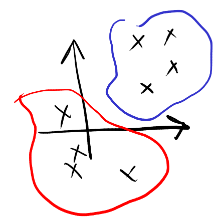
CS 4/5789 24 Jan 2022 Prof Sarah Dean

Lecture 1: Introduction

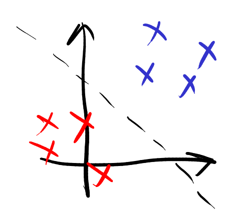
Slides: 1) Examples
2) Logistics

3) Types of Machine Learning



i) unsupervised learning
→ goal: summarization — ex- PCA, clustering
→ dataset: $\{x_i\}_{i=1}^N$
→ evaluation: qualitative — "descriptive"

ii) supervised
→ goal: prediction — ex- classification, regression
→ dataset: $\{(x_i, y_i)\}_{i=1}$ (features: x_i; labels: y_i)
→ evaluation: accuracy — "predictive"
$\hat{y}_i$ vs. y_i



iii) Reinforcement learning

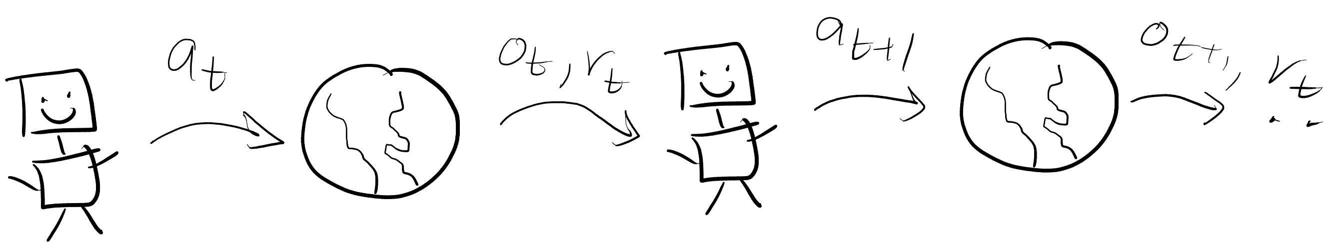
→ goal: action/decision

→ dataset: $\{(o_t, a_t, r_t)\}_{t=1}^{N}$ "prescriptive"

observation action reward sequential

→ evaluation: cumulative reward

Unlike supervised/unsupervised learing, data is not drawn iid from some distribution. Instead, it arrives sequentially.

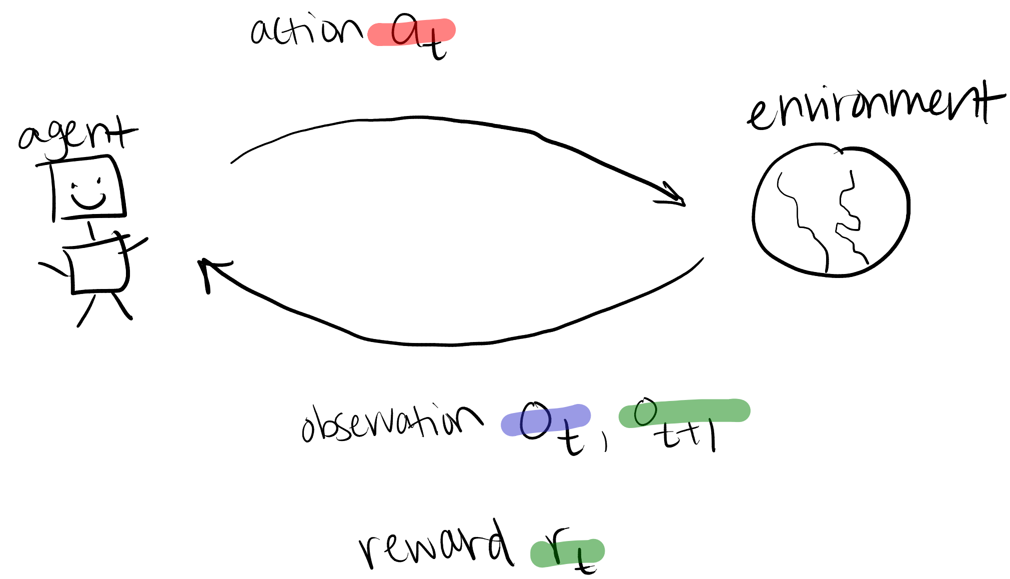


1) may start with <u>no data</u>
2) actions have <u>consequences</u> — will affect future observations and rewards.
3) solving a task requires <u>long sequence</u> of correct actions

4) Markov Decision Processes (MDP)

