Bandit Problems

Robert Platt
Northeastern University

(some slides/material borrowed from Rich Sutton)

A 1-armed Bandit

You walk into a casino…

A 1-armed Bandit

You walk into a casino…

… and there it is – your Nemisis!

A 1-armed Bandit

You walk into a casino…

… and there it is – your Nemisis!

You walk up to the machine and start pulling the lever

In this case:

– there’s just one possible lever to pull

– you will eventually be able to estimate the probability of a payout

A 1-armed Bandit

You walk into a casino…

… and there it is – your Nemisis!

You walk up to the machine and start pulling the lever

In this case:

– there’s just one possible lever to pull

– you will eventually be able to estimate the probability of a payout

For example:

$$\text{expected payout per pull} = 1.0 \times -0.01 + 0.09 \times 0.02 + 0.01 \times 10.00 = 0.09$$

A 1-armed Bandit

You walk into a casino…

… and there it is – your Nemisis!

You walk up to the machin

r

– there one possible lever to pull

– you will ev tually be able to estimate
the probabi of a payout

For example:

One pull costs
one cent

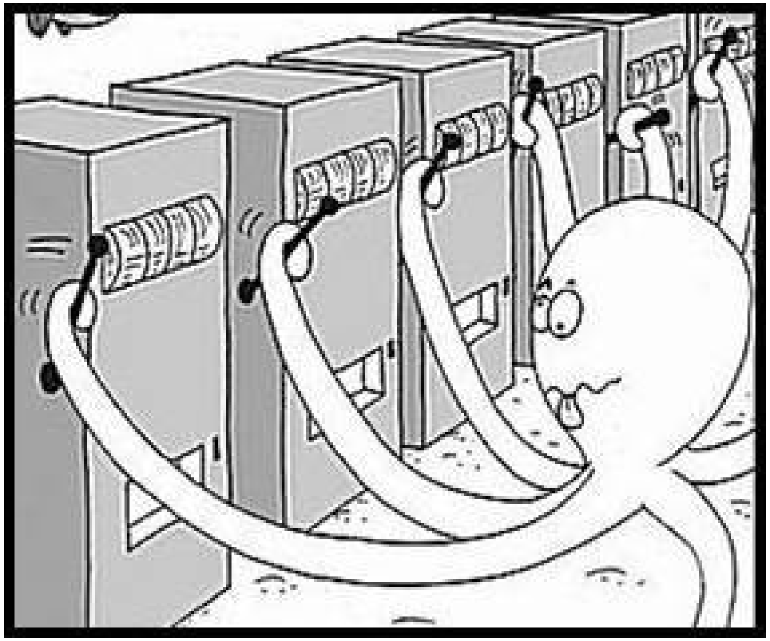
Pays out 2 cents
9% of the time

Pays out $10
1% of time

$$\text{expected payout per pull} = 1.0 \times -0.01 + 0.09 \times 0.02 + 0.01 \times 10.00 = 0.09$$

The k-armed bandit problem

Same as 1-armed bandit except that there are now k-levers instead of just one.



– at each time step, t=1, 2, 3, …, you choose one action from a set of k possible actions

– you receive a real valued reward after taking the action

– reward depends only on action taken; it is identically, independently distributed (iid)

– the expected reward for any action is unknown; distribution of rewards is unknown

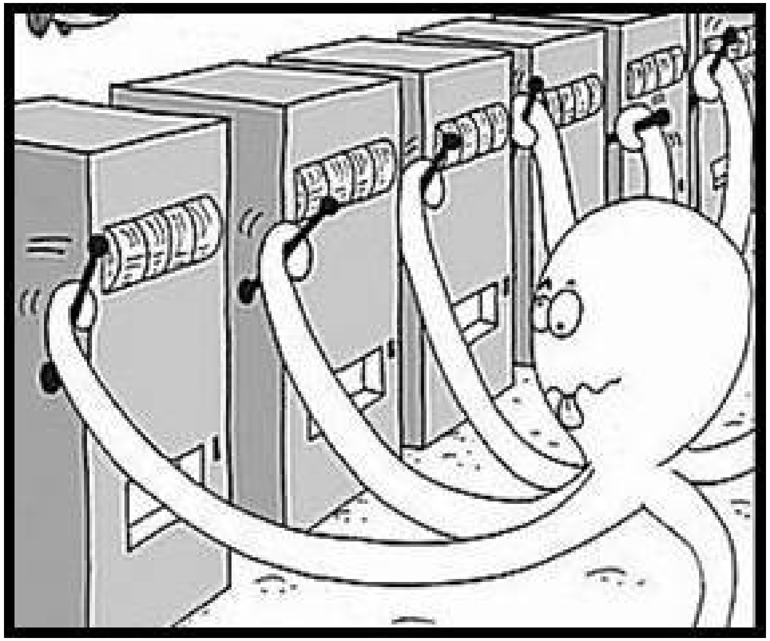
The k-armed bandit problem

Same as 1-armed bandit except that there are now k-levers instead of just one.

– at each time step, t=1, 2, 3, …, you choose one action from a set of k possible actions

– you receive a real valued reward after taking the action

– reward depends only on action taken; it is identically, independently distributed (iid)



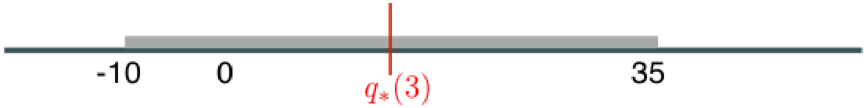
– the expected reward for any action is unknown; distribution of rewards is unknown

– Goal: maximize total reward. You must explore different actions and eventually maximize reward by selecting the action with the highest estimated expected reward.

Think-pair-share question

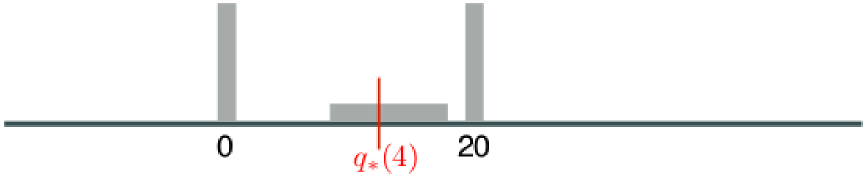
Example of a 4-armed bandit:

- Action 1 — Reward is always 8
  - value of action 1 is $q_*(1) =$
- Action 2 — 88% chance of 0, 12% chance of 100!
  - value of action 2 is $q_*(2) = .88 \times 0 + .12 \times 100 =$
- Action 3 — Randomly between -10 and 35, equiprobable



$q_*(3) =$

- Action 4 — a third 0, a third 20, and a third from {8,9,…, 18}



$q_*(4) =$

So, how should you act?

