Meta-RL——Efficient Off-Policy Meta-Reinforcement Learning via Probabilistic Context Variables

作者：凯鲁嘎吉 - 博客园 http://www.cnblogs.com/kailugaji/

这篇博客是"Efficient Off-Policy Meta-Reinforcement Learning via Probabilistic Context Variables"的简要阅读笔记，围绕如何从过去学习的任务中针对新的任务获取有效的信息以及如何对新任务的不确定性做出更准确的判断这两个问题，论文通过一个任务编码器来学习任务的表征(概率上下文变量z)，融合软演员评论员算法(Soft Actor-Critic, SAC)与变分推断KL损失，提出一种异策略元强化学习算法，通过分离任务推断与智能体学习过程来提高元学习中任务的学习样本利用效率。

深度强化学习算法需要大量的经验来学习单个任务。虽然元强化学习(meta-RL)算法可以让智能体从少量经验中学习新技能，但一些主要的挑战阻碍了它们的实用性。目前的方法严重依赖于同策略经验，限制了它们的样本效率。在适应新任务时，它们也缺乏推理任务不确定性的机制，这限制了它们在稀疏奖励问题上的有效性。本文通过开发一种异策略元强化学习算法来解决这些挑战，所提算法(Probabilistic Embeddings for Actor-critic meta-RL, PEARL)将任务推理和控制分离开来。算法对潜在的任务变量进行在线概率滤波，从少量的经验中推断出如何解决新任务。这种概率解释使得后验采样能够用于结构化和高效的探索。论文证明了如何将这些任务变量与异策略强化学习算法集成，以实现元训练和自适应效率。所提方法在样本效率和在几个元强化学习基准上的渐近性能上都比以前的算法高出20-100倍。

1. 基础知识

1.1 传统的强化学习 vs 元强化学习 (Standard RL vs Meta-RL)

Efficient Off-Policy Meta-Reinforcement Learning via Probabilistic Context Variables

➢ 传统的强化学习 vs 元强化学习

Standard RL

$$\theta^* = \arg\max_\theta \underbrace{\mathbb{E}_{\pi_\theta(\tau)}\left[R(\tau)\right]}_{J(\theta)}$$

$$= f_{RL}(\mathcal{M})$$

MDP

$\mathcal{M}$

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$$

Meta-RL

$$\theta^* = \arg\max_\theta \sum_{i=1}^{n} \underbrace{\mathbb{E}_{\pi_{\phi_i}(\tau)}\left[R(\tau)\right]}_{J_i(\theta)}$$

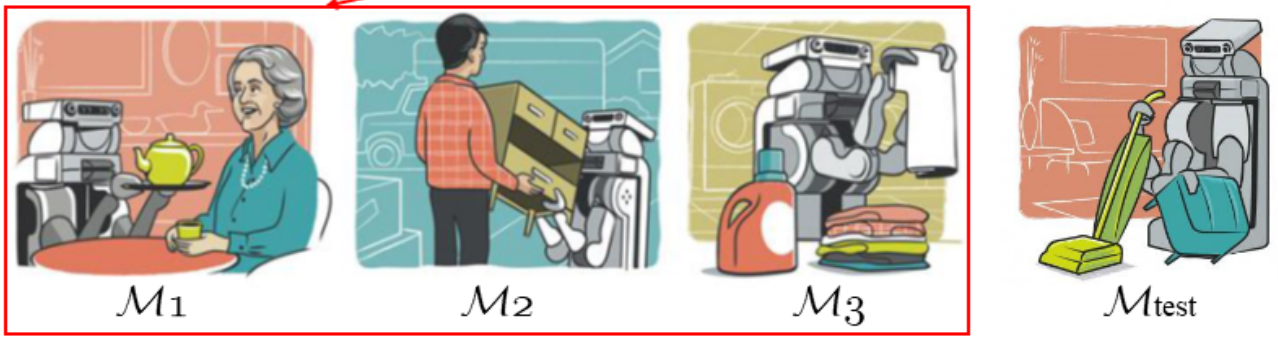
Meta-training/ Outer loop

where $\phi_i = f_\theta(\mathcal{M}_i)$

Adaptation/ Inner loop

MDP for task i

- Explore: Collect the most informative data
- Adapt: Use that data to obtain the optimal policy



$\mathcal{M}_1$ $\mathcal{M}_2$ $\mathcal{M}_3$ $\mathcal{M}_{\text{test}}$

$$\theta \leftarrow \theta + \beta \sum_i \nabla_\theta J_i\left[\theta + \alpha \nabla_\theta J_i(\theta)\right]$$

MAML

Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, Deirdre Quillen. Efficient Off-Policy Meta-Reinforcement Learning via Probabilistic Context Variables. ICML, 2019.

