Steps to understanding Policy-gradient methods

- Policy approximation $\pi(a|s, \boldsymbol{\theta})$
- The average-reward (reward rate) objective $\bar{r}(\boldsymbol{\theta})$
- Stochastic gradient ascent/descent $\Delta\boldsymbol{\theta}_t \approx \alpha \frac{\partial \bar{r}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$
- The policy-gradient theorem and its proof
- Approximating the gradient
- Eligibility functions for a few cases
- A final algorithm

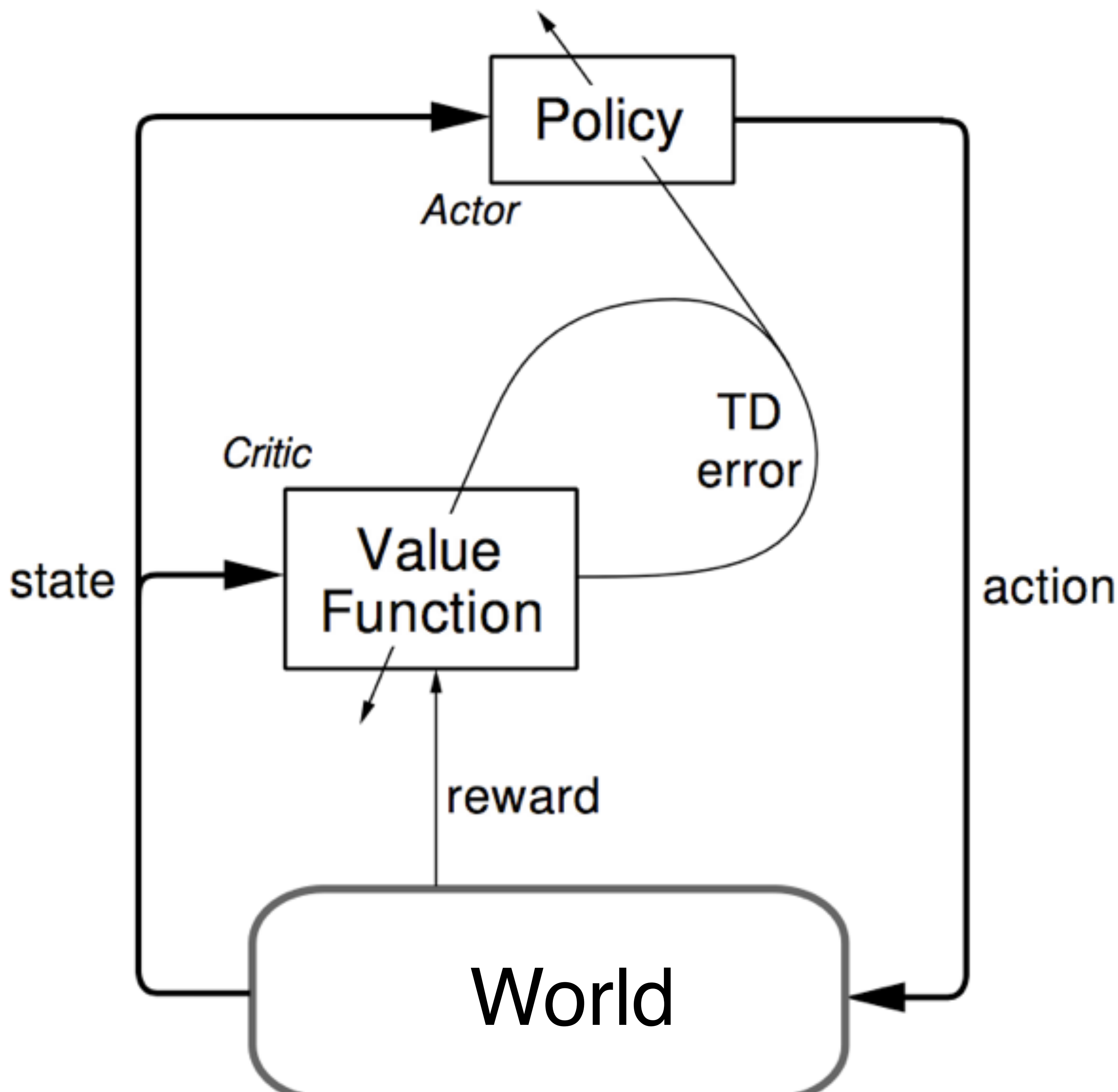
Policy Approximation

- Policy = a function from state to action
  - How does the agent select actions?
  - In such a way that it can be affected by learning?
  - In such a way as to assure exploration?
- Approximation: there are too many states and/or actions to represent all policies
  - To handle large/continuous action spaces

What is learned and stored?

