Exercise Set 5 - Reinforcement Learning

Advanced policy-based methods

Giulio Starace - 13010840

October 14, 2022

9.4 Homework: Limits of policy gradients

1. Given our policy:

$$\pi(a|s,\theta) = \frac{1}{\sigma\left(\theta_\sigma\right)\sqrt{2\pi}} \exp\left(-\frac{\left(a-\mu\left(\theta_\mu\right)\right)^2}{2\sigma\left(\theta_\sigma\right)^2}\right), \tag{1}$$

We can compute $\nabla \log \pi(a|s,\theta)$ w.r.t. θ_μ and θ_σ by applying the chain rule. Let $\log \pi(a|s,\theta)$ be $L(\theta)$, then w.r.t. a given param θ_i we have:

$$\begin{aligned}\nabla_{\theta_i} \log \pi(a|s,\theta) &= \nabla_{\theta_i} L(\theta) \\ &= \frac{\partial L(\theta)}{\partial \pi} \cdot \frac{\partial \pi(a|s,\theta)}{\partial \theta_i}.\end{aligned}$$

When $\theta_i = \theta_\mu$, we have:

$$\begin{aligned}\nabla_{\theta_\mu} \log \pi(a|s,\theta) &= \frac{\partial L(\theta)}{\partial \pi,} \cdot \frac{\partial \pi(a|s,\theta)}{\partial \theta_\mu} \\ &= \frac{\partial L(\theta)}{\partial \pi(a|s,\theta)} \cdot \frac{\partial \pi(a|s,\theta)}{\partial \mu} \cdot \frac{\partial \mu(\theta_\mu)}{\partial \theta_\mu}.\end{aligned} \tag{2}$$

When $\theta_i = \theta_\sigma$, we have:

$$\begin{aligned}\nabla_{\theta_\sigma} \log \pi(a|s,\theta) &= \frac{\partial L(\theta)}{\partial \pi,} \cdot \frac{\partial \pi(a|s,\theta)}{\partial \theta_\sigma} \\ &= \frac{\partial L(\theta)}{\partial \pi(a|s,\theta)} \cdot \frac{\partial \pi(a|s,\theta)}{\partial \sigma} \cdot \frac{\partial \sigma(\theta_\sigma)}{\partial \theta_\sigma}.\end{aligned} \tag{3}$$

The first two terms of equations (2) and (3) will be the same regardless of parametrization. We get

$$\frac{\partial L(\theta)}{\partial \theta} = \frac{1}{\pi(a|s,\theta)}, \tag{4}$$

$$\frac{\partial \pi(a|s,\theta)}{\partial \mu} = \frac{a-\mu(\theta_\mu)}{\sqrt{2\pi}\sigma(\theta_\sigma)^3} \exp\left[-\frac{(a-\mu(\theta_\mu))^2}{2\sigma(\theta_\sigma)^2}\right] = \frac{a-\mu(\theta_\mu)}{\sigma(\theta_\sigma)^2}\pi(a|s,\theta), \tag{5}$$

$$\frac{\partial \pi(a|s,\theta)}{\partial \sigma} = \frac{(a-\mu(\theta_\mu))^2-\sigma(\theta_\sigma)^2}{\sqrt{2\pi}\sigma(\theta_\sigma)^4} \exp\left[-\frac{(a-\mu(\theta_\mu))^2}{2\sigma(\theta_\sigma)^2}\right] = \frac{(a-\mu(\theta_\mu))^2-\sigma(\theta_\sigma)^2}{\sigma(\theta_\sigma)^3}\pi(a|s,\theta). \tag{6}$$

Equations (2) and (3) can be then further simplified as:

$$\nabla_{\theta_\mu} \log \pi(a|s,\theta) = \frac{a-\mu(\theta_\mu)}{\sigma(\theta_\sigma)^2} \cdot \frac{\partial \mu(\theta_\mu)}{\partial \theta_\mu}. \tag{7}$$

$$\nabla_{\theta_\sigma} \log \pi(a|s,\theta) = \frac{(a-\mu(\theta_\mu))^2 - \sigma(\theta_\sigma)^2}{\sigma(\theta_\sigma)^3} \cdot \frac{\partial \sigma(\theta_\sigma)}{\partial \theta_\sigma}. \tag{8}$$

We are then left with determining the final terms of equations (7) and (8) for different parametrizations.

