Policy

COMP4211

Agent's Learning Task

At time t, state s, follow policy π,

- obtain rewards $r_t, r_{t+1}, \ldots$

Learn action policy π that maximizes the expected future reward

$$V^{\pi}(s) \equiv E[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots]$$

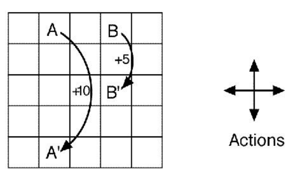
from any starting state in S

- (state) value function: value of a state
- maximize the long-term total discounted reward

Note that the target function is $\pi : S \to A$, but we have no training examples of form $\langle s, a \rangle$ (therefore, not supervised learning)

- training examples are of form $\langle \langle s, a \rangle, r \rangle$

Example

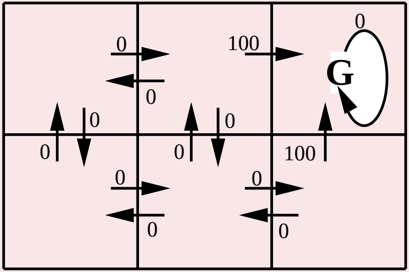


- actions: north, south, east, west (by one cell)
- if would take agent off the grid: no move but reward $= -1$
- other actions produce reward $= 0$, except actions (all four) that move agent out of special states A and B as shown

State-value function for random policy

3.3	8.8	4.4	5.3	1.5
1.5	3.0	2.3	1.9	0.5
0.1	0.7	0.7	0.4	-0.4
-1.0	-0.4	-0.4	-0.6	-1.2
-1.9	-1.3	-1.2	-1.4	-2.0

Example



What should be the “optimal” policy?

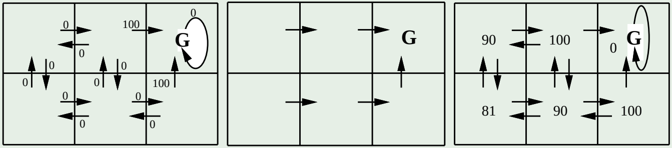
Policy Evaluation

for a given policy π, compute the state value function V^π

- consider first deterministic world
  $V^\pi(s) \equiv E[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots] = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots$

Example

Problem; policy π; value function V^π



e.g., at bottom right state: move up

- discounted future reward: $100 + \gamma \cdot 0 + \gamma^2 \cdot 0 + \cdots = 100$

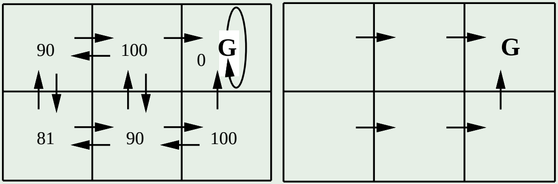
e.g., at bottom center state: move right, then up

- discounted future reward: $(\gamma = 0.9)$
$$0 + \gamma \cdot 100 + \gamma^2 \cdot 0 + \gamma^3 \cdot 0 + \cdots = 90$$

Policy Evaluation...

At state s, (deterministic policy) take action a

Example



- obtain immediate reward $r(s, a)$
- value of the immediate successor state $V^\pi(\delta(s, a))$
- total discounted future reward: $r(s, a) + \gamma V^\pi(\delta(s, a))$

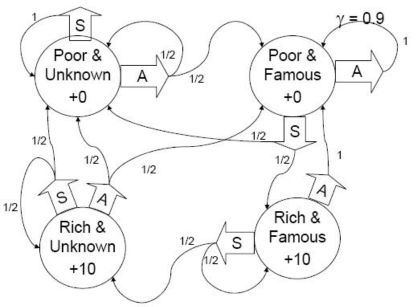
$$V^\pi(s) = r(s, a) + \gamma V^\pi(\delta(s, a)) \qquad \text{(Bellman equation)}$$

Solve a linear system involving $V^\pi(s_1), V^\pi(s_2), \ldots$

Policy Evaluation...

In nondeterministic worlds:

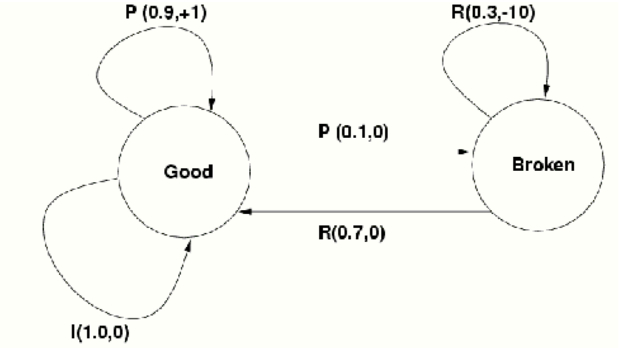
Example



- at state s, take action a with probability $\pi(s, a)$
- from s, take action a, probability of transition to s': $P(s, s', a)$
- expected reward on transition s to s' given action a: $R(s, s', a)$

$$V^{\pi}(s) = \sum_a \pi(s, a) \sum_{s'} P(s, s', a)[R(s, s', a) + \gamma V^{\pi}(s')]$$

Manufacturing Example (Non-deterministic)



"good" state:

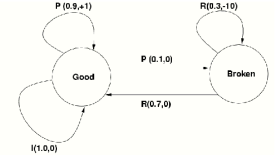
- actions: "produce" / "inactive"

"broken" state:

- action: "repair"

Manufacturing Example (Non-deterministic)...

