Exercise Set 4 - Reinforcement Learning
Chapter 7,8 - Policy-based methods

Instructions

This is the fourth exercise booklet for Reinforcement Learning. It covers both ungraded exercises to practice at home or during the tutorial sessions as well as graded homework exercises and graded coding assignments. The graded assignments are clearly marked.

- Make sure you deliver answers in a clear and structured format. LaTeXhas our preference. Messy handwritten answers will not be graded.
- Pre-pend the name of your TA to the file name you hand in and remember to put your name and student ID on the submission;
- The deadline for this first assignment is **September 30th 2020 at 17:00** and will cover the material of chapter 7-8. All questions marked 'Homework' in this booklet need to be handed in on Canvas. The coding assignments need to be handed in separately through the codegra.de platform integrated on canvas.

Contents

7 Basic policy search: REINFORCE; approximations **2**
- 7.1 Homework: Coding Assignment - Policy Gradients 2
- 7.2 Baseline and gradient variance 2
- 7.3 Limits of policy gradients 3
- 7.4 *Exam Question: Policy Gradient Methods 3

8 Policy gradient methods: PGT & NPG **5**
- 8.1 Homework: Natural policy gradient 5
- 8.2 *Exam Question: Policy gradient methods 6

Chapter 7: Basic policy search: REINFORCE; approximations

7.1 Homework: Coding Assignment - Policy Gradients

1. We have spent a lot of time working on value based methods. We will now switch to policy based methods, i.e. learn a policy directly rather than learn a value function from which the policy follows. Mention two advantages of using a policy based method.

2. Download the notebook *RLLab5_PG.zip* from canvas assignments and follow the instructions.

7.2 Baseline and gradient variance

In the lecture, we have seen that introducing a constant baseline b for the trajectory reward $G(\tau)$ does not introduce a bias to our policy gradient.

$$\nabla J = \mathbb{E}_\tau \Big[\Big(G(\tau) - b \Big) \nabla \log p(\tau) \Big] \tag{1}$$

We now want to consider the variance when introducing a baseline.

1. Derive the optimal constant baseline that minimizes the variance of the policy gradient. Interpret your result.
   Hint: First use the definition of variance to write out the variance of the gradient estimate. What should the derivative of this function w.r.t. b look like at optimality? Keep in mind the likelihood-ratio trick (Deisenroth et al, p.28).

2. Consider the simple example in a bandit setting (i.e. no states):

$$r = a + 2 \tag{2}$$
$$a \sim \mathcal{N}(\theta, 1) \tag{3}$$
$$\nabla_\theta \log \pi(a) = a - \theta \tag{4}$$

   Can you argue what should be the optimal constant baseline in this case?
   Hint: Use your result from 1.

3. Now consider a baseline that is not constant, but dependent on the state $b(s_t)$. We want to establish that in this case, the policy gradient remains unbiased. Show that

$$\mathbb{E}_\tau \left[\sum_{t=1}^{T} \nabla \log \pi(a_t|s_t) b(s_t) \right] = 0. \tag{5}$$

   Hint: You can use the linearity of expectation or the law of iterated expectation to "decouple" the full trajectory τ in two parts, $s_1, a_1, \ldots, s_t$ and $a_t, r_{t+1}, s_{t+1}, \ldots$.

7.3 Limits of policy gradients

To explore the behavior of PGT gradients with different policy parametrizations we will use a stateless continuous bandit environment. In this environment, the agent performs a single action and receives a single reward before the episode is terminated. Furthermore, we assume that the reward function is known and the policy is represented by a normal distribution $\mathcal{N}(\mu(\theta_\mu), \sigma(\theta_\sigma))$.