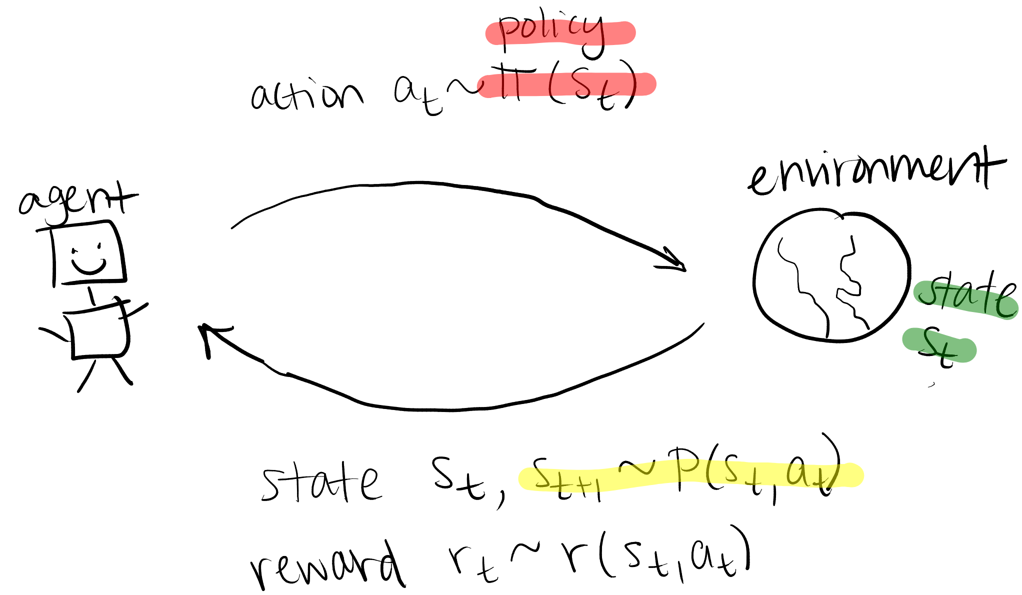
First, recall our general setup:

1) agent observes environment
2) agent takes action
3) environment changes and sends reward



In a Markov Decision Process, we have more structure:

Environment has a state which updates (stochastically) depending on previous state and action according to transition function



Markovian assumption: $P\{s_{t+1} = s \mid s_t, s_{t-1}, \ldots, s_0, a_t, \ldots a_0\}$
$$= P\{s_{t+1} = s \mid s_t, a_t\}$$

In this class, we usually asume state is observed ($o_t = s_t$)

Actions determined by state according to policy

Example: Robot manipulation

State s : finger configuration and object pose

action a : joint motor commands

transition : physics (gravity, contact forces, friction)
$s' \sim P(s,a)$

Policy $\pi(s)$: map from configuration to motor commands

Reward $r(s,a)$: negative distance to goal
(other factors: torque magnitude, dropping object, etc)

Question: if there are S states and A actions, how many policies are there?

Answer: since we can choose to map each s to A many actions, A^S

Infinite Horizon Discounted MDP

$$\mathcal{M} = \{\mathcal{S}, \mathcal{A}, P, r, \gamma\}$$

$\mathcal{S}$: space of possible states $s \in \mathcal{S}$

$\mathcal{A}$: space of possible actions $a \in \mathcal{A}$

P : transition function $P : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$

r : reward function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

γ : discount factor $0 < \gamma < 1$

In this notation we can write the goal:

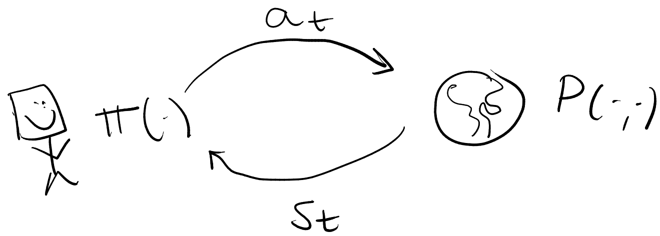
finding a policy $\pi : \mathcal{S} \to \mathcal{A}$
that maximizes the (discounted)
cumulative reward.

$$\boxed{\begin{aligned} &\underset{\pi}{\text{maximize}} \quad \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right] \\ &\text{s.t.} \quad s_{t+1} \sim P(s_t, a_t), \; s_0 \sim P_0 \\ &\qquad\quad a_t \sim \pi(s_t) \end{aligned}}$$

We will spend the semester learning how to solve this problem. In RL, we do not assume that $P(\cdot,\cdot)$ is known, and therefore we have to solve the optimization using data.

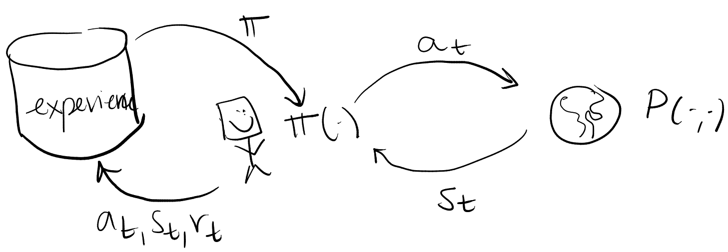
5) Layers of Feedback in RL

1) control feedback "reaction"



- feedback between states & actions
- historically studied in control theory
  "automatic feedback control"
  ex – thermostat regulates temperature
- we focus on this level for unit 1

2) Data Feedback "adaptation"



- feedback between policy and data
- connections to machine learning
  ex – smart thermostat learns preferences
- we consider this level starting in Unit 2

Recap of Today's Lecture

1) RL solves sequential decision-making problems
2) RL is different from supervised & unsupervised learning
3) Markov Decision Process setting for RL
4) There are two levels of feedback in RL agents