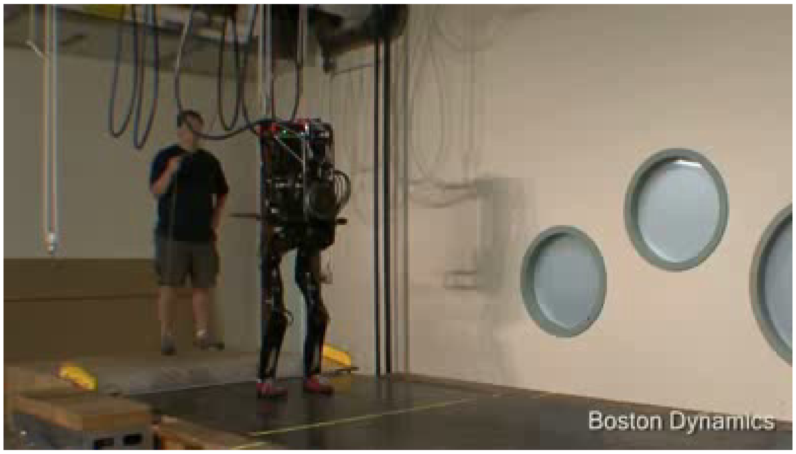
Boston Dynamics PetMan

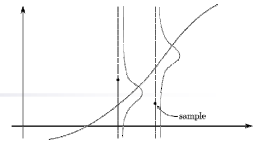


CS 287: Advanced Robotics
Fall 2009

Lecture 18: Policy search

Pieter Abbeel
UC Berkeley EECS

Policy gradient



	Deterministic		Stochastic	
	Known Dynamics	Unknown Dynamics	Known Dynamics	Unknown Dynamics
Analytical	OK Taking derivatives---potentially time consuming and error-prone	N/A	OK Often computationally impractical	N/A
Finite differences	OK Sometimes computationally more expensive than analytical	OK	OK N = #roll-outs: Naive: $O(N^{-1/4})$, or $O(N^{-2/5})$ Fix random seed: $O(N^{-1/2})$ [1]	Same as known dynamics, but no fixing of random seed.
Likelihood ratio method	OK	OK	OK $O(N^{-1/2})$ [1]	OK $O(N^{-1/2})$ [1]

[1] P. Glynn, "Likelihood ratio gradient estimation: an overview," in *Proceedings of the 1987 Winter Simulation Conference, Atlanta, GA, 1987,* pp. 366–375.

Likelihood ratio method

- Assumption:
  - Stochastic policy $\pi_\theta(u_t \mid s_t)$
  - Stochasticity:
    - Required for the methodology
    - \+ Helpful to ensure exploration
    - \- Optimal policy within the policy class is not always stochastic (though it can be!!)

Likelihood ratio method

We let τ denote a state-action sequence $s_0, u_0, \ldots, s_H, u_H$. We overload notation: $R(\tau) = \sum_{t=0}^{H} R(s_t, u_t)$.

$$U(\theta) = \mathrm{E}[\sum_{t=0}^{H} R(s_t, u_t); \pi_\theta] = \sum_\tau P(\tau; \theta) R(\tau)$$

Our goal is to find θ:

$$\max_\theta U(\theta) = \max_\theta \sum_\tau P(\tau; \theta) R(\tau)$$

Likelihood ratio method derivation

$$U(\theta) = \sum_\tau P(\tau; \theta) R(\tau)$$

Taking the gradient w.r.t. θ gives

$$\begin{aligned}
\nabla_\theta U(\theta) &= \nabla_\theta \sum_\tau P(\tau; \theta) R(\tau) \\
&= \sum_\tau \nabla_\theta P(\tau; \theta) R(\tau) \\
&= \sum_\tau \frac{P(\tau; \theta)}{P(\tau; \theta)} \nabla_\theta P(\tau; \theta) R(\tau) \\
&= \sum_\tau P(\tau; \theta) \frac{\nabla_\theta P(\tau; \theta)}{P(\tau; \theta)} R(\tau) \\
&= \sum_\tau P(\tau; \theta) \nabla_\theta \log P(\tau; \theta) R(\tau)
\end{aligned}$$