1. *Action-value methods*: learn the value of each action; pick the max (usually)
2. *Policy-gradient methods*: learn the parameters $\mathbf{u}$ of a stochastic *policy*, update by $\nabla_{\mathbf{u}}\text{Performance}$
   - including *actor-critic methods*, which learn *both* value and policy parameters
3. *Dynamic Policy Programming*
4. *Drift-diffusion models* (Psychology)

Actor-critic architecture



Action-value methods

- The *value of an action in a state given a policy* is the expected future reward starting from the state taking that first action, then following the policy thereafter

$$q_\pi(s,a) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} R_t \middle| S_0 = s, A_0 = a\right]$$

- Policy: pick the max most of the time

$$A_t = \arg\max_a \hat{Q}_t(S_t, a)$$

but sometimes pick at random (ε-greedy)

Why approximate policies rather than values?

- In many problems, the policy is simpler to approximate than the value function
- In many problems, the optimal policy is stochastic
  - e.g., bluffing, POMDPs
- To enable smoother change in policies
- To avoid a search on every step (the max)
- To better relate to biology

Gradient-bandit algorithm

- Store action preferences $H_t(a)$ rather than action-value estimates $Q_t(a)$
- Instead of ε-greedy, pick actions by an exponential soft-max:

$$\Pr\{A_t = a\} \doteq \frac{e^{H_t(a)}}{\sum_{b=1}^{k} e^{H_t(b)}} \doteq \pi_t(a)$$

- Also store the sample average of rewards as $\bar{R}_t$
- Then update:

$$H_{t+1}(a) = H_t(a) + \alpha\big(R_t - \bar{R}_t\big)\big(\mathbf{1}_{a=A_t} - \pi_t(a)\big)$$