(a) When $\mu(\theta_\mu) = \theta_\mu$ and $\sigma(\theta_\sigma) = \exp(\theta_\sigma)$, we get

$$\frac{\partial \mu(\theta_\mu)}{\partial \theta_\mu} = 1, \tag{9}$$

$$\frac{\partial \sigma(\theta_\sigma)}{\partial \theta_\sigma} = \exp(\theta_\sigma). \tag{10}$$

We can plug this into equations (7) and (8) along with the updated terms from previously and get:

$$\nabla_{\theta_\mu} \log \pi(a|s,\theta) = \frac{a-\theta_\mu}{\exp^2(\theta_\sigma)} \tag{11}$$

$$\nabla_{\theta_\sigma} \log \pi(a|s,\theta) = \frac{(a-\theta_\mu)^2 - \exp^2(\theta_\sigma)}{\exp^2(\theta_\sigma)}. \tag{12}$$

(b) When $\mu(\theta_\mu) = \theta_\mu$ and $\sigma(\theta_\sigma) = \theta_\sigma^2$, we get

$$\frac{\partial \mu(\theta_\mu)}{\partial \theta_\mu} = 1, \tag{13}$$

$$\frac{\partial \sigma(\theta_\sigma)}{\partial \theta_\sigma} = 2\theta_\sigma. \tag{14}$$

We can once again plug this into equations (7) and (8) along with the updated terms from previously and get:

$$\nabla_{\theta_\mu} \log \pi(a|s,\theta) = \frac{a-\theta_\mu}{\theta_\sigma^4} \tag{15}$$

$$\nabla_{\theta_\sigma} \log \pi(a|s,\theta) = 2 \cdot \frac{(a-\theta_\mu)^2 - \theta_\sigma^4}{\theta_\sigma^5}. \tag{16}$$

2. The policy gradient update for a given parameter θ_i can be computed with

$$\theta_i' = \theta_i + \alpha \cdot r \cdot \nabla_{\theta_i} \log \pi(a_t|s_t,\theta), \tag{17}$$

where r is the reward and α is the learning rate. Given $r = 3$ and $\alpha = 0.1$, we write

$$\theta_i' = \theta_i + 0.3 \cdot \nabla_{\theta_i} \log \pi(a_t|s_t,\theta). \tag{18}$$

(a) When $\mu(\theta_\mu) = \theta_\mu = 0$ and $\sigma(\theta_\sigma) = \exp(\theta_\sigma) = 4$, we get that $\theta_\mu = 0$ and $\theta_\sigma = \log(4)$. We can plug these values into equations (11) and (12), along with the given $a = 3$ and get:

$$\nabla_{\theta_\mu} \log \pi(a|s,\theta) = \frac{3-0}{\exp^2(\log(4))} = \frac{3}{16} = 0.1875,$$

$$\nabla_{\theta_\sigma} \log \pi(a|s,\theta) = \frac{(3-0)^2 - \exp^2(\log(4))}{\exp^2(\log(4)))} = \frac{9-16}{16} = -0.4375.$$

We can finish plugging in values for the update and get:

$$\theta'_\mu = 0 + 0.3 \cdot 0.1875 = 0.05625, \tag{19}$$
$$\theta'_\sigma = \log(4) - 0.3 \cdot 0.4375 = 1.2550443611. \tag{20}$$

The new policy $\mathcal{N}(\sigma(\theta'_\mu), \sigma(\theta'_\sigma))$ is

$$\pi(a|s,\theta) = \frac{1}{1.2550443611 \cdot \sqrt{2\pi}} \exp\left[-\frac{(3-0.05625)^2}{2 \cdot 1.2550443611^2}\right] \tag{21}$$

(b) When $\mu(\theta_\mu) = \theta_\mu = 0$ and $\sigma(\theta_\sigma) = \theta_\sigma^2 = 4$, we get that $\theta_\mu = 0$ and $\theta_\sigma = \pm\, 2$. We can plug these values into equations (15) and (16), along with the given $a = 3$ and get:

$$\nabla_{\theta_\mu} \log \pi(a|s,\theta) = \frac{3-0}{(\pm\, 2)^4} = \frac{3}{16} = 0.1875,$$
$$\nabla_{\theta_\sigma} \log \pi(a|s,\theta) = 2 \cdot \frac{(3-0)^2 - (\pm\, 2)^4}{(\pm\, 2)^5} = \pm 2 \cdot \frac{9-16}{32} = \mp 0.4375$$

We can finish plugging in values for the update and get:

$$\theta'_\mu = 0 + 0.3 \cdot 0.1875 = 0.05625, \tag{22}$$
$$\theta'_\sigma = \pm\, 2 \mp\, 0.3 \cdot 0.4375 = \pm\, 1.86875. \tag{23}$$

