535514: Reinforcement Learning Lecture 8 — Stochastic PG and Variance Reduction

Ping-Chun Hsieh

March 18, 2024

Quick Review: Policy Gradient

- **Expressions of Policy Gradient (aka Policy Gradient Theorem)**:

(P1) Total reward: $\nabla_\theta V^{\pi_\theta}(\mu) = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}}\left[G(\tau) \sum_{t=0}^{\infty} \nabla_\theta \log \pi_\theta(a_t \mid s_t)\right]$

(P2) REINFORCE: $\nabla_\theta V^{\pi_\theta}(\mu) = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}}\left[\sum_{t=0}^{\infty} \gamma^t Q^{\pi_\theta}(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t \mid s_t)\right]$

(P3) Q-value and discounted state visitation:

$$\nabla_\theta V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_\mu^{\pi_\theta}} \mathbb{E}_{a \sim \pi_\theta(\cdot \mid s)}\left[Q^{\pi_\theta}(s, a) \nabla_\theta \log \pi_\theta(a \mid s)\right]$$

- Idea: Get an estimate of the true gradient $\nabla_\theta V^{\pi_\theta}(\mu)$

Recall: The REINFORCE Algorithm

- **REINFORCE algorithm (aka *Monte Carlo policy gradient*)**

Step 1: Initialize θ_0 and step size η

Step 2: Sample a trajectory $\tau \sim P_\mu^{\pi_\theta}$ and make the update as

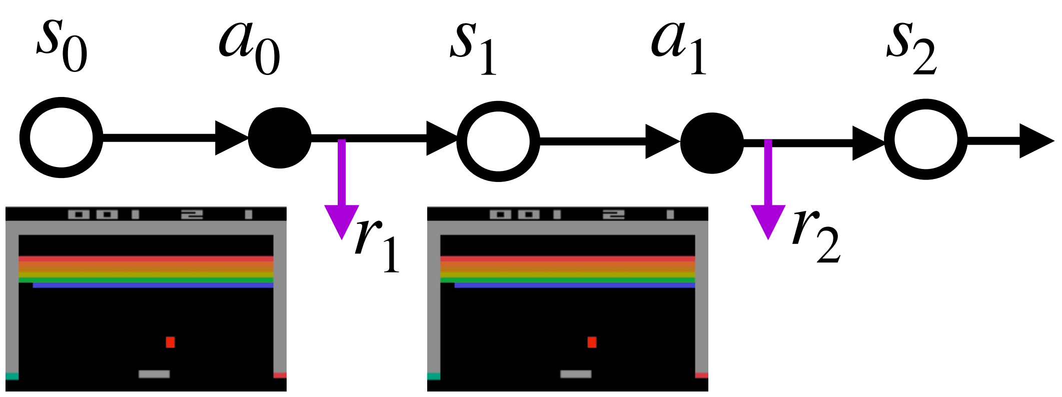
$$\begin{aligned}\theta_{k+1} &= \theta_k + \eta \cdot \hat{\nabla}_\tau \\ &= \theta_k + \eta\Big(\sum_{t=0}^{\infty} \gamma^t G_t(\tau)\, \nabla_\theta \log \pi_\theta(a_t \,|\, s_t)\Big)\end{aligned}$$

(Repeat Step 2 until termination)

- Remark: $\hat{\nabla}_\tau$ is an **unbiased** estimate of $\nabla_\theta V^{\pi_\theta}(\mu)$

Ronald J. Williams, "Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning," ML 1992

- (P2): $\nabla_\theta V^{\pi_\theta}(\mu) = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[\sum_{t=0}^{\infty} \gamma^t Q^{\pi_\theta}(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t \mid s_t) \right]$
- Show that $\hat{\nabla}_\tau$ is an **unbiased** estimate of $\nabla_\theta V^{\pi_\theta}(\mu)$



$$G_t(\tau) := \sum_{m=t}^{\infty} \gamma^m r_{m+1}$$

$$\hat{\nabla}_\tau := \sum_{t=0}^{\infty} \gamma^t G_t(\tau) \nabla_\theta \log \pi_\theta(a_t \mid s_t)$$

More Generally: Stochastic Gradient Descent (SGD) for Stochastic Optimization

Stochastic Optimization:

$$\theta^* = \arg\min_{\theta \in \Theta} F(\theta), \text{ where } F(\theta) := \mathbb{E}_{\xi}[f(\theta; \xi)]$$

where ξ is the randomness in our problem

Stochastic Gradient Descent:

$$\theta_{k+1} = \theta_k - \eta_k \cdot \mathbb{E}[\nabla_\theta f(\theta_k; \xi)]$$

(GD)

$$\theta_{k+1} = \theta_k - \eta_k \cdot g(\theta_k; \xi_k)$$

(SGD)

- $g(\theta_k; \xi_k)$ is an estimate of the true gradient (constructed from 1 or multiple samples)
- Advantage: SGD has a low computational cost in each iteration

H. Robbins and S. Monro, "A Stochastic Approximation Method," The Annals of Mathematical Statistics, 1951

Almost All ML Problems are Stochastic Optimization Problems!

Policy Optimization in RL:

$$\max_{\theta} V^{\pi_\theta}(\mu)$$

where $V^{\pi_\theta}(\mu) = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}}[G(\tau)]$

Regression / Classification:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim D}\left[\ell\left(f_\theta(x), y\right)\right]$$

where ℓ is some loss function

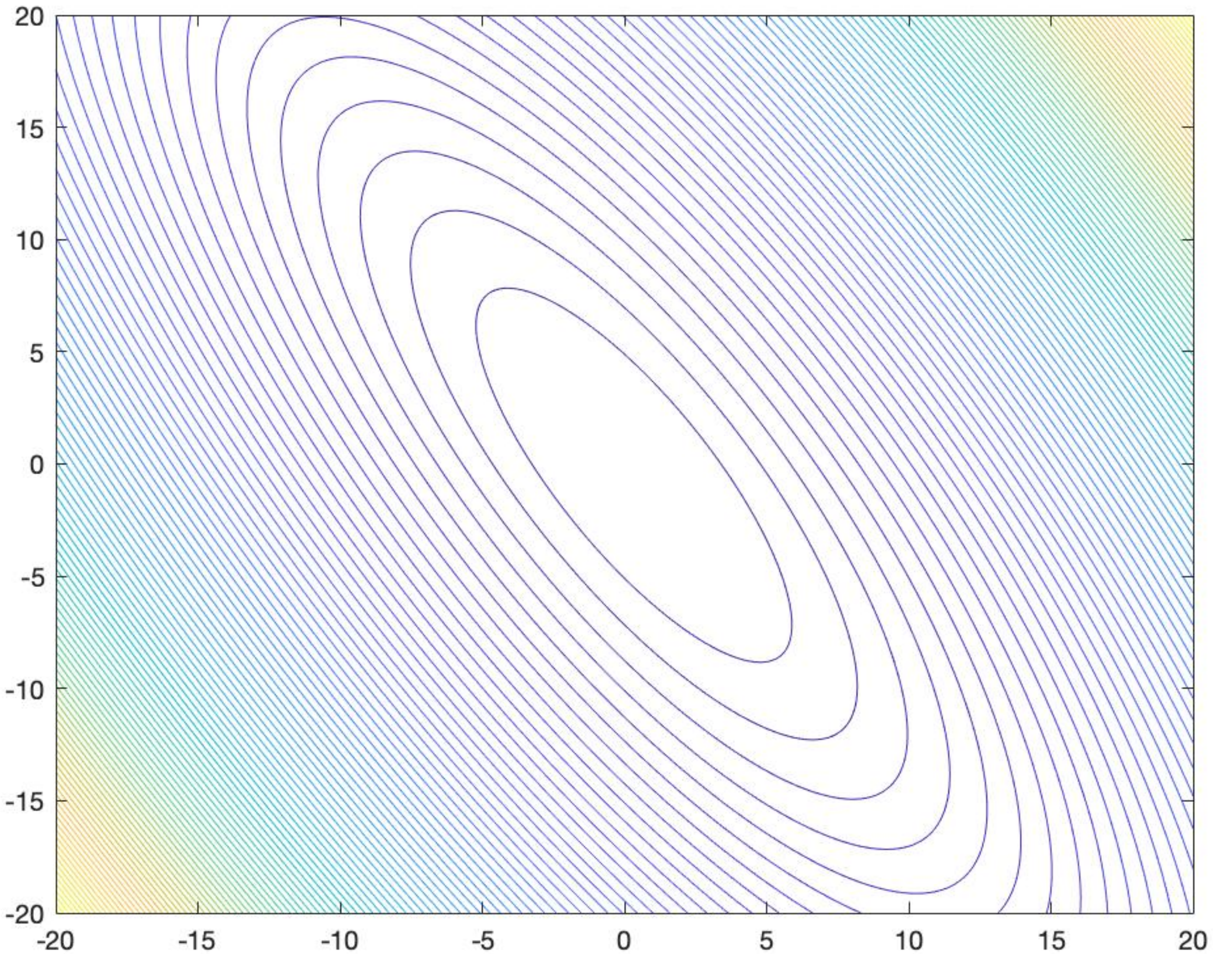
Fine-Tuning of Language Models [Ziegler et al., 2020]**:**

