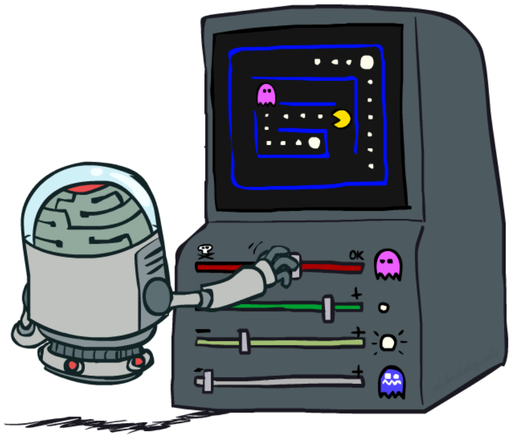
CSE 473: Artificial Intelligence

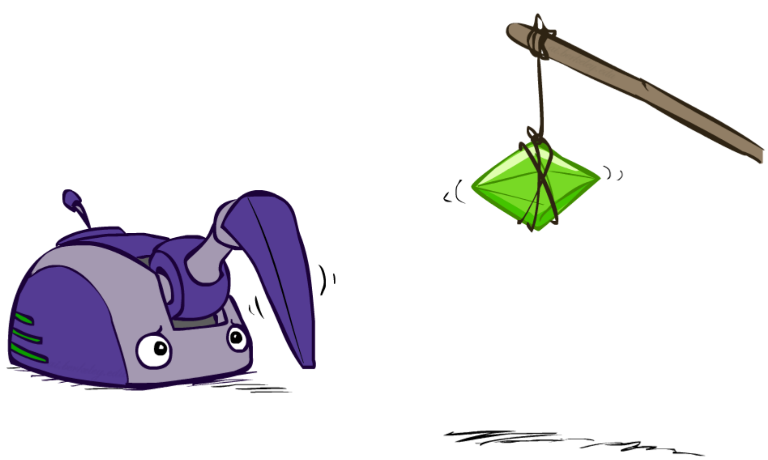
Reinforcement Learning



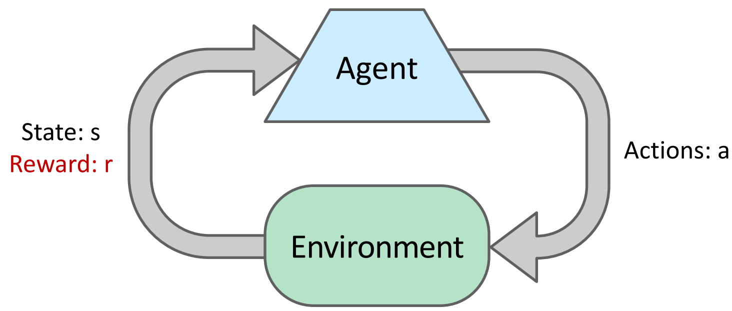
Dan Weld/ University of Washington

[Many slides taken from Dan Klein and Pieter Abbeel / CS188 Intro to AI at UC Berkeley – materials available at http://ai.berkeley.edu.]

Reinforcement Learning

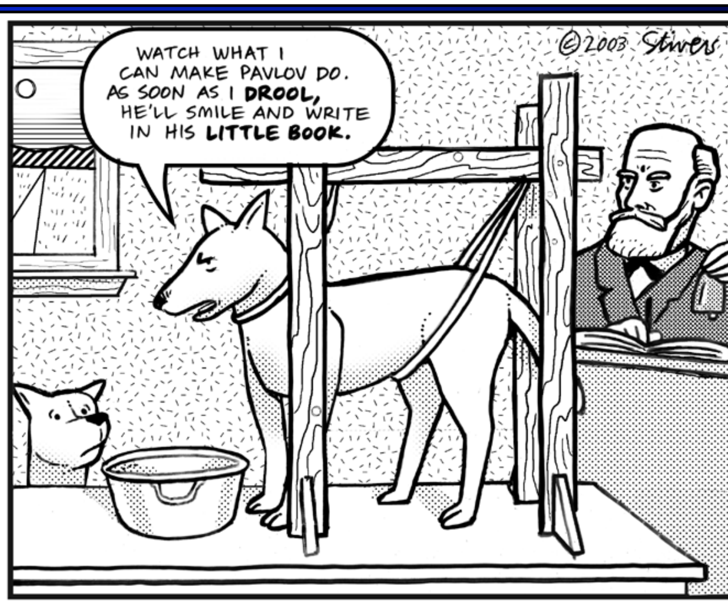


Reinforcement Learning



- Basic idea:
  - Receive feedback in the form of rewards
  - Agent's utility is defined by the reward function
  - Must (learn to) act so as to maximize expected rewards
  - All learning is based on observed samples of outcomes!

