Planning under uncertainty

Markov decision processes

Christos Dimitrakakis

Chalmers

August 31, 2014

Contents

Objective Probability

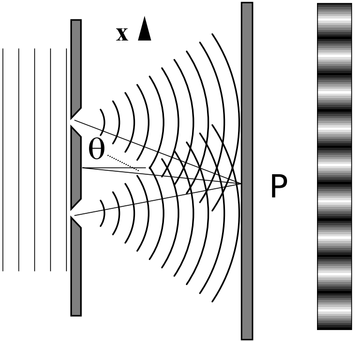


Figure: The double slit experiment

Objective Probability

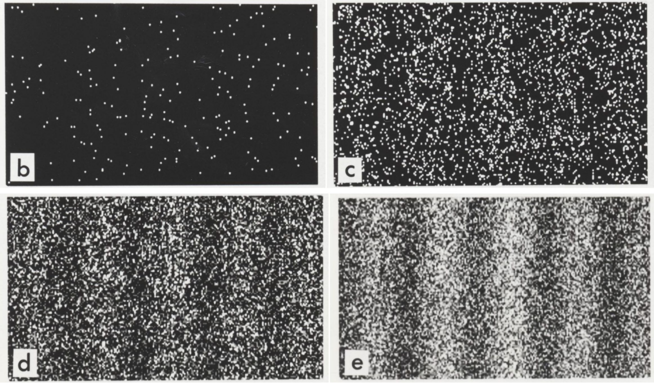


Figure: The double slit experiment

Objective Probability

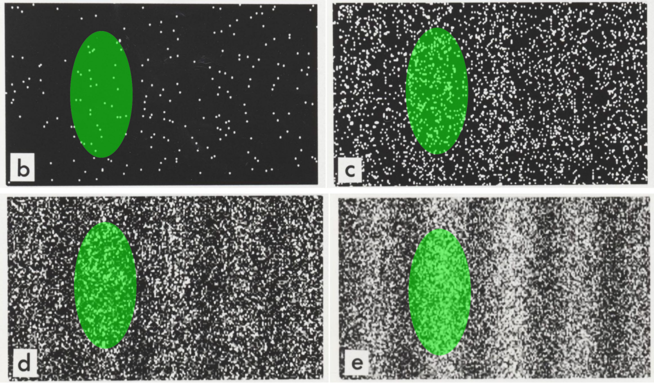


Figure: The double slit experiment

What about everyday life?

Subjective probability

- Making decisions requires making predictions.

Subjective probability

- Making decisions requires making predictions.
- Outcomes of decisions are uncertain.

Subjective probability

- Making decisions requires making predictions.
- Outcomes of decisions are uncertain.
- How can we represent this uncertainty?

Subjective probability

- Making decisions requires making predictions.
- Outcomes of decisions are uncertain.
- How can we represent this uncertainty?

Subjective probability

- Describe which events we think are more likely.
- We quantify this with probability.

Why probability?

- Quantifies uncertainty in a “natural” way.
- A framework for drawing conclusions from data.
- Computationally convenient for decision making.

Assumptions about our beliefs

Our beliefs must be consistent. This can be achieved if they satisfy some assumptions:

Assumption 1 (SP1)

It is always possible to say whether one event is more likely than the other.

Assumptions about our beliefs

Our beliefs must be consistent. This can be achieved if they satisfy some assumptions:

Assumption 1 (SP1)

It is always possible to say whether one event is more likely than the other.

Assumption 2 (SP2)

If we can split events A, B in such a way that each part of A is less likely than its counterpart in B, then A is less likely than B.

Assumptions about our beliefs

Our beliefs must be consistent. This can be achieved if they satisfy some assumptions:

Assumption 1 (SP1)

It is always possible to say whether one event is more likely than the other.

Assumption 2 (SP2)

If we can split events A, B in such a way that each part of A is less likely than its counterpart in B, then A is less likely than B.

There also a couple of technical assumptions..

Resulting properties of relative likelihoods

Theorem 1 (Transitivity)

If A, B, D such that $A \precsim B$ and $B \precsim D$, then $A \precsim D$.

Theorem 2 (Complement)

For any A, B: $A \precsim B$ iff $A^{\complement} \succsim B^{\complement}$.

Theorem 3 (Fundamental property of relative likelihoods)

If $A \subset B$ then $A \precsim B$. Furthermore, $\emptyset \precsim A \precsim S$ for any event A.

Theorem 4

For a given likelihood relation between events, there exists a unique probability distribution P such that

$$P(A) \geq P(B) \Leftrightarrow A \succsim B$$

Similar results can be derived for *conditional* likelihoods and probabilities.

Rewards

- We are going to receive a reward r from a set R of possible rewards.
- We prefer some rewards to others.

Example 5 (Possible sets of rewards R)

- R is a set of tickets to different musical events.
- R is a set of financial commodities.

When we cannot select rewards directly

- In most problems, we cannot just choose which reward to receive.

When we cannot select rewards directly

- In most problems, we cannot just choose which reward to receive.
- We can only specify a distribution on rewards.

When we cannot select rewards directly

- In most problems, we cannot just choose which reward to receive.
- We can only specify a distribution on rewards.

Example 6 (Route selection)

- Each reward $r \in R$ is the time it takes to travel from A to B.
- Route P_1 is faster than P_2 in heavy traffic and vice-versa.
- Which route should be preferred, given a certain probability for heavy traffic?

In order to choose between random rewards, we use the concept of utility.

Utility

Definition 7 (Utility)

The utility is a function $U : R \to \mathbb{R}$, such that for all $a, b \in R$

$$a \succsim^* b \quad \text{iff} \quad U(a) \geq U(b), \tag{1.1}$$

The expected utility of a distribution P on R is:

$$\mathbb{E}_P(U) = \int_R U(r) \, \mathrm{d}P(r) \tag{1.2}$$

Utility

Definition 7 (Utility)

The utility is a function $U : R \to \mathbb{R}$, such that for all $a, b \in R$

$$a \succsim^* b \quad \text{iff} \quad U(a) \geq U(b), \tag{1.1}$$

The expected utility of a distribution P on R is:

$$\mathbb{E}_P(U) = \int_R U(r) \, \mathrm{d}P(r) \tag{1.2}$$

Utility

Definition 7 (Utility)

The utility is a function $U : R \to \mathbb{R}$, such that for all $a, b \in R$

$$a \succsim^* b \quad \text{iff} \quad U(a) \geq U(b), \tag{1.1}$$

The expected utility of a distribution P on R is:

$$\mathbb{E}_P(U) = \int_R U(r) \, \mathrm{d}P(r) \tag{1.2}$$

Assumption 3 (The expected utility hypothesis)

The utility of P is equal to the expected utility of the reward under P. Consequently,

$$P \succsim^* Q \quad \textit{iff} \quad \mathbb{E}_P(U) \geq \mathbb{E}_Q(U). \tag{1.3}$$

Example 8

r	U(r)	P	Q
did not enter	0	1	0
paid 1 CU and lost	-1	0	0.99
paid 1 CU and won 10	9	0	0.01

Table: A simple gambling problem

	P	Q
$\mathbb{E}(U \mid \cdot)$	0	-0.9

Table: Expected utility for the gambling problem

The St. Petersburg Paradox

A simple game [Bernoulli, 1713]

- A fair coin is tossed until a head is obtained.
- If the first head is obtained on the n-th toss, our reward will be 2^n currency units.

The St. Petersburg Paradox

A simple game [Bernoulli, 1713]

- A fair coin is tossed until a head is obtained.
- If the first head is obtained on the n-th toss, our reward will be 2^n currency units.

How much are you willing to pay, to play this game once?

The St. Petersburg Paradox

A simple game [Bernoulli, 1713]

- A fair coin is tossed until a head is obtained.
- If the first head is obtained on the n-th toss, our reward will be 2^n currency units.

- The probability to stop at round n is 2^{-n}.

The St. Petersburg Paradox

A simple game [Bernoulli, 1713]

- A fair coin is tossed until a head is obtained.
- If the first head is obtained on the n-th toss, our reward will be 2^n currency units.

- The probability to stop at round n is 2^{-n}.
- Thus, the expected monetary gain of the game is

$$\sum_{n=1}^{\infty} 2^n 2^{-n} = \infty.$$

The St. Petersburg Paradox

A simple game [Bernoulli, 1713]

- A fair coin is tossed until a head is obtained.
- If the first head is obtained on the n-th toss, our reward will be 2^n currency units.

- The probability to stop at round n is 2^{-n}.
- Thus, the expected monetary gain of the game is

$$\sum_{n=1}^{\infty} 2^n 2^{-n} = \infty.$$

- If your utility function were linear you'd be willing to pay any amount to play.