1 or 0, depending on whether the predicate (subscript) is true

Gradient-bandit algorithms on the 10-armed testbed

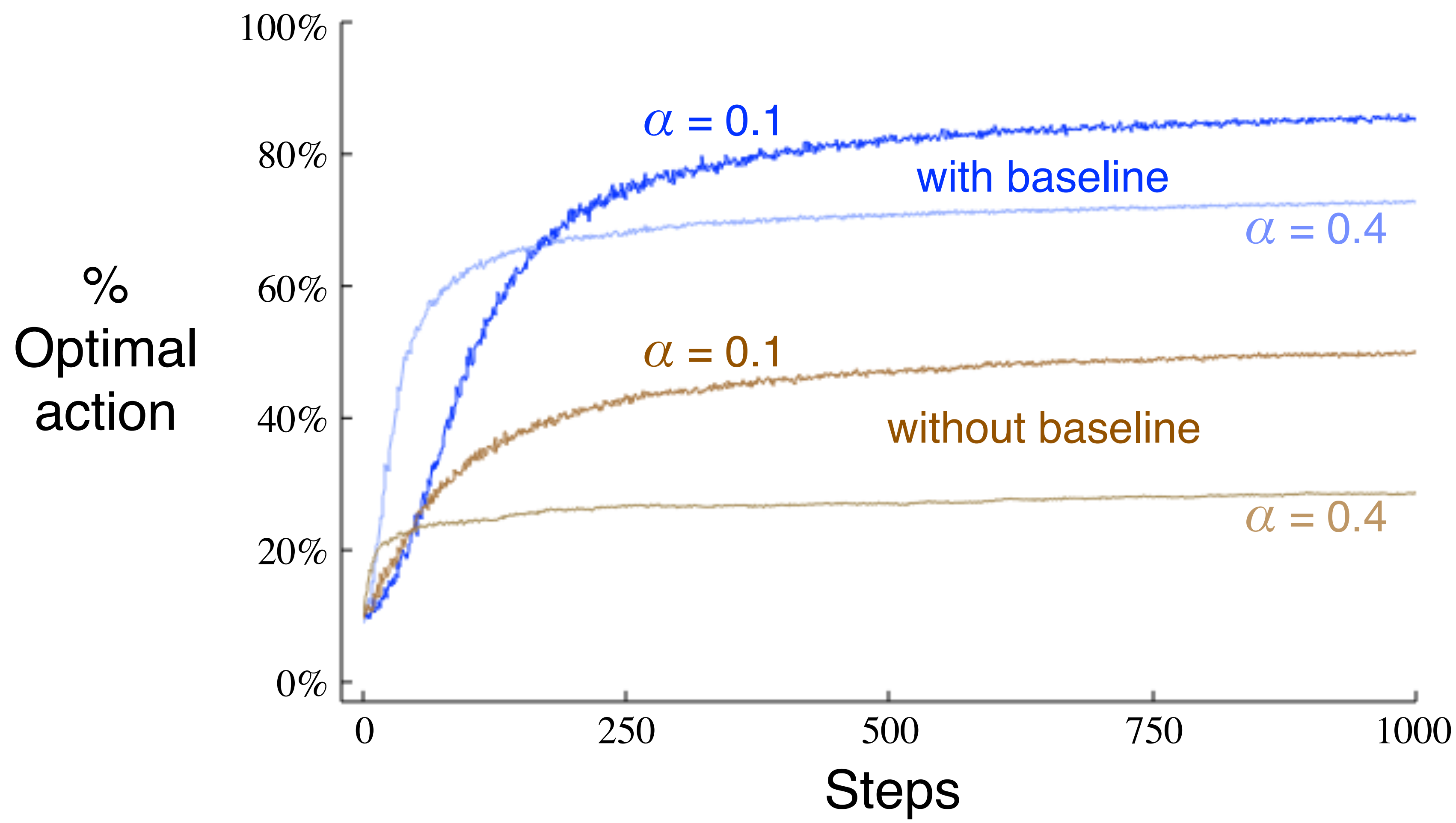


Figure 2.6: Average performance of the gradient-bandit algorithm with and without a reward baseline on the 10-armed testbed when the $q_*(a)$ are chosen to be near $+4$ rather than near zero.

$$\frac{\partial}{\partial x}\left[\frac{f(x)}{g(x)}\right] = \frac{\frac{\partial f(x)}{\partial x} g(x) - f(x)\frac{\partial g(x)}{\partial x}}{g(x)^2}$$

$$\begin{aligned}
\frac{\partial \pi_t(b)}{\partial H_t(a)} &= \frac{\partial}{\partial H_t(a)} \pi_t(b) \\
&= \frac{\partial}{\partial H_t(a)} \left[\frac{e^{H_t(b)}}{\sum_{c=1}^k e^{H_t(c)}}\right] \\
&= \frac{\frac{\partial e^{H_t(b)}}{\partial H_t(a)} \sum_{c=1}^k e^{H_t(c)} - e^{H_t(b)} \frac{\partial \sum_{c=1}^k e^{H_t(c)}}{\partial H_t(a)}}{\left(\sum_{c=1}^k e^{H_t(c)}\right)^2} && \text{(by the quotient rule)} \\
&= \frac{\mathbf{1}_{a=b} e^{H_t(a)} \sum_{c=1}^k e^{H_t(c)} - e^{H_t(b)} e^{H_t(a)}}{\left(\sum_{c=1}^k e^{H_t(c)}\right)^2} && \left(\text{because } \frac{\partial e^x}{\partial x} = e^x\right) \\
&= \frac{\mathbf{1}_{a=b} e^{H_t(b)}}{\sum_{c=1}^k e^{H_t(c)}} - \frac{e^{H_t(b)} e^{H_t(a)}}{\left(\sum_{c=1}^k e^{H_t(c)}\right)^2} \\
&= \mathbf{1}_{a=b} \pi_t(b) - \pi_t(b)\pi_t(a) \\
&= \pi_t(b)\big(\mathbf{1}_{a=b} - \pi_t(a)\big). && \text{Q.E.D.}
\end{aligned}$$