Example 2 – More Animal Learning

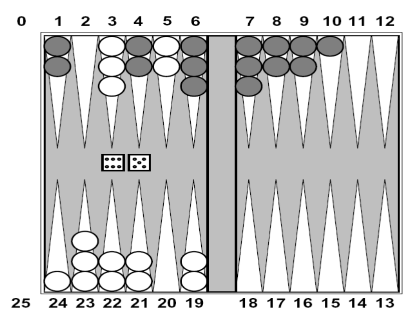


Example: Animal Learning

- RL studied experimentally for more than 60 years in psychology
  - Rewards: food, pain, hunger, drugs, etc.
  - Mechanisms and sophistication debated
- Example: foraging
  - Bees learn near-optimal foraging plan in field of artificial flowers with controlled nectar supplies
  - Bees have a direct neural connection from nectar intake measurement to motor planning area

Example: Backgammon

- Reward only for win / loss in terminal states, zero otherwise
- TD-Gammon learns a function approximation to V(s) using a neural network
- Combined with depth 3 search, one of the top 3 players in the world
- You could imagine training Pacman this way…
- … but it's tricky! (It's also PS 3)



Example: Learning to Walk

Initial

[Kohl and Stone, ICRA 2004]

[Video: AIBO WALK – initial]

Example: Learning to Walk

Finished

[Kohl and Stone, ICRA 2004]

[Video: AIBO WALK – finished]

Example: Sidewinding

[Andrew Ng]

[Video: SNAKE – climbStep+sidewinding]

Video of Demo Crawler Bot

More demos at: http://inst.eecs.berkeley.edu/~ee128/fa11/videos.html

"Few driving tasks are as intimidating as parallel parking….

https://www.youtube.com/watch?v=pB_iFY2jIdI

12

Parallel Parking

"Few driving tasks are as intimidating as parallel parking….

https://www.youtube.com/watch?v=pB_iFY2jIdI

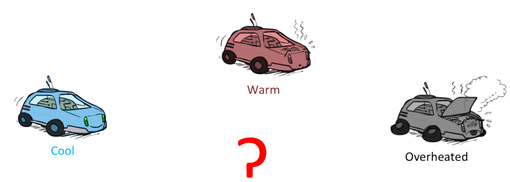
13

Other Applications

- Go playing
- Robotic control
  - helicopter maneuvering, autonomous vehicles
  - Mars rover - path planning, oversubscription planning
  - elevator planning
- Game playing - backgammon, tetris, checkers
- Neuroscience
- Computational Finance, Sequential Auctions
- Assisting elderly in simple tasks
- Spoken dialog management
- Communication Networks – switching, routing, flow control
- War planning, evacuation planning

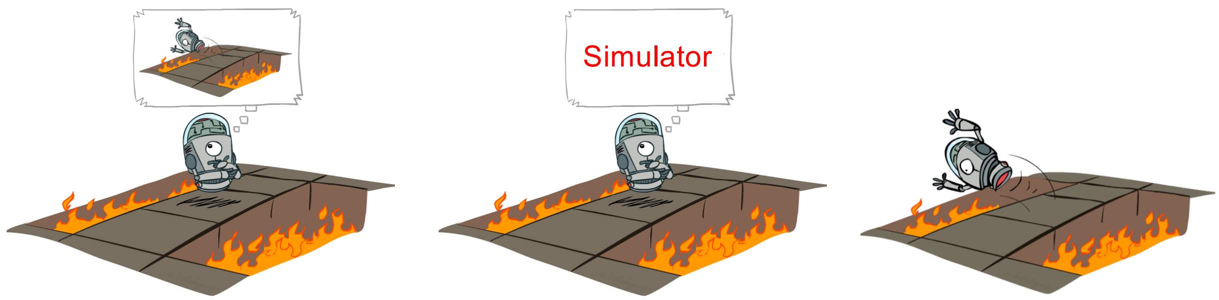
Reinforcement Learning

- Still assume a Markov decision process (MDP):
  - A set of states $s \in S$
  - A set of actions (per state) A
  - A model $T(s,a,s')$
  - A reward function $R(s,a,s')$ & discount γ
- Still looking for a policy $\pi(s)$



