Policy - Based . Approach:

Policy Gradient :

Tip1: policy = Actor = Action(observation) $= \pi_\theta(s)$

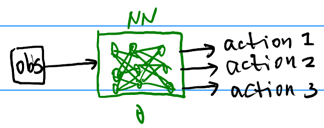
How to learn a actor? 3 Steps.

Step 1 : Define a set of functions:

Step2 : 决定这些函数的好坏

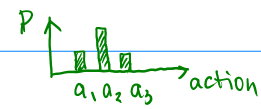
Step 3 : 得到最好的那个函数

Step1: 定义很多的函数



θ是NN的参数

对于Action离散情形, Actor的output是一个关于动作的分布:



P.1

Q: 对于Action连续的情形, Actor的output是什么呢?

Step 2: 确定函数的好坏

用 π_θ 去和 ENV 做互动:

τ^1 : $\{s_1, a_1, r_1, s_2, a_2, r_2, \cdots, s_T, a_T, r_T\}$, Total Reward: $R_\theta(\tau^1) = \sum_{t=1}^{T} r_t$

但 R_θ 是个随机变量, 不足以用来衡量 Actor/policy 的好坏. 于是我们用一个统计平均值去衡量 Actor的好坏, 以此降低随机性. 是让 π_θ 去玩 N 次游戏:

τ^1 : $\{s_1, a_1, r_1, s_2, a_2, r_2, \cdots, s_T, a_T, r_T\}$, Total Reward: $R_\theta(\tau^1) = \sum_{t=1}^{T} r_t$

τ^2 : $\{s_1, a_1, r_1, s_2, a_2, r_2, \cdots, s_T, a_T, r_T\}$, Total Reward: $R_\theta(\tau^2) = \sum_{t=1}^{T} r_t$

⋮

τ^N : $\{s_1, a_1, r_1, s_2, a_2, r_2, \cdots, s_T, a_T, r_T\}$, Total Reward: $R_\theta(\tau^N) = \sum_{t=1}^{T} r_t$

$$\frac{1}{N}\sum_{n=1}^{N} R_\theta(\tau^n) \approx \bar{R}_\theta = \sum_{\tau} R_\theta(\tau) P(\tau|\theta)$$

P.2

Step 3. 得到最好的那个函数:

目标: $\theta^* = \arg\max_\theta \bar{R}_\theta = \arg\max_\theta \sum_\tau R_\theta(\tau) p(\tau|\theta)$

怎么做: Gradient Ascent:

θ^0 开始 → 玩N把游戏 (N个episod)

$\theta^1 = \theta^0 + \eta \cdot \nabla \bar{R}_{\theta^0}$ → 玩N把游戏 (N个episod)

$\theta^2 = \theta^1 + \eta \cdot \nabla \bar{R}_{\theta^1}$ → 玩N把游戏 (N个episod)

⋮

$\theta^* = \theta^{*-1} + \eta \cdot \nabla \bar{R}_{\theta^{*-1}}$ ⓪

其中 $\nabla \bar{R}_\theta$ 怎么算?

Q: $\bar{R}_\theta \approx \frac{1}{N}\sum_{n=1}^{N} R_\theta(\tau^n)$, 不能对θ求偏导吗?

不知道应该怎么求?

这里写τ,而不是τ^n是因为这里是理论分析,要对所有的Trajectory τ 都要求和。

$$\bar{R}_\theta = \sum_\tau R(\tau)\cdot P(\tau|\theta)$$

$$\nabla \bar{R}_\theta = \sum_\tau R(\tau) \nabla P(\tau|\theta)$$

$$= \sum_\tau R(\tau)\cdot P(\tau|\theta)\cdot \frac{\nabla P(\tau|\theta)}{P(\tau|\theta)}$$

Q: R(τ) 和 θ 无关? 怎么理解.