Q values

Define the Q-function to be: $Q^*(a) = \mathbb{E}[R_t | A_t = a]$

Q values

Define the Q-function to be: $Q^*(a) = \mathbb{E}[R_t | A_t = a]$

Suppose agent gets a lot of experience executing different actions.
How estimate q-function for a given action?

Q values

Define the Q-function to be: $Q^*(a) = \mathbb{E}[R_t | A_t = a]$

Suppose agent gets a lot of experience executing different actions.
How estimate q-function for a given action?

$$Q_t(a) \doteq \frac{\text{sum of rewards when } a \text{ taken prior to } t}{\text{number of times } a \text{ taken prior to } t} = \frac{\sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i=a}}{\sum_{i=1}^{t-1} \mathbb{1}_{A_i=a}}$$

Q values

Define the Q-function to be: $Q^*(a) = \mathbb{E}[R_t | A_t = a]$

Suppose agent gets a lot of experience executing different actions.
How estimate q-function for a given action?

$$Q_t(a) \doteq \frac{\text{sum of rewards when } a \text{ taken prior to } t}{\text{number of times } a \text{ taken prior to } t} = \frac{\sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i=a}}{\sum_{i=1}^{t-1} \mathbb{1}_{A_i=a}}$$

Call this the “q-value”

Q values

Define the Q-function to be: $Q^*(a) = \mathbb{E}[R_t | A_t = a]$

Suppose agent gets a lot of experience executing different actions.
How estimate q-function for a given action?

$$Q_t(a) \doteq \frac{\text{sum of rewards when } a \text{ taken prior to } t}{\text{number of times } a \text{ taken prior to } t} = \frac{\sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i=a}}{\sum_{i=1}^{t-1} \mathbb{1}_{A_i=a}}$$

Call this the “q-value”

In the limit, this estimate converges to the true value: $\lim_{N_t(a) \to \infty} Q_t(a) = Q^*(a)$

Exploration vs Exploitation

Given an estimate of $Q_t(a)$, how do we decide how to act?

Two possibilities:

1. Greedy action selection: $A_t = \arg\max_a Q_t(a)$

2. Do something else

Exploration vs Exploitation

Given an estimate of $Q_t(a)$, act?

Exploitation

<u>Two possibilities</u>:

1. Greedy action selection: $A_t = \arg\max_a Q_t(a)$

2. Do something else

Exploration

Exploration vs Exploitation

Given an estimate of $Q_t(a)$, act?

Exploitation

Two possibilities:

1. Greedy action selection: $A_t = \arg\max_a Q_t(a)$

2. Do something else

Exploration

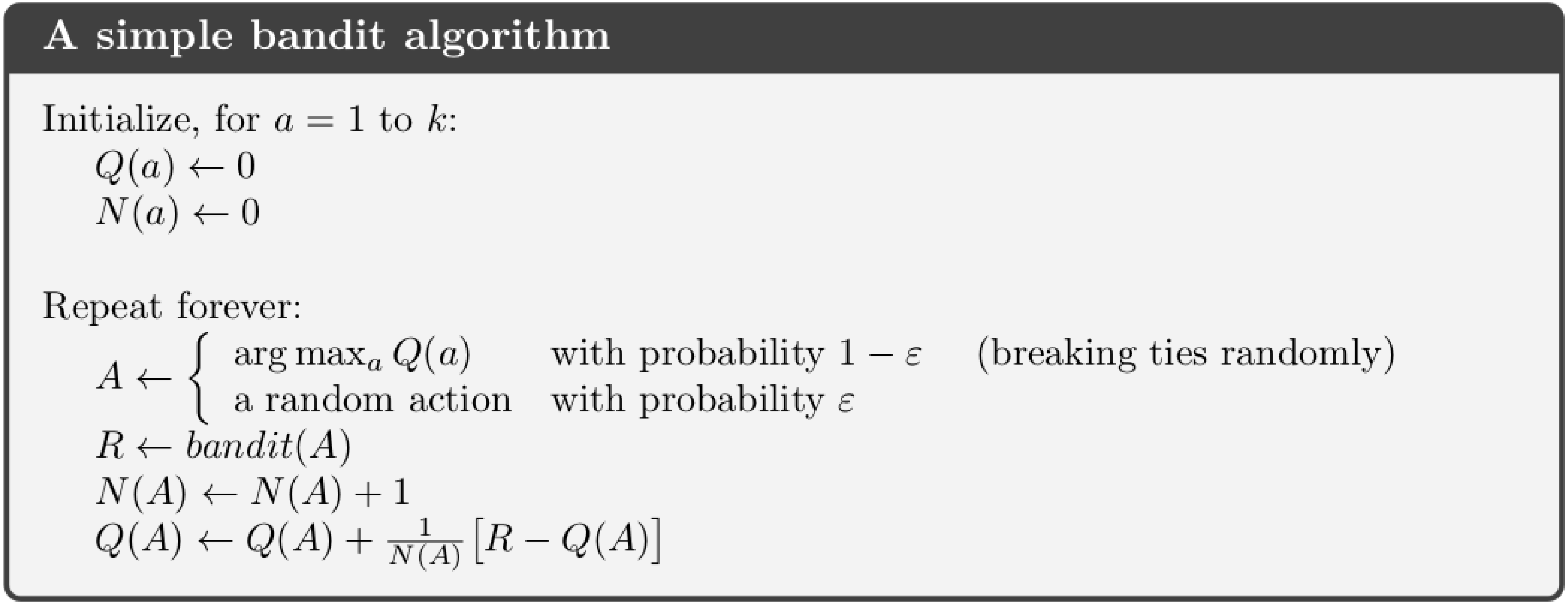
– if we don’t explore, then our q-value estimates may be wrong!

– if we don’t exploit, then we never utilize our knowledge!

