Markov Decision Processes

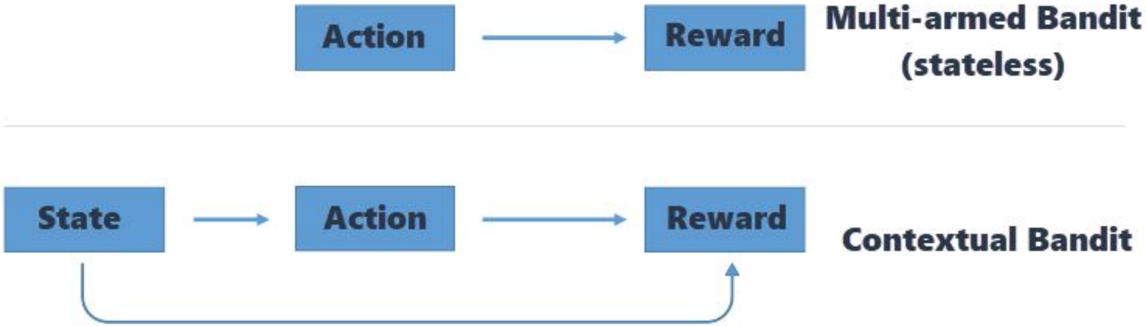
Reinforcement Learning

Roberto Capobianco

Sapienza Università di Roma

Recap

From Multi-Armed to Contextual Bandits



Contextual bandits add back some context (state)

Contextual Bandits: Interaction

The interactive process that we deal with in CB is the following:

For t = 0, ..., T-1:

1. A new i.i.d. context x_t in X appears
2. Select an action a_t in A based on historical information and context
3. Observe reward $r(x_t,a_t)$ (which is context and arm dependent)

For simplicity we assume deterministic rewards, as the context is the challenge here

Contextual Bandits: Regret

Optimal policy: $\pi^\star = \arg\max_{\pi \in \Pi} \mathbb{E}_{x \sim \mu} r(x, \pi(x))$

At every iteration $a_t = \pi_t(x_t)$ is selected and a reward $r(x_t, a_t)$ is received: the regret is the **total expected reward if we always use π^*** VS the **total expected reward if we use our learned sequence of policies**

$$\text{Regret}_T = \boxed{T\mathbb{E}_{x\sim\mu}[r(x, \pi^\star(x))]} - \boxed{\sum_{t=0}^{T-1} \mathbb{E}_{x\sim\mu}[r(x, \pi^t(x))]}$$

Note that policies are different at every iteration t

Explore & Commit Algorithm

1. For t = 0, ..., N-1: **(explore)**
   - observe state $x_t \sim \mu$
   - uniform-randomly sample $a_t \sim \mathrm{Unif}(A)$
   - observe reward $r_t = r(x_t, a_t)$
   - build, for x_t, an unbiased estimate of $\boxed{\mathbb{E}_{a_t \sim p}\hat{\mathbf{r}}[a] = r(x_t, a), \forall a}$
2. Compute policy

$$\hat{\pi} = \arg\max_{\pi \in \Pi} \sum_{i=0}^{N-1} \hat{\mathbf{r}}_i[\pi(x_i)]$$

Given we are sampling from Unif(A)

$$\hat{\mathbf{r}}_t[a] = \begin{cases} 0 & a \neq a_t \\ \dfrac{r_t}{1/|\mathscr{A}|} & a = a_t \end{cases}$$

3. For t = N, ..., T-1: **(commit)**
   - observe state $x_t \sim \mu$
   - play arm

$$\mathrm{Regret}_T = T\mathbb{E}_{x \sim \mu}[r(x, \pi^\star(x))] - \sum_{t=0}^{T-1} \mathbb{E}_{x \sim \mu}[r(x, \pi^t(x))] = O\left(T^{2/3} K^{1/3} \cdot \ln(|\Pi|)^{1/3}\right)$$

ε-Greedy

Instead of setting a threshold for exploring and then committing, we can try to interleave exploration and exploitation

1. For t = 0, ..., T: **(interleave exploration & exploitation)**
   - observe state $x_t \sim \mu$
   - $a_t \sim p_t = \boxed{(1-\varepsilon)\delta(\pi^t(x_t)}) + \boxed{\varepsilon \mathrm{Unif}(A}$)
   - observe reward $r_t = r(x_t, a_t)$
   - build, for x_t, an unbiased estimate of $\mathbb{E}_{a_t \sim p}\hat{\mathbf{r}}[a] = r(x_t, a), \forall a$
2. Update policy

$$\pi^{t+1} = \arg\max_{\pi \in \Pi} \sum_{i=0}^{t} \hat{\mathbf{r}}_i[\pi(x_i)]$$