当τ给定, R(τ)由环境决定, 与θ无关

$$= \sum_\tau R(\tau) P(\tau|\theta) \nabla \log P(\tau|\theta)$$

Q: $\nabla P(\tau|\theta)$ 不能直接算吗?

不知道怎么算.

玩N个episodes

$$= E_{\tau\sim p(\tau|\theta)}[R(\tau) \nabla \log P(\tau|\theta)]$$

$$\approx \frac{1}{N}\sum_{n=1}^{N}[R(\tau^n) \nabla \log P(\tau^n|\theta)] \quad ①$$

Q: 为什么要强行化成log的形式?

方便后面消掉环境的那几项.

$\nabla \log P(\tau^n|\theta)$ 怎么算?

$\tau: \{s_1, a_1, r_1, s_2, a_2, r_2, \cdots, s_T, a_T, r_T\}$

$$P(\tau|\theta) = p(s_1)\cdot \overset{\text{Actor}}{p(a_1|s_1,\theta)}\cdot \overset{\text{ENV}}{p(r_1,s_2|s_1,a_1)}\cdot \overset{\text{Actor}}{p(a_2|s_2,\theta)}\cdot \overset{\text{ENV}}{p(r_2,s_3|s_2,a_2)} \cdots \overset{\text{Actor}}{p(a_T|s_T,\theta)}\cdot \overset{\text{ENV}}{p(r_T,s_{T+1}|s_T,a_T)}$$

$$= p(s_1)\prod_{t=1}^{T} p(a_t|s_t,\theta)\cdot p(r_t,s_{t+1}|s_t,a_t)$$

同样的意思：

$$P(\tau^n|\theta) = P_\theta(\tau^n)$$

$$\log P(\tau^n|\theta) = \log p(s_1) + \sum_{t=1}^{T} \log p(a_t|s_t,\theta) + \sum_{t=1}^{T} \log p(r_t, s_{t+1}|s_t,a_t)$$

进而求梯度：

$$\nabla \log P(\tau^n|\theta) = \nabla \left(\cancel{\log p(s_1)} + \sum_{t=1}^{T} \log p(a_t|s_t,\theta) + \cancel{\sum_{t=1}^{T} \log p(r_t, s_{t+1}|s_t,a_t)} \right)$$

去掉了环境的因素，简化了好多

$$= \sum_{t=1}^{T} \nabla \log p(a_t|s_t,\theta) \qquad ②$$

将 ② 代入 ① 可以得到：

$$\nabla \bar{R}_\theta \approx \frac{1}{N} \sum_{n=1}^{N} \left[R(\tau^n) \nabla \log P(\tau^n|\theta) \right] = \frac{1}{N} \sum_{n=1}^{N} \left[R(\tau^n) \sum_{t=1}^{T} \nabla \log p(a_t^n|s_t^n,\theta) \right]$$

$$= \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T} \underset{\text{统计得到}}{R(\tau^n)} \nabla \log \underset{\text{通过NN得到}}{p(a_t^n|s_t^n,\theta)}$$

$$\approx \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T} \left[R_\theta(\tau^n) - \underset{\text{baseline}}{b} \right] \nabla \log p(a_t^n|s_t^n,\theta) \qquad ③$$

Q：添加 Baseline 是为了使得 $[R_\theta(\tau^n) - b]$ 有正有负，为什么要这样设计？不要可以吗？

b 是多少需要设计，$R_\theta(\tau^n) - b$，给它一个新的名字

Advantage Function：

$$A^\theta(s_t, a_t) = R_\theta(\tau^n) - b$$

代表的意义是：在 s_t 下采取 a_t 相比其它动作有多好。

$$= \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T} A^\theta(s_t, a_t) \nabla \log p(a_t^n|s_t^n,\theta)$$

$$\nabla \log f(x) = \frac{1}{f(x)} \cdot \nabla f(x)$$

$$\nabla \bar{R}_\theta \approx \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T} \left[R(\tau^n) - b \right] \nabla \log p(a_t^n|s_t^n,\theta)$$