$$\max_{\theta} \mathbb{E}_{x\sim D}\left[\mathbb{E}_{y\sim\pi_\theta(\cdot|x)}\left[r_\phi(x,y)\right] - \beta D_{KL}\left(\pi_\theta(\cdot\,|x)\|\pi_{ref}(\cdot\,|x)\right)\right]$$

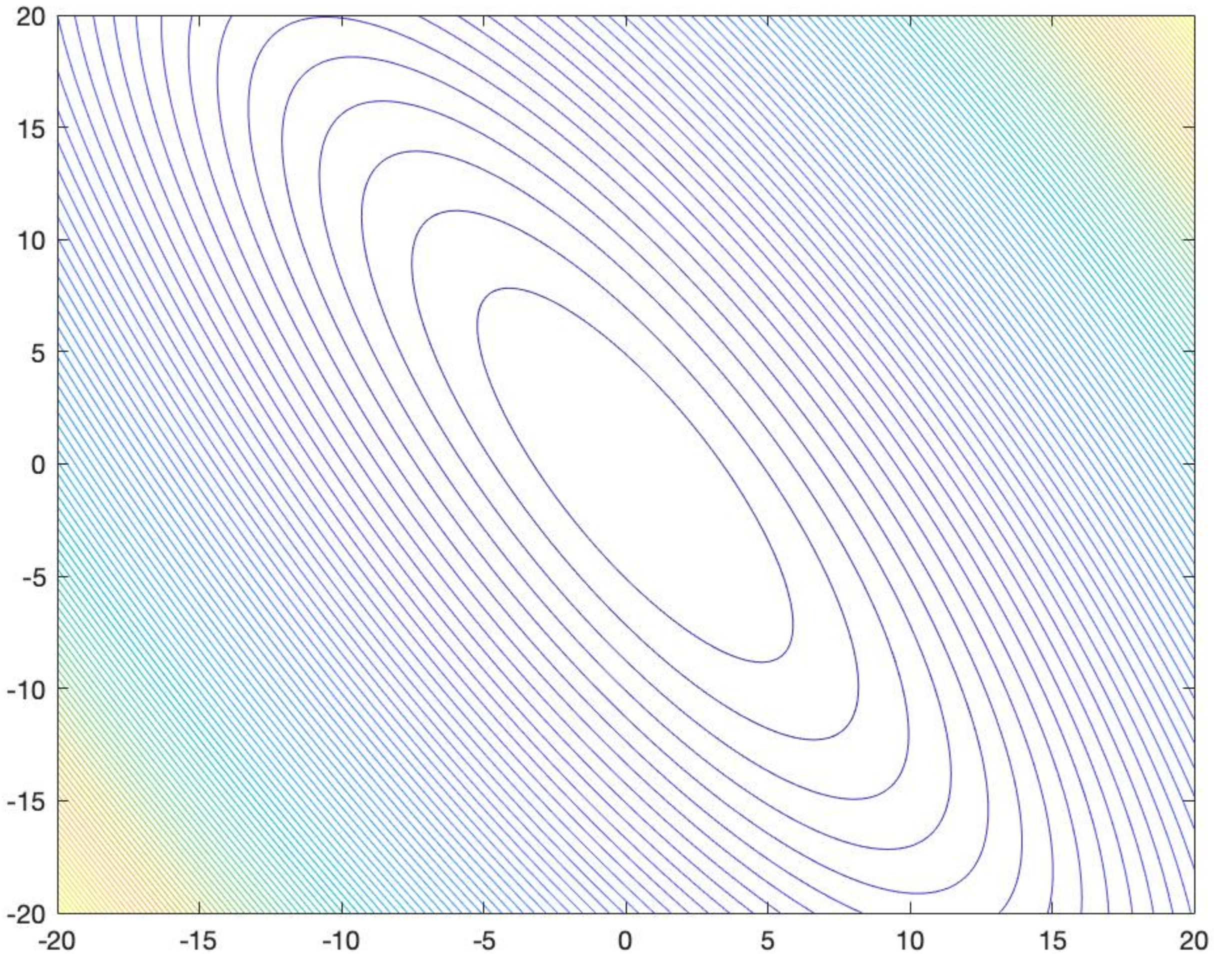
D. M. Ziegler, N. Stiennon, J. Wu, T. B. Brown, A. Radford, D. Amodei, P. Christiano, and G. Irving, "Fine-tuning language models from human preferences," 2020

Visualization: SGD vs GD

- SGD usually exhibits more “random” behavior than GD



GD



SGD

SGD: A Special Case of "Stochastic Approximation"

A STOCHASTIC APPROXIMATION METHOD[1]

By Herbert Robbins and Sutton Monro
University of North Carolina

1. Summary. Let $M(x)$ denote the expected value at level x of the response to a certain experiment. $M(x)$ is assumed to be a monotone function of x but is unknown to the experimenter, and it is desired to find the solution $x = \theta$ of the equation $M(x) = \alpha$, where α is a given constant. We give a method for making successive experiments at levels $x_1, x_2, \cdots$ in such a way that x_n will tend to θ in probability.

2. Introduction. Let $M(x)$ be a given function and α a given constant such that the equation

$$M(x) = \alpha \tag{1}$$

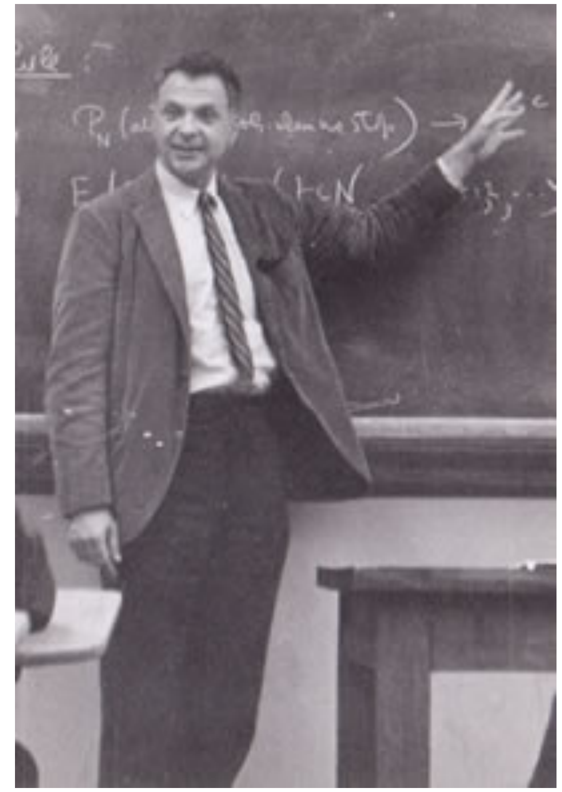
has a unique root $x = \theta$. There are many methods for determining the value of θ by successive approximation. With any such method we begin by choosing one or more values $x_1, \cdots, x_r$ more or less arbitrarily, and then successively obtain new values x_n as certain functions of the previously obtained $x_1, \cdots, x_{n-1}$, the values $M(x_1), \cdots, M(x_{n-1})$, and possibly those of the derivatives $M'(x_1), \cdots, M'(x_{n-1})$, etc. If

$$\lim_{n\to\infty} x_n = \theta, \tag{2}$$

irrespective of the arbitrary initial values $x_1, \cdots, x_r$, then the method is effective for the particular function $M(x)$ and value α. The speed of the convergence in (2) and the ease with which the x_n can be computed determine the practical utility of the method.

