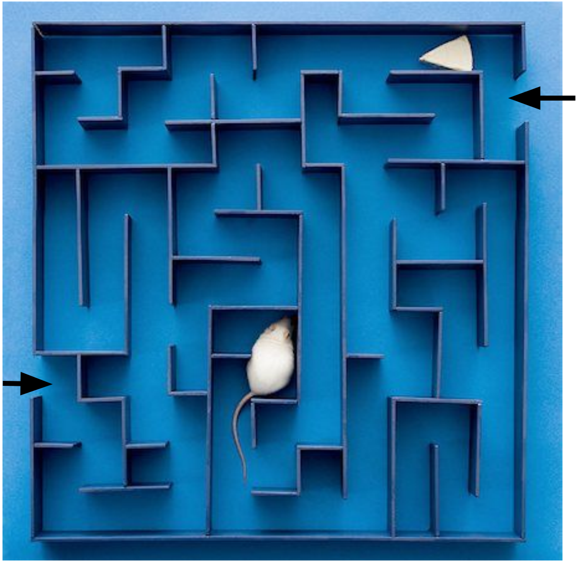
A First Glimpse at Reinforcement Learning

John D. Martin[α]

December 24, 2021

[α]University of Alberta

jmartin8@ualberta.ca



Goal

Start

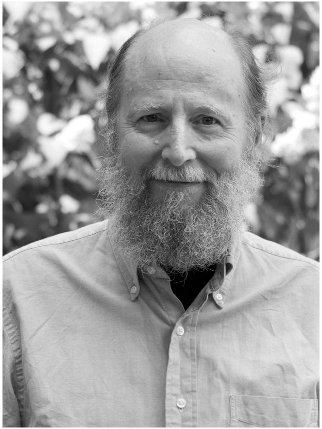
A Preview of What's to Come

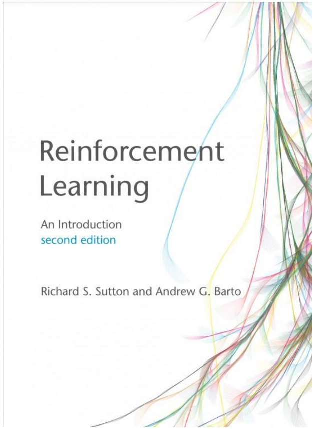
- RL day consists of four lectures
  1. A First Glimpse at Reinforcement Learning
  2. Essentials of RL
  3. RL in Modernity
  4. Applications of RL
- This lecture will cover
  - What RL is about.
  - Important concepts in RL.
  - What's goes into an RL system.
  - The RL problem.
- The course is intended to prepare you for RL research.

`https://github.com/jdmartin86/2021-naamii-rl-practical`

Practical Assignment

- Assignment will be released on Github after this lecture.
- Uses Google Colab, Python and JAX.
- Content focuses on some important RL concepts, algorithms, and phenomena.





Reinforcement Learning

An Introduction
second edition

Richard S. Sutton and Andrew G. Barto

Resources

- Course will follow selected chapters from Sutton & Barto, 2018.
- Textbook is available for free online [link].
- Other resources are linked in the last slide of this presentation.

Reinforcement Learning in Context.

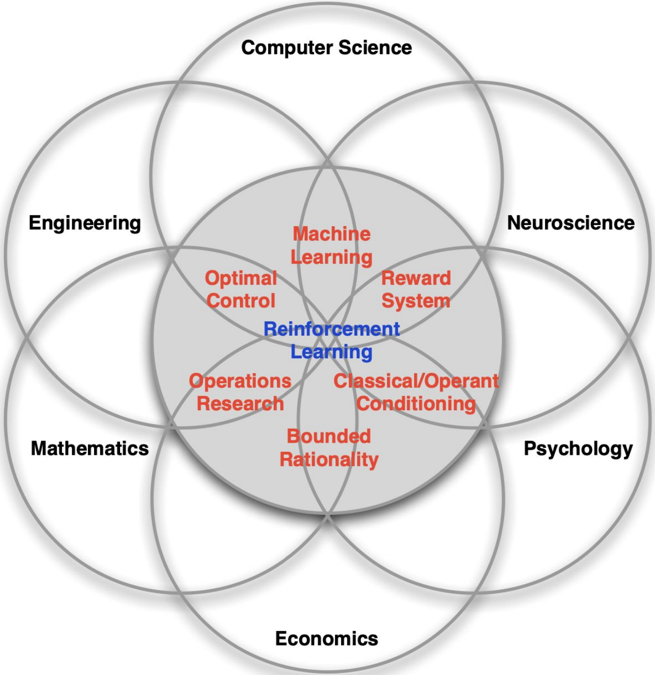


Image source: David Silver

	Planning	Supervised	Unsupervised	Reinforcement
Mode of learning	none			
Delayed consequences	yes			
Need for exploration	no			
Knowledge derives from	model			