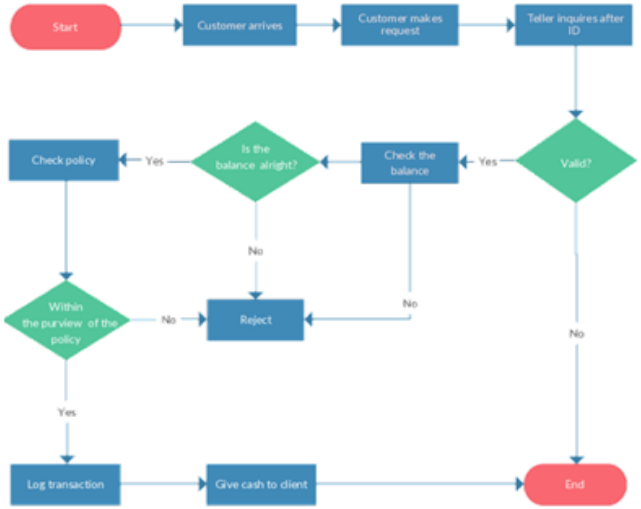
1

1.2 同策略 vs 异策略 (On-Policy vs Off-Policy)

➢ 同策略 vs 异策略

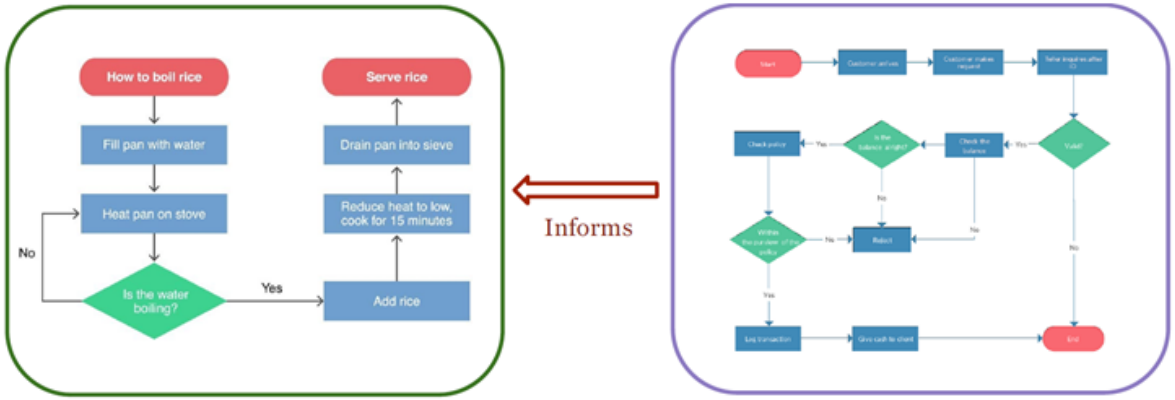
On-Policy

- 采样策略与优化策略相同(Only one policy used throughout the system to both explore and select actions).
- **Pros:** Less costly.
- **Cons:** Not optimal because policy covers exploration as well.



Off-Policy

- 采样策略与优化策略不同(Two policies, one for exploring and the other for action selection).
- **Pros:** More optimal solution achieved with fewer samples.
- **Cons:** Expensive computationally.



Target Policy (Exploitation)

Behaviour Policy (Exploration)

Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, Deirdre Quillen. Efficient Off-Policy Meta-Reinforcement Learning via Probabilistic Context Variables. ICML, 2019.

1.3 软演员评论员算法(Soft Actor-Critic, SAC)

➢ 软演员评论员(Soft Actor-Critic, SAC)算法

1. Q-function update

Update Q-function to evaluate current policy:

$$Q(\mathbf{s},\mathbf{a}) \leftarrow r(\mathbf{s},\mathbf{a}) + \mathbb{E}_{\mathbf{s}'\sim p_{\mathbf{s}},\ \mathbf{a}'\sim\pi}\left[Q(\mathbf{s}',\mathbf{a}') - \log\pi(\mathbf{a}'|\mathbf{s}')\right]$$

This converges to Q^{π}.

