OptionGAN: Learning Joint Reward-Policy Options Using Generative Adversarial Inverse Reinforcement Learning

Peter Henderson[1] *, Wei-Di Chang[1], Pierre-Luc Bacon[1], David Meger[1], Joelle Pineau[1], and Doina Precup[1]

[1]McGill University *peter.henderson@mail.mcgill.ca

Introduction

Finding a generalized reward function for noisy demonstrations is difficult...

- Imagine seeing many people walking and trying to find the underlying reward, not every demonstration will have the same reward function
- Learning a reward function that generalizes across all demonstrations may lose important information (e.g. something between a speed walk and a skip might result in falling)
- One solution is to try to separate the underlying rewards and learn specific policies which can be easily matched by your learner

We define a method for doing this via *joint reward-policy options* and learn these in the context of Inverse Reinforcement Learning (IRL) using adversarial methods [2].

Reward Options in IRL

Generative Adversarial Inverse Reinforcement Learning (IRLGAN) provides a convenient framework for this, the goal is to optimize a min-max objective:

$$\max_{\pi_\Theta} \min_{R_{\hat{\Theta}}} - \left[\mathbb{E}_{\pi_\Theta}[\log R_{\hat{\Theta}}(s)] + \mathbb{E}_{\pi_E}[\log(1 - R_{\hat{\Theta}}(s))]\right] \tag{1}$$

But in our case, the reward is an optionated mixture:

$$R_{\Omega,\hat{\Theta}}(s) = \sum_\omega \pi_{\Omega,\zeta}(\omega|s) r_{\omega,\hat{\theta}}(s) \tag{2}$$

where $\zeta, \theta \in \Theta, \hat{\theta} \in \hat{\Theta}$ are the parameters of the policy-over-options, policy options, and reward options, respectively. This allows us to update the parameters of the policy-over-options and reward options together during the discriminator update.

$$L_\Omega = \mathbb{E}_\omega \left[\pi_{\Omega,\zeta}(\omega|s) L_{\hat{\theta},\omega}\right] + L_{reg} \tag{3}$$

Here, $L_{\hat{\theta},\omega}$ is the sigmoid cross-entropy loss of the reward options (discriminators). L_{reg} is a penalty or set of penalties which can encourage certain properties of the policy-over-options or the overall reward signal.

Mixture-of-Experts as Options

The proper loss formulation and regularizers can drive Mixture-of-Experts to converge to options. As in [1], a loss function of the form $L = (y - \frac{1}{||\Omega||}\sum_\omega \pi_\Omega(\omega|s) y_\omega(s))^2$ draws cooperation between experts, but a reformulation of the loss, $L = \frac{1}{||\Omega||}\sum_\omega \pi_\Omega(\omega|s)(y - y_\omega(s))^2$, encourages specialization. In our discriminator loss, the gating function will thus move toward deterministic selection of experts (a policy over options). To further encourage this and an even distribution of information across options we add several regularizers to our loss.

$$\mathcal{L}_{reg} = \lambda_b L_b + \lambda_e L_e + \lambda_v L_v + \lambda_{\text{MI}} L_{\text{MI}} \tag{4}$$

$$L_b = \sum_\omega ||\mathbb{E}_s[\pi_\Omega(\omega|s)] - \tau||_2 \tag{5}$$

$$L_e = \mathbb{E}_s\left[||\left(\frac{1}{||\Omega||}\sum_\omega \pi_\Omega(\omega|s)\right) - \tau||_2\right] \tag{6}$$

$$L_v = -\sum_\omega var_\omega\{\pi_\Omega(\omega|s)\} \tag{7}$$

$$L_{\text{MI}} = \sum_{\omega\in\Omega}\sum_{\hat{\omega}\in\Omega,\omega\neq\hat{\omega}} I(\pi_\omega, \pi_{\hat{\omega}}). \tag{8}$$

- τ is the target sparsity rate (which we set to .5 for all cases)
- L_b encourages a uniform distribution of options over the data
- L_e drives toward a target sparsity of activations per example (doubly encouraging our mixtures to be sparse)
- L_v also encourages varied π_Ω activations while discouraging uniform selection
- L_{MI} penalizes duplicated information, where $I(\pi_\omega, \pi_{\hat{\omega}})$ is the mutual information between two policy options

Benefits and the Future

- Using a Mixture-of-Experts solution for learning reward-policy options allows for distribution and scalability as described in [3]
- Learning reward options allows for transfer from noisy demonstrations (one-shot transfer learning in inverse reinforcement learning). That is, where the demonstrations come from many experts in environments with different dynamics.
- This formulation of Mixtures-of-Experts as options can be expanded easily to hierarchical learning

OptionGAN: The Algorithm

We can combine these to learn the policy-over-options along with reward options end-to-end.

Algorithm 1: OptionGAN

Input : Expert trajectories $\tau_E \sim \pi_E$.