E-greedy action selection

- In greedy action selection, you always exploit
- In ε-greedy, you are usually greedy, but with probability ε you instead pick an action at random (possibly the greedy action again)
- This is perhaps the simplest way to balance exploration and exploitation

E-greedy k-armed bandit algorithm

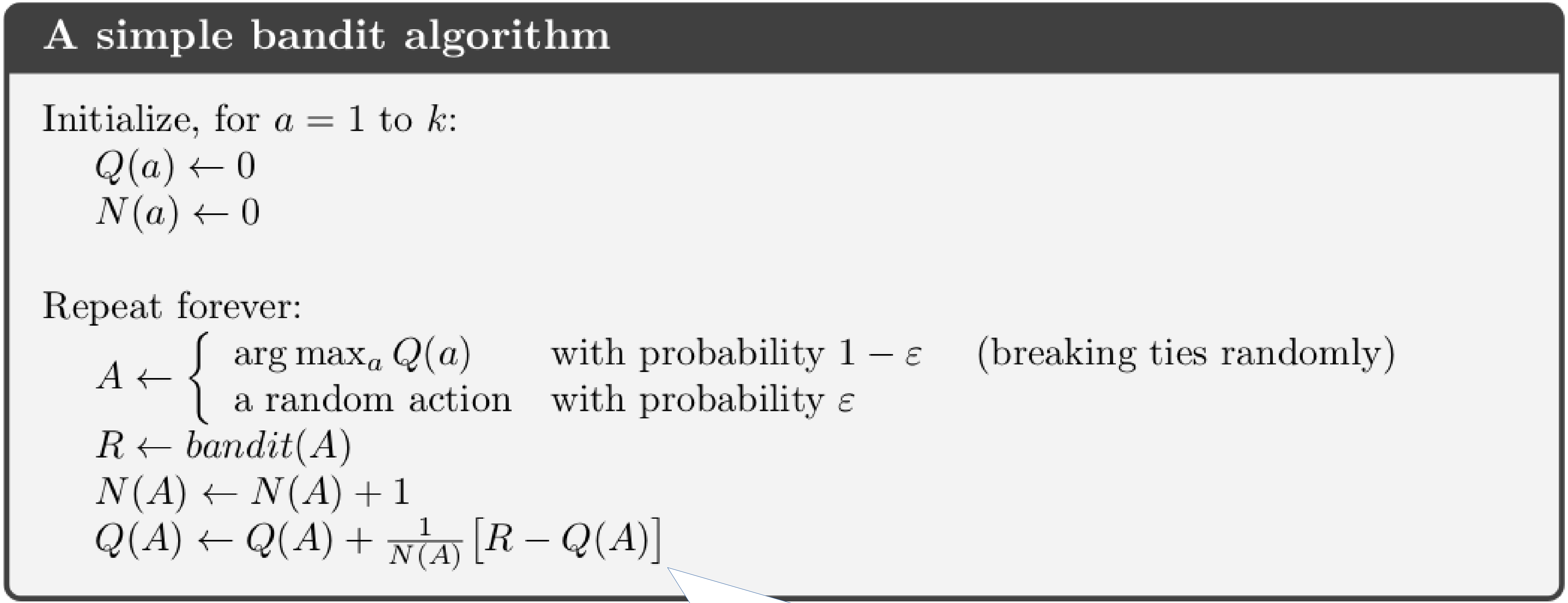


A simple bandit algorithm

```
Initialize, for $a = 1$ to $k$:
    $Q(a) \leftarrow 0$
    $N(a) \leftarrow 0$

Repeat forever:
    $A \leftarrow \begin{cases} \arg\max_a Q(a) & \text{with probability } 1 - \varepsilon \quad \text{(breaking ties randomly)} \\ \text{a random action} & \text{with probability } \varepsilon \end{cases}$
    $R \leftarrow bandit(A)$
    $N(A) \leftarrow N(A) + 1$
    $Q(A) \leftarrow Q(A) + \frac{1}{N(A)}\big[R - Q(A)\big]$
```

E-greedy k-armed bandit algorithm



A simple bandit algorithm

```
Initialize, for $a = 1$ to $k$:
    $Q(a) \leftarrow 0$
    $N(a) \leftarrow 0$

Repeat forever:
    $A \leftarrow \begin{cases} \arg\max_a Q(a) & \text{with probability } 1-\varepsilon \quad \text{(breaking ties randomly)} \\ \text{a random action} & \text{with probability } \varepsilon \end{cases}$
    $R \leftarrow bandit(A)$
    $N(A) \leftarrow N(A) + 1$
    $Q(A) \leftarrow Q(A) + \frac{1}{N(A)}\left[R - Q(A)\right]$
```

Incremental estimate of $Q(A) = \dfrac{1}{n}\sum_{i=1}^{n} R_i$

Incremental q-value estimate

- To simplify notation, let us focus on one action
    - We consider only its rewards, and its estimate after $n{-}1$ rewards:

$$Q_n \doteq \frac{R_1 + R_2 + \cdots + R_{n-1}}{n-1}$$

- How can we do this incrementally (without storing all the rewards)?
- Could store a running sum and count (and divide), or equivalently:

$$Q_{n+1} = Q_n + \frac{1}{n}\Big[R_n - Q_n\Big]$$

- This is a standard form for learning/update rules:

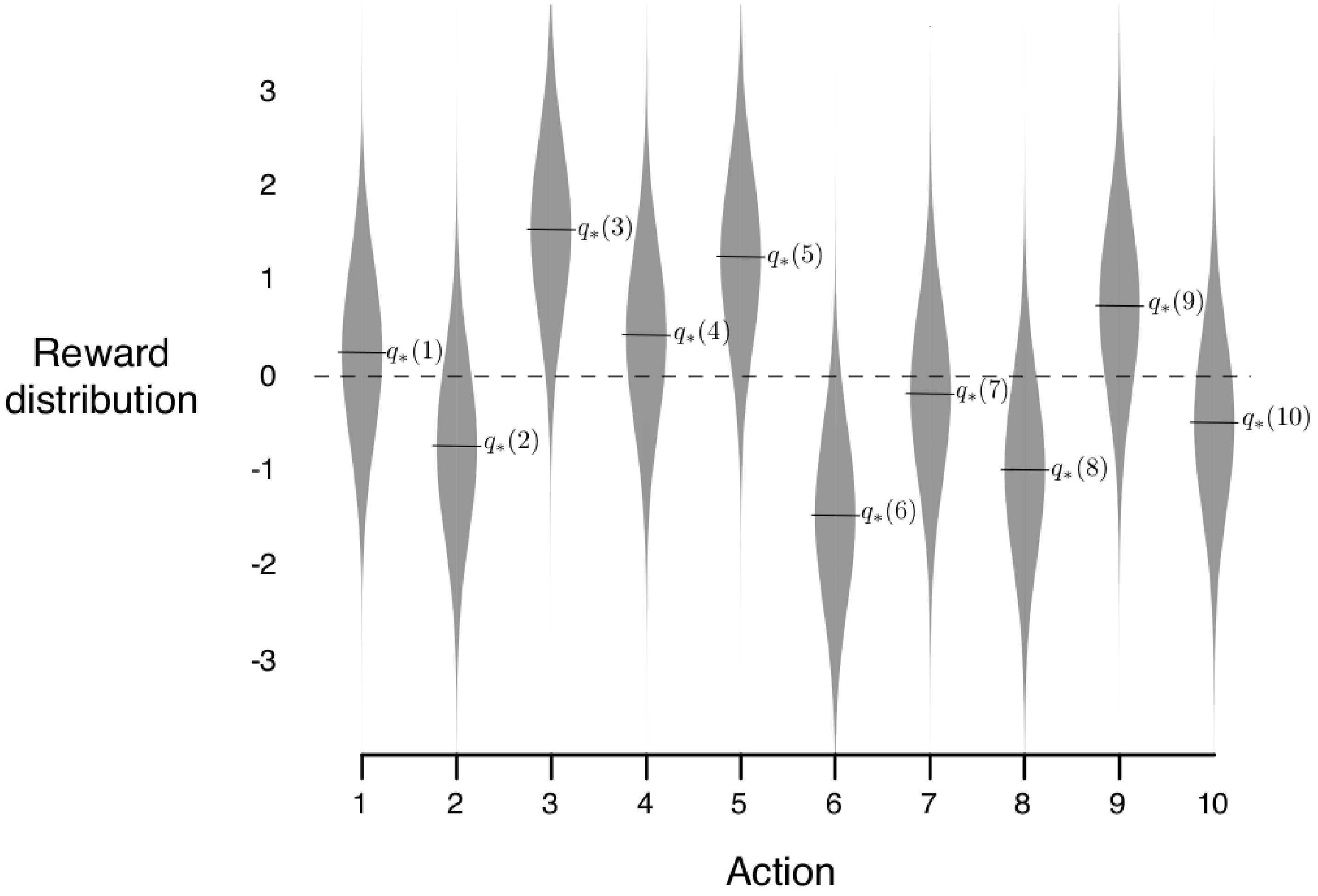
$$NewEstimate \leftarrow OldEstimate \;+\; StepSize \Big[Target - OldEstimate\Big]$$