Approximate with the empirical estimate for m sample paths under policy π_θ:

$$\nabla_\theta U(\theta) \approx \hat{g} = \frac{1}{m} \sum_{i=1}^{m} \nabla_\theta \log P(\tau^{(i)}; \theta) R(\tau^{(i)})$$

$$\begin{aligned}
\nabla_\theta \log P(\tau^{(i)}; \theta) &= \nabla_\theta \log \left[\prod_{t=0}^{H} \underbrace{P(s_{t+1}^{(i)} | s_t^{(i)}, u_t^{(i)})}_{\text{dynamics model}} \cdot \underbrace{\pi_\theta(u_t^{(i)} | s_t^{(i)})}_{\text{policy}} \right] \\
&= \nabla_\theta \left[\sum_{t=0}^{H} \log P(s_{t+1}^{(i)} | s_t^{(i)}, u_t^{(i)}) + \sum_{t=0}^{H} \log \pi_\theta(u_t^{(i)} | s_t^{(i)}) \right] \\
&= \nabla_\theta \sum_{t=0}^{H} \log \pi_\theta(u_t^{(i)} | s_t^{(i)}) \\
&= \sum_{t=0}^{H} \underbrace{\nabla_\theta \log \pi_\theta(u_t^{(i)} | s_t^{(i)})}_{\text{no dynamics model required!!}}
\end{aligned}$$

Likelihood ratio method: result recap

The following expression provides us with an unbiased estimate of the gradient, and we can compute it without access to a dynamics model:

$$\hat{g} = \frac{1}{m}\sum_{i=1}^{m} \nabla_\theta \log P(\tau^{(i)};\theta) R(\tau^{(i)})$$

Here:

$$\nabla_\theta \log P(\tau^{(i)};\theta) = \sum_{t=0}^{H} \underbrace{\nabla_\theta \log \pi_\theta(u_t^{(i)}|s_t^{(i)})}_{\text{no dynamics model required!!}}$$

Unbiased means:

$$\mathrm{E}[\hat{g}] = \nabla_\theta U(\theta)$$

Likelihood ratio method in practice

- As formulated thus far: yes, unbiased estimator, but very noisy, hence would take very long
- Set of critical fixes that have led to real-world practicality:
  - Add a free parameter to the estimator called “baseline” and set it such that the variance of the estimator is minimized
  - Exploit temporal structure + incorporate value function estimates (= actor-critic learning)
  - Don’t step in the direction of the gradient, follow the “natural” gradient direction instead

Consider the following scenario:

There are two envelopes, each of which has an unknown amount of money in it. You get to choose one of the envelopes. Given this is all you get to know, how should you choose?

Consider the changed scenario:

Same as above, but before you get to choose, you can ask me to disclose the amount in one of the envelopes. Without any distributional assumptions on the amounts of money, is there a strategy that could improve your expected pay-off over simply picking an envelope at random?

Envelopes riddle

Envelopes riddle

Envelopes riddle

- MDP:
  - horizon of 1, always start in state 0
  - Transition to state 1 or 2 according to choice made
  - Observe the reward in the visited state
- Policy

$$\pi_\theta(1|0) = \frac{\exp(\theta)}{1+\exp(\theta)}$$
$$\pi_\theta(2|0) = \frac{1}{1+\exp(\theta)}$$

- Choose to see an envelope's contents according to π_θ
- Perform a gradient update:

$$\nabla_\theta \log P(\tau = 1; \theta) R(\tau = 1) = \frac{1}{1+\exp(\theta)} R(1)$$

$$\nabla_\theta \log P(\tau = 2; \theta) R(\tau = 2) = -\frac{\exp(\theta)}{1+\exp(\theta)} R(2)$$

Envelopes riddle

$$\pi_\theta(1|0) = \frac{\exp(\theta)}{1+\exp(\theta)}$$
$$\pi_\theta(2|0) = \frac{1}{1+\exp(\theta)}$$

- Perform a gradient update:

$$\nabla_\theta \log P(\tau = 1; \theta) R(\tau = 1) = \frac{1}{1+\exp(\theta)} R(1)$$

$$\nabla_\theta \log P(\tau = 2; \theta) R(\tau = 2) = -\frac{\exp(\theta)}{1+\exp(\theta)} R(2)$$

- This gradient update is simply making the recently observed path more likely; and how much more likely depends on the observed R for the observed path
- → rather than let it depend simply on R, if we had a "baseline" b which is an estimate of the expected reward under the current policy, then could update scaled by (R-b) instead,

  i.e. the baseline enables updating such that better than average paths become more likely, less than average paths become less likely

Likelihood ratio method with baseline

- Gradient estimate with baseline:

$$\hat{g} = \frac{1}{m} \sum_{i=1}^{m} \nabla_\theta \log P(\tau^{(i)}; \theta)(R(\tau^{(i)}) - b)$$

- This will (crudely speaking) increase the log-likelihood of paths with higher than baseline reward, and decrease the log-likelihood of observed paths with lower than baseline reward.
- Is this still an unbiased gradient estimate?

