Advanced Topics in Reinforcement Learning

Lecture 18: Policy Gradients I

Josiah Hanna
University of Wisconsin — Madison

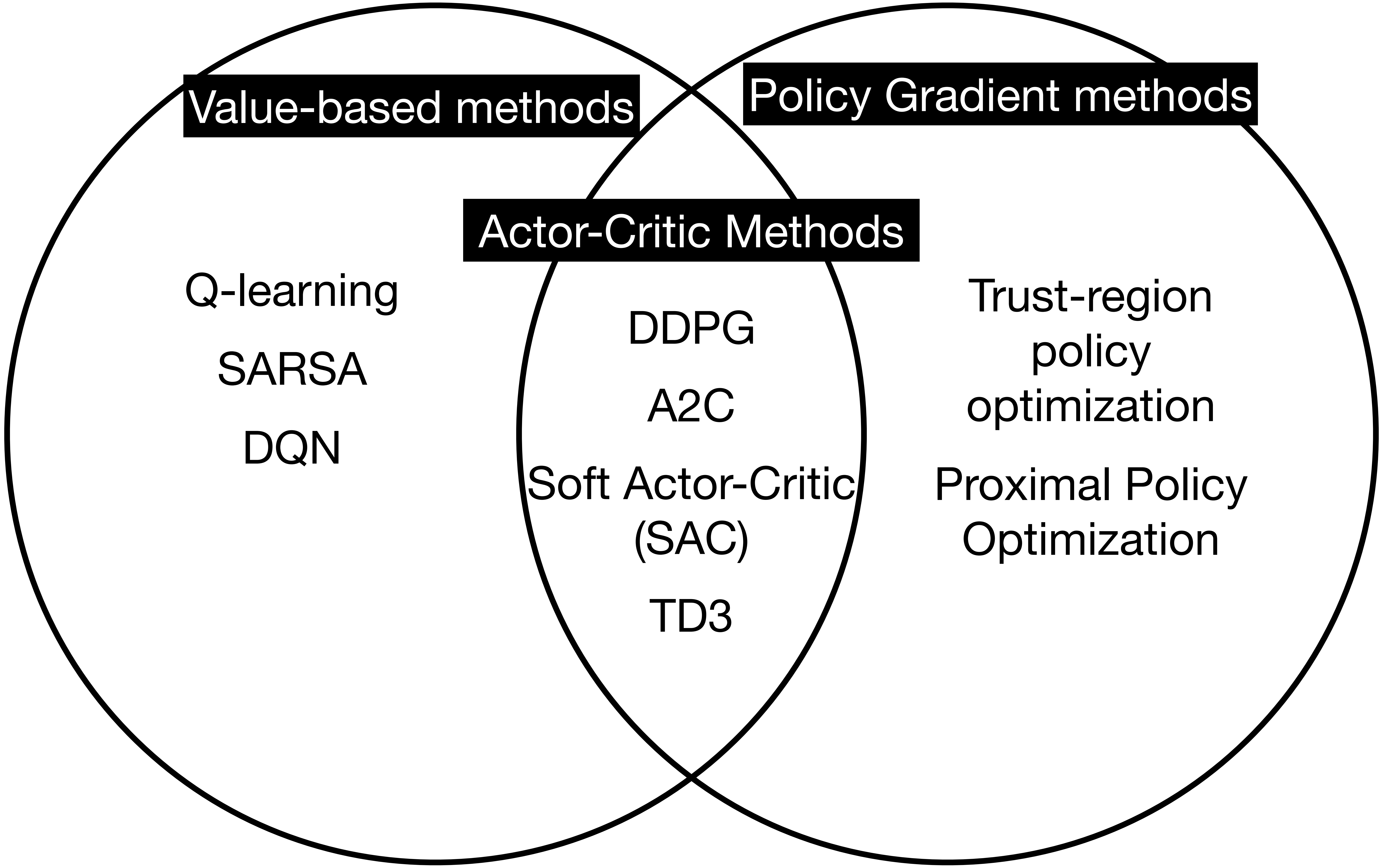
Announcements

- Homework 4 due November 17 (next week).
- Next week: abstraction and hierarchy
- Grading
  - Homework 2 is being re-graded.
  - Literature reviews are being graded.

