Notation

A task $\mathcal{T} = \{p(s_0), p(s_{t+1}|s_t, a_t), r(s_t, a_t)\}$ consists of an initial state distribution, transition distribution and reward function.

$p(\mathcal{T})$ is a distribution over tasks.

$c_n^{\mathcal{T}} = (s_n, a_n, r_n, s'_n)$ is one transition in the task $\mathcal{T}$.

$c_{1:N}^{\mathcal{T}}$ refers to all the data collected so far for task $\mathcal{T}$, denoted by c when there is no confusion.

Motivations

1. Sample efficiency during both meta-training and meta-test phase.

2. Sample efficiency during meta-training means requiring fewer samples from previous experience.

3. Sample efficiency during meta-test means requiring fewer samples from the new task.

Probabilistic Latent Context for sample efficiency during meta-testing

1. When faced with a new task, the policy needs to quickly infer what the task is using the data collected from this new task.

2. The paper does this by using a learned function $q_\phi(z|c)$ to map the recently collected data c to latent context variable z. The role of z is to summarize c and capture minimally sufficient statistics about the task, without modeling irrelevant dependencies.

3. To encourage temporally correlated exploration, we want a source of stochasticity in action selection that varies across episode, but stay constant within each episode. Just sampling action from a stochastic policy does not fulfill this requirement.

4. This role of encouraging temporally correlated exploration is assigned to $q_\phi(z|c)$, which outputs a distribution over the latent context variable z, instead of a point estimate.

5. At the beginning of each episode, a z is sampled from $q_\phi(z|c)$. The value of this z is kept fixed throughout the episode. z is used as an input to the policy in addition to the state, i.e. $a \sim \pi_\theta(a|s, z)$.

6. c is an unordered set of transition tuple $\{(s_n, a_n, r_n, s'_n)\}_{i=1}^N$. Thus, we want an architecture for q_ϕ that does not care about the ordering of its input. The input permutation-invariant architecture the paper uses is:

$$q_\phi(z|c) = \prod_{n=1}^{N} \mathcal{N}\left(f_\phi^\mu(c_n), f_\phi^\sigma(c_n)\right)$$

7. When faced with a completely new task, z is sampled from a unit Gaussian prior for the first episode. The recently collected samples are then used to obtain the approximate posterior $q_\phi(z|c)$.

8. To train the parameter ϕ, the paper adopts an amortized variational inference approach, where the variational lower bound is:

$$E_{\mathcal{T}}\left[E_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{c}^{\mathcal{T}})}\left[R(\mathcal{T}, \mathbf{z}) + \beta D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{c}^{\mathcal{T}}) \| p(\mathbf{z})\right)\right]\right]$$

where $p(z)$ is the unit Gaussian prior. $R(\mathcal{T}, \mathbf{z})$ can be chosen to recover the state-action value, maximize the return of the policy or reconstruct the state and reward function.

Off-policy training for sample efficiency during meta-training

1. They would like to make use of off-policy data for meta-training to increase sample efficiency of the meta-training phase.

2. However, meta-learning is predicated on the assumption that the distribution of data used for adaptation will match across meta-training and meta-test phase.

3. They use this reason to justify why they were not able to combine meta-learning and value-based RL method. Thus, they use an actor-critic method SAC.

4. During training, the parameters of ϕ and θ (actor, critic) are both trained with off-policy data. However, the data for ϕ comes from recently collected data, recollected every 1000 meta-training optimization steps. The data for training θ is sampled uniformly from all previously collected data.

5. The parameter ϕ is trained to recover the state-action value, which performed better than training it to maximize actor returns or reconstruct states and rewards.

Experiments - environments and tasks

1. The environments and tasks require adaptation, either across different reward functions or different dynamics function.

2. All tasks have a fixed horizon length of 200.

Comment

1. I get the point about temporally extended exploration. But wouldn't we want to update the exploration "intent" as new transition tuple (information) becomes available?

2. What exactly does match mean in the following sentence "Meta-learning typically operates on the principle that meta-training time should match meta-test time"?

3. Does the following sentences and the surrounding text makes sense? "This problem inherently cannot be solved by off-policy RL methods that minimize temporal-difference error, as they do not have the ability to directly optimize for distributions of states visited. In contrast, policy gradient methods have direct control over the actions taken by the policy"

4. I was surprised to find out that this paper does not have meta-level optimization. It is purely about inferring the task.

5. Why is the task horizon limited to 200 in this work?

6. Formalize this sentence "designing off-policy meta-RL algorithms is non-trivial partly because modern meta-learning is predicated on the assumption that the distribution of data used for adaptation will match across meta-training and meta-test.".

7. When we sample the context c_i in line 13 in algorithm 1, we probably want to sample transitions with high reward. Otherwise, because the transition function is constant, the inference network can not infer the correct current task.