Summary

- We can subjectively indicate which events we think are more likely.
- Using relative likelihoods, we can define a subjective probability P for all events.
- Similarly, we can subjectively indicate preferences for rewards.
- We can determine a utility function for all rewards.
- Hypothesis: we prefer the probability distribution (over rewards) with the highest expected utility.
- Concave utility functions imply risk aversion (and convex, risk-taking).

Experimental design and Markov decision processes

The following problems

- Shortest path problems.
- Optimal stopping problems.
- Reinforcement learning problems.
- Experiment design (clinical trial) problems
- Advertising.

can be all formalised as Markov decision processes.

Applications

- Robotics.
- Economics.
- Automatic control.
- Resource allocation

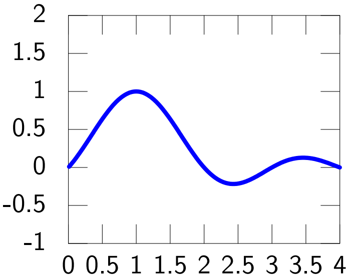
Contents

Bandit problems

Bandit problems

Applications

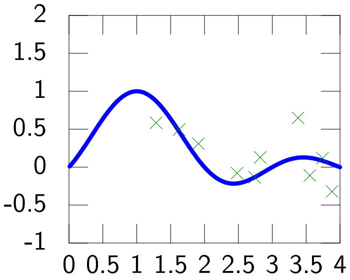
- Efficient optimisation.



Bandit problems

Applications

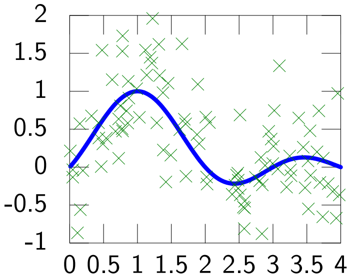
- Efficient optimisation.



Bandit problems

Applications

- Efficient optimisation.



Bandit problems

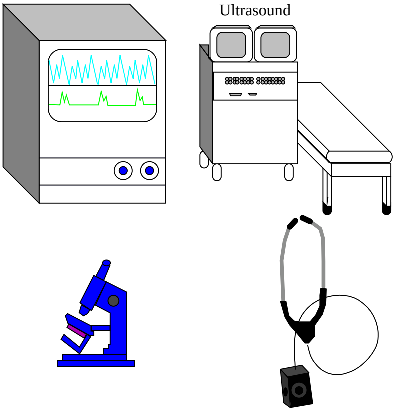
Applications

- Efficient optimisation.
- Online advertising.

Bandit problems

Applications

- Efficient optimisation.
- Online advertising.
- Clinical trials.



Bandit problems

Applications

- Efficient optimisation.
- Online advertising.
- Clinical trials.
- ROBOT SCIENTIST.

The stochastic n-armed bandit problem

Actions and rewards

- A set of actions $\mathcal{A} = \{1, \ldots, n\}$.
- Each action gives you a random reward with distribution $\mathbb{P}(r_t \mid a_t = i)$.
- The expected reward of the i-th arm is $\rho_i \triangleq \mathbb{E}(r_t \mid a_t = i)$.

Utility

The utility is the sum of the rewards obtained

$$U \triangleq \sum_t r_t.$$

Policy

Definition 9 (Policies)

A policy π is an algorithm for taking actions given the observed history.

$$\mathbb{P}^{\pi}(a_{t+1} \mid a_1, r_1, \ldots, a_t, r_t)$$

is the probability of the next action a_{t+1}.

Bernoulli bandits

Example 10 (Bernoulli bandits)

Consider n Bernoulli distributions with parameters ω_i $(i = 1, \ldots, n)$ such that $r_t \mid a_t = i \sim \mathcal{Bern}(\omega_i)$. Then,

$$\mathbb{P}(r_t = 1 \mid a_t = i) = \omega_i \qquad \mathbb{P}(r_t = 0 \mid a_t = i) = 1 - \omega_i \tag{2.1}$$

Then the expected reward for the i-th bandit is $\rho_i \triangleq \mathbb{E}(r_t \mid a_t = i) = ?$.

Bernoulli bandits

Example 10 (Bernoulli bandits)

Consider n Bernoulli distributions with parameters ω_i $(i = 1, \dots, n)$ such that $r_t \mid a_t = i \sim \mathcal{Bern}(\omega_i)$. Then,

$$\mathbb{P}(r_t = 1 \mid a_t = i) = \omega_i \qquad \mathbb{P}(r_t = 0 \mid a_t = i) = 1 - \omega_i \tag{2.1}$$

Then the expected reward for the i-th bandit is $\rho_i \triangleq \mathbb{E}(r_t \mid a_t = i) = \omega_i$.

Bernoulli bandits

Example 10 (Bernoulli bandits)

Consider n Bernoulli distributions with parameters ω_i $(i = 1, \ldots, n)$ such that $r_t \mid a_t = i \sim \mathcal{Bern}(\omega_i)$. Then,

$$\mathbb{P}(r_t = 1 \mid a_t = i) = \omega_i \qquad \mathbb{P}(r_t = 0 \mid a_t = i) = 1 - \omega_i \tag{2.1}$$

Then the expected reward for the i-th bandit is $\rho_i \triangleq \mathbb{E}(r_t \mid a_t = i) = \omega_i$.

Exercise 1 (The optimal policy under perfect knowledge)

If we know ω_i for all i, what is the best policy?

A *At every step, play the bandit i with the greatest ω_i.*

B *At every step, play the bandit i with probability increasing with ω_i.*

C *There is no right answer. It depends on the horizon T.*

D *It is too complicated.*

The unknown reward case

Say you keep a running average of the reward obtained by each arm

$$\hat{\rho}_{t,i} = R_{t,i}/n_{t,i}$$

where $n_{t,i}$ is the number of times you played arm i and $R_{t,i}$ the total reward received from i so that whenever you play $a_t = i$:

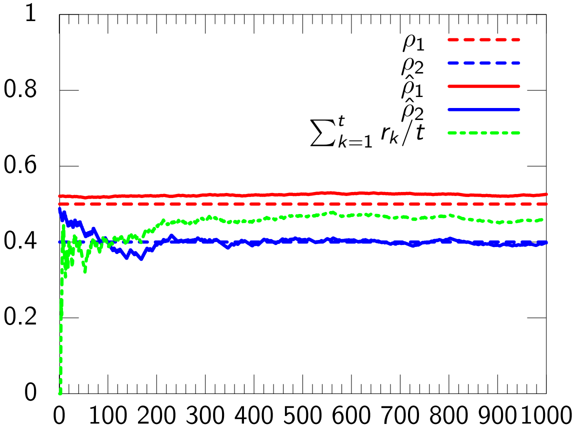
$$R_{t+1,i} = R_{t,i} + r_t, \qquad n_{t+1,i} = n_{t,i} + 1.$$

You could then choose to play the strategy

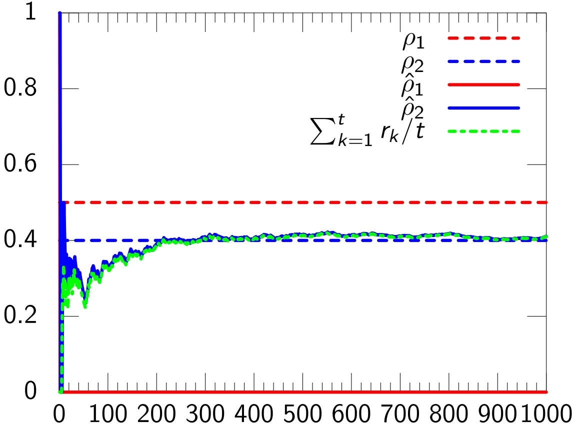
$$a_t = \arg\max_i \hat{\rho}_{t,i}.$$

What should the initial values $n_{0,i}, R_{0,i}$ be?

