598 RL

Fall 2020

Policy Gradient

BRETL

POLICY GRADIENT - RUNNING EXAMPLE

left, right

$S = \{0, 1\}$

stay still, move

$A = \{0, 1\}$

with probability $\beta \in [0, 1]$ we take the action that we did not intend

rewards

if we end up in front of the left arm

+10

otherwise

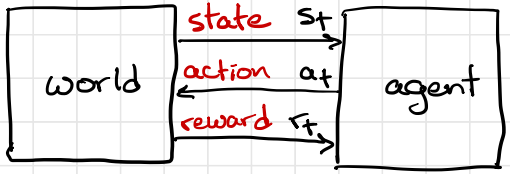
+0

if $s_0 = 0$ and $a_0 = 0$

with prob $1-\beta$, $s_1 = 0$ and $r_0 = 10$

with prob β, $s_1 = 1$ and $r_0 = 0$

MODEL



$p(s_0)$ = probability that the initial state is s_0

$p(s_{t+1} \mid s_t, a_t)$ = probability that the next state is s_{t+1} given that the state is s_t and the agent takes action a_t

often called $\pi_\theta(a_t|s_t)$

$p(a_t \mid s_t ; \theta)$ = probability that the agent takes action a_t given that the state is s_t

POLICY

it is parameterized by θ

$r_t = r(s_t, a_t)$ = mean reward for taking action a_t at state s_t

Tabular policy for finite state and action space

θ_{as} = weight of action a in state s

θ_{00} ← state 0, action 0

$\theta_{00}, \theta_{10}, \theta_{01}, \theta_{11}$

$$p(a|s;\theta) = \frac{e^{\theta_{as}}}{\sum_{a'=1}^{n_a} e^{\theta_{a's}}}$$

assume the policy is time-invariant

softmax – exponentiate to make sure it is positive, then normalize

~~$p(a|s;\theta) = \theta_{as}$~~

~~s.t. $\theta_{as} \in [0,1]$ and $\sum_{a'} \theta_{a's} = 1$~~

$$p(a=0|s=0;\theta) = \frac{e^{\theta_{00}}}{e^{\theta_{00}} + e^{\theta_{10}}}$$

$$p(a=1|s=0;\theta) = \frac{e^{\theta_{10}}}{e^{\theta_{00}} + e^{\theta_{10}}}$$

$$p(a=0 \mid s=0;\theta) + p(a=1 \mid s=0;\theta) = 1$$

$$\nabla_\theta (\qquad) = \nabla_\theta (1) = 0$$

$$\nabla_\theta p(a=0 \mid s=0;\theta) + \nabla_\theta p(a=1 \mid s=0;\theta) = 0$$

GOAL

$$\tau = (s_0, a_{01} \cdots, s_{T-1}, a_{T-1}, s_T) \quad \leftarrow \text{trajectory}$$

$$p(\tau;\theta) = p(s_0) \prod_{t=0}^{T-1} p(a_t|s_t;\theta)\, p(s_{t+1}|s_t, a_t)$$

↑ probability of generating this trajectory with a given policy

PAYOFF

$$J(\theta) = E_{\tau \sim p(\tau;\theta)} \left[\underbrace{\sum_{t=0}^{T-1} r(s_t, a_t)}_{r(\tau)\ \text{total reward}} \right]$$

WE WANT TO MAXIMIZE THIS

take expectation wrt the distribution $p(\tau;\theta)$ on the space of trajectories

$$p(\tau;\theta) = p(s_0)\, p(a_0|s_0;\theta)\, p(s_1|s_0,a_0)\, p(a_1|s_1;\theta)$$

$$\nabla_\theta p(\tau;\theta) = p(s_0)\, \nabla_\theta p(a_0|s_0;\theta)\, p(s_1|s_0,a_0)\, p(a_1|s_1;\theta) + p(s_0)\, p(a_0|s_0;\theta)\, p(s_1|s_0,a_0)\, \nabla_\theta p(a_1|s_1;\theta)$$

METHOD - POLICY GRADIENT

we want to find

$$\theta^* = \arg\max_{\theta} J(\theta)$$

we can do so by gradient ascent

$$\theta_{k+1} = \theta_k + \alpha \boxed{\nabla_\theta J(\theta_k)}$$

← NEED TO COMPUTE THIS

↑ learning rate (step size)

$$J(\theta) = E_{\tau \sim p(\tau;\theta)}\left[r(\tau)\right]$$

$$= \sum_{\tau} r(\tau)\, p(\tau;\theta)$$

- $p(\tau;\theta)$: probability of generating this trajectory
- $r(\tau)$: total reward for this trajectory
- $\sum_{\tau}$: sum over all possible trajectories

$$\nabla_\theta J(\theta) = \sum_{\tau} r(\tau)\, \nabla_\theta p(\tau;\theta)$$

- $r(\tau)$: how good is τ?
- $\nabla_\theta p(\tau;\theta)$: what change in θ would make the probability of τ increase the fastest?

