Advanced Topics in Reinforcement Learning

Lecture 6: Monte Carlo methods

Josiah Hanna
University of Wisconsin — Madison

Announcements

- Homework 1 due Thursday, September 29.
- Homework 2 released on Thursday.
- Start reading chapter 6 for next week.
- Project proposals due next week.

Course Overview

- So far we've seen:
  - Learning in a simplified setting (k-armed bandits).
  - Formalized reinforcement learning problems (MDPs).
  - Exact solution methods for MDPs (dynamic programming methods).
- Today: first learning methods for MDPs.
- Next week: learning methods that bootstrap like dynamic programming methods.

This Week

- General Monte Carlo
- On-policy Monte Carlo Prediction
- On-policy Monte Carlo Control
- Off-policy Monte Carlo Prediction
- Off-policy Monte Carlo Control

Statistics Review

- We have random variable $X \sim d$ and use X as an estimate of unknown value μ. The expected value of X is $E_d[X]$.

- **Variance** of X:

$$\mathtt{Var}_d[X] = \mathbf{E}_d[(X - \mathbf{E}_d[X])^2]$$

- **Bias** of X:

$$\mathtt{Bias}_d[X] = \mu - \mathbf{E}_d[X]$$

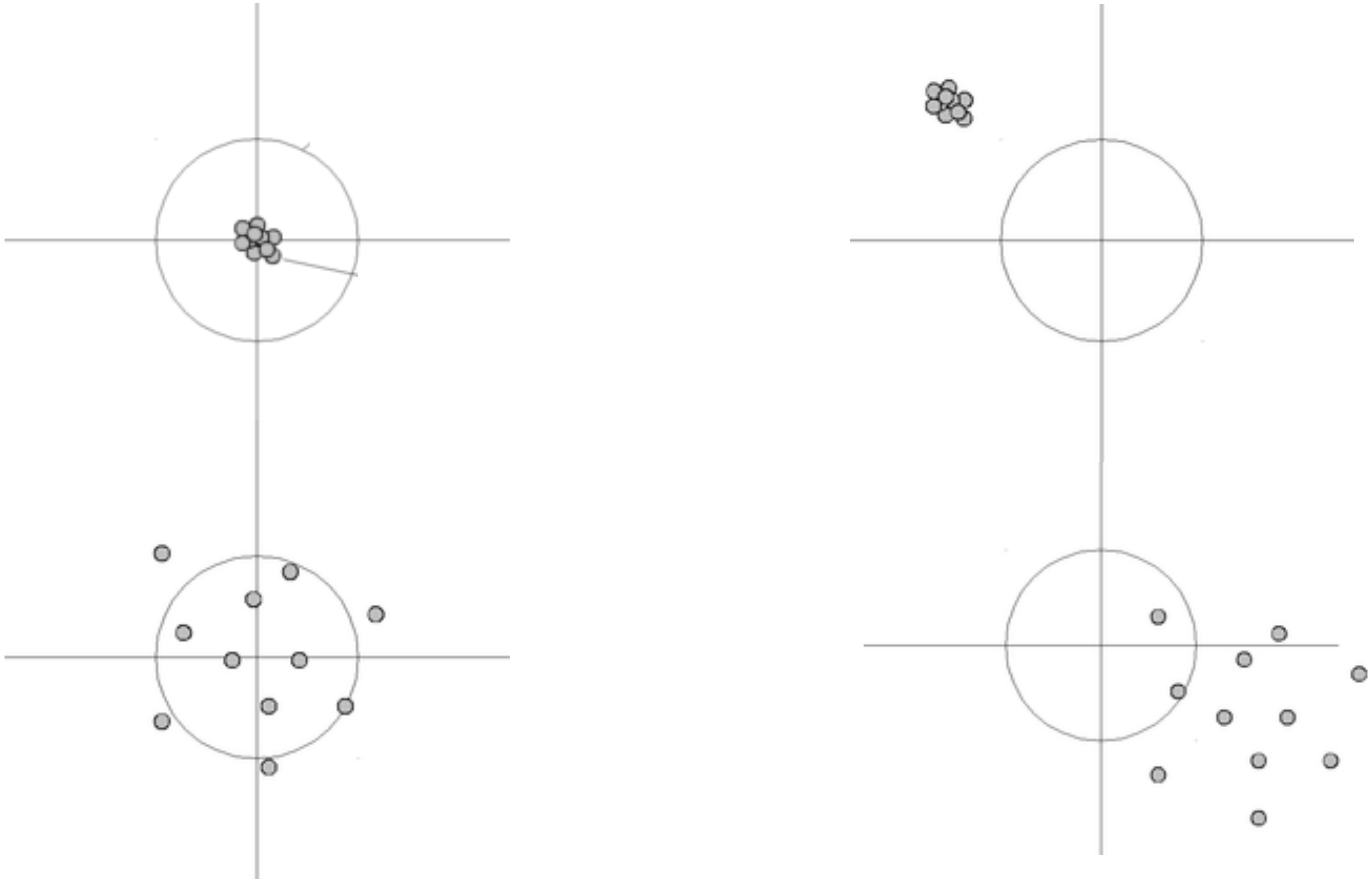
- An estimate is a **consistent** estimator of an unknown value if it converges (probabilistically) to the value being estimated.