The uniform policy

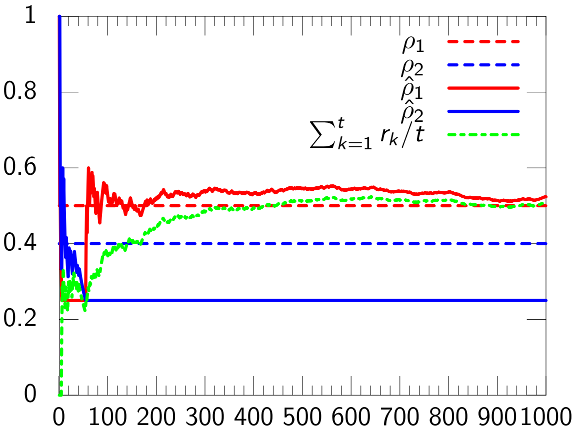


The greedy policy



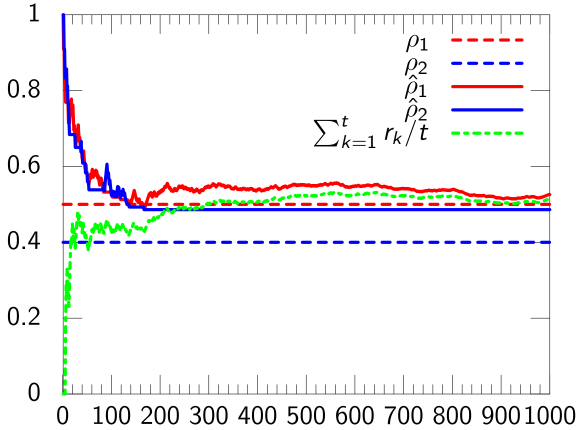
For $n_{0,i} = R_{0,i} = 0$

The greedy policy



For $n_{0,i} = R_{0,i} = 1$

The greedy policy

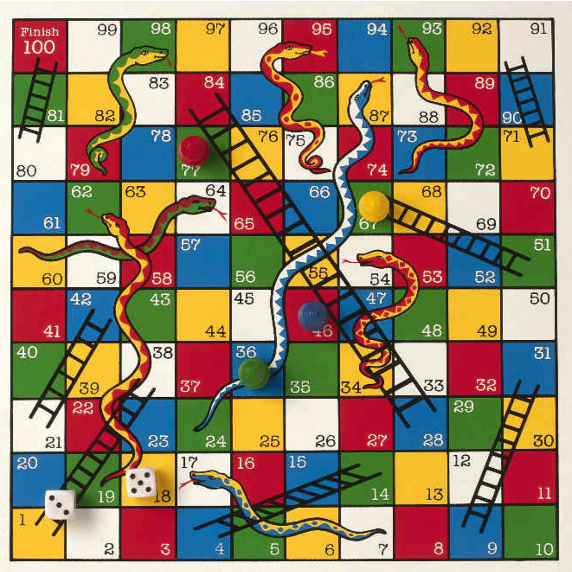


For

$n_{0,i} = R_{0,i} = 10$

Contents

A Markov processes



Markov process

$$s_{t-1} \longrightarrow s_t \longrightarrow s_{t+1}$$

Definition 11 (Markov Process – or Markov Chain)

The sequence $\{s_t \mid t = 1, \ldots\}$ of random variables $s_t : \Omega \to \mathcal{S}$ is a Markov process if

$$\mathbb{P}(s_{t+1} \mid s_t, \ldots, s_1) = \mathbb{P}(s_{t+1} \mid s_t). \tag{3.1}$$

- s_t is state of the Markov process at time t.
- $\mathbb{P}(s_{t+1} \mid s_t)$ is the transition kernel of the process.

The state of an algorithm

Observe that the R, n vectors of our greedy bandit algorithm form a Markov process. They also summarise our belief about which arm is the best.

Reinforcement learning

The reinforcement learning problem.

Learning to act in an unknown environment, by interaction and reinforcement.

- The environment has a changing state s_t.
- The agents observes the state s_t (simplest case).
- The agent takes action a_t.
- It receives rewards r_t.

The goal (informally)

Maximise total reward $\sum_t r_t$

Types of environments

- Markov decision processes (MDPs).
- Partially observable MDPs (POMDPs).
- (Partially observable) Markov games.

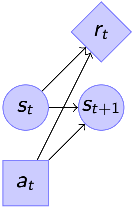
First deal with the case when μ is known.

Markov decision processes

Markov decision processes (MDP).

At each time step t:

- We observe state $s_t \in \mathcal{S}$.
- We take action $a_t \in \mathcal{A}$.
- We receive a reward $r_t \in \mathbb{R}$.



Markov property of the reward and state distribution

$$\mathbb{P}_\mu(s_{t+1} \mid s_t, a_t) \qquad \text{(Transition distribution)}$$

$$\mathbb{P}_\mu(r_t \mid s_t, a_t) \qquad \text{(Reward distribution)}$$

The agent

The agent's policy π

$$\mathbb{P}^\pi(a_t \mid s_t, \ldots, s_1, a_{t-1}, \ldots, a_1) \qquad \text{(history-dependent policy)}$$

$$\mathbb{P}^\pi(a_t \mid s_t) \qquad \text{(Markov policy)}$$

Definition 12 (Utility)

Given a horizon T, the utility can be defined as

$$U_t \triangleq \sum_{k=0}^{T-t} r_{t+k} \tag{3.2}$$

The agent wants to to find π maximising the expected total future reward

$$\mathbb{E}^\pi_\mu U_t = \mathbb{E}^\pi_\mu \sum_{k=0}^{T-t} r_{t+k}. \qquad \text{(expected utility)}$$

State value function

$$V^{\pi}_{\mu,t}(s) \triangleq \mathbb{E}^{\pi}_{\mu}(U_t \mid s_t = s) \tag{3.3}$$

The optimal policy π^*

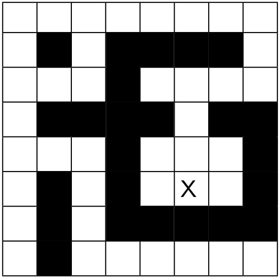
$$\pi^*(\mu) : V^{\pi^*(\mu)}_{t,\mu}(s) \geq V^{\pi}_{t,\mu}(s) \quad \forall \pi, t, s \tag{3.4}$$

dominates all other policies π everywhere in $\mathcal{S}$.
The optimal value function V^*

$$V^*_{t,\mu}(s) \triangleq V^{\pi^*(\mu)}_{t,\mu}(s), \tag{3.5}$$

is the value function of the optimal policy π^*.

Deterministic shortest-path problems



Properties

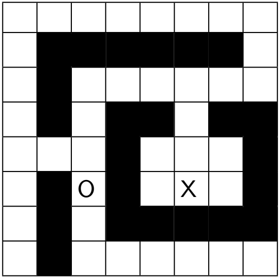
- $T \to \infty$.
- $r_t = -1$ unless $s_t = X$, in which case $r_t = 0$.
- $\mathbb{P}_\mu(s_{t+1} = X | s_t = X) = 1$.
- $\mathcal{A} = \{\text{North}, \text{South}, \text{East}, \text{West}\}$
- Transitions are deterministic and walls block.

14	13	12	11	10	9	8	7
15		13					6
16	15	14		4	3	4	5
17					2		
18	19	20		2	1	2	
19		21		1	0	1	
20		22					
21		23	24	25	26	27	28

Properties

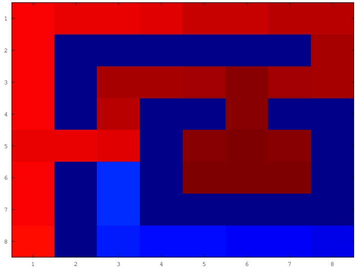
- $\gamma = 1,\ T \to \infty$.
- $r_t = -1$ unless $s_t = X$, in which case $r_t = 0$.
- The length of the shortest path from s equals the negative value of the optimal policy.
- Also called *cost-to-go*.

Stochastic shortest path problem with a pit

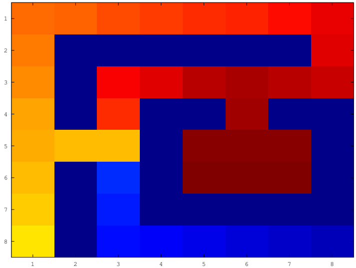


Properties

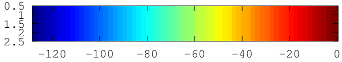
- $T \to \infty$.
- $r_t = -1$, but $r_t = 0$ at X and -100 at O and the problem ends.
- $\mathbb{P}_\mu(s_{t+1} = X | s_t = X) = 1$.
- $\mathcal{A} = \{\mathrm{North}, \mathrm{South}, \mathrm{East}, \mathrm{West}\}$
- Moves to a random direction with probability ω. Walls block.



(a) $\omega = 0.1$



(b) $\omega = 0.5$



(c) value

Figure: Pit maze solutions for two values of ω.

Exercise 2

- *Why should we only take the shortcut in (a)?*
- *Why does the agent commit suicide at the bottom?*

Contents

How to evaluate a policy

$$V^{\pi}_{\mu,t}(s) \triangleq \mathbb{E}^{\pi}_{\mu}(U_t \mid s_t = s) \tag{4.1}$$

$$\tag{4.2}$$

This derivation directly gives a number of policy evaluation algorithms.

How to evaluate a policy

$$V^{\pi}_{\mu,t}(s) \triangleq \mathbb{E}^{\pi}_{\mu}(U_t \mid s_t = s) \tag{4.1}$$

$$= \sum_{k=0}^{T-t} \mathbb{E}^{\pi}_{\mu}(r_{t+k} \mid s_t = s) \tag{4.2}$$

$$\tag{4.3}$$

This derivation directly gives a number of policy evaluation algorithms.

