CS 598 Statistical Reinforcement Learning

Nan Jiang
08/28/2018

If you are not registered for the class, please take a seat **after** class begins. Thanks!

Run > Hide > Fight

Emergencies can happen anywhere and at any time. It is important that we take a minute to prepare for a situation in which our safety or even our lives could depend on our ability to react quickly. When we're faced with almost any kind of emergency – like severe weather or if someone is trying to hurt you – we have three options: Run, hide or fight.

Run

Leaving the area quickly is the best option if it is safe to do so.

- Take time now to learn the different ways to leave your building.
- Leave personal items behind.
- Assist those who need help, but consider whether doing so puts yourself at risk.
- Alert authorities of the emergency when it is safe to do so.

Hide

When you can't or don't want to run, take shelter indoors.

- Take time now to learn different ways to seek shelter in your building.
- If severe weather is imminent, go to the nearest indoor storm refuge area.
- If someone is trying to hurt you and you can't evacuate, get to a place where you can't be seen, lock or barricade your area if possible, silence your phone, don't make any noise and don't come out until you receive an Illini-Alert indicating it is safe to do so.

Fight

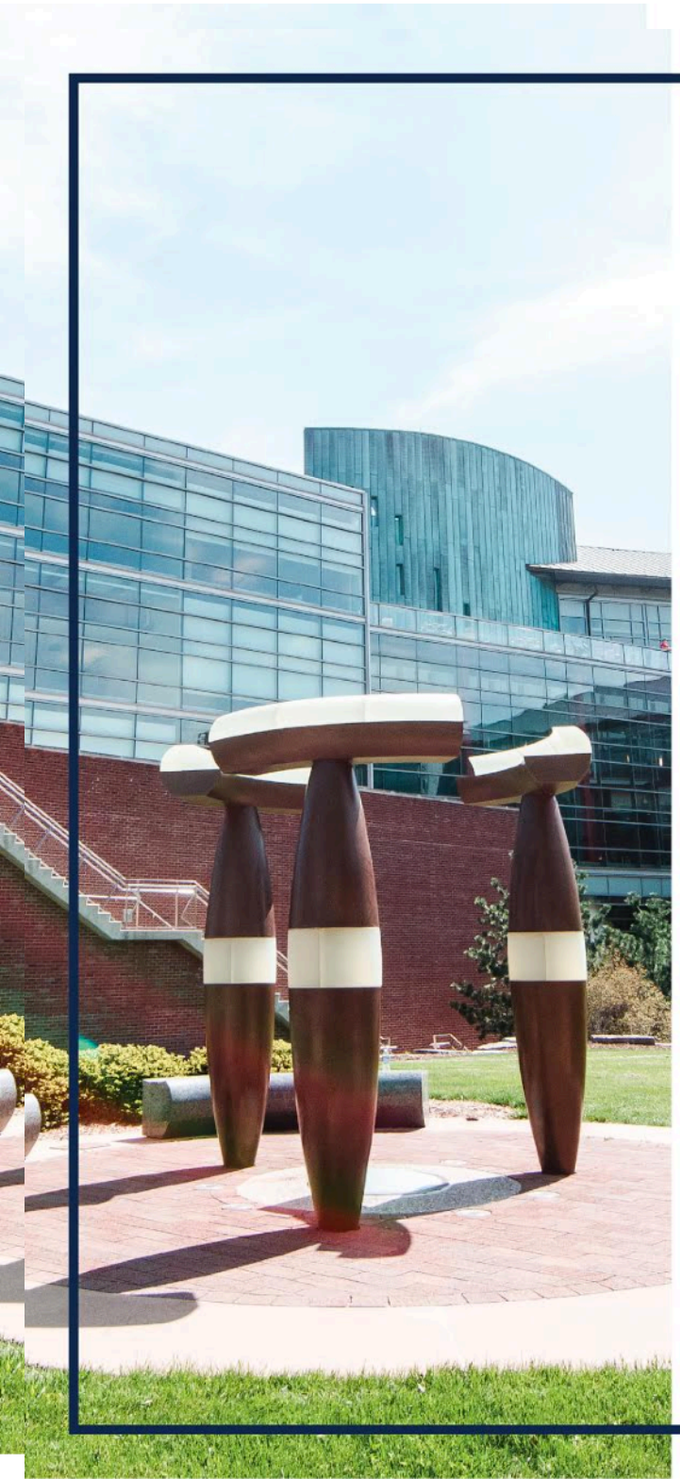
As a last resort, you may need to fight to increase your chances of survival.

- Think about what kind of common items are in your area which you can use to defend yourself.
- Team up with others to fight if the situation allows.
- Mentally prepare yourself – you may be in a fight for your life.

Please be aware of people with disabilities who may need additional assistance in emergency situations.

Other resources

- **police.illinois.edu/safe** for more information on how to prepare for emergencies, including how to run, hide or fight and building floor plans that can show you safe areas.
- **emergency.illinois.edu** to sign up for Illini-Alert text messages.
- **Follow the University of Illinois Police Department** on Twitter and Facebook to get regular updates about campus safety.



LOOKING FOR FUNDING? SCHOLARSHIPS? FELLOWSHIPS?

Mark your calendar for these upcoming deadlines:

September 5	NEW! FLIP Fellows
September 5	John Deere WCS Scholarship ($2,000 x4!)
September 5	JPMorgan Chase WCS Scholarship ($2,500 x2!)
September 5	NEW! IMC Trading Scholarship ($7,500 x2!)
October 1	NEW! Ada Lovelace Fellowship

Find details at: http://go.cs.illinois.edu/AwardDeadlines

Questions? Contact Samantha at shendon@illinois.edu

ILLINOIS
Computer Science

Overview

What's this course about?

- A grad-level seminar course on **theory of RL**
- with focus on sample complexity analyses
- proofs, proofs, and nothing else (not exactly but you get the idea)
- Seminar course can be anywhere between students presenting papers and a rigorous course
  - In this course I will deliver most (or all) of the lectures
  - I create my own material and this is the first time I teach this class, so things are not always polished
  - I will share slides. I will also do whiteboard proofs (maybe with OneNote so that I can share electronically)

Example of what we will get to…

$$\begin{aligned}\|V_f - V_{f_k}\|_\nu^2 &= \int_{\mathcal{X}} (\max_{a\in\mathcal{A}} f(x,a) - \max_{a'\in\mathcal{A}} f_k(x,a'))^2 \nu(x)dx \\ &\le \int_{\mathcal{X}} (f(x,\pi_{f,f_k}(x)) - f_k(x,\pi_{f,f_k}(x)))^2 \nu(x)dx = \|f - f_k\|^2_{\nu\times\pi_{f,f_k}}. \qquad \square\end{aligned}$$

Now we can bound $\|f^\star - f_k\|_{\nu\times\pi}$ using Lemma 1:

$$\begin{aligned}\|f_k - f^\star\|_{\nu\times\pi} &= \|f_k - \mathcal{T}f_{k-1} + \mathcal{T}f_{k-1} - f^\star\|_{\nu\times\pi} \\ &\le \|f_k - \mathcal{T}f_{k-1}\|_{\nu\times\pi} + \|\mathcal{T}f_{k-1} - \mathcal{T}f^\star\|_{\nu\times\pi} \\ &\le \sqrt{|\mathcal{A}|C}\,\|f_k - \mathcal{T}f_{k-1}\|_{\mu\times U} + \gamma\|V_{f_{k-1}} - V^\star\|_{P(\nu\times\pi)} \quad \text{(passing through the dynamics)} \\ &\le \sqrt{|\mathcal{A}|C}\,\|f_k - \mathcal{T}f_{k-1}\|_{\mu\times U} + \gamma\|f_{k-1} - f^\star\|_{P(\nu\times\pi)\times\pi_{f_{k-1},f^\star}}. \qquad \text{(Lemma 1)}\end{aligned}$$

Here $P(\cdot)$ is defined by the following generative procedure: $x' \sim P(\cdot) \Leftrightarrow x \sim (\cdot), x' \sim P(\cdot|x,a)$.

Note that we can apply the same analysis on $P(\nu\times\pi)\times\pi_{f_{k-1},f^\star}$ and expand the inequality k times. It then suffices to upper bound $\|f_k - \mathcal{T}f_{k-1}\|_{\mu\times U}$.

$$\begin{aligned}\|f_k - \mathcal{T}f_{k-1}\|^2_{\mu\times U} &= L_{\mu\times U}(f_k; f_{k-1}) - L_{\mu\times U}(\mathcal{T}f_{k-1}; f_{k-1}) \quad (L \text{ squared loss + } \mathcal{T}f_{k-1} \text{ Bayes optimal}) \\ &\le L_D(f_k; f_{k-1}) - L_D(\mathcal{T}f_{k-1}; f_{k-1}) + 2\epsilon \qquad (\mathcal{T}f_{k-1} \in \mathcal{F}) \\ &\le 2\epsilon. \qquad (f_k \text{ minimizes } L_D(\cdot\,; f_{k-1}))\end{aligned}$$

Note that the RHS does not depend on k, so we conclude that

$$\|f_k - f^\star\|_{\nu\times\pi} \le \frac{1-\gamma^k}{1-\gamma}\sqrt{2|\mathcal{A}|C\epsilon} + \gamma^k\frac{R_{\max}}{1-\gamma}.$$

Apply this to Equation 1 and we get

$$v^\star - v^{\pi_{f_k}} \le \frac{2}{1-\gamma}\left(\frac{1-\gamma^k}{1-\gamma}\sqrt{2|\mathcal{A}|C\epsilon} + \gamma^k\frac{R_{\max}}{1-\gamma}\right).$$

Who should take this course?

- This course will be a good fit for you if you…
  - are interested in understanding RL mathematically
  - want to do research in theory of RL or related area
  - want to work with me
  - are comfortable with maths
- This course will not be a good fit if you…
  - are mostly interested in implementing RL algorithms and making things to work
    we won't cover any engineering tricks (which are essential to e.g., Deep RL); check other RL courses offered across the campus—I believe some are much more empirical

Prerequisites

- Maths
  - Linear algebra, probability & statistics, basic calculus
  - Markov chains
  - Optional: stochastic processes, numerical analysis
  - Useful for research: TCS background, empirical processes and statistical learning theory, optimization, online learning
- Exposure to ML
  - e.g., CS 446 Machine Learning
  - Experience with RL

Coursework

- Some readings after/before class
- Ad hoc homework to help digest particular material. Deadline will be lenient & TBA at the time of assignment
- Main assignment: course project
  - Baseline: reproduce theoretical analysis in existing papers
  - Advanced: identify an interesting/challenging extension to the paper and explore the novel research question yourself
  - Or, just work on a novel research question (need to talk with me)
- Work on your own. Working in pairs might be allowed; TBA after course registration becomes stable.

Course project (cont.)