Bias / Variance

Low Bias | High Bias

Low Variance

High Variance



Wikipedia: Bias-variance tradeoff

Monte Carlo Methods

- Given distribution $d(X)$ and real-valued function $f(X)$, estimate:

$$\mathbf{E}_d[f(X)] = \sum_x d(x)f(x)$$

- The distribution d is unknown but we can sample $X \sim d$.
- Monte Carlo approximation:

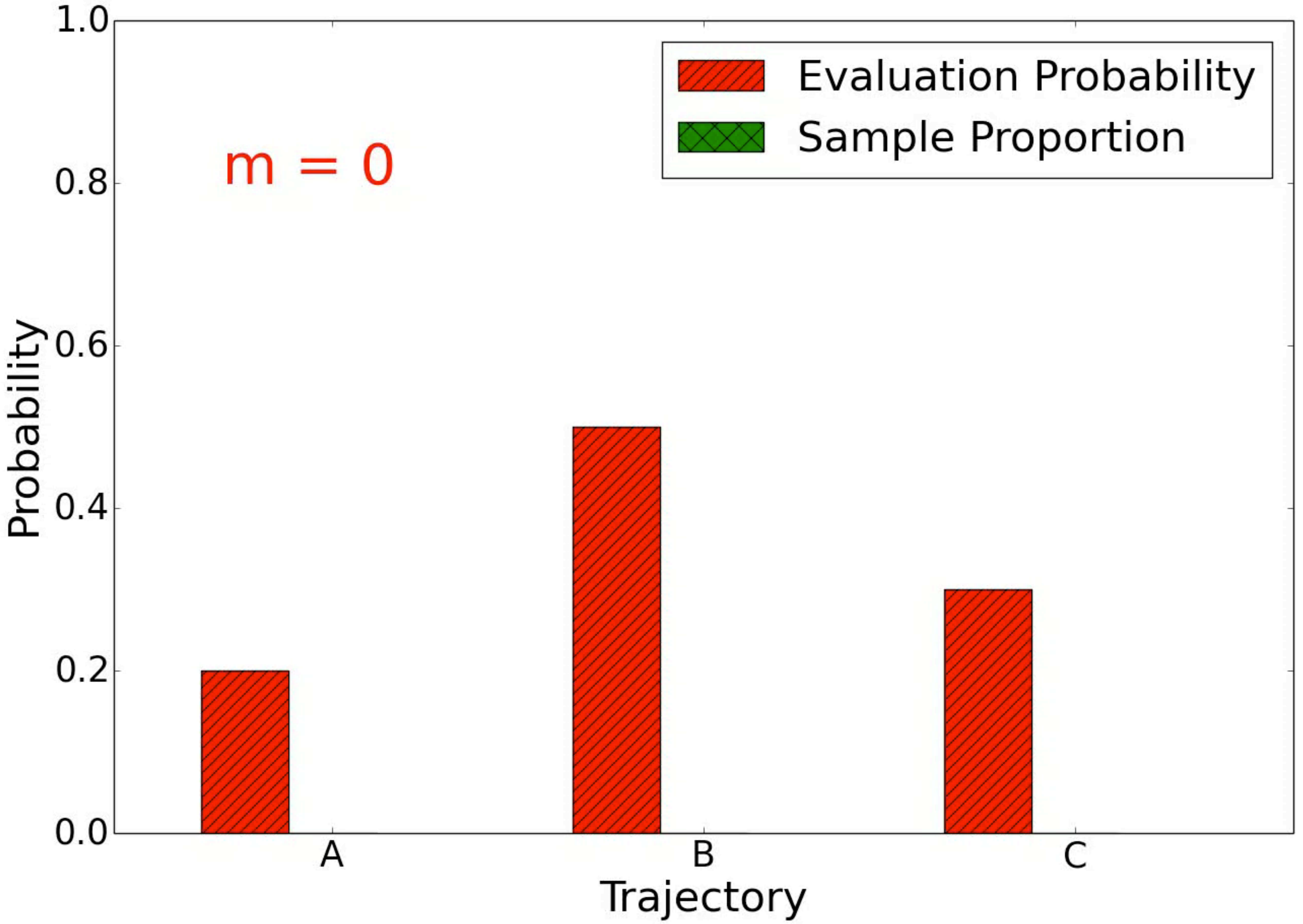
$$\sum_x d(x)f(x) \approx \frac{1}{n}\sum_{i=1}^{n} f(X_i) \qquad X_i \sim d$$

- Law of large numbers tells us that as $n \to \infty$ that error in the approximation goes to zero.
- Error is order $1/\sqrt{n}$.

