CHAPTER 4

Risk-Sensitive Actor-Critic Algorithm

Since we found a suitable risk-adjusted value function in Chapter 3 that satisfies a Bellman equation, we now aim to connect sections 2.2 and 3.4. The goal is to incorporate the Bellman equation (3.7) directly into a reinforcement learning algorithm and exploit the fact that risk-sensitive strategies can then be learned. Since we want to experiment in environments with continuous action and state spaces later, the authors in [14] propose a slightly modified version of the (soft) actor-critic algorithm.

Algorithm 1 Multiple risk aversion actor-critic

Initialize actor network π^θ and critic network V^ϕ with weights θ, ϕ
for Episode 1,..., M **do**
  Sample a batch of N initial states s_0 and risk aversion levels λ
  for $t = 0, ..., T$ **do**
    Select actions from current policy $a_t = \pi(s_t)$
    Execute actions, receive rewards r_t and transition into new states s_{t+1}
    Compute the target
    $y_t = r_t + \overline{V}(s_{t+1})$
    Update the critic by gradient descent with respect to the loss
      $\mathscr{L}(\phi) = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{1}{\lambda}\exp(-\lambda(y_t - V(s_t))) - V(s_t)\right]$
    Update the actor by gradient descent with respect to the loss
      $\mathscr{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\lambda}\exp(-\lambda(r_t + V(s_{t+1})))$
    Update target critic network $\overline{\phi} = \tau\phi + (1-\tau)\overline{\phi}$
  end for
end for

The goal of this chapter is to investigate whether this proposal is truly compatible with our risk-adjusted Bellman equation. Therefore, we focus on the loss functions

$$\mathscr{L}_{\text{actor}}(\theta) = \mathbb{E}_{s_t,a_t,s_{t+1}}\left[\frac{1}{\lambda}\exp\big(-\lambda\,(r_t + V_\phi(s_{t+1}))\big)\right], \tag{4.1}$$

$$\mathscr{L}_{\text{critic}}(\phi) = \mathbb{E}_{s_t,a_t,s_{t+1}}\left[\frac{1}{\lambda}\exp\big(-\lambda\,(y_t - V_\phi(s_t))\big) - V_\phi(s_t)\right], \tag{4.2}$$

where

$$y_t := r_t + V_{\text{target}}(s_{t+1})$$

is the bootstrapped target (with a slowly updated target network V_{target}), and to show that these losses correspond to the risk-averse Bellman equation based on exponential utility.

We recall the risk-averse value function V_λ^π, which satisfies the Bellman equation (3.7) (and which our policy should maximize), and define the exponentiated value function $W_\lambda^\pi(s)$.[1]

$$V_\lambda^\pi(s_t) = -\frac{1}{\lambda}\log\mathbb{E}\Big[\exp\big(-\lambda\big(r_t + V_\lambda^\pi(s_{t+1})\big)\big)\Big|\, s_t\Big]$$
$$W_\lambda^\pi(s) := \exp\big(-\lambda V_\lambda^\pi(s)\big) = \mathbb{E}\Big[\exp\big(-\lambda(r_t + V_\lambda^\pi(s_{t+1}))\big)\Big|s_t\Big]. \tag{4.3}$$

4.1 Actor Loss

With a distribution μ over initial states $s_0 \in S$ in our environment, the natural risk-averse objective is to simply maximize

$$J(\pi) := \mathbb{E}_{s_0\sim\mu}\big[V_\lambda^\pi(s_0)\big]. \tag{4.4}$$

Equivalently, the optimal value function at $t=0$ is $V_\lambda^*(s_0) = \sup_\pi V_\lambda^\pi(s_0)$ and we want $\pi^* \in \arg\max_\pi J(\pi)$. For fixed $\lambda > 0$, the map $f(x) := \frac{1}{\lambda}e^{-\lambda x}$ is strictly decreasing in x. Hence, maximizing (4.4) is, in terms of policies, equivalent to minimizing