  Unbiased means: $\mathrm{E}[\hat{g}] = \nabla_\theta U(\theta)$

Even with baseline, we obtain an unbiased estimate of the gradient

$$
\begin{aligned}
&\sum_\tau P(\tau;\theta) = 1 \\
\Rightarrow &\frac{\partial}{\partial\theta_i}\sum_\tau P(\tau;\theta) = 0 \\
\Leftrightarrow &\sum_\tau \frac{\partial}{\partial\theta_j} P(\tau;\theta) = 0 \\
\Leftrightarrow &\sum_\tau \frac{P(\tau;\theta)}{P(\tau;\theta)} \frac{\partial}{\partial\theta_j} P(\tau;\theta) = 0 \\
\Rightarrow &\sum_\tau P(\tau;\theta) \frac{\partial}{\partial\theta_j} \log P(\tau;\theta) = 0 \\
\Rightarrow &\mathrm{E}_\tau\left[\frac{\partial}{\partial\theta_j} \log P(\tau;\theta)\right] = 0 \\
\Rightarrow &\mathrm{E}_\tau\left[\frac{\partial}{\partial\theta_j} \log P(\tau;\theta) b_j\right] = 0
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial\theta_j} U(\theta) &= \mathrm{E}_\tau\left[\frac{\partial}{\partial\theta_j} \log P(\tau;\theta) R(\tau)\right] \\
&= \mathrm{E}_\tau\left[\frac{\partial}{\partial\theta_j} \log P(\tau;\theta) (R(\tau) - b_j)\right] \\
&\approx \frac{1}{m}\sum_{i=1}^m \left[\frac{\partial}{\partial\theta_j} \log P(\tau^{(i)};\theta) (R(\tau) - b_j^{(i)})\right]
\end{aligned}
$$

Natural choices for b:

- Estimate of utility U(θ)
- Choose b_j to minimize the variance of the gradient estimates.

Our gradient estimate:

$$\hat{g}_j = \frac{\partial}{\partial\theta_j} \log P(\tau^{(i)};\theta) \cdot (R(\tau) - b_j),$$

It is unbiased, i.e.:

$$\mathrm{E}\hat{g}_j = \frac{\partial U(\theta)}{\partial\theta_j}$$

Its variance is given by:

$$\mathrm{E}\left[(\hat{g}_j - \mathrm{E}[\hat{g}_j])^2\right]$$

which we would like to minimize over b_j:

$$
\begin{aligned}
\min_{b_j} \mathrm{E}\left[(\hat{g}_j - \mathrm{E}[\hat{g}_j])^2\right] &= \mathrm{E}\hat{g}_j^2 + \mathrm{E}\left[(\mathrm{E}\hat{g}_j)^2\right] - 2\mathrm{E}\left[\hat{g}_j - \mathrm{E}[\hat{g}_j]\right] \\
&= \mathrm{E}\hat{g}_j^2 + (\mathrm{E}\hat{g}_j)^2 - 2\mathrm{E}[\hat{g}_j]\,\mathrm{E}[\hat{g}_j] \\
&= \mathrm{E}\hat{g}_j^2 - \underbrace{(\mathrm{E}\hat{g}_j)^2}_{=\frac{\partial U(\theta)}{\partial\theta_j} - \text{independent of } b_j}
\end{aligned}
$$

$$\min_{b_j} \mathrm{E}\hat{g}_j^2 = \min_{b_j} \mathrm{E}_\tau\left[\left(\frac{\partial}{\partial\theta_j}\log P(\tau;\theta)\cdot(R(\tau)-b_j)\right)^2\right]$$

$$= \min_{b_j} \mathrm{E}_\tau\left[\left(\frac{\partial}{\partial\theta_j} log(\tau;\theta)\right)^2\cdot\left(R(\tau)^2+b_j^2-2b_jR(\tau)\right)\right]$$

$$= \underbrace{\min_{b_j} \mathrm{E}_\tau\left[\left(\frac{\partial}{\partial\theta_j}\log P(\tau;\theta)\right)^2\cdot R(\tau)^2\right]}_{\text{independent of } b_j} + \mathrm{E}\left[\left(\frac{\partial}{\partial\theta_j}\log P(\tau;\theta)\right)^2\cdot b_j^2\right]$$

