Lecture 6: State Abstractions

SUMS 707 - Basic Reinforcement Learning

Gabriela Moisescu-Pareja and Viet Nguyen

McGill University, Mila

February 25, 2021

Today

- Definition of a state abstraction
- Types of state abstractions
- Results related to abstractions

Motivation I

Abstraction is in general a mapping from one problem representation to a new representation, while preserving some properties.

Motivation II

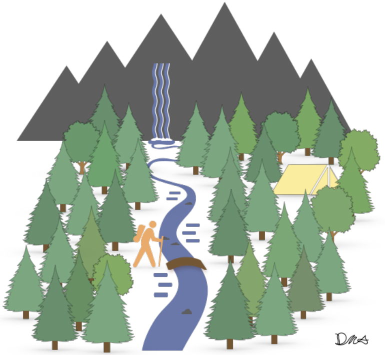
Abstraction is powerful.

Motivation III

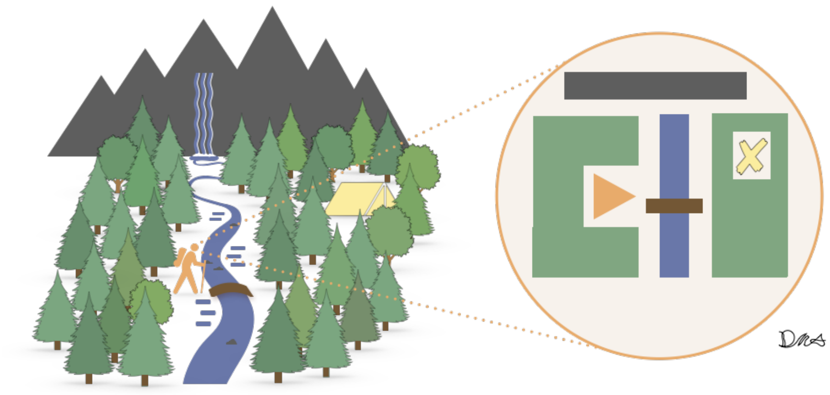
- On some level, we as humans are always performing abstraction.
- It is useful to discard irrelevant details in order to reason about things.
- We avoid getting bogged down by considering unnecessary details, which leads to faster decision-making.

Motivation IV

Consider a hiker:



Motivation V



Motivation VI

Can we do the same thing in reinforcement learning? Yes we can!

State Abstraction in RL I

How would we go about doing this in reinforcement learning?

- The abstraction is on the level of states.
- We want to **group states together**, where we consider this grouping to now represent a single state.
- This process is sometimes also referred to as state aggregation.

State Abstraction in RL II

So how would we do this grouping of states?

- In principal, we can group states together however we want.
- We just need to take care of some technical details to ensure that that we can define a proper MDP with the new state space consisting of aggregated states.
- After all, we want to do this to solve our original problem with RL methods!

Definition of State Abstraction I

We will focus on the tabular setting for now.

Definition

Suppose we have an MDP $M = (S, A, P, R)$. Let $\sim$ be an equivalence class on S, and let $S/\sim$ be the quotient set of S by $\sim$. We define the surjective map

$$\phi : S \to S/\sim$$

to be a **state abstraction**. Given a weight function w, such that for each $s' \in S'$ $\sum_{\{s:\phi(s)=s'\}} w(s) = 1$, we can construct the corresponding abstract MDP $\overline{M} = (\overline{S}, \overline{A}, \overline{P}, \overline{R})$, where $\overline{S} = S/\sim$.

Intuitively, w is measuring how much a low-level state contributes to the abstract state, and it is necessary to guarantee that $\overline{P}$ and $\overline{R}$ are well-defined.

Definition of State Abstraction II

The abstract MDP dynamics are:

$$\overline{R}^a_{\overline{s}} = \sum_{s \in \phi^{-1}(\overline{s})} w(s) R^a_s,$$

$$\overline{P}^a_{\overline{s},\overline{s}'} = \sum_{s \in \phi^{-1}(\overline{s})} \sum_{s' \in \phi^{-1}(\overline{s}')} w(s) P^a_{s,s'}$$

Exercise: Show that $\overline{P}^a_{\overline{s},\overline{s}'}$ is a well-defined next-state distribution, i.e. that $\sum_{\overline{s}'} \overline{P}^a_{\overline{s},\overline{s}'} = 1$, for any action a and $\overline{s} \in \overline{S}$.

