The Stochastic Policy Gradient

michele.garibbo

October 2023

1 Background

We can distinigiush 3 broad approaches to policy gradient in the RL literature: Stochastic Policy Gradient (SPG), Deterministic Policy Gradient (DPG) and, more broadly, model-based policy gradient (e.g., Stochstic Value gradient Heess et al., 2015). Here, we focus on the Stochastic Policy Gradient (SPG).

1.1 Stochastic Policy Gradient

The policy gradient objective can be defined as,

$$J(\phi) = V^{\pi_\phi}(s) \tag{1}$$

for $s \in S$. Namely, we want to maximise the value function across all possible states (note, a similar objective can be defined in terms of the expected value function across all visited states). For better clarity, I use the notation $V^\pi = V^{\pi_\phi}$ from now on. By taking the gradient of this objective, we get:

$$\begin{align}
\nabla_\phi J(\phi) &= \nabla_\phi V^\pi(s) \tag{2}\\
&= \nabla_\phi \left[\sum_a \pi_\phi(a, s) Q^\pi(s, a) \right] \tag{3}\\
&= \sum_a \left(\nabla_\phi \pi_\phi(s, a) Q^\pi(s, a) + \pi_\phi(a, s) \nabla_\phi Q^\pi(s, a) \right) \tag{4}
\end{align}$$

The term $\nabla_\phi Q^\pi(s, a)$ is actually hard to compute explictily. This is because the policy parameters ϕ not only affect the current action a, but affect all subsequent visited states (i.e., through the policy), which in turn affect Q itself (i.e., since Q is recursively defined in terms of the values at future states - i.e., $Q(s,a) = r(s,a) + \sum_{s'} p(s' \mid s,a) V^{\pi}(s')$). As we will see in the following proof, there us no need to explicitly compute this gradient term during **on-policy learning**. Conversely, we will be able to indirectly estimate $\nabla_{\phi} Q^{\pi}(s,a)$ by sampling policy gradient updates along the on-policy state distribution. Let's continue (for simplify I assume the reward is a deterministic function of state-action pairs),

$$\nabla_{\phi} V^{\pi}(s) = \sum_{a} \left(\nabla_{\phi} \pi_{\phi}(s,a) Q^{\pi}(s,a) + \pi_{\phi}(s,a) \nabla_{\phi} Q^{\pi}(s,a) \right) \tag{5}$$

$$= \sum_{a} \left(\nabla_{\phi} \pi_{\phi}(s,a) Q^{\pi}(s,a) + \pi_{\phi}(s,a) \nabla_{\phi} \left(\sum_{s',r} p(r,s' \mid s,a)(r + V^{\pi}(s')) \right) \right) \tag{6}$$

The second term in the main sum, requires computing the expected gradient of the reward (relative to ϕ) plus the expected gradient of the next state value. Nevertheless, the the expected gradient of the reward relative to the (policy) parameter is zero, simplifying things to,

$$= \sum_{a} \left(\nabla_{\phi} \pi_{\phi}(s,a) Q^{\pi}(s,a) + \pi_{\phi}(s,a) \left(\sum_{r,s'} p(r,s' \mid s,a) \nabla_{\phi} V^{\pi}(s') \right) \right) \tag{7}$$

where I moved the gradient operator inside the sum and over the transition distribution, which is independent of the policy parameters. It may not be immediately clear why the expected gradient of the reward relative to the (policy) parameter is zero, since, on an intuitive level, the reward depends on the actions, which in turn, depend on the policy parameters ϕ. To understand why this is the case see section 3. Next, we can marginalises the distribution $p(r,s' \mid s,a)$ over the sum of rewards, leading to,

$$\nabla_{\phi} V^{\pi}(s) = \sum_{a} \left(\nabla_{\phi} \pi_{\phi}(s,a) Q^{\pi}(s,a) + \pi_{\phi}(s,a) \left(\sum_{s'} p(s' \mid s,a) \nabla_{\phi} V^{\pi}(s') \right) \right) \tag{8}$$

we can further unroll the gradient of state-value term $\nabla_{\phi} V^{\pi}(s')$,

