ISE/ECE 7202: Reinforcement Learning

Ohio State University, Autumn 2021

Parinaz Naghizadeh
naghizadeh.1@osu.edu

Lecture 2: Main Terminology and an Extended Example

August 26, 2021

Outline

- General concepts in sequential decision making
- Example: Gridworld

Previous lecture

- Introduction to reinforcement learning
- Relation to other fields
  - Roots in DP. Can choose RL instead of DP when model unknown or not learnable, and to address the curse of dimensionality.
  - Different from other ML paradigms due to evaluative feedback and sequential nature. Raises the challenge of exploration vs exploitation.
- Main elements of an RL problem: agent, action, environment, state, reward, model, policy, and value functions.

Previous lecture

- Introduction to reinforcement learning
- Relation to other fields
  - Roots in DP. Can choose RL instead of DP when model unknown or not learnable, and to address the curse of dimensionality.
  - Different from other ML paradigms due to evaluative feedback and sequential nature. Raises the challenge of exploration vs exploitation.
- Main elements of an RL problem: agent, action, environment, state, reward, model, policy, and value functions.

Sequential decision making

Recall that a key feature of reinforcement learning is its sequential nature. In a sequential decision making problem

- The agent is making decisions over time, with the goal of maximizing (a notion of) long-run reward
- Agents' actions have consequences, and in particular, may change the future states of the environment
- Rewards may only be realized at a future state
- Even if getting immediate rewards at each state, the agent may forego high immediate rewards when accounting for long-term effects

Sequential decision making

Recall that a key feature of reinforcement learning is its sequential nature. In a sequential decision making problem

- The agent is making decisions over time, with the goal of maximizing (a notion of) long-run reward
- Agents' actions have consequences, and in particular, may change the future states of the environment
- Rewards may only be realized at a future state
- Even if getting immediate rewards at each state, the agent may forego high immediate rewards when accounting for long-term effects

Examples: playing chess, investing in the stock market, a robot on a disaster recovery mission.

The Agent-Environment Interaction

At each time t:

- The agent observes s_t
- The agent takes action a_t
- The environment generates a reward r_t
- The environment transitions to state s_{t+1}

The Agent-Environment Interaction

At each time t:

- The agent observes s_t
- The agent takes action a_t
- The environment generates a reward r_t
- The environment transitions to state s_{t+1}

In this course, we assume the state is Markovian, i.e.,

$$\mathbb{P}[S_{t+1} \mid S_t] = \mathbb{P}[S_{t+1} \mid S_1, S_2, \ldots, S_t]$$

The Agent-Environment Interaction

At each time t:

- The agent observes s_t
- The agent takes action a_t
- The environment generates a reward r_t
- The environment transitions to state s_{t+1}

In this course, we assume the state is Markovian, i.e.,

$$\mathbb{P}[S_{t+1} \mid S_t] = \mathbb{P}[S_{t+1} \mid S_1, S_2, \ldots, S_t]$$

Q: What is a Markovian state for the rescue robot example?

The Agent-Environment Interaction

At each time t:

- The agent observes s_t
- The agent takes action a_t
- The environment generates a reward r_t
- The environment transitions to state s_{t+1}

In this course, we assume the state is Markovian, i.e.,

$$\mathbb{P}[S_{t+1} | \ S_t] = \mathbb{P}[S_{t+1} | \ S_1, S_2, \ldots, S_t]$$

Q: What is a Markovian state for the rescue robot example? **Q:** What about a game of chess?

The Agent-Environment Interaction

At each time t:

- The agent observes o_t
- The agent takes action a_t
- The environment generates a reward r_t
- The environment omits observation o_{t+1}

In this course, we assume the state is Markovian, i.e.,

$$\mathbb{P}[S_{t+1} |\ S_t] = \mathbb{P}[S_{t+1} |\ S_1, S_2, \ldots, S_t]$$

Q: What is a Markovian state for the rescue robot example? **Q:** What about a game of chess?

Observations vs states

More on the state

- The history is a sequence of all variables observed up to time t

$$H_t := \{o_1, a_1, r_1, o_2, a_2, r_2, \ldots, o_{t-1}, a_{t-1}, r_{t-1}\} \; .$$

- Observations vs states: Not everything in the history matters for determining what happens next. The part that does, is the state. Formally, the state is a function of the history.

More on the state

- The history is a sequence of all variables observed up to time t

$$H_t := \{o_1, a_1, r_1, o_2, a_2, r_2, \ldots, o_{t-1}, a_{t-1}, r_{t-1}\} \; .$$

- Observations vs states: Not everything in the history matters for determining what happens next. The part that does, is the state. Formally, the state is a function of the history.

More on the state

- The history is a sequence of all variables observed up to time t

$$H_t := \{o_1, a_1, r_1, o_2, a_2, r_2, \ldots, o_{t-1}, a_{t-1}, r_{t-1}\}\ .$$

- Observations vs states: Not everything in the history matters for determining what happens next. The part that does, is the state. Formally, the state is a function of the history.