Homework 3 Feedback

- Overall nice job!
- A few common reasons for point deduction:
  - Hypothesis lacked precision.
  - Unclear what question was of interest.
  - Justification for the hypothesis used experimental results.
  - Analysis only summarized figures.
- An experiment supports or does not support a hypothesis. Does not prove it right.

Model-Free RL



Policy-based RL

- So far the policy is implicit. It is derived from a parameterized action-value function and we learn the action-value function parameters.

- Policy gradient methods use a parameterized policy and learn policy parameters with gradient ascent.

- $\pi_\theta(a \mid s) = \Pr(A_t = a \mid S_t = s, \theta_t = \theta)$

- $J(\theta) = v_{\pi_\theta}(s_0)$

- $\theta_{t+1} \leftarrow \theta_t + \alpha \nabla_\theta \widehat{J(\theta_t)}$

Why Policy-based?

- What are advantages to policy-based methods?
  - More easily handle continuous actions.
  - Policy gradient theorem provides stronger convergence guarantees under function approximation.
  - Useful for partial observability.
  - Policy may be simpler to approximate.
  - Can inject prior knowledge into policy class.
- Disadvantages?
  - May be easier to approximate action-values.
  - Policy is a simple function of the action-values.

Policy Parameterizations

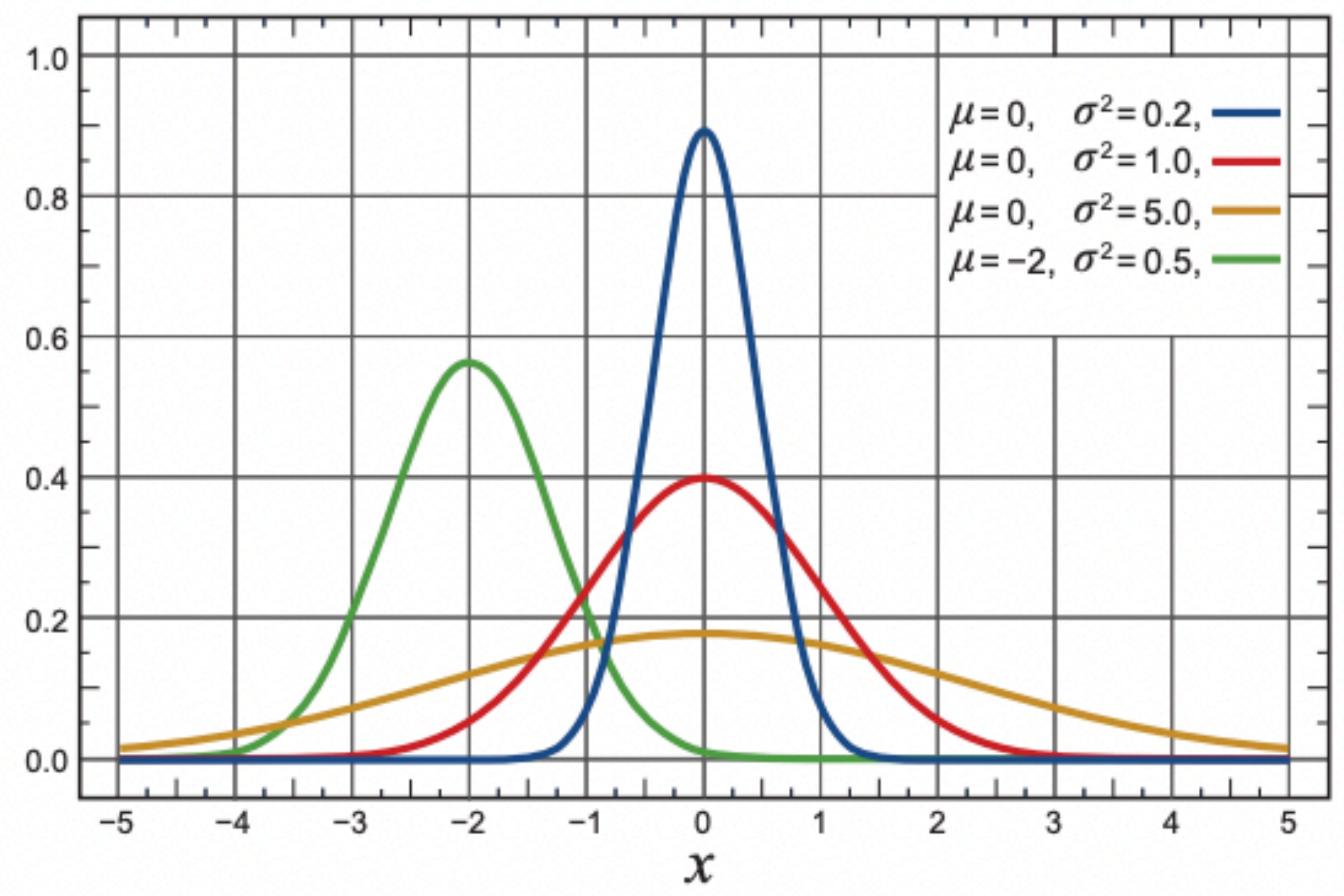
- Policy can be **any** parameterized and differentiable distribution.
- Need $\pi_\theta(A_t = a \mid s)$ and $\nabla_\theta \pi_\theta(A_t = a \mid s)$ exists.

