GAE 补充材料

在讲解 A2C 算法时，前面的课程部分介绍了优势函数（Advantage Function）这个概念，本文首先简单进行回顾：

$$\nabla J_\theta = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=0}^{T_n-1}\boxed{G_t(\tau)}\,\nabla log p_\theta(a_t|s_t) \qquad \text{+ no bias} \qquad \text{- higher variance}$$

$$\nabla J_\theta = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=0}^{T_n-1}\boxed{Q_\phi(s_t, a_t)}\,\nabla log p_\theta(a_t|s_t) \qquad \text{- not unbiased} \qquad \text{+ lower variance}$$

$$\nabla J_\theta = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=0}^{T_n-1}\boxed{(G_t(\tau) - V_\phi(s_t))}\,\nabla log p_\theta(a_t|s_t) \qquad \text{+ no bias} \qquad \text{+ lower variance}$$

可以用 Advantage 函数来表示

图 1： 优势函数概念回顾

- 第一个公式是 REINFORCE 方法的更新公式，它的优点在于对梯度的估计是无偏的，但方差较高。
- 第二个公式是 Actor-Critic 方法的更新公式，它的优点在于对梯度估计的方差较小，但不是无偏估计。
- 第三个公式就是加入了优势函数的 A2C 方法，既保证了梯度估计的无偏性，又能尽可能减小方差

然而需要说明的是，在实践中，这一优势函数的方差仍然较大，因此往往不直接使用这个优势函数进行优化，而是需要对此优势函数进行估计，这接下来就涉及到如何估计的问题了。在这里，本文介绍三种常用的估计方法：N-step，GAE (Generalized Advantage Estimation) [1], V-Trace [2] 。

N-step

N-step 的计算公式如下：

$$\hat{V}_t^{(N)} = \sum_{i=t}^{t+N-1} \gamma^{i-t} r_i + \gamma^N \, V_\phi(s_{t+N})$$
$$\hat{A}_t^{(N)} = \hat{V}_t^{(N)} - V_\phi(s_t)$$

这个方法本质上就是对价值函数计算一个 N 步的 TD 损失，需要权衡梯度的偏差和方差。一般来说，N 越大，梯度的偏差越小，方差越大；反之 N 越小，梯度的偏差越大，方差越小。当 N 趋于无穷大时，上述的 N-step 估计方法就完全等价于 A2C 公式中的优势函数，即：

$$\hat{A}_t^{(\infty)} = G_t(\tau) - V_\phi(s_t)$$

在这种条件下，可以保证梯度的计算完全是无偏的。从对 N-step 的介绍可以看出，当取不同的 N 时，可以得到各有优缺点的优势函数估计。那么一个问题就自然产生了：如何综合这些估计，从而得到更好的结果？这就引出了接下来的算法。

GAE

GAE 就是解决这个问题的一种方法，计算公式如下：

$$\hat{V}_t^{GAE(\lambda)} = (1-\lambda)\sum_{N>0}\lambda^{N-1}\hat{V}_t^{(N)} \quad ,0<\lambda<1$$
$$\hat{A}_t^{GAE(\lambda)} = \hat{V}_t^{GAE(\lambda)} - V_\phi(s_t)$$

简单来说，GAE 就是对取不同 N 的 N-step 估计进行了一次加权平均：对于 $\hat{V}_t^{(N)}$ 乘上了加权系数 λ^{N-1} ，这样一来求和号内就近似可以看作一个等比数列求和，因此需要在求和号前乘上 $1-\lambda$ 系数进行归一化。

可以看出，当 λ 较小时， $\hat{V}_t^{GAE(\lambda)}$ 中占主要成分的是 N 较小的 $\hat{V}_t^{(N)}$ ，因此偏差更大，方差更小；反之当 λ 较大时，偏差更小，方差更大。而在极端情况下，当 $\lambda=0$ 时，方法就退化为了 N-step 中 N 为 1 的特殊情况；而当 $\lambda=1$ 时，方法则退化为 N-step 中 N 无穷大的特殊情况。具体证明如下.

当 $\lambda=0$ 时，有：

$$\begin{aligned}\lim_{\lambda\to 0}\hat{A}_t^{GAE(\lambda)} &= \lim_{\lambda\to 0}\lambda^0\hat{V}_t^{(1)} - V_\phi(s_t) \\ &= \hat{V}_t^{(1)} - V_\phi(s_t) = \hat{A}_t^{(1)}\end{aligned}$$

当 $\lambda=1$ 时，有：

$$\begin{aligned}\lim_{\lambda\to 1}\hat{A}_t^{GAE(\lambda)} &= \lim_{\lambda\to 1}(1-\lambda)\sum_{N>0}\lambda^{N-1}\hat{V}_t^{(N)} - V_\phi(s_t) \\ &= \lim_{\lambda\to 1}(1-\lambda)\sum_{N>0}\lambda^{N-1}\hat{V}_t^{(N)} - V_\phi(s_t)\end{aligned}$$

