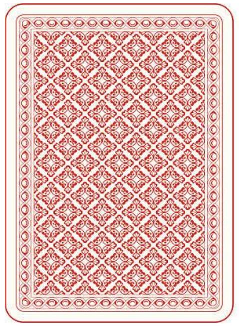
Are we still playing games?

Dealer

Player

Dealer



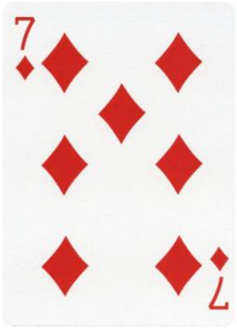
Player

Dealer

Player

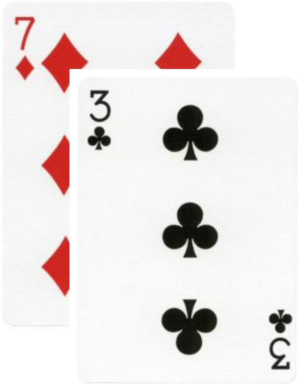
Dealer

Player



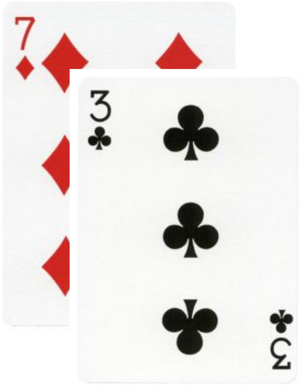
Dealer

Player



Dealer

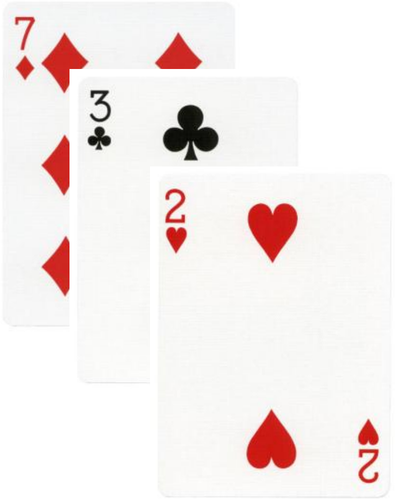
Player



Hit or Stay?

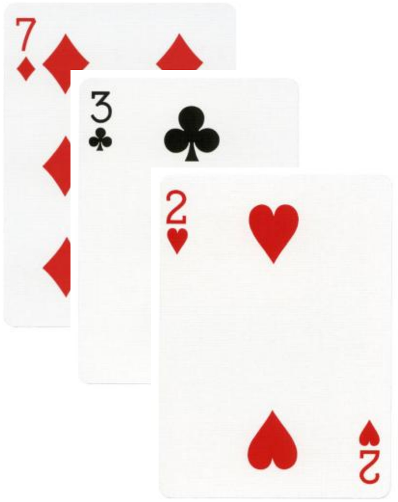
Dealer

Player



Dealer

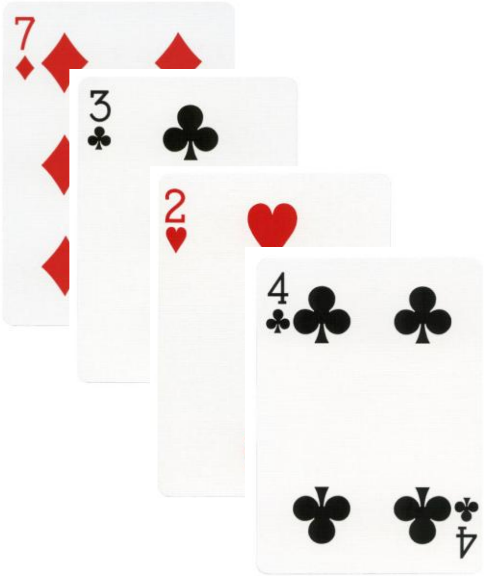
Player



Hit or Stay?

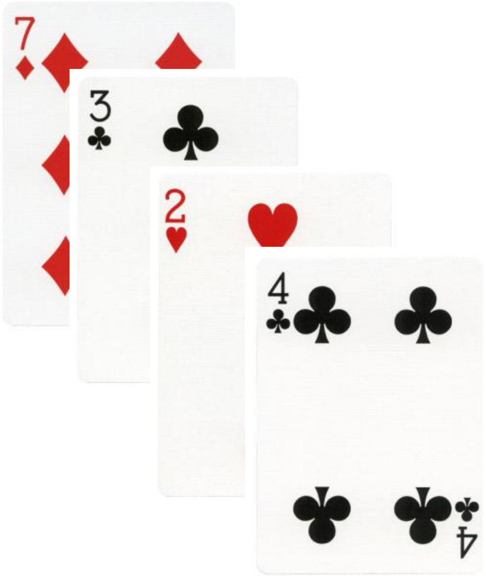
Dealer

Player



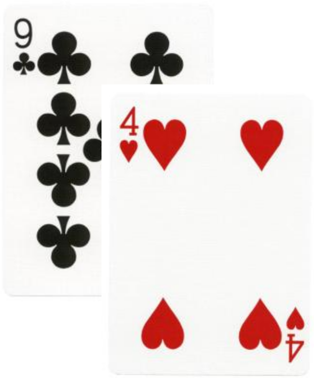
Dealer

Player

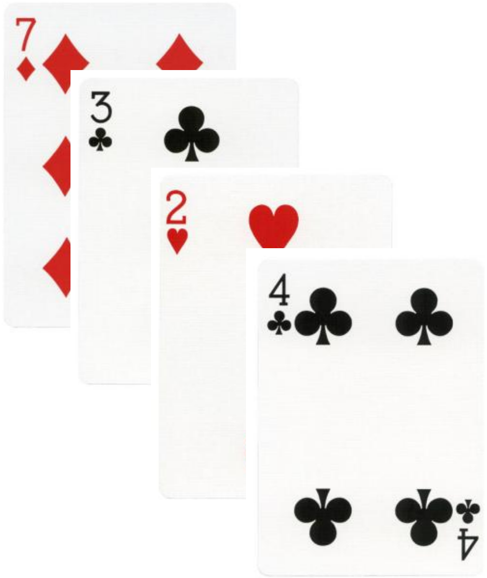


Hit or Stay?