Steps to understanding Policy-gradient methods

- Policy approximation $\pi(a|s,\boldsymbol{\theta})$
- The average-reward (reward rate) objective $\bar{r}(\boldsymbol{\theta})$
- Stochastic gradient ascent/descent $\Delta\boldsymbol{\theta}_t \approx \alpha \frac{\partial \bar{r}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$
- The policy-gradient theorem and its proof
- Approximating the gradient
- Eligibility functions for a few cases
- A complete algorithm

eg, linear-exponential policies (discrete actions)

- The "preference" for action a in state s is linear in θ and a state-action feature vector $\phi(s,a)$

- The probability of action a in state s is exponential in its preference

$$\pi(a|s,\boldsymbol{\theta}) \doteq \frac{\exp(\boldsymbol{\theta}^\top \boldsymbol{\phi}(s,a))}{\sum_b \exp(\boldsymbol{\theta}^\top \boldsymbol{\phi}(s,b))}$$

- Corresponding eligibility function:

$$\frac{\nabla\pi(a|s,\boldsymbol{\theta})}{\pi(a|s,\boldsymbol{\theta})} = \boldsymbol{\phi}(s,a) - \sum_b \pi(b|s,\boldsymbol{\theta})\boldsymbol{\phi}(s,b)$$

Policy-gradient setup

parameterized policies	$\pi(a\|s, \boldsymbol{\theta}) \doteq \Pr\{A_t = a \mid S_t = s\}$
average-reward objective	$r(\pi) \doteq \lim_{n\to\infty} \frac{1}{n} \sum_{t=1}^{n} \mathbb{E}_\pi[R_t] = \sum_s d_\pi(s) \sum_a \pi(a\|s) \sum_{s',r} p(s', r\|s, a) r$
steady-state distribution	$d_\pi \doteq \lim_{t\to\infty} \Pr\{S_t = s\}$
differential state-value fn	$\tilde{v}_\pi(s) \doteq \sum_{k=1}^{\infty} \mathbb{E}_\pi[R_{t+k} - r(\pi) \mid S_t = s]$
differential action-value fn	$\tilde{q}_\pi(s, a) \doteq \sum_{k=1}^{\infty} \mathbb{E}_\pi[R_{t+k} - r(\pi) \mid S_t = s, A_t = a]$
stochastic gradient ascent	$\Delta\boldsymbol{\theta}_t \approx \alpha \frac{\partial r(\pi)}{\partial \boldsymbol{\theta}} \doteq \alpha \nabla r(\pi)$

stochastic gradient ascent $$\Delta\boldsymbol{\theta}_t \approx \alpha\frac{\partial r(\pi)}{\partial\boldsymbol{\theta}} \doteq \alpha\nabla r(\pi)$$

policy-gradient theorem
$$\begin{aligned}
\nabla r(\pi) &= \sum_s d_\pi(s)\sum_a \tilde{q}_\pi(s,a)\nabla\pi(a|s,\boldsymbol{\theta}) \\
&= \mathbb{E}\left[\Big(\tilde{q}_\pi(S_t,A_t) - v(S_t)\Big)\frac{\nabla\pi(A_t|S_t,\boldsymbol{\theta})}{\pi(A_t|S_t)} \;\middle|\; S_t \sim d_\pi, A_t \sim \pi(\cdot|S_t,\boldsymbol{\theta})\right] \\
&= \mathbb{E}\left[\Big(\tilde{G}_t^\lambda - \hat{v}(S_t,\mathbf{w})\Big)\frac{\nabla\pi(A_t|S_t,\boldsymbol{\theta})}{\pi(A_t|S_t)} \;\middle|\; S_t \sim d_\pi, A_{t:\infty} \sim \pi\right] \\
&\approx \Big(\tilde{G}_t^\lambda - \hat{v}(S_t,\mathbf{w})\Big)\frac{\nabla\pi(A_t|S_t,\boldsymbol{\theta})}{\pi(A_t|S_t)} \qquad \text{(by sampling under } \pi\text{)}
\end{aligned}$$