Monte Carlo Methods

$$\sum_x d(x) f(x) \approx \frac{1}{n} \sum_{i=1}^{n} f(X_i)$$

$$X_i \sim d$$



Monte Carlo in RL

- Given a policy, compute its state- or action-value function.

$$q_\pi(s,a) = \mathbf{E}_\pi[\sum_{t=0}^{T} \gamma^t R_{t+1} \mid S_t = s, A_t = a]$$

- X is a trajectory $S_0, A_0, R_1, S_1, A_1, \ldots R_T, S_T$ generated by following π.

- d is a probability distribution over trajectories that is induced from MDP and π.

$$\Pr(s_0, a_0, r_1, s_1, \ldots r_T, s_t) = \prod_{t=0}^{T-1} \boxed{\pi(a_t \mid s_t) p(s_{t+1}, r_{t+1} \mid s_t, a_t)}$$

- f is the sum of discounted rewards along a trajectory: $\sum_{t=0}^{T} \gamma^t R_{t+1}$

First-Visit Monte Carlo

- Estimate $q_\pi(s_0, a_0)$ for a fixed state, s_0, a_0.
- Assume we always start in state s_0 and all episodes eventually terminate.
- To evaluate policy π, set `total` $\leftarrow 0$, and repeat n times:
  - Start at s_0, take action a_0.
  - Until termination: $S_t, R_t \sim p(S', R \mid S_{t-1}, A_{t-1})$, $\boxed{A_t \sim \pi(A \mid S_t).}$
  - $\texttt{total} \leftarrow \texttt{total} + \sum_{t=0} \gamma^t R_{t+1}$.
- Return $Q_n(s_0, a_0) \leftarrow \texttt{total}/n$
- As $n \rightarrow \infty$, $Q_n(s_0, a_0) \rightarrow q_\pi(s_0, a_0)$.

How would you change for state-values?

What is storage requirement for first-visit Monte Carlo?

Every-Visit Monte Carlo

- In general, we may see the same state multiple times per-episode.
- How does every-visit Monte Carlo differ from first-visit Monte Carlo?
  - Uses return following each occurrence of a state-action pair.
  - May converge faster depending on number of extra occurrences.
- Does every-visit Monte Carlo give unbiased estimates of values?
  - Yes, follows from the Markov assumption. Once we're in a state, how we got there does not matter.
- When would first-visit be preferred to Monte Carlo?

Monte Carlo or Dynamic Programming?

- When would you prefer Monte Carlo methods?
  - No model of the environment or simulation-only model.
  - No Markov state.
- When would you prefer dynamic programming methods?
  - No episode termination.
  - Model known, small number of Markov states and actions.

Policy Evaluation for Control

- Either first-visit or every-visit Monte Carlo can estimate v_π or q_π from experience generated by following policy π. What else is needed for control?

- Must estimate action-values (not state-values). Why?
  - With state-values, the best action is: $a^\star = \arg\max \sum_{s',r} p(s', r \mid s, a)[r + \gamma v_\star(s')]$
  - One-step search requires model to be known.

- Must see all states and actions but π may only select a single action in any given state.
  - Need exploration!

Exploring Starts

- Simple idea to provide exploration.
- How does it work?
  - Non-zero probability of starting in any state and then taking a random action.
- Is it practical?
  - Depends.
  - Inapplicable to continuing problems or problems where we do not control the initial state distribution.
  - Is applicable and potentially beneficial when we DO control the initial state distribution.

Monte Carlo Policy Iteration

- To find $\pi^\star$, start with arbitrary π_0, and alternate:
  - Run Monte Carlo policy evaluation of π_k for n episodes.
  - Make π_{k+1} the greedy policy w.r.t. q_k.
- How large must n be?
- Exploring starts ensures convergence only if all returns averaged come from same policy.
  - Conjectured that there is no need to discard returns as policy changes but no formal proof.

Generalized Policy Iteration

- What is it?
  - We can be quite permissive in how we mix evaluation and improvement.
  - As long as q becomes closer to q_π and π becomes greedy w.r.t. q we will converge to $q_\star, \pi^\star$.
- A general framework for all algorithms we will introduce in this class.
- Do you think this holds when q_π must generalize across states? I.e., increasing the value of $q_\pi(s, a)$ will also increase the value of $q_\pi(s', a')$ for s',a' close to s.

Summary

- Monte Carlo methods learn value functions for the observed return without model knowledge.
- Must learn action-values for control and require an exploration mechanism to ensure coverage of all state-action pairs.
- Basic idea of policy iteration still applies even though we only have an approximate policy evaluation step.

Action Items

- Homework 1 due Thursday @ 9:29 am.
- Start reading chapter 6 for next week.
- Be thinking about final project — proposal due next week.
    - The more concrete your proposal is, the better guidance you will receive!