12 - PPO算法

重要性采样

重要性采样是一种估计随机变量的期望或者概率分布的统计方法。它的原理也很简单，假设有一个函数 $f(x)$ ，需要从分布 $p(x)$ 中采样来计算其期望值，但是在某些情况下我们可能很难从 $p(x)$ 中采样，这个时候我们可以从另一个比较容易采样的分布 $q(x)$ 中采样，来间接地达到从 $p(x)$ 中采样的效果。

$$\mathbb{E}_{p(x)}[f(x)] = \int_a^b f(x)\frac{p(x)}{q(x)}q(x)dx = \mathbb{E}_{q(x)}[f(x)\frac{p(x)}{q(x)}]$$

对于离散分布的情况：

$$\mathbb{E}_{p(x)}[f(x)] = \frac{1}{N}\sum f(x_i)\frac{p(x_i)}{q(x_i)}$$

这样一来原问题就变成了只需要从 $q(x)$ 中采样，然后计算两个分布之间的比例 $\frac{p(x)}{q(x)}$ 即可，这个比例称之为重要性权重。

PPO算法

PPO 算法的核心思想就是通过重要性采样来优化原来的策略梯度估计

$$J^{\mathrm{TRPO}}(\theta) = \mathbb{E}[r(\theta)\hat{A}_{\theta_{\mathrm{old}}}(s,a)]$$
$$r(\theta) = \frac{\pi_\theta(a|s)}{\pi_{\theta_{\mathrm{old}}}(a|s)}$$

这个损失就是置信区间的部分，一般称作 TRPO 损失。这里旧策略分布 $\pi_{\theta_{\mathrm{old}}}(a|s)$ 就是重要性权重部分的目标分布 $p(x)$ ，目标分布是很难采样的，所以在计算重要性权重的时候这部分通常用上一次与环境交互采样中的概率分布来近似。相应地， $\pi_{\theta(a|s)}$ 则是提议分布，即通过当前网络输出的 `probs` 形成的类别分布 Catagorical 分布（离散动作）或者 Gaussian 分布（连续动作）。

读者们可能对这个写法感到陌生，似乎少了 Actor-Critic 算法中的 `logit_p`，但其实这个公式等价于

$$J^{\mathrm{TRPO}}(\theta) = \mathbb{E}_{(s_t,a_t)\sim\pi\theta'}[\frac{p_\theta(a_t|s_t)}{p_\theta^{'}(a_t|s_t)}A^{\theta'}(s_t,a_t)\nabla\log p_\theta(a_t^n|s_t^n)]$$

换句话说，本质上 PPO 算法就是在 Actor-Critic 算法的基础上增加了重要性采样的约束而已，从而确保每次的策略梯度估计都不会过分偏离当前的策略，也就是减少了策略梯度估计的方差，从而提高算法的稳定性和收敛性。

前面我们提到过，重要性权重最好尽可能地等于 1 ，而在训练过程中这个权重它是不会自动地约束到 1 附近的，因此我们需要在损失函数中加入一个约束项或者说正则项，保证重要性权重不会偏离 1 太远。

CLIP约束定义

$$J_{\mathrm{clip}}(\theta) = \hat{\mathbb{E}}_t[\min(r_t(\theta)\hat{A}_t, \mathrm{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)]$$

其中 ϵ 是一个较小的超参，一般取 0.1 左右。这个 clip 约束的意思就是始终将重要性权重 $r(\theta)$ 裁剪在 1 的邻域范围内，实现起来非常简单。

KL约束定义

$$J^{\mathrm{KL}}(\theta)=\hat{\mathbb{E}}_t[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\mathrm{old}}}(a_t|s_t)}\hat{A}_t-\beta\mathrm{KL}[\pi_{\theta_{\mathrm{old}}}(\cdot|s_t),\pi_\theta(\cdot|s_t)]]$$

KL 约束一般也叫 KL-penalty，它的意思是在 TRPO 损失的基础上，加上一个 KL 散度的惩罚项，这个惩罚项的系数 β 一般取 0.01 左右。