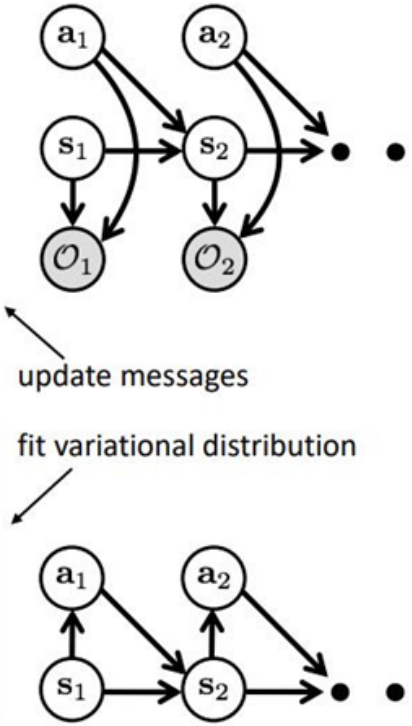
2. Update policy

Update the policy with gradient of information projection:

$$\pi_{\text{new}} = \arg\min_{\pi'} \mathrm{D}_{\mathrm{KL}}\left(\pi'(\cdot|\mathbf{s}) \,\middle\|\, \frac{1}{Z}\exp Q^{\pi_{\text{old}}}(\mathbf{s},\cdot)\right)$$

In practice, only take one gradient step on this objective

3. Interact with the world, collect more data



Haarnoja, Zhou, Hartikainen, Tucker, Ha, Tan, Kumar, Zhu, Gupta, Abbeel, L. **Soft Actor-Critic Algorithms and Applications**. '18

2. 概率潜在上下文 (Probabilistic Latent Context)

➢ 概率潜在上下文 (Probabilistic Latent Context)

任务分布$p(\mathcal{T})$, 任务$\mathcal{T}=\{p(s_0),p(s_{t+1}\mid s_t,a_t),r(s_t,a_t)\}$, 初始状态分布$p(s_0)$, 转移分布$p(s_{t+1}\mid s_t,a_t)$, 奖励函数$r(s_t,a_t)$

- 每个任务都是一个马尔可夫决策过程(Markov Decision Process, MDP)，它由一组状态、动作、一个转换函数和一个有界的奖励函数组成。设转移和奖励函数未知，但可以通过在环境中采取动作进行采样。这个问题定义包含了带有可变转移函数(例如，具有不同动态的机器人)和可变奖励函数(例如，导航到不同的位置)的任务分配。
- **元训练**：给定从$p(\mathcal{T})$采样而来的一组训练任务，元训练过程学习一种策略，该策略通过对过去转移历史的条件作用来适应新任务，将其称为上下文c。$c_n^{\mathcal{T}}=(s_n,a_n,r_n,s_n')$, $c_{1:N}^{\mathcal{T}}$组成到目前为止收集到的经验(为简化，用c表示$c_{1:N}^{\mathcal{T}}$)
- **元测试**：策略必须适应来自$p(\mathcal{T})$的新任务。
- 我们获取关于当前任务应该如何在潜在概率上下文变量Z中执行的知识，在此基础上，将策略设置为$\pi_\theta(a|s,Z)$，以便使其行为适应任务。元训练包括利用来自各种训练任务的数据，从新任务的最近经验历史中学习推断Z的值，以及优化策略以解决来自Z的后验样本的任务。下面描述元训练推理机制的结构。
- 为了能够适应，潜在上下文Z必须编码关于任务的显著信息。采用摊销变分推断方法来学习推断Z。
- 通过训练推断网络$q_\phi(z\mid c)$来估计后验$p(z\mid c)$，推断网络可以以一种模型无关的方式进行优化，以建模状态-动作值函数，或者通过任务分布上的策略最大化回报。假设这个目标是对数似然，由此得到的变分下界为

元训练进行优化参数

$$\mathbb{E}_{\mathcal{T}}\left[\mathbb{E}_{z\sim q_\phi(z|c^{\mathcal{T}})}\left[R(\mathcal{T},z)+\beta D_{KL}\left(q_\phi(z\mid c^{\mathcal{T}})\,\|\,p(z)\right)\right]\right]$$

本文：SAC

为简化，该项为标准正态分布

- $R(\mathcal{T},z)$可以是各种目标函数，KL散度项也可以解释为对限制Z和c之间互信息的信息瓶颈进行变分近似的结果。
- 直观地说，这一瓶颈限制z仅包含适应新任务所需的上下文信息，从而减轻对训练任务的过拟合。

Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, Deirdre Quillen. Efficient Off-Policy Meta-Reinforcement Learning via Probabilistic Context Variables. ICML, 2019.