Definition of State Abstraction III

Policies $\overline{\pi}$ in the abstract MDP are translated to policies π in the original MDP in the following way:

$$\pi(s, a) = \overline{\pi}(\phi(s), a)$$

for all s and a.

Definition of State Abstraction IV

The abstract value functions

- $V^{\overline{\pi}}(\overline{s})$
- $V^{*}(\overline{s})$
- $Q^{\overline{\pi}}(\overline{s}, z)$
- $Q^{*}(\overline{s}, a)$

are defined in the straightforward way.

5 Types of Abstractions I

So we've created an abstract MDP by the method above, which we want to work with due to its lower complexity. Questions we might ask ourselves:

- If we solved abstract MDP, how does its solution relate to the original MDP?
- *How do we group states in a "good" way?* i.e. How do we pick an abstraction?
- What information is lost when an abstraction is applied?

5 Types of Abstractions II

We will focus on abstractions that aim to preserve properties that are needed for an agent to make decisions that lead to optimal behaviour.

5 Types of Abstractions III

In presenting the next 5 abstraction types, we assume we are given an MDP $M = (S, A, P, R)$, and an arbitrary but fixed weighting function $w(s)$. For any states $s_1, s_2 \in S$:

5 Types of Abstractions IV

Definition

A **model-irrelevance** abstraction ϕ_{Model} is such that for any action a and any abstract state $\bar{s}$, $\phi_{\text{Model}}(s_1) = \phi_{\text{Model}}(s_2)$ implies

$$R^a_{s_1} = R^a_{s_2}$$
$$\sum_{s' \in \phi^{-1}_{\text{Model}}(\bar{s})} P^a_{s_1,s'} = \sum_{s' \in \phi^{-1}_{\text{Model}}(\bar{s})} P^a_{s_2,s'}$$

This abstraction preserves the one-step model dynamics.

5 Types of Abstractions V

Definition

A Q^π-**irrelevance** abstraction ϕ_{Q^π} is such that for any policy π and any action a,

$$\phi_{Q^\pi}(s_1) = \phi_{Q^\pi}(s_2)$$

implies

$$Q^\pi(s_1, a) = Q^\pi(s_2, a).$$

This abstraction preserves the state-action value function for all policies.

5 Types of Abstractions VI

Definition

A Q^*-**irrelevance** abstraction ϕ_{Q^*} is such that for any action a,

$$\phi_{Q^*}(s_1) = \phi_{Q^*}(s_2)$$

implies

$$Q^*(s_1, a) = Q^*(s_2, a).$$

This abstraction preserves the optimal state-action value function.

5 Types of Abstractions VII

Definition

An a^*-**irrelevance** abstraction ϕ_{a^*} is such that every abstract class has an action a^* that is optimal for all the states in that class, and

$$\phi_{a^*}(s_1) = \phi_{a^*}(s_2)$$

implies

$$Q^*(s_1, a^*) = \max_a Q^*(s_1, a) = \max_a Q^*(s_2, a) = Q^*(s_2, a^*).$$

This abstraction preserves the optimal action and its value.

Definition

An π^*-**irrelevance** abstraction ϕ_{π^*} is such that there is an optimal π^* for all the states in that class, and

$$\phi_{\pi^*}(s_1) = \phi_{\pi^*}(s_2)$$

implies

$$\pi^*(s_1) = \pi^*(s_2).$$

5 Types of Abstractions IX

We can define a partial order on the equivalence classes induced by the abstractions in the following way: ϕ_1 is *finer* than ϕ_2,

$$\phi_1 \geq \phi_2$$

iff for any states s_1, s_2, $\phi_1(s_1) = \phi_1(s_2)$ implies $\phi_2(s_1) = \phi_2(s_2)$. We say that they are comparable if $\phi_1 \geq \phi_2$ or $\phi_2 \geq \phi_1$.

Theorem

For any MDP, we have that

$$\phi_{Id} \geq \phi_{Model} \geq \phi_{Q^\pi} \geq \phi_{Q^*} \geq \phi_{a^*} \geq \phi_{\pi^*}$$

Therefore, if we prove a property about some abstraction, it automatically applies to the finer abstractions.

5 Types of Abstractions XI

Remember: We care about preserving aspects that ensure optimality. For what abstractions are we guaranteed that when we transfer the optimal policy in the abstract MDP, it is optimal in the original MDP?