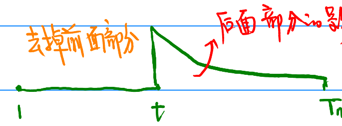
$$= \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T} \left[\sum_{t'=t}^{T_n} \gamma^{t'-t} r_{t'}^n - b \right] \nabla \log p(a_t^n|s_t^n,\theta)$$

Tip：

$R(\tau^n) = \sum_{t=1}^{T_n} r_t^n$，在实际中是用的

$$R(\tau^n) = \sum_{t'=t}^{T_n} \gamma^{t'-t} r_{t'}^n = G_t^n$$

去掉前面部分



后面部分的影响逐渐减弱。

why？

	Before			After		
	t=1	t=2	t=3	t=1	t=2	t=3
$\gamma=1$, b=0	(s_1, a_1, +5,	s_2, a_2, 0,	s_3, a_3, −2)	(s_1, a_1, +5,	s_2, a_2, 0,	s_3, a_3, −2)
	A=3	A=3	A=3	A=3	A=−2	A=−2

Example: $\tau^n \{s_1^n, a_1^n, r_1^n, s_2^n, a_2^n, r_2^n, s_3^n, a_3^n, r_3^n, s_4^n, a_4^n, r_4^n, s_{4+1}^n\}$, $\gamma=0.9$

当 $t=2$ 时, $R_\theta(\tau^n) = \sum_{t'=2}^{4} 0.9^{t'-2} r_{t'}^n = 0.9^0 \cdot r_2^n + 0.9^1 \cdot r_3^n + 0.9^2 \cdot r_4^n$

$$\nabla \bar{R}_\theta \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T}\Big[\overbrace{\boxed{\sum_{t'=t}^{T_n}\gamma^{t'-t} r_{t'}^n}}^{G_t^n} - b\Big]\nabla \log p(a_t^n|s_t^n,\theta)$$

Q-Value Function 的定义:

$q_\pi(s,a) = E_\pi[G_t|S_t, a_t]$

其中: $G_t = \sum_{k=0}^{\infty}\gamma^k r_{t+k+1}$

由于 G_t^n 很不稳定, 是一个随机变量. 于是考虑对其求期望。

$E[G_t^n|S_t,a_t] = Q^{\pi_\theta}(s_t^n,a_t^n)$

值函数 $V(s)$ 从此刻到未来, 回报的期望

值函数 $Q(s,a)$ 代表执行动作 a 后, 未来回报之期望。

b 如何得到: $b = V^{\pi_\theta}(s_t^n)$

Q: Q和V的关系? $E(X) = \sum_i p(x_i)\cdot x_i$

$q_\pi(s,a) = \sum_{r,s'} p(r,s'|s,a)[r + \gamma v_\pi(s')]$

$= E[R_t + \gamma V_\pi(S_{t+1})]$

就有两个网络.

$$\nabla \bar{R}_\theta \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T}\Big[\overset{NN_2}{Q^{\pi_\theta}(s_t^n,a_t^n)} - \overset{NN_3}{V^{\pi_\theta}(s_t^n)}\Big]\nabla \log p(a_t^n|s_t^n,\theta)$$

推到: $Q^{\pi_\theta}(s_t^n,a_t^n) = E(r_t^n + \gamma V^\pi(s_{t+1}^n)) \approx r_t^n + \gamma V^\pi(s_{t+1}^n)$

这里的 $\approx$ 虽然增大了随机性, 但是 $r_t^n + V^\pi(s_{t+1}^n)$ 的随机性相比 G_t^n 会小很多。

于是: $A^\theta(s_t,a_t) = R_\theta(\tau^n) - b \approx \overbrace{\boxed{\sum_{t'=t}^{T_n}\gamma^{t'-t} r_{t'}^n}}^{G_t^n} - b$