We consider a stochastic generalization of the above problem in which the nature of the function $M(x)$ is unknown to the experimenter. Instead, we suppose that to each value x corresponds a random variable $Y = Y(x)$ with distribution function $Pr[Y(x) \le y] = H(y \mid x)$, such that

$$M(x) = \int_{-\infty}^{\infty} y \, dH(y \mid x) \tag{3}$$



Herbert Robbins

Sutton Monro

"Stochastic approximation" is the core of many RL methods, e.g., Q-learning and TD learning

A seminal paper on stochastic approximation in 1951
(Cited for more than 12000 times)

Can we design RL algorithms by (P1) and (P3)?

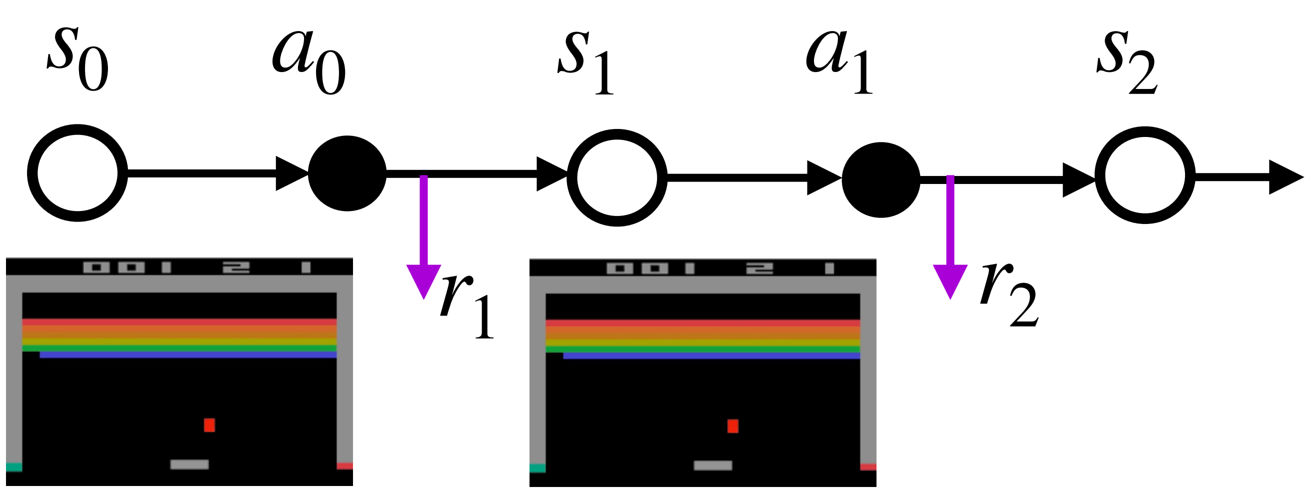
An Alternative Monte-Carlo PG Algorithm

- Recall (P1): $\nabla_\theta V^{\pi_\theta}(\mu) = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}}\left[G(\tau) \sum_{t=0}^{\infty} \nabla_\theta \log \pi_\theta(a_t \mid s_t)\right]$

Step 1: In each iteration k, draw a trajectory $\tau = (s_0, a_0, r_1, s_1, a_1 \cdots)$ under π_θ and μ, and then construct:

$$G(\tau) := \sum_{t=0}^{\infty} \gamma^t r_t$$

$$\bar{\nabla}_\tau := R(\tau) \sum_{t=0}^{\infty} \nabla_\theta \log \pi_\theta(a_t \mid s_t)$$



Step 2: Apply $\theta_{k+1} = \theta_k + \eta \cdot \bar{\nabla}_\tau$

- Question: Is $\bar{\nabla}_\tau$ an unbiased estimate of $\nabla_\theta V^{\pi_\theta}(\mu)$?
- Question: Any difference from REINFORCE?

How About Using (P3)?

- Recall (P3): $\nabla_\theta V^{\pi_\theta}(\mu) = \dfrac{1}{1-\gamma}\mathbb{E}_{s\sim d_\mu^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\left[Q^{\pi_\theta}(s,a)\nabla_\theta \log \pi_\theta(a\,|\,s)\right]$

Step 1: In each iteration k, draw a batch B of n state-action pairs by following π_θ and construct

$$\tilde{\nabla}_\tau := \frac{1}{1-\gamma}\cdot\Big(\frac{1}{n}\sum_{(s,a)\in B} Q^{\pi_\theta}(s,a)\nabla_\theta \log \pi_\theta(a\,|\,s)\Big)$$

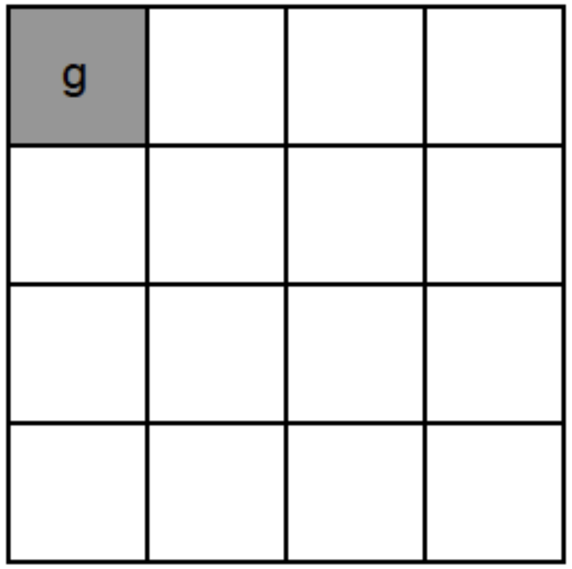
Step 2: Apply $\theta_{k+1} = \theta_k + \eta\cdot\tilde{\nabla}_\tau$

- Question: What does "draw samples by following π_θ" mean? Are the samples i.i.d.?

One Untold Secret in RL Community…

$\tilde{\nabla}_\tau$ for (P3) is actually NOT an unbiased estimator of the true PG!

- But for large n, $\tilde{\nabla}_\tau$ can still nicely approximate the true PG (Why?)



A Fundamental Property:

Empirical distribution uniformly approximates the true distribution!

(This is known as Gilvenko-Cantelli Theorem)

https://www.wolfram.com/mathematica/new-in-8/probability-and-statistics-solvers-and-properties/demonstrate-the-glivenko-cantelli-theorem.html

Gilvenko-Cantelli Theorem (Formally)

Empirical distribution uniformly approximates the true distribution

- Let $\{X_n, n \geq 1\}$ be a sequence of i.i.d. random variables with a common CDF F
- Define the empirical CDF as $\hat{F}_n(x) := \frac{1}{n}\sum_{i=1}^{n} I\{X_i \leq x\}$
- Define $D_n := \sup_{x \in \mathbb{R}} |\hat{F}_n(x) - F(x)|$

- **Gilvenko-Cantelli Theorem:**

$$D_n \to 0, \text{ as } n \to \infty$$

- This result could be directly extended to Markov chains

Stute and Schmann, "A General Gilvenko-Cantelli Theorem for Stationary Sequences of Random Observations," 1980

Yet Another Untold Secret in RL Community…

- RL people usually ignore the effect of γ on $d_{\mu}^{\pi_\theta}$ (which is not theoretically justified)

Is the Policy Gradient a Gradient?

Chris Nota
College of Information and Computer Sciences
University of Massachusetts Amherst
cnota@cs.umass.edu

Philip S. Thomas
College of Information and Computer Sciences
University of Massachusetts Amherst
pthomas@cs.umass.edu

ABSTRACT

The policy gradient theorem describes the gradient of the expected discounted return with respect to an agent's policy parameters. However, most policy gradient methods drop the discount factor from the state distribution and therefore do not optimize the discounted objective. What do they optimize instead? This has been an open question for several years, and this lack of theoretical clarity has lead to an abundance of misstatements in the literature. We answer this question by proving that the update direction approximated by most methods is not the gradient of any function. Further, we argue that algorithms that follow this direction are not guaranteed to converge to a "reasonable" fixed point by constructing a counterexample wherein the fixed point is globally *pessimal* with respect to both the discounted and undiscounted objectives. We motivate this work by surveying the literature and showing that there remains a widespread misunderstanding regarding discounted policy gradient methods, with errors present even in highly-cited papers published at top conferences.

is pessimal, regardless of whether the discounted or undiscounted objective is considered.

