conditional independence of an observation given the current state-action pair from other past pairs.

One may utilize the observation model to pick the most likely state-action pair at time step i while disregarding information from previous time steps. However, this method is naive because it disregards the effect of the transition function and the policy of the expert in getting information about the expert's true trajectory. In the context of these limitations, we present a revised formulation of the maximum entropy IRL that takes an expectation jointly over the trajectories (hidden data) and the sequence of observations. As we

Appears in: *Proc. of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*

show below, this allows considering the effect of the transition function and expert's policy in constructing a distribution over possible trajectories. Let $\vec{\omega} \in \Omega$, then

$$\max_{\Delta} \left(-\sum_{\vec{\omega},\tau} Pr(\vec{\omega},\tau)\, log Pr(\vec{\omega},\tau) \right)$$
$$\textbf{subject to} \quad \sum_{\vec{\omega},\tau} Pr(\vec{\omega},\tau) = 1$$
$$\sum_{\vec{\omega}\in\Omega} \sum_{\tau\in\mathcal{T}} Pr(\vec{\omega},\tau) \sum_{\langle s,a\rangle\in\tau} \phi_k(s,a) = \hat{\phi}_k \quad \forall k \tag{1}$$

Here, the joint probability is obtained as,

$$Pr(\vec{\omega},\tau) = Pr(\vec{\omega}|\tau) Pr(\tau) \tag{2}$$

where, $Pr(\tau) = Pr(s_1) \prod_{i=2}^{N} Pr(a_{i-1}|s_{i-1})\ Pr(s_i|s_{i-1}, a_{i-1})$. Let $\tilde{\Omega}$ be the set of received observation sequences of length N in the demonstration. The empirical feature expectation $\hat{\phi}_k$ is now obtained as,

$$\hat{\phi}_k = \frac{1}{|\tilde{\Omega}|} \sum_{\vec{\omega}\in\tilde{\Omega}} \sum_{\tau\in T} Pr(\tau|\vec{\omega}) \sum_{\langle s,a\rangle\in\tau} \phi_k(s,a)$$

Of course, $Pr(\tau|\vec{\omega}) \propto Pr(\vec{\omega},\tau)$, and latter is given by Eq. 2.

We take the logical next step of applying Lagrangian relaxation to the nonlinear program of (1), by bringing the parameterized constraints into the objective function, which gives $\mathcal{L}(Pr(\vec{\omega},\tau),\theta,\eta)$. Optimizing this Lagrangian for $Pr(\vec{\omega},\tau)$ gives us,

$$\frac{\partial \mathcal{L}}{\partial Pr(\vec{\omega},\tau)} = -log\ Pr(\vec{\omega},\tau) - 1 + \sum_{k=1}^{K} \theta_k \sum_{\langle s,a\rangle\in\tau} \phi_k(s,a) + \sum_{k=1}^{K} \theta_k$$
$$\left(\sum_{\vec{\omega}\in\tilde{\Omega}} \tilde{Pr}(\vec{\omega}) \frac{\sum_{\tau'\in\mathcal{T}} \left[\sum_{\langle s,a\rangle\in\tau'} \phi_k(s,a) - \sum_{\langle s,a\rangle\in\tau} \phi_k(s,a) \right] Pr(\vec{\omega}',\tau')}{Pr(\vec{\omega})^2} \right)$$
$$+ \eta$$

where $\vec{\omega}'$ and τ' are some other observation sequence and trajectory. Rather than setting the above complicated partial derivative to 0 and finding the optima, Wang et al. [8] and recently Bogert et al. [5], utilize its approximation:

$$\frac{\partial \mathcal{L}}{\partial Pr(\vec{\omega},\tau)} \approx -log\ Pr(\vec{\omega},\tau) - 1 + \sum_{k=1}^{K} \theta_k \sum_{(s,a)\in\tau} \phi_k(s,a) + \eta \tag{3}$$