Discrete Action Example

- $\Pr(A_t = a \mid S_t = s) \propto \exp(h(s, a, \theta))$
- $h(s, a, \theta) = f_\theta(s, a)$
- $\nabla_\theta \ln \pi_\theta(a \mid s, \theta) = \nabla_\theta f_\theta(s, a) - \mathbf{E}_{\pi_\theta}[\nabla_\theta f_\theta(s, A)]$

Continuous Action Example

- $\Pr(A_t = a \mid S_t = s) = \mathcal{N}(\mu(s), \sigma(s))$
- $\mu(s) = f_\theta(s); \log \sigma(s) = f_\sigma(s)$
- $\nabla_\theta \ln \pi_\theta(a \mid s, \theta) = \frac{1}{\sigma(s)^2}(a - \mu(s)) \nabla_\theta f_\theta(s)$



Policy Gradient Theorem

- $J(\theta) := v_{\pi_\theta}(s_0) = \sum_a \pi_\theta(a \mid s_0) \sum_{s',r} p(s', r \mid s_0, a)[r + \gamma v_{\pi_\theta}(s')]$

- $\nabla_\theta J(\theta) \propto \sum_s \sum_a \mu_\theta(s) q_{\pi_\theta}(s, a) \nabla_\theta \pi_\theta(a \mid s)$

- The direction in which an infinitesimally small change to θ produces the maximum increase in $J(\theta)$.

- $\nabla_\theta J(\theta)$ does not depend on any gradients of p.

REINFORCE

- Stochastic gradient ascent instead of true gradient ascent. Why?
  - $\nabla_\theta J(\theta)$ can only be estimated.
- $\nabla_\theta J(\theta) \propto \mathbf{E}[\sum_a \nabla_\theta \pi(a \mid S_t) q_\pi(S_t, a)] = \mathbf{E}[\sum_a \pi(a \mid S_t) \frac{\nabla_\theta \pi(a \mid S_t)}{\pi(a \mid S_t)} q_\pi(S_t, a)]$
- Finally, replace $q_\pi(s, a)$ with G_t.
- $\theta_{t+1} \leftarrow \theta_t + \alpha G_t \nabla_\theta \ln \pi(A_t \mid S_t)$
- On- or off-policy?

REINFORCE

REINFORCE: Monte-Carlo Policy-Gradient Control (episodic) for π_*

Input: a differentiable policy parameterization $\pi(a|s, \boldsymbol{\theta})$
Algorithm parameter: step size $\alpha > 0$
Initialize policy parameter $\boldsymbol{\theta} \in \mathbb{R}^{d'}$ (e.g., to $\mathbf{0}$)

Loop forever (for each episode):
  Generate an episode $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$, following $\pi(\cdot|\cdot, \boldsymbol{\theta})$
  Loop for each step of the episode $t = 0, 1, \ldots, T-1$:
    $G \leftarrow \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k$ $\quad (G_t)$
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha \boxed{\gamma^t} G \nabla \ln \pi(A_t|S_t, \boldsymbol{\theta})$

Usually dropped in practice

Is the policy gradient a gradient? Nota and Thomas. 2020.
Bias in Natural Actor-Critic Algorithms. Thomas. 2014.