The analysis in this paper applies to nearly all state-of-the-art policy gradient methods. In Section 6, we review all of the policy gradient algorithms included in the popular `stable-baselines` repository [9] and their associated papers, including A2C/A3C [13], ACER [28], ACKTR [30], DDPG [11], PPO [18], TD3 [6], TRPO [16], and SAC [8]. We motivate this choice in Section 6, but we note that all of these papers were published at top conferences[1] and have received hundreds or thousands of citations. We found that all of the implementations of the algorithms used the "incorrect" policy gradient that we discuss in this paper. While this is a valid algorithmic choice if properly acknowledged, we found that only *one* of the eight papers acknowledged this choice, while three of the papers made erroneous claims regarding the discounted policy gradient and others made claims that were misleading. The purpose of identifying these errors is not to criticize the authors or the algorithms, but to draw attention to the fact that confusion regarding the behavior of policy gradient algorithm exists at the very core of the RL community and has gone largely unnoticed by reviewers.

Philip Thomas

Chris Nota and Philip Thomas, "Is the Policy Gradient a Gradient?," AAMAS 2020

Variance Reduction

Variance Issue of Estimated Policy Gradient

- Monte-Carlo policy gradient is known to have *high variance*

- High variance → *a large number of steps* is needed to obtain a good estimate of the policy gradient

- Recall: REINFORCE update

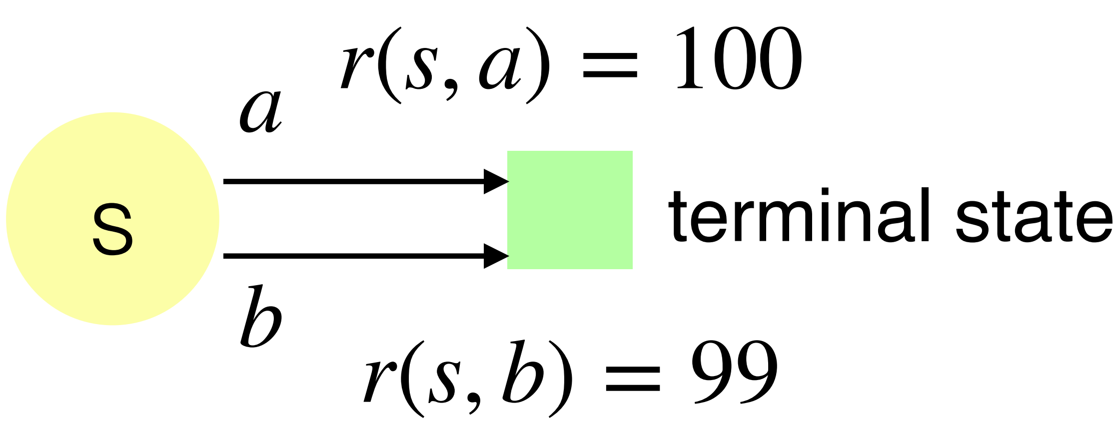
$$\theta_{k+1} = \theta_k + \eta\Big(\sum_{t=0}^{T-1}\gamma^t G_t(\tau)\,\nabla_\theta \log \pi_\theta(a_t\,|\,s_t)\Big)$$

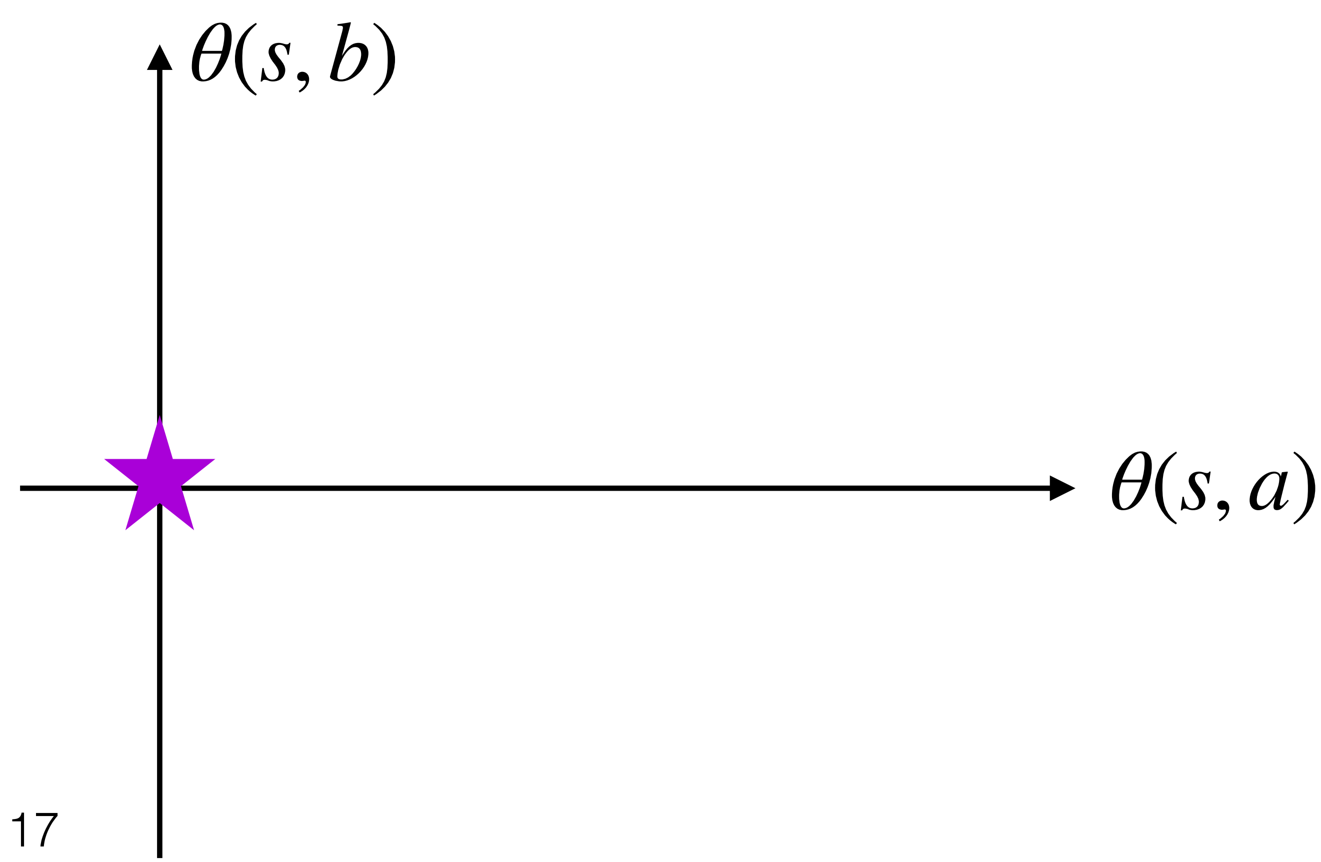
  - This update suffers from high variance (why?)

Why Variance Issue? A Motivating Example

(P2) REINFORCE: $\nabla_\theta V^{\pi_\theta}(\mu) = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}}\left[\sum_{t=0}^{\infty} \gamma^t Q^{\pi_\theta}(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t | s_t) \right]$

- Example: Simple 1-state, 2-action MDP
  - Softmax policy with two parameters $\theta(s, a), \theta(s, b)$
  - $\dfrac{\partial \log \pi_\theta(s, a)}{\partial \theta(s, a)} = 1 - \pi_\theta(s, a), \ \dfrac{\partial \log \pi_\theta(s, a)}{\partial \theta(s, b)} = - \pi_\theta(s, b)$
  - Currently $\theta(s, a) = \theta(s, b) = 0$ and hence $\pi_\theta(s, a) = \pi_\theta(s, b) = 0.5$





- Question: Plot the true gradient and the sample gradient?