The new policy $\mathcal{N}(\sigma(\theta'_\mu), \sigma(\theta'_\sigma))$ is

$$\pi(a|s,\theta) = \pm\, \frac{1}{1.86875 \cdot \sqrt{2\pi}} \exp\left[-\frac{(3-0.05625)^2}{2 \cdot 1.86875^2}\right] \tag{24}$$

3. A drawback of a simple policy gradient like the one we have applied is that it acts in the parameter space, considering the gradient change in parameters. This is not necessarily what we care about, as we are more-so interested in directly updating the policy. We see this in our work, where the updates are kept small and the same gradients are found for both parameters, leading to similar update results for our μ and σ. However, small changes in parameters do not guarantee small changes in policy, particularly under different parametrizations. For instance, if we were to take the new σ value obtained under the second parametrization and use it to define our policy under the first parametrization, our new policy would be drastically different from what we started (Our σ would change from 4 to ≈ 6.5 rather than from 4 to ≈ 3.5). Such large changes in policy are undesirable since they can lead to large changes in the agent's behavior, which if in the wrong direction, can cause situations where the agent is unable to recover from the state space it ends up in.

9.5 Homework: Coding Assignment - Policy Gradients

1. Two advantages of using policy based methods over value based methods are:

   (a) Policy based methods can more easily be applied to problems with large and/or continuous state spaces. Unlike value based methods, policy based methods do not need to compute the value of each state, and furthermore do not need to find a way to compute a maximum over all possible state values, which can be prohibitively expensive. Policy based methods circumvent this issue by directly adjusting the parameters of the policy function, directly estimating the action probability distribution for a given input state.

(b) Policy based methods provide a natural way of learning stochastic policies. In value based methods, stochasticity is typically manually governed by the ϵ hyper-parameter used in *epsilon*-greedy policies. In policy based methods, the optimal stochasticity with arbitrary action probabilities can be learned.

2. Coding answers have been submitted on codegra under the group "stalwart cocky sawly".

10.3 Homework: Update Directions

The update equation for Policy Gradient is given by

$$\theta_{t+1} = \theta_t + \alpha \nabla_\theta J(\theta), \tag{25}$$

where α is the learning rate and J is the expected return. The update equation for Natural Policy Gradient is given by

$$\theta_{t+1} = \theta_t + \alpha F^{-1} \nabla_{\theta_t} J(\theta_t), \tag{26}$$

where F is the Fisher Information Matrix. The update equation for Trust Region Policy Optimization (TRPO) is given by

$$\theta_{t+1} = \theta_t + \beta F^{-1} \nabla_{\theta_t} J(\theta_t), \tag{27}$$

Where β is the optimal step-size which can be solved for from the following

$$c \approx \beta^2 (\nabla J)^T F (\nabla J)/2, \tag{28}$$

where c is some user-set constant.

1. Alice is correct, in that she would always obtain the same ordering ($\theta_{\text{scissors}} > \theta\text{rock} > \theta_{\text{paper}}$) if she had used TRPO instead of NPG. This is because both NPG and TRPO use essentially the same update equation, as can be noted when comparing equations (26) and (27), where the only difference is α vs β, which are scalars and hence only affect the magnitude (not direction) of change. As such, *after a single update*, none of the factors listed by Bob would make a difference.

2. Alice is once again correct, in that she would again always obtain the same ordering ($\theta_{\text{scissors}} > \theta\text{rock} > \theta_{\text{paper}}$) if she had used PG instead of NPG, this time for a slightly different reason. As we can see, the difference between equations (25) and (26) is that the latter introduces the additional inverted Fisher information matrix to the update. However, in our case F is initialized under a uniform policy, meaning that we initialize with $\theta_{\text{scissors}} = \theta_{\text{rock}} = \theta_{\text{paper}}$. Combined with the fact that the matrix is diagonal, the Fisher information matrix is essentially a scalar, and as such, there is no difference in direction between the two updates, only a difference in step size. Because the update direction is the same, the ordering found would be the same.

3. NPG and TRPO update parameters in the same direction in the parameter space, which can be different from the direction PG updates parameters in. In general PG updates parameters in the direction of increasing return. NPG and TRPO instead modify their constraints to quantify how fast the *policy* changes due to our update. The result is that NPG and TRPO update parameters in the direction defined as the direction of increasing return *constrained* by constant change in behaviour policy. This constraint can cause the direction to differ from the direction taken by PG. Between one and the other, NPG and TRPO update in the same direction, because as discussed in question 1, the main difference between the two is the step size, which does not affect the direction of the update.