➢ 概率潜在上下文 (Probabilistic Latent Context)

- 公式由来的个人理解 $$\mathbb{E}_T\left[\mathbb{E}_{z\sim q_\phi(z|c^T)}\left[R(\mathcal{T},z)+\beta D_{KL}\left(q_\phi(z\mid c^T)\,\|\,p(z)\right)\right]\right]$$

- **解读一**：从变分推断角度
- 由变分推断， $$\ln p(c^T)=\mathbb{E}_{z\sim q_\phi(z)}\left[\ln\frac{p(c^T\mid z)p(z)}{q_\phi(z)}\right]+D_{KL}\left(q_\phi(z)\,\|\,p(z\mid c^T)\right)$$
- 变分下界ELBO又可以分解为

$$\max L(q)=\mathbb{E}_{z\sim q_\phi(z)}\left[\ln\frac{p(c^T\mid z)p(z)}{q_\phi(z)}\right]=\underbrace{\mathbb{E}_{z\sim q_\phi(z|c^T)}\left[\ln p(c^T\mid z)\right]}_{\text{生成}}-\underbrace{D_{KL}\left(q_\phi(z\mid c^T)\,\|\,p(z)\right)}_{\text{推断}}$$

- 若把后面KL项也放入期望里，则上式变为

$$\min\mathbb{E}_{z\sim q_\phi(z|c^T)}\left[-\ln p(c^T\mid z)+D_{KL}\left(q_\phi(z\mid c^T)\,\|\,p(z)\right)\right]$$

可替换为RL目标函数

- **解读二**：从RL角度出发，目标函数为RL的目标，约束条件为KL散度小于某一常数，通过拉格朗日乘数法，将其转化为无约束问题。

$$\min\mathbb{E}_T\left[R(\mathcal{T})\right]=\mathbb{E}_T\left[\mathbb{E}_{z\sim q_\phi(z|c^T)}\left[R(\mathcal{T},z)\right]\right]$$

$$s.t.\ D_{KL}\left(q_\phi(z\mid c^T)\,\|\,p(z)\right)\le\varepsilon$$

变分推断与变分自编码器–凯鲁嘎吉–博客园 https://www.cnblogs.com/kailugaji/p/12463966.html

➢ Probabilistic Latent Context

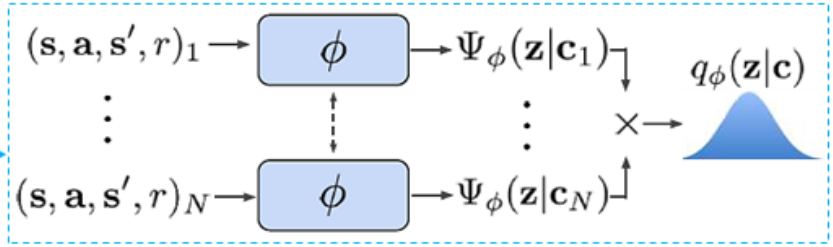
改变网络的输入顺序，产生的输出未变。

- 我们注意到，一个完全可观测MDP的编码可以是置换不变的，关于采样转移$\{s_i, a_i, s_i', r_i\}$。
- 转换和奖励函数(以及MDP)可以从这种转换的无序集合中重建。因此，这些转换的集合足以训练值函数或推断任务是什么。考虑到这一点，我们为$q_\phi(z \mid c_{1:N})$选择一个置换不变的表示，将其建模为N个独立因子(高斯分布)的乘积，

$$q_\phi(z \mid c_{1:N}) \propto \prod_{n=1}^{N} \Psi_\phi(z \mid c_n) = \prod_{n=1}^{N} \mathcal{N}(f_\phi^\mu(c_n), f_\phi^\sigma(c_n))$$