5 Types of Abstractions XII

Theorem

With abstractions ϕ_{Model} , ϕ_{Q^π} , ϕ_{Q^*} and ϕ_{a^*} , the optimal abstract policy $\overline{\pi}*$ is optimal in the ground MDP. However, there exist examples where the optimal policy with abstraction ϕ_{π^*} is suboptimal in the ground MDP.

I'm skipping a few results on Q-learning due to a lack of time to properly cover them, but I will add them to the slides.

5 Types of Abstractions XIV

- Clearly coarser and coarser abstractions lose more information
- but they also result in a larger reduction in the state space, providing an increase of efficiency in solving the problem.

5 Types of Abstractions XV

There is a trade-off between *minimizing information loss* and *maximizing state space reduction* when selecting abstractions.

Some Examples of Abstractions

- Bisimulation
- MDP Homomorphisms
- π-Bisimulation

Bisimulation I

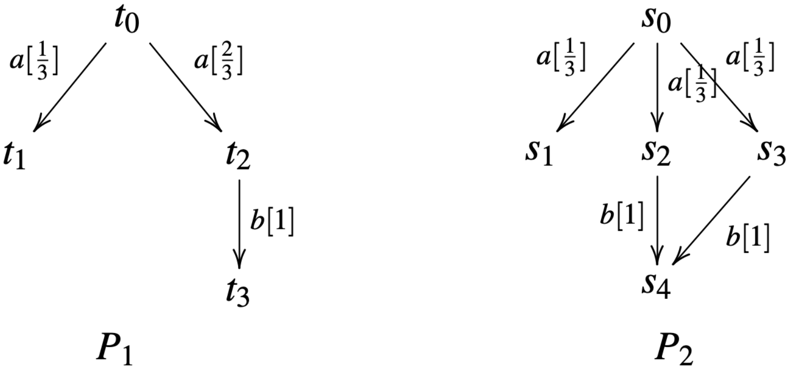


Figure: Probabilistic transition system.

Bisimulation II

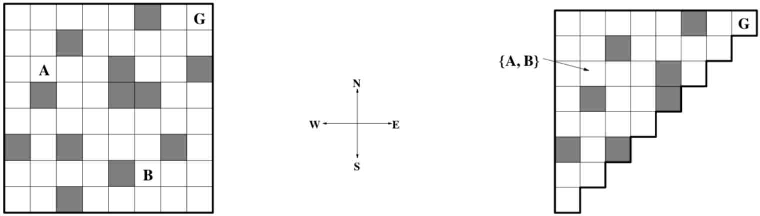
Definition

Given an MDP M, an equivalence relation $E \subseteq S \times S$ is a **bisimulation relation** if whenever $(s, t) \in E$, the following properties hold:

$$R_s^a = R_t^a$$
$$\forall C \in S_E, P_s^a(C) = P_t^a(C)$$

Two states $s, t \in S$ are **bisimilar** if there exists a bisimulation relation E such that $(s, t) \in E$.

MDP Homomorphisms I



(a) (b)

Figure: Abstraction combines together symmetrically equivalent states that also share dynamics and rewards

MDP Homomorphisms II

Definition

An *MDP homomorphism h* from an MDP $M = (S, A, \Psi, P, R)$ to an MDP $M' = (S, A, \Psi', P', R')$ is a surjection from Ψ to Ψ', defined by a tuple of surjections $(f, \{g_s | s \in S\})$, with $h((s,a)) = (f(s), g_s(a))$, where $f : S \to S'$ and $g_s : A_s \to A'_{f(s)}$ for $s \in S$, such that:

$$P'(f(s_1), g_{s_1}(a), f(s_2)) = T(s_1, a, [s_2]_{B_h|S}), \forall s_1, s_2 \in S, a \in A_{s_1} \quad (1)$$

$$R'(f(s_1), g_{s_1}(a)) = R(s_1, a), \forall s_1 \in S, a \in A_{s_1} \quad (2)$$

In practice?

- All the abstractions we saw today are in some way *exact*
- We form equivalence classes based on whether states satisfy some specific property.
    - For example, model irrelevance abstractions like bisimulation and MDP homomorphisms must agree exactly on dynamics. However, this seems fickle.
- In practice, states might not match up exactly.
- We need to group together states that are different under these conditions, but still "close" in some way.

Next lecture

- Approximate Abstractions
- Metrics
- Temporal abstraction