ε = 0 -> exploit

ε = 1 -> uniformly explore

SAPIENZA UNIVERSITÀ DI ROMA

Bayesian Bandits

So far we have made no assumptions about the reward distribution ν_i, we only derived bounds on rewards

In Bayesian Bandits, however:

- We exploit *prior* knowledge of rewards
- Update a *posterior distribution* of rewards based on historical information
- Use posterior to guide exploration using:
  - upper confidence bounds (Bayesian UCB)
  - probability matching (Thompson Sampling)

Gaussian Bayesian Bandits: UCB

Now we are modelling a distribution, so we already have confidence

What is confidence for Gaussians? **standard deviation**

Let's do UCB by selecting the action with highest standard deviation

$$a_t = \text{argmax}_{i \text{ in } K}\, \mu_t(i) + c\sigma_t(i)/\sqrt{N_t(i)}$$

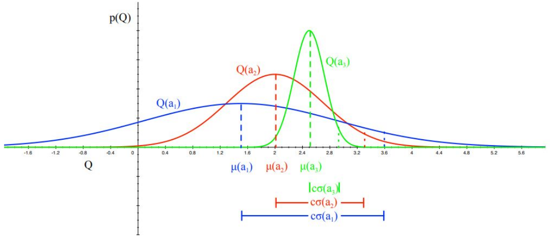
Gaussian Bayesian Bandits: Thompson Sampling

This is an estimation of the reward, in more generic MDPs this can be replaced with the Q function: we estimate a distribution of Q

For t = 0, ..., T:

1. for each arm i = 1, ..., K:
   - sample $\boxed{\hat{\mathbf{r}}_i}$ independently from $N(\mu_{t-1}(i), \sigma^2_{t-1}(i))$
2. pull arm

$$I_t = \arg\max_{i \in [K]} \hat{\mathbf{r}}_i$$

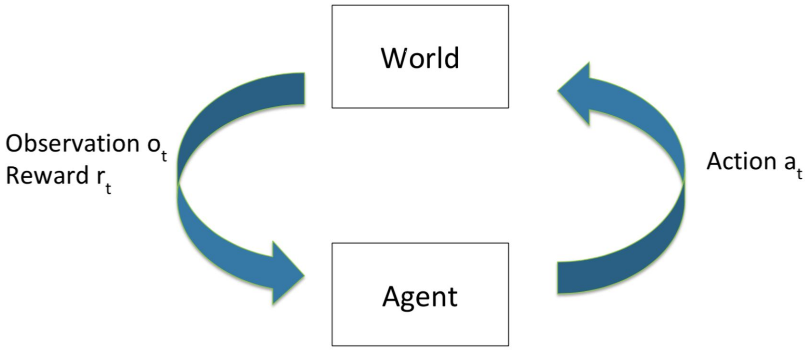


3. observe reward r_t
4. update posterior distribution $p(\mu_t(i), \sigma^2_t(i) | r_t)$

This can be done with different distributions as well

End Recap

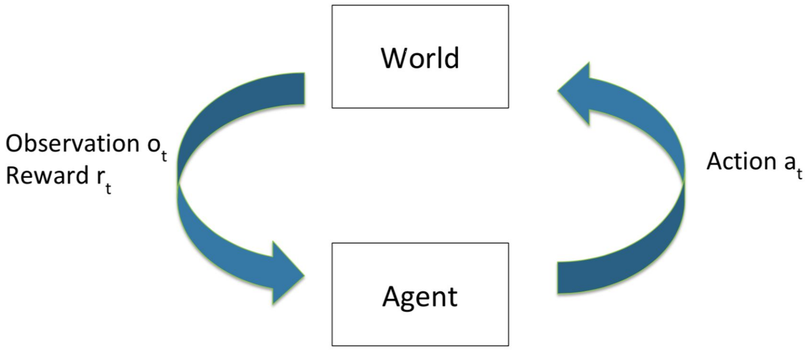
Sequential Decision Making



The agent interacts with the environment:

- at discrete timesteps;
- by receiving observations o_t and reward r_t from the environment;
- by taking actions a_t;