策略: $\pi_\theta(a_t \mid s_t, z_t)$

推断网络: $q_\phi(z_t \mid s_1, a_1, r_1, \ldots, s_t, a_t, r_t)$



- 将潜在上下文建模为概率的方法可以使我们在元测试时利用后验采样进行有效的探索。在强化学习中，探索的后验采样开始于MDPs上的先验分布，根据到目前为止所得到的经验计算一个后验分布，并在一个回合的持续时序内为采样MDP执行最优策略，作为一种有效的探索方法。特别是，根据随机MDP进行的最优动作允许时序扩展的探索，这意味着即使在动作的结果不能立即提供任务信息的情况下，智能体也可以进行检验假设。

- 在单任务DRL设置中，Osband等人研究了后验采样和深度探索的优势，通过自举保持值函数的近似后验。而本文PEARL直接推断潜在上下文Z的后验。

  如果对MDP重构进行优化，则可能编码MDP本身；

  如果对策略进行优化，则可能编码最优行为；

  如果对评论员进行优化，则可能编码值函数。

✓ **元训练：** 利用训练任务来学习Z之上的先验，这捕获了任务的分布，并学习有效地利用已有经验来推断新任务。

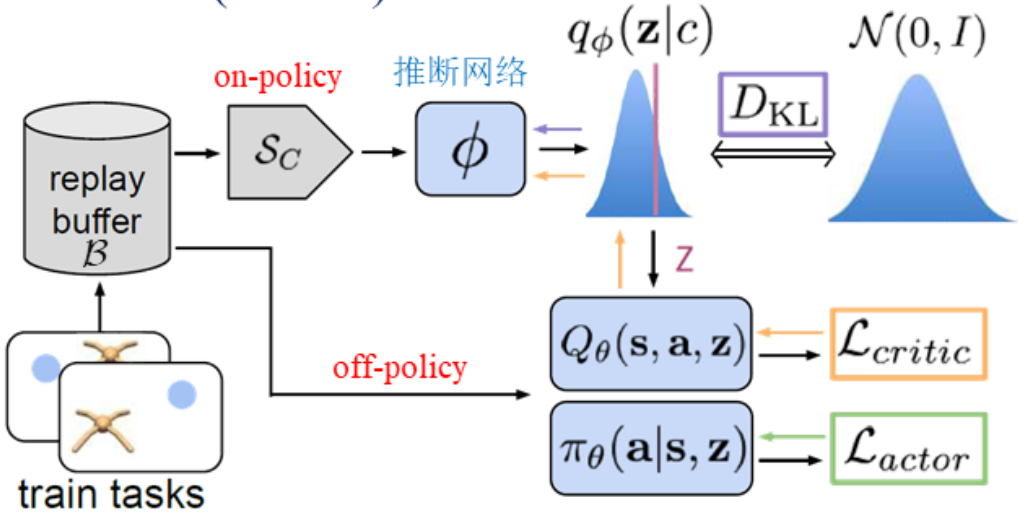
✓ **元测试：** 首先从之前的z中抽取样本，并根据每一个z执行一个回合，从而以在时序上扩展和多样化的方式进行探索。然后，利用收集到的经验来更新后验，并以一种随着置信区间变窄而作用越来越优的方式继续连贯地探索，类似于后验抽样。

Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, Deirdre Quillen. Efficient Off-Policy Meta-Reinforcement Learning via Probabilistic Context Variables. ICML, 2019.

3. Probabilistic Embeddings for Actor-critic meta-RL (PEARL)

➢ Probabilistic Embeddings for Actor-critic meta-RL (PEARL)

- 虽然上述概率上下文模型可以直接与同策略的策略梯度方法相结合，但本文工作一个主要目标是实现有效的**异策略**元强化学习，其中元训练和快速适应的样本数量都是很小的。在之前的工作中，元训练过程的效率在很大程度上被忽视了，这些工作使用了稳定但相对低效的同策略算法。
- 然而，设计异策略元强化学习算法并非易事，部分原因是现代元学习是基于这样的假设：
  1. **元训练和元测试之间的数据分布需要相匹配**。在强化学习中，这意味着由于在元测试时使用同策略的数据进行自适应，因此在元训练时也应该使用同策略的数据。
  2. **元强化学习要求策略对分布进行推理，从而学习有效的随机探索策略**。这个问题本质上不能通过最小化时序差分损失的异策略强化学习方法来解决，因为它们不能直接优化所访问的状态分布。相反，策略梯度法直接控制策略所采取的行动。
- 考虑到这两个挑战，元学习和基于值函数的强化学习的简单结合可能是低效的。在实践中，我们无法优化这种方法。