记 $\delta_t = -V_\phi(s_t) + r_t + V_\phi(s_{t+1})$ ，容易证明： $\hat{V}_t^{(N)} = \sum_{i=1}^{N}\gamma^{i-1}\delta_{t+i-1} + V_\phi(s_t)$ ，带入上式可得：

$$
\begin{aligned}
&\lim_{\lambda\to 1}(1-\lambda)\sum_{N>0}\lambda^{N-1}\sum_{i=1}^{N}\gamma^{i-1}\delta_{t+i-1}+V_\phi(s_t)\ -\ V_\phi(s_t)\\
&=\lim_{\lambda\to 1}(1-\lambda)\sum_{N>0}\lambda^{N-1}\sum_{i=1}^{N}\gamma^{i-1}\delta_{t+i-1}\ +(1-\lambda)\sum_{N>0}\lambda^{N-1}V_\phi(s_t)\ -\ V_\phi(s_t)\\
&=\ \lim_{\lambda\to 1}(1-\lambda)\sum_{N>0}\lambda^{N-1}\sum_{i=1}^{N}\gamma^{i-1}\delta_{t+i-1}\ +(1-\lambda)\lim_{N\to\infty}\frac{1-\lambda^N}{1-\lambda}V_\phi(s_t)\ -\ V_\phi(s_t)\\
&=\ \lim_{\lambda\to 1}(1-\lambda)\sum_{N>0}\lambda^{N-1}\sum_{i=1}^{N}\gamma^{i-1}\delta_{t+i-1}\ +(1-\lambda)\frac{1}{1-\lambda}V_\phi(s_t)\ -\ V_\phi(s_t)\\
&=\ \lim_{\lambda\to 1}(1-\lambda)\sum_{N>0}\lambda^{N-1}\sum_{i=1}^{N}\gamma^{i-1}\delta_{t+i-1}\\
&=\lim_{\lambda\to 1}\lim_{N_0\to\infty}(1-\lambda)\sum_{i=1}^{N_0}\gamma^{i-1}\lambda^{i-1}(\lim_{N_1\to\infty}\sum_{j=0}^{N_1}\lambda)\delta_{t+i-1}\\
&=\ \lim_{\lambda\to 1}\lim_{N_0\to\infty}(1-\lambda)\sum_{i=1}^{N_0}\gamma^{i-1}\lambda^{i-1}\frac{1}{1-\lambda}\delta_{t+i-1}\\
&=\ \lim_{\lambda\to 1}\lim_{N_0\to\infty}\sum_{i=1}^{N_0}\gamma^{i-1}\lambda^{i-1}\delta_{t+i-1}\\
&=\ \sum_{i=1}^{\infty}\gamma^{i-1}\delta_{t+i-1}\ =\ \hat{V}_t^{(\infty)}-V_\phi(s_t)\ =\ \hat{A}_t^{(\infty)}
\end{aligned}
$$

V-trace

接下来，本文再介绍一种可以适用于 off-policy 算法的优势函数估计方法 V-trace 。当收集数据的策略和当前需要更新的策略不同时，直接使用上述方法进行更新会导致偏差。因此，V-trace 在原本的基础上参考了重要性采样 (Importance Sampling) 的技术，使得算法能适用于 off-policy 的情况。

接下来是关于 V-trace 计算部分的介绍。假设收集数据时使用的策略是 μ ，当前需要更新的策略是 π 。则 V-trace 有如下的计算公式：

$$
\hat{V}_t^{\mathrm{VTrace}(\lambda,\bar{\rho},\bar{c})}=V_\phi(s_t)+\sum_{i\ge t}\gamma^{i-t}(\prod_{j=t}^{i-1}c_j)\rho_i(r_i+\gamma V_\phi(s_{i+1})-V_\phi(s_i))
$$

其中 ρ_i 和 c_j 是重要性权重，定义如下：

$$
\rho_i=\min\ (\bar{\rho},\ \frac{\pi(a_i|s_i)}{\mu(a_i|s_i)})
$$
$$
c_i=\min\ (\bar{c},\ \frac{\pi(a_i|s_i)}{\mu(a_i|s_i)})
$$

不难发现，当回到 on-policy 的情况下时（即：$\pi(a_i|s_i)=\mu(a_i|s_i)$ 时），只要满足 $\bar{\rho}\ \ge\ 1$ 和 $\bar{c}\ \ge\ 1$ ，就会有：

$$
\hat{V}_t^{\mathrm{VTrace}(\lambda,\bar{\rho},\bar{c})}=V_\phi(s_t)+\sum_{i\ge t}\gamma^{i-t}(r_i+\gamma V_\phi(s_{i+1})-V_\phi(s_i))=G_t(\tau)
$$