Paradigms of Machine Learning

- Planning example: given a map navigate from Kathmandu to Janakpur.
- Supervised learning example:
- Unsupervised learning example:
- Reinforcement learning example:

	Planning	Supervised	Unsupervised	Reinforcement
Mode of learning	none	reflective		
Delayed consequences	yes	no		
Need for exploration	no	no		
Knowledge derives from	model	labeled dataset		

Paradigms of Machine Learning

- Planning example: given a map navigate from Kathmandu to Janakpur.
- Supervised learning example: predict amount of rainfall from historical data.
- Unsupervised learning example:
- Reinforcement learning example:

	Planning	Supervised	Unsupervised	Reinforcement
Mode of learning	none	reflective	reflective	
Delayed consequences	yes	no	no	
Need for exploration	no	no	no	
Knowledge derives from	model	labeled dataset	dataset	

Paradigms of Machine Learning

- Planning example: given a map navigate from Kathmandu to Janakpur.
- Supervised learning example: predict amount of rainfall from historical data.
- Unsupervised learning example: find the dominant shapes of Nepali deserts.
- Reinforcement learning example:

	Planning	Supervised	Unsupervised	Reinforcement
Mode of learning	none	reflective	reflective	experiential
Delayed consequences	yes	no	no	yes
Need for exploration	no	no	no	yes
Knowledge derives from	model	labeled dataset	dataset	datastream

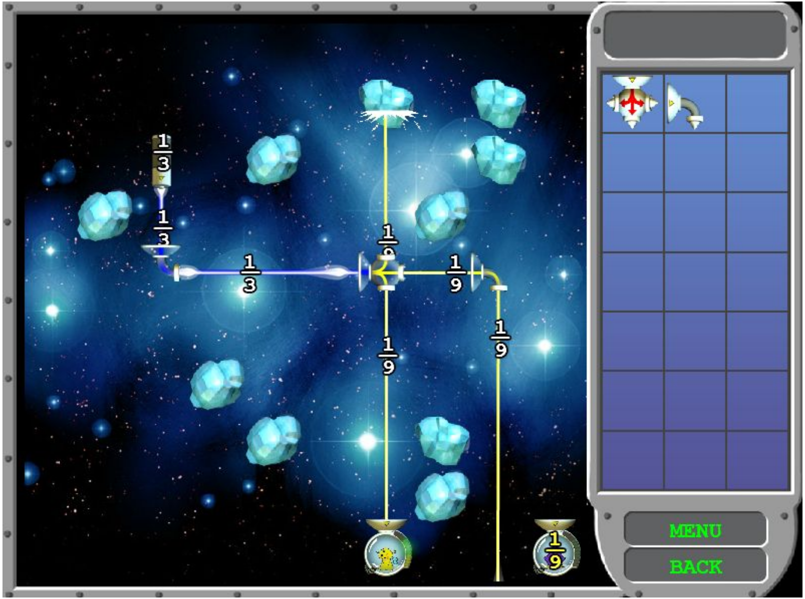
Paradigms of Machine Learning

- Planning example: given a map navigate from Kathmandu to Janakpur.
- Supervised learning example: predict amount of rainfall from historical data.
- Unsupervised learning example: find the dominant shapes of Nepali deserts.
- Reinforcement learning example: navigate to Janakpur without a map.