本文将潜在上下文变量建模为概率，通过后验采样实现元学习探索策略的异策略问题。值得注意的是用于训练编码器的数据不需要与用于训练策略的数据相同，本文解决了异策略元训练数据和同策略测试时间适应数据之间的分布偏移。策略可以将上下文z作为异策略强化学习循环内状态的一部分，而探索过程的随机性由编码器$q(z|c)$中的不确定性提供。演员与评论员总是使用从整个回放缓冲池$\mathcal{B}$(replay buffer)中采样的异策略数据进行训练。定义一个采样器Sc对上下文批次(batch)进行采样来训练编码器。允许Sc从整个缓冲池进行采样，会出现与同策略测试数据极不匹配的分布，且会影响经验性能。然而，上下文不需要是严格的同策略；我们发现，从回放缓冲池对最近收集的数据进行采样的中间策略可以更好地保持同策略的性能。

Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, Deirdre Quillen. Efficient Off-Policy Meta-Reinforcement Learning via Probabilistic Context Variables. ICML, 2019.

➢ Probabilistic Embeddings for Actor-critic meta-RL (PEARL)

- 本文在软演员评论员算法(Soft Actor-Critic, SAC)的基础上构建了所提算法，SAC是一种基于最大熵强化学习目标的异策略演员评论员方法，它用策略的信息熵来增强传统的折扣回报之和。SAC具有良好的样本效率和稳定性。

Gradients are not being computed through it

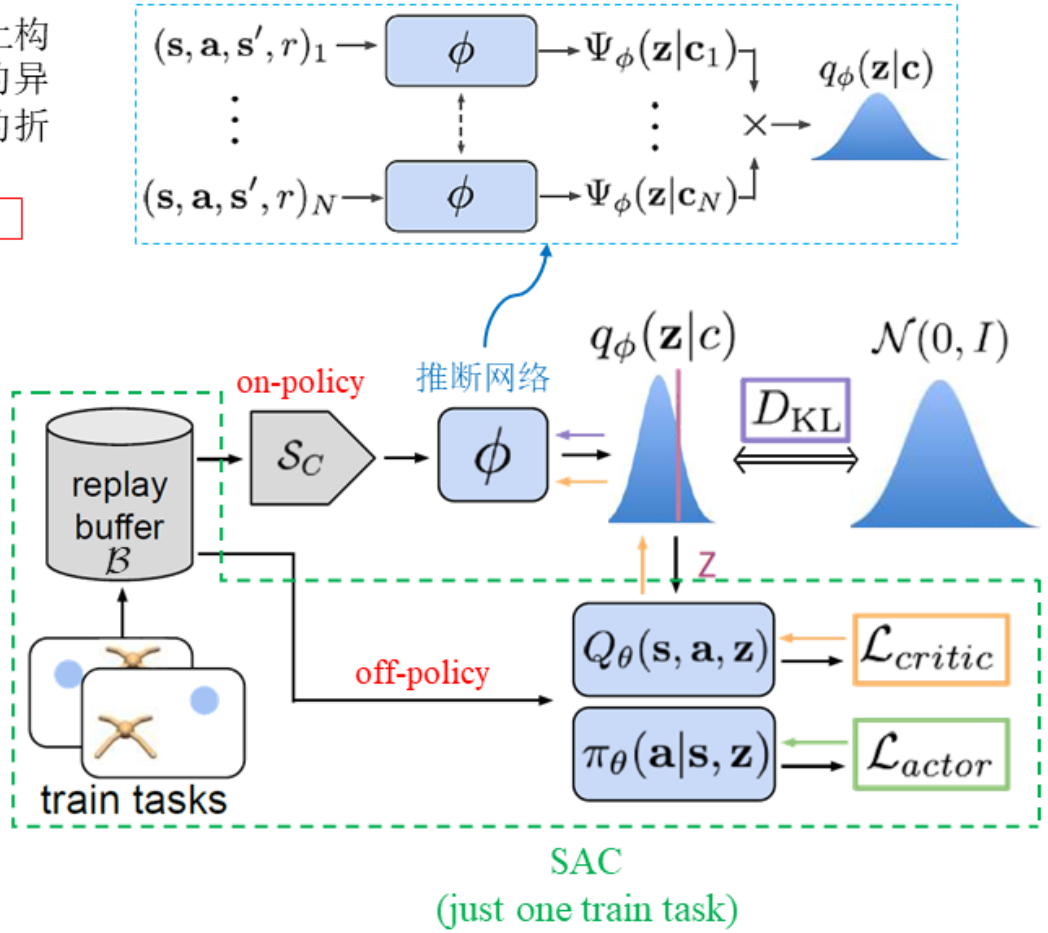
$$\mathcal{L}_{critic} = \mathbb{E}_{\substack{(s,a,r,s')\sim\mathcal{B} \\ z\sim q_\phi(z|c)}} \left[Q_\theta(s,a,z) - (r + \bar{V}(s',\bar{z})) \right]^2$$

