DSC291: Machine Learning with Few Labels

Reinforcement Learning

Zhiting Hu

Lecture 13, May 13, 2025

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

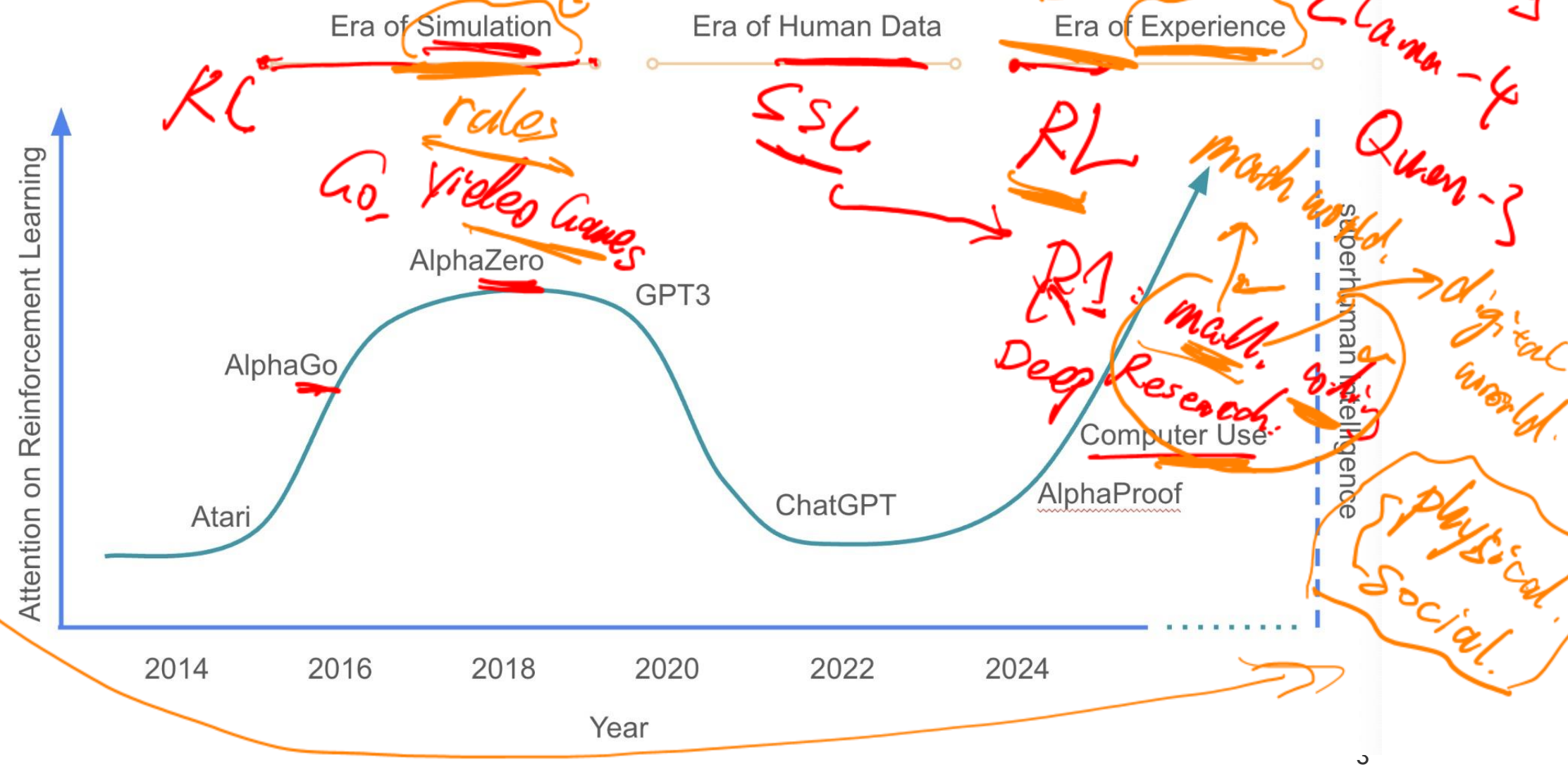
Outline

- Reinforcement Learning

- Paper presentation:
  - David Lurie, Ben TenWolde: “Mixture of Agents with LLMs”
  - Qing Cheng, Zhaomei Geng: “ChatGPT in medicine: an overview of its applications, advantages, limitations, future prospects, and ethical considerations”

Welcome to the Era of Experience

David Silver, Richard S. Sutton*



So far… Supervised Learning

Data: (x, y)
x is data, y is label

Goal: Learn a *function* to map x -> y

Examples: Classification, regression, object detection, semantic segmentation, image captioning, etc.