How to evaluate a policy

$$
\begin{aligned}
V^{\pi}_{\mu,t}(s) &\triangleq \mathbb{E}^{\pi}_{\mu}(U_t \mid s_t = s) && (4.1)\\
&= \sum_{k=0}^{T-t} \mathbb{E}^{\pi}_{\mu}(r_{t+k} \mid s_t = s) && (4.2)\\
&= \mathbb{E}^{\pi}_{\mu}(r_t \mid s_t = s) + \mathbb{E}^{\pi}_{\mu}(U_{t+1} \mid s_t = s) && (4.3)\\
& && (4.4)
\end{aligned}
$$

This derivation directly gives a number of policy evaluation algorithms.

How to evaluate a policy

$$V^{\pi}_{\mu,t}(s) \triangleq \mathbb{E}^{\pi}_{\mu}(U_t \mid s_t = s) \tag{4.1}$$

$$= \sum_{k=0}^{T-t} \mathbb{E}^{\pi}_{\mu}(r_{t+k} \mid s_t = s) \tag{4.2}$$

$$= \mathbb{E}^{\pi}_{\mu}(r_t \mid s_t = s) + \mathbb{E}^{\pi}_{\mu}(U_{t+1} \mid s_t = s) \tag{4.3}$$

$$= \mathbb{E}^{\pi}_{\mu}(r_t \mid s_t = s) + \sum_{i \in \mathcal{S}} V^{\pi}_{\mu,t+1}(i)\, \mathbb{P}^{\pi}_{\mu}(s_{t+1} = i | s_t = s). \tag{4.4}$$

This derivation directly gives a number of policy evaluation algorithms.

Monte-Carlo Policy evaluation

for $s \in \mathcal{S}$ **do**

end for

Monte-Carlo Policy evaluation

for $s \in \mathcal{S}$ **do**

  for $k = 1, \ldots, K$ **do**

    Execute policy π and record total reward K times:

$$\hat{R}_k(s) = \sum_{t=1}^{T} r_{t,k}.$$

  end for

end for

Monte-Carlo Policy evaluation

for $s \in \mathcal{S}$ **do**
  for $k = 1, \dots, K$ **do**
    Execute policy π and record total reward K times:

$$\hat{R}_k(s) = \sum_{t=1}^{T} r_{t,k}.$$

  end for
  Calculate estimate:

$$v_1(s) = \frac{1}{K} \sum_{k=1}^{K} \hat{R}_k(s).$$

end for

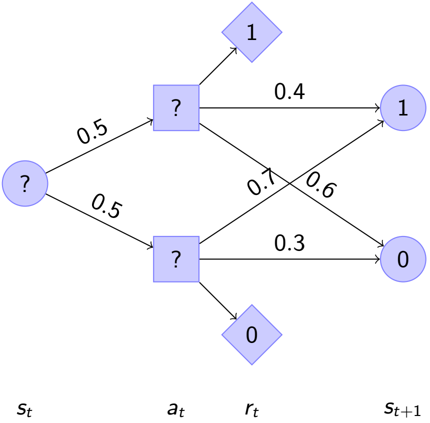
Backwards induction policy evaluation

for State $s \in S$, $t = T, \ldots, 1$ **do**

Update values

$$v_t(s) = \mathbb{E}^{\pi}_{\mu}(r_t \mid s_t = s) + \sum_{j \in S} \mathbb{P}^{\pi}_{\mu}(s_{t+1} = j \mid s_t = s) v_{t+1}(j), \tag{4.5}$$

end for



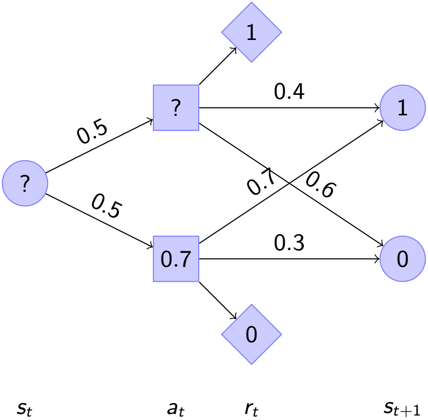
Backwards induction policy evaluation

for State $s \in S$, $t = T, \ldots, 1$ **do**

Update values

$$v_t(s) = \mathbb{E}^\pi_\mu(r_t \mid s_t = s) + \sum_{j \in S} \mathbb{P}^\pi_\mu(s_{t+1} = j \mid s_t = s) v_{t+1}(j), \tag{4.5}$$

end for



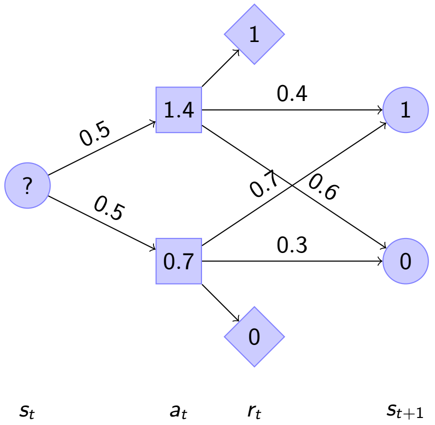
Backwards induction policy evaluation

for State $s \in S$, $t = T, \ldots, 1$ **do**

Update values

$$v_t(s) = \mathbb{E}^{\pi}_{\mu}(r_t \mid s_t = s) + \sum_{j \in S} \mathbb{P}^{\pi}_{\mu}(s_{t+1} = j \mid s_t = s) v_{t+1}(j), \tag{4.5}$$

end for



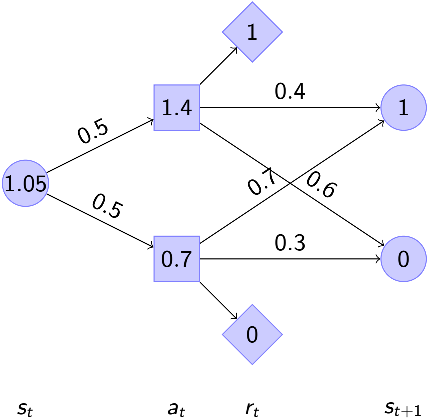
Backwards induction policy evaluation

for State $s \in S$, $t = T, \ldots, 1$ **do**

Update values

$$v_t(s) = \mathbb{E}^{\pi}_{\mu}(r_t \mid s_t = s) + \sum_{j \in S} \mathbb{P}^{\pi}_{\mu}(s_{t+1} = j \mid s_t = s) v_{t+1}(j), \tag{4.5}$$

end for



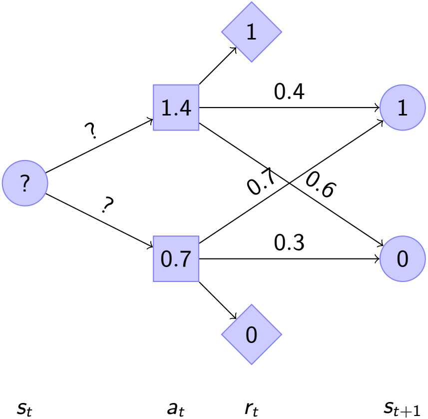
Backwards induction policy optimization

for State $s \in S$, $t = T, \ldots, 1$ **do**

Update values

$$v_t(s) = \max_a \mathbb{E}_\mu(r_t \mid s_t = s, a_t = a) + \sum_{j \in S} \mathbb{P}_\mu(s_{t+1} = j \mid s_t = s, a_t = a) v_{t+1}(j), \tag{4.6}$$

end for



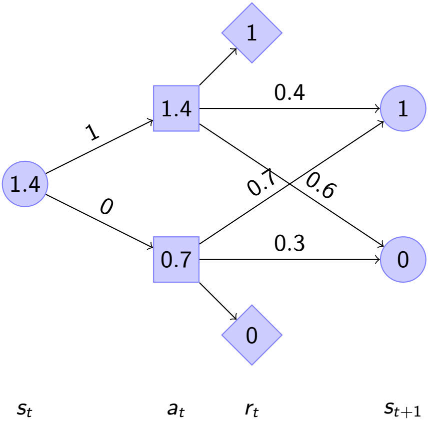
Backwards induction policy optimization

for State $s \in S$, $t = T, \ldots, 1$ **do**

Update values

$$v_t(s) = \max_a \mathbb{E}_\mu(r_t \mid s_t = s, a_t = a) + \sum_{j \in S} \mathbb{P}_\mu(s_{t+1} = j \mid s_t = s, a_t = a) v_{t+1}(j), \tag{4.6}$$

end for



Contents

Discounted total reward.

$$U_t = \lim_{T \to \infty} \sum_{k=t}^{T} \gamma^k r_k, \qquad \gamma \in (0, 1)$$

Definition 13

A policy π is stationary if $\pi(a_t \mid s_t)$ does not depend on t.