- I will give a list of references and potential topics within the first month (hopefully sooner).
- You will need to submit:
  - A brief proposal (~1/2 page). Tentative deadline: end of Oct
    - what's the topic and what papers you plan to work on
    - why you choose the topic: what interest you?
    - which aspect(s) you will focus on?
  - Final report: clarity, precision, and brevity are greatly valued. More details to come…
- All docs should be in **pdf**. Final report should be prepared using **LaTeX**.

Course project (cont. 2)

Rule of thumb

1. learn something that interests you
2. teach me something! (I wouldn't learn if I could not understand your report due to lack of clarity)
3. write a report similar to (or better than!) the notes I will share with you

Contents of the course

- See website
- lectures are highly tilted—many important topics in RL will not be covered in depth (e.g., TD). read more if you want to get a more comprehensive view of RL
    - the other opportunity to learn what's not covered in lectures is the project, as potential topics for projects are much broader than what's covered in class.

My goals

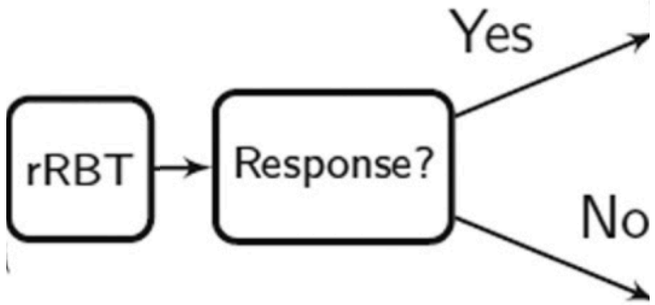
- Encourage you to do research in this area
- Introduce useful mathematical tools to you
- Refresh my memory on some topics and organize them into clean notes
- Learn from you

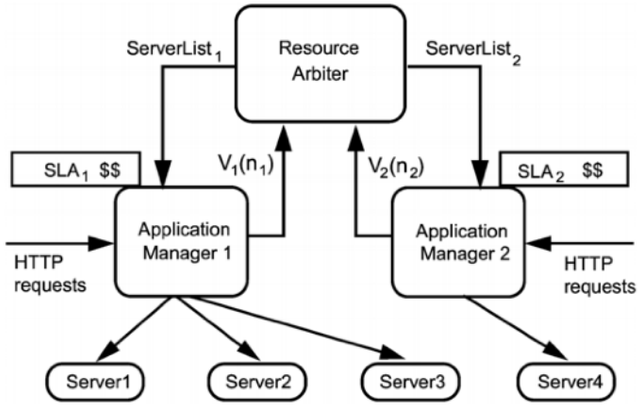
Logistics

- Course website: http://nanjiang.cs.illinois.edu/cs598/
  - logistics, links to slides/notes (uploaded after lectures), and resources (e.g., textbooks to consult, related courses), deadline & announcements
- Time & Location: Tue & Thu 2-3:15pm, 1302 Siebel.
- Office hours: By appointment. 3322 Siebel.
  - Questions about material: ad hoc meetings after lectures subject to my availability
  - Other things about the class (e.g., project): by appointment

Introduction to MDPs and RL

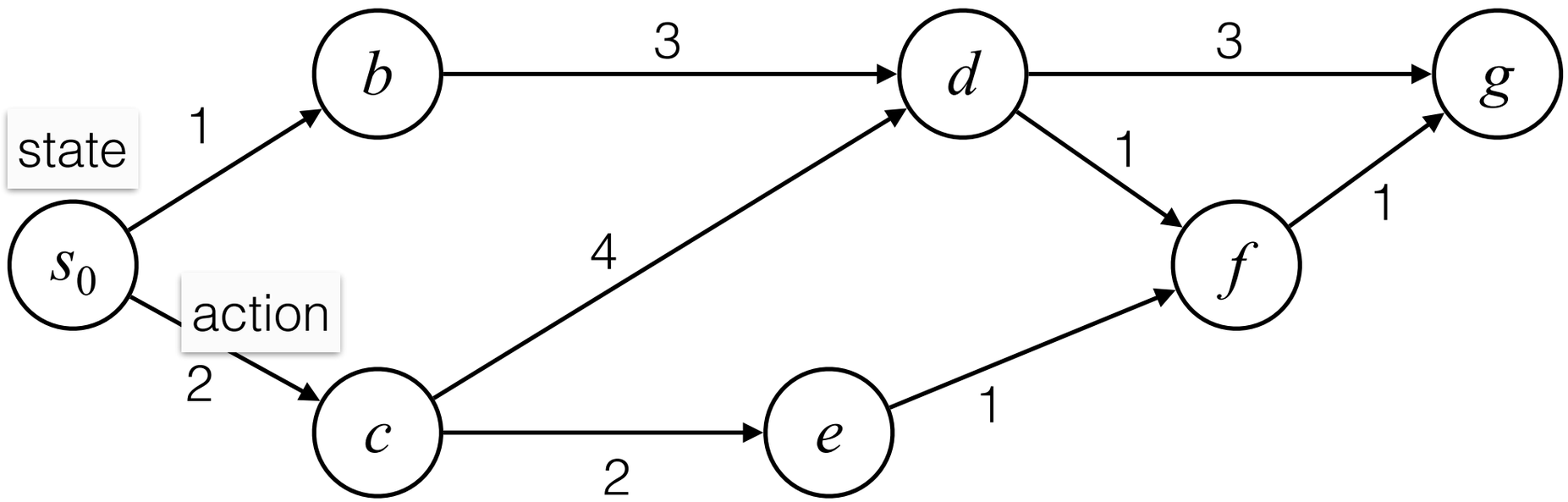
Reinforcement Learning (RL) Applications





[Levine et al'16] [Ng et al'03] [Singh et al'02] [Lei et al'12]
[Mandel et al'16] [Tesauro et al'07] [Mnih et al'15] [Silver et al'16]

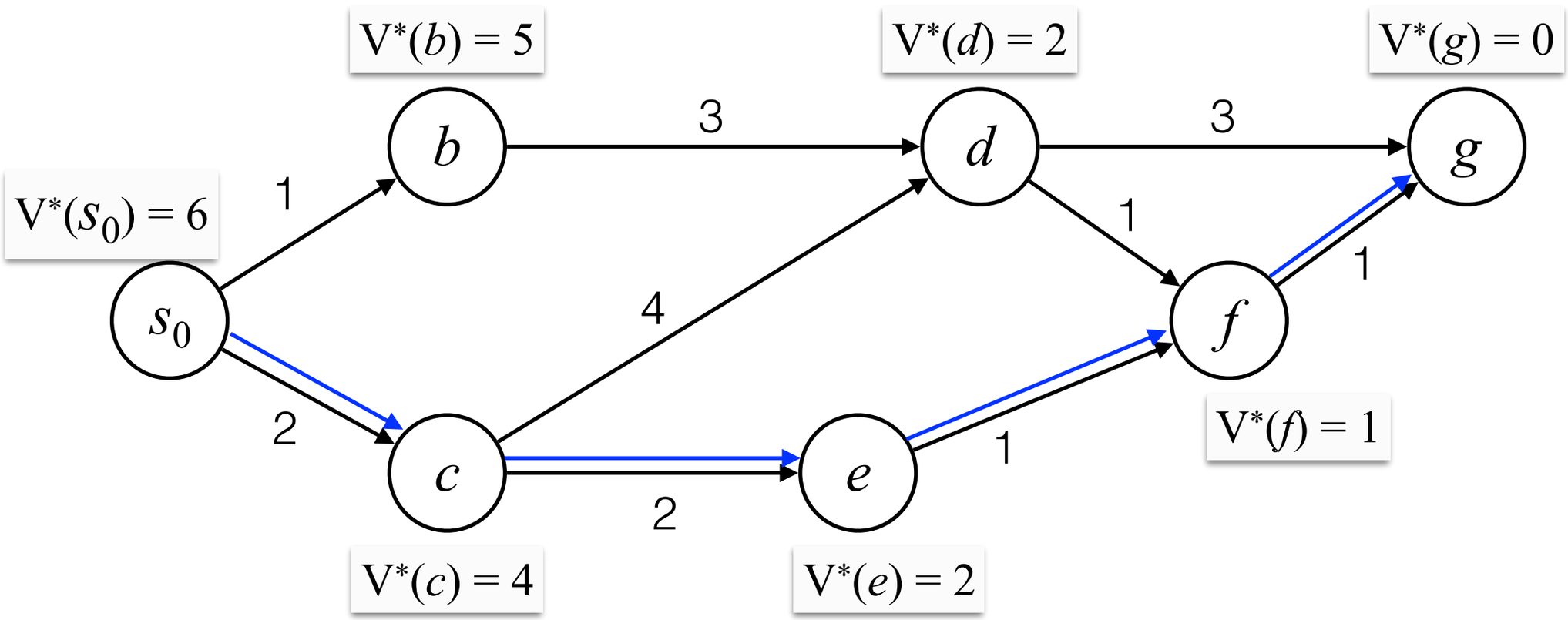
Shortest Path



Greedy is suboptimal due to delayed effects

Need **long-term planning**

Shortest Path



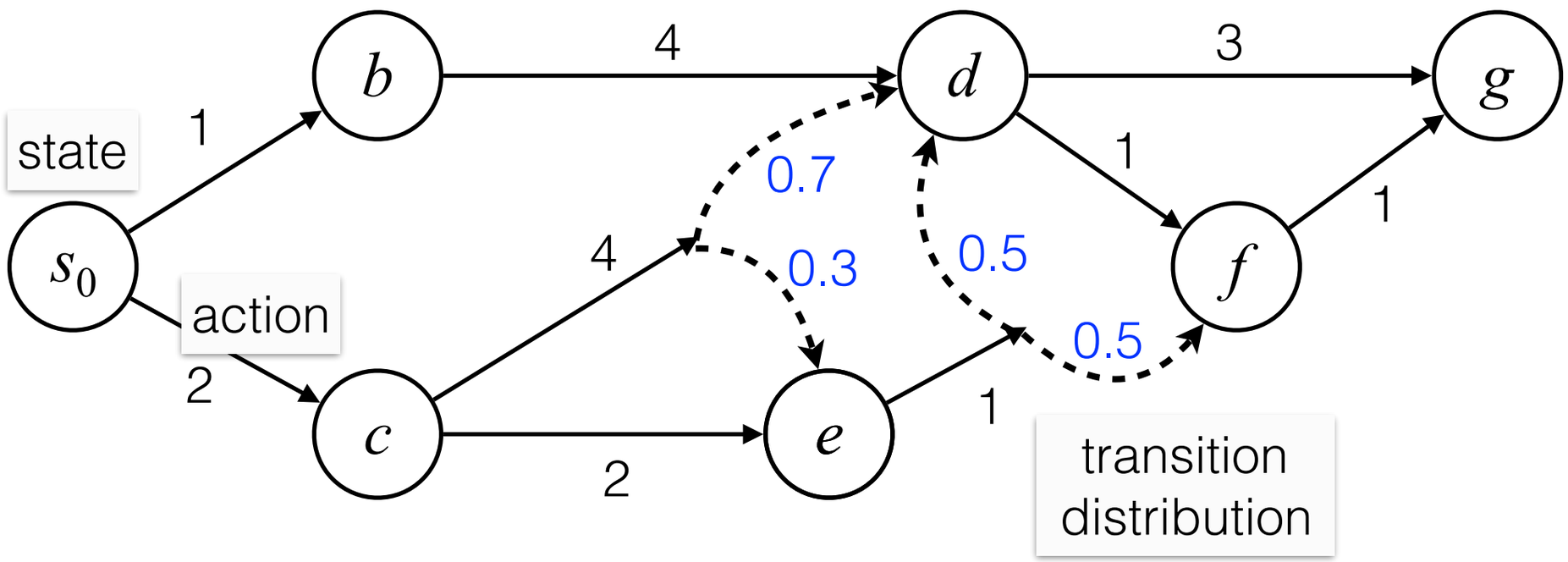
Bellman Equation $V^*(d) = \min\{3 + V^*(g)\ ,\ 2 + V^*(f)\}$