Incremental q-value estimate

$$\begin{aligned}
Q_{n+1} &= \frac{1}{n}\sum_{i=1}^{n} R_i \\
&= \frac{1}{n}\left(R_n + \sum_{i=1}^{n-1} R_i\right) \\
&= \frac{1}{n}\left(R_n + (n-1)\frac{1}{n-1}\sum_{i=1}^{n-1} R_i\right) \\
&= \frac{1}{n}\Big(R_n + (n-1)Q_n\Big) \\
&= \frac{1}{n}\Big(R_n + nQ_n - Q_n\Big) \\
&= Q_n + \frac{1}{n}\Big[R_n - Q_n\Big],
\end{aligned}$$

SB, eq 2.3

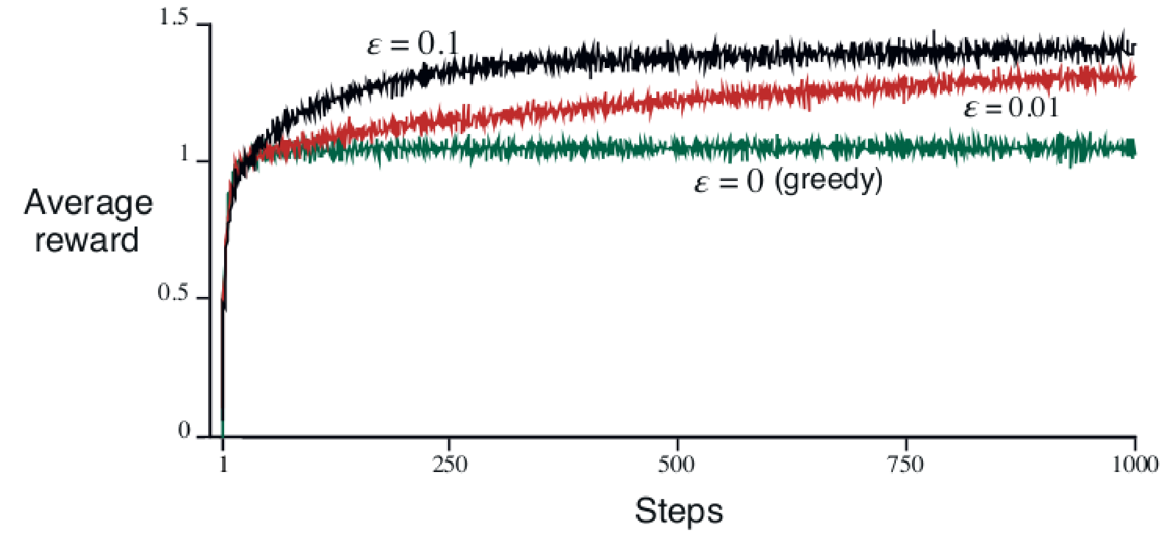
Example: 10-armed bandit problem

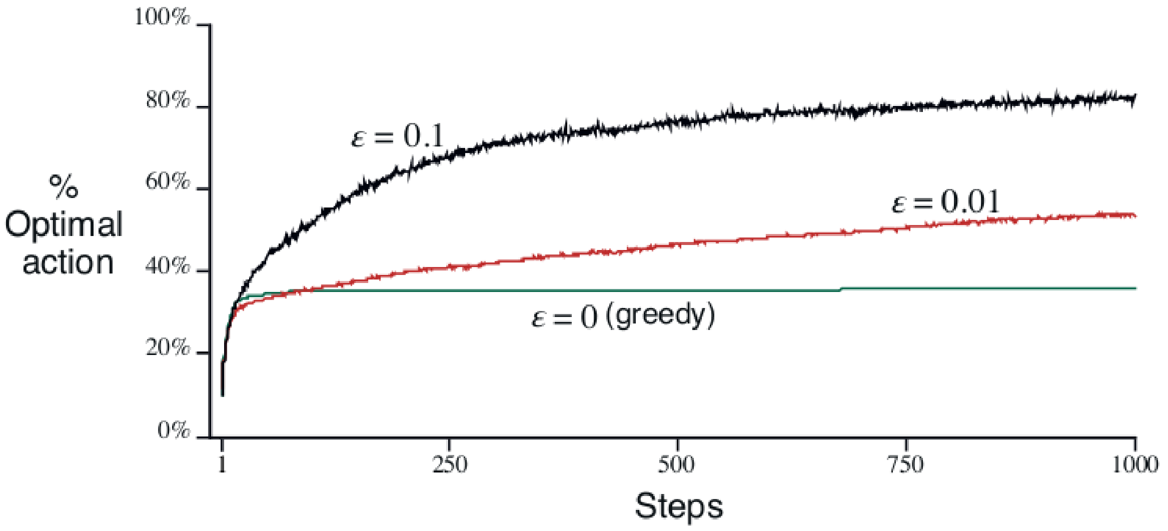


Create a new 10-armed bandit by sampling: $Q^*(a) \sim N(0,1)$

For each action, reward drawn from a Gaussian distribution: $R_t \sim N(Q^*(a),1)$

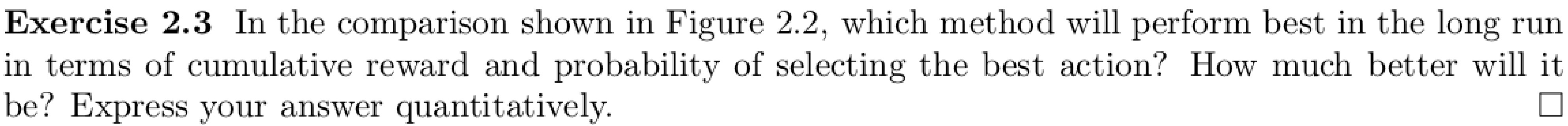
How does e-greedy action selection perform on the 10-armed bandit problem?