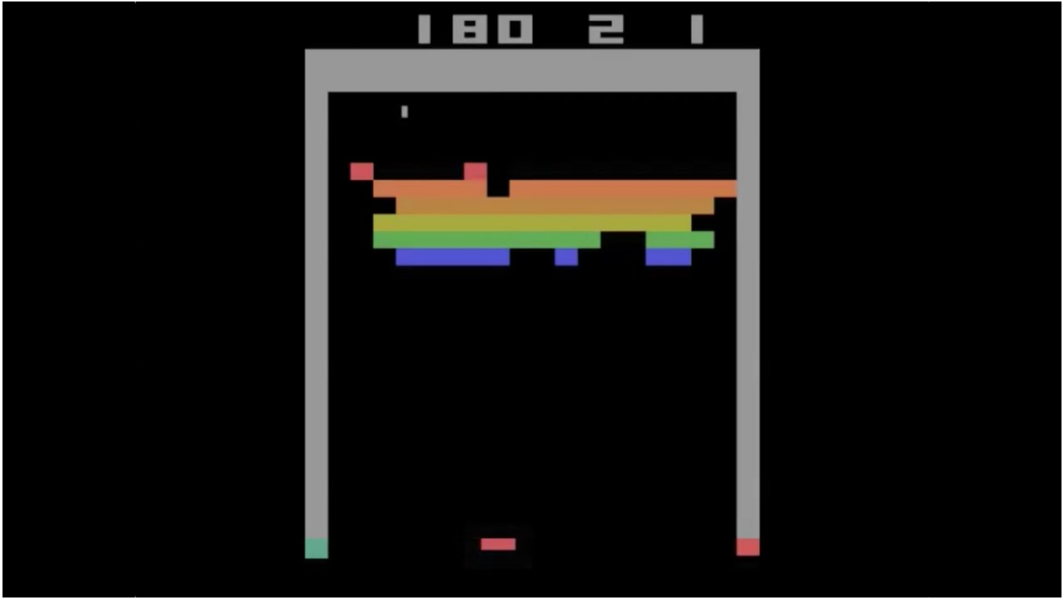
Use Cases of Reinforcement Learning

Bringing internet connectivity to the world.

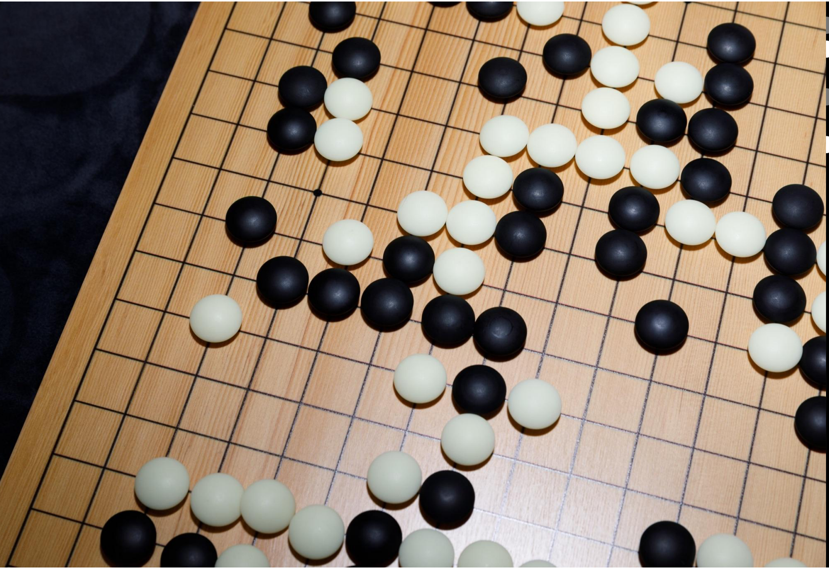
Bellemare et al., Nature (2020)

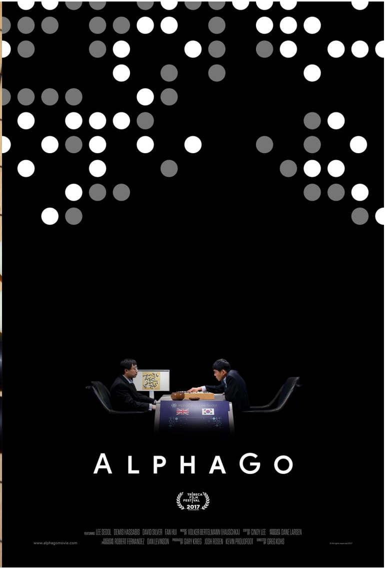


Improving the effectivity of educational games.
Madel et al., AAMAS (2014)