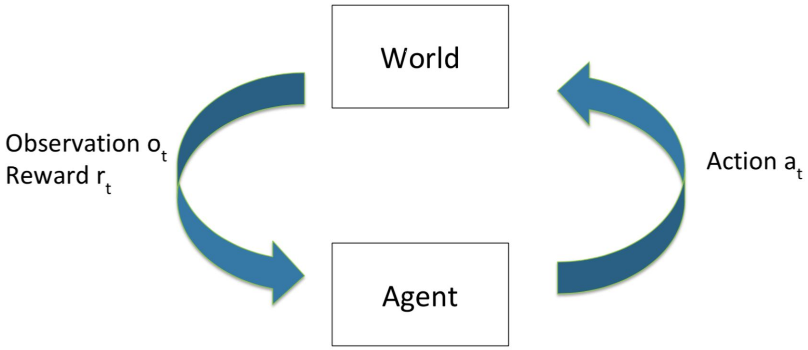
Sequential Decision Making



Such discrete interaction generates a trajectory, or history at each timestep *t*, that is used by the agent to take action:

$$h_t = (o_0, a_0, r_1, o_1, a_1, \dots r_t, o_t, a_t)$$

Sequential Decision Making



The state is a function of the history:

$$s_t = f(h_t)$$

and it is typically hidden or unknown

Markov Assumption

A state st is Markovian iff future is independent of the past given the present

$$p(s_{t+1}|s_t,a_t) = p(s_{t+1}|h_{t,}a_t)$$

Markov Assumption

A state s_t is Markovian iff future is independent of the past given the present

$$p(s_{t+1}|s_t,a_t) = p(s_{t+1}|h_t,a_t)$$

Is this problem Markovian?

Markov Assumption

- A state can always be made markovian by setting it to be equal to the history

$$s_t = h_t$$

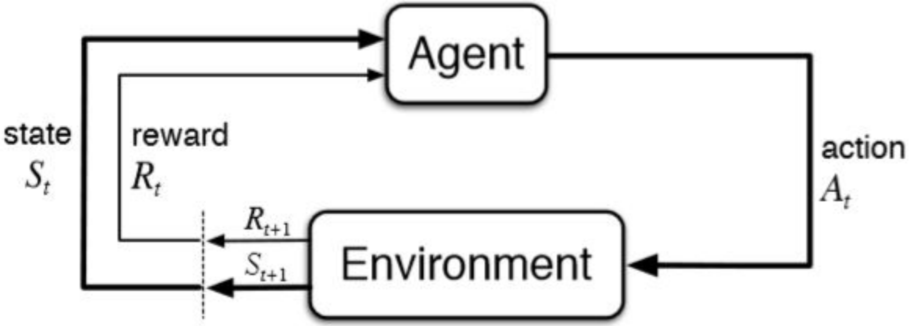
- The best case (used in practice) is: current state corresponds to (or is a sufficient statistic of) latest observation

$$s_t = o_t$$

- In this case the state is said to be *fully observable*

Markov Decision Process (MDP)

- Set of states S
- Set of actions A

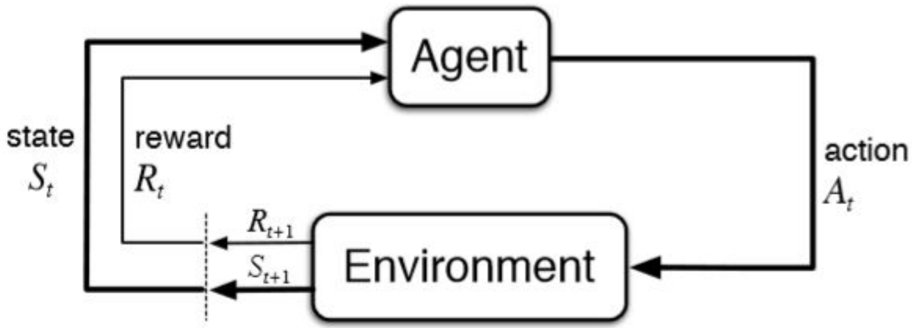


Sequential Decision Making under Markov Assumption

- Markovian transition dynamics
- Full Observability
- The transition dynamics *T* is (generally) stochastic $p(s_{t+1}|s_t,a_t)$

Markov Decision Process (MDP)

- Set of states S
- Set of actions A



Alternative notation

$s_{t+1} \sim p(.|s_t,a_t)$ or

$s' \sim p(.|s,a)$

Sequential Decision Making under Markov Assumption

- Markovian transition dynamics
- Full Observability
- The transition dynamics *T* is (generally) stochastic $p(s_{t+1}|s_t,a_t)$