Results averaged over 2000 10-armed bandit problems and 1000 runs per problem.

Think-pair-share question



Exercise 2.3 In the comparison shown in Figure 2.2, which method will perform best in the long run in terms of cumulative reward and probability of selecting the best action? How much better will it be? Express your answer quantitatively. □

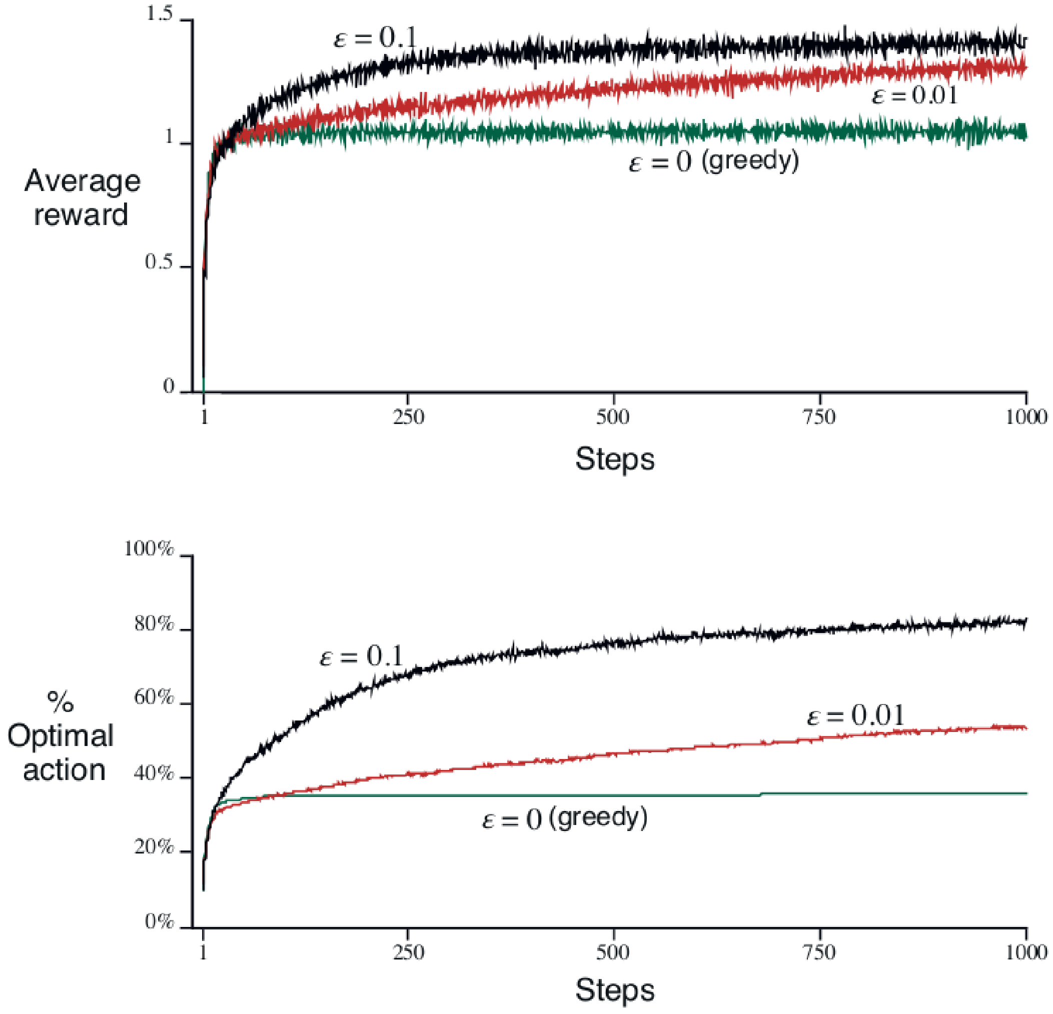
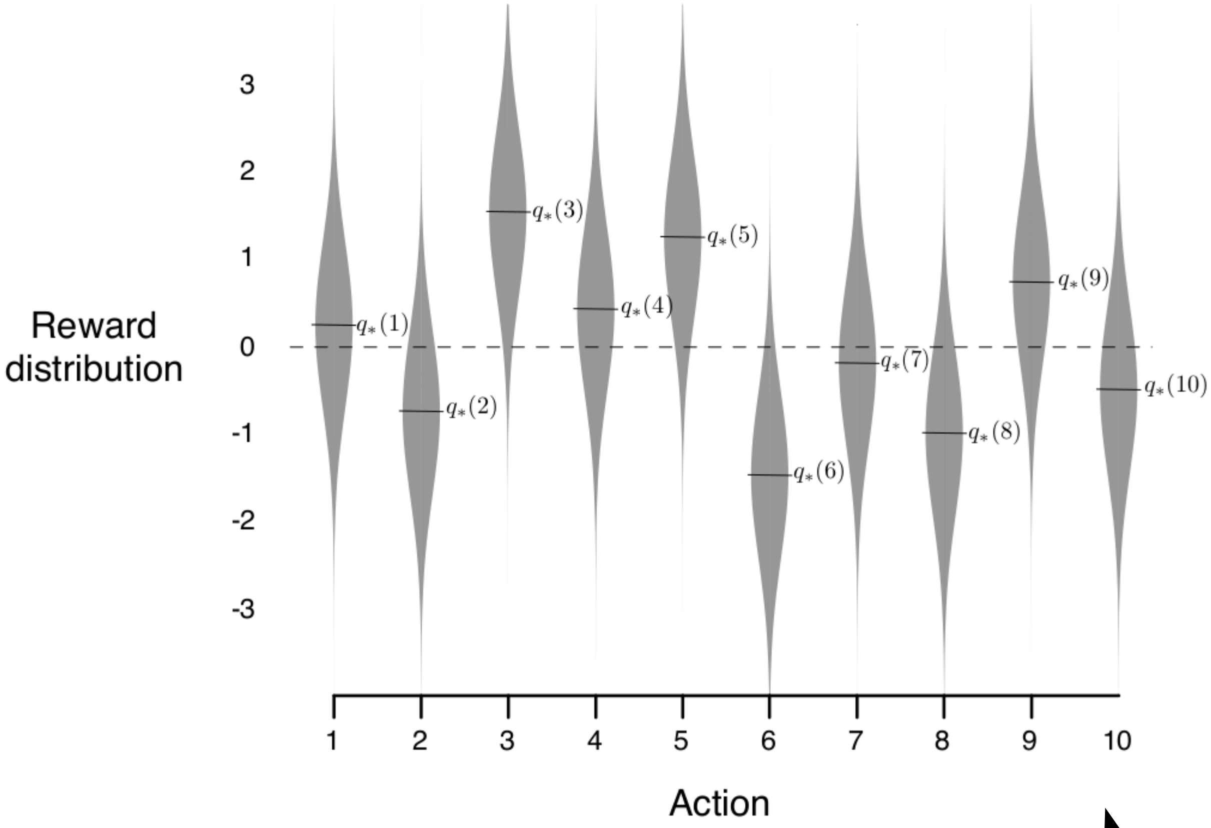