Solutions to Variance Reduction

(S1) Baseline ($\equiv$ Set a reference level)

(S2) Critic ($\equiv$ Learn $Q(s, a)$)

(S3) Baseline + Critic ($\equiv$ Advantage function)

(S1) Reducing Variance Using a Baseline

- Recall: (P3) Q-value and discounted state visitation:

$$\nabla_\theta V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma} \mathbb{E}_{s\sim d_\mu^{\pi_\theta}} \mathbb{E}_{a\sim\pi_\theta(\cdot|s)} \left[Q^{\pi_\theta}(s,a) \nabla_\theta \log \pi_\theta(a\,|\,s) \right]$$

- Subtract a baseline function $B(s)$ from the policy gradient

$$\mathbb{E}_{s\sim d_\mu^{\pi_\theta}} \mathbb{E}_{a\sim\pi_\theta(\cdot|s)} \left[\left(Q^{\pi_\theta}(s,a) - B(s) \right) \nabla_\theta \log \pi_\theta(a\,|\,s) \right]$$

- The introduction of $B(s)$ does not change the expectation

$$\mathbb{E}_{s\sim d_\mu^{\pi_\theta}} \mathbb{E}_{a\sim\pi_\theta(\cdot|s)} \left[B(s) \nabla_\theta \log \pi_\theta(a\,|\,s) \right]$$

$$= \sum_s d^{\pi_\theta}(s) \sum_a \pi_\theta(a\,|\,s) \nabla_\theta \log \pi_\theta(a\,|\,s) B(s)$$

$$= \sum_s d^{\pi_\theta}(s) B(s) \nabla_\theta \sum_a \pi_\theta(a\,|\,s) = 0$$

(S1) Reducing Variance Using a Baseline (Cont.)

- Recall: (P2) REINFORCE:

$$\nabla_\theta V^{\pi_\theta}(\mu) = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}}\Big[\sum_{t=0}^{\infty} \gamma^t Q^{\pi_\theta}(s_t, a_t)\, \nabla_\theta \log \pi_\theta(a_t \mid s_t)\Big]$$

- Subtract a baseline function $B(s)$ from the policy gradient

$$\mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}}\Big[\sum_{t=0}^{\infty} \gamma^t \big(Q^{\pi_\theta}(s_t, a_t) - B(s_t)\big)\, \nabla_\theta \log \pi_\theta(a_t \mid s_t)\Big]$$

- The introduction of $B(s)$ does not change the expectation

$$\mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}}\big[B(s_t)\, \nabla_\theta \log \pi_\theta(a_t \mid s_t)\big]$$

$$= \sum_s P(s_t = s) \sum_a \pi_\theta(a \mid s)\, \nabla_\theta \log \pi_\theta(a \mid s) B(s)$$

$$= \sum_s P(s_t = s) B(s)\, \nabla_\theta \sum_a \pi_\theta(a \mid s) = 0$$

REINFORCE with Baseline

- REINFORCE with baseline

Step 1: Initialize θ_0 and step size η

Step 2: Sample a trajectory $\tau \sim P_\mu^{\pi_\theta}$ and make the update as

$$\theta_{k+1} = \theta_k + \eta \left(\sum_{t=0}^{\infty} \gamma^t \left(G_t - B(s_t) \right) \nabla_\theta \log \pi_\theta(a_t \mid s_t) \right)$$

(Repeat Step 2 until termination)

By How Much Can $B(s)$ Reduce Variance?

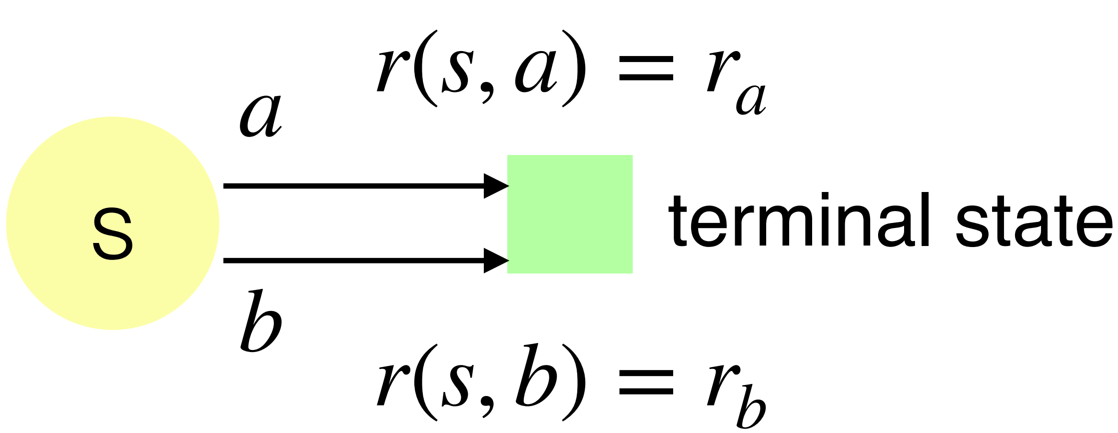
- In REINFORCE, estimate policy gradient with G_t

Original: $$\nabla_\theta V^{\pi_\theta}(\mu) \approx \sum_{t=0}^{\infty} \gamma^t G_t \nabla_\theta \log \pi_\theta(a_t \mid s_t)$$

With baseline: $$\nabla_\theta V^{\pi_\theta}(\mu) \approx \sum_{t=0}^{\infty} \gamma^t (G_t - B(s_t)) \nabla_\theta \log \pi_\theta(a_t \mid s_t)$$

- Question: What's the variance of the above two estimates?

(Let's get some intuition by considering the 1-state MDP example)



$$\mathbb{V}\Big[G_0 \frac{\partial}{\partial \theta_i} \log \pi_\theta(a_0 \,|\, s_0)\Big]$$

(Original)

$$= \sum_s P(s_0 = s)\Big(\mathbb{E}[G_0^2(\frac{\partial}{\partial \theta_i} \log \pi_\theta(a_0 \,|\, s))^2 \,|\, s]\Big) - \Big(\sum_s P(s_0 = s)\mathbb{E}[G_0 \frac{\partial}{\partial \theta_i} \log \pi_\theta(a_0 \,|\, s_0) \,|\, s]\Big)^2$$

$$= \sum_s P(s_0 = s)\Big(\sum_a \pi_\theta(a \,|\, s)\mathbb{E}[G_0^2(\frac{\partial}{\partial \theta_i} \log \pi_\theta(a \,|\, s))^2 \,|\, s, a]\Big) - \Big(\sum_s P(s_0 = s)\sum_a \pi_\theta(a \,|\, s)\mathbb{E}[G_t \frac{\partial}{\partial \theta_i} \log \pi_\theta(a \,|\, s) \,|\, s, a]\Big)^2$$

$$= \sum_s P(s_0 = s)\Big(\sum_a \pi_\theta(a \,|\, s)\big(\frac{\partial}{\partial \theta_i} \log \pi_\theta(a \,|\, s)\big)^2 \mathbb{E}[G_0^2 \,|\, s, a]\Big) - \Big(\sum_s P(s_0 = s)\sum_a \pi_\theta(a \,|\, s)\frac{\partial}{\partial \theta_i} \log \pi_\theta(a \,|\, s)\mathbb{E}[G_0 \,|\, s, a]\Big)^2$$

(With baseline)

$$\mathbb{V}\Big[(G_0 - B(s_0))\frac{\partial}{\partial\theta_i}\log\pi_\theta(a_0\,|\,s_0)\Big]$$

$$= \sum_s P(s_0 = s)\Big(\mathbb{E}[(G_0 - B(s))^2(\frac{\partial}{\partial\theta_i}\log\pi_\theta(a_0\,|\,s))^2\,|\,s]\Big)$$

$$-\Big(\sum_s P(s_0 = s)\mathbb{E}[(G_0 - B(s))\frac{\partial}{\partial\theta_i}\log\pi_\theta(a_0\,|\,s_0)\,|\,s]\Big)^2$$

$$= \sum_s P(s_0 = s)\Big(\sum_a \pi_\theta(a\,|\,s)(\mathbb{E}[(G_0 - B(s))^2(\frac{\partial}{\partial\theta_i}\log\pi_\theta(a\,|\,s))^2\,|\,s, a]\Big)$$

$$-\Big(\sum_s P(s_0 = s)\sum_a \pi_\theta(a\,|\,s)\mathbb{E}[(G_0 - B(s))\frac{\partial}{\partial\theta_i}\log\pi_\theta(a\,|\,s)\,|\,s, a]\Big)^2$$

$$= \sum_s P(s_0 = s)\Big(\sum_a \pi_\theta(a\,|\,s)\big(\frac{\partial}{\partial\theta_i}\log\pi_\theta(a\,|\,s)\big)^2\mathbb{E}[(G_0 - B(s))^2\,|\,s, a]\Big)$$

$$-\Big(\sum_s P(s_0 = s)\sum_a \pi_\theta(a\,|\,s)\frac{\partial}{\partial\theta_i}\log\pi_\theta(a\,|\,s)\mathbb{E}[G_0\,|\,s, a]\Big)^2$$