$(s_0, a_0, s_1, a_1) \rightarrow$ 16 possible trajectories

PROBLEM #1 $\nabla_\theta J(\theta) = \sum_\tau r(\tau) \nabla_\theta p(\tau;\theta)$

This does not have the form

$$\sum_\tau f(\tau)\ p(\tau;\theta) \leftarrow E_{\tau \sim p(\tau;\theta)}\left[f(\tau) \right]$$

so it can't be approximated by sampling τ with θ as

$$\frac{1}{N} \sum_{i=1}^{N} f(\tau^i).$$

Instead, τ must be sampled uniformly, and so $r(\tau)$ is likely to be very small.

PROBLEM #2

$$\nabla_\theta J(\theta) = \sum_\tau r(\tau)\, \nabla_\theta p(\tau;\theta)$$

$$p(\tau;\theta) = p(s_0, a_0, \ldots, s_{T-1}, a_{T-1}, s_T; \theta)$$

$$= p(s_0) \prod_{t=0}^{T-1} p(a_t|s_t;\theta)\, p(s_{t+1}|s_t, a_t)$$

$$\nabla_\theta p(\tau;\theta) = p(s_0) \sum_{t=0}^{T-1} \left(\left(\prod_{\substack{k=0 \\ k\neq t}}^{T-1} p(a_k|s_k;\theta)\, p(s_{k+1}|s_k, a_k) \right) \cdot \nabla_\theta p(a_t|s_t;\theta)\, p(s_{t+1}|s_t, a_t) \right)$$

this is a long product of probabilities, which is also likely to be very small (and numerically bad)

if only we were working with log-probabilities ...

$$\log p(\tau;\theta) = \log\left(p(s_0)\prod_{t=0}^{T-1} p(a_t|s_t;\theta)\,p(s_{t+1}|s_t,a_t)\right)$$

$$= \log p(s_0) + \sum_{t=0}^{T-1}\left(\log p(a_t|s_t;\theta) + \log p(s_{t+1}|s_t,a_t)\right)$$

then ...

$$\nabla_\theta \log p(\tau;\theta) = \sum_{t=0}^{T-1} \nabla_\theta \log p(a_t|s_t;\theta)$$

this is a sum and not a product and so will be much better behaved

it also doesn't need $p(s_{t+1}|s_t,a_t)$!!

BEHOLD!

$$\nabla_\theta \log p(\tau;\theta) = \frac{1}{p(\tau;\theta)} \nabla_\theta p(\tau;\theta)$$

what we want to find

what we are supposed to find

$$\Downarrow$$

$$\nabla_\theta p(\tau;\theta) = p(\tau;\theta)\, \nabla_\theta \log p(\tau;\theta)$$

$$\Downarrow$$

$$\begin{aligned}
\nabla_\theta J(\theta) &= \sum_\tau r(\tau)\, \nabla_\theta p(\tau;\theta) \\
&= \sum_\tau \left(r(\tau)\, \nabla_\theta \log p(\tau;\theta) \right) p(\tau;\theta) \\
&= \mathbb{E}_{\tau \sim p(\tau;\theta)} \left[r(\tau)\, \nabla_\theta \log p(\tau;\theta) \right]
\end{aligned}$$

can be computed by sampling τ with θ

THE PAYOFF GRADIENT

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim p(\tau;\theta)} \left[\left(\sum_{t=0}^{T-1} \nabla_\theta \log p(a_t|s_t;\theta) \right) \left(\sum_{t=0}^{T-1} r(s_t, a_t) \right) \right]$$

$$\approx \frac{1}{N} \sum_{i=1}^{N} \left(\sum_{t=0}^{T-1} \nabla_\theta \log p(a_t^i|s_t^i;\theta) \right) \left(\sum_{t=0}^{T-1} r(s_t^i, a_t^i) \right)$$

$\tau^i = (s_0^i, a_{0,} \cdots, s_{T-1}^i, a_{T-1,}^i s_T^i)$ is generated with θ

$$\ell(\theta) = \frac{1}{N} \sum_{i=1}^{N} \sum_{t=0}^{T-1} \log p(a_t^i|s_t^i;\theta) r(\tau^i)$$

$$\approx \frac{1}{N} \sum_{i=1}^{N} \nabla_\theta \log p(\tau^i;\theta)\, r(\tau^i)$$

$$\approx \frac{1}{N} \sum_{i=1}^{N} \sum_{t=0}^{T-1} \nabla_\theta \log p(a_t^i|s_t^i;\theta)\, r(\tau^i)$$

REINFORCE algorithm

1. sample $\tau^1, \dots, \tau^N$ by running the policy $p(a_t|s_t;\theta)$
2. estimate $\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_{i=1}^{N} \left(\sum_{t=0}^{T-1} \nabla_\theta \log p(a_t^i|s_t^i;\theta) \right) \left(\sum_{t=0}^{T-1} r(s_t^i, a_t^i) \right)$
3. update $\theta_{k+1} = \theta_k + \alpha \nabla_\theta J(\theta_k)$

repeat until convergence

how to compute this?