Figure 2.2

Non-stationary problems



What if the true action-values change over time?

Non-stationary problems

- Suppose the true action values change slowly over time
  - then we say that the problem is *nonstationary*
- In this case, sample averages are not a good idea (Why?)
- Better is an "exponential, recency-weighted average":

$$Q_{n+1} = Q_n + \alpha\Big[R_n - Q_n\Big]$$

$$= (1-\alpha)^n Q_1 + \sum_{i=1}^{n} \alpha(1-\alpha)^{n-i} R_i$$

where α is a constant, *step-size parameter*, $0 < \alpha \leq 1$

- There is bias due to Q_1 that becomes smaller over time

Convergence conditions

- To assure convergence with probability 1:

$$\sum_{n=1}^{\infty} \alpha_n(a) = \infty \qquad \text{and} \qquad \sum_{n=1}^{\infty} \alpha_n^2(a) < \infty$$

- e.g., $\alpha_n = \frac{1}{n}$

- not $\alpha_n = \frac{1}{n^2}$

if $\alpha_n = n^{-p}, \quad p \in (0,1)$
then convergence is
at the optimal rate:

$$O(1/\sqrt{n})$$

These are standard conditions for convergence for any monte carlo estimate

Think-pair-share

Exercise 2.4 If the step-size parameters, α_n, are not constant, then the estimate Q_n is a weighted average of previously received rewards with a weighting different from that given by (2.6). What is the weighting on each prior reward for the general case, analogous to (2.6), in terms of the sequence of step-size parameters? □

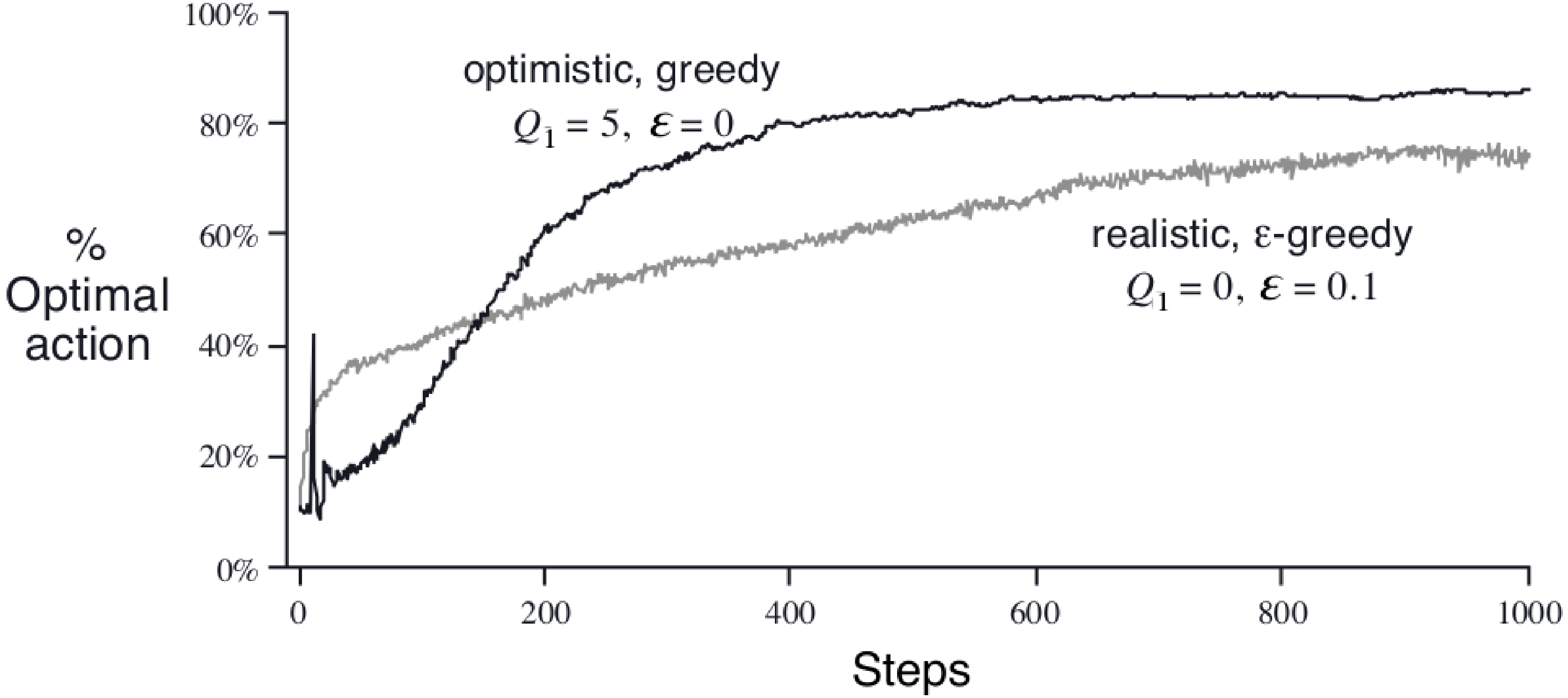
$$\begin{aligned}
Q_{n+1} &= Q_n + \alpha\Big[R_n - Q_n\Big] \\
&= \alpha R_n + (1-\alpha)Q_n \\
&= \alpha R_n + (1-\alpha)\left[\alpha R_{n-1} + (1-\alpha)Q_{n-1}\right] \\
&= \alpha R_n + (1-\alpha)\alpha R_{n-1} + (1-\alpha)^2 Q_{n-1} \\
&= \alpha R_n + (1-\alpha)\alpha R_{n-1} + (1-\alpha)^2 \alpha R_{n-2} + \\
&\qquad\qquad \cdots + (1-\alpha)^{n-1}\alpha R_1 + (1-\alpha)^n Q_1 \\
&= (1-\alpha)^n Q_1 + \sum_{i=1}^{n} \underbrace{\alpha(1-\alpha)^{n-i}} R_i.
\end{aligned}$$