Remark 1

We can use the Markov chain kernel $\boldsymbol{P}_{\mu,\pi}$ to write the expected utility vector as

$$\boldsymbol{v}^\pi = \sum_{t=0}^{\infty} \gamma^t \boldsymbol{P}_{\mu,\pi}^t \boldsymbol{r} \tag{5.1}$$

Theorem 14

For any stationary policy π, $\boldsymbol{v}^\pi$ is the unique solution of

$$\boldsymbol{v} = \boldsymbol{r} + \gamma \boldsymbol{P}_{\mu,\pi}\boldsymbol{v}. \quad \leftarrow \textit{fixed point} \tag{5.2}$$

In addition, the solution is:

$$\boldsymbol{v}^\pi = (\boldsymbol{I} - \gamma \boldsymbol{P}_{\mu,\pi})^{-1}\boldsymbol{r}. \tag{5.3}$$

Example 15

Similar to the geometric series:

$$\sum_{t=0}^{\infty} \alpha^t = \frac{1}{1-\alpha}$$

Backward induction for discounted infinite horizon problems

- We can also apply backwards induction to the infinite case.
- The resulting policy is stationary.
- So memory does not grow with T.

Value iteration

for $n = 1, 2, \ldots$ and $s \in \mathcal{S}$ **do**
    $\boldsymbol{v}_n(s) = \max_a r(s, a) + \gamma \sum_{s' \in \mathcal{S}} P_\mu(s' \mid s, a) \boldsymbol{v}_{n-1}(s')$
end for

Policy Iteration

Input μ, $\mathcal{S}$.
Initialise $\boldsymbol{v}_0$.
for $n = 1, 2, \ldots$ **do**
  $\pi_{n+1} = \arg\max_\pi \{\boldsymbol{r} + \gamma \boldsymbol{P}_\pi \boldsymbol{v}_n\}$ `// policy improvement`
  $\boldsymbol{v}_{n+1} = V_\mu^{\pi_{n+1}}$ `// policy evaluation`
  break if $\pi_{n+1} = \pi_n$.
end for
Return $\pi_n, \boldsymbol{v}_n$.

Summary

- Markov decision processes model controllable dynamical systems.
- Optimal policies maximise expected utility can be found with:
  - Backwards induction / value iteration.
  - Policy iteration.
- The MDP state can be seen as
  - The state of a dynamic controllable process.
  - The internal state of an agent.

Contents

The reinforcement learning problem

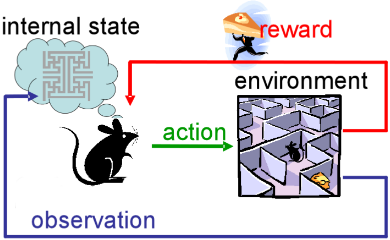
Learning to act in an unknown world, by interaction and reinforcement.

The reinforcement learning problem

Learning to act in an unknown world, by interaction and reinforcement.

World μ; Policy π; at time t

- μ generates observation $x_t \in \mathcal{X}$.
- We take action $a_t \in \mathcal{A}$ using π.
- μ gives us reward $r_t \in \mathbb{R}$.



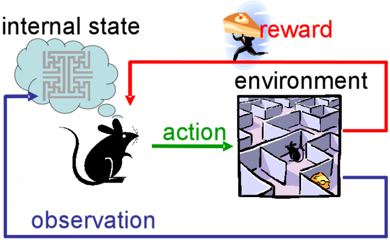
Learning by interaction

The reinforcement learning problem

Learning to act in an unknown world, by interaction and reinforcement.

World μ; Policy π; at time t

- μ generates observation $x_t \in \mathcal{X}$.
- We take action $a_t \in \mathcal{A}$ using π.
- μ gives us reward $r_t \in \mathbb{R}$.



Learning by interaction

Definition 16 (Utility)

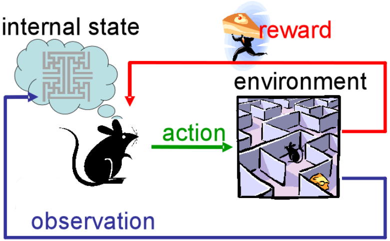
$$U_t = \sum_{k=t}^{T} r_k$$

The reinforcement learning problem

Learning to act in an unknown world, by interaction and reinforcement.

World μ; Policy π; at time t

- μ generates observation $x_t \in \mathcal{X}$.
- We take action $a_t \in \mathcal{A}$ using π.
- μ gives us reward $r_t \in \mathbb{R}$.



Learning by interaction

Definition 16 (Expected utility)

$$\mathbb{E}^{\pi}_{\mu} U_t = \mathbb{E}^{\pi}_{\mu} \sum_{k=t}^{T} r_k$$

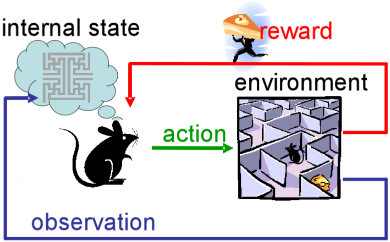
When μ is known, calculate $\max_{\pi} \mathbb{E}^{\pi}_{\mu} U$.

The reinforcement learning problem

Learning to act in an unknown world, by interaction and reinforcement.

World μ; Policy π; at time t

- μ generates observation $x_t \in \mathcal{X}$.
- We take action $a_t \in \mathcal{A}$ using π.
- μ gives us reward $r_t \in \mathbb{R}$.



Learning by interaction

Definition 16 (Expected utility)

$$\mathbb{E}^\pi_\mu U_t = \mathbb{E}^\pi_\mu \sum_{k=t}^{T} r_k$$

Knowing μ is contrary to the problem definition

When μ is not known

Bayesian idea: use a subjective belief $\xi(\mu)$ on $\mathcal{M}$

- Initial belief $\xi(\mu)$.

When μ is not known

Bayesian idea: use a subjective belief $\xi(\mu)$ on $\mathcal{M}$

- Initial belief $\xi(\mu)$.
- The probability of observing history h is $\mathbb{P}^{\pi}_{\mu}(h)$.

When μ is not known

Bayesian idea: use a subjective belief $\xi(\mu)$ on $\mathcal{M}$

- Initial belief $\xi(\mu)$.
- The probability of observing history h is $\mathbb{P}^{\pi}_{\mu}(h)$.
- We can use this to adjust our belief via Bayes' theorem:

$$\xi(\mu \mid h, \pi) \propto \mathbb{P}^{\pi}_{\mu}(h)\xi(\mu)$$

When μ is not known

Bayesian idea: use a subjective belief $\xi(\mu)$ on $\mathcal{M}$

- Initial belief $\xi(\mu)$.
- The probability of observing history h is $\mathbb{P}^\pi_\mu(h)$.
- We can use this to adjust our belief via Bayes' theorem:

$$\xi(\mu \mid h, \pi) \propto \mathbb{P}^\pi_\mu(h)\xi(\mu)$$

- We can thus conclude which μ is more likely.

When μ is not known

Bayesian idea: use a subjective belief $\xi(\mu)$ on $\mathcal{M}$

- Initial belief $\xi(\mu)$.
- The probability of observing history h is $\mathbb{P}^{\pi}_{\mu}(h)$.
- We can use this to adjust our belief via Bayes' theorem:

$$\xi(\mu \mid h, \pi) \propto \mathbb{P}^{\pi}_{\mu}(h)\xi(\mu)$$

- We can thus conclude which μ is more likely.

The subjective expected utility

$$\mathbb{E}^{\pi}_{\xi} U = \sum_{\mu} \left(\mathbb{E}^{\pi}_{\mu} U\right) \xi(\mu).$$

When μ is not known

Bayesian idea: use a subjective belief $\xi(\mu)$ on $\mathcal{M}$

- Initial belief $\xi(\mu)$.
- The probability of observing history h is $\mathbb{P}^\pi_\mu(h)$.
- We can use this to adjust our belief via Bayes' theorem:

$$\xi(\mu \mid h, \pi) \propto \mathbb{P}^\pi_\mu(h)\xi(\mu)$$

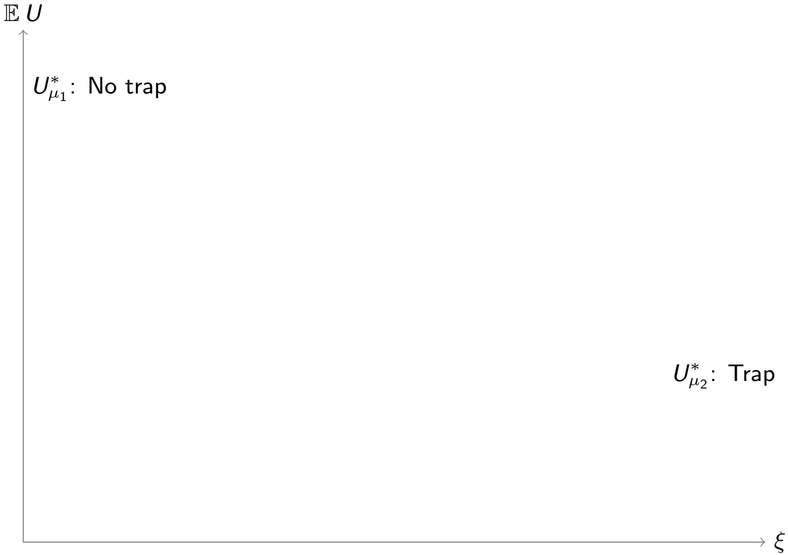
- We can thus conclude which μ is more likely.