- There are different states: the environment state (the information the environment uses to determine its evolution) vs. the agent state (what the agent/RL algorithm knows and uses for making decisions).

More on the state

- The history is a sequence of all variables observed up to time t

$$H_t := \{o_1, a_1, r_1, o_2, a_2, r_2, \ldots, o_{t-1}, a_{t-1}, r_{t-1}\} \ .$$

- Observations vs states: Not everything in the history matters for determining what happens next. The part that does, is the state. Formally, the state is a function of the history.

- There are different states: the environment state (the information the environment uses to determine its evolution) vs. the agent state (what the agent/RL algorithm knows and uses for making decisions).

- The environment state is, by definition, an information (Markov) state.

More on the state

- The history is a sequence of all variables observed up to time t

$$H_t := \{o_1, a_1, r_1, o_2, a_2, r_2, \ldots, o_{t-1}, a_{t-1}, r_{t-1}\} .$$

- Observations vs states: Not everything in the history matters for determining what happens next. The part that does, is the state. Formally, the state is a function of the history.
- There are different states: the environment state (the information the environment uses to determine its evolution) vs. the agent state (what the agent/RL algorithm knows and uses for making decisions).
- The environment state is, by definition, an information (Markov) state.
- In a perfectly observable MDP, the agent state is the environment state. In partially observable MDPs, the agent has to identify its own state representation (all the history, belief states, etc).

The Agent-Environment Interaction

The other components of RL

- **Model**: What the environment does next. Includes:
  - State transitions $p(s, a, s') = \mathbb{P}(S_{t+1} = s' | S_t = s, A_t = a)$.
  - Reward function $r(s, a) = \mathbb{E}(R_t | S_t = s, A_t = a)$.

The other components of RL

- **Model**: What the environment does next. Includes:
  - State transitions $p(s, a, s') = \mathbb{P}(S_{t+1} = s' | S_t = s, A_t = a)$.
  - Reward function $r(s, a) = \mathbb{E}(R_t | S_t = s, A_t = a)$.
- **Policy**: How will the agent behave?

$$\pi(s) = a \qquad \text{(deterministic)},$$

$$\pi(a|s) = \mathbb{P}[A_t = a|\ S_t = s] \qquad \text{(stochastic)}.$$

The other components of RL

- **Model**: What the environment does next. Includes:
  - State transitions $p(s, a, s') = \mathbb{P}(S_{t+1} = s' | S_t = s, A_t = a)$.
  - Reward function $r(s, a) = \mathbb{E}(R_t | S_t = s, A_t = a)$.
- **Policy**: How will the agent behave?

$$\pi(s) = a \qquad \text{(deterministic)},$$

$$\pi(a|s) = \mathbb{P}[A_t = a|\ S_t = s] \qquad \text{(stochastic)}.$$

- **Value function**: what is the long-term "goodness" of a state?

$$v_\pi(s) = \mathbb{E}_\pi[R_t + \delta R_{t+1} + \delta^2 R_{t+2} + \ldots\ |S_t = s]\ ,$$

where δ is a discount factor.

The other components of RL

- **Model**: What the environment does next. Includes:
  - State transitions $p(s, a, s') = \mathbb{P}(S_{t+1} = s' | S_t = s, A_t = a)$.
  - Reward function $r(s, a) = \mathbb{E}(R_t | S_t = s, A_t = a)$.
- **Policy**: How will the agent behave?

$$\pi(s) = a \qquad \text{(deterministic)},$$

$$\pi(a|s) = \mathbb{P}[A_t = a|\ S_t = s] \qquad \text{(stochastic)}.$$

- **Value function**: what is the long-term "goodness" of a state?

$$v_\pi(s) = \mathbb{E}_\pi[R_t + \delta R_{t+1} + \delta^2 R_{t+2} + \dots\ |S_t = s]\ ,$$

  where δ is a discount factor.

We'll see these notions many more time throughout the course. We'll look at them more carefully in our lectures on MDPs and DP.

Example: Grid World

What are the agent, the environment, the state, the actions, and the state transitions? What about rewards?

Example 1: Grid World – Policy

What is the optimal policy?

Example 1: Grid World – Value functions

How about the value functions? Can we use them to determine the optimal policy?

Example 1: Grid World – Model

Finally, what if the agent builds a model as it progresses through the environment?

Categories of RL algorithms

Based on the way the agent learns:

- Value-based (policy is implicit)
- Policy-based (value functions not explicitly calculated)
- Both: Actor-Critic

Categories of RL algorithms

Based on the way the agent learns:

- Value-based (policy is implicit)
- Policy-based (value functions not explicitly calculated)
- Both: Actor-Critic

Another categorization:

- Model-free (no model, just value and/or policy functions)
- Model-based (estimate the model as well)
- Note: learning vs planning

Example: Stochastic Grid World

What if the agent's actions are unreliable?

Next lecture

- Multi-armed bandit problems

- **Homework 0** due next Tuesday