It is common to parametrize standard deviation as an exponential or logarithm of the parameter but in this exercise, we will consider a different parametrization (that serves better for illustration purposes) with hyperparameter k (we treat k as a design choice and thus its value cannot be changed during optimization). We will also ignore the fact that with this parametrization, the standard deviation could technically become negative:

$$r = a - a^2 \tag{6}$$

$$\pi(a|\theta) = \frac{1}{\sigma(\theta_\sigma)\sqrt{2\pi}} \exp\left(- \frac{(a - \mu(\theta_\mu))^2}{2\sigma(\theta_\sigma)^2} \right) \tag{7}$$

$$\mu(\theta_\mu) = \theta_\mu \tag{8}$$

$$\sigma(\theta_\sigma) = k\theta_\sigma \tag{9}$$

1. Compute gradients of expected reward $\mathbb{E}_a[r]$ with respect to parameters θ_σ and θ_μ
2. Consider two different parameterizations that represent the same policy $\mathcal{N}(0, 0.1)$. :
   (a) $\theta_\mu = 0$, $\theta_\sigma = 1$, $k = 0.1$
   (b) $\theta_\mu = 0$, $\theta_\sigma = 0.01$, $k = 10$

   Calculate gradients ∇_{θ_μ} and ∇_{θ_σ} for both of them and perform a single gradient update with learning rate $\alpha = 0.001$. What can you observe when you compare new policies?
3. Plot the gradient directions with different $k = \{0.1, 1, 10\}$ using the notebook we provided. How does choice of k affect the gradient? Do you think that this choice can affect convergence speed? Why (not)?
4. If you consider the results you obtained in this exercise, what is the drawback of simple policy gradient?

7.4 *Exam Question: Policy Gradient Methods

You decide to build a simple e-mail answering app, that dependent on an email will decide to choose an action out of of 6 standard answers. Each answer will receive a rating between 0 and 5 stars.The action is chosen using a feedforward neural network that takes as input and ends with a softmax layer with 6 nodes, one for each answer. You want to train the network using reinforcement learning. Choosing an action and receiving the reward directly ends the episode. The next email will be chosen independently from the initial state distribution.

1. Having chosen an answer to an email , resulting in a rating , you want to update the network weights using the REINFORCE algorithm. Give the REINFORCE update for this setting, and explain how you would implement this update. You can assume the use of an automatic differentiation framework that can give you the gradient of the network outputs with respect to the network weights.

2. Some e-mails can easily be identified as coming from dissatisfied customers, and these customers will give you a low rating no matter what. You decide to use a state-dependent baseline. What would be an advantage of this?

3. How would you train an additional neural network such that its output can be used as a state-dependent baseline?

Chapter 8: Policy gradient methods: PGT & NPG

8.1 Homework: Natural policy gradient

In this section we revisit the problem setting from 7.3:

$$r = a - a^2 \tag{10}$$

$$\pi(a|\theta) = \frac{1}{\sigma(\theta_\sigma)\sqrt{2\pi}} \exp\left(- \frac{(a - \mu(\theta_\mu))^2}{2\sigma(\theta_\sigma)^2} \right) \tag{11}$$

$$\mu(\theta_\mu) = \theta_\mu \tag{12}$$

$$\sigma(\theta_\sigma) = k\theta_\sigma \tag{13}$$

1. Calculate $\nabla \log \pi(a|\theta)$ w.r.t. θ_μ and θ_σ.

2. By applying the natural policy gradient we want to the solve the contraint optimization problem

$$\theta^* - \theta_0 = \max_{d\theta} J(\theta_0 + d\theta) \;\text{ s.t. }\; d\theta^T F_\theta d\theta = c \tag{14}$$

   where we limit our gradient change to c in KL-divergence between the old and updated policy. The fisher information matrix F and the natural policy gradient update step is then given by

$$F_\theta = \mathbb{E}_\tau \Big[\nabla_{d\theta} \log \pi(a|\theta_0 + d\theta) \nabla_{d\theta} \log \pi(a|\theta_0 + d\theta)^T \Big] \tag{15}$$

$$\theta^* - \theta_0 \propto F^{-1} \nabla_\theta J(\theta_0). \tag{16}$$

   Calculate the Fisher information matrix F_θ for our Gaussian policy.