Dealer

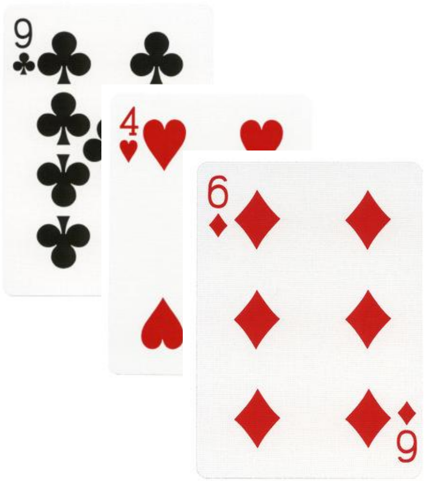


Player

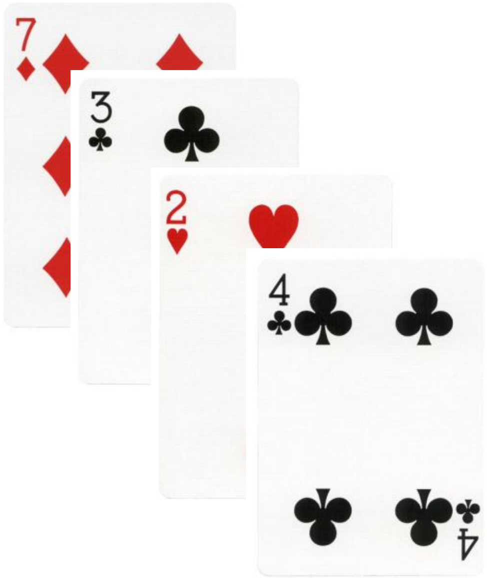


Total points: 16

Dealer

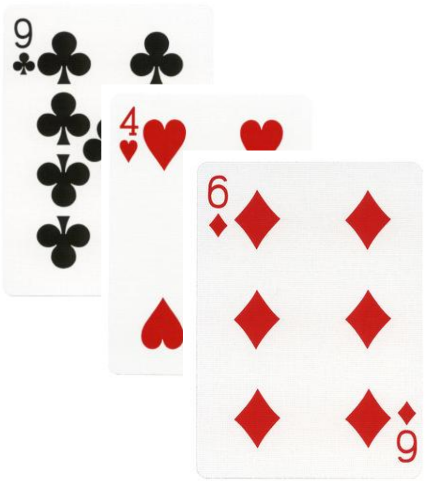


Player



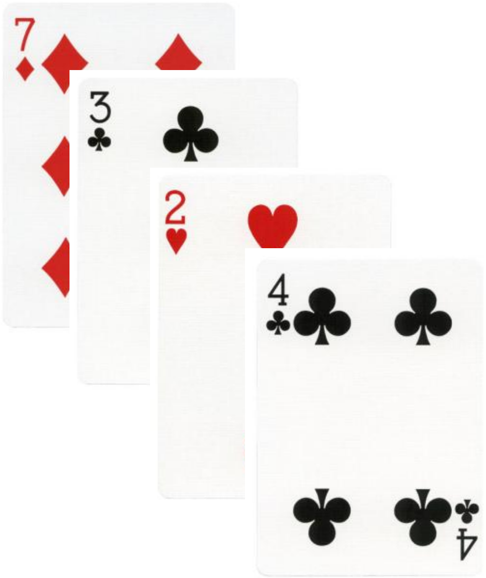
Total points: 16

Dealer



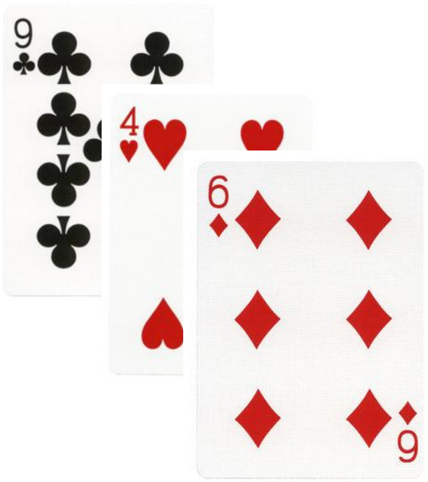
Total points: 19

Player



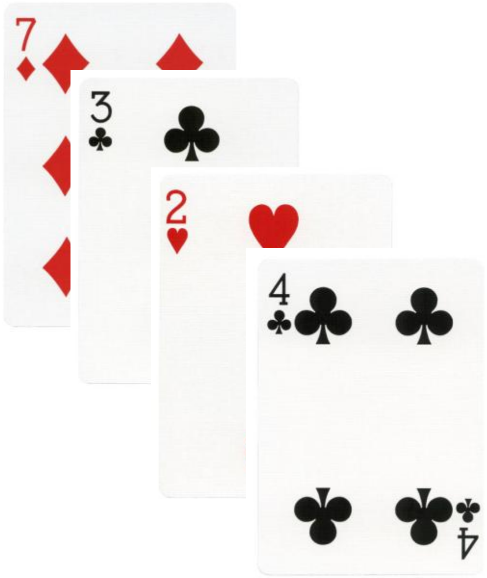
Total points: 16

Dealer



Total points: 19

Player



Total points: 16

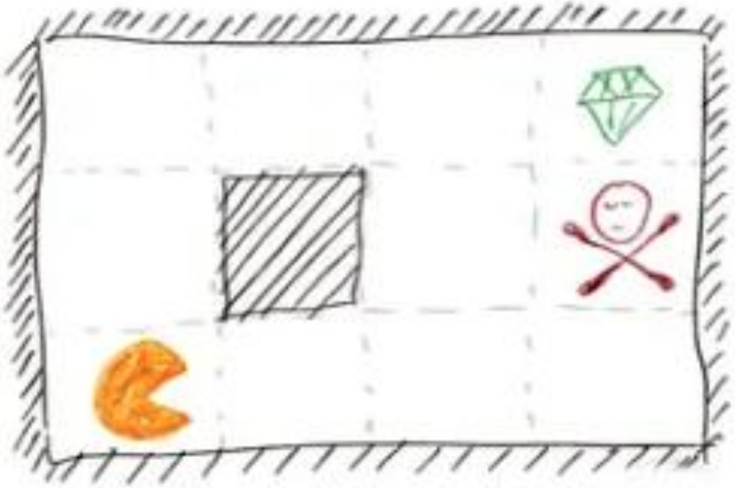
Dealer Wins.

How is this different?