Reward

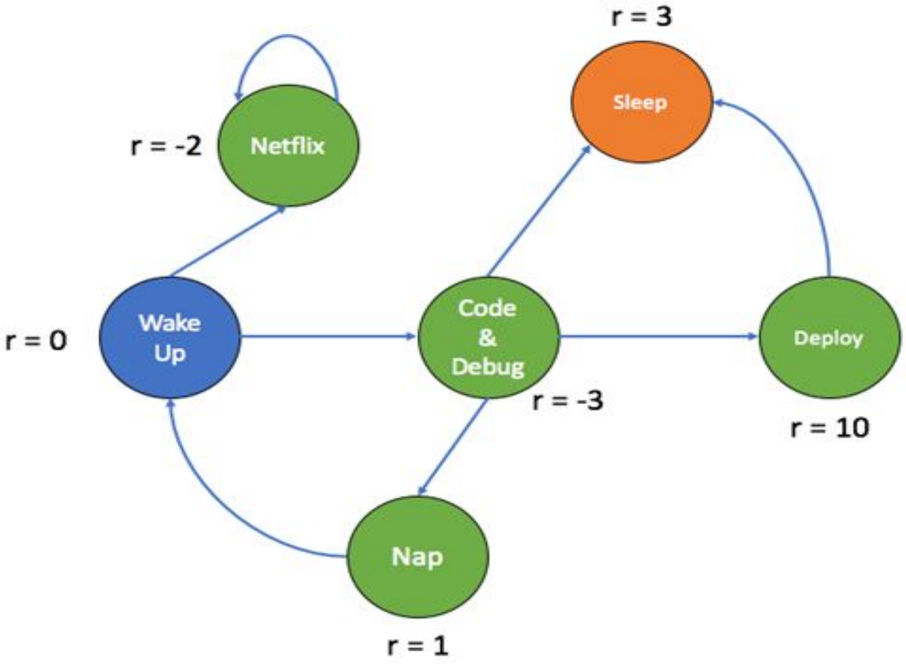
A reward r_t is a:

- scalar signal representing a feedback
- indicates how well an agent is doing at step t
- the reward is a function of state and action (often indicated as R(s,a) and sometimes R(s',a,s))
- cost is the inverse of the reward

Reward hypothesis: *can all goals be achieved through the maximization of a numerical reward?*

It's an open question

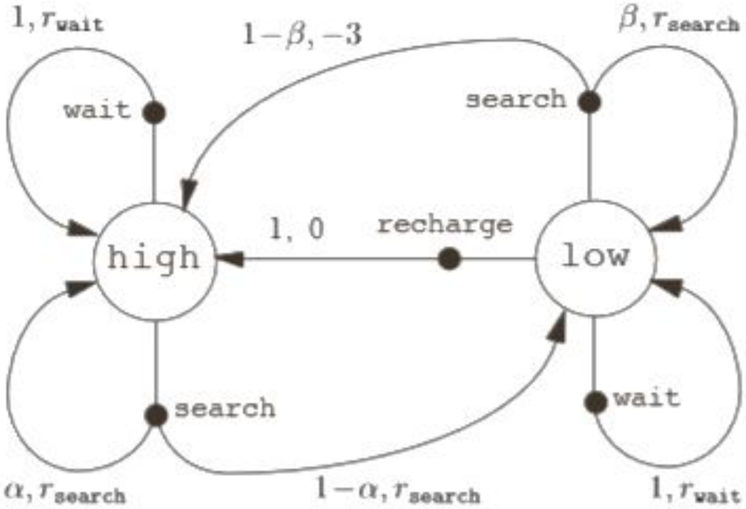
Deterministic MDP Example



Stochastic MDP Example

Recycling robot

s	a	s'	$p(s' \mid s, a)$	$r(s, a, s')$
high	search	high	α	r_{search}
high	search	low	$1 - \alpha$	r_{search}
low	search	high	$1 - \beta$	-3
low	search	low	β	r_{search}
high	wait	high	1	r_{wait}
high	wait	low	0	-
low	wait	high	0	-
low	wait	low	1	r_{wait}
low	recharge	high	1	0
low	recharge	low	0	-

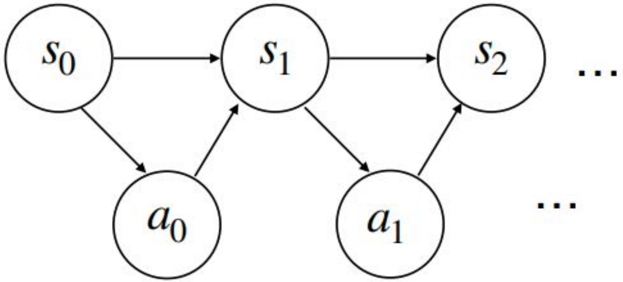


Policy

A policy π:

- is a mapping from (all) states to actions;
- determines how agents select actions;
- can be deterministic ($a = \pi(s)$) or stochastic ($\pi(a|s)$ or $p(a|s)$ or $a \sim \pi(.|s)$)

Trajectory Probability



What's the probability of seeing a trajectory at time t according to π starting at s_0?