$$-2\mathrm{E}\left[\left(\frac{\partial}{\partial\theta_i}\log P(\tau;\theta)^2\right)\cdot b_jR(\tau)\right]$$

$$= \min_{b_j} b_j^2\cdot\mathrm{E}_\tau\left[\left(\frac{\partial}{\partial\theta_j}\log P(\tau;\theta)\right)^2\right] - 2b_j\mathrm{E}_\tau\left[\left(\frac{\partial}{\partial\theta_j}\log P(\tau;\theta)^2\right)^2 R(\tau)\right]$$

$$\frac{\partial}{\partial b_j} = 0 \Rightarrow 2b_j\mathrm{E}_\tau\left[\ldots\right] - 2\mathrm{E}_\tau\left[\ldots\right] = 0$$

$$\boxed{\Rightarrow b_j = \frac{\mathrm{E}_\tau\left[\left(\frac{\partial}{\partial\theta_j}\log P(\tau;\theta)\right)^2\cdot R(\tau)\right]}{\mathrm{E}_\tau\left[\left(\frac{\partial}{\partial\theta_j}\log P(\tau;\theta)\right)^2\right]}}$$

→ Could estimate optimal baseline from samples.

Exploiting temporal structure

Our gradient estimate:

$$\hat{g}_j = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{\partial}{\partial\theta_j}\log P(\tau^{(i)};\theta)\right)\left(R(\tau^{(i)})-b_j\right),$$

$$= \frac{1}{m}\sum_{i=1}^{m}\left(\sum_{t=0}^{H-1}\frac{\partial}{\partial\theta_j}\log\pi_\theta(u_t^{(i)}|s_t^{(i)})\right)\left(\sum_{t=0}^{H-1}R(s_t^{(i)},u_t^{(i)})-b_j\right)$$

Future actions do not depend on past rewards (assuming a fixed policy). This can be formalized as

$$\mathrm{E}\left[\frac{\partial}{\partial\theta_j}\log\pi_\theta(u_t|s_t)R(s_k,u_k)\right] = 0 \quad \forall k<t$$

Removing these terms with zero expected value from our gradient estimate we obtain:

$$\hat{g}_j = \frac{1}{m}\sum_{i=1}^{m}\sum_{t=0}^{H-1}\frac{\partial}{\partial\theta_j}\log\pi_\theta(u_t^{(i)}|s_t^{(i)})\left(\sum_{k=t}^{H-1}R(s_k^{(i)},u_k^{(i)})-b_j\right)$$

Actor-Critic

Our gradient estimate:

$$\hat{g}_j = \frac{1}{m}\sum_{i=1}^{m}\sum_{t=0}^{H-1}\frac{\partial}{\partial\theta_j}\log\pi_\theta(u_t^{(i)}|s_t^{(i)})\left(\sum_{k=t}^{H-1}R(s_k^{(i)},u_k^{(i)})-b_j\right)$$

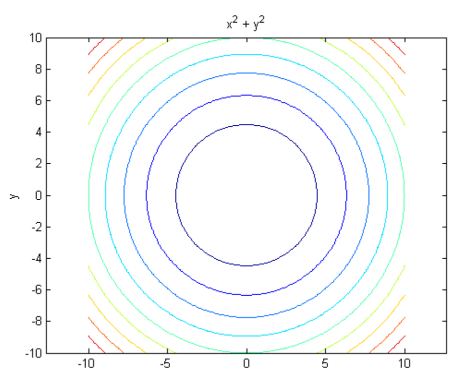
The term $\sum_{k=t}^{H-1}R(s_k^{(i)},u_k^{(i)})$ is a sample based estimate of $Q^{\pi_\theta}(s_t^{(i)},u_k^{(i)})$. If we simultaneously run a temporal difference (TD) learning method to estimate Q^{π_θ}, then we could substitute its estimate for Q instead of the sample based estimate!

Our gradient estimate becomes:

$$\hat{g}_j = \frac{1}{m}\sum_{i=1}^{m}\sum_{t=0}^{H-1}\frac{\partial}{\partial\theta_j}\log\pi_\theta(u_t^{(i)}|s_t^{(i)})\left(\hat{Q}^{\pi_\theta}(s_t^{(i)},u_t^{(i)})-b_j\right)$$

Natural gradient

- Is the gradient the correct direction?



Natural gradient

- Is the gradient the correct direction?