$$\boldsymbol{\theta}_{t+1} \doteq \boldsymbol{\theta}_t + \alpha\Big(\tilde{G}_t^\lambda - \hat{v}(S_t,\mathbf{w})\Big)\frac{\nabla\pi(A_t|S_t,\boldsymbol{\theta})}{\pi(A_t|S_t)}$$

e.g., in the one-step linear case:

$$= \boldsymbol{\theta}_t + \alpha\Big(R_{t+1} - \bar{R}_t + \mathbf{w}_t^\top\boldsymbol{\phi}_{t+1} - \mathbf{w}_t^\top\boldsymbol{\phi}_t)\Big)\frac{\nabla\pi(A_t|S_t,\boldsymbol{\theta})}{\pi(A_t|S_t)}$$

Deriving the policy-gradient theorem: $\nabla r(\pi) = \sum_s d_\pi(s) \sum_a \tilde{q}_\pi(s,a) \nabla \pi(a|s,\boldsymbol{\theta})$:

$$\begin{aligned}
\nabla \tilde{v}_\pi(s) &= \nabla \sum_a \pi(a|s,\boldsymbol{\theta}) \tilde{q}_\pi(s,a) \\
&= \sum_a \Big[\nabla \pi(a|s,\boldsymbol{\theta}) \tilde{q}_\pi(s,a) + \pi(a|s,\boldsymbol{\theta}) \nabla \tilde{q}_\pi(s,a) \Big] \\
&= \sum_a \Big[\nabla \pi(a|s,\boldsymbol{\theta}) \tilde{q}_\pi(s,a) + \pi(a|s,\boldsymbol{\theta}) \nabla \sum_{s',r} p(s',r|s,a) \big[r - r(\pi) + \tilde{v}_\pi(s') \big] \Big] \\
&= \sum_a \Big[\nabla \pi(a|s,\boldsymbol{\theta}) \tilde{q}_\pi(s,a) + \pi(a|s,\boldsymbol{\theta}) \Big[-\nabla r(\pi) + \sum_{s',r} p(s'|s,a) \nabla \tilde{v}_\pi(s') \Big] \Big]
\end{aligned}$$

$$\therefore \nabla r(\pi) = \sum_a \Big[\nabla \pi(a|s,\boldsymbol{\theta}) \tilde{q}_\pi(s,a) + \pi(a|s,\boldsymbol{\theta}) \sum_{s'} p(s'|s,a) \nabla \tilde{v}_\pi(s') \Big] - \nabla \tilde{v}_\pi(s)$$

$$\therefore \nabla r(\pi) = \sum_a \left[\nabla \pi(a|s, \boldsymbol{\theta}) \tilde{q}_\pi(s, a) + \pi(a|s, \boldsymbol{\theta}) \sum_{s'} p(s'|s, a) \nabla \tilde{v}_\pi(s') \right] - \nabla \tilde{v}_\pi(s)$$

$$\begin{aligned}
\therefore \sum_s d_\pi(s) \nabla r(\pi) &= \sum_s d_\pi(s) \sum_a \nabla \pi(a|s, \boldsymbol{\theta}) \tilde{q}_\pi(s, a) \\
&\quad + \sum_s d_\pi(s) \sum_a \pi(a|s, \boldsymbol{\theta}) \sum_{s'} p(s'|s, a) \nabla \tilde{v}_\pi(s') - \sum_s d_\pi(s) \nabla \tilde{v}_\pi(s) \\
&= \sum_s d_\pi(s) \sum_a \nabla \pi(a|s, \boldsymbol{\theta}) \tilde{q}_\pi(s, a) \\
&\quad + \sum_{s'} \sum_s d_\pi(s) \sum_a \pi(a|s, \boldsymbol{\theta}) p(s'|s, a)\, \nabla \tilde{v}_\pi(s') - \sum_s d_\pi(s) \nabla \tilde{v}_\pi(s)
\end{aligned}$$

$$\nabla r(\pi) = \sum_s d_\pi(s) \sum_a \nabla \pi(a|s, \boldsymbol{\theta}) \tilde{q}_\pi(s, a)$$