The subjective expected utility

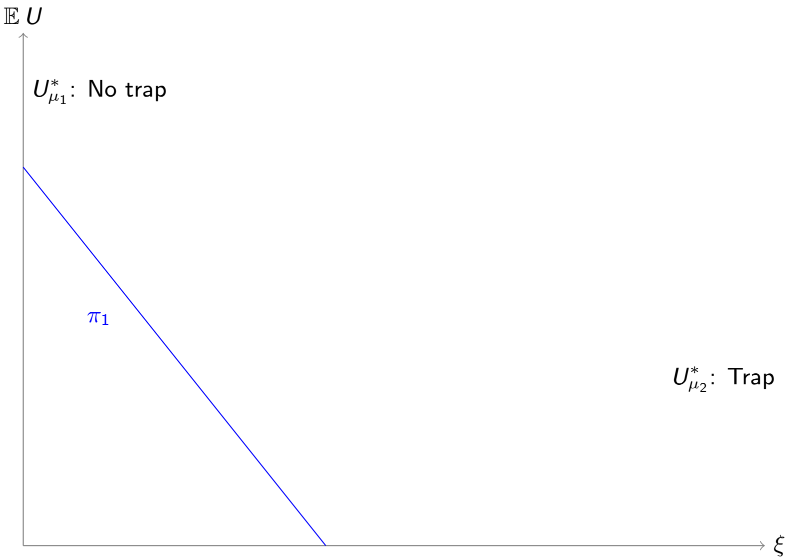
$$U^*_\xi \triangleq \max_\pi \mathbb{E}^\pi_\xi U = \max_\pi \sum_\mu \left(\mathbb{E}^\pi_\mu U\right) \xi(\mu).$$

Integrates planning and learning, and the exploration-exploitation trade-off

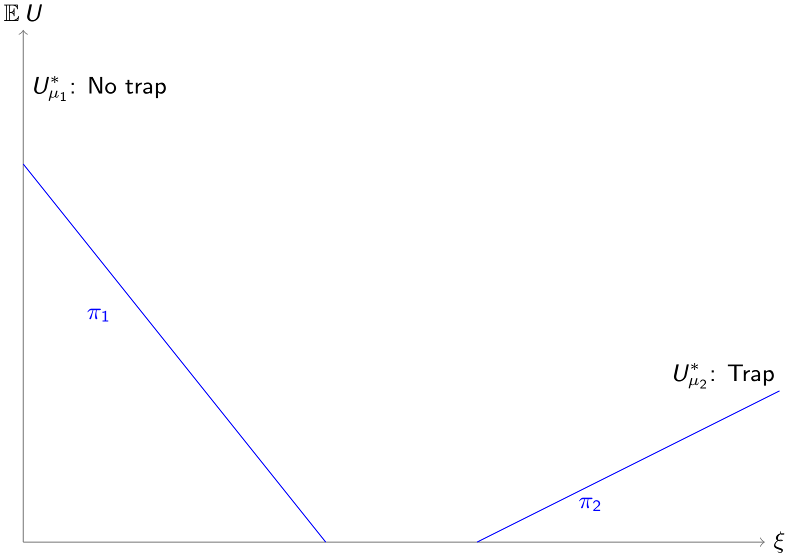
Bounds on the ξ-optimal utility $U^*_\xi \triangleq \max_\pi \mathbb{E}^\pi_\xi\, U$



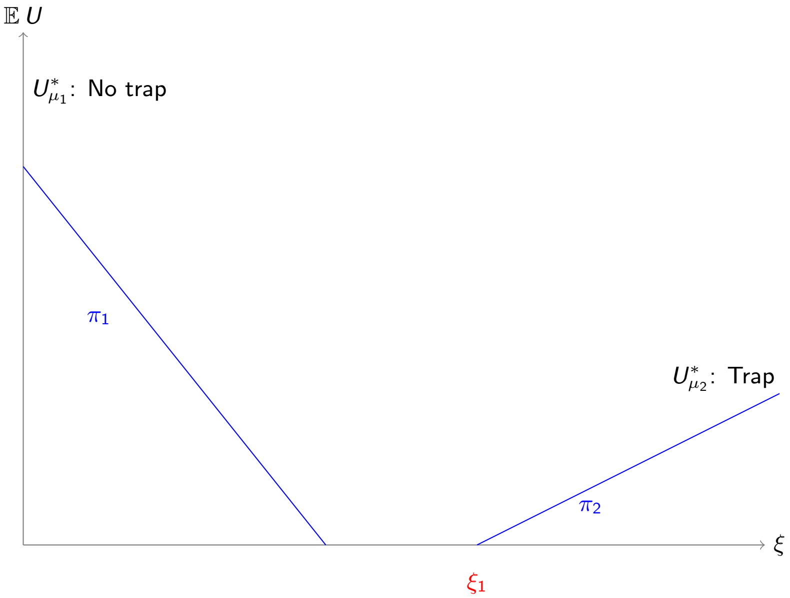
Bounds on the ξ-optimal utility $U^*_\xi \triangleq \max_\pi \mathbb{E}^\pi_\xi U$



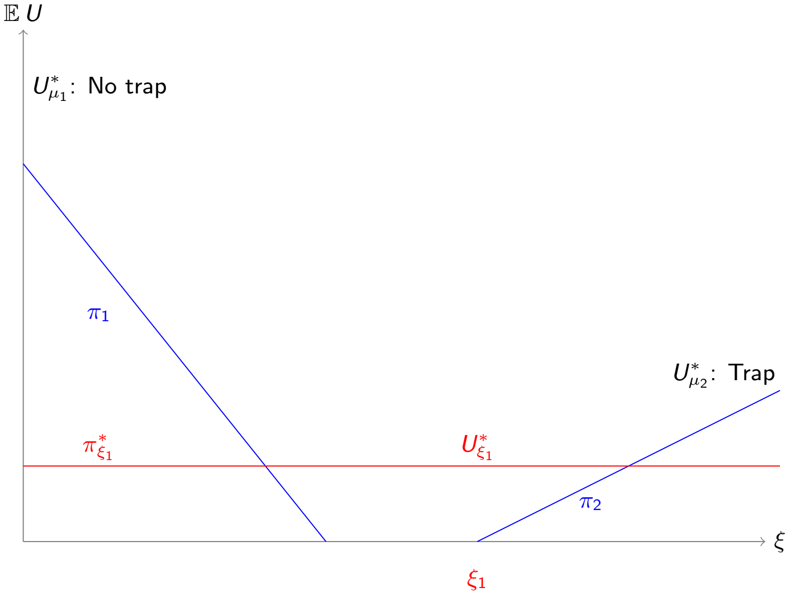
Bounds on the ξ-optimal utility $U^*_\xi \triangleq \max_\pi \mathbb{E}^\pi_\xi\, U$



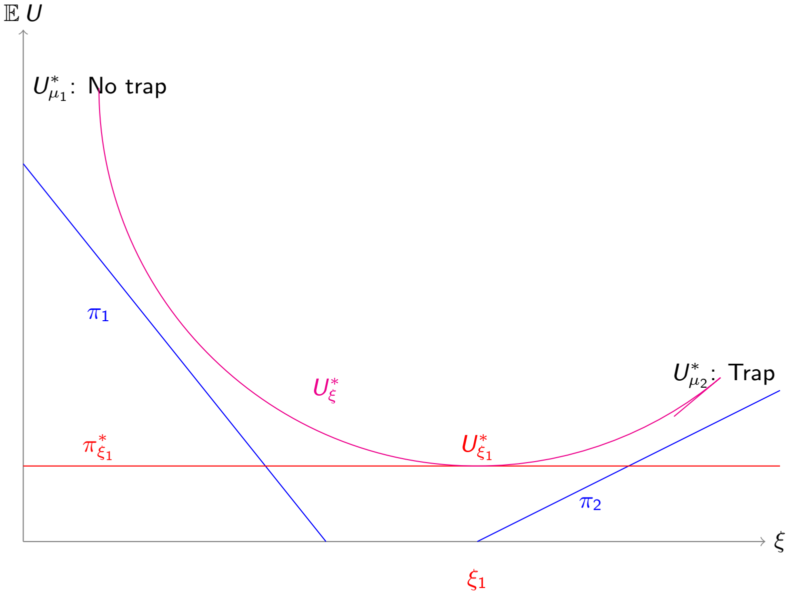
Bounds on the ξ-optimal utility $U^*_\xi \triangleq \max_\pi \mathbb{E}^\pi_\xi U$