Greedy is suboptimal due to delayed effects

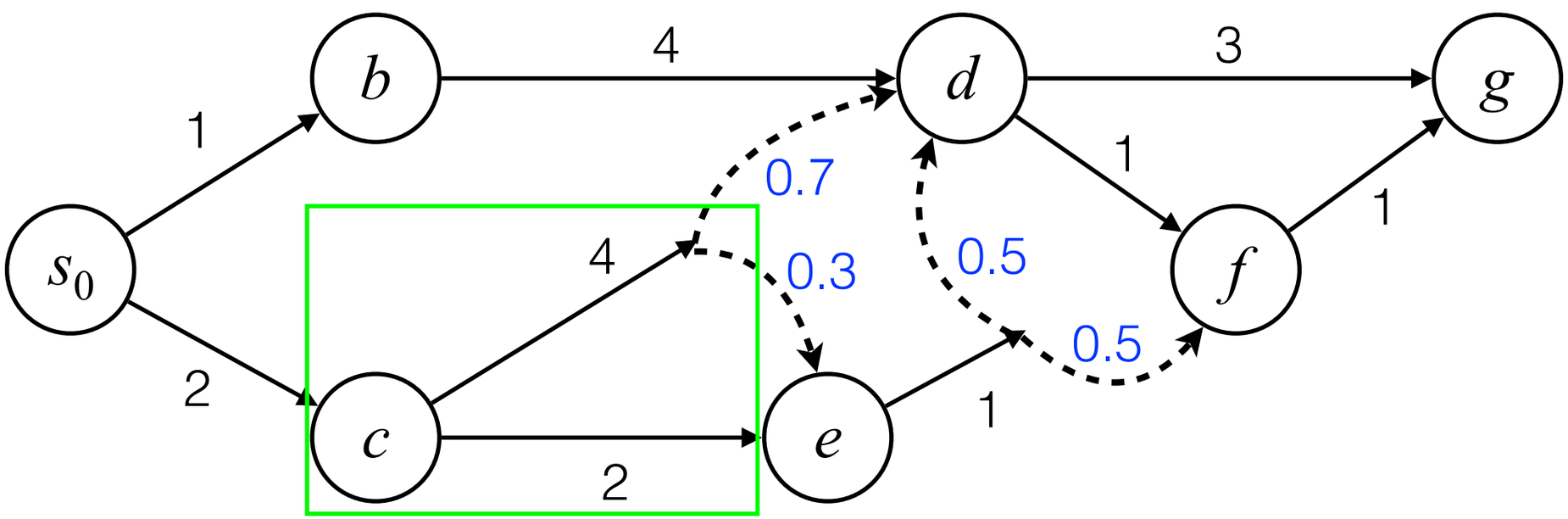
Need **long-term planning**

Stochastic Shortest Path

Markov Decision Process (MDP)



Stochastic Shortest Path



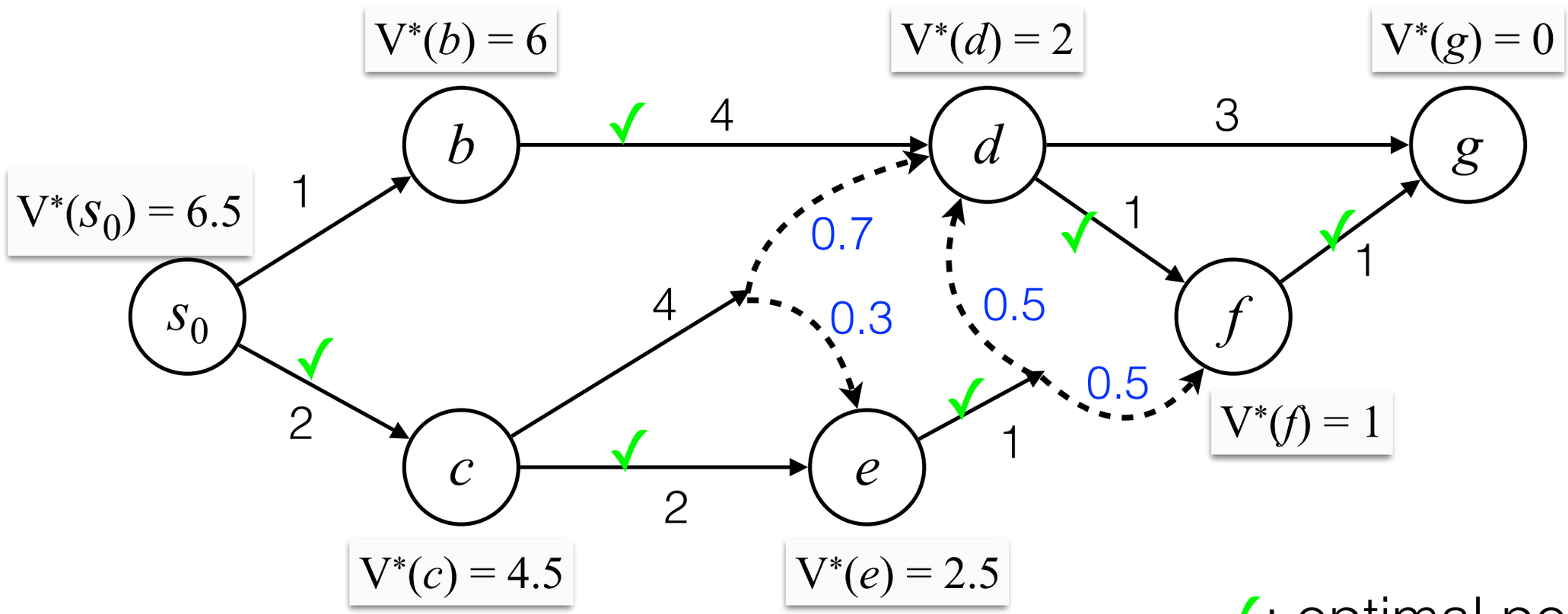
Bellman Equation

$$V^*(c) = \min\{4 + 0.7 \times V^*(d) + 0.3 \times V^*(e)\ ,\ 2 + V^*(e)\}$$

Greedy is suboptimal due to delayed effects

Need **long-term planning**

Stochastic Shortest Path



✓: optimal policy π^*

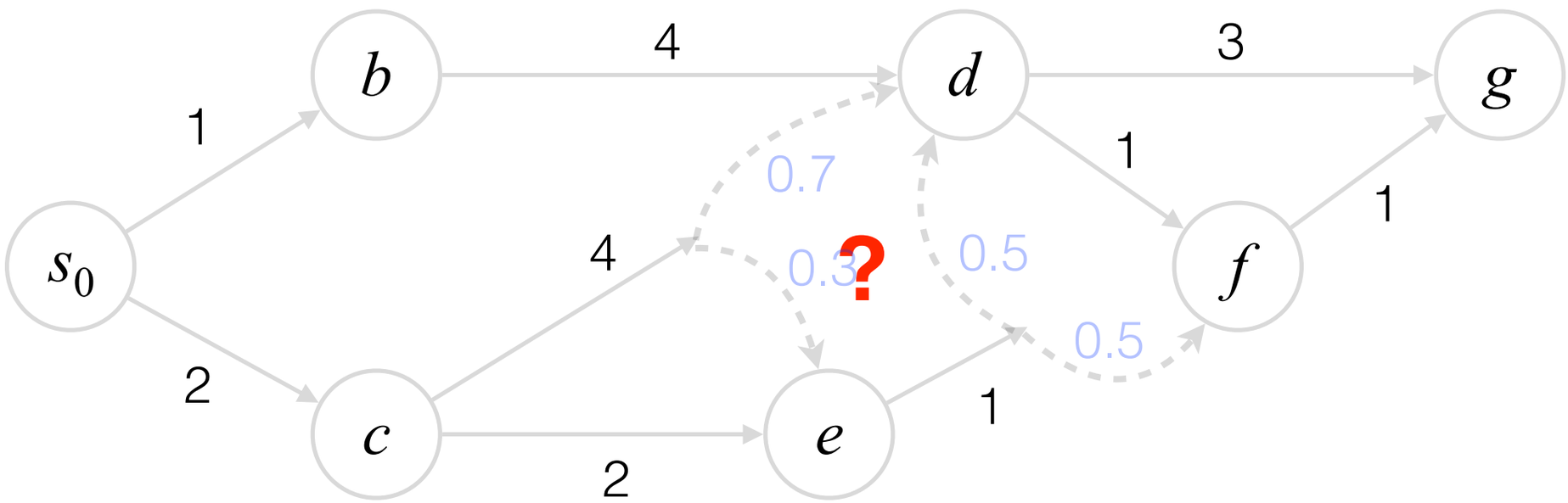
Bellman Equation

$$V^*(c) = \min\{4 + 0.7 \times V^*(d) + 0.3 \times V^*(e)\ ,\ 2 + V^*(e)\}$$

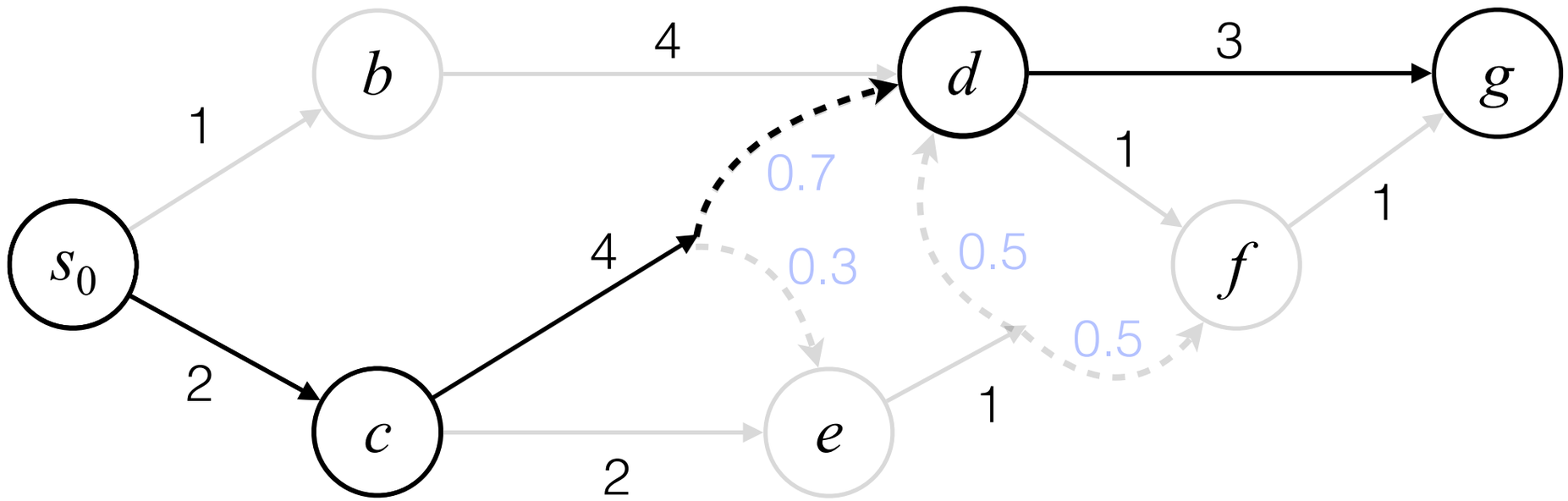
Greedy is suboptimal due to delayed effects