$$= \sum_{a} \left(\nabla_{\phi} \pi_{\phi}(s,a) Q^{\pi}(s,a) + \pi_{\phi}(s,a) \left(\sum_{s',a'} p(s') \left(\nabla_{\phi} \pi_{\phi}(s',a') Q^{\pi}(s',a') + \pi_{\phi}(s',a') \left(\sum_{s''} p(s'') \nabla_{\phi} V^{\pi}(s'') \right) \right) \right) \right) \tag{9}$$

where I literally applied the definition of $\nabla_{\phi} V^{\pi}(s)$ found in eq. 8 to $\nabla_{\phi} V^{\pi}(s')$, merging the sums over the next states and actions. In order to fit the equation in one line I drop the conditional on the transition functions (e.g., $p(s'') = p(s'' \mid s', a')$. Based on eq. 9, we can see that if we further unroll this equation (i.e., by expanding $\nabla_{\phi} V^{\pi}(s'')$), the 'problematic' gradient term $\nabla_{\phi} V^{\pi}$ get pushed away.

By making some considerations about the recursive relation in eq. 9, we can formally re-write it without the problematic gradient term by unrolloing it for ∞ many steps. In eq. 9, for any given next state s', the terms $\sum_a \pi_\phi(s,a)(p' \mid s,a)$ denotes the probability of transition from a state s to a state s' over one time step based on the policy, $p^\pi(s \to s', 1)$. This term gets multiplied by the probability of $p^\pi(s' \to s'', 1)$ for all possible s'. Hence, it is computing the probability of going from s to s'' over two time steps, $p^\pi(s \to s'', 2)$. As we unroll, V^π, we can see how this probability terms expand over multiple time-steps - e.g., $p^\pi(s \to x, k)$ for k steps. Given some future state x, these probability terms $p^\pi(s \to x, k)$ weight the corresponding gradient term, $\sum_a \nabla_\phi \pi_\phi(x,a) Q^\pi(x,a)$. Hence, we can re-write eq. 9 as,

$$\nabla_\phi V^\pi(s) = \sum_{x \in S} \sum_k^\infty p^\pi(s \to x, k) \sum_a \nabla_\phi \pi_\phi(x,a) Q^\pi(x,a) \tag{10}$$

Since, we are unrolling the recursive relation over ∞ steps, we never need to compute the problematic gradient term, $\nabla_\phi V^\pi$ for any future state x. Now, the term $\sum_k^\infty p^\pi(s \to x, k)$ may look intimidating, but it is just the unnormalized probability of visiting a state x under the policy π (i.e., the unnormalized on-policy state distribution), given we are in the state s. For episodic cases, the on-policy state distribution for a state s can be defined based on the number of times we transition into that state s (from any state $\bar{s}$) plus the initial probability, h of starting in that state, normalised by the number of all visits.

$$\eta(s) = h(s) + \sum_{\bar{s}} \eta(\bar{s}) \sum_a \pi(a \mid \bar{s}) p(s \mid \bar{s}, a) \tag{11}$$

Now, we can normalize this term to get the on-policy state distribution,

$$d^\pi(s) = \frac{\eta(s)}{\sum_s \eta(s)} \tag{12}$$

Then term $\eta(x)$ is exactly what the $\sum_k^\infty p^\pi(s \to x, k)$ computes given we are at a state s (i.e., don't need to sum over all possible routes to x starting at different states from s, plus for $k = 0$ in the sum, we account for the probability of starting in that state x). So, we just need to normalize this term to get the on-policy distribution,

$$\nabla_\phi V^\pi(s) = \frac{\sum_{x \in s} \eta(x)}{\sum_{x \in s} \eta(x)} \sum_{x \in S} \eta(x) \sum_a \nabla_\phi \pi_\phi(x,a) Q^\pi(x,a) \tag{13}$$

$$= \left(\sum_{x \in s} \eta(x) \right) \sum_{x \in S} \frac{\eta(x)}{\sum_{x \in s} \eta(x)} \sum_a \nabla_\phi \pi_\phi(x,a) Q^\pi(x,a) \tag{14}$$

Since $\sum_{x \in s} \eta(x)$ is a (normalizing) constant we can re-write this as,

$$\tag{15}$$

$$\propto \sum_{x \in S} \frac{\eta(x)}{\sum_{x \in s} \eta(x)} \sum_a \nabla_\phi \pi_\phi(x,a) Q^\pi(x,a) \tag{16}$$

$$= \sum_{x \in S} d^\pi(x) \sum_a \nabla_\phi \pi_\phi(x,a) Q^\pi(x,a) \tag{17}$$