Baselines

- REINFORCE has high-variance updates.
  - On average, updates improve the policy but any single update could be bad.
  - Reinforce actions in proportion to how much reward follows.
- Replacing G_t with $G_t - b(S_t)$, where $b(S_t)$ is constant w.r.t. action A_t; can substantially lower variance.
  - $$\mathbf{E}[G_t \log \pi(A \mid S)] = \mathbf{E}[G_t \log \pi(A \mid S)] + \underbrace{\mathbf{E}[b(S_t) \log \pi(A \mid S)]}_{=0} = \mathbf{E}[G_t - b(S_t) \log \pi(A \mid S)]$$
  - Reinforce actions in proportion to how much better than average they are.

Optimal Baselines

- In practice, use an approximation for $v_\pi(S_t)$ as the baseline $b(S_t)$.
    - $G_t - v_\pi(S_t) \approx q_\pi(S_t, A_t) - v_\pi(S_t) = A_\pi(S_t, A_t)$, i.e., the advantage function.
- Removes randomness due to $S_t \sim \mu(s)$.
- Better choices exist in theory but seldom used in practice.

Variance Reduction Techniques for Gradient Estimates in Reinforcement Learning. Greensmith et al. 2004.
The Mirage of Action-Dependent Baselines in Reinforcement Learning. Tucker et al. 2018.