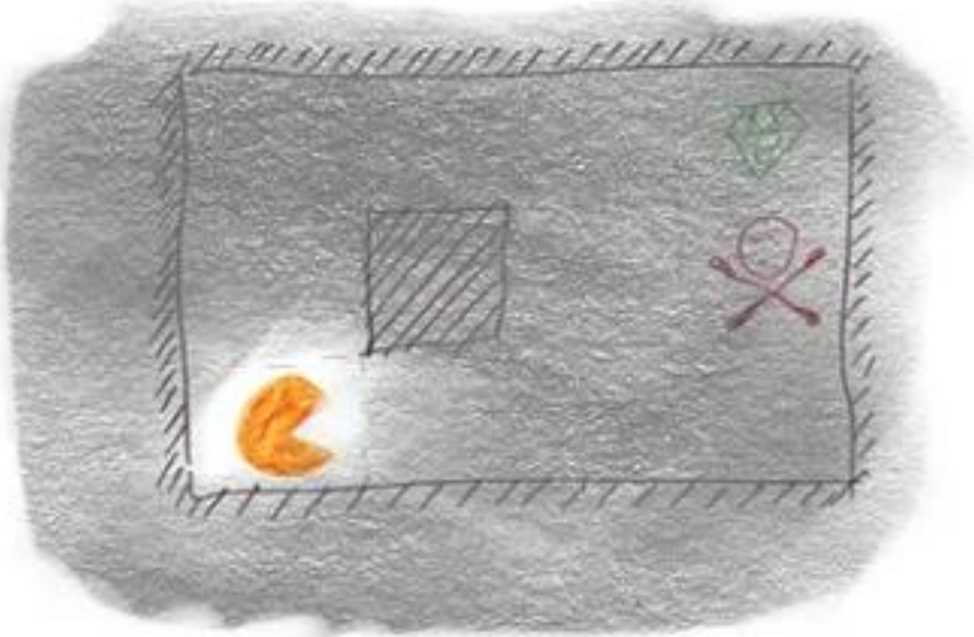
I can be somewhere in a region but **where am I?**

A Markov Decision Process (MDP) is a sequential decision process for a fully observable, stochastic environment with a Markovian transition model and additive rewards.

Good old MDP



Les Miserables Real life

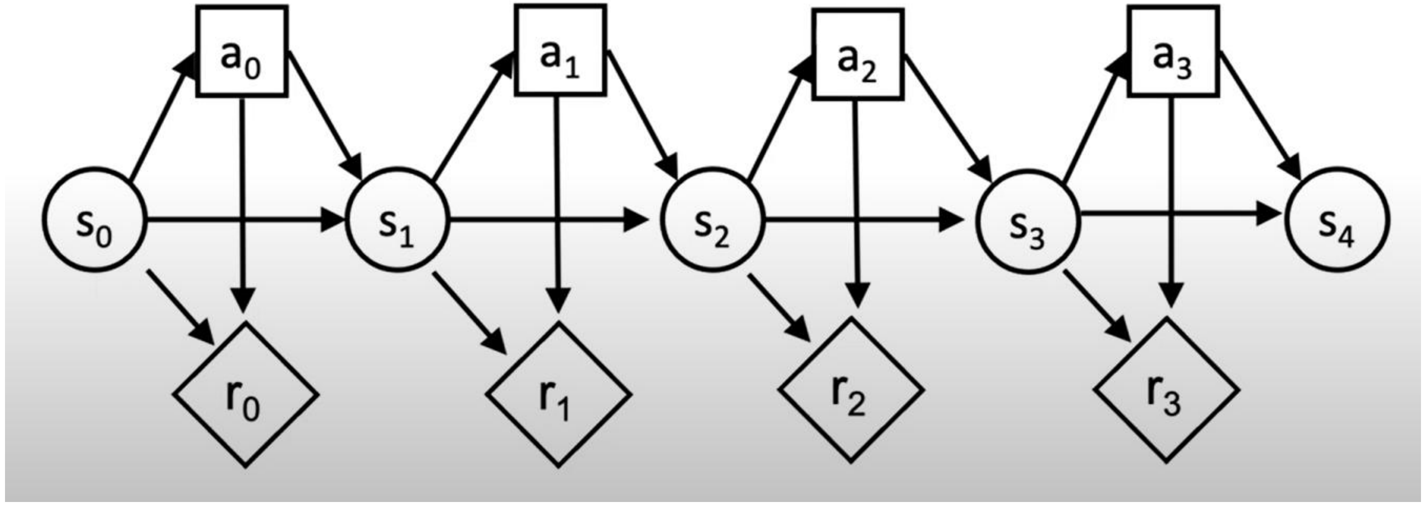


Partially Observable Markov Decision Process (POMDP) is a <u>generalization of a MDP</u> but **does not assume that the state is fully observable**.

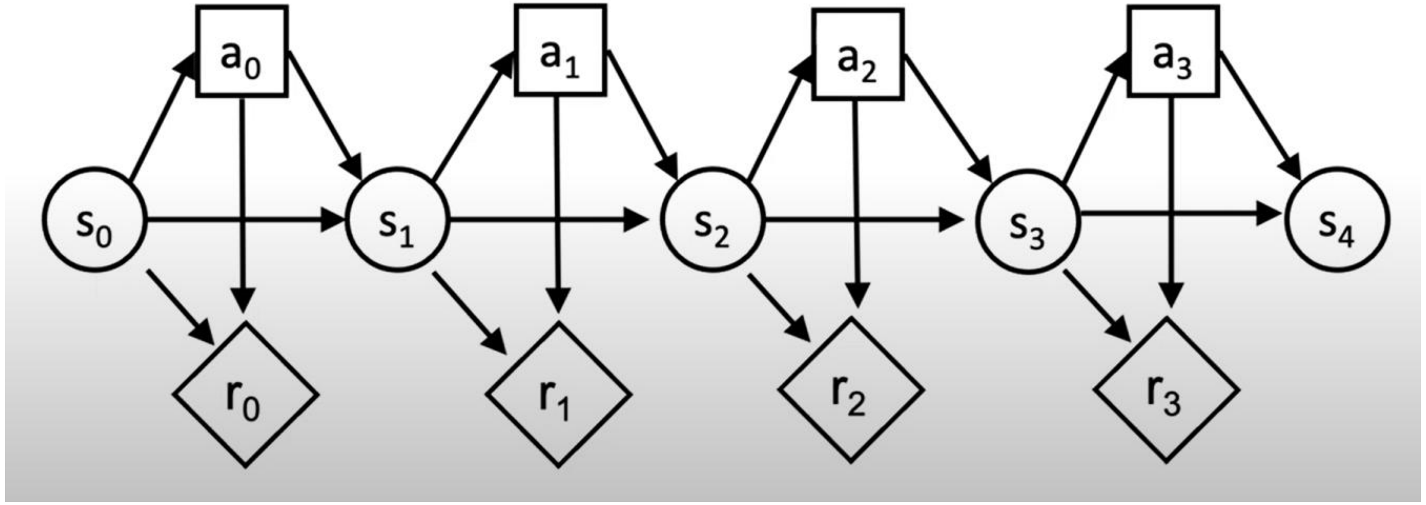
Let’s formalise this

(We are not playing games anymore)