- New twist: don't know T or R
  - I.e. we don't know which states are good or what the actions do
  - Must actually try actions and states out to learn

Offline (MDPs) vs. Online (RL)



Offline Solution (Planning)

Monte Carlo Planning

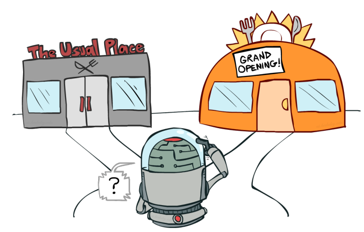
Online Learning (RL)

Three Key Ideas for RL

- Model-based *vs* model-free learning
  - What function is being learned?
- Approximating the Value Function
  - Smaller → easier to learn & better generalization
- Exploration-exploitation tradeoff

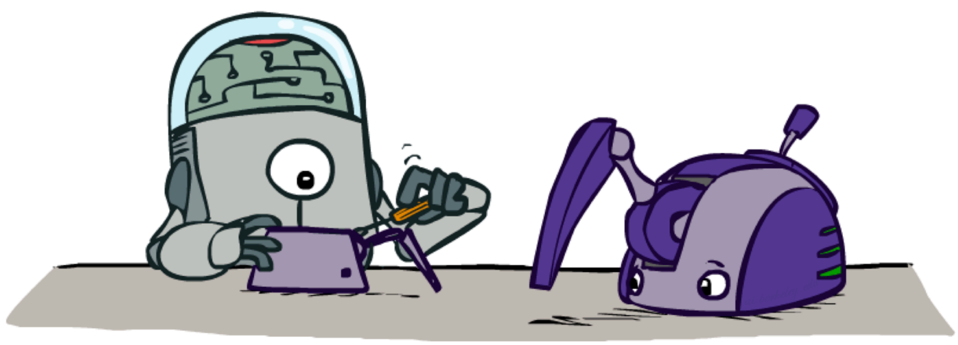
Exploration-Exploitation tradeoff

- You have visited part of the state space and found a reward of 100
  - is this the best you can hope for???
- **Exploitation**: should I stick with what I know and find a good policy w.r.t. this knowledge?
  - at risk of missing out on a better reward somewhere
- **Exploration**: should I look for states w/ more reward?
  - at risk of wasting time & getting some negative reward



18

Model-Based Learning

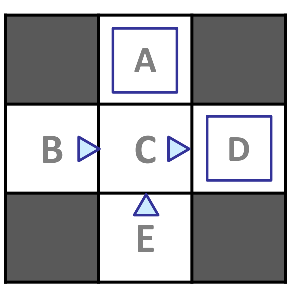


Model-Based Learning

- Model-Based Idea:
  - Learn an approximate model based on experiences
  - Solve for values as if the learned model were correct
- Step 1: Learn empirical MDP model
  - Count outcomes s' for each s, a
  - Normalize to give an estimate of $\widehat{T}(s,a,s')$
  - Discover each $\widehat{R}(s,a,s')$ when we experience (s, a, s')
- Step 2: Solve the learned MDP
  - For example, use value iteration, as before

Example: Model-Based Learning

Random π



Assume: $\gamma = 1$

Observed Episodes (Training)

Episode 1

B, east, C, -1
C, east, D, -1
D, exit, x, +10

Episode 2

B, east, C, -1
C, east, D, -1
D, exit, x, +10

Episode 3

E, north, C, -1
C, east, D, -1
D, exit, x, +10

Episode 4

E, north, C, -1
C, east, A, -1
A, exit, x, -10

Learned Model

$\widehat{T}(s,a,s')$

T(B, east, C) = 1.00
T(C, east, D) = 0.75
T(C, east, A) = 0.25
…

$\widehat{R}(s,a,s')$

R(B, east, C) = -1
R(C, east, D) = -1
R(D, exit, x) = +10
…

Convergence

- If policy explores "enough" – doesn't starve any state
- Then T & R converge

- So, VI, PI, Lao* *etc.* will find optimal policy
  - Using Bellman Equations

- When can agent start exploiting??
  - (We'll answer this question later)

23

Two main reinforcement learning approaches

- Model-based approaches:

  Learn T + R

  $|S|^2|A| + |S||A|$ parameters (40,400)

- Model-free approach:

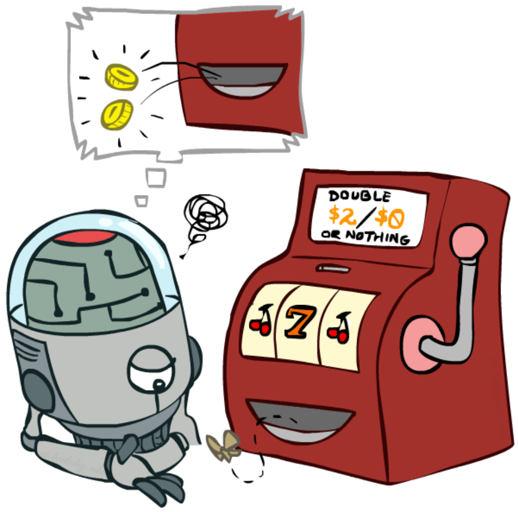
  Learn Q

  $|S||A|$ parameters (400)

Suppose 100 states, 4 actions

24

Model-Free Learning



Reminder: **Q-Value Iteration**

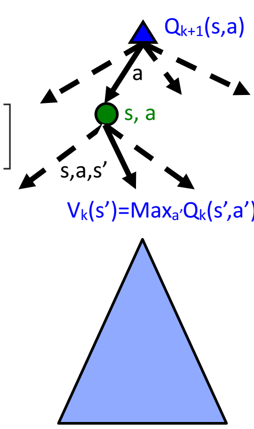
- **Forall s, a**
  - **Initialize Q_0(s, a) = 0** *no time steps left means an expected reward of zero*
- **K = 0**
- **Repeat** *do Bellman backups*

  For every (s,a) pair:

$$Q_{k+1}(s,a) \leftarrow \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma \max_{a'} Q_k(s',a') \right]$$

  K += 1
- **Until convergence** *I.e., Q values*

This is easy….



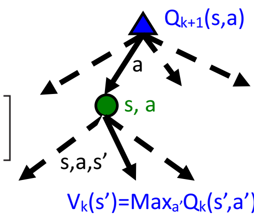
Puzzle: Q-Learning

- **Forall s, a**
  - **Initialize $Q_0(s, a) = 0$** *no time steps left means an expected reward of zero*
- **K = 0**
- **Repeat** *do Bellman backups*

  For every (s,a) pair:

$$Q_{k+1}(s,a) \leftarrow \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma \max_{a'} Q_k(s',a') \right]$$

  K += 1
- **Until convergence**



Q: How can we compute without R, T ?!?

A: Compute averages using sampled outcomes

Simple Example: Expected Age

Goal: Compute expected age of CSE students

Known P(A)

$$E[A] = \sum_a P(a) \cdot a \qquad = 0.35 \times 20 + \ldots$$

Note: never know **P(age=22)**

Without P(A), instead collect samples [a_1, a_2, … a_N]

Unknown P(A): "Model Based"

$$\hat{P}(a) = \frac{\text{num}(a)}{N}$$

$$E[A] \approx \sum_a \hat{P}(a) \cdot a$$

Why does this work? Because eventually you learn the right model.

Unknown P(A): "Model Free"

$$E[A] \approx \frac{1}{N} \sum_i a_i$$

Why does this work? Because samples appear with the right frequencies.

Anytime Model-Free Expected Age

Goal: Compute expected age of CSE students

```
Let A=0
Loop for i = 1 to ∞
    a_i ← ask "what is your age?"
    A ← (1-α)*A + α*a_i
```

Without P(A), instead collect samples $[a_1, a_2, \ldots a_N]$

```
Let A=0
Loop for i = 1 to ∞
    a_i ← ask "what is your age?"
    A ← (i-1)/i * A + (1/i) * a_i
```

Unknown P(A): "Model Free"

$$E[A] \approx \frac{1}{N}\sum_i a_i$$

Exponential Moving Average

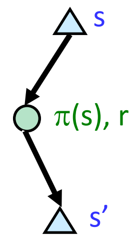
- Exponential moving average
  - The running interpolation update: $\bar{x}_n = (1-\alpha)\cdot\bar{x}_{n-1} + \alpha\cdot x_n$
  - Makes recent samples more important:

$$\bar{x}_n = \frac{x_n + (1-\alpha)\cdot x_{n-1} + (1-\alpha)^2\cdot x_{n-2} + \ldots}{1 + (1-\alpha) + (1-\alpha)^2 + \ldots}$$

  - Forgets about the past (distant past values were wrong anyway)
- Decreasing learning rate (alpha) can give converging averages
  - E.g., $\alpha = 1/i$