这样也就退化到了最标准的优势函数形式。

需要注意的是，在原文 [2] 中，实验均在 $\bar{\rho} = 1$ 和 $\bar{c} = 1$ 的条件下进行。

实验

以下是对比上述三种估计方案效果的实验。更丰富的实验结果可以参考论文 [3]。

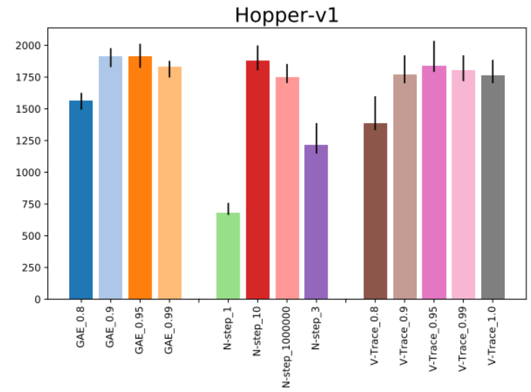
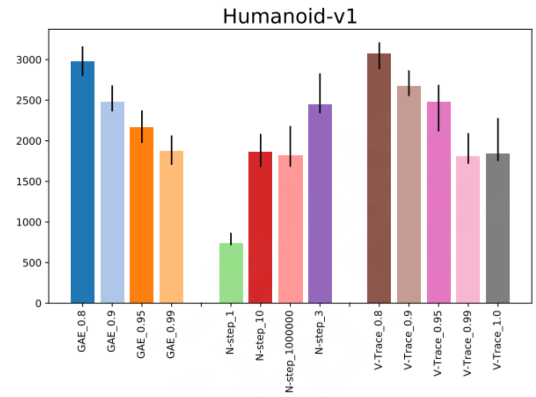
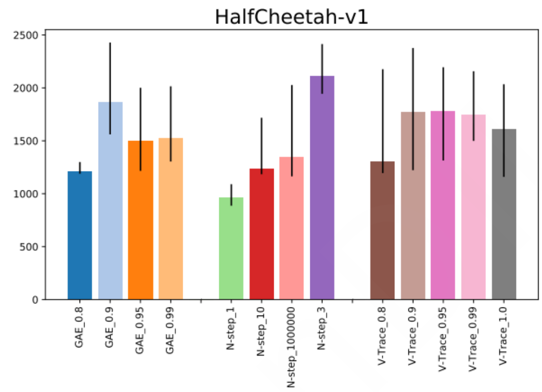
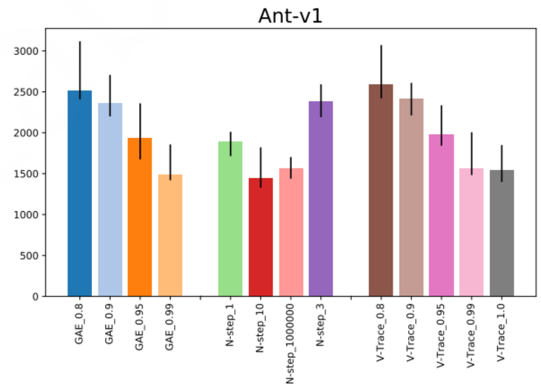
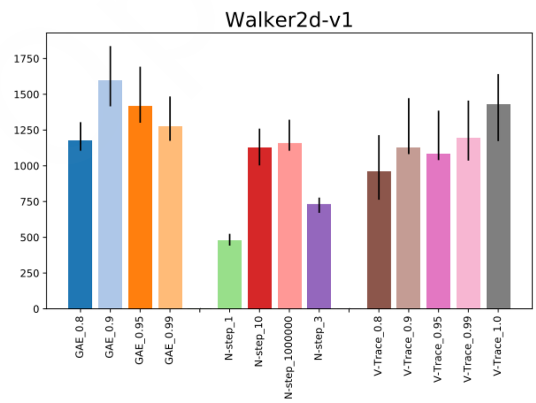
环境	Hopper-v1	Humanoid-v1
实验对比		
环境	HalfCheetah-v1	Ant-v1
实验对比		
环境	Walker-2d-v1	
实验对比		

表 1：三种估计方法在各个环境上的效果对比

可以看出，在多数情况下，GAE 和 V-trace 都能取得比 N-step 更好的结果。因此，在一般情况下，应当优先使用这两种估计方法。

参考文献

[1] HIGH-DIMENSIONAL CONTINUOUS CONTROL USING GENERALIZED ADVANTAGE ESTIMATION. URL: https://arxiv.org/pdf/1506.02438.pdf

[2] IMPALA: Scalable Distributed Deep-RL with Importance Weighted Actor-Learner Architectures. URL: https://arxiv.org/pdf/1802.01561.pdf

[3] What Matters In On-Policy Reinforcement Learning? A Large-Scale Empirical Study. URL: https://arxiv.org/pdf/2006.05990.pdf