Complete PG algorithm

Initialize parameters of policy $\boldsymbol{\theta} \in \mathbb{R}^n$, and state-value function $\mathbf{w} \in \mathbb{R}^m$

Initialize eligibility traces $\mathbf{e}^{\boldsymbol{\theta}} \in \mathbb{R}^n$ and $\mathbf{e}^{\mathbf{w}} \in \mathbb{R}^m$ to $\mathbf{0}$

Initialize $\bar{R} = 0$

On each step, in state S:

Choose A according to $\pi(\cdot|S, \boldsymbol{\theta})$

Take action A, observe S', R

$\delta \leftarrow R - \bar{R} + \hat{v}(S', \mathbf{w}) - \hat{v}(S, \mathbf{w})$ — form TD error from critic

$\bar{R} \leftarrow \bar{R} + \alpha^{\boldsymbol{\theta}} \delta$ — update average reward estimate

$\mathbf{e}^{\mathbf{w}} \leftarrow \lambda \mathbf{e}^{\mathbf{w}} + \nabla_{\mathbf{w}} \hat{v}(S, \mathbf{w})$ — update eligibility trace for critic

$\mathbf{w} \leftarrow \mathbf{w} + \alpha^{\mathbf{w}} \delta \mathbf{e}^{\mathbf{w}}$ — update critic parameters

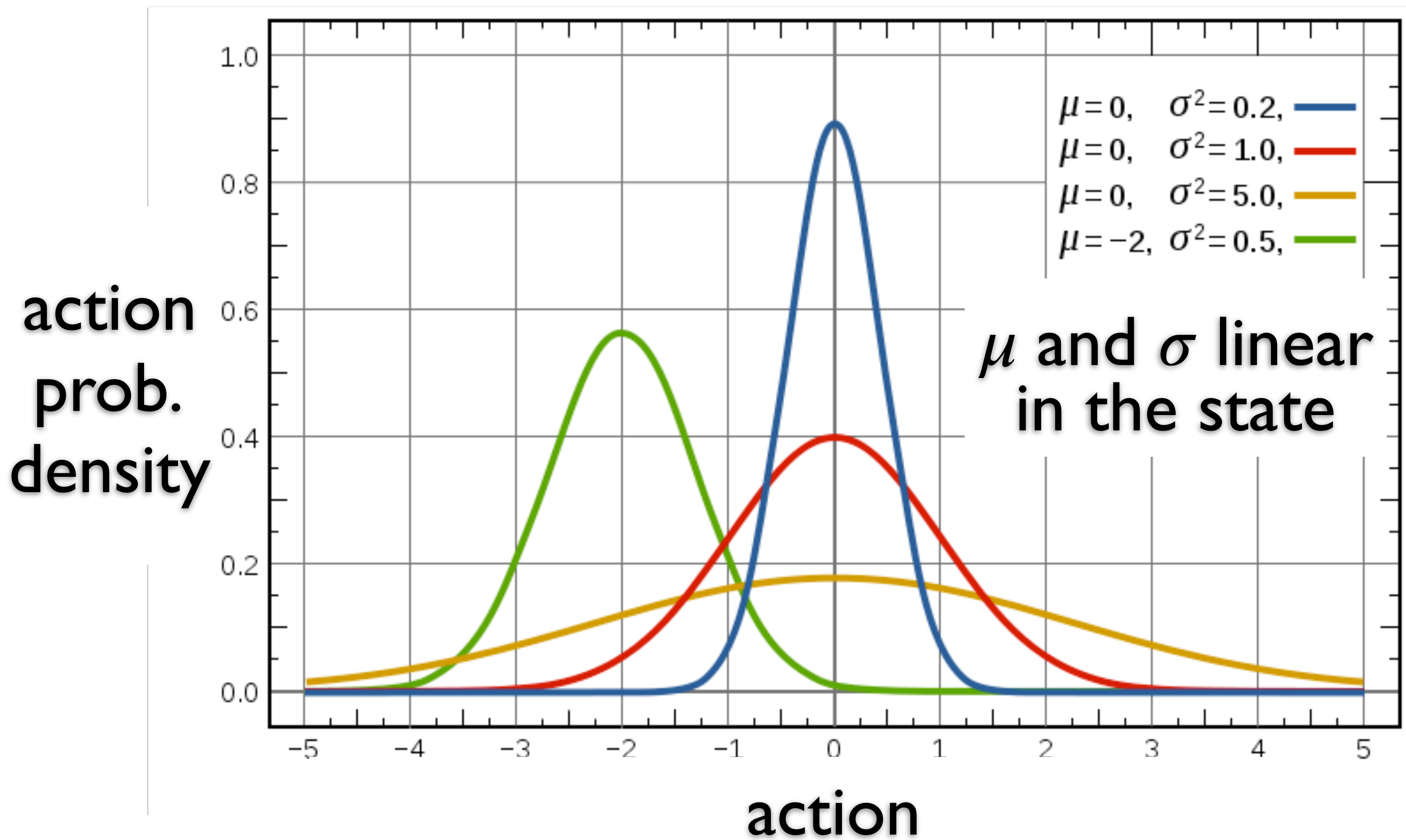
$\mathbf{e}^{\boldsymbol{\theta}} \leftarrow \lambda \mathbf{e}^{\boldsymbol{\theta}} + \frac{\nabla \pi(A|S,\boldsymbol{\theta})}{\pi(A|S,\boldsymbol{\theta})}$ — update eligibility trace for actor