target network

$$\mathcal{L}_{actor} = \mathbb{E}_{\substack{s\sim\mathcal{B}, a\sim\pi_\theta \\ z\sim q_\phi(z|c)}} \left[D_{KL}\left(\pi_\theta(a \mid s,\bar{z}) \middle\| \frac{\exp(Q_\theta(s,a,\bar{z}))}{\mathcal{Z}_\theta(s)} \right) \right]$$

$$\mathcal{L}_{KL} = \mathbb{E}_{z\sim q_\phi(z|c)} \left[D_{KL}\left(q_\phi(z \mid c) \,\|\, \mathcal{N}(0,I) \right) \right]$$

- 在元训练期间，对上下文批次和演员-评论员批次分别进行采样。具体来说，上下文采样器Sc从最近收集的一批数据中统一采样，每1000个元训练优化步骤重新收集一次。演员和评论员通过从整个回放缓冲池一致抽取的转移批次进行训练。



Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, Deirdre Quillen. Efficient Off-Policy Meta-Reinforcement Learning via Probabilistic Context Variables. ICML, 2019.

➢ Probabilistic Embeddings for Actor-critic meta-RL (PEARL)

Algorithm 1 PEARL Meta-training

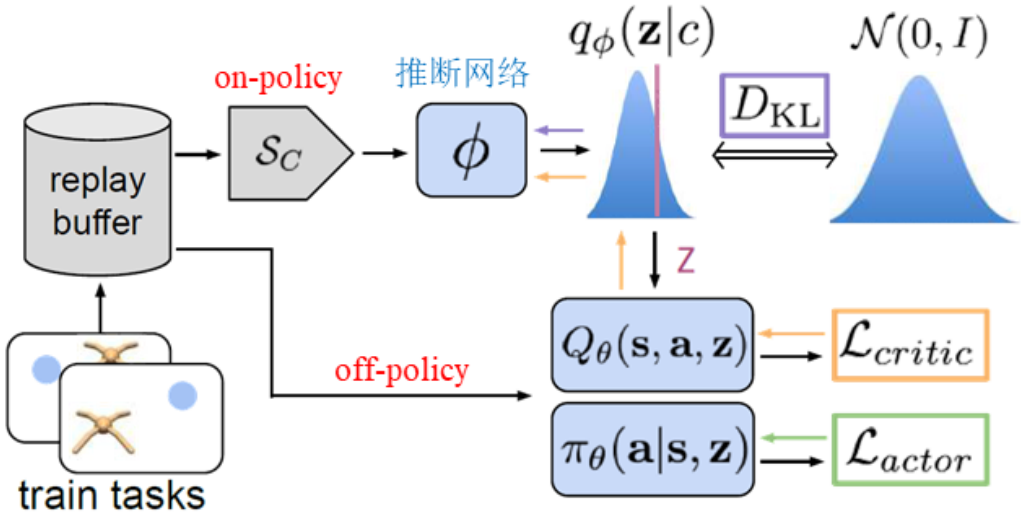
Require: Batch of training tasks $\{\mathcal{T}_i\}_{i=1...T}$ from $p(\mathcal{T})$, learning rates $\alpha_1, \alpha_2, \alpha_3$