$$\tilde{J}(\pi) := \mathbb{E}_{s_0\sim\mu}\left[\frac{1}{\lambda}\exp\big(-\lambda V_\lambda^\pi(s_0)\big)\right] = \mathbb{E}_{s_0\sim\mu}\left[\frac{1}{\lambda}W_\lambda^\pi(s_0)\right]. \tag{4.5}$$

such that we will continue with our equivalent control problem (4.5) which is related to the exponentiated Bellman equation (4.3). For any $t \in \mathbb{N}$, taking expectation over s_t under the state distribution induced by the current policy π (the on-policy distribution or $P(\cdot \mid s_t, a_t^\pi)$ from chapter 2.1), gives

$$\mathbb{E}_{s_t}\left[\frac{1}{\lambda}W_\lambda^\pi(s_t)\right] = \mathbb{E}_{s_t}\left[\frac{1}{\lambda}\mathbb{E}_{s_{t+1}|a_t}\Big[\exp\big(-\lambda(r_t + V_\lambda^\pi(s_{t+1}))\big)\Big|s_t\Big]\right] = \frac{1}{\lambda}\mathbb{E}_{s_t,a_t,s_{t+1}}\Big[\exp\big(-\lambda(r_t + V_\lambda^\pi(s_{t+1}))\big)\Big]. \tag{4.6}$$

Thus the exponentiated value function is equal (in expectation) to the exponentiated one-step reward plus next value, and this Bellman identity also shows that $J(\pi)$ is maximized if it pointwise minimizes $\mathbb{E}_{s_t}\left[\frac{1}{\lambda}W_\lambda^\pi(s_t)\right]$. We now define parameterizations of our networks:

- π_θ is a deterministic policy network with parameters θ.
- V_ϕ is a value network (critic) with parameters ϕ, approximating $V_\lambda^{\pi_\theta}$.

Now our on-policy transformations (s_t, a_t, s_{t+1}) are sampled using the current policy of our actor π_θ. If V_ϕ were exactly equal to $V_\lambda^{\pi_\theta}$, then, from (4.6),

$$\mathbb{E}_{s_t}\left[\frac{1}{\lambda}W_\lambda^\pi(s_t)\right] = \frac{1}{\lambda}\mathbb{E}_{s_t,a_t,s_{t+1}}\Big[\exp\big(-\lambda(r_t + V_\lambda^{\pi_\theta}(s_{t+1}))\big)\Big] = \frac{1}{\lambda}\mathbb{E}_{s_t,a_t,s_{t+1}}\Big[\exp\big(-\lambda(r_t + V_\phi(s_{t+1}))\big)\Big]. \tag{4.7}$$

This motivates defining the actor loss

$$\mathscr{L}_{\text{actor}}(\theta) := \mathbb{E}_{s_t,a_t^{\pi_\theta},s_{t+1}}\left[\frac{1}{\lambda}\exp\big(-\lambda(r_t + V_\phi(s_{t+1}))\big)\right],^{2} \tag{4.8}$$

which can be approximated using one-step transitions and the critic V_ϕ. From our above derivation follows that a policy π that minimizes $\mathscr{L}_{\text{actor}}$ is optimal for the risk-averse control problem (4.4) under the assumption that V_ϕ is a decent approximation of $V_\lambda^{\pi_\theta}$, which we will investigate in the next section.

[1] Since we are in the one-step Bellman environment, if we do not specify $\mathbb{E}$ we mean the one-step $\mathbb{E}_{s_{t+1}|a_t}$.
[2] In [14] they use single transitions, which is why their $\mathbb{E}_{s_t,a_t,s_{t+1}}$ reduces to $\mathbb{E}_{s_t}$.