$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha^{\boldsymbol{\theta}} \delta \mathbf{e}^{\boldsymbol{\theta}}$ — update actor parameters

The generality of the policy-gradient strategy

- Can be applied whenever we can compute the effect of parameter changes on the action probabilities, $\nabla \pi(A_t|S_t, \boldsymbol{\theta})$
- E.g., has been applied to spiking neuron models
- There are many possibilities other than linear-exponential and linear-gaussian
  - e.g., mixture of random, argmax, and fixed-width gaussian; learn the mixing weights, drift/diffusion models

eg, linear-gaussian policies (continuous actions)



eg, linear-gaussian policies (continuous actions)

- The mean and std. dev. for the action taken in state s are linear and linear-exponential in

$$\boldsymbol{\theta} \doteq (\boldsymbol{\theta}_\mu^\top; \boldsymbol{\theta}_\sigma^\top)^\top \qquad \mu(s) \doteq \boldsymbol{\theta}_\mu^\top \boldsymbol{\phi}(s) \qquad \sigma(s) \doteq \exp(\boldsymbol{\theta}_\sigma^\top \boldsymbol{\phi}(s))$$

- The probability density function for the action taken in state s is gaussian

$$\pi(a|s,\boldsymbol{\theta}) \doteq \frac{1}{\sigma(s)\sqrt{2\pi}} \exp\left(-\frac{(a-\mu(s))^2}{2\sigma(s)^2}\right)$$

Gaussian eligibility functions

$$\frac{\nabla_{\boldsymbol{\theta}_\mu} \pi(a|s, \boldsymbol{\theta})}{\pi(a|s, \boldsymbol{\theta})} = \frac{1}{\sigma(s)^2}(a - \mu(s))\boldsymbol{\phi}_\mu(s)$$

$$\frac{\nabla_{\boldsymbol{\theta}_\sigma} \pi(a|s, \boldsymbol{\theta})}{\pi(a|s, \boldsymbol{\theta})} = \left(\frac{(a - \mu(s))^2}{\sigma(s)^2} - 1\right)\boldsymbol{\phi}_\sigma(s)$$

The generality of the policy-gradient strategy (2)

- Can be applied whenever we can compute the effect of parameter changes
on the action probabilities, $\nabla\pi(A_t|S_t, \boldsymbol{\theta})$
- Can we apply PG when outcomes are viewed as action?
  - e.g., the action of "turning on the light"
or the action of "going to the bank"
  - is this an alternate strategy for temporal abstraction?
- We would need to learn—not compute—the gradient of these states w.r.t. the policy parameter

Have we eliminated action?

- If any state can be an action, then what is still special about actions?
- The parameters/weights are what we can really, directly control
- We have always, in effect, "sensed" our actions (even in the ε-greedy case)
- Perhaps actions are just sensory signals that we can usually control easily
- Perhaps there is no longer any need for a special concept of action in the RL framework