$\approx E[G_t^n] - b = Q^{\pi_\theta}(s_t^n,a_t^n) - b = Q^{\pi_\theta}(s_t^n,a_t^n) - V^{\pi_\theta}(s_t^n)$

$= E(r_t^n + V^\pi(s_{t+1}^n)) - V^{\pi_\theta}(s_t^n) \approx r_t^n + V^\pi(s_{t+1}^n) - V^{\pi_\theta}(s_t^n)$

Q: $Q(s_t,a_t) = E(r_t^n + V^\pi(s_{t+1}^n))$

还是 $E(r_t^n + \gamma \cdot V^\pi(s_{t+1}^n))$?

gamma ✓

V: NN

p: NN

$$\nabla \bar{R}_\theta \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T}\Big[\underbrace{\boxed{r_t^n + V^{\pi_\theta}(s_{t+1}^n) - V^{\pi_\theta}(s_t^n)}}_{\text{Advantage}}\Big]\nabla \log p(a_t^n|s_t^n,\theta)$$

④

流程:

π_θ 去和环境互动

TD/MC

$\pi_\theta = \pi_{\theta'}$

更新 $\pi_\theta \to \pi_{\theta'}$ ← ⑦ 学习 $V^{\pi_\theta}(s)$

Actor　　Critic

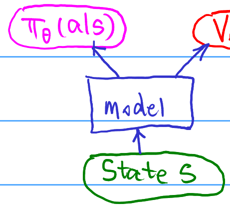
Tip:
和环境作互动的 policy|Actor
就是要学习的 policy|Actor
因此这里方法是 on-policy
的学习方法。

Actor: Policy-Based

critic: Value-Based

因此这个方法叫做: Advantage Actor-Critic : AAC

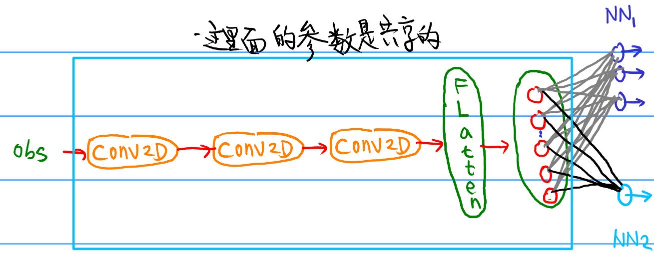
A2C 有两个网络：
$$\begin{cases} \pi_\theta(s) : NN_1 \\ V^{\pi_\theta}(s) : NN_2 \end{cases}$$
两个网络是可以共用参数的。



Q：所谓共用参数，应该不是所有的参数都共用吧？

$\pi_\theta(a|s)$ 和 $V_\theta(s)$ 中的两个θ是完全一样的吗？

不一样，最后一部分不一样。



$$L_{critic} \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T}\left(V_\theta(S_t) - [r + \gamma \cdot V_\theta(S_{t+1})]\right)^2$$

$$\nabla J_{actor} \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T}\Big[\underbrace{\boxed{r_t^n + V^{\pi_\theta}(S_{t+1}^n) - V^{\pi_\theta}(S_t^n)}}_{\text{Advantage}}\Big]\nabla \log p(a_t^n | S_t^n, \theta)$$

Tip：V是通过TD的方法得到的

Q：Off-Policy 相比 on-Policy 的优点是什么？

Q：实际当中还会将一个entropy项添加到loss项中，为什么？

Q：输出是怎么处理的？

NN_1 的输出 → out1, out2, out3

$$\text{policy} = \frac{[e^{out_1}, e^{out_2}, e^{out_3}]}{\sum_{i=1}^{3} e^{out_i}}$$

概率分布

a= np.random.choice([0,1,2], p= policy)

Q：实际当中A3C是怎么处理的？
多个worker是怎么协同工作的。

Q：请再仔细描述一下整个A3C的流程？

Q：A3C擅长处理什么？
又有什么局限？

Q：相比ppo，A3C哪些方面比较弱？