Initialize replay buffers $\mathcal{B}^i$ for each training task
while not done **do**
  for each $\mathcal{T}_i$ **do** # sampling data
    Initialize context $\mathbf{c}^i = \{\}$
    for $k = 1, \ldots, K$ **do**
      Sample $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{c}^i)$ # posterior distribution
      Gather data from $\pi_\theta(\mathbf{a}|\mathbf{s}, \mathbf{z})$ and add to $\mathcal{B}^i$
      Update $\mathbf{c}^i = \{(\mathbf{s}_j, \mathbf{a}_j, \mathbf{s}'_j, r_j)\}_{j:1...N} \sim \mathcal{B}^i$
    end for
  end for
  for step in training steps **do** # training
    for each $\mathcal{T}_i$ **do**
      Sample context $\mathbf{c}^i \sim \mathcal{S}_c(\mathcal{B}^i)$ and RL batch $b^i \sim \mathcal{B}^i$
      Sample $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{c}^i)$ # posterior distribution
      $\mathcal{L}^i_{actor} = \mathcal{L}_{actor}(b^i, \mathbf{z})$
      $\mathcal{L}^i_{critic} = \mathcal{L}_{critic}(b^i, \mathbf{z})$
      $\mathcal{L}^i_{KL} = \beta D_{\mathrm{KL}}(q(\mathbf{z}|\mathbf{c}^i)||r(\mathbf{z}))$ # $r(z)$: prior distribution
    end for
    $\phi \leftarrow \phi - \alpha_1 \nabla_\phi \sum_i \left(\mathcal{L}^i_{critic} + \mathcal{L}^i_{KL}\right)$ # Bellman update
    $\theta_\pi \leftarrow \theta_\pi - \alpha_2 \nabla_\theta \sum_i \mathcal{L}^i_{actor}$
    $\theta_Q \leftarrow \theta_Q - \alpha_3 \nabla_\theta \sum_i \mathcal{L}^i_{critic}$
  end for
end while

Algorithm 2 PEARL Meta-testing

Require: test task $\mathcal{T} \sim p(\mathcal{T})$
Initialize context $\mathbf{c}^\mathcal{T} = \{\}$
for $k = 1, \ldots, K$ **do**
  Sample $z \sim q_\phi(\mathbf{z}|c^\mathcal{T})$
  Roll out policy $\pi_\theta(\mathbf{a}|\mathbf{s}, \mathbf{z})$ to collect data $D^\mathcal{T}_k = \{(\mathbf{s}_j, \mathbf{a}_j, \mathbf{s}'_j, r_j)\}_{j:1...N}$
  Accumulate context $\mathbf{c}^\mathcal{T} = \mathbf{c}^\mathcal{T} \cup D^\mathcal{T}_k$
end for

元测试：完成推断网络/编码器的训练后，面对一个新任务，PEARL就可以在学习中加入已编码过的信息，从而利用过去学习过的任务进行新任务的学习。



4. 参考文献

[1] Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, Deirdre Quillen. Efficient Off-Policy Meta-Reinforcement Learning via Probabilistic Context Variables. ICML, 2019.

Paper: http://proceedings.mlr.press/v97/rakelly19a.html

Code: https://github.com/katerakelly/oyster

Slides: https://icml.cc/media/Slides/icml/2019/hallb(12-11-00)-12-12-15-4607-efficient_off-p.pdf and https://cs.uwaterloo.ca/~ppoupart/teaching/cs885-spring20/slides/cs885-efficient-off-policy-meta-reinforcement-learning.pdf

Video: https://youtube.videoken.com/embed/D0UmVbbJxS8?tocitem=100

[2] CS285 Meta-Learning http://rail.eecs.berkeley.edu/deeprlcourse-fa20/static/slides/lec-22.pdf

[3] CS330 Meta-RL http://cs330.stanford.edu/fall2019/slides/Exploration%20in%20Meta-RL.pdf and https://web.stanford.edu/class/cs330/slides/cs330_metarl2_2021.pdf

[4] PEARL — Probabilistic Embedding for Actor-critic RL | Zero https://xlnwel.github.io/blog/reinforcement%20learning/PEARL/

[5] RL——Deep Reinforcement Learning amidst Continual/Lifelong Structured Non-Stationarity - 凯鲁嘎吉 - 博客园 https://www.cnblogs.com/kailugaji/p/15562366.html

[6] 变分推断与变分自编码器 - 凯鲁嘎吉 - 博客园 https://www.cnblogs.com/kailugaji/p/12463966.html