Need **long-term planning**

Stochastic Shortest Path
via trial-and-error

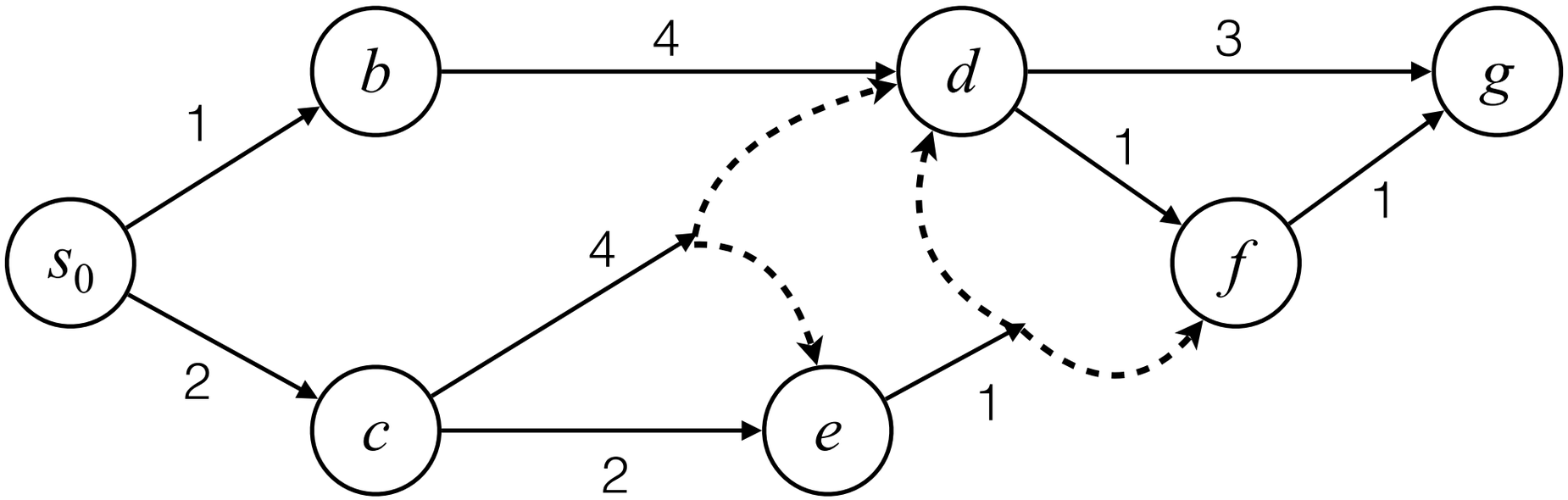


Stochastic Shortest Path
via trial-and-error



Trajectory 1: $s_0 \searrow c \nearrow d \rightarrow g$

Stochastic Shortest Path
via trial-and-error

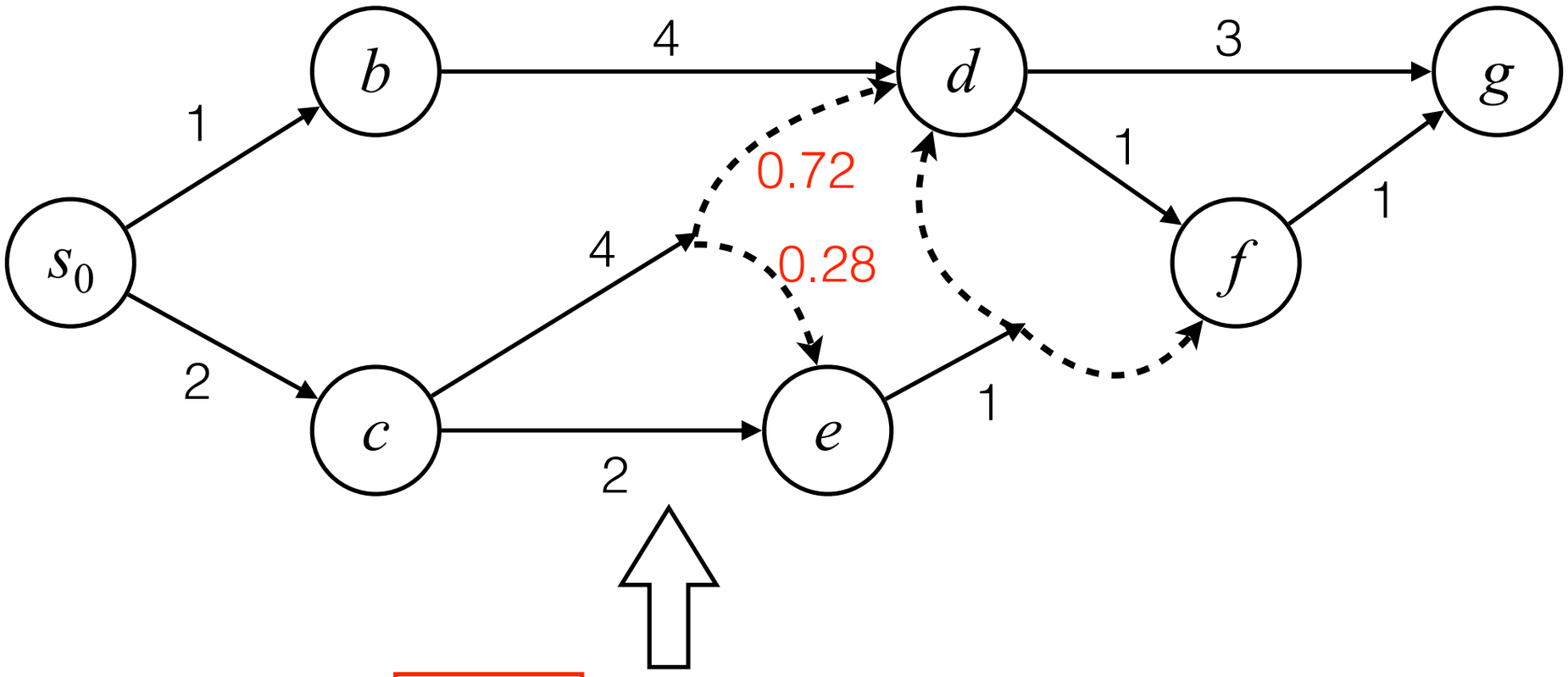


Trajectory 1: $s_0 \searrow c \nearrow d \rightarrow g$

Trajectory 2: $s_0 \searrow c \nearrow e \rightarrow f \nearrow g$

...

Stochastic Shortest Path
via trial-and-error

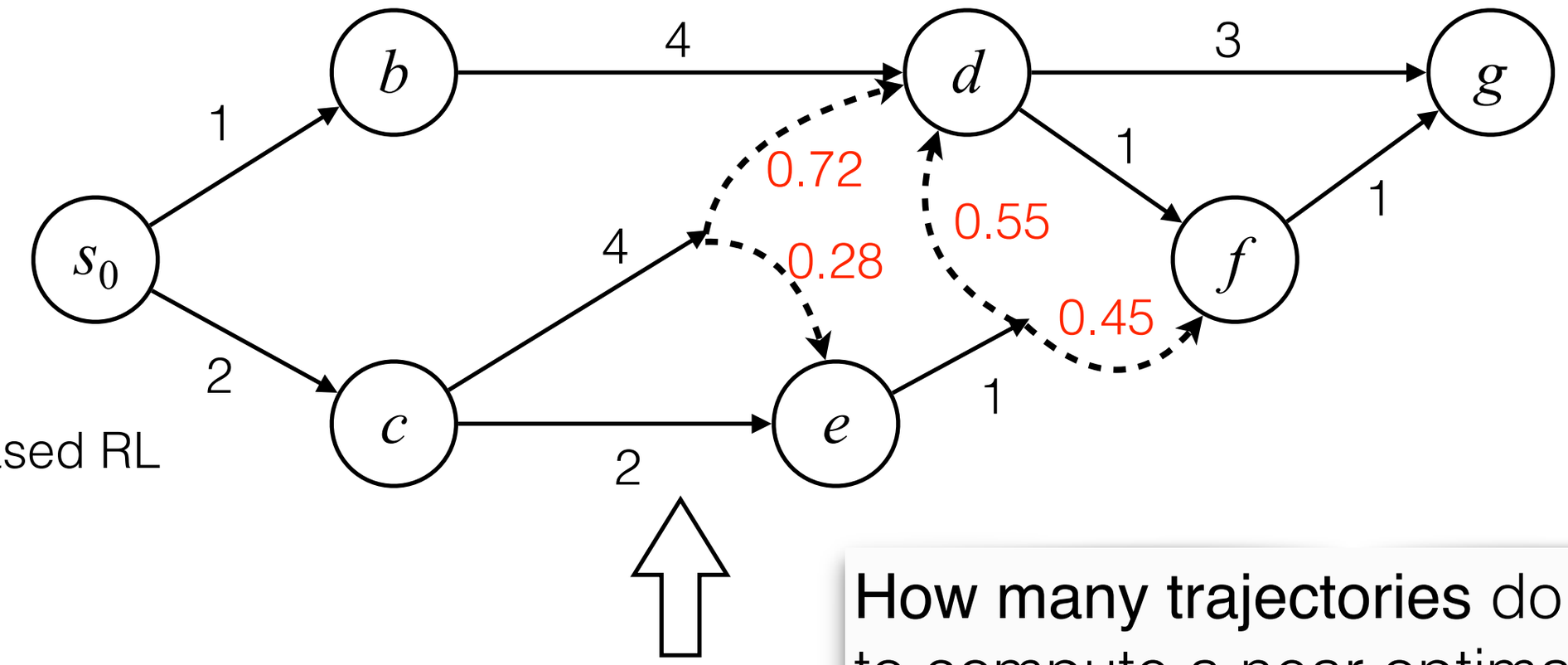


Trajectory 1: $s_0 \searrow \boxed{c \nearrow d} \rightarrow g$

Trajectory 2: $s_0 \searrow \boxed{c \nearrow e} \rightarrow f \nearrow g$

...