Cat

Classification

So far… Unsupervised Learning

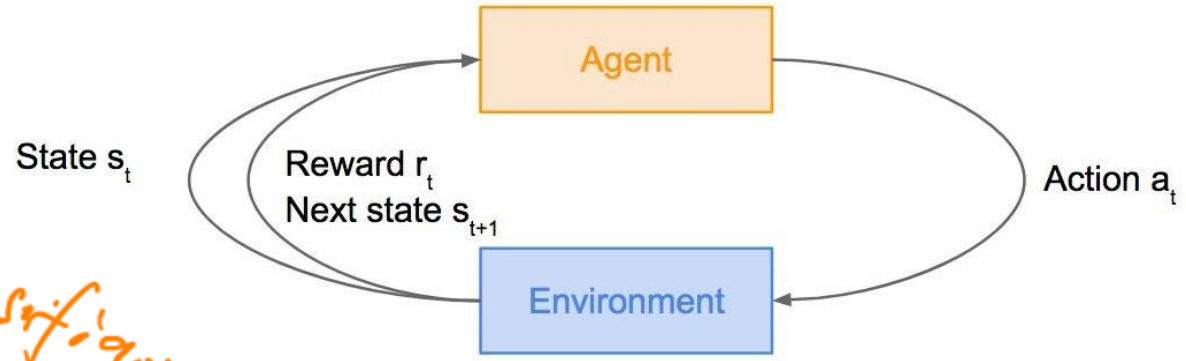
Data: x
no labels!

Goal: Learn some underlying hidden *structure* of the data

Examples: Clustering, dimensionality reduction, feature learning, density estimation, etc.

Today: Reinforcement Learning

Problems involving an **agent** interacting with an **environment**, which provides numeric **reward** signals



Goal: Learn how to take actions in order to maximize reward

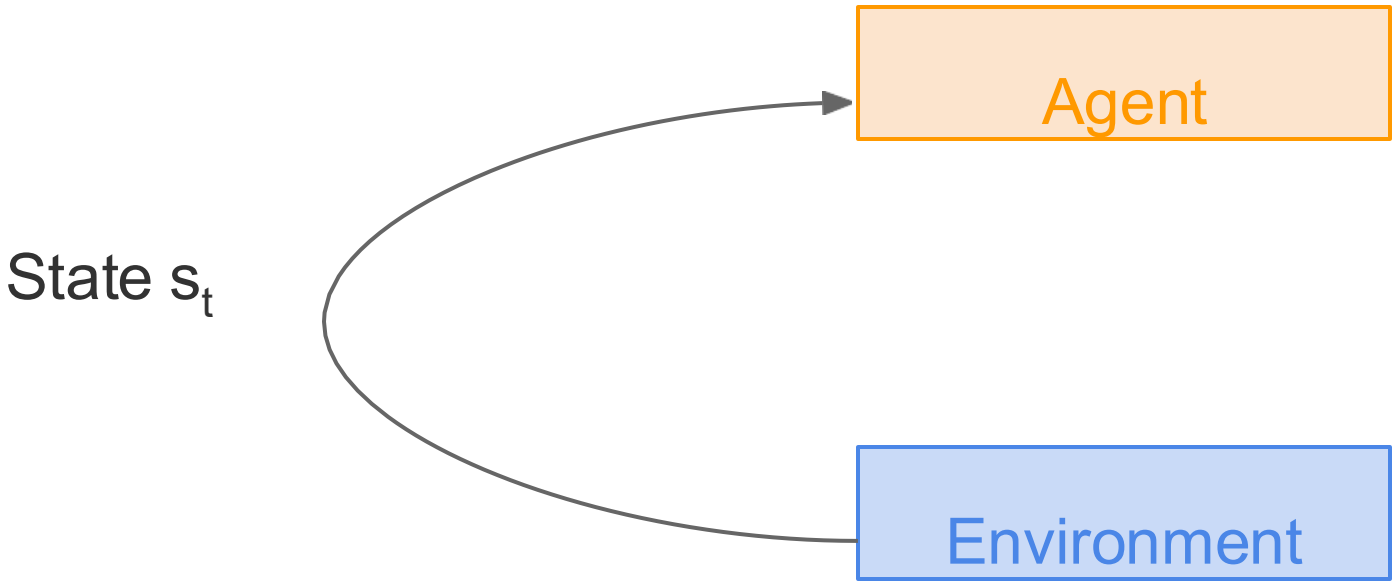
Atari games figure copyright Volodymyr Mnih et al., 2013. Reproduced with permission.