Initialize $\theta, \hat{\theta}$

for $i = 0, 1, 2, \ldots$ **do**

  Sample trajectories $\tau_N \sim \pi_{\Theta_i}$

  Update discriminator options parameters $\hat{\theta}, \omega$ and policy-over-options parameters ζ, to minimize:

$$L_\Omega = \mathbb{E}_\omega \left[\pi_{\Omega,\zeta}(\omega|s) L_{\hat{\theta},\omega}\right] + L_{reg}$$

$$L_{\hat{\theta},\omega} = \mathbb{E}_{\tau_N}[\log r_{\hat{\theta},\omega}(s)] + \mathbb{E}_{\tau_E}[\log(1 - r_{\hat{\theta},\omega}(s))]$$

  Update policy options (with constrained update step and parameters $\theta_\omega \in \Theta_\Omega$) according to:

$$\mathbb{E}_{\tau_N}[\nabla_\theta \log \pi_\Theta(a|s) \mathbb{E}_{\tau_N}[\log(R_{\Omega,\hat{\Theta}}(s))|s_0 = \bar{s}]]$$

end

The Results

Decomposition into reward-policy options helps in most scenarios, especially transfer learning...

Task	Expert	IRLGAN	OptionGAN (2ops)	OptionGAN (4ops)
Hopper-v1	3778.8 ± 0.3	**3736.3 ± 152.4**	*3641.2 ± 105.9*	*3715.5 ± 17.6*
HalfCheetah-v1	4156.9 ± 8.7	3212.9 ± 69.9	**3714.7 ± 87.5**	*3616.1 ± 127.3*
Walker2d-v1	5528.5 ± 7.3	*4158.7 ± 247.3*	*3858.5 ± 504.9*	**4239.3 ± 314.2**
Hopper (One-Shot)	3657.7 ± 25.4	2775.1 ± 203.3	*3409.4 ± 80.8*	**3464.0 ± 67.8**
HalfCheetah (One-Shot)	4156.9 ± 51.3	1296.3 ± 177.8	1679.0 ± 284.2	**2219.4 ± 231.8**
Walker (One-Shot)	4218.1 ± 43.1	3229.8 ± 145.3	**3925.3 ± 138.9**	*3769.40 ± 170.4*
HopperSimpleWall-v0	3218.2 ± 315.7	*2897.5 ± 753.5*	*3140.3 ± 674.3*	**3272.3 ± 569.0**
RoboschoolHumanoidFlagrun-v1	2822.1 ± 531.1	*1455.2 ± 567.6*	*1868.9 ± 723.7*	**2113.6 ± 862.9**

Table 1: True Average Return with the standard error across 10 trials on the 25 final evaluation rollouts using the final policy.

Properties of reward-policy options

Option selection demonstrates temporal cohesion and learning of macro-actions...

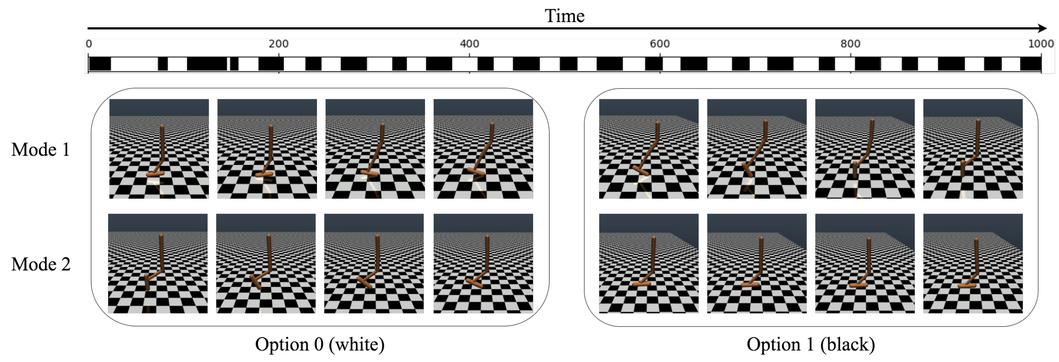

Figure 1: The policy-over-options elicits two interpretable behaviour modes per option, but temporal cohesion and specialization is seen between these behaviour modes across time within a sample rollout trajectory.

Structure is found in expert demonstrations and divided among reward options...

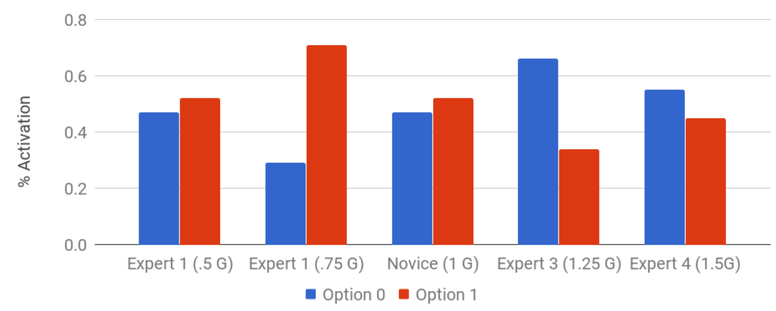

Figure 2: Probability distribution of π_Ω over options on expert demonstrations. Inherent structure is found in the underlying demonstrations. The .75G demonstration state spaces are significantly assigned to Option 1 and similarly, the 1.25G state spaces to Option 0.

References

[1] Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. "Adaptive mixtures of local experts." *Neural computation*, 1991.

[2] Jonathan Ho and Stefano Ermon. "Generative adversarial imitation learning." In *Advances in Neural Information Processing Systems*, 2016.

[3] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer." *International Conference on Learning Representations*, 2017.

McGill NSERC CRSNG amazon web services | Cloud Credits for Research