HW7: Policy Search

CS 4300: Artificial Intelligence **William Frank, u1067292**
University of Utah **3/20/2019**

1 Policy Gradient

In order to do policy gradient, we need to be able to compute the gradient of the value function J with respect to a parameter vector θ: $\nabla_\theta J(\theta)$. By our algebraic magic, we expressed this as:

$$\nabla_\theta J(\theta) = \sum_a \pi_\theta(s_0, a) R(a) \underbrace{\nabla_\theta \log\left(\pi_\theta(s_0, a)\right)}_{g(s_0,a)} \tag{1}$$

If we us a linear function thrown through a soft-max as our stochastic policy, we have:

$$\pi_\theta(s, a) = \frac{\exp\left(\sum_{i=1}^n \theta_i f_i(s, a)\right)}{\sum_{a'} \exp\left(\sum_{i=1}^n \theta_i f_i(s, a')\right)} \tag{2}$$

Compute a closed form solution for $g(s_0, a)$. Explain in a few sentences *why* this leads to a sensible update for gradience ascent (i.e., if we plug this in to Eq (1) and do gradient ascent, why is the derived form reasonable)?

$\nabla_\theta \log\left(\pi_\theta(s_0, a)\right)$

$\nabla_\theta \log\left(\frac{\exp\left(\sum_{i=1}^n \theta_i f_i(s,a)\right)}{\sum_{a'} \exp\left(\sum_{i=1}^n \theta_i f_i(s,a')\right)}\right)$

$\nabla_\theta \log\left(\frac{\exp\left(\theta_1 f_1(s,a') + \ldots \theta_n f_n(s,a')\right)}{\sum_{a'} \exp(\theta_1 f_1(s,a') + \ldots \theta_n f_n(s,a'))}\right)$

$\left(\frac{\sum_{a'} \exp\left(\theta_1 f_1(s,a') + \ldots \theta_n f_n(s,a')\right)}{\exp(\theta_1 f_1(s,a') + \ldots \theta_n f_n(s,a'))}\right) \nabla_\theta \left(\frac{\exp\left(\theta_i f_i(s,a') + \ldots \theta_n f_n(s,a')\right)}{\sum_{a'} \exp(\theta_1 f_1(s,a') + \ldots \theta_n f_n(s,a'))}\right)$

$\left(\frac{\sum_{a'} \exp\left(\sum_{i=1}^n \theta_i f_i(s,a)\right)}{\exp\left(\sum_{i=1}^n \theta_i f_i(s,a)\right)}\right) \begin{bmatrix} \frac{f_1(s,a) e^{\theta_1 f_1(s,a)}}{\sum_{a'} f_1(s,a) e^{\theta_1 f_1(s,a')}} \\ \vdots \\ \frac{f_n(s,a) e^{\theta_n f_n(s,a)}}{\sum_{a'} f_n(s,a) e^{\theta_n f_n(s,a')}} \end{bmatrix}$

$\frac{1}{\pi_\theta(s,a)} \begin{bmatrix} \frac{e^{\theta_1 f_1(s,a)}}{\sum_{a'} e^{\theta_1 f_1(s,a')}} \\ \vdots \\ \frac{e^{\theta_n f_n(s,a)}}{\sum_{a'} e^{\theta_n f_n(s,a')}} \end{bmatrix}$

This gives us $\nabla_\theta J(\theta) = \sum_a R(a) \begin{bmatrix} \frac{e^{\theta_1 f_1(s,a)}}{\sum_{a'} e^{\theta_1 f_1(s,a')}} \\ \vdots \\ \frac{e^{\theta_n f_n(s,a)}}{\sum_{a'} e^{\theta_n f_n(s,a')}} \end{bmatrix}$. This gives essentially the expected reward for all actions, weighted by the importance of a feature in regards to every other weight, in respect to each weight.

This is a sensible update because the resultant vector added to the original will move J closer to more positive rewards. Additionally, attributes with a relatively larger weight will grow more quickly, moving the most important weights closer to their convergence.