Stochastic Shortest Path
via trial-and-error



Model-based RL

Trajectory 1: $s_0 \searrow c \nearrow d \rightarrow g$

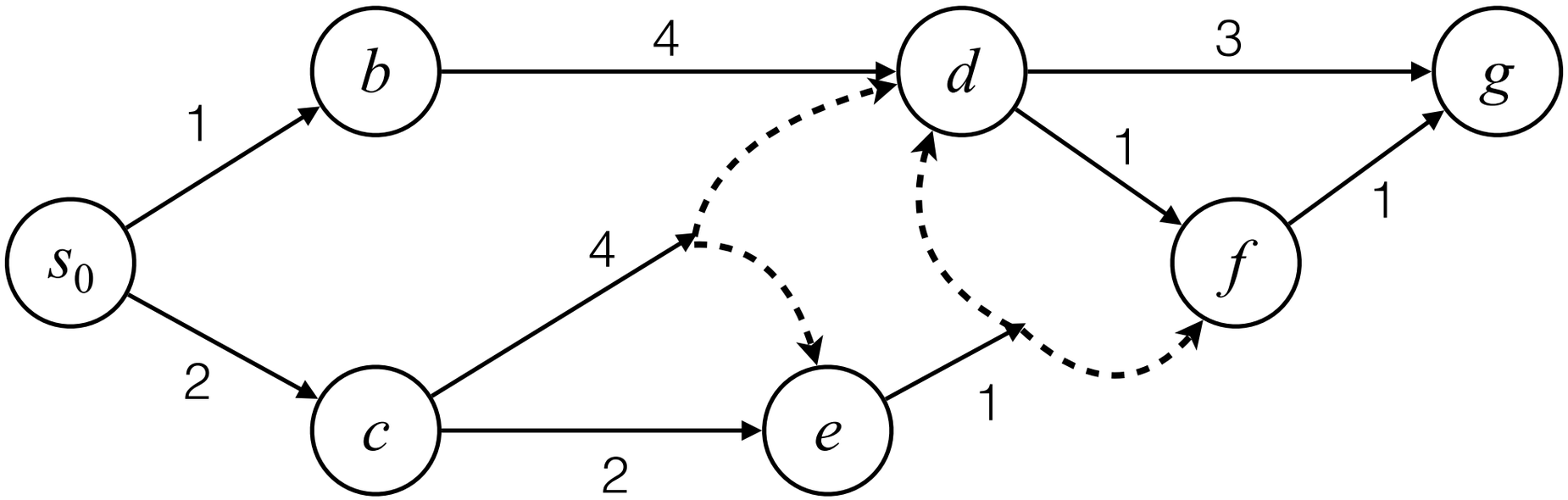
Trajectory 2: $s_0 \searrow c \nearrow e \rightarrow f \nearrow g$

…

How many trajectories do we need to compute a near-optimal policy?

sample complexity

Stochastic Shortest Path
via trial-and-error

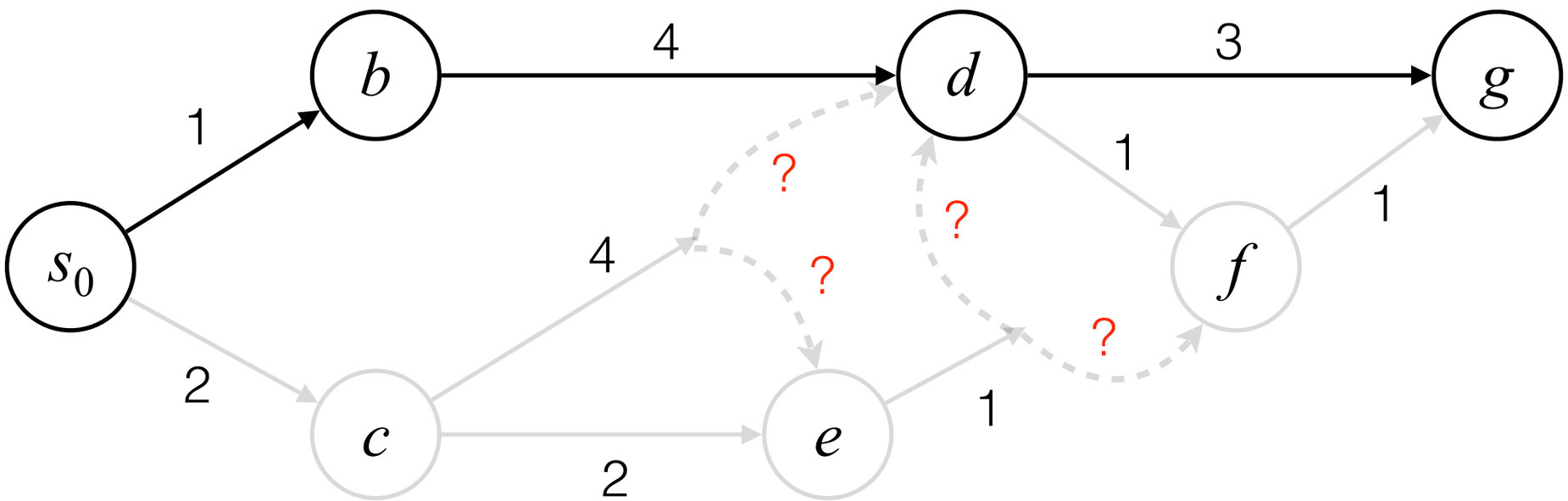


How many trajectories do we need to compute a near-optimal policy?

- Assume states & actions are visited uniformly
- #trajectories needed $\leq n \cdot$ (#state-action pairs)

#samples needed to estimate a multinomial distribution

Stochastic Shortest Path
via trial-and-error



How many trajectories do we need to compute a near-optimal policy?

Nontrivial! Need **exploration**

- Assume states & actions are visited uniformly
- #trajectories needed $\leq n \cdot$ (#state-action pairs)

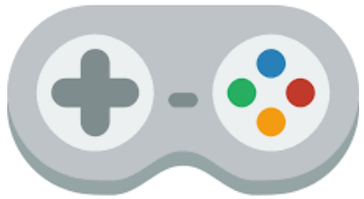
#samples needed to estimate a multinomial distribution

Video game playing

reward $r_t = R\,(s_t\,,\,a_t)$

state
$s_t \in S$

+20



e.g., random
spawn of enemies

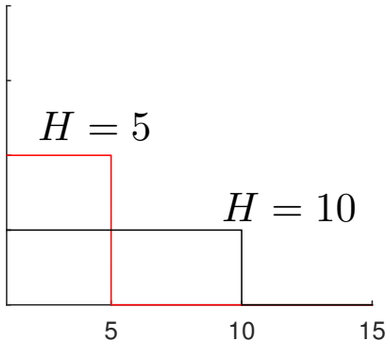
policy
$\pi\colon S \to A$

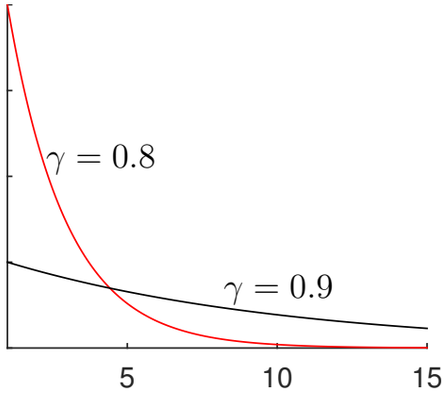
action $a_t \in A$

transition dynamics
$P\,(\,\cdot\,|\,s_t\,,\,a_t)$

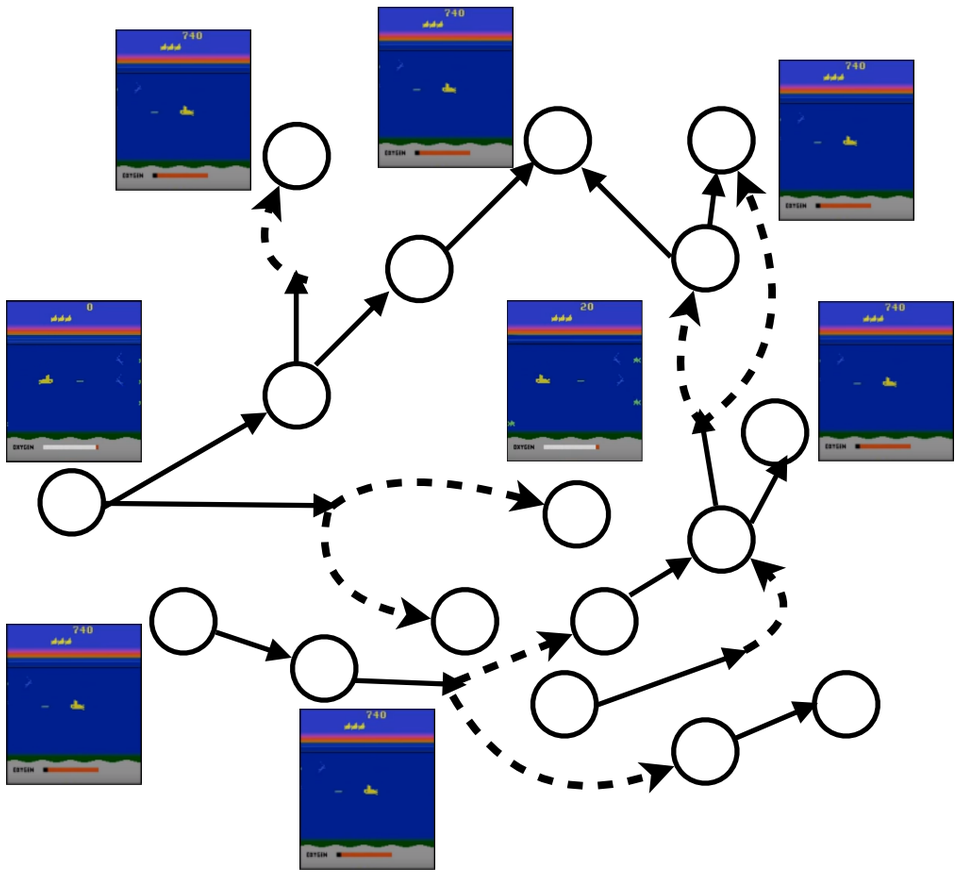
(unknown)

objective: maximize $\mathbb{E}\left[\sum_{t=1}^{H} r_t \,|\, \pi\right]$ (or $\mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \,|\, \pi\right]$)