$$(s_0,a_0,s_1,a_1,\ldots s_t,a_t)$$

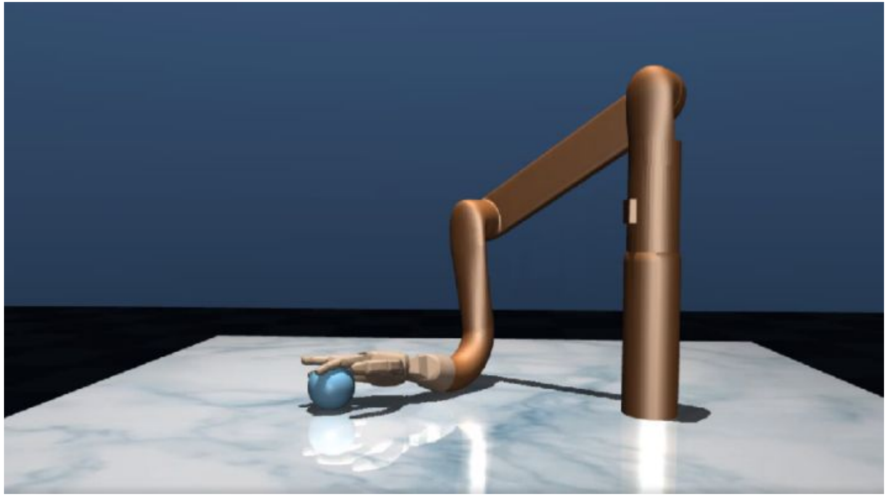
$$\mathbb{P}^{\pi}(s_0,a_0,\ldots s_t,a_t)=\pi(a_0|s_0)p(s_1|s_0,a_0)\pi(a_1|s_1)p(s_2|s_1,a_1)\ldots p(s_t|s_{t-1},a_{t-1})\pi(a_t|s_t)$$

State Visitation Probability

What's the probability of visiting state s, a at time t according to π starting at s_0?

$$\mathbb{P}^{\pi}{}_{t}(s,a;s_0)=\sum_{a0,s1,a1,\ldots st-1,at-1}\mathbb{P}^{\pi}(s_0,a_0,\ldots s_t=s,a_t=a)$$

Another Example MDP



- **state:** robot configuration (joint states) and ball position
- **action:** torque on arm and finger joints
- **transition:** stochastic, physics plus noise
- **policy:** mapping from robot state and ball position to torque
- **cost:** magnitude of the torque and distance to the goal

Infinite Horizon Discounted Setting

So far in our MDP we have (S, A, T, R)

Now we add the discount factor γ to reason on the policy's long term effects

- γ is in [0, 1]
- γ = 0 means: I only care about immediate rewards
- γ = 1 means: Immediate and future rewards are equally important

How so?

Value Function

- We estimate the goodness of states and actions based on their value
- It's also a measure to compare policies

$V^{\pi}(s_t) = \mathbb{E}_{\pi}[r_t+\gamma r_{t+1}+\gamma^2 r_{t+2}+\gamma^3 r_{t+3}+...|s_t] = \mathbb{E}[\sum^{\infty}_{h=0}\gamma^h r_h|s_0=s_t,a_h=\pi(s_h),s_{h+1}\sim p(.|s_h,a_h)]$

Value Function/Q-Function

- We estimate the goodness of states and actions based on their value
- It's also a measure to compare policies

$V^{\pi}(s_t) = \mathbb{E}_{\pi}[r_t+\gamma r_{t+1}+\gamma^2 r_{t+2}+\gamma^3 r_{t+3}+...|s_t] = \mathbb{E}[\sum^{\infty}_{h=0}\gamma^h r_h|s_0=s_t,a_h=\pi(s_h),s_{h+1}\sim p(.|s_h,a_h)]$

$Q^{\pi}(s_t,a_t) = \mathbb{E}[\sum^{\infty}_{h=0}\gamma^h r_h|(s_0,a_0)=(s_t,a_t),a_{h+1}=\pi(s_h),s_{h+1}\sim p(.|s_h,a_h)]$

Back to Discount Factor

Setting $\gamma = 1$ for infinite tasks is a bad idea!