Markov Decision Processes

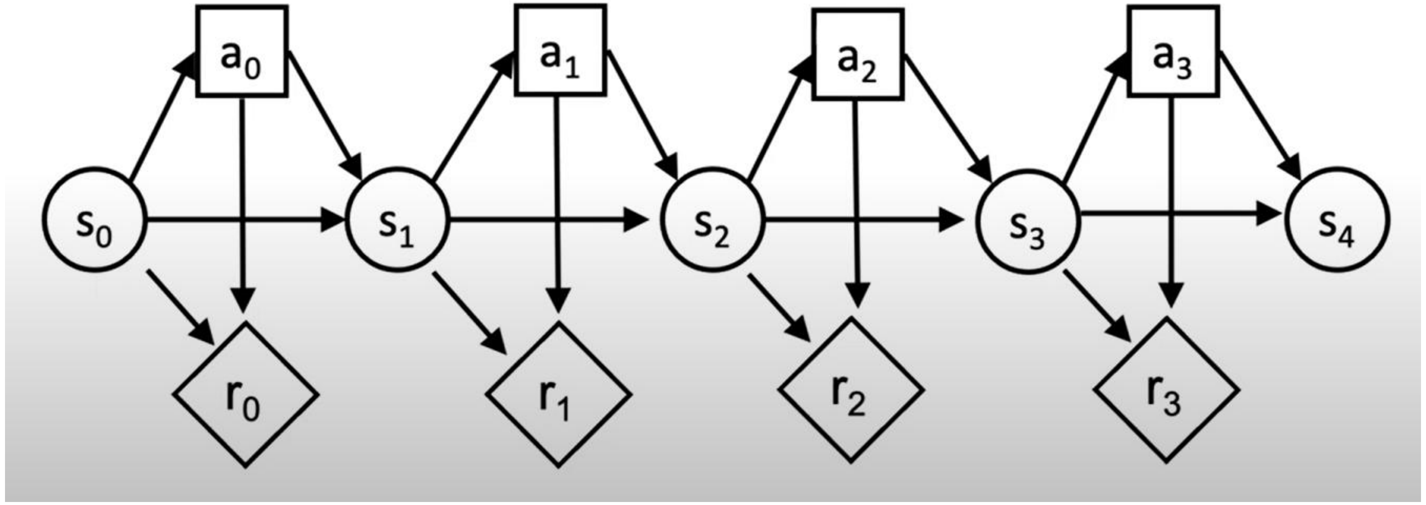


Markov Decision Processes



$$MDP \coloneqq (S, A, P, R, \gamma)$$

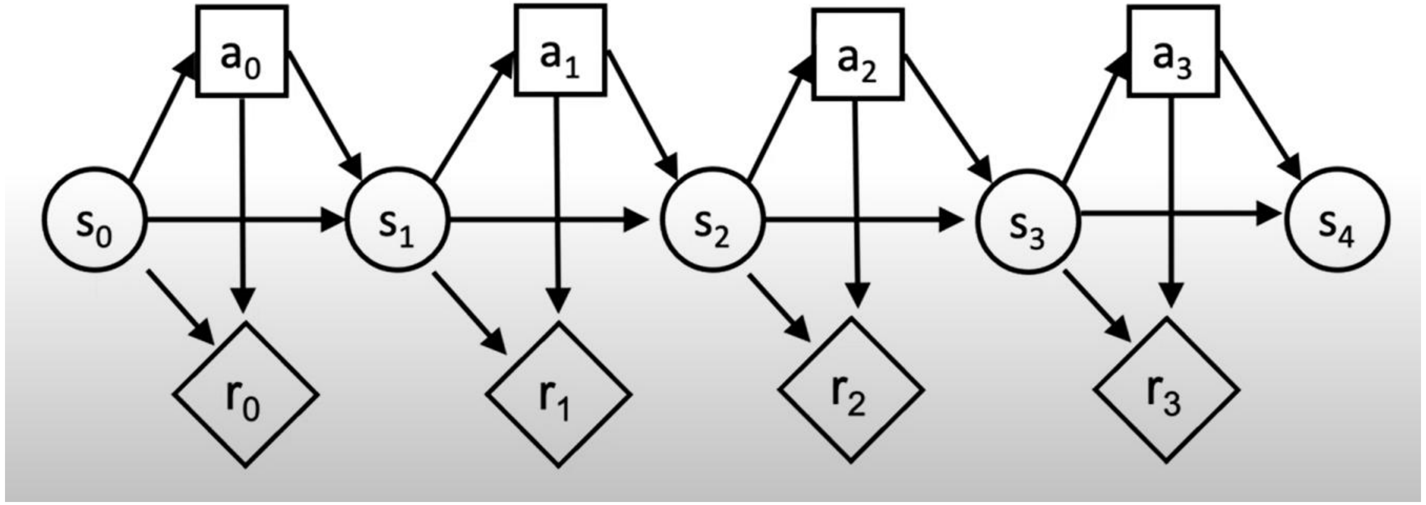
Markov Decision Processes



$$MDP \coloneqq (S, A, P, R, \gamma)$$

State
Space

Markov Decision Processes

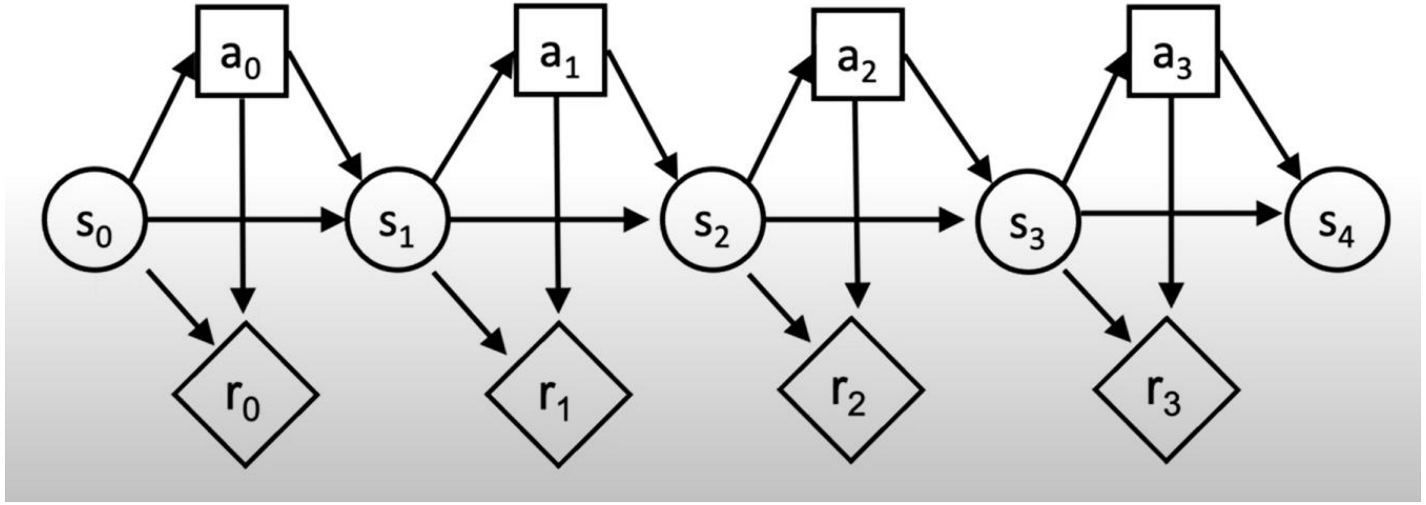


$$MDP \coloneqq (S, A, P, R, \gamma)$$

State Space

Action Space