Weighting on prior rewards for constant-alpha case

Optimistic initial values

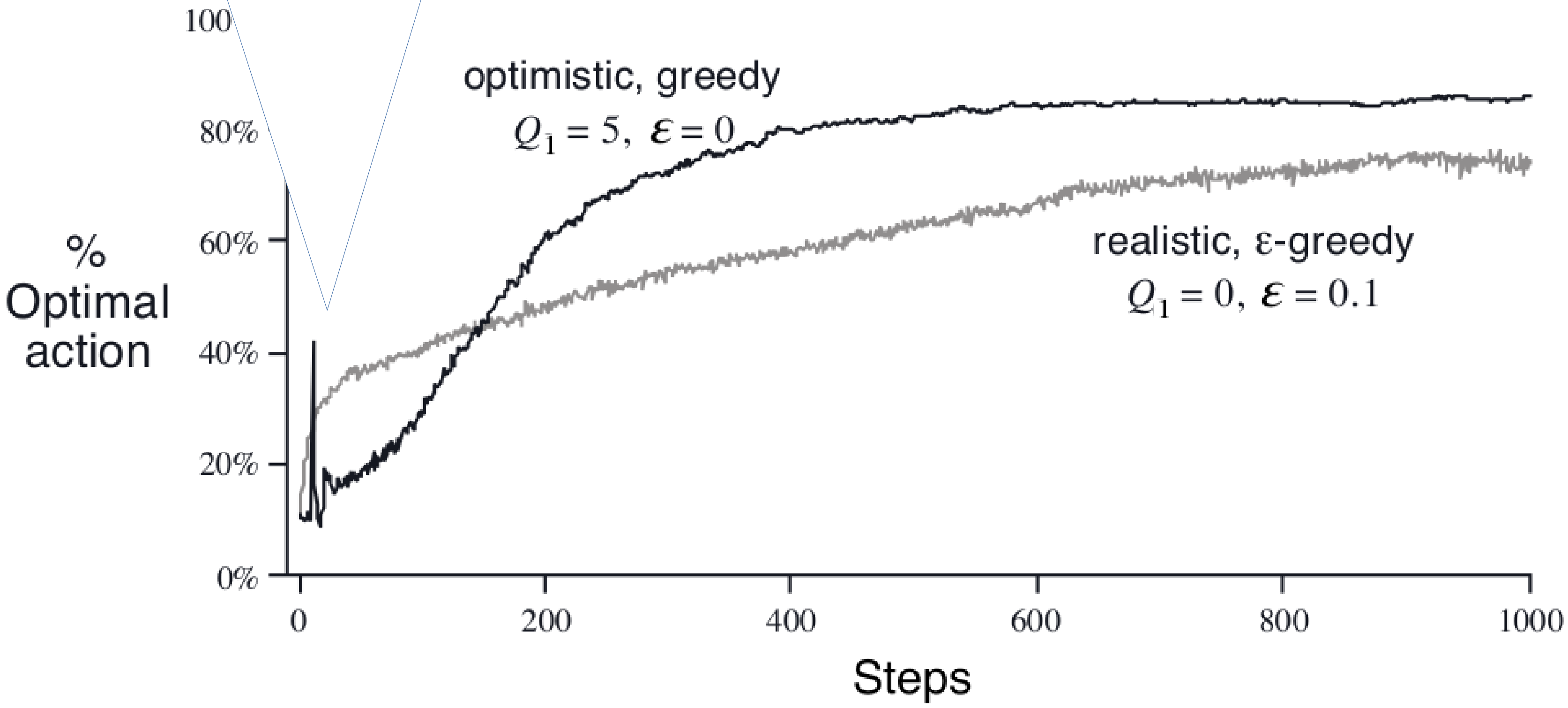
- All methods so far depend on $Q_1(a)$, i.e., they are biased. So far we have used $Q_1(a) = 0$

- Suppose we initialize the action values *optimistically* ($Q_1(a) = 5$), e.g., on the 10-armed testbed (with $\alpha = 0.1$)



Think-pair-share question

What do you think accounts for these spikes?
– these curves are averages over 2000
different 10-armed bandit tasks



UCB action selection

- A clever way of reducing exploration over time
- Estimate an upper bound on the true action values
- Select the action with the largest (estimated) upper bound

$$A_t \doteq \operatorname*{argmax}_a \left[Q_t(a) + c\sqrt{\frac{\log t}{N_t(a)}} \right]$$

UCB action selection

- A clever way of reducing exploration over time
- Estimate an upper bound on the true action values
- Select the action with the largest (estimated) upper bound

$$A_t \doteq \underset{a}{\arg\max}\left[Q_t(a) + c\sqrt{\frac{\log t}{N_t(a)}}\,\right]$$

This term is an upper bound on the likely value of this action based on our uncertainty

The cool thing about the UCB form is that *regret* is bounded logarithmically w/ the number of actions.

UCB action selection

- A clever way of reducing exploration over time
- Estimate an upper bound on the true action values
- Select the action with the largest (estimated) upper bound

$$A_t \doteq \underset{a}{\arg\max}\left[Q_t(a) + c\sqrt{\frac{\log t}{N_t(a)}}\right]$$