Note that $\sum^{\infty}_{h=0} \gamma^h$ is a geometric series and for γ in [0,1] this is equivalent to $1/(1-\gamma)$

So, the value of γ approximately determines how many steps ahead we are considering

E.g., $\gamma=0.99$ -> 99 timesteps ahead

Bellman Equation

The value of a certain state is expanded in terms of the current reward and the value of the next states according to the policy

$V^{\pi}(s_t) = \mathbb{E}_{\pi}[r_t+\gamma r_{t+1}+\gamma^2 r_{t+2}+\gamma^3 r_{t+3}+...|s_t] = r_t+\gamma\mathbb{E}_{s'\sim p(.|s,\pi(s))}[V^{\pi}(s')]$

Bellman Equation also for Q

The value of a certain state is expanded in terms of the current reward and the value of the next states according to the policy

$V^{\pi}(s_t) = \mathbb{E}_{\pi}[r_t+\gamma r_{t+1}+\gamma^2 r_{t+2}+\gamma^3 r_{t+3}+...|s_t] = r_t+\gamma\mathbb{E}_{s'\sim p(.|s,\pi(s))}[V^{\pi}(s')]$

$Q^{\pi}(s_t,a) = r_t+\gamma\mathbb{E}_{s'\sim p(.|s,a)}[V^{\pi}(s')]$

Bellman Equation also for Q

The value of a certain state is expanded in terms of the current reward and the value of the next states according to the policy

r here is function of s and π(s)

$$V^{\pi}(s_t) = \mathbb{E}_{\pi}[r_t+\gamma r_{t+1}+\gamma^2 r_{t+2}+\gamma^3 r_{t+3}+...|s_t] = r_t+\gamma\mathbb{E}_{s'\sim p(.|s,\pi(s))}[V^{\pi}(s')]$$

$$Q^{\pi}(s_t,a) = r_t+\gamma\mathbb{E}_{s'\sim p(.|s,a)}[V^{\pi}(s')]$$

r here is function of s and a

SAPIENZA
UNIVERSITÀ DI ROMA

Bellman Equation also for Q

The value of a certain state is expanded in terms of the current reward and the value of the next states according to the policy

r here is function of s and π(s)

$$V^{\pi}(s_t) = \mathbb{E}_{\pi}[r_t+\gamma r_{t+1}+\gamma^2 r_{t+2}+\gamma^3 r_{t+3}+...|s_t] = r_t+\gamma\mathbb{E}_{s'\sim p(.|s,\pi(s))}[V^{\pi}(s')]$$

$$Q^{\pi}(s_t,a) = r_t+\gamma\mathbb{E}_{s'\sim p(.|s,a)}[V^{\pi}(s')]$$

r here is function of s and a

As a result $V(s) = Q(s,\pi(s))$

Discounted State-Action Distribution

$$d^{\pi}_{s0}(s,a) = (1-\gamma)\sum^{\infty}_{h=0}\gamma^{h}\mathbb{P}^{\pi}_{h}(s,a;s_{0})$$

Discounted State-Action Distribution

$$d^{\pi}_{s0}(s,a) = (1-\gamma)\sum^{\infty}_{h=0}\gamma^{h}\mathbb{P}^{\pi}_{h}(s,a;s_0)$$

This gives us a probability distribution
(remember $\sum^{\infty}_{h=0}\gamma^{h}$ equals $1/(1-\gamma)$)

Optimal Policy

For infinite horizon MDPs there always exists a deterministic policy π^* such that

$$V^{\pi^*}(s) \geq V^{\pi}(s) \ \forall \ s, \pi$$

meaning that π^* dominates all other policies π in each state

Optimal Policy

For infinite horizon MDPs there always exists a deterministic policy π^* such that it returns optimal actions a^* and

$$V^{\pi^*}(s) \geq V^{\pi}(s) \ \forall \ s, \pi$$

Alternative notation
$V^{\pi^*} = V^*$ and $Q^{\pi^*} = Q^*$

meaning that π^* dominates all other policies π in each state

Bellman Optimality

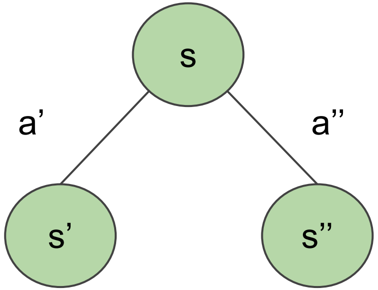
$$V^*(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V^*(s')]$$

Bellman Optimality

$$V^*(s)=\max_a\Big[\underbrace{r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V^*(s')}_{Q^*(s,a)}\Big]$$

Bellman Optimality Example

$$V^*(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V^*(s')]$$

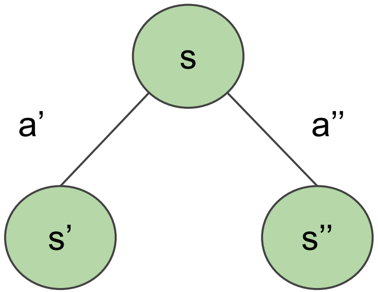


Assume we know V^* at s' and s''