We can now apply the log-trick (e.g., see my notes available at link),

$$= \sum_{x \in S} d^{\pi}(x) \sum_{a} \pi_{\phi}(a \mid x) \nabla_{\phi} \log \pi_{\phi}(x, a) Q^{\pi}(x, a) \tag{18}$$

$$= \mathrm{E}_{\pi} \left[\nabla_{\phi} \log \pi_{\phi}(x, a) Q^{\pi}(x, a) \right] \tag{19}$$

Note if we assume we are in some state s_0, we can re-write this in terms of the usual s notation,

$$\nabla_{\phi} V^{\pi}(s_0) = \mathrm{E}_{\pi} \left[\nabla_{\phi} \log \pi_{\phi}(s, a) Q^{\pi}(s, a) \right] \tag{20}$$

2 Notes on why SPG does apply to off-policy learning

When we estimate SPG by sampling state-action pairs, we are assuming all gradients are computed at states s coming from the target on-policy distribution d^{π}. This is because, in order to avoid computing the complex gradient term $\nabla_{\phi} Q^{\pi}(s, a)$ explicitly, we unrolled $Q^{\pi}(s, a)$ through the $\sum_{k}^{\infty} p^{\pi}(s \to x, k)$ terms (i.e., the unnormalized on-policy state distribution). Particularly, the $\sum_{k}^{\infty} p^{\pi}(s \to x, k)$ terms assume the states we will visit come from the state distribution induced by target policy π_{ϕ}. This is very important because it is the reason why we can not apply SPG to off-policy learning and, instead, we must to rely on an approximate gradient computation for off-policy learning of stochastic target policies. In off-policy learning, the states are sampled according to behavioural policies, β instead of the target policy, π_{ϕ} for which we need to compute the policy gradient. Therefore, we have now way to sample states according to the target policy and avoid computing $\nabla_{\phi} Q^{\pi}(s, a)$ explicitly. In off-policy learning, Degris et al. (2012) propose to just ignore the $\nabla_{\phi} Q^{\pi}$ gradient term, providing some justifications. Namely for some behavioural policy β,

$$\nabla_{\phi} J(\phi) = \nabla_{\phi} \mathrm{E}_{s \sim d^{\beta}} [V^{\pi}(s)] \tag{21}$$

Since d^{β} does not depend on β, we can bring the gradient inside the expectation,

$$= \mathrm{E}_{s \sim d^{\beta}} \left[\sum_{a} \left(\nabla_{\phi} \pi_{\phi}(s, a) Q^{\pi}(s, a) + \pi_{\phi}(s, a) \nabla_{\phi} Q^{\pi}(s, a) \right) \right] \tag{22}$$

Now, we can approximate this gradient by just ignoring the second term in the sum,

$$\approx \mathrm{E}_{s \sim d^{\beta}} \left[\sum_{a} \nabla_{\phi} \pi_{\phi}(s, a) Q^{\pi}(s, a) \right] \tag{23}$$

This gradient assumes the actions come from the target policy π_ϕ. Hence, if we want to evaluate this gradient by sampling actions off-policy, we can use importance sampling

$$= \mathrm{E}_{s \sim d^\beta} \left[\sum_a \frac{\beta(a \mid s)}{\beta(a \mid s)} \nabla_\phi \pi_\phi(s,a) Q^\pi(s,a) \right] \tag{24}$$

$$= \mathrm{E}_\beta \left[\frac{\nabla_\phi \pi_\phi(s,a)}{\beta(s,a)} Q^\pi(s,a) \right] \tag{25}$$

Note, we can also re-write this gradient through the log-trick,

$$\nabla_\phi J(\phi) = \mathrm{E}_\beta \left[\frac{\pi_\phi(s,a)}{\beta(s,a)} Q^\pi(s,a) \nabla_\phi \log \pi_\phi(s,a) \right] \tag{26}$$