Overview

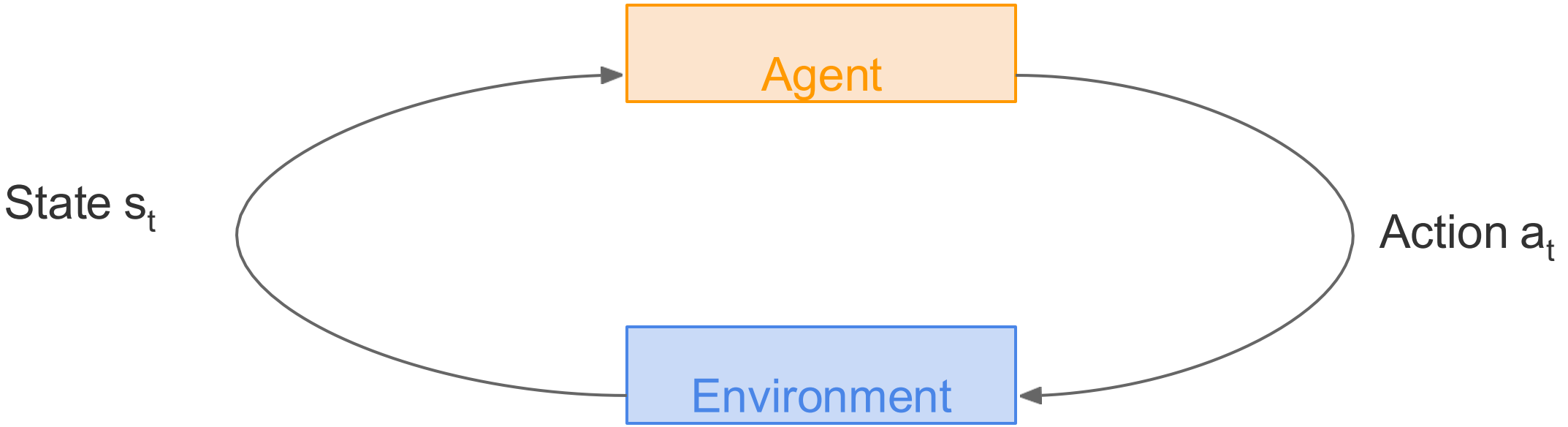
- What is Reinforcement Learning?
- Markov Decision Processes
- Q-Learning → value-based RL || off-policy RL
- Policy Gradients → policy-based RL || on-policy RL