Bellman Optimality Example

$$V^*(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V^*(s')]$$

- Try a', get r(s,a'), compute $Q^*(s,a')=r(s,a')+\gamma V^*(s')$
- Try a'', get r(s,a''), compute $Q^*(s,a'')=r(s,a'')+\gamma V^*(s'')$

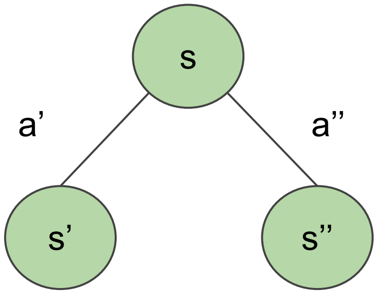


Assume we know V^* at s' and s''

Bellman Optimality Example

$$V^*(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V^*(s')]$$

- Try a', get r(s,a'), compute $Q^*(s,a')=r(s,a')+\gamma V^*(s')$
- Try a'', get r(s,a''), compute $Q^*(s,a'')=r(s,a'')+\gamma V^*(s'')$



Assume we know V* at s' and s''

$$V^*(s)=\max_{a',a''}\{Q^*(s,a'),Q^*(s,a'')\}$$

Bellman Optimality (Theorem 1)

$$V^*(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V^*(s')]$$

given $\hat{\pi}=\arg\max_a Q^*(s,a)$, we can show $V^{\hat{\pi}}=V^*$

Bellman Optimality (Theorem 1)

$$V^*(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V^*(s')]$$

given $\hat{\pi}=\text{argmax}_a Q^*(s,a)$, we can show $V^{\hat{\pi}}=V^*$

$$\begin{aligned}
V^\star(s) &= r(s,\pi^\star(s)) + \gamma\mathbb{E}_{s'\sim P(s,\pi^\star(s))}V^\star(s') \\
&\leq \max_a \left[r(s,a) + \gamma\mathbb{E}_{s'\sim P(s,a)}V^\star(s')\right] = r(s,\widehat{\pi}(s)) + \gamma\mathbb{E}_{s'\sim P(s,\widehat{\pi}(s))}V^\star(s') \\
&= r(s,\widehat{\pi}(s)) + \gamma\mathbb{E}_{s'\sim P(s,\widehat{\pi}(s))}\left[r(s',\pi^\star(s')) + \gamma\mathbb{E}_{s''\sim P(s',\pi^\star(s'))}V^\star(s'')\right] \\
&\leq r(s,\widehat{\pi}(s)) + \gamma\mathbb{E}_{s'\sim P(s,\widehat{\pi}(s))}\left[r(s',\widehat{\pi}(s')) + \gamma\mathbb{E}_{s''\sim P(s',\widehat{\pi}(s'))}V^\star(s'')\right] \\
&\leq r(s,\widehat{\pi}(s)) + \gamma\mathbb{E}_{s'\sim P(s,\widehat{\pi}(s))}\left[r(s',\widehat{\pi}(s')) + \gamma\mathbb{E}_{s''\sim P(s',\widehat{\pi}(s'))}\left[r(s'',\widehat{\pi}(s'')) + \gamma\mathbb{E}_{s'''\sim P(s'',\widehat{\pi}(s''))}V^\star(s''')\right]\right] \\
&\leq \mathbb{E}\left[r(s,\widehat{\pi}(s)) + \gamma r(s',\widehat{\pi}(s')) + \ldots\right] = V^{\widehat{\pi}}(s)
\end{aligned}$$

Bellman Optimality (Theorem 1)

$$V^*(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V^*(s')]$$

given $\hat{\pi}=\text{argmax}_a Q^*(s,a)$, we can show $V^{\hat{\pi}}=V^*$

$V^{\hat{\pi}}\geq V^*$ and $V^*\geq V^{\hat{\pi}}$

$$
\begin{aligned}
V^\star(s) &= r(s,\pi^\star(s)) + \gamma\mathbb{E}_{s'\sim P(s,\pi^\star(s))}V^\star(s') \\
&\leq \max_a\left[r(s,a)+\gamma\mathbb{E}_{s'\sim P(s,a)}V^\star(s')\right] = r(s,\hat{\pi}(s)) + \gamma\mathbb{E}_{s'\sim P(s,\hat{\pi}(s))}V^\star(s') \\
&= r(s,\hat{\pi}(s)) + \gamma\mathbb{E}_{s'\sim P(s,\hat{\pi}(s))}\left[r(s',\pi^\star(s')) + \gamma\mathbb{E}_{s''\sim P(s',\pi^\star(s'))}V^\star(s'')\right] \\
&\leq r(s,\hat{\pi}(s)) + \gamma\mathbb{E}_{s'\sim P(s,\hat{\pi}(s))}\left[r(s',\hat{\pi}(s')) + \gamma\mathbb{E}_{s''\sim P(s',\hat{\pi}(s'))}V^\star(s'')\right] \\
&\leq r(s,\hat{\pi}(s)) + \gamma\mathbb{E}_{s'\sim P(s,\hat{\pi}(s))}\left[r(s',\hat{\pi}(s')) + \gamma\mathbb{E}_{s''\sim P(s',\hat{\pi}(s'))}\left[r(s'',\hat{\pi}(s'')) + \gamma\mathbb{E}_{s'''\sim P(s'',\hat{\pi}(s''))}V^\star(s''')\right]\right] \\
&\leq \mathbb{E}\left[r(s,\hat{\pi}(s)) + \gamma r(s',\hat{\pi}(s')) + \ldots\right] = V^{\hat{\pi}}(s)
\end{aligned}
$$