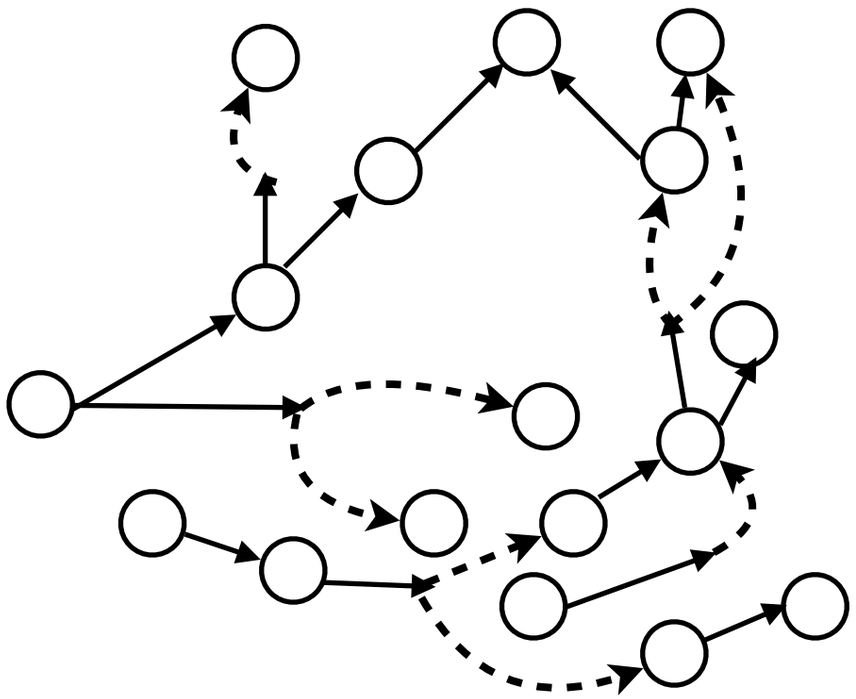
Video game playing

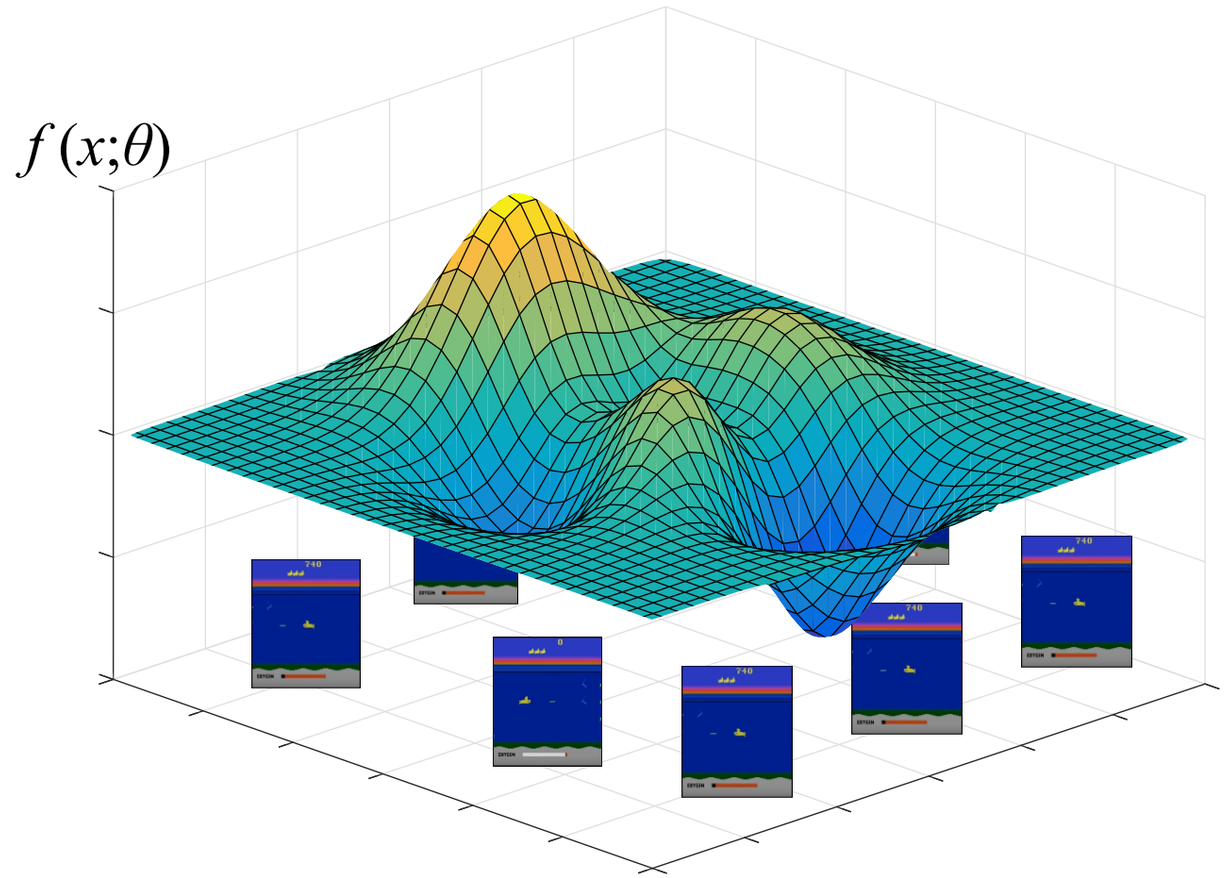


Need **generalization**

Value function approximation

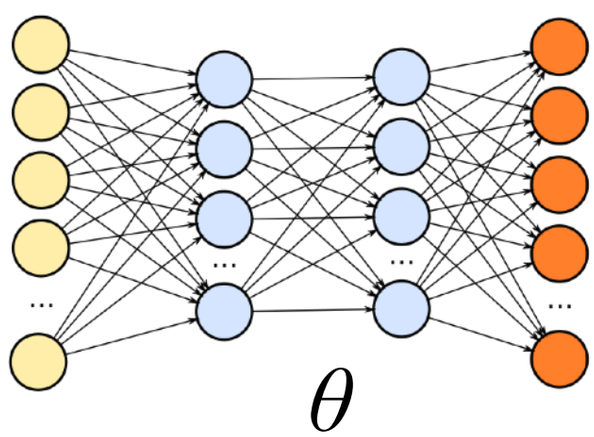
Video game playing





state features x

state features x

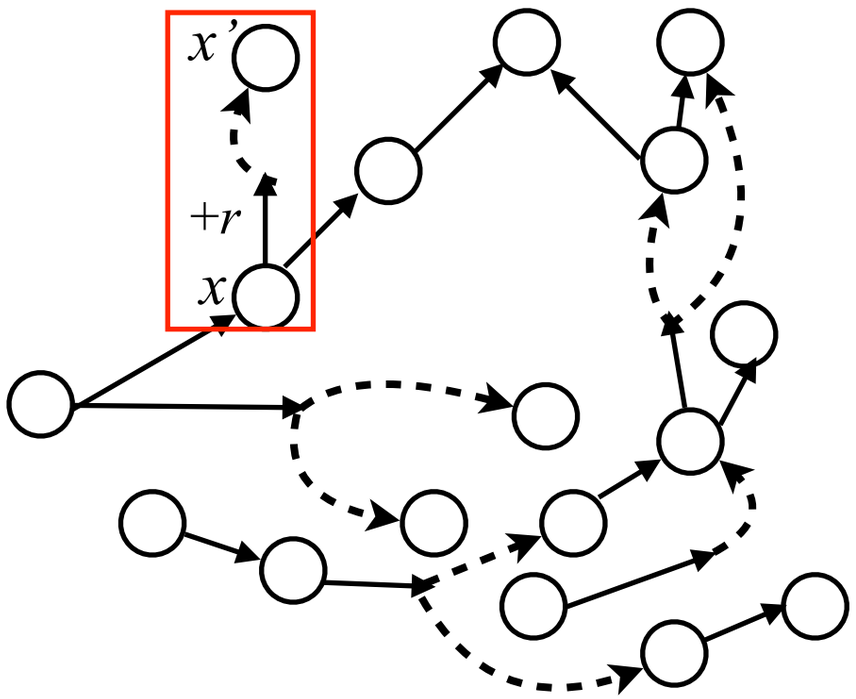


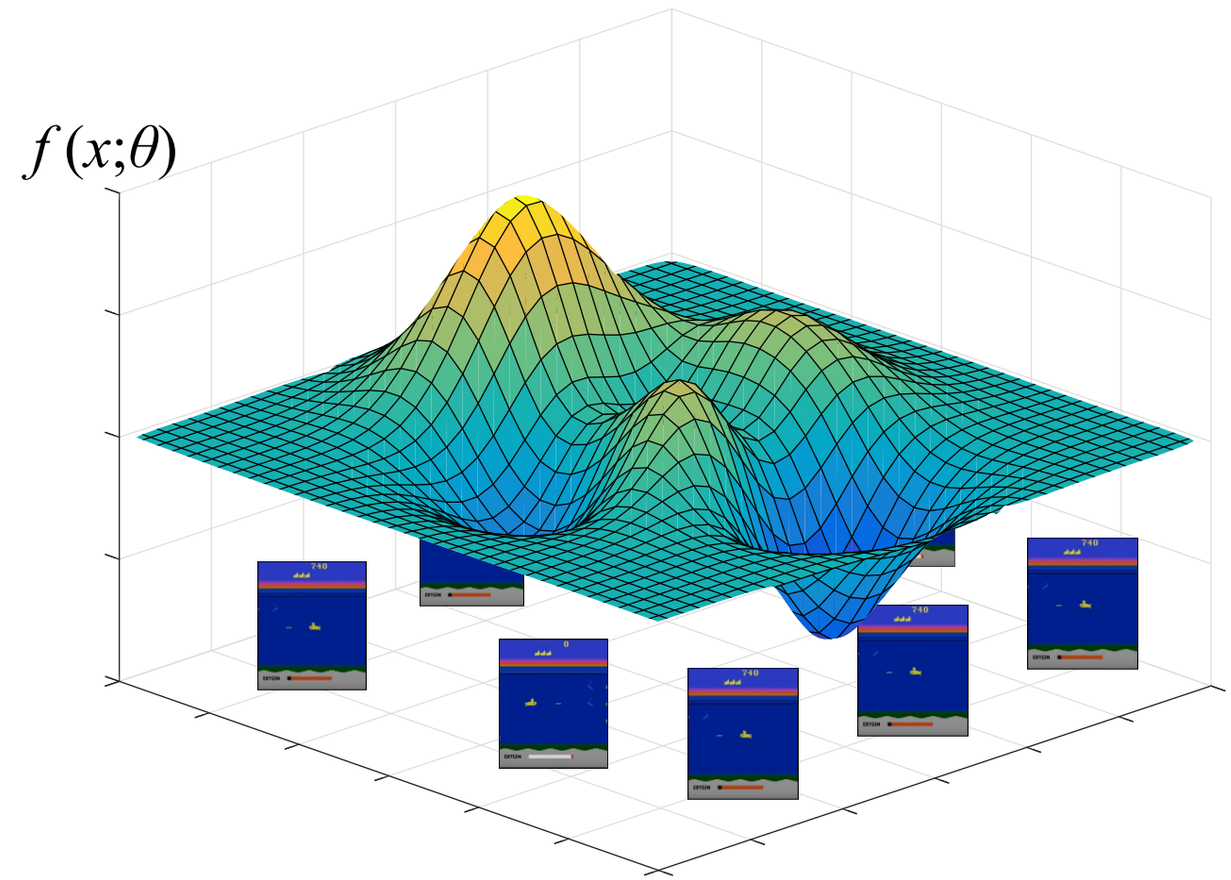
$f(x;\theta)$

Need **generalization**

Value function approximation

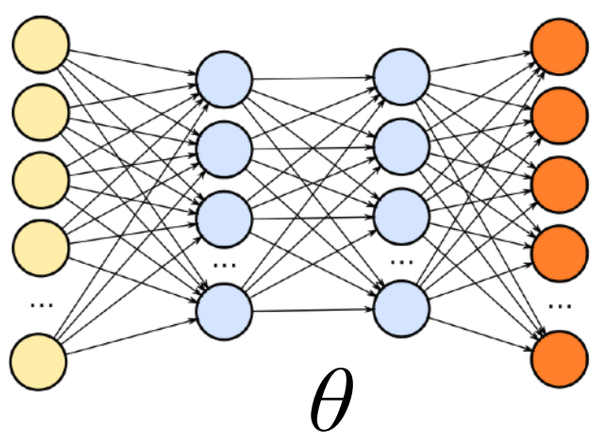
Video game playing





state features x

state features x ⇨



⇨ $f(x;\theta)$

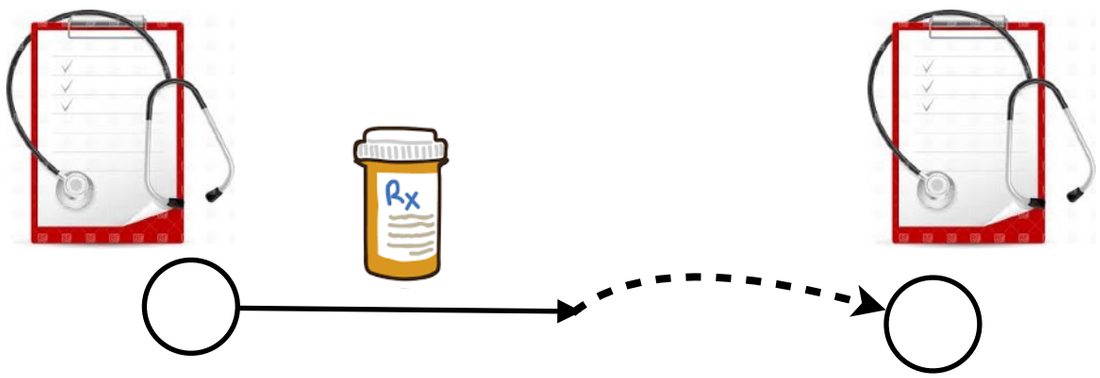
Need **generalization**

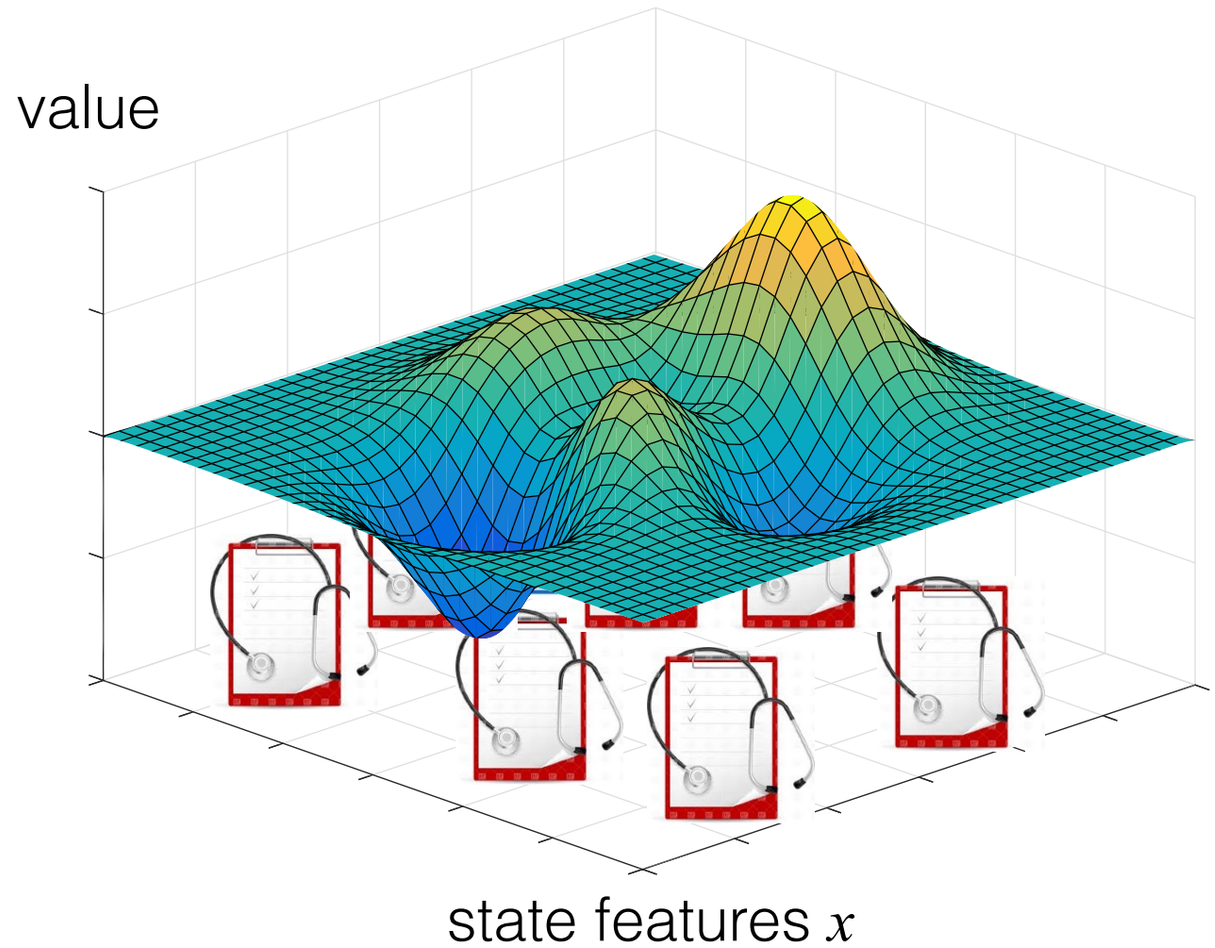
Value function approximation

