# CS 295: Optimal Control and Reinforcement Learning

Winter 2020

# Assignment 4

**due Monday, March 23 2020, 11pm**

## Part I

1. Consider a model-based reinforcement learning algorithm that estimates a model $\hat{p}$ of the true dynamics $p$, and then uses it for planning. In all parts of this question, we assume that we can plan optimally in the estimated model, with the true non-negative reward function.

   (a) Suppose that the estimated model is guaranteed to have

   $$\|p(s'|s,a) - \hat{p}(s'|s,a)\|_1 \leqslant \epsilon,$$

   for all $s$ and $a$, and that the initial distribution $p(s_0)$ is known.
   Show that $|\mathbb{E}_{p_\pi}[r_t] - \mathbb{E}_{\hat{p}_\pi}[r_t]| \leqslant \epsilon t r_{\max}$, for any policy $\pi(a|s)$.
   Hint: show by induction that $\|p_\pi(s_t) - \hat{p}_\pi(s_t)\|_1 \leqslant \epsilon t$.
   Bonus: show the tighter bound $|\mathbb{E}_{p_\pi}[r_t] - \mathbb{E}_{\hat{p}_\pi}[r_t]| \leqslant \frac{1}{2}\epsilon t r_{\max}$.

   (b) Conclude that planning in $\hat{p}$ is near-optimal: $\mathbb{E}_{p_\pi}[R] - \mathbb{E}_{p_{\hat{\pi}}}[R] \leqslant 2\frac{\gamma}{(1-\gamma)^2}\epsilon r_{\max}$ (or without the 2, given the bonus question above), where $\pi$ is optimal for $p$ and $\hat{\pi}$ is optimal for $\hat{p}$. Note that $\sum_t \gamma^t t = \frac{\gamma}{(1-\gamma)^2}$.

   (c) Suppose that the state space is continuous, and that both the true dynamics $f$ and the model $\hat{f}$ are deterministic, with a known initial state $s_0$. Determinism implies that there exists an optimal open-loop policy, i.e. a sequence of actions.
   Suppose that the true dynamics, the model, and the reward function are all Lipschitz, i.e. there exists a constant $L$ such that $\|f(s,a) - f(\hat{s},a)\| \leqslant L\|s - \hat{s}\|$, for all $s$, $\hat{s}$, and $a$, and similarly for $\hat{f}$; and for $r$, i.e. $|r(s,a) - r(\hat{s},a)| \leqslant L\|s - \hat{s}\|$. Suppose $L > 1$. Suppose further that the estimated model is guaranteed to have

   $$\|f(s,a) - \hat{f}(s,a)\| \leqslant \epsilon,$$

   for all $s$ and $a$.
   Let $r_t$ and $\hat{r}_t$ be the rewards in step $t$ when the same sequence of actions is taken in $f$ and, respectively, in $\hat{f}$. Show that $|r_t - \hat{r}_t| \leqslant \frac{L^t-1}{L-1}L\epsilon$.

2. A finite-state controller (FSC) is a finite-state machine with state space $\mathcal{M}$; an internal state update distribution, upon observing $o_t$, from internal state $m_{t-1}$ to $m_t$ with probability $\pi(m_t|m_{t-1}, o_t)$; and an action emission distribution $\pi(a_t|m_t)$.

   Given a FSC and POMDP dynamics $p(s_{t+1}|s_t, a_t)$ and $p(o_t|s_t)$, write down a forward recursion for computing the joint distribution of $m_{t-1}$ and $s_t$; that is, show how to compute $p_\pi(m_t, s_{t+1})$ using $p$, $\pi$, and $p_\pi(m_{t-1}, s_t)$. Show how to recover from this joint distribution the predictive belief $p(s_t|m_{t-1})$.

   Given also a reward function $r(s_t, a_t)$, write down a backward recursion for evaluating $V_\pi(s_t, m_t)$; that is, show how to compute $V_\pi(s_t, m_t)$ using $p$, $\pi$, $r$, and $V_\pi(s_{t+1}, m_{t+1})$.

3. Recall that in the A2C algorithm we have an actor $\pi_\theta$ and a critic $V_\phi$. For on-policy experience $(s, a, r, s')$, with advantage $A_\phi = r + \gamma V_{\bar{\phi}}(s') - V_\phi(s)$, we have a value loss $\mathcal{L}_\phi = A_\phi^2$ and a policy gradient $\nabla_\theta \log \pi_\theta(a|s) A_\phi$.

   Also recall that, in the "control as inference" framework, we optimally have that the policy is $\pi(a|s) = \frac{\pi_0(a|s) \exp \beta Q(s,a)}{\exp \beta V(s)}$, and therefore

$$Q(s, a) = V(s) + \frac{1}{\beta} \log \frac{\pi(a|s)}{\pi_0(a|s)}. \tag{1}$$

   In the SQL algorithm, the loss is the square Bellman error $(r + \gamma V_{\bar{\theta}}(s') - Q_\theta(s, a))^2$.

   Consider implementing the SQL algorithm by parametrizing $Q_{\theta,\phi}$ as the function (1) of an actor $\pi_\theta$ and a critic $V_\phi$. Write down the SQL loss for on-policy experience $(s, a, r, s')$, in terms of this $Q_{\theta,\phi}$. Expand the expression of its gradient, to show that it is equivalent to the gradient in A2C.

   What is the equivalent of $\beta$ in A2C?