Policy: always "produce" in the "good" state



value at "good": $V^\pi(g)$; value at "broken": $V^\pi(b)$

"good" state:

- "produce" and move to "good": $1 + \gamma V^\pi(g)$
- "produce" and move to "broken": $0 + \gamma V^\pi(b)$

$$V^\pi(g) = (0.9)(1 + \gamma V^\pi(g)) + (0.1)(0 + \gamma V^\pi(b))$$

"broken" state:

- "repair" and move to "good": $0 + \gamma V^\pi(g)$
- "repair" and move to "broken": $-10 + \gamma V^\pi(b)$

$$V^\pi(b) = (0.7)(0 + \gamma V^\pi(g)) + (0.3)(-10 + \gamma V^\pi(b))$$

For $\gamma = 0.5$, we can solve these two equations (system of linear equations) to get $V^\pi(g) = 1.36$, $V^\pi(b) = -2.97$

Solving a Markov System as a Linear System

Upside: You get an exact answer
Downside: If you have 1,000,000 states, you're solving 1,000,000 equations with 1,000,000 unknowns

Let's do Iteration

$$V^{\pi}(s) = \sum_{a} \pi(s,a) \sum_{s'} P(s,s',a)[R(s,s',a) + \gamma V^{\pi}(s')]$$

Instead of solving the linear system, we can also use an iterative method

- initialize V_0
- $V_0 \to V_1 \to V_2 \to \cdots V_k \to V_{k+1} \cdots \to V^{\pi}$

Update rule:

$$V_{k+1}(s) \leftarrow \sum_{a} \pi(s,a) \sum_{s'} P(s,s',a)[R(s,s',a) + \gamma V_k(s')]$$

- iterative policy evaluation

Input π, the policy to be evaluated

Initialize $V(s) = 0$, for all $s \in \mathcal{S}^+$

Repeat

$\quad \Delta \leftarrow 0$

$\quad$ For each $s \in \mathcal{S}$:

$\quad\quad v \leftarrow V(s)$

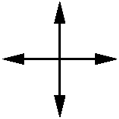
$\quad\quad V(s) \leftarrow \sum_a \pi(s,a) \sum_{s'} \mathcal{P}^a_{ss'} \left[\mathcal{R}^a_{ss'} + \gamma V(s')\right]$

$\quad\quad \Delta \leftarrow \max(\Delta, |v - V(s)|)$

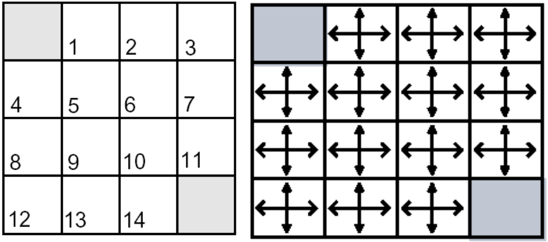
until $\Delta < \theta$ (a small positive number)

Output $V \approx V^\pi$

Example



actions



- terminal states: shaded squares
- reward: -1
- actions that would take agent off the grid leave state unchanged

Example: Random Policy

$k = 0$

0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0

$k = 3$

0.0	-2.4	-2.9	-3.0
-2.4	-2.9	-3.0	-2.9
-2.9	-3.0	-2.9	-2.4
-3.0	-2.9	-2.4	0.0

$k = 1$

0.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	0.0

$k = 10$

0.0	-6.1	-8.4	-9.0
-6.1	-7.7	-8.4	-8.4
-8.4	-8.4	-7.7	-6.1
-9.0	-8.4	-6.1	0.0

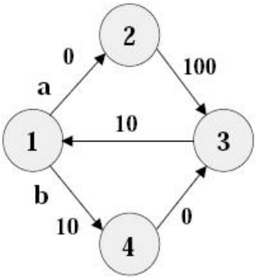
$k = 2$

0.0	-1.7	-2.0	-2.0
-1.7	-2.0	-2.0	-2.0
-2.0	-2.0	-2.0	-1.7
-2.0	-2.0	-1.7	0.0

$k = \infty$

0.0	-14.	-20.	-22.
-14.	-18.	-20.	-20.
-20.	-20.	-18.	-14.
-22.	-20.	-14.	0.0

Example with Two Policies



$$
\begin{aligned}
V^{\pi_a}(1) &= 0 + \gamma V^{\pi_a}(2) \\
V^{\pi_a}(2) &= 100 + \gamma V^{\pi_a}(3) \\
V^{\pi_a}(3) &= 10 + \gamma V^{\pi_a}(1) \\
V^{\pi_a}(1) &= \gamma(100 + \gamma(10 + \gamma V^{\pi_a}(1))) \\
&= 100\gamma + 10\gamma^2 + \gamma^3 V^{\pi_a}(1) \\
V^{\pi_a}(1) &= \frac{100\gamma + 10\gamma^2}{1 - \gamma^3}
\end{aligned}
$$

$$
\begin{aligned}
V^{\pi_b}(1) &= 10 + \gamma V^{\pi_b}(4) \\
V^{\pi_b}(4) &= 0 + \gamma V^{\pi_b}(3) \\
V^{\pi_b}(3) &= 10 + \gamma V^{\pi_b}(1) \\
V^{\pi_b}(1) &= 10 + \gamma(\gamma(10 + \gamma V^{\pi_b}(1))) = \frac{10 + 10\gamma^2}{1 - \gamma^3}
\end{aligned}
$$

If $\frac{100\gamma+10\gamma^2}{1-\gamma^3} \geq \frac{10+10\gamma^2}{1-\gamma^3}$ (i.e., $\gamma \geq 0.1$), then policy a is better