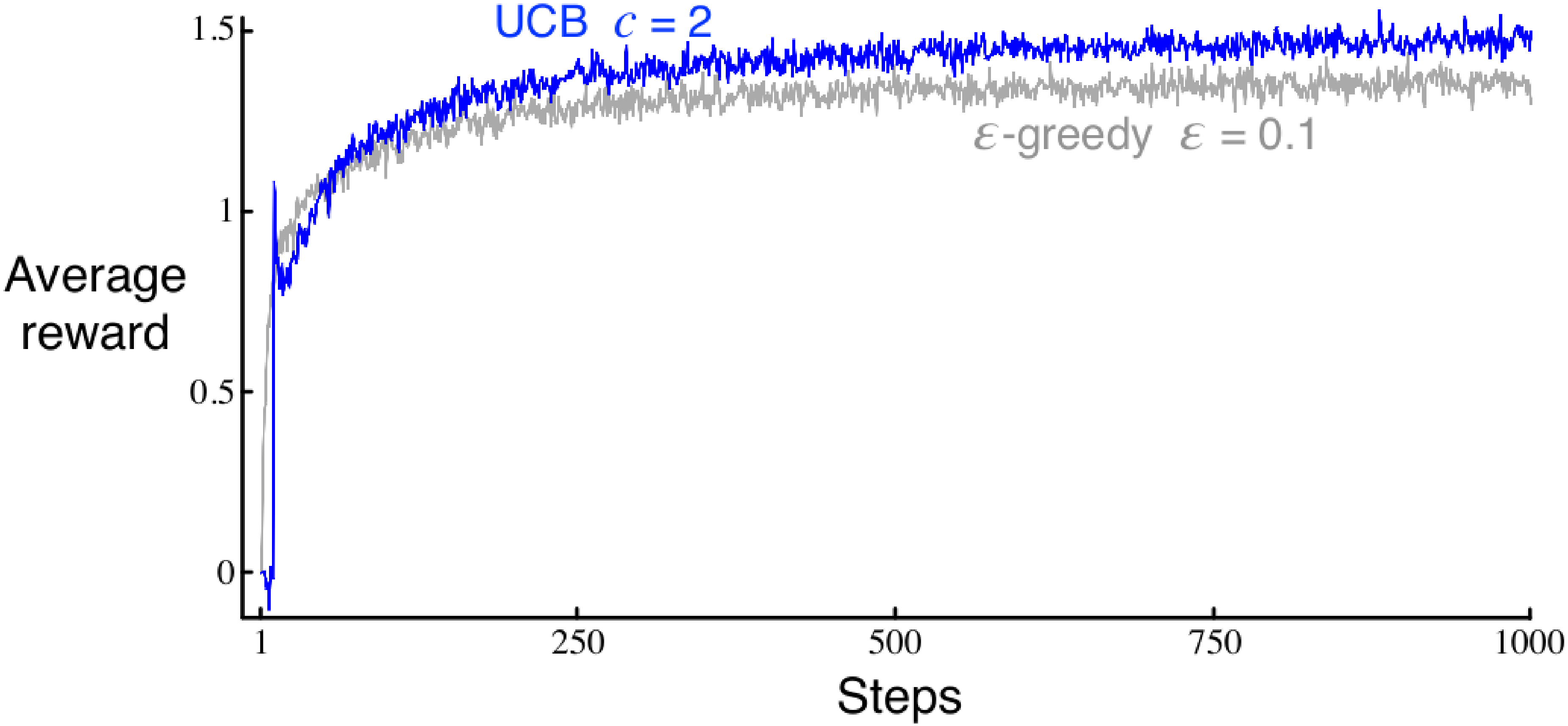
Regret: difference between your expected return using this strategy and how well you might have done if you knew which arm was best in advance

ikely

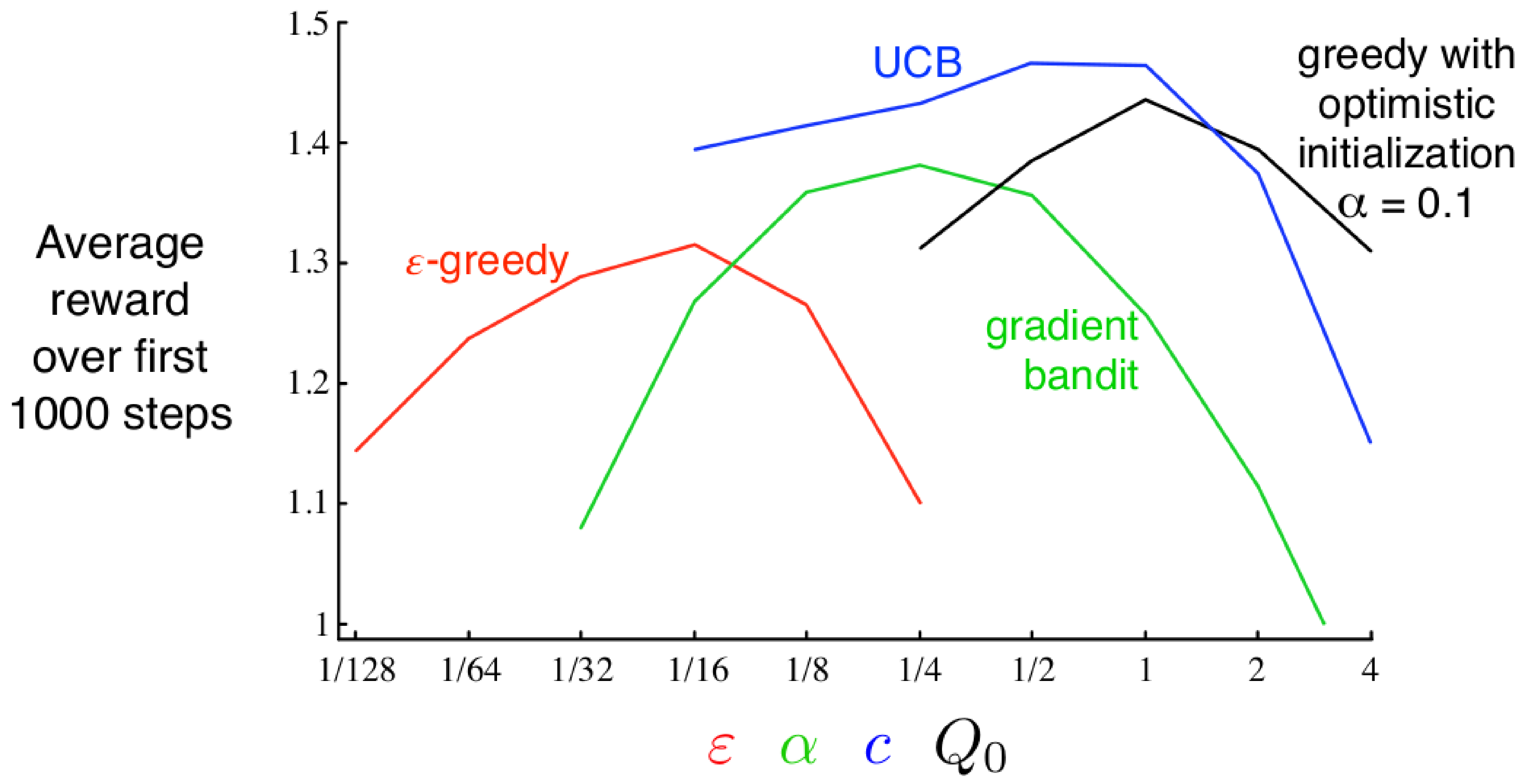
certainty

The cool thing about the UCB form is that *regret* is bounded logarithmically w/ the number of actions.

UCB action selection



Summary


Average reward over first 1000 steps
1.5
1.4
1.3
1.2
1.1
1
UCB
greedy with optimistic initialization α = 0.1
ε-greedy
gradient bandit
1/128
1/64
1/32
1/16
1/8
1/4
1/2
1
2
4
ε α c Q_0