Reinforcement Learning

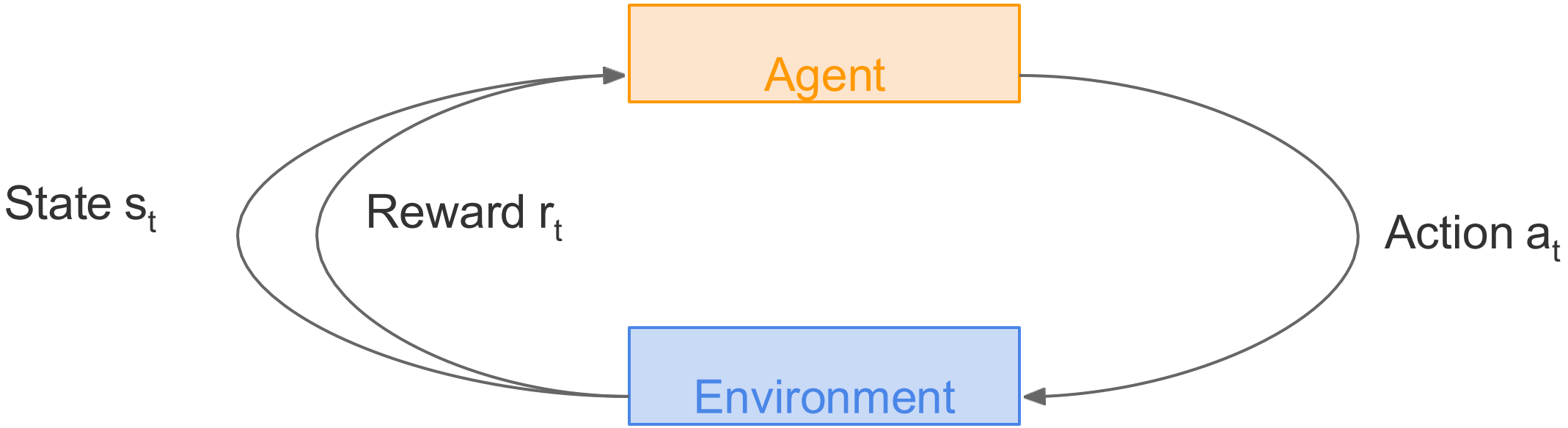
Reinforcement Learning



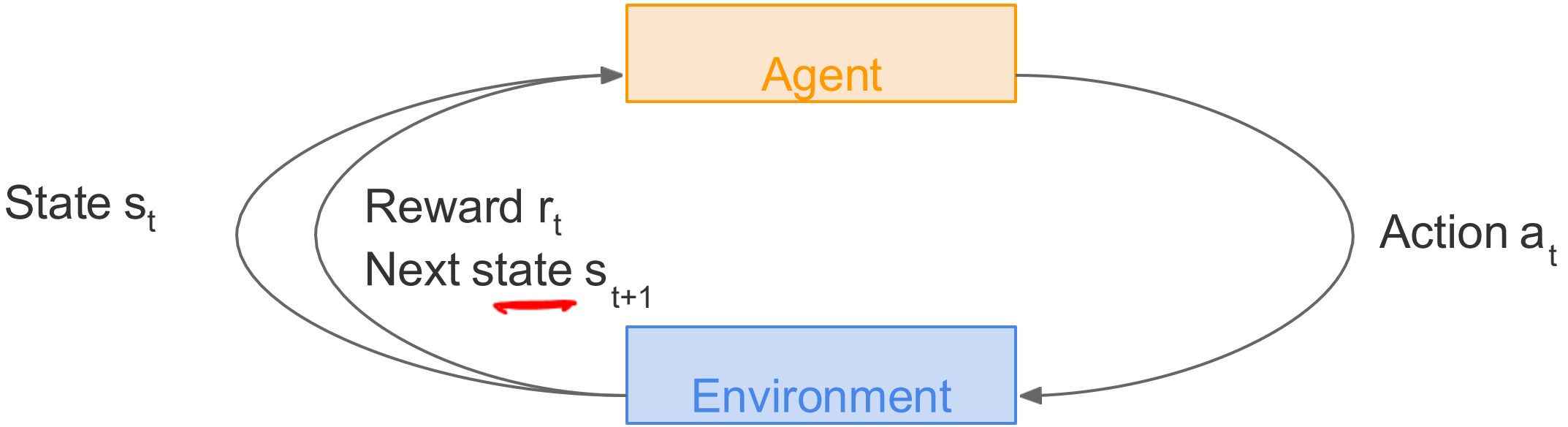
Reinforcement Learning



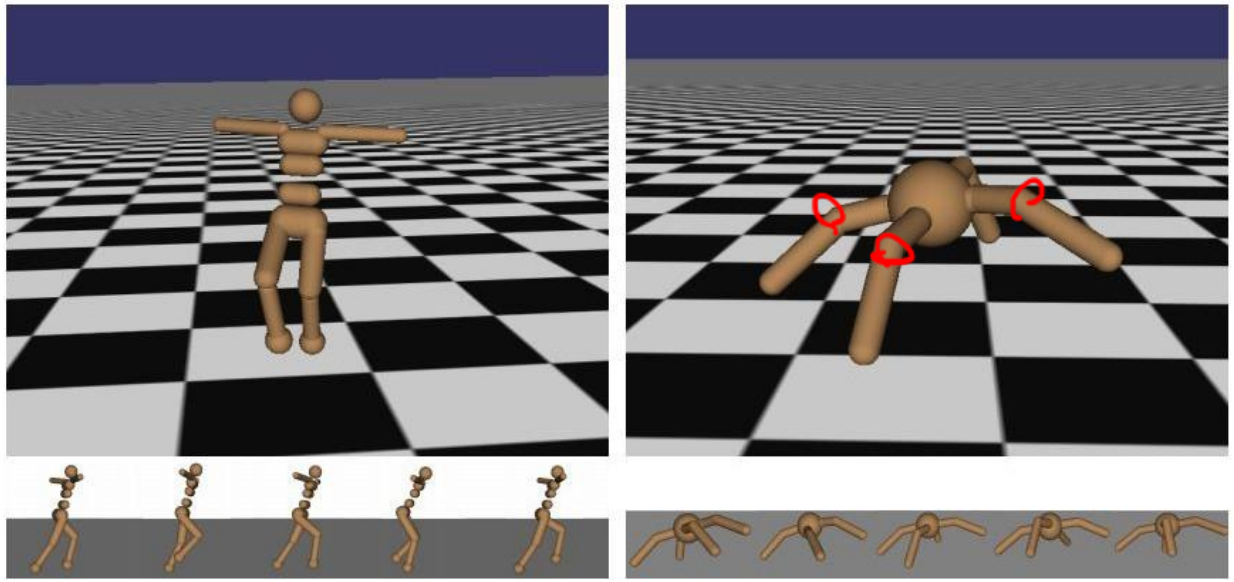
Reinforcement Learning



Reinforcement Learning