Markov Decision Processes



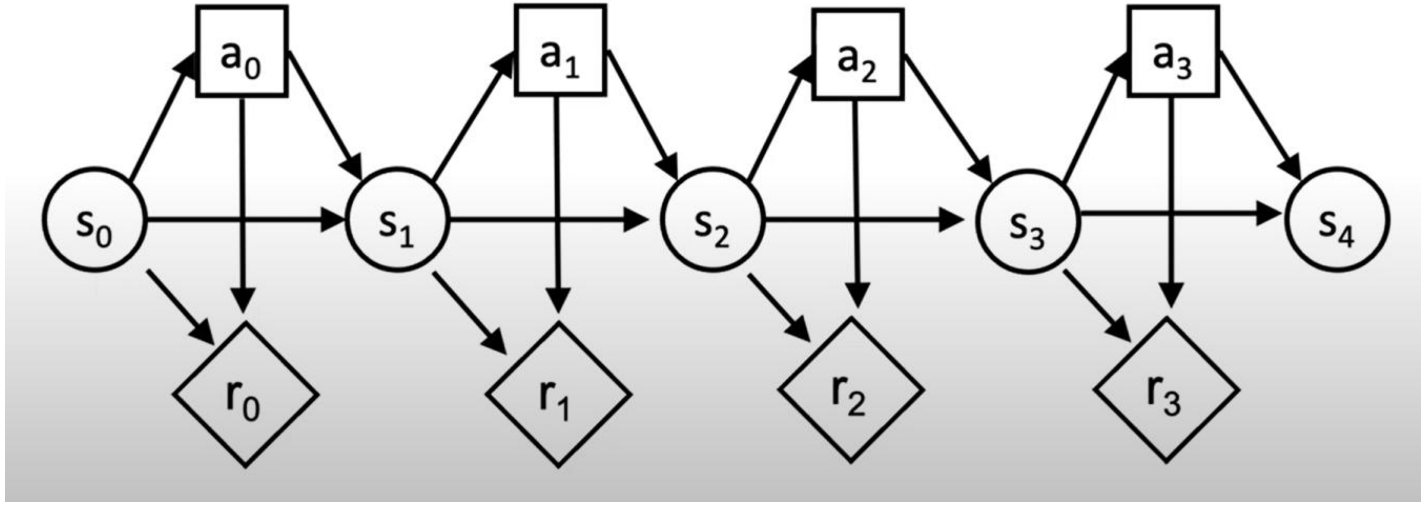
$$MDP \coloneqq (S, A, P, R, \gamma)$$

State Space

Action Space

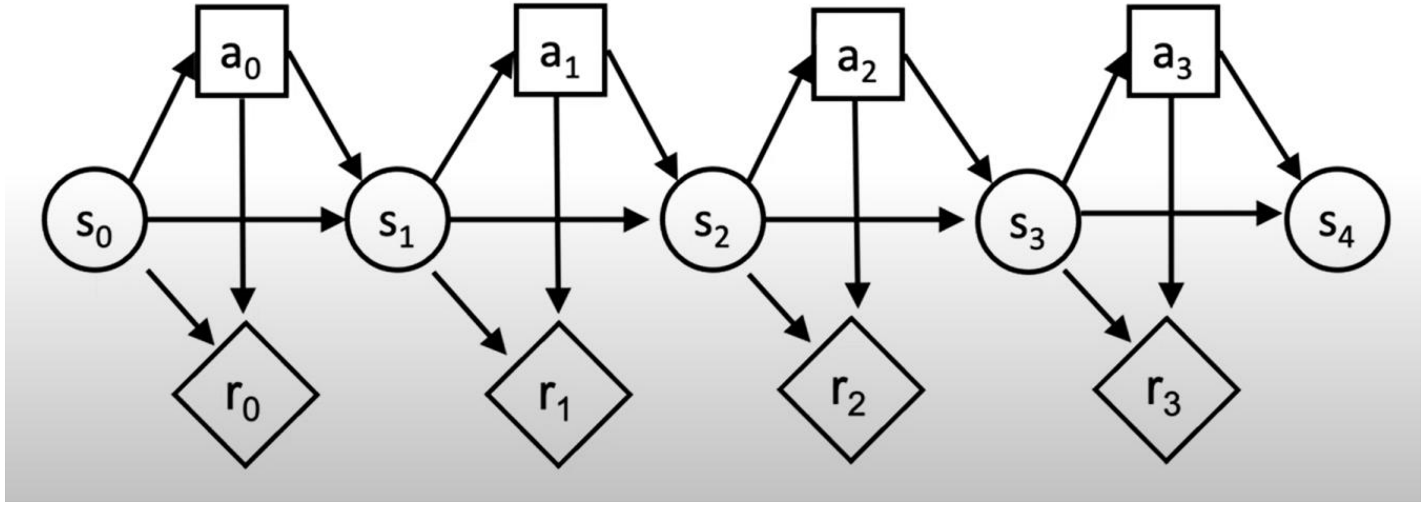
Transition Probability

Markov Decision Processes



$$MDP \coloneqq (\underbrace{S}_{\text{State Space}}, \underbrace{A}_{\text{Action Space}}, \underbrace{P}_{\text{Transition Probability}}, \underbrace{R}_{\text{Reward Function}}, \gamma)$$

