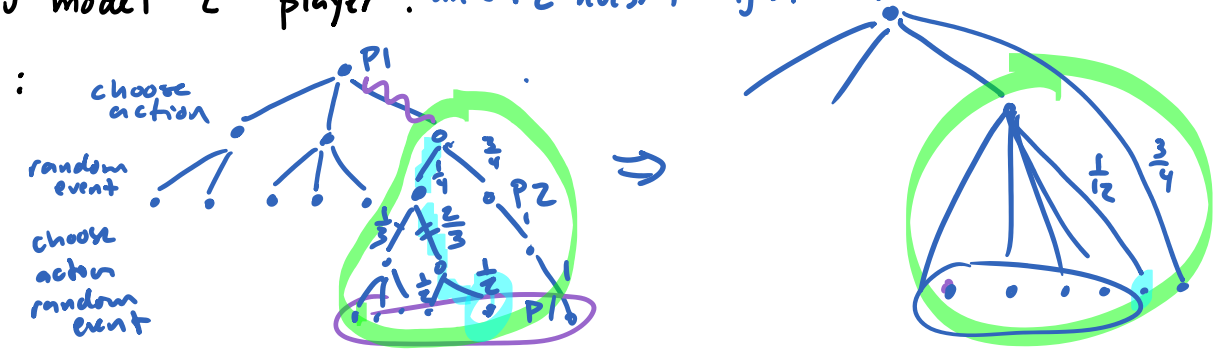
$p(s', r \mid s, a)$ satisfies $p(s', r \mid s, a) = 0$ if s' nonterminal and $r \neq 0$
or s' is P1 win and $r \neq 1$
P2 win and $r \neq -1$
s' is draw and $r \neq 0$

Can you model 2nd player? and P2 doesn't adjust to P1

If so:



If not: for zero-sum game (or constant-sum)

expected reward for P1

$$v^*(s) = \max_a \sum_{s',r} p(s', r \mid s, a) \cdot [r + \gamma \cdot v^*(s')]$$ if P1's turn

$$v^*(s) = \min_a \sum_{s',r} p(s', r \mid s, a) \cdot [r + \gamma \cdot v^*(s')]$$ if P2's turn

finite-time → solve w/ dynamic programming

Two-player Yahtzee

anchors:

Two-player Pig

$v(x,y,p) = 1 - v(y,x,1-p)$

P1 score, P2 score, current turn

$v(x,y,t,n)$ = expected wins for next player with score x-y, turn total t

middle of turn; turn total, # of turns remaining

$v(x,y,n)$ = expected wins for next player with score x-y at start of turn

start of turn

$$v(x,y,n) = \begin{cases} 1.0 & \text{if } x \geq 100 \\ 0.0 & \text{if } y \geq 100 \\ 0.5 & \text{if } n = 0 \\ \sum_{r=2}^{6} \frac{1}{6} \cdot v(x,y,r,n) + \frac{1}{6} \cdot (1 - v(y,x,n-1)) \end{cases}$$

$$v(x,y,t,n) = \max\left(1 - v(y,x+t,n-1), \sum_{r=2}^{6} \frac{1}{6} \cdot v(x,y,t+r,n-1) + \frac{1}{6}(1 - v(y,x))\right)$$

end turn

Evaluating a Policy

$$v_\pi(s) = \sum_a \pi(a|s) \cdot \sum_{s'} \sum_r p(s', r | s, a) \cdot (r + \gamma v_\pi(s'))$$

$$= \sum_{s'} \sum_r p(s', r | s, \pi(s)) \cdot (r + \gamma v_\pi(s'))$$

for deterministic policy π

current estimate of $v_\pi(s)$

Initialize $v[s] \leftarrow 0$ for terminal and arbitrarily for other

← or use (during policy iteration) v_π for prev π

$\Delta \leftarrow 0$

repeat

for each state s (in any order)

$v_{old} \leftarrow v[s]$

let $v[s] \leftarrow$

$\Delta \leftarrow \max(\Delta, |v[s] - v_{old}|)$

until Δ small enough

Policy Iteration

a_2 a_1 a_4 ...
$\pi[0]$ $\pi[1]$ $\pi[2]$

1000000 states
4 actions
$4^{1000000}$ policies

Initialize $\pi[s]$ arbitrarily

repeat
evaluate π to get v_π (prev. page)

change $\leftarrow$ false
for each state s
$a_{old} \leftarrow \pi[s]$

$q_{\pi_{old}}(s,a)$

$\pi[s] \leftarrow \arg\max_a \sum_{s'} \sum_r p(s',r|s,a) \cdot (r + \gamma \cdot v_\pi[s'])$

incrementally improves policy

if change $\leftarrow$ false and $\pi[s] \neq a_{old}$
change $\leftarrow$ true

until change = false

Value Iteration

Initialize $v[s] \leftarrow 0$ for terminal s
$\leftarrow$ arbitrary for nonterminal s

repeat
$\Delta \leftarrow 0$
for each state s

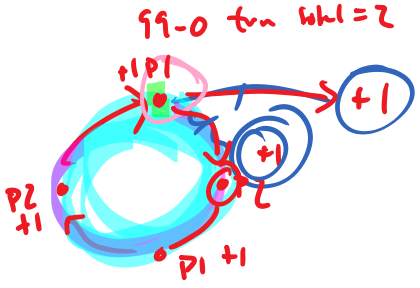
$v_{old} \leftarrow v[s]$

$$v[s] \leftarrow \max_a \sum_{s'} \sum_r p(s', r \mid s, a) \cdot (r + \gamma v[s'])$$

$\Delta \leftarrow \max(\Delta, |v[s] - v_{old}|)$

until Δ small enough

for each s
$$\pi[s] \leftarrow \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r \mid s, a) \cdot (r + \gamma \cdot v[s'])$$



partition and value iteration within each partition by strongly connected components in order of reverse topo sort of SCCs