Baselines

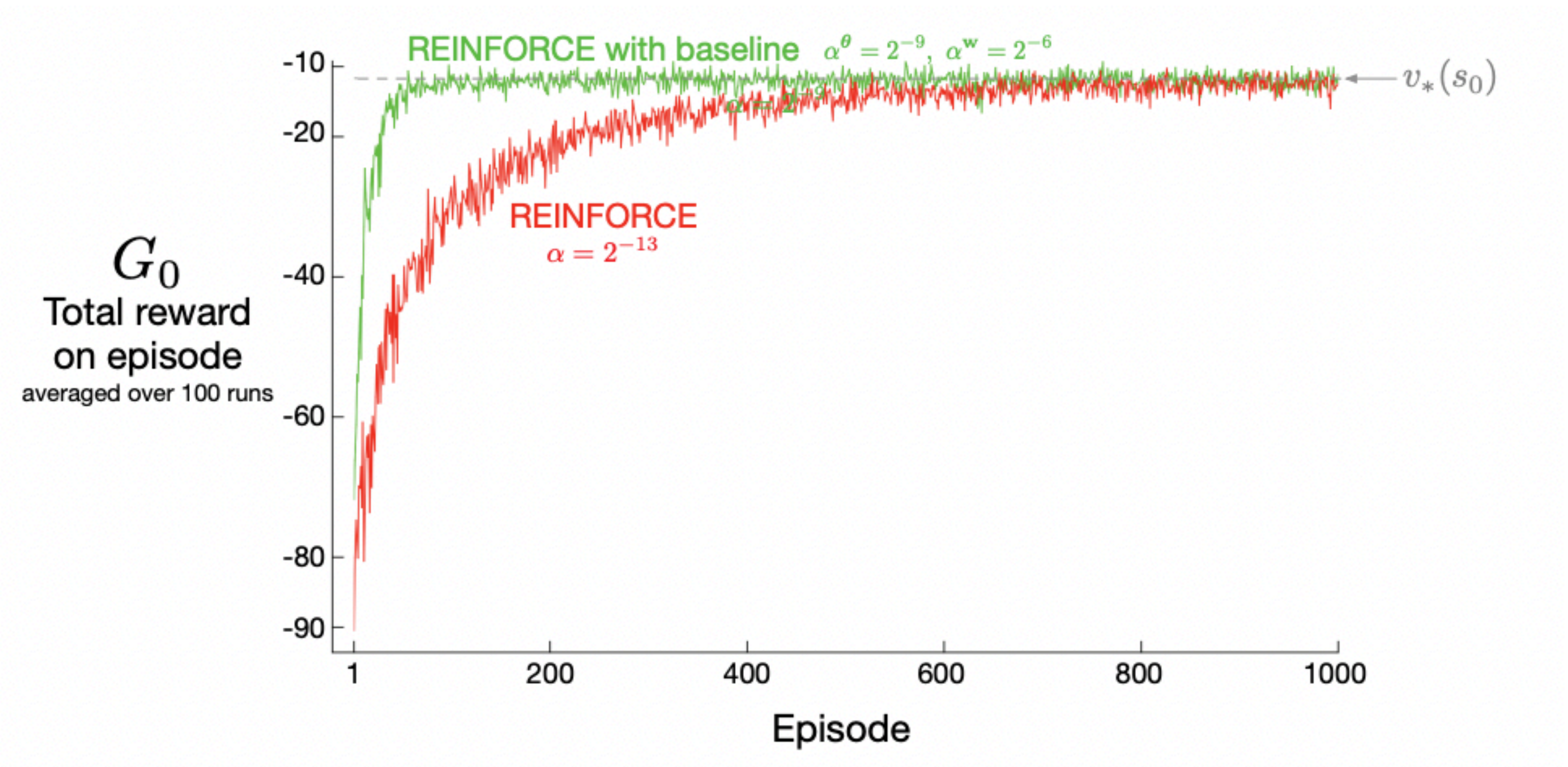


Figure 13.2 in textbook.

Actor-Critic Methods

- REINFORCE uses a learned value function only to lower variance.
  - Monte Carlo return still drives which actions are reinforced.
- Actor-critic methods use learned value functions to drive policy changes.
  - Actor: the policy.
  - Critic: value function.
- Can use state-value or action-value functions:
  - $\theta_{t+1} \leftarrow \theta_t + \alpha \delta_t \nabla_\theta \ln \pi(A_t \mid S_t)$ $\qquad \delta_t \leftarrow R_{t+1} + \gamma \hat{v}(S_{t+1}) - \hat{v}(S_t)$
  - $\theta_{t+1} \leftarrow \theta_t + \alpha \hat{q}(S_t, A_t) \nabla_\theta \ln \pi(A_t \mid S_t)$

Actor-Critic Methods

One-step Actor–Critic (episodic), for estimating $\pi_{\boldsymbol{\theta}} \approx \pi_*$

Input: a differentiable policy parameterization $\pi(a|s, \boldsymbol{\theta})$
Input: a differentiable state-value function parameterization $\hat{v}(s,\mathbf{w})$
Parameters: step sizes $\alpha^{\boldsymbol{\theta}} > 0$, $\alpha^{\mathbf{w}} > 0$
Initialize policy parameter $\boldsymbol{\theta} \in \mathbb{R}^{d'}$ and state-value weights $\mathbf{w} \in \mathbb{R}^{d}$ (e.g., to $\mathbf{0}$)
Loop forever (for each episode):
    Initialize S (first state of episode)
    $I \leftarrow 1$
    Loop while S is not terminal (for each time step):
        $A \sim \pi(\cdot|S, \boldsymbol{\theta})$
        Take action A, observe S', R
        $\delta \leftarrow R + \gamma \hat{v}(S',\mathbf{w}) - \hat{v}(S,\mathbf{w})$ (if S' is terminal, then $\hat{v}(S',\mathbf{w}) \doteq 0$)
        $\mathbf{w} \leftarrow \mathbf{w} + \alpha^{\mathbf{w}} \delta \nabla \hat{v}(S,\mathbf{w})$
        $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha^{\boldsymbol{\theta}} I \delta \nabla \ln \pi(A|S, \boldsymbol{\theta})$
        $\cancel{I \leftarrow \gamma I}$
        $S \leftarrow S'$

Summary

- Policy improvement theorem provides theoretical foundation for guaranteed policy improvement even with function approximation.
- Policy gradient methods can learn with or without learned value functions.
- Actor-critic methods use a learned value function as a replacement for the return in basic policy gradient methods.

Action Items

- Get started on final project!
- Homework 4