Markov Decision Processes



$$MDP \coloneqq (S, A, P, R, \gamma)$$

State Space

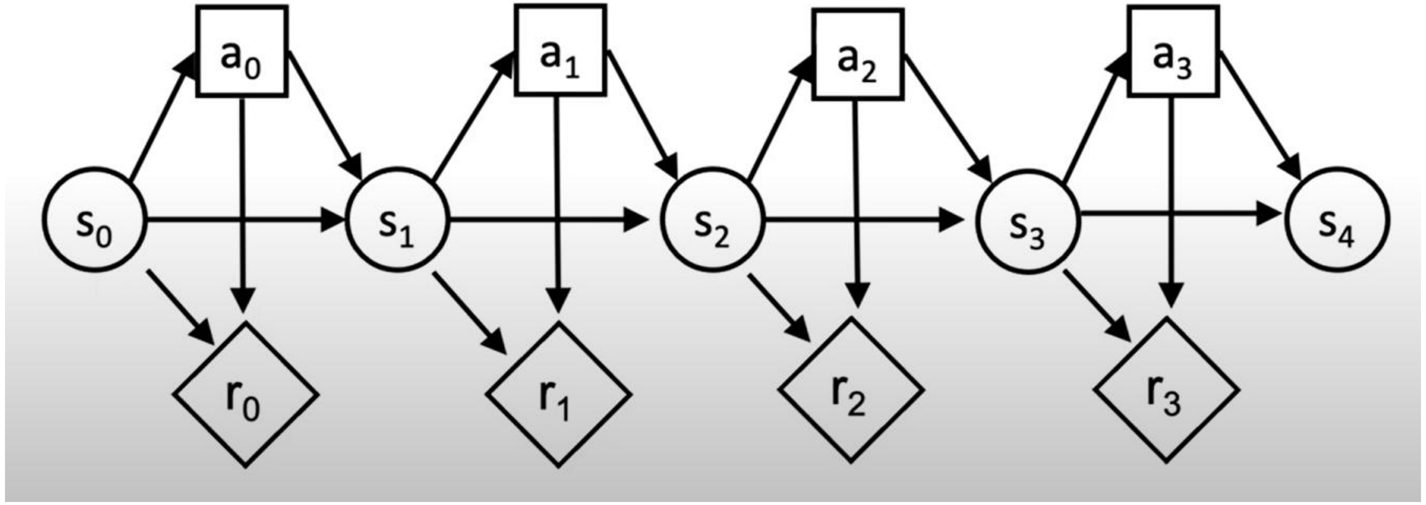
Action Space

Transition Probability

Reward Function

Discount factor

Markov Decision Processes



$$MDP \coloneqq (S, A, P, R, \gamma)$$

At each discrete time t, an agent selects an action $a_t \in A$ in state $s_t \in S$, transitions to the next state s_{t+1} with probability $P(s_{t+1} | s_t, a_t)$, and receives the immediate reward $R(s_t, a_t, s_{t+1})$

GOAL

Choose actions at each step that maximize its expected future discounted reward

Find a strategy (policy) $\pi: s_t \in S \rightarrow a_t \in A(s)$ that maximize value,

$$\mathrm{v} = \left[\sum_{t=0}^{\infty} \gamma^t r^t\right]$$

where

- r^t is the reward earned at time t.
- γ is the discount factor.

How to solve this?

Value Iteration Algorithm

Value Iteration Algorithm

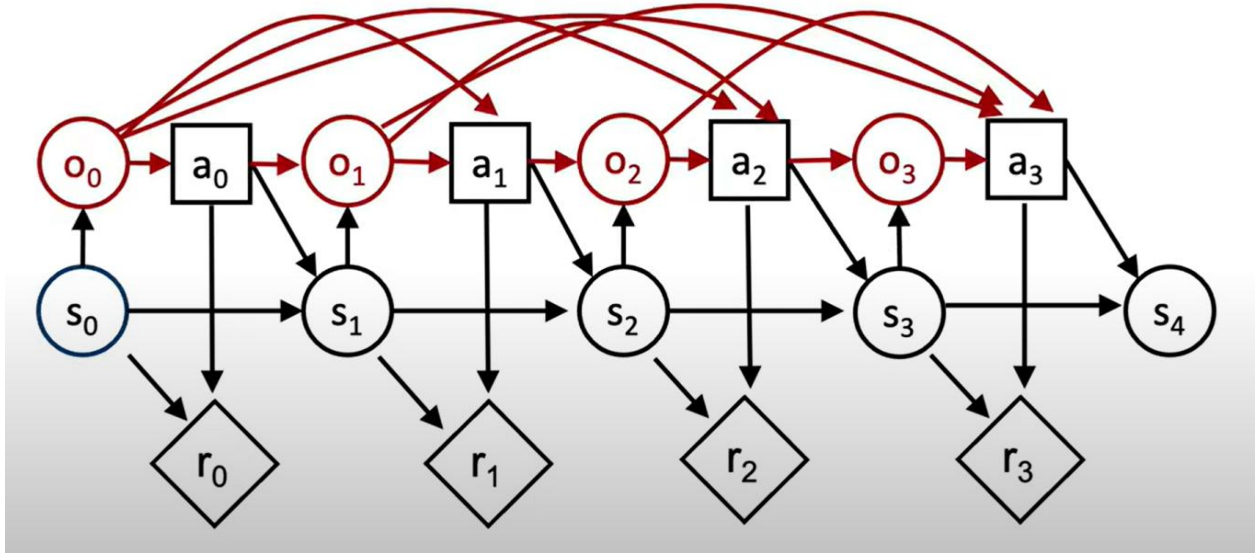
Input : MDP $M = \langle S, s_0, A, P_a(s' \mid s), r(s, a, s') \rangle$
Output : Value function V

Set V to arbitrary value function; e.g., $V(s) = 0$ for all s

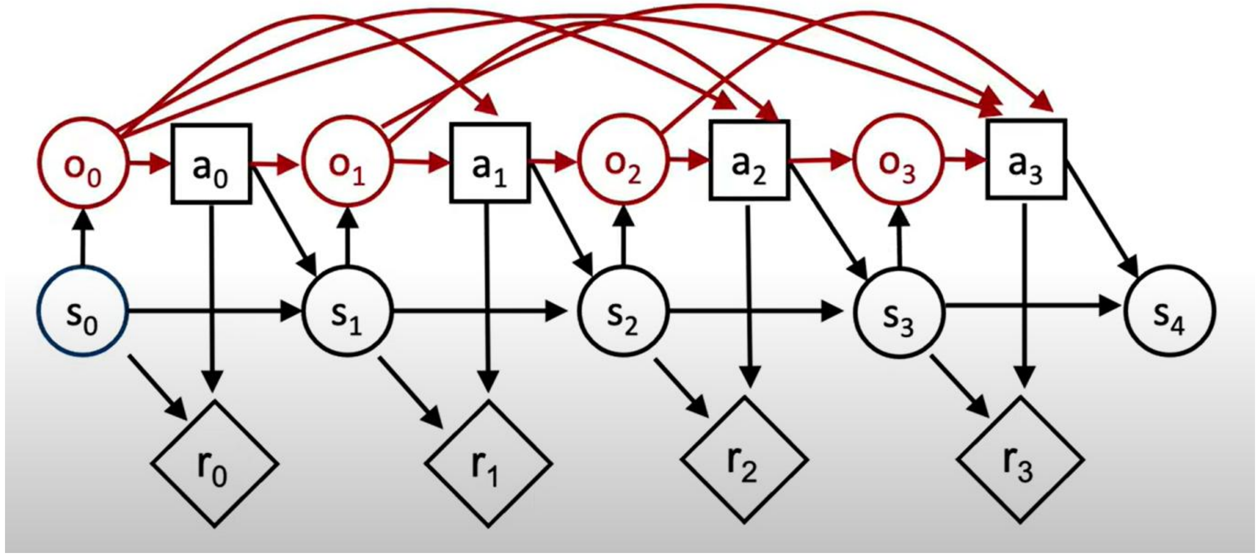
repeat
$\quad \Delta \leftarrow 0$
$\quad$ **for each** $s \in S$
$$\underbrace{V'(s) \leftarrow \max_{a \in A(s)} \sum_{s' \in S} P_a(s' \mid s)\, [r(s, a, s') + \gamma\, V(s')]}_{\text{Bellman equation}}$$
$\quad\quad \Delta \leftarrow \max(\Delta, |V'(s) - V(s)|)$
$\quad V \leftarrow V'$
until $\Delta \leq \theta$

*Theorem: Value iteration converges.