Robot Locomotion



Objective: Make the robot move forward

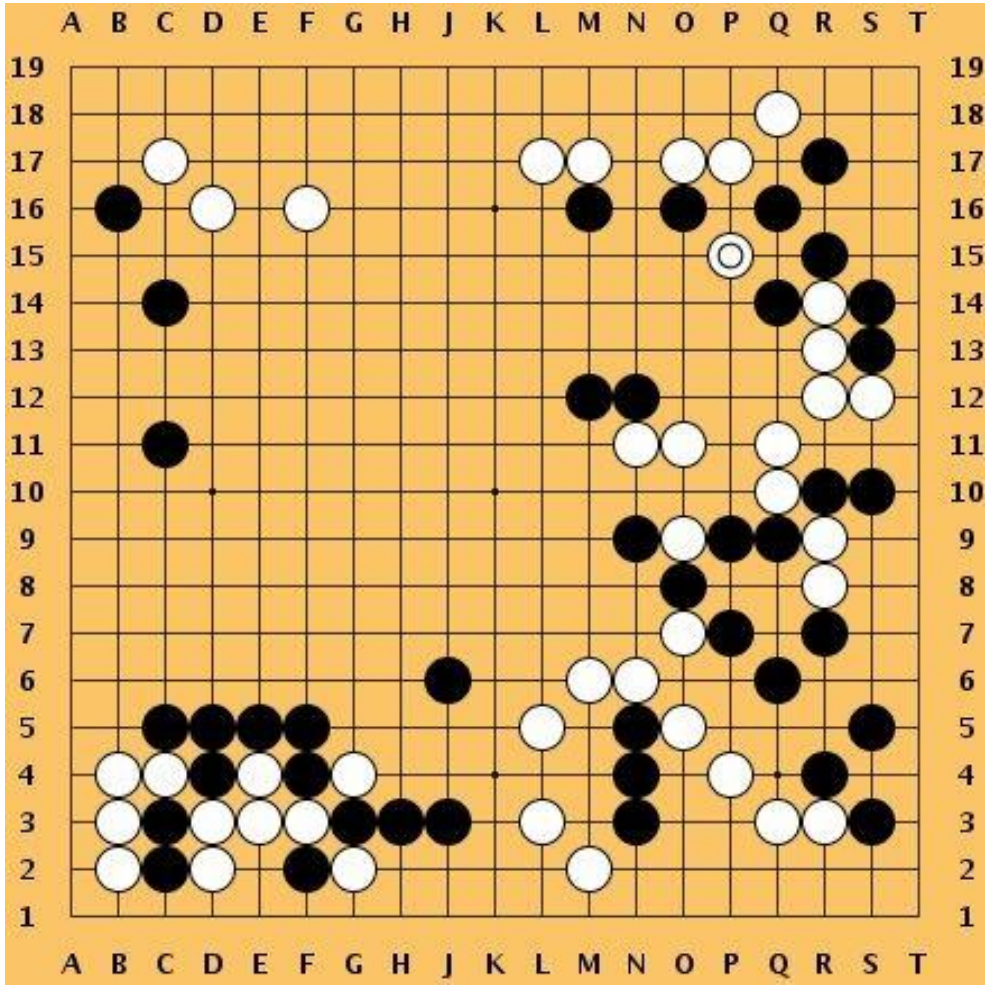
State: Angle and position of the joints
Action: Torque applied on joints
Reward: 1 at each time step upright + forward movement