Quantifying Variance Reduction By $B(s)$

$$\mathbb{V}\Big[G_0 \frac{\partial}{\partial \theta_i} \log \pi_\theta(a_0 \mid s_0)\Big] - \mathbb{V}\Big[(G_0 - B(s_0)) \frac{\partial}{\partial \theta_i} \log \pi_\theta(a_0 \mid s_0)\Big]$$

$$= \sum_s P(s_0 = s) \left(\sum_a \underbrace{\pi_\theta(a \mid s) \Big(\frac{\partial}{\partial \theta_i} \log \pi_\theta(a \mid s) \Big)^2}_{:= c_a} \big(\mathbb{E}[G_0^2 \mid s, a] - \mathbb{E}[(G_0 - B(s))^2 \mid s, a]\big) \right)$$

$$= \sum_s P(s_0 = s) \sum_a c_a \big(\mathbb{E}[2B(s)G_0 - B(s)^2 \mid s, a]\big)$$

- Suppose $\mathbb{E}[G_0 \mid s, a] \equiv Q^{\pi_\theta}(s, a) \approx V^{\pi_\theta}(s)$, then we may choose $B(s) = V^{\pi_\theta}(s)$
- In practice, $B(s) = V^{\pi_\theta}(s)$ is a popular choice

(S2) Reducing Variance Using a Critic

- Monte Carlo policy gradient requires G_t, which has high variance

- Recall:

  (P3) Q-value and discounted state visitation:

$$\nabla_\theta V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\Big[Q^{\pi_\theta}(s,a)\,\nabla_\theta \log \pi_\theta(a\,|\,s)\Big]$$

- Idea: Learn a critic to estimate action-value function

$$Q_w(s,a) \approx Q^{\pi_\theta}(s,a)$$

(S2) Reducing Variance Using a Critic (Cont.)

- Actor-critic algorithms maintain 2 sets of parameters
  - Critic: updates action-value function parameter w
  - Actor: updates policy parameters θ, in the direction suggested by critic

- Actor-critic algorithms follow an approximate policy gradient

$$\nabla_\theta V^{\pi_\theta}(\mu) \approx \frac{1}{1-\gamma} \mathbb{E}_{s \sim d^{\pi_\theta}} \mathbb{E}_{a \sim \pi_\theta(\cdot|s)} \left[Q_w(s,a) \nabla_\theta \log \pi_\theta(a|s) \right]$$

- Stochastic PG methods would use the following for policy update

$$Q_w(s,a) \nabla_\theta \log \pi_\theta(a|s)$$

Q-Value Actor-Critic Algorithm

- A simple actor-critic algorithm based on a Q-function critic

Step 1: Initialize θ, w, step size η, s_0 and sample $a_0 \sim \pi_\theta$

Step 2: For each step $t = 0,1,2,\cdots$

Sample reward r_{t+1}; sample transition s_{t+1}

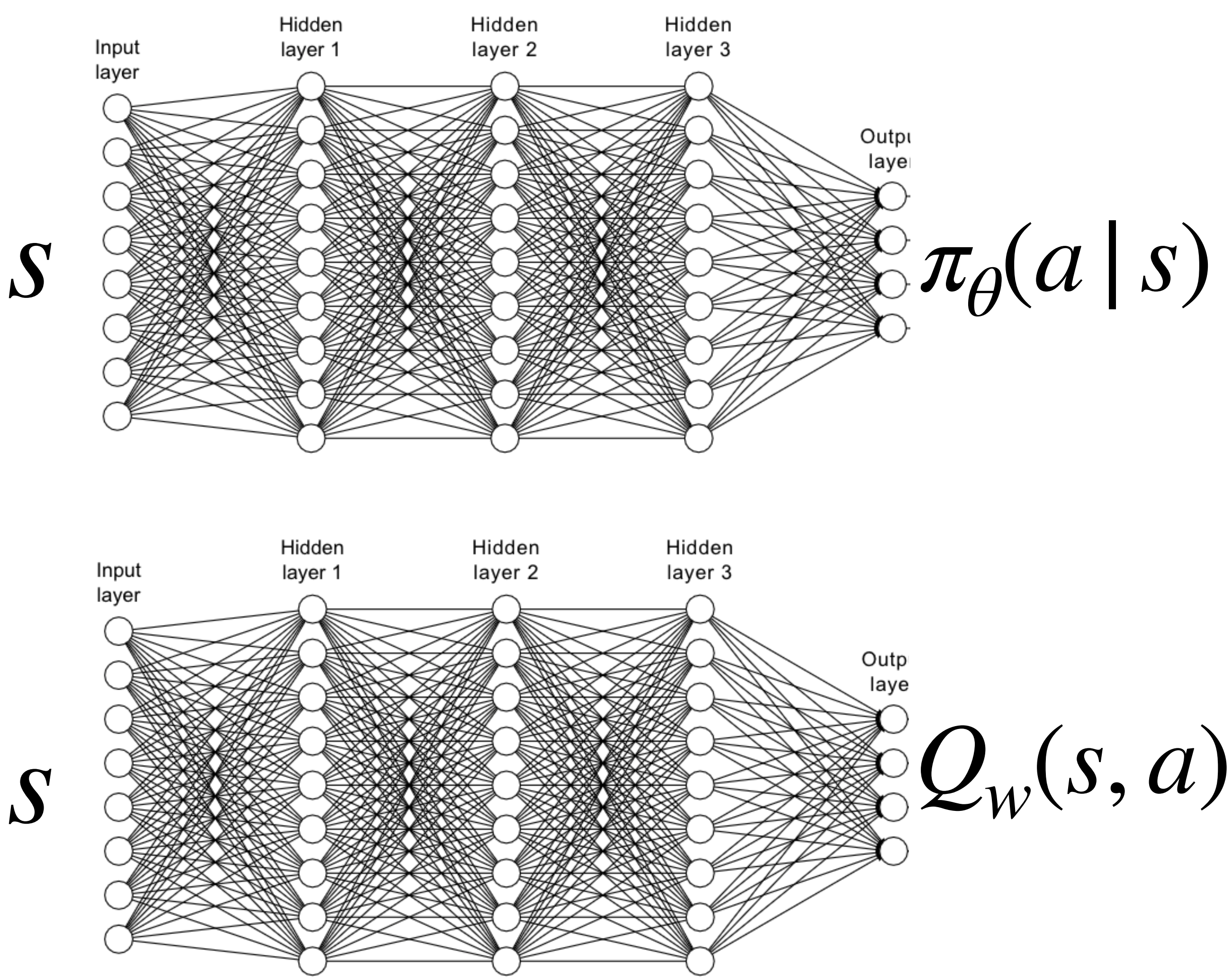
Sample action $a_{t+1} \sim \pi_\theta(s_{t+1}, a_{t+1})$

$\theta \leftarrow \theta + \eta Q_w(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t \mid s_t)$

