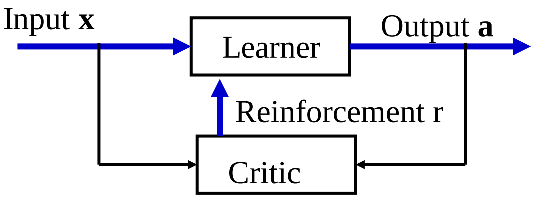
CS 2750 Machine Learning
Lecture 21b

Reinforcement learning

Milos Hauskrecht
milos@cs.pitt.edu
5329 Sennott Square

Reinforcement learning

- **We want to learn the control policy:** $\pi : X \rightarrow A$
- We see examples of **x** (but outputs *a* are not given)
- Instead of *a* we get a feedback *r* (reinforcement, reward) from a **critic** quantifying how good the selected output was



- The reinforcements may not be deterministic
- **Goal:** find $\pi : X \rightarrow A$ with the best expected reinforcements

Gambling example.

- **Game:** 3 different biased coins are tossed
  - The coin to be tossed is selected randomly from the three options and I always see which coin I am going to play next
  - I make bets on head or tail and I always wage $1
  - If I win I get $1, otherwise I lose my bet
- **RL model:**
  - **Input:** X – a coin chosen for the next toss,
  - **Action:** A – choice of head or tail,
  - **Reinforcements:** {1, -1}
- **A policy** $\pi : X \rightarrow A$

Example: $\pi :$

Coin1 →	***head***
Coin2 →	***tail***
Coin3 →	***head***

Gambling example

- **RL model:**
  - **Input:** X – a coin chosen for the next toss,
  - **Action:** A – choice of head or tail,
  - **Reinforcements:** {1, -1}
  - **A policy** $\pi :$

Coin1 →	***head***
Coin2 →	***tail***
Coin3 →	***head***

- **Learning goal: find** $\pi : X \rightarrow A$ $\quad \pi :$

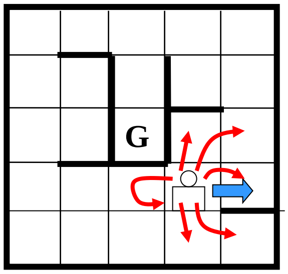
Coin1 →	**?**
Coin2 →	**?**
Coin3 →	**?**

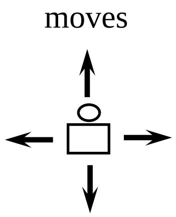
maximizing future expected profits

$E(\sum_{t=0}^{\infty} \gamma^t r_t)$ $\quad \gamma$ a discount factor = present value of money

Agent navigation example.

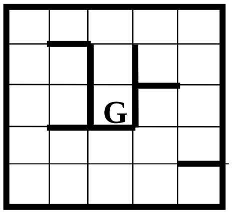
- **Agent navigation in the Maze**:
  - 4 moves in compass directions
  - Effects of moves are stochastic – we may wind up in other than intended location with non-zero probability
  - **Objective:** reach the goal state in the shortest expected time

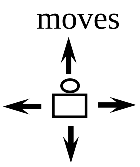




Agent navigation example

- **The RL model:**
  - **Input:** X – position of an agent
  - **Output:** A –a move
  - **Reinforcements:** R
    - -1 for each move
    - +100 for reaching the goal
  - **A policy:** $\pi : X \rightarrow A$





$\pi :$ **Position 1** → ***right***
Position 2 → ***right***
…
Position 20 → ***left***

- **Goal: find the policy maximizing future expected rewards**

$$E(\sum_{t=0}^{\infty} \gamma^t r_t)$$

Objectives of RL learning

- **Objective:**

  Find a mapping $\pi^* : X \rightarrow A$

  That maximizes some combination of future reinforcements (rewards) received over time
- **Valuation models (quantify how good the mapping is):**
  - **Finite horizon model**

    $E(\sum_{t=0}^{T} r_t)$ Time horizon: $T > 0$
  - **Infinite horizon discounted model**

    $E(\sum_{t=0}^{\infty} \gamma^t r_t)$ Discount factor: $0 < \gamma < 1$
  - **Average reward**

    $\lim_{T \rightarrow \infty} \frac{1}{T} E(\sum_{t=0}^{T} r_t)$