Setting Eq. 3 to zero gives us a log linear approximation,

$$Pr(\vec{\omega},\tau) \approx \frac{e^{\sum_{k=1}^{K} \theta_k \sum_{\langle s,a\rangle\in\tau} \phi_k(s,a)}}{\eta(\theta)} \tag{4}$$

where $\eta(\theta)$ is the normalizing constant. By plugging the above result into $\mathcal{L}(Pr(\vec{\omega},\tau),\theta,\eta)$ we obtain the dual:

$$\mathcal{L}^{dual}(\theta) = log\ \eta(\theta) - \frac{1}{|\tilde{\Omega}|} \sum\nolimits_{k=1}^{K} \theta_k \sum\nolimits_{\vec{\omega}\in\tilde{\Omega}} \sum\nolimits_{\tau\in T} Pr(\tau|\vec{\omega})$$
$$\times \sum\nolimits_{(s,a)\in\tau} \phi_k(s,a) \tag{5}$$

2.3 Expectation-Maximization

Because of the presence of the conditional $Pr(\tau|\vec{\omega})$ in Eq. 5, we cannot use the usual exponentiated gradient descent to obtain the optimal value of the parameter vector. Analogously to Bogert et al., which extends Wang et al. [8], we develop an iterative EM approach for solving the above dual Lagrangian. As the first step toward the EM, we begin with establishing the log likelihood of feature weights.

$$\begin{aligned} LL(\boldsymbol{\theta}|\tilde{\Omega}) &= \log \prod\nolimits_{\vec{\omega}\in\tilde{\Omega}} Pr(\vec{\omega};\boldsymbol{\theta})^{\tilde{Pr}(\vec{\omega})} \\ &= \sum\nolimits_{\vec{\omega}\in\tilde{\Omega}} \tilde{Pr}(\vec{\omega}) \log Pr(\vec{\omega};\boldsymbol{\theta}) \sum\nolimits_{\tau\in\mathcal{T}} Pr(\tau|\vec{\omega};\boldsymbol{\theta}) \\ &= \sum\nolimits_{\vec{\omega}\in\Omega} \tilde{Pr}(\vec{\omega}) \sum\nolimits_{\tau\in\mathcal{T}} Pr(\tau|\vec{\omega};\theta) log Pr(\vec{\omega};\boldsymbol{\theta}) \end{aligned}$$

Rewriting $Pr(\vec{\omega};\boldsymbol{\theta})$ as $\frac{Pr(\vec{\omega},\tau;\boldsymbol{\theta})}{Pr(\tau|\vec{\omega};\boldsymbol{\theta})}$ in the last step we get,

$$\begin{aligned} LL(\boldsymbol{\theta}|\tilde{\Omega}) &= \sum_{\vec{\omega}\in\tilde{\Omega}} \tilde{Pr}(\vec{\omega}) \sum_{\tau\in\mathcal{T}} Pr(\tau|\vec{\omega};\boldsymbol{\theta}) \log \frac{Pr(\vec{\omega},\tau;\boldsymbol{\theta})}{Pr(\tau|\vec{\omega};\boldsymbol{\theta})} \\ &= \sum_{\vec{\omega}\in\Omega} \tilde{Pr}(\vec{\omega}) \sum_{\tau\in\mathcal{T}} Pr(\tau|\vec{\omega};\boldsymbol{\theta}) (\log\ Pr(\vec{\omega},\tau;\boldsymbol{\theta}) \\ &\quad - \log\ Pr(\tau|\vec{\omega};\boldsymbol{\theta})) \end{aligned} \tag{6}$$

Reformulating the likelihood as $Q(\boldsymbol{\theta},\boldsymbol{\theta}^i) + C(\boldsymbol{\theta},\boldsymbol{\theta}^i)$ where,

$$Q(\boldsymbol{\theta},\boldsymbol{\theta}^i) = \sum_{\vec{\omega}\in\tilde{\Omega}} \tilde{Pr}(\vec{\omega}) \sum_{\tau\in\mathcal{T}} Pr(\tau|\vec{\omega};\boldsymbol{\theta}^i) \log Pr(\vec{\omega},\tau;\boldsymbol{\theta})) \tag{7}$$

and,

$$C(\boldsymbol{\theta},\boldsymbol{\theta}^i) = -\sum_{\vec{\omega}\in\tilde{\Omega}} \tilde{Pr}(\vec{\omega}) \sum_{\tau\in\mathcal{T}} Pr(\tau|\vec{\omega};\boldsymbol{\theta}^i) \log(Pr(\tau|\vec{\omega};\boldsymbol{\theta}))$$