3 Notes on why SPG does not apply to deterministic policies

First, note that,

$$\mathrm{E}_{a \sim \pi_\phi}[\nabla_\phi r(s,a)] = 0 \tag{27}$$

for a stochastic policy, π_ϕ. This is because the gradient operator, ∇_ϕ, is taken after the action, a, is sampled. Therefore, a must be treated as constant in ϕ when computing the gradient. In other words, the reward depends on ϕ through the actions, which means that if the gradient operator is applied after sampling the actions, there is no dependency between r and ϕ in the gradient computation. Indeed, eq. 27 is not equivalent to computing, $\nabla_\phi \mathrm{E}_{a \sim \pi_\phi}[r(s,a)]$, where the gradient operator is applied before sampling the actions, a and thus requires accounting for the dependence between r and ϕ through the sampling of a when computing ∇_ϕ (i.e., requirying to use the log-trick or the reparameterization trick).

Now, for a deterministic policy μ_ϕ, there is a dependence between r and ϕ. This is because in that case there is no expectation were actions are sampled before taking the gradient operator, ∇_ϕ. Conversely, actions directly depend on μ_ϕ. Therefore, the ('expected') deterministic policy gradient of the reward is not necessary zero and must be computed through the chain-rule,

$$\mathrm{E}_{\mu_\phi}[\nabla_\phi r(s, \mu_\phi(s))] = \nabla_\phi r(s, \mu_\phi(s)) = \nabla_\phi \mu_\phi(s) \nabla_a r(s,a)|_{a=\mu_\phi(s)} \tag{28}$$

Now, if you recall, in eq. 6, we had the following expression,

$$\nabla_\phi V^\pi(s) = \sum_a \left(\nabla_\phi \pi_\phi(s,a) Q^\pi(s,a) + \pi_\phi(s,a) \nabla_\phi \left(\sum_{s',r} p(r, s' \mid s,a)(r + V^\pi(s')) \right) \right) \tag{29}$$

Here, I want to show step by step why we could get rid of the gradient of the reward, r, relative to the policy parameter, ϕ. First, we can bring the sum over actions inside the parenthesis to get,

$$= \sum_a \nabla_\phi \pi_\phi(s,a) Q^\pi(s,a) + \sum_a \pi_\phi(s,a) \nabla_\phi \left(\sum_{s',r} p(r, s' \mid s, a)(r + V^\pi(s')) \right) \tag{30}$$

Now this can be further expanded to (and marginalising r and s in the respective joint distributions),

$$= \cdots + \sum_a \pi_\phi(s,a) \nabla_\phi \sum_r p(r \mid s,a)\, r + \sum_a \pi_\phi(s,a) \nabla_\phi \sum_{s'} p(s' \mid s,a) V^\pi(s') \tag{31}$$

if we re-organise things around the sum over rewards we get,

$$= \cdots + \sum_r p(r \mid s,a) \sum_a \pi_\phi(s,a)\ \nabla_\phi r + \dots \tag{32}$$

$$= \cdots + \sum_r p(r \mid s,a)\, \mathrm{E}_{a \sim \pi_\phi}[\nabla_\phi r(s,a)] + \dots \tag{33}$$

we know this expectation equals zero and so we are left with,

$$= \sum_a \nabla_\phi \pi_\phi(s,a) Q^\pi(s,a) + 0 + \sum_a \pi_\phi(s,a) \nabla_\phi \sum_{s'} p(s' \mid s,a) V^\pi(s') \tag{34}$$

I think in DPG, we cannot make this step of turning the expected policy gradient of reward to zero, thus requiring a different derivation as reported by Silver et al. (2014). In the stochastic policy gradient after putting everything back together we get eq. 8,

$$= \sum_a \left(\nabla_\phi \pi_\phi(s,a) Q^\pi(s,a) + \pi_\phi(s,a) \left(\sum_{s'} p(s' \mid s,a) \nabla_\phi V^\pi(s') \right) \right) \tag{35}$$

References

Degris, T., White, M., & Sutton, R. S. (2012). Off-policy actor-critic. *arXiv preprint arXiv:1205.4839*.

Heess, N., Wayne, G., Silver, D., Lillicrap, T., Erez, T., & Tassa, Y. (2015). Learning continuous control policies by stochastic value gradients. *Advances in neural information processing systems*, *28*.

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., & Riedmiller, M. (2014). Deterministic policy gradient algorithms. *International conference on machine learning*, 387–395.