Project Milestone Report: Learning From Your Peers

Harvey Hu, Chirag Sharma

September 2022

1 Initial Experiment

For our initial experiment, we set up a 2-agent peer system such that both agents are influenced by the other during training via value signals (see the *Peer influence via value signals* method description in the project proposal).

Specifically, we initialize two actor-critic agents, A_1 and A_2, where each $A_k = \{\pi_{\theta_k}, V_{\phi_k}\}$. Here, π_{θ_k} corresponds to the actor/policy of agent A_k and V_{ϕ_k} is the critic network used to predict $V(\mathbf{s})$ values for states when training π_{θ_k}. Then, in each iteration of training, each actor independently collects training trajectories to populate its replay buffer to ensure that it is always being trained on trajectories that are collected from a similar policy (avoids off-policy learning issues). Both actors and critics are trained as per the standard actor-critic algorithm:

$$\phi_k \leftarrow \arg\min_{\phi} \sum_{i,t} (V_\phi(s_{it}) - y_{it})^2 \qquad \theta_k \leftarrow \arg\max_{\theta} \frac{1}{N} \sum_{i=1}^{N} \sum_{t} \nabla_\theta \log \pi_\theta(a_{it}|s_{it}) \left(y_{it} - V_{\phi_k}(s_{it})\right)$$

where the targets y_{it} $r(s_{it}, a_{it}) + \gamma V_{\phi_k}(s_{i(t+1)})$ use the critics' predictions. However, we make a change during training such that each critic network V_{ϕ_k} takes in a new input $\alpha_k = f_\psi(V_{\phi_{k'}})$, where k' is the other agent, with some probability $1 - \epsilon$ (where ϵ is a hyperparameter). Here f is just another neural network encoder that encodes the "value signals" from the other critic in the system. Otherwise, with probability ϵ, this α_k input is zeroed out. During evaluation, the α_k input is always zeroed out. Thus, this architecture captures the notion of obtaining "advice" from peers while learning.

2 Results

We ran our experiment on the Cartpole-v0 environment to see if the peer architecture is able to easily learn a simple task.

Both agents converged to the optimal policy fairly quickly in most cases across our five trial runs.

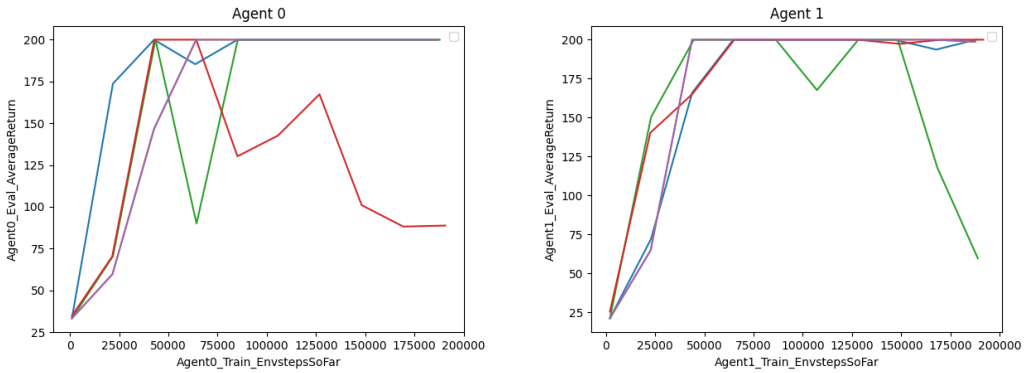

Figure 1: Experimental Results