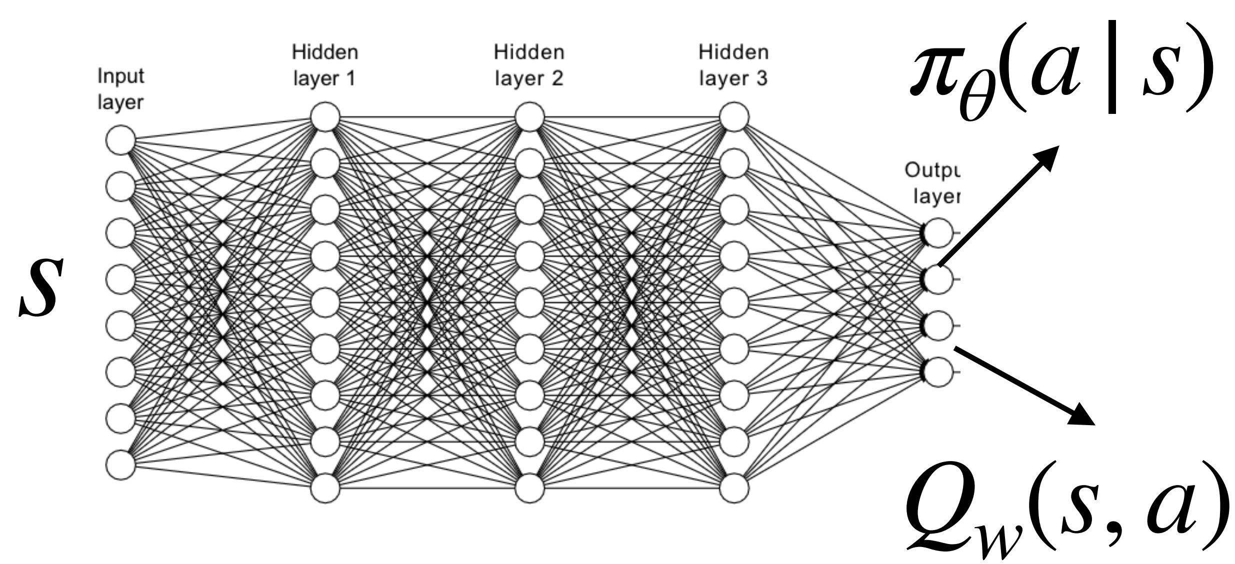
Update w for $Q_w(s, a)$ (possibly using $r_{t+1}, s_{t+1}, a_{t+1}$)

Actor-Critic Architecture

- Two popular choices:



Two separate networks



One shared network

(S3) Reducing Variance Using Advantage Functions

- Question: Can we combine both baseline and critic?

- Define advantage function as

$$A^{\pi_\theta}(s,a) = Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s)$$

- Recall:

  (P3) Q-value and discounted state visitation:

$$\nabla_\theta V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\Big[Q^{\pi_\theta}(s,a)\nabla_\theta \log \pi_\theta(a\,|\,s)\Big]$$

- We have:

$$\nabla_\theta V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\Big[A^{\pi_\theta}(s,a)\nabla_\theta \log \pi_\theta(a\,|\,s)\Big]$$

Policy Gradient With Advantage Functions

- **Policy Gradient With Advantage Function**:

(P4) Advantage:

$$\nabla_\theta V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\left[A^{\pi_\theta}(s,a)\,\nabla_\theta \log \pi_\theta(a\,|\,s)\right]$$

(P5) REINFORCE with advantage:

$$\nabla_\theta V^{\pi_\theta}(\mu) = \mathbb{E}_{\tau\sim P_\mu^{\pi_\theta}}\left[\sum_{t=0}^{\infty}\gamma^t A^{\pi_\theta}(s_t,a_t)\,\nabla_\theta \log \pi_\theta(a_t\,|\,s_t)\right]$$

Optimal Baseline for Variance Reduction?

The Optimal Reward Baseline for Gradient-Based Reinforcement Learning

Lex Weaver
Department of Computer Science
Australian National University
ACT AUSTRALIA 0200
Lex.Weaver@cs.anu.edu.au

Nigel Tao
Department of Computer Science
Australian National University
ACT AUSTRALIA 0200
Nigel.Tao@cs.anu.edu.au

Abstract

There exist a number of reinforcement learning algorithms which learn by climbing the gradient of expected reward. Their long-run convergence has been proved, even in partially observable environments with non-deterministic actions, and without the need for a system model. However, the variance of the gradient estimator has been found to be a significant practical problem. Recent approaches have discounted future rewards, introducing a bias-variance trade-off into the gradient estimate. We incorporate a reward baseline into the learning system, and show that it affects variance without introducing further bias. In particular, as we approach the zero-bias, high-variance parameterization, the optimal (or variance minimizing) constant reward baseline is equal to the long-term average expected reward. Modified policy-gradient algorithms are presented, and a number of experiments demonstrate their improvement over previous work.

the reliance on both a system model and a need to identify a specific recurrent state, and operate in partially observable environments with non-deterministic actions (POMDPs).

However, the variance of the gradient estimator remains a significant practical problem for policy-gradient applications, although discounting is an effective technique. Discounting future rewards introduces a bias-variance trade-off: variance in the gradient estimates can be reduced by heavily discounting future rewards, but the estimates will be biased; the bias can be reduced by not discounting so heavily, but the variance will be higher. Our work complements the discounting technique by introducing a *reward baseline*[1] which is designed to reduce variance, especially as we approach the zero-bias, high-variance discount factor.

The use of a reward baseline has been considered a number of times before, but we are not aware of any analysis of its effect on variance in the context of the recent policy-gradient algorithms. (Sutton, 1984) empirically investigated the inclusion of a reinforcement comparison term in several stochastic learning equations, and argued that it should result in faster learning for unbalanced reinforce-

- One could find an optimal baseline $b^*(s)$ by directly minimizing the covariance of a PG estimator

(A practice problem of HW2)

[UAI 2001]
Available at https://arxiv.org/pdf/1301.2315.pdf

How to Estimate the Action-Value Function?

- A critic = solving the policy evaluation problem
  - How good is a policy π_θ?

- In Lecture 3, we discussed both non-iterative and iterative policy evaluation given the MDP model parameters

- Question: How to do policy evaluation without knowing MDP model parameters?

Next Topic: Model-free prediction!

Model-Free Prediction

= Policy evaluation with **unknown** dynamics & rewards

Monte-Carlo for Policy Evaluation

- Recall: Monte-Carlo policy gradient
  - Use sample return G_t for the estimate of policy gradient

$$\nabla_\theta V^{\pi_\theta}(\mu) \approx \sum_{t=0}^{\infty} \gamma^t G_t \nabla_\theta \log \pi_\theta(a_t \mid s_t)$$

- Question: Can we use the same idea for policy evaluation (i.e., finding $V^\pi(s)$)?

Monte-Carlo for Policy Evaluation (Cont.)

To find the value function V^{π} under a fixed policy π:

- For episodic environments ➡ sample a set of trajectories & calculate average returns

- For continuing environments ➡ sample a set of trajectories (but with proper ***truncation***) and calculate average returns

Features of MC

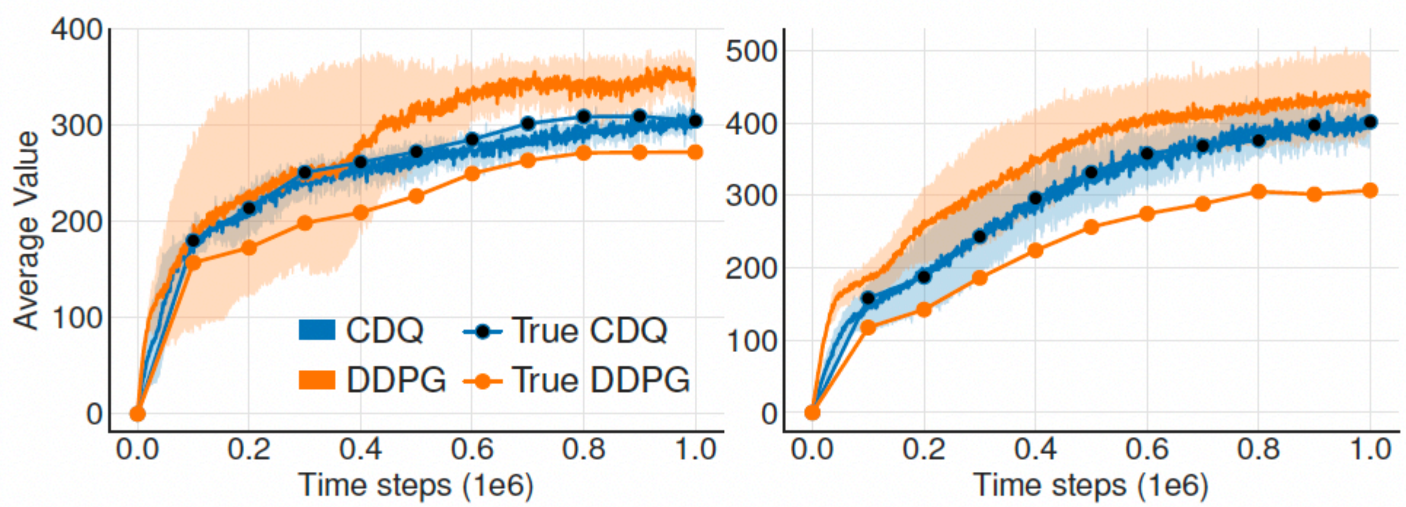
1. MC is model-free
   - MC learns directly from episodes *without* estimating MDP transition probabilities or reward function

2. MC learns from complete episodes

Is MC Policy Evaluation Useful in Practice?