Sampling Q-Values

- Big idea: learn from every experience!
  - Follow exploration policy a ← π(s)
  - Update Q(s,a) each time we experience a transition (s, a, s', r)
  - Likely outcomes s' will contribute updates more often



- Update towards running average:

Get a sample of Q(s,a): *sample* = R(s,a,s') + γ $\text{Max}_{a'}$ Q(s', a')

Update to Q(s,a): Q(s,a) ← (1-**α**)Q(s,a) + (**α**)*sample*

Q Learning

- **Forall s, a**
  - **Initialize Q(s, a) = 0**
- **Repeat Forever**

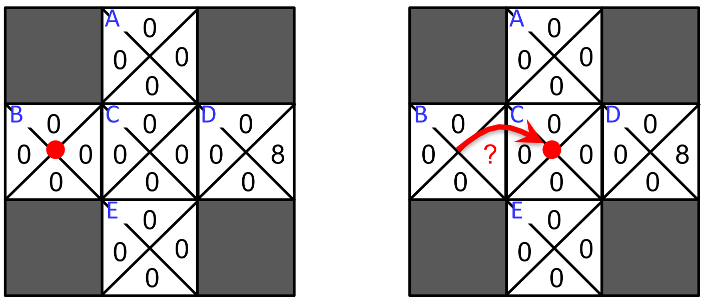
  Where are you? s.
  Choose some action a
  Execute it in real world: *(s, a, r, s')*
  Do update:

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + (\alpha)\left[r + \gamma \max_{a'} Q(s',a')\right]$$

Example

Assume: γ = 1, α = 1/2

Observed Transition: B, east, C, -2



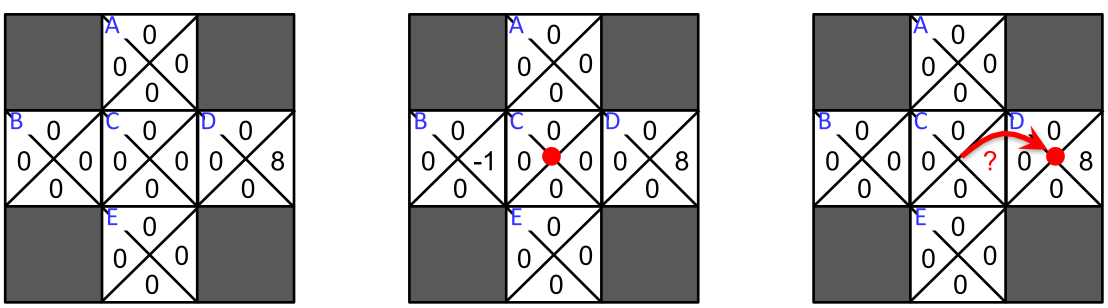
$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + (\alpha)\left[r + \gamma \max_{a'} Q(s',a')\right]$$

-1 ½ 0 ½ -2 0

Example

Assume: γ = 1, α = 1/2

Observed Transition: B, east, C, -2 C, east, D, -2



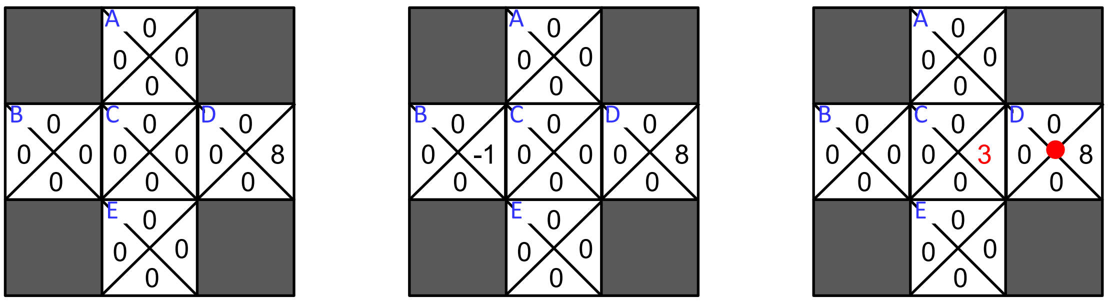
$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + (\alpha)\left[r + \gamma \max_{a'} Q(s',a')\right]$$

3 ½ 0 ½ -2 8

Example

Assume: $\gamma = 1$, $\alpha = 1/2$

Observed Transition: B, east, C, -2   C, east, D, -2



$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + (\alpha)\left[r + \gamma \max_{a'} Q(s',a')\right]$$