Replacing $Pr(\vec{\omega},\tau;\boldsymbol{\theta})$ in Eq. 7 with Eq. 4, we obtain,

$$\begin{aligned} Q(\boldsymbol{\theta},\boldsymbol{\theta}^i) = &-\log \eta(\theta) - \frac{1}{|\tilde{\Omega}|} \sum_{k=1}^{K} \theta_k \\ &+ \sum_{\vec{\omega}\in\tilde{\Omega}} \sum_{\tau\in\mathcal{T}} Pr(\tau|\vec{\omega};\theta^i) \sum_{\langle s,a\rangle\in\tau} \phi_k(s,a)) \end{aligned} \tag{8}$$

Notice that the Q function is the negative of the dual presented in Eq. 5. Therefore, maximizing the Q function is equivalent to minimizing the dual. Using these facts, we may reformulate the original problem stated in (1) as follows.

In the **E-step** we use the parameter $\boldsymbol{\theta}^i$ from the previous iteration to calculate the feature expectation of the expert.

$$\hat{\phi}_k^{\tau|\vec{\omega},i} = \sum_{\vec{\omega}\in\tilde{\Omega}} \tilde{Pr}(\vec{\omega}) \sum_{\tau\in\mathcal{T}} Pr(\tau|\vec{\omega};\boldsymbol{\theta}^i) \sum_{\langle s,a\rangle\in\tau} \phi_k(s,a) \tag{9}$$

where $Pr(\tau|\vec{\omega};\boldsymbol{\theta}^i) \propto Pr(\tau,\vec{\omega};\boldsymbol{\theta}^i)$ computed as in Eq. 4.

In the **M-Step**, we utilize the empirical feature expectation that has been calculated in the E-Step to obtain $\boldsymbol{\theta}$. Specifically, the computed $\hat{\phi}_k^{\tau|\vec{\omega},i}$ forms the right-hand side of the constraint of the program given in (1). The resulting program is easier to solve because the available feature expectation value is treated as a constant, thereby considerably simplifying the Lagrangian relaxation.

We iterate over the E- and M-steps until the parameter vector $\boldsymbol{\theta}$ stops changing. Notice that the E-step involves finding the distribution over each trajectory for each observed sequence. This is computationally expensive because the space of all trajectories is large and grows exponentially in the length.

REFERENCES

[1] Stuart Russell. Learning agents for uncertain environments (extended abstract). In *Eleventh Annual Conference on Computational Learning Theory*, pages 101–103, 1998.

[2] Andrew Y Ng and Stuart J Russell. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning (ICML)*, pages 663–670, 2000.

[3] Pieter Abbeel and Andrew Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Twenty-first International Conference on Machine Learning (ICML)*, page 1, 2004.

[4] Kenneth Bogert and Prashant Doshi. Multi-robot inverse reinforcement learning under occlusion with interactions. In *International Conference on Autonomous Agents and Multi-agent Systems (AAMAS)*, pages 173–180, 2014.

[5] Kenneth Bogert, Jonathan Feng-Shun Lin, Prashant Doshi, and Dana Kulic. Expectation-maximization for inverse reinforcement learning with hidden data. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1034–1042, 2016.

[6] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *AAAI Conference on Artificial Intelligence*, pages 1433–1438, 2008.

[7] Kris M Kitani, Brian D Ziebart, James Andrew Bagnell, and Martial Hebert. Activity forecasting. In *European Conference on Computer Vision (ECCV)*, pages 201–214. Springer, 2012.

[8] Shaojun Wang, Dale Schuurmans, and Yunxin Zhao. The latent maximum entropy principle. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 6(2):8, 2012.