4.2 Critic Loss Fixed Point

We want a loss function whose minimizer (for each fixed s_t) is exactly the Bellman value $V_\lambda^\pi(s_t)$. Fix a state $s_t \in S$ and consider y as a random variable with distribution given by the next-step dynamics under policy π:

$$y := r(s,a) + V_\lambda^\pi(s_{t+1}), \quad s_{t+1} \sim P(\cdot \mid s_t, a_t).$$

We seek a scalar function $f(v)$ such that $v^* := V_\lambda^\pi(s_t)$ is the unique minimizer of $f(v)$. The paper [14] proposes for each state s_t,

$$f_s(v) := \mathbb{E}_{s_{t+1}|a_t}\Big[\frac{1}{\lambda}\exp\big(-\lambda(y-v)\big) - v \,\Big|\, s_t\Big]. \tag{4.9}$$

For fixed y,

$$\frac{\partial}{\partial v}\left(\frac{1}{\lambda}\exp\big(-\lambda(y-v)\big) - v\right) = \exp\big(-\lambda(y-v)\big) - 1.$$

Thus[3] $f_s'(v) = \mathbb{E}_{s_{t+1}|a_t}\big[\exp\big(-\lambda(y-v)\big) - 1 \mid s_t\big]$. A stationary point v^* must satisfy $f_s'(v^*) = 0$, i.e.

$$\mathbb{E}_{s_{t+1}|a_t}\big[\exp\big(-\lambda(y-v^*)\big) \mid s_t\big] = 1.$$

We make the following observation.

Proposition 1. For any real v, the following are equivalent:

$$\mathbb{E}\big[\exp\big(-\lambda(y-v)\big) \mid s_t\big] = 1, \tag{4.10}$$

$$\exp(-\lambda v) = \mathbb{E}\big[\exp(-\lambda y) \mid s_t\big]. \tag{4.11}$$

In particular, $v^* = V_\lambda^\pi(s_t)$ satisfies both.

Proof. The equivalence is immediate.

$$1 = \mathbb{E}\big[\exp\big(-\lambda(y-v)\big) \mid s_t\big] = \exp(\lambda v)\,\mathbb{E}\big[\exp(-\lambda y) \mid s_t\big].$$

Moreover, note that $v^* = V_\lambda^\pi(s_t)$ satisfies (4.11) by construction (4.3). Hence it also satisfies (4.10). □

We have proven that $V_\lambda^\pi(s_t)$ minimizes $f_s(v)$. Arguing that it is strictly convex in v, it is also unique.

Parametrized critic loss

We now generalize from a scalar v to a value function $V_\phi : S \to \mathbb{R}$. Since our actor evaluates $V_\lambda^{\pi_\theta}(s_t)$ state-wise, or more precisely, its expectation. Then, the state-wise function f_s naturally leads to a loss

$$\begin{aligned}\mathscr{L}_{\text{critic}}(\phi) &:= \mathbb{E}_{s_t}\big[f_{s_t}(V_\phi(s_t))\big] = \mathbb{E}_{s_t}\left[\mathbb{E}_{a_t^{\pi_\theta}, s_{t+1}}\Big[\frac{1}{\lambda}\exp\big(-\lambda(y_t - V_\phi(s_t))\big) - V_\phi(s_t) \,\Big|\, s_t\Big]\right] \\ &= \mathbb{E}_{s_t, a_t^{\pi_\theta}, s_{t+1}}\left[\frac{1}{\lambda}\exp\big(-\lambda(y_t - V_\phi(s_t))\big) - V_\phi(s_t)\right].\end{aligned} \tag{4.12}$$

Thus, for each state $s_t \in S$, the critic value $V_\phi(s_t)$ being the Bellman value $V_\lambda^{\pi_\theta}(s_t)$ uniquely minimizes the critic loss used in the algorithm. Note that in practice y_t is defined using a separate target network V_{target}, i.e. $y_t := r_t + V_{\text{target}}(s_{t+1})$, and gradients are not backpropagated through y_t.

[3] under standard integrability assumptions.