Find θ s.t. $f(x;\theta) \approx r + \gamma \cdot \mathrm{E}_{x'|x}[f(x';\theta)] \;\Rightarrow\; f(\cdot\,;\theta) \approx \mathrm{V}^*$

$\ldots$

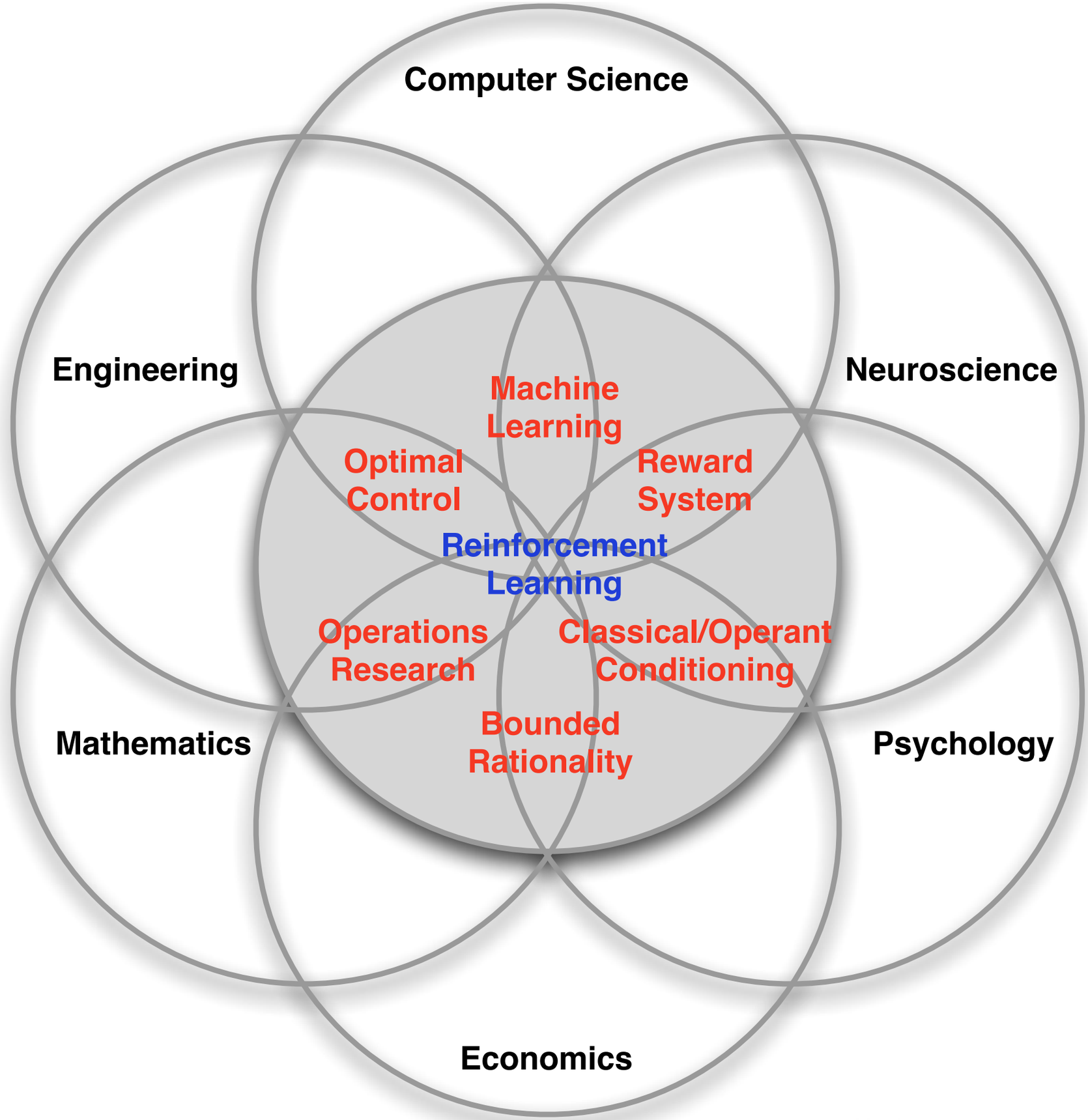
Adaptive medical treatment





- State: diagnosis
- Action: treatment
- Reward: progress in recovery

A Machine Learning view of RL



slide credit: David Silver

Supervised Learning

Given $\{(x^{(i)}, y^{(i)})\}$, learn $f : x \mapsto y$

- Online version: for round $t = 1, 2, \ldots,$ **the learner**
  - observes $x^{(t)}$
  - predicts $\hat{y}^{(t)}$
  - receives $y^{(t)}$
- Want to maximize # of correct predictions
- e.g., classifies if an image is about a dog, a cat, a plane, etc. (multi-class classification)
- Dataset is fixed for everyone
- “Full information setting”
- Core challenge: generalization