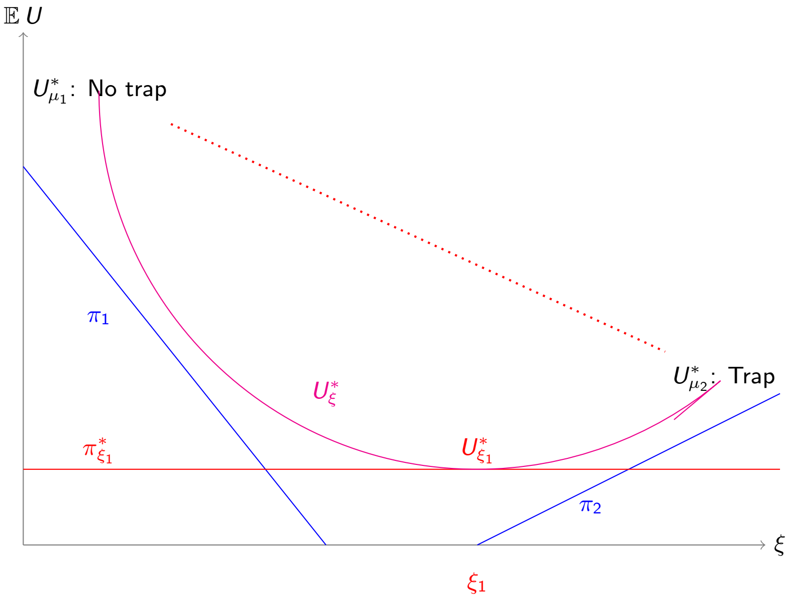
Bounds on the ξ-optimal utility $U^*_\xi \triangleq \max_\pi \mathbb{E}^\pi_\xi U$



Bounds on the ξ-optimal utility $U^*_\xi \triangleq \max_\pi \mathbb{E}^\pi_\xi U$



Bounds on the ξ-optimal utility $U^*_\xi \triangleq \max_\pi \mathbb{E}^\pi_\xi U$



ABC (Approximate Bayesian Computation) RL[1]

[1] Dimitrakakis, Tziortiotis. ABC Reinforcement Learning: ICML 2013

ABC (Approximate Bayesian Computation) RL[1]

How to deal with an arbitrary model space $\mathcal{M}$

- The models $\mu \in \mathcal{M}$ may be non-probabilistic simulators.
- We may not know how to choose the simulator parameters.

[1] Dimitrakakis, Tziortiotis. ABC Reinforcement Learning: ICML 2013

ABC (Approximate Bayesian Computation) RL[1]

How to deal with an arbitrary model space $\mathcal{M}$

- The models $\mu \in \mathcal{M}$ may be non-probabilistic simulators.
- We may not know how to choose the simulator parameters.

Overview of the approach

- Place a prior on the simulator parameters.
- Observe some data h on the real system.
- Approximate the posterior by statistics on simulated data.
- Calculate a near-optimal policy for the posterior.

[1] Dimitrakakis, Tziortiotis. ABC Reinforcement Learning: ICML 2013

ABC (Approximate Bayesian Computation) RL[1]

How to deal with an arbitrary model space $\mathcal{M}$

- The models $\mu \in \mathcal{M}$ may be non-probabilistic simulators.
- We may not know how to choose the simulator parameters.

Overview of the approach

- Place a prior on the simulator parameters.
- Observe some data h on the real system.
- Approximate the posterior by statistics on simulated data.
- Calculate a near-optimal policy for the posterior.

Results

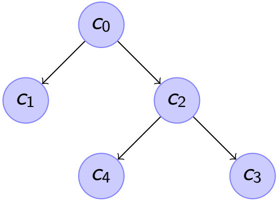
- We prove soundness with general properties on the statistics.
- In practice, can require much less data than a general model.

[1] Dimitrakakis, Tziortiotis. ABC Reinforcement Learning: ICML 2013

Cover tree Bayesian reinforcement learning

The model idea

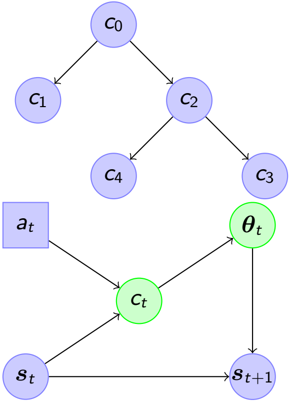
- Cover the space using a cover tree.



Cover tree Bayesian reinforcement learning

The model idea

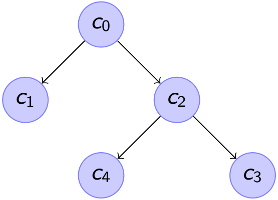
- Cover the space using a cover tree.
- A linear model for each set.



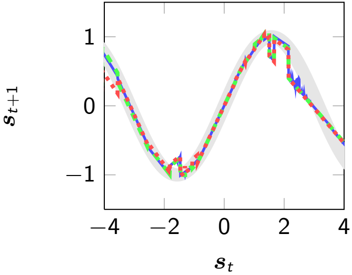
Cover tree Bayesian reinforcement learning

The model idea

- Cover the space using a cover tree.
- A linear model for each set.
- The tree defines a distribution on piecewise-linear models.



10^4 samples



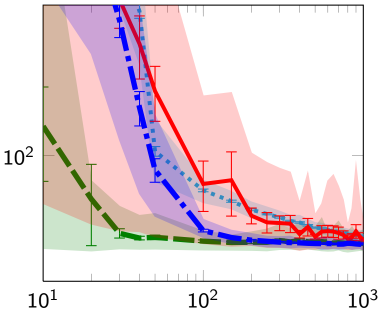
Cover tree Bayesian reinforcement learning

The model idea

- Cover the space using a cover tree.
- A linear model for each set.
- The tree defines a distribution on piecewise-linear models.

Algorithm overview

- Build the tree online
- Do Bayesian inference on the tree.
- Sample a model from the tree.
- Get a policy for the model.



A comparison

ABC RL

- Any simulator can be used $\Rightarrow$ enables detailed prior knowledge
- Our theoretical results prove soundness of ABC.
- Downside: Computationally intensive.

Cover Tree Bayesian RL

- Very general model.
- Inference in logarithmic time due to the tree strcuture.
- Downside: Hard to insert domain-specific prior knowledge.

Future work

Advanced algorithms (e.g. tree or gradient methods) for policy optimisation.

- Unknown MDPs can be handled in a Bayesian framework.
- This defines a belief-augmented MDP with
  - A state for the MDP.
  - A state for the agent's belief.
- The Bayes-optimal utility is convex, enabling approximations.
- A big problem in specifying the “right” prior.

Questions?

ABC (Approximate Bayesian Computation)

When there is no probabilistic model ($\mathbb{P}_\mu$ is not available): ABC!

- A prior ξ on a class of simulators $\mathcal{M}$
- History $h \in \mathcal{H}$ from policy π.
- Statistic $f : \mathcal{H} \to (\mathcal{W}, \| \cdot \|)$
- Threshold $\epsilon > 0$.

ABC (Approximate Bayesian Computation)

When there is no probabilistic model ($\mathbb{P}_\mu$ is not available): ABC!

- A prior ξ on a class of simulators $\mathcal{M}$
- History $h \in \mathcal{H}$ from policy π.
- Statistic $f : \mathcal{H} \to (\mathcal{W}, \| \cdot \|)$
- Threshold $\epsilon > 0$.

ABC (Approximate Bayesian Computation)

When there is no probabilistic model ($\mathbb{P}_\mu$ is not available): ABC!

- A prior ξ on a class of simulators $\mathcal{M}$
- History $h \in \mathcal{H}$ from policy π.
- Statistic $f : \mathcal{H} \to (\mathcal{W}, \|\cdot\|)$
- Threshold $\epsilon > 0$.

Example 17 (Cumulative features)

Feature function $\phi : \mathcal{X} \to \mathbb{R}^k$.

$$f(h) \triangleq \sum_t \phi(x_t)$$

ABC (Approximate Bayesian Computation)

When there is no probabilistic model ($\mathbb{P}_\mu$ is not available): ABC!