Partially Observable Markov Decision Processes

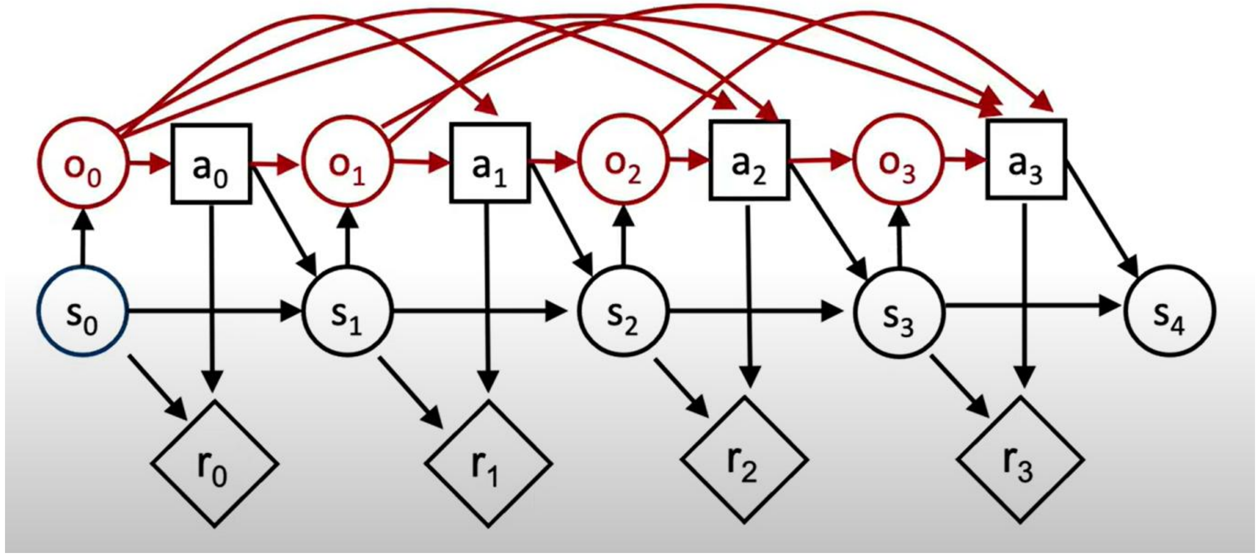


Partially Observable Markov Decision Processes



$$POMDP \coloneqq (S, A, P, R, \gamma, O, \Omega)$$

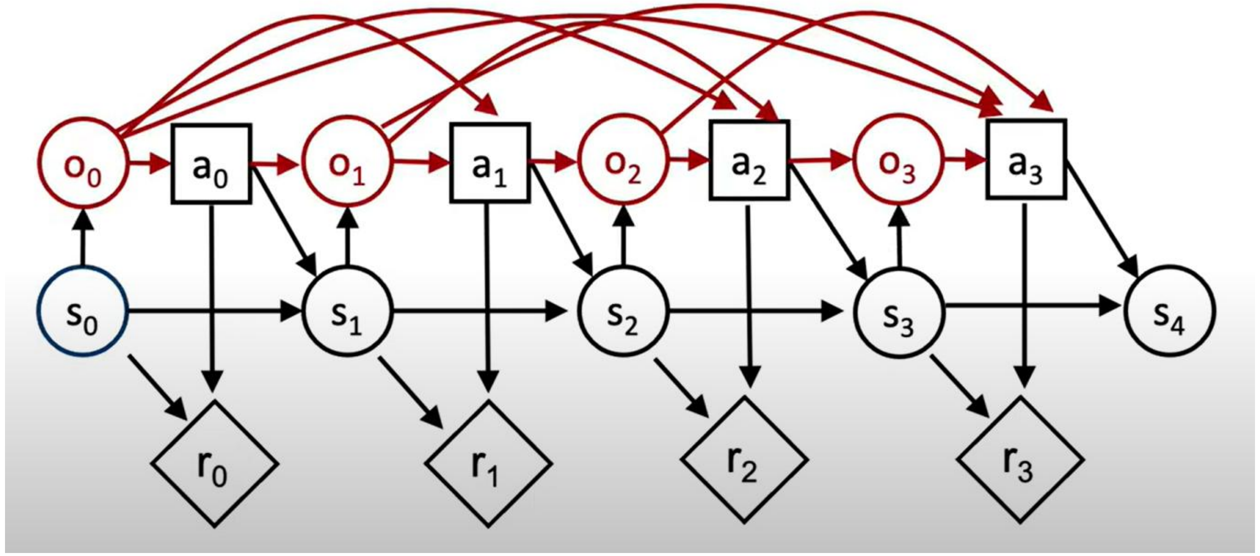
Partially Observable Markov Decision Processes