Contextual bandits

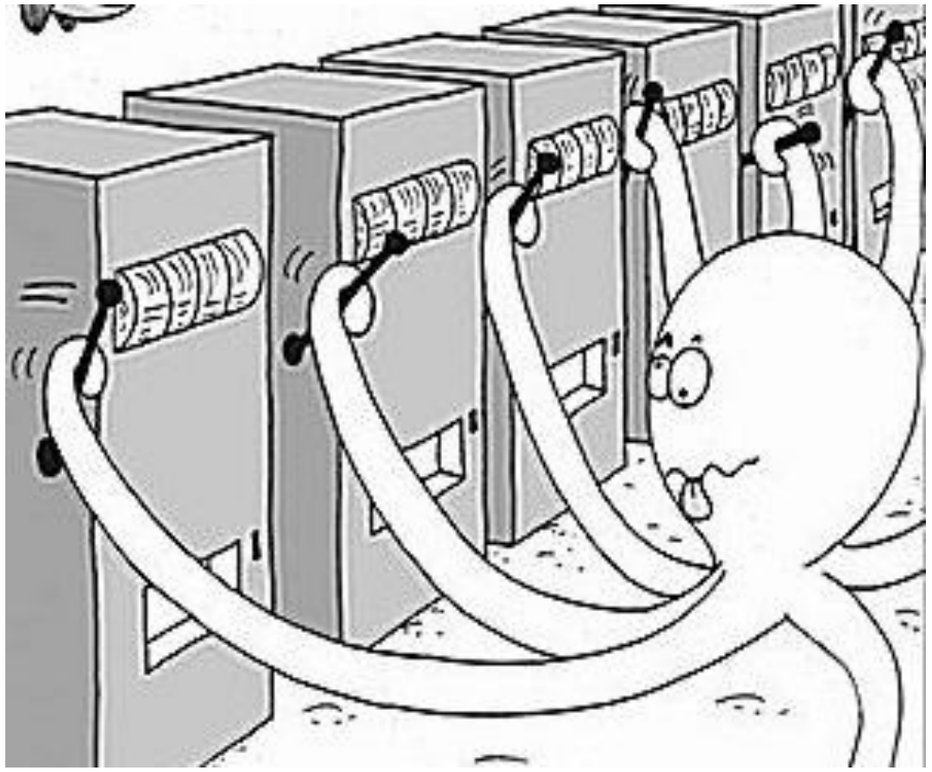
For round $t = 1, 2, \ldots,$ **the learner**

- Given x_i, chooses from a set of actions $a_i \in A$
- Receives reward $r^{(t)} \sim R(x^{(t)}, a^{(t)})$ (i.e., can be random)
- Want to maximize total reward
- You generate your own dataset $\{(x^{(t)}, a^{(t)}, r^{(t)})\}$!
  - e.g., for an image, the learner guesses a label, and is told whether correct or not (reward = 1 if correct and 0 otherwise). **Do not know what's the true label.**
  - e.g., for an user, the website recommends a movie, and observes whether the user likes it or not. **Do not know what movies the user really want to see.**
  - "Partial information setting"

Contextual bandits

Contextual Bandits (cont.)

- Simplification: no x, Multi-Armed Bandits (MAB)
- Bandit is a research area by itself. we will not do a lot of bandits but may go through some material that have important implications on general RL (e.g., lower bounds)



RL

For round $t = 1, 2, \ldots,$

- For time step h=1, 2, …, H, the learner
  - Observes $x_h^{(t)}$
  - Chooses $a_h^{(t)}$
  - Receives $r_h^{(t)} \sim R(x_h^{(t)}, a_h^{(t)})$
  - Next $x_{h+1}^{(t)}$ is generated as a function of $x_h^{(t)}$ and $a_h^{(t)}$ (or sometimes, all previous x's and a's within round t)
- Bandits + "Delayed rewards/consequences"
- The protocol here is for episodic RL (each t is an *episode*).

Why statistical RL?

Two types of scenarios in RL research

1. Solving a large planning problem using a learning approach
   - e.g., AlphaGo, video game playing, simulated robotics
   - Transition dynamics (Go rules) known, but too many states
   - Run the simulator to collect data
2. Solving a learning problem
   - e.g., adaptive medical treatment
   - Transition dynamics unknown (and too many states)
   - Interact with the environment to collect data

Why statistical RL?

Two types of scenarios in RL research

1. Solving a large planning problem using a learning approach
2. Solving a learning problem

- I am more interested in #2. More challenging in some aspects.
- Data (real-world interactions) is highest priority. Computation second.
- Even for #1, sample complexity lower-bounds computational complexity — statistical-first approach is reasonable.
  - caveat to this argument: you can do a lot more in a simulator; see http://hunch.net/?p=8825714

MDP Planning

Infinite-horizon discounted MDPs

An MDP $M = (S, A, P, R, \gamma)$

- State space S.
- Action space A.

We will only consider discrete and finite spaces in this course.

- Transition function $P : S \times A \rightarrow \Delta(S)$. $\Delta(S)$ is the probability simplex over S, i.e., all non-negative vectors of length $|S|$ that sums up to 1
- Reward function R: $S \times A \rightarrow \mathbb{R}$. (deterministic reward function)
- Discount factor $\gamma \in [0,1)$
- The agent starts in some state s_1, takes action a_1, receives reward $r_2 \sim R(s_1, a_1)$, transitions to $s_2 \sim P(s_1, a_1)$, takes action a_2, so on so forth — the process continues indefinitely

Value and policy

- Want to take actions in a way that maximizes value (or return):

$$\mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t\right]$$

  - This value depends on where you start and how you act
- Often assume boundedness of rewards: $r_t \in [0, R_{\max}]$
  - What's the range of $\mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t\right]$? $\left[0, \frac{R_{\max}}{1-\gamma}\right]$
- A (deterministic) policy π: $S \rightarrow A$ describes how the agent acts: at state s_t, chooses action $a_t = \pi(s_t)$.
- More generally, the agent may choose actions randomly (π: $S \rightarrow \Delta(A)$), or even depending the action choice on the time step t ("non-stationary policies")
- Define $$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \,\middle|\, s_1 = s, \pi\right]$$

Bellman equation for policy evaluation

$$
\begin{aligned}
V^\pi(s) &= \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \;\middle|\, s_1 = s, \pi\right] \\
&= \mathbb{E}\left[r_1 + \sum_{t=2}^{\infty} \gamma^{t-1} r_t \;\middle|\, s_1 = s, \pi\right] \\
&= R(s, \pi(s)) + \sum_{s' \in \mathcal{S}} P(s'|s, \pi(s))\, \mathbb{E}\left[\gamma \sum_{t=2}^{\infty} \gamma^{t-2} r_t \;\middle|\, s_1 = s, s_2 = s', \pi\right] \\
&= R(s, \pi(s)) + \sum_{s' \in \mathcal{S}} P(s'|s, \pi(s))\, \mathbb{E}\left[\gamma \sum_{t=2}^{\infty} \gamma^{t-2} r_t \;\middle|\, s_2 = s', \pi\right] \\
&= R(s, \pi(s)) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s, \pi(s))\, \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \;\middle|\, s_1 = s', \pi\right] \\
&= R(s, \pi(s)) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s, \pi(s))\, V^\pi(s') \\
&= R(s, \pi(s)) + \gamma \langle P(\cdot|s, \pi(s)),\, V^\pi(\cdot)\rangle
\end{aligned}
$$

Bellman equation for policy evaluation

$$V^\pi(s) = R(s, \pi(s)) + \gamma \langle P(\,\cdot\,|\, s, \pi(s)), V^\pi(\,\cdot\,)\rangle$$

Matrix form: define

- V^π as the $|S|\times 1$ vector $[V^\pi(s)]_{s\in S}$
- R^π as the vector $[R(s, \pi(s))]_{s\in S}$
- P^π as the matrix $[P(s' \,|\, s, \pi(s))]_{s\in S,\, s'\in S}$

$$V^\pi = R^\pi + \gamma P^\pi V^\pi$$

$$(I - \gamma P^\pi) V^\pi = R^\pi$$

$$V^\pi = (I - \gamma P^\pi)^{-1} R^\pi$$

This is always invertible. Proof?

State occupancy

$$(I - \gamma P^{\pi})^{-1}$$

Each row (indexed by s) is the discounted state occupancy $d_s{}^{\pi}$, whose (s')-th entry is

$$d_s^{\pi}(s') = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1}\ \mathbb{I}[s_t = s'] \,\middle|\, s_1 = s, \pi\right]$$

- Each row is like a distribution vector—except that the entries sum up to $1/(1\text{-}\gamma)$. Let $\eta_s^{\pi} = (1-\gamma)\ d_s^{\pi}$ denote the normalized vector.
- V^{π}(s) is the dot product between d_s^{π} and reward vector
- Can also be interpreted as the value function of indicator reward function

Optimality

- For infinite-horizon discounted MDPs, there always exists a stationary and deterministic policy that is optimal for all starting states simultaneously
  - Proof: Puterman'94, Thm 6.2.7 (reference due to Shipra Agrawal)
- Let π^* denote this optimal policy, and $V^* := V^{\pi^*}$
- Bellman Optimality Equation:

$$V^\star(s) = \max_{a \in A} \left(\underline{R(s,a) + \gamma \mathbb{E}_{s' \sim P(s,a)} \left[V^\star(s') \right]} \right)$$

- If we know V^*, how to get π^* ?
- Easier to work with Q-values: $Q^*(s, a)$, as $\pi^\star(s) = \arg\max_{a \in A} Q^\star(s,a)$

$$Q^\star(s,a) = R(s,a) + \gamma \mathbb{E}_{s' \sim P(s,a)} \left[\max_{a' \in A} Q^\star(s',a') \right]$$

Ad Hoc Homework 1

- uploaded on course website
- help understand the relationships between alternative MDP formulations
- more like readings w/ questions to think about
- no need to submit
- will go through in class next week