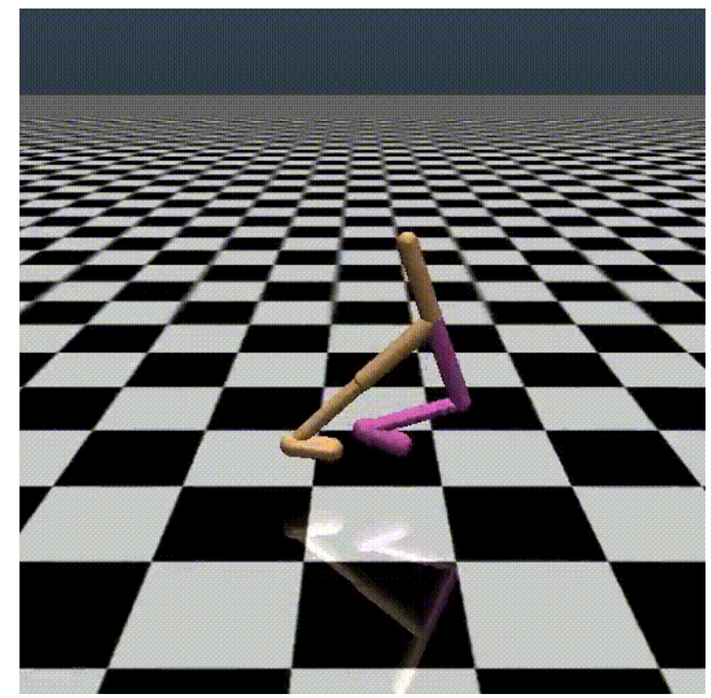
Yes! MC serves as a pseudo-oracle for true $V^{\pi}(s)$ or $Q^{\pi}(s, a)$

Example: Finding the “true value functions” in the TD3 paper



(a) Hopper-v1 (b) Walker2d-v1

Figure 1. Measuring overestimation bias in the value estimates of DDPG and our proposed method, Clipped Double Q-learning (CDQ), on MuJoCo environments over 1 million time steps.



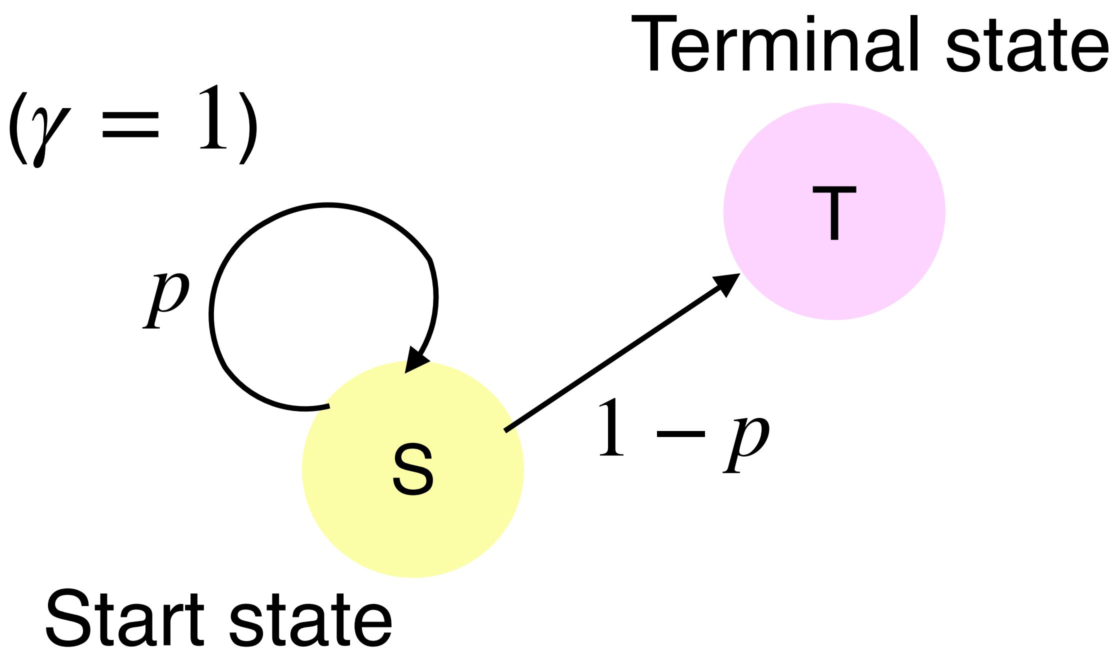
- If the policy is *deterministic*, how many trajectories do we need?
- What if the policy is *stochastic*?

Fujimoto et al., Addressing Function Approximation Error in Actor-Critic Methods, ICML 2018

First-Visit and Every-Visit MC

- A visit to s: an occurrence of a state s in an episode
- First-visit MC: Estimate the value of a state as the average of the returns that have followed the first visit to the state in an episode
- Every-visit MC: Estimate the value of a state as the average of the returns that have followed all visits to the state

Example: 2-State MRP



@Start state: reward = 1

@Terminal state: reward = 0

- Consider a sample trajectory: $S \rightarrow S \rightarrow S \rightarrow S \rightarrow T$
- Question: First-visit MC estimate of $V(S)$ = ? 4
- Question: Every-visit MC estimate of $V(S)$ = ?

$$(4 + 3 + 2 + 1)/4 = 2.5$$

- Question: Which estimate is better?

First-Visit MC Policy Evaluation (Formally)

Initialize $N(s) = 0,\ G(s) = 0\ \forall s \in S$
Loop

- Sample episode $i = s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$
- Define $G_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \cdots \gamma^{T_i - 1} r_{i,T_i}$ as return from time step t onwards in ith episode
- For each state s visited in episode i
  - For **first** time t that state s is visited in episode i
    - Increment counter of total first visits: $N(s) = N(s) + 1$
    - Increment total return $G(s) = G(s) + G_{i,t}$
    - Update estimate $V^\pi(s) = G(s)/N(s)$

Properties:

- V^π estimator is an unbiased estimator of true $\mathbb{E}_\pi[G_t | s_t = s]$
- By law of large numbers, as $N(s) \to \infty$, $V^\pi(s) \to \mathbb{E}_\pi[G_t | s_t = s]$

(Slide Credit: Emma Brunskill)

Every-Visit MC Policy Evaluation (Formally)

Initialize $N(s) = 0,\ G(s) = 0\ \forall s \in S$
Loop

- Sample episode $i = s_{i,1}, a_{i,1}, r_{i,1}, s_{i,2}, a_{i,2}, r_{i,2}, \ldots, s_{i,T_i}$
- Define $G_{i,t} = r_{i,t} + \gamma r_{i,t+1} + \gamma^2 r_{i,t+2} + \cdots \gamma^{T_i-1} r_{i,T_i}$ as return from time step t onwards in ith episode
- For each state s visited in episode i
  - For **every** time t that state s is visited in episode i
    - Increment counter of total first visits: $N(s) = N(s) + 1$
    - Increment total return $G(s) = G(s) + G_{i,t}$
    - Update estimate $V^\pi(s) = G(s)/N(s)$

Properties:

- V^π every-visit MC estimator is a **biased** estimator of V^π
- But consistent estimator and often has better MSE

(Slide Credit: Emma Brunskill)