$$POMDP \coloneqq (S, A, P, R, \gamma, O, \Omega)$$

Observation Space

Partially Observable Markov Decision Processes

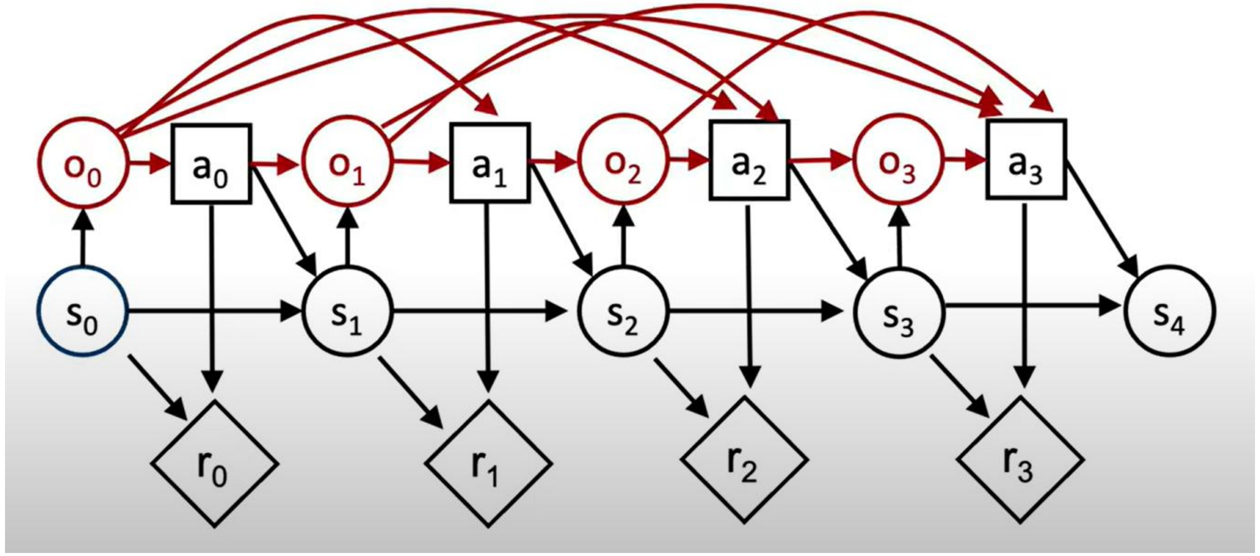


$$POMDP \coloneqq (S, A, P, R, \gamma, O, \Omega)$$

Observation Space

Observation Function

Partially Observable Markov Decision Processes



$$POMDP \coloneqq (S, A, P, R, \gamma, O, \Omega)$$

At each discrete time t, an agent makes observation $o \in O$, selects an action $a_t \in A$, transitions to the next state s_{t+1} with probability $\Omega(o|\, s_{t+1}, a_t)$, and receives the immediate reward $R(s_t, a_t, s_{t+1})$

After having taken the action a_t and observing o_t, a player (agent) needs to update its belief in the state the environment may (or not) be in.

After having taken the action a_t and observing o_t, a player (agent) needs to update its **belief** in the state the environment may (or not) be in.

What is belief?

Belief State and Space

After reaching s_{t+1} , the agent observes $o_t \in O$ with probability $\Omega(o_t \mid s_{t+1}, a_t)$. Let b be a probability distribution over the state space S. $b(s_t)$ denotes the probability that the environment is in state s_t. Given $b(s_t)$, then after taking action and observing o_t,

$$b'(s_{t+1}) = \eta\, \Omega(\, o_t \mid s_{t+1}, a_t\,) \sum_{s \in S} P(s_{t+1} \mid s_t, a_t)\, b(s_t)$$

where

$$\eta \; = \; \frac{1}{\Pr(o_t \mid b, a_t)}$$

is a normalizing constant with

$$\Pr(\, o_t \mid b, a_t\,) = \sum_{s_{t+1} \in S} \Omega(\, o_t \mid s_{t+1}, a_t\,) \sum P(\, s_{t+1} \mid s_t, a_t\,) b(s_t)$$

What's the point?

Find a strategy (policy) $\pi: b(s_t) \in \beta \rightarrow a_t \in A(s)$ that maximize value,

$$\mathrm{v(b)} = \left[\sum_{t=0}^{\infty} \gamma^t r^t\right]$$

where

- r^t is the reward earned at time t.
- γ is the discount factor.

Let’s model Blackjack

Blackjack as POMDP

- $S = \big\{(p, d): p, d \in \{1,2, \cdots, 21\}\big\} \cup \{Win, Lose, Draw\} \cup \{R, NR\}$

Blackjack as POMDP

- $S = \left\{(p,d): p,d \in \{1,2,\cdots,21\}\right\} \cup \{Win, Lose, Draw\} \cup \{R, NR\}$
- $A = \{Hit, Stay\}$

Blackjack as POMDP

- $S = \left\{(p, d): p, d \in \{1,2,\cdots,21\}\right\} \cup \{Win, Lose, Draw\} \cup \{R, NR\}$
- $A = \{Hit, Stay\}$
- $P \coloneqq S \times A \times S \ \rightarrow [0,1]$ where the probability of each action is $\frac{1}{13}$

Blackjack as POMDP

- $S = \left\{(p, d): p, d \in \{1,2,\cdots,21\}\right\} \cup \{Win, Lose, Draw\} \cup \{R, NR\}$
- $A = \{Hit, Stay\}$
- $P \coloneqq S \times A \times S \ \rightarrow [0,1]$ where the probability of each action is $\frac{1}{13}$
- $R \in [0,1]$

Blackjack as POMDP

- $S = \{(p, d): p, d \in \{1,2,\cdots,21\}\} \cup \{Win, Lose, Draw\} \cup \{R, NR\}$
- $A = \{Hit, Stay\}$
- $P \coloneqq S \times A \times S \rightarrow [0,1]$ where the probability of each action is $\frac{1}{13}$
- $R \in [0,1]$
- $\gamma \in [0,1]$

Blackjack as POMDP

- $S = \{(p, d): p, d \in \{1,2, \cdots, 21\}\} \cup \{Win, Lose, Draw\} \cup \{R, NR\}$
- $A = \{Hit, Stay\}$
- $P \coloneqq S \times A \times S \ \rightarrow [0,1]$ where the probability of each action is $\frac{1}{13}$
- $R \in [0,1]$
- $\gamma \in [0,1]$
- $O \coloneqq \ S\ with\ NR$

Blackjack as POMDP

- $S = \big\{(p,d): p,d \in \{1,2,\cdots,21\}\big\} \cup \{Win, Lose, Draw\} \cup \{R, NR\}$
- $A = \{Hit, Stay\}$
- $P \coloneqq S \times A \times S \rightarrow [0,1]$ where the probability of each action is $\frac{1}{13}$
- $R \in [0,1]$
- $\gamma \in [0,1]$
- $O \coloneqq S\ with\ NR$
- $\Omega \coloneqq \mathrm{s} \in S$ with uniform probability

Can we use the same algorithm to solve POMDPs?

Can we use the same algorithm to solve POMDPs?

Input : MDP $M = \langle S, s_0, A, P_a(s' \mid s), r(s,a,s') \rangle$
Output : Value function V

Set V to arbitrary value function; e.g., $V(s) = 0$ for all s

repeat
  $\Delta \leftarrow 0$
  for each $s \in S$ $\boldsymbol{b(s_t) \in \beta}$
    $$\underbrace{V'(s) \leftarrow \max_{a \in A(s)} \sum_{s' \in S} P_a(s' \mid s)\,[r(s,a,s') + \gamma\, V(s')]}_{\text{Bellman equation}}$$
    $\Delta \leftarrow \max(\Delta, |V'(s) - V(s)|)$
  $V \leftarrow V'$
until $\Delta \leq \theta$

NO

Value iteration updates cannot be carried out because uncountable number of belief states

Resources

- POMDP Tutorial: https://www.pomdp.org/tutorial/pomdp-solving.html
- Lovejoy 1991: A survey of algorithmic methods for partially observed Markov decision processes
- Wikipedia: https://en.wikipedia.org/wiki/Partially_observable_Markov_decision_process