3. Consider two different parameterizations that represent the same policy $\mathcal{N}(0, 0.1)$:

   (a) $\theta_\mu = 0$, $\theta_\sigma = 1$, $k = 0.1$

   (b) $\theta_\mu = 0$, $\theta_\sigma = 0.01$, $k = 10$

   Perform a single gradient update with learning rate $\alpha = 1$ for both of them using natural policy gradient. What can you observe when you look at mean and variance of new policies? How does this compare to vanilla policy gradient?

4. Explain how the Fisher information matrix influences the gradient of θ, considering different values of hypereparamter $k = \{0.1, 10\}$.

5. Plot the gradient directions with different $k = \{0.1, 1, 10\}$ for both vanilla policy gradient and natural policy gradient using the notebook we provided (include plots in your answer sheet). Is there a difference between vanilla policy gradient and natural policy gradients plots? Explain why there is (not) a difference.

6. Why would we want use the natural policy gradient as opposed to the "vanilla" policy gradient?

8.2 *Exam Question: Policy gradient methods

1. Which of the following are unbiased estimators of the gradient $\nabla_\theta \mathbb{E}_\tau [G(\tau)]$ of undiscounted returns $G(\tau)$? Check all that apply.

   (a)
   $$\mathbb{E}_\tau \left[\sum_{t'=1}^{T} r_{t'} \sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \right]$$

   (b)
   $$\mathbb{E}_\tau \left[\sum_{t'=1}^{T} r_{t'} \sum_{t=t'}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \right]$$

   (c)
   $$\mathbb{E}_\tau \left[\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \left(G_t(\tau) - \hat{V}_w(s_t) \right) \right]$$

   (d)
   $$\mathbb{E}_\tau \left[\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) q_\pi(a_t, s_t) \right]$$

   (e)
   $$\mathbb{E}_\tau \left[\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \left(G_t(\tau) - \hat{q}_w(s_t, a_t) \right) \right]$$

   (f)
   $$\mathbb{E}_\tau \left[\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \hat{q}_w(s_t, a_t) \right]$$

2. Both vanilla policy gradient and natural policy gradient try to perform a gradient ascent while using a constraint to limit the size of the update. However, this constraint is not the same in both methods. In other words, each of them limits the size of the update in a different way. Explain what each of these two constraints enforces and how the updates differ between the resulting algorithms.

3. We consider a stateless bandit scenario where you can assume that an episode solely consists of one action and one reward. Parameters for the mean θ_μ and standard deviation $\exp(\theta_\sigma)$ are learned (the exponential transformation guarantees the variance is positive).
   $$\pi(a|\theta) = \frac{1}{\exp(\theta_\sigma)\sqrt{2\pi}} \exp\left(-\frac{(a-\theta_\mu)^2}{2\exp(\theta_\sigma)^2} \right)$$
   The Fisher information matrix is in this case given by:
   $$\mathbb{E}_a \left[\nabla_\theta \log \pi(a|\theta) \nabla_\theta \log \pi(a|\theta)^T \right] = \begin{bmatrix} \frac{1}{\exp(\theta_\sigma)^2} & 0 \\ 0 & BLANK \end{bmatrix}$$
   Calculate the BLANK entry $F_{\theta_\sigma \theta_\sigma}$ of F.
   Hint: You can use results for central moments of a normal distribution:
   $$\mathbb{E}_a\Big[(a-\theta_\mu) \Big] = 0, \quad \mathbb{E}_a\Big[(a-\theta_\mu)^2 \Big] = \exp(\theta_\sigma)^2$$
   $$\mathbb{E}_a\Big[(a-\theta_\mu)^3 \Big] = 0, \quad \mathbb{E}_a\Big[(a-\theta_\mu)^4 \Big] = 3\exp(\theta_\sigma)^4$$