Bellman Optimality (Theorem 1)

$$V^*(s) = \max_a [r(s,a) + \gamma \mathbb{E}_{s' \sim p(.|s,a)} V^*(s')]$$

given $\hat{\pi} = \mathrm{argmax}_a Q^*(s,a)$, we can show $V^{\hat{\pi}} = V^*$

This implies $\pi^* = \mathrm{argmax}_a Q^*(s,a)$ is an optimal policy

Bellman Optimality (Theorem 2)

For any V, if $V(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V(s')]$ for all s, then $V(s)=V^*(s)$

Bellman Optimality (Theorem 2)

For any V, if $V(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V(s')]$ for all s,
then $V(s)=V^*(s)$

We need to check if $|V(s)-V^*(s)|=0$

Bellman Optimality (Theorem 2)

For any V, if $V(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V(s')]$ for all s, then $V(s)=V^*(s)$

We need to check if
$$
\begin{aligned}
|V(s)-V^\star(s)| &= \left|\max_a(r(s,a)+\gamma\mathbb{E}_{s'\sim P(s,a)}V(s')) - \max_a(r(s,a)+\gamma\mathbb{E}_{s'\sim P(s,a)}V^\star(s'))\right| \\
&\le \max_a \left|(r(s,a)+\gamma\mathbb{E}_{s'\sim P(s,a)}V(s')) - (r(s,a)+\gamma\mathbb{E}_{s'\sim P(s,a)}V^\star(s'))\right| \\
&\le \max_a \gamma\mathbb{E}_{s'\sim P(s,a)}\left|V(s')-V^\star(s')\right| \\
&\le \max_a \gamma\mathbb{E}_{s'\sim P(s,a)}\left(\max_{a'}\gamma\mathbb{E}_{s''\sim P(s',a')}\left|V(s'')-V^\star(s'')\right|\right) \\
&\le \max_{a_1,a_2,\ldots a_{k-1}} \gamma^k\mathbb{E}_{s_k}|V(s_k)-V^\star(s_k)|
\end{aligned}
$$

SAPIENZA
UNIVERSITÀ DI ROMA

Bellman Optimality (Theorem 2)

For any V, if $V(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V(s')]$ for all s, then $V(s)=V^*(s)$

We need to check if $$|V(s)-V^\star(s)| = \left|\max_a(r(s,a)+\gamma\mathbb{E}_{s\sim P(s,a)}V(s')) - \max_a(r(s,a)+\gamma\mathbb{E}_{s\sim P(s,a)}V^\star(s'))\right|$$

$$\leq \max_a \left|(r(s,a)+\gamma\mathbb{E}_{s\sim P(s,a)}V(s')) - (r(s,a)+\gamma\mathbb{E}_{s\sim P(s,a)}V^\star(s'))\right|$$

$$\leq \max_a \gamma\mathbb{E}_{s'\sim P(s,a)}\left|V(s')-V^\star(s')\right|$$

$$\leq \max_a \gamma\mathbb{E}_{s'\sim P(s,a)}\left(\max_{a'}\gamma\mathbb{E}_{s''\sim P(s',a')}\left|V(s'')-V^\star(s'')\right|\right)$$

At infinity, this goes to zero

$$\leq \max_{a_1,a_2,\ldots a_{k-1}} \gamma^k\mathbb{E}_{s_k}|V(s_k)-V^\star(s_k)|$$

SAPIENZA
UNIVERSITÀ DI ROMA

Bellman Optimality (Theorem 2)

For any V, if $V(s)=\max_a[r(s,a)+\gamma\mathbb{E}_{s'\sim p(.|s,a)}V(s')]$ for all s, then $V(s)=V^*(s)$

This means we can focus on one step at each time (leaving the remaining "problem" to V(s'), and any V that satisfies this formula is in fact V^*