Atari Games

Objective: Complete the game with the highest score

State: Raw pixel inputs of the game state
Action: Game controls e.g. Left, Right, Up, Down
Reward: Score increase/decrease at each time step

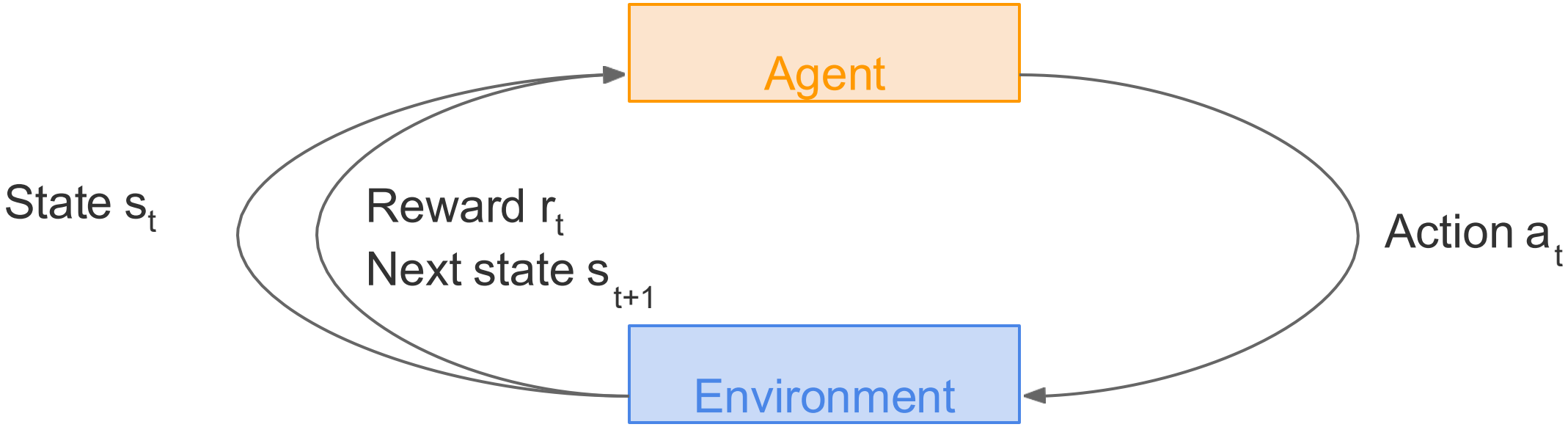
Go



Objective: Win the game!

State: Position of all pieces
Action: Where to put the next piece down
Reward: 1 if win at the end of the game, 0 otherwise

How can we mathematically formalize the RL problem?