- A prior ξ on a class of simulators $\mathcal{M}$
- History $h \in \mathcal{H}$ from policy π.
- Statistic $f : \mathcal{H} \to (\mathcal{W}, \| \cdot \|)$
- Threshold $\epsilon > 0$.

Example 17 (Utility)

$$f(h) \triangleq \sum_t r_t$$

ABC (Approximate Bayesian Computation)

When there is no probabilistic model ($\mathbb{P}_\mu$ is not available): ABC!

- A prior ξ on a class of simulators $\mathcal{M}$
- History $h \in \mathcal{H}$ from policy π.
- Statistic $f : \mathcal{H} \to (\mathcal{W}, \| \cdot \|)$
- Threshold $\epsilon > 0$.

ABC (Approximate Bayesian Computation)

When there is no probabilistic model ($\mathbb{P}_\mu$ is not available): ABC!

- A prior ξ on a class of simulators $\mathcal{M}$
- History $h \in \mathcal{H}$ from policy π.
- Statistic $f : \mathcal{H} \to (\mathcal{W}, \|\cdot\|)$
- Threshold $\epsilon > 0$.

ABC-RL using Thompson sampling

- **do** $\hat{\mu} \sim \xi$, $h' \sim \mathbb{P}^{\pi}_{\hat{\mu}}$ // sample a model and history
- **until** $\|f(h') - f(h)\| \leq \epsilon$ // until the statistics are close
- $\mu^{(k)} = \hat{\mu}$ // approximate posterior sample $\mu^{(k)} \sim \xi_\epsilon(\cdot \mid h_t)$
- $\pi^{(k)} \approx \arg\max \mathbb{E}^{\pi}_{\mu^{(k)}} U_t$ // approximate optimal policy for sample

ABC (Approximate Bayesian Computation)

When there is no probabilistic model ($\mathbb{P}_\mu$ is not available): ABC!

- A prior ξ on a class of simulators $\mathcal{M}$
- History $h \in \mathcal{H}$ from policy π.
- Statistic $f : \mathcal{H} \to (\mathcal{W}, \|\cdot\|)$
- Threshold $\epsilon > 0$.

ABC-RL using Thompson sampling

- **do** $\hat{\mu} \sim \xi$, $h' \sim \mathbb{P}^{\pi}_{\hat{\mu}}$ // sample a model and history
- **until** $\|f(h') - f(h)\| \leq \epsilon$ // until the statistics are close
- $\mu^{(k)} = \hat{\mu}$ // approximate posterior sample $\mu^{(k)} \sim \xi_\epsilon(\cdot \mid h_t)$
- $\pi^{(k)} \approx \arg\max \mathbb{E}^{\pi}_{\mu^{(k)}} U_t$ // approximate optimal policy for sample

ABC (Approximate Bayesian Computation)

When there is no probabilistic model ($\mathbb{P}_\mu$ is not available): ABC!

- A prior ξ on a class of simulators $\mathcal{M}$
- History $h \in \mathcal{H}$ from policy π.
- Statistic $f : \mathcal{H} \to (\mathcal{W}, \|\cdot\|)$
- Threshold $\epsilon > 0$.

ABC-RL using Thompson sampling

- **do** $\hat{\mu} \sim \xi$, $h' \sim \mathbb{P}^{\pi}_{\hat{\mu}}$ // sample a model and history
- **until** $\|f(h') - f(h)\| \leq \epsilon$ // until the statistics are close
- $\mu^{(k)} = \hat{\mu}$ // approximate posterior sample $\mu^{(k)} \sim \xi_\epsilon(\cdot \mid h_t)$
- $\pi^{(k)} \approx \arg\max \mathbb{E}^{\pi}_{\mu^{(k)}} U_t$ // approximate optimal policy for sample

ABC (Approximate Bayesian Computation)

When there is no probabilistic model ($\mathbb{P}_\mu$ is not available): ABC!

- A prior ξ on a class of simulators $\mathcal{M}$
- History $h \in \mathcal{H}$ from policy π.
- Statistic $f : \mathcal{H} \to (\mathcal{W}, \|\cdot\|)$
- Threshold $\epsilon > 0$.

ABC-RL using Thompson sampling

- **do** $\hat{\mu} \sim \xi$, $h' \sim \mathbb{P}^\pi_{\hat{\mu}}$ // sample a model and history
- **until** $\|f(h') - f(h)\| \leq \epsilon$ // until the statistics are close
- $\mu^{(k)} = \hat{\mu}$ // approximate posterior sample $\mu^{(k)} \sim \xi_\epsilon(\cdot \mid h_t)$
- $\pi^{(k)} \approx \arg\max \mathbb{E}^\pi_{\mu^{(k)}} U_t$ // approximate optimal policy for sample

The approximate posterior $\xi_\epsilon(\cdot \mid h)$

Corollary 17

If f is a sufficient statistic and $\epsilon = 0$, then $\xi(\cdot \mid h) = \xi_\epsilon(\cdot \mid h)$.

The approximate posterior $\xi_\epsilon(\cdot \mid h)$

Corollary 17

If f is a sufficient statistic and $\epsilon = 0$, then $\xi(\cdot \mid h) = \xi_\epsilon(\cdot \mid h)$.

Assumption 4 (A1. Lipschitz log-probabilities)

For the policy π, $\exists L > 0$ s.t. $\forall h, h' \in \mathcal{H}$ and $\forall \mu \in \mathcal{M}$

$$\left| \ln \left[\mathbb{P}_\mu^\pi(h) / \mathbb{P}_\mu^\pi(h') \right] \right| \leq L \| f(h) - f(h') \|$$

The approximate posterior $\xi_\epsilon(\cdot \mid h)$

Corollary 17

If f is a sufficient statistic and $\epsilon = 0$, then $\xi(\cdot \mid h) = \xi_\epsilon(\cdot \mid h)$.

Assumption 4 (A1. Lipschitz log-probabilities)

For the policy π, $\exists L > 0$ s.t. $\forall h, h' \in \mathcal{H}$ and $\forall \mu \in \mathcal{M}$

$$\left| \ln \left[\mathbb{P}^\pi_\mu(h) / \mathbb{P}^\pi_\mu(h') \right] \right| \leq L \| f(h) - f(h') \|$$

Theorem 18 (The approximate posterior $\xi_\epsilon(\cdot \mid h)$ is close to $\xi(\cdot \mid h)$)

If A1 holds then $\forall \epsilon > 0$:

$$D\left(\xi(\cdot \mid h) \parallel \xi_\epsilon(\cdot \mid h)\right) \leq 2L\epsilon + \ln |A^h_\epsilon|, \tag{6.1}$$

where $A^h_\epsilon \triangleq \{z \in \mathcal{H} \mid \|f(z) - f(h)\| \leq \epsilon\}$.

The approximate posterior $\xi_\epsilon(\cdot \mid h)$

Corollary 17

If f is a sufficient statistic and $\epsilon = 0$, then $\xi(\cdot \mid h) = \xi_\epsilon(\cdot \mid h)$.

Assumption 4 (A1. Lipschitz log-probabilities)

For the policy π, $\exists L > 0$ s.t. $\forall h, h' \in \mathcal{H}$ and $\forall \mu \in \mathcal{M}$

$$\left|\ln \left[\mathbb{P}^\pi_\mu(h) / \mathbb{P}^\pi_\mu(h')\right]\right| \leq L \|f(h) - f(h')\|$$

Theorem 18 (The approximate posterior $\xi_\epsilon(\cdot \mid h)$ is close to $\xi(\cdot \mid h)$)

If A1 holds then $\forall \epsilon > 0$:

$$D\left(\xi(\cdot \mid h) \parallel \xi_\epsilon(\cdot \mid h)\right) \leq 2L\epsilon + \ln |A^h_\epsilon|, \tag{6.1}$$

where $A^h_\epsilon \triangleq \{z \in \mathcal{H} \mid \|f(z) - f(h)\| \leq \epsilon\}$.

[1] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2/3):235–256, 2002.

[2] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 2001.

[3] Dimitri P. Bertsekas and John N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[4] Herman Chernoff. Sequential design of experiments. *Annals of Mathematical Statistics*, 30(3):755–770, 1959.

[5] Herman Chernoff. Sequential Models for Clinical Trials. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol.4*, pages 805–812. Univ. of Calif Press, 1966.

[6] Morris H. DeGroot. *Optimal Statistical Decisions*. John Wiley & Sons, 1970.

[7] Milton Friedman and Leonard J. Savage. The expected-utility hypothesis and the measurability of utility. *The Journal of Political Economy*, 60(6):463, 1952.

[8] Marting L. Puterman. *Markov Decision Processes : Discrete Stochastic Dynamic Programming*. John Wiley & Sons, New Jersey, US, 1994.

[9] Leonard J. Savage. *The Foundations of Statistics*. Dover Publications, 1972.

[10] Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *ICML 2010*, 2010.

[11] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.