Markov Decision Process

- Mathematical formulation of the RL problem
- **Markov property**: Current state completely characterises the state of the world

Defined by: $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathbb{P}, \gamma)$

- $\mathcal{S}$: set of possible states
- $\mathcal{A}$: set of possible actions
- $\mathcal{R}$: distribution of reward given (state, action) pair
- $\mathbb{P}$: transition probability i.e. distribution over next state given (state, action) pair
- γ : discount factor

Markov Decision Process

- At time step t=0, environment samples initial state $s_0 \sim p(s_0)$
- Then, for t=0 until done:
  - Agent selects action a_t
  - Environment samples reward $r_t \sim R(\,.\,|\,s_t, a_t)$
  - Environment samples next state $s_{t+1} \sim P(\,.\,|\,s_t, a_t)$
  - Agent receives reward r_t and next state s_{t+1}

- A policy π is a function from S to A that specifies what action to take in each state

- Following a policy produces sample *trajectories* (or paths) $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$

- **Objective**: find policy π^* that maximizes cumulative discounted reward: $\sum_{t \geq 0} \gamma^t r_t$

A simple MDP: Grid World

$MDP = (\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathbb{P}, \gamma)$

$\mathcal{A}$:

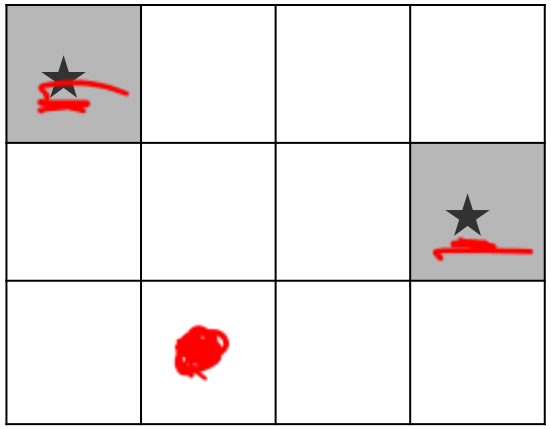
actions = {

1. right
2. left
3. up
4. down

}

$\mathcal{S}$:

states



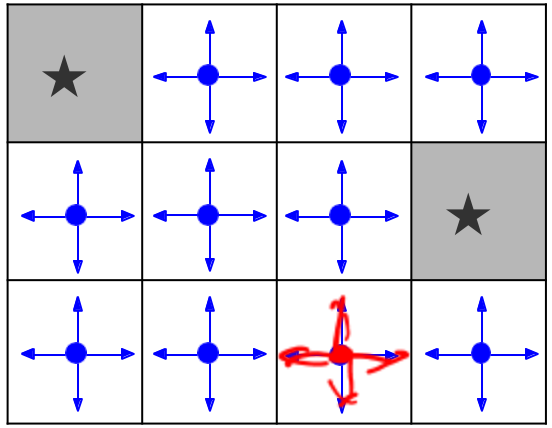
$\mathcal{R}$:

Set a negative "reward" for each transition (e.g. $r = -1$)

Objective: reach one of terminal states (greyed out) in least number of actions

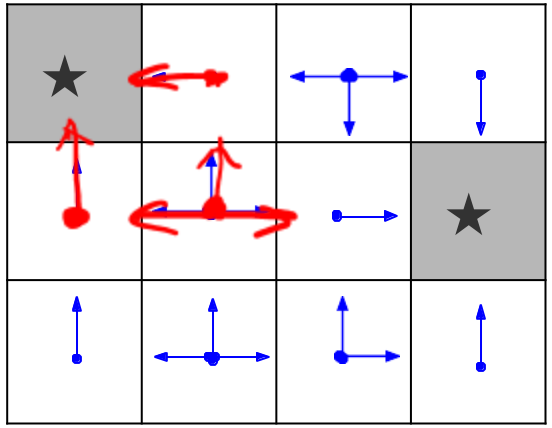
A simple MDP: Grid World

$$(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathbb{P}, \gamma)$$



Random Policy

$\pi(a|s)$



Optimal Policy

$\pi^*(a|s)$

MDP for language generation

$$(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathbb{P}, \gamma)$$

- (Autoregressive) text generation

Sentence $\boldsymbol{y} = (y_0, \ldots, y_T)$

$$\pi_\theta(y_t \mid \boldsymbol{y}_{<t}) = \text{softmax}(f_\theta(y_t \mid \boldsymbol{y}_{<t}))$$

Questions?