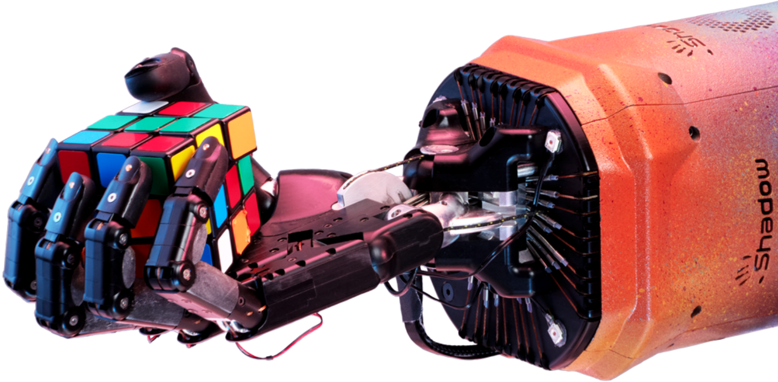
Learning to play Atari from pixels.
Mnih et al., Nature (2015)




ALPHAGO
TRIBECA FILM FESTIVAL 2017
FEATURING LEE SEDOL DEMIS HASSABIS DAVID SILVER FAN HUI MUSIC BY VOLKER BERTELMANN (HAUSCHKA) EDITED BY CINDY LEE ASSOCIATE PRODUCER DANE LARSEN
EXECUTIVE PRODUCERS ROBERT FERNANDEZ DAN LEVINSON PRODUCED BY GARY KRIEG JOSH ROSEN KEVIN PROUDFOOT DIRECTED BY GREG KOHS
www.alphagomovie.com

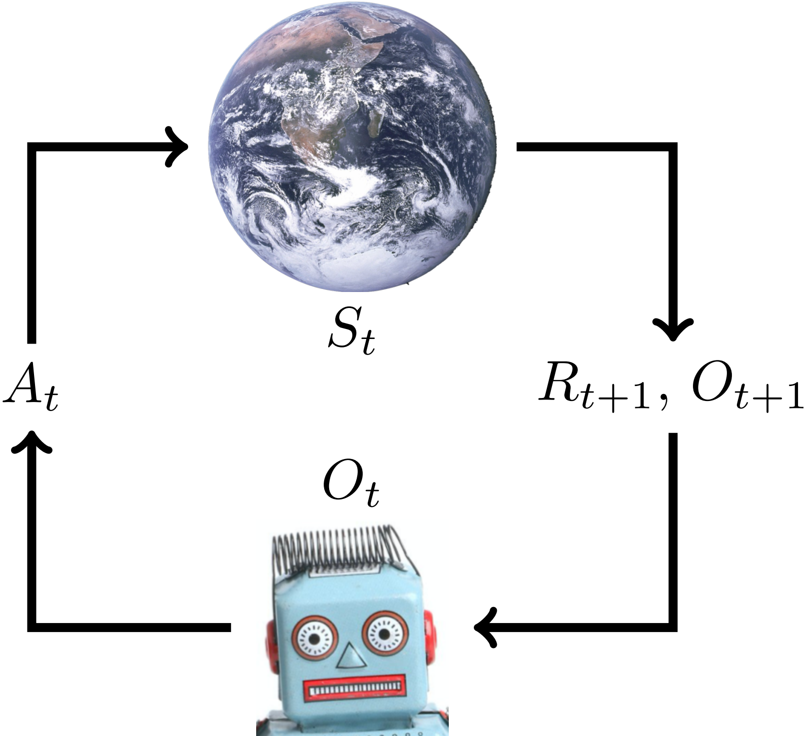
The game of Go.

DeepMind, Silver et. al, Nature (2016)



Solving a Rubik's cube.
OpenAI et al. (2019)

Concepts in Reinforcement Learning



S_t

A_t

R_{t+1}, O_{t+1}

O_t

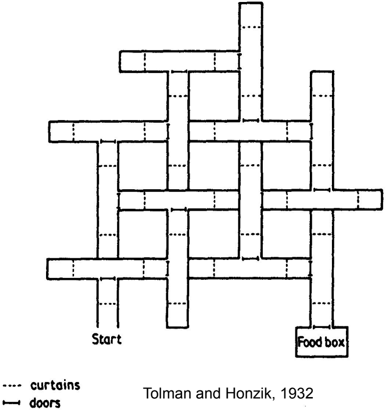
Image: Keren Su

Thinking about reward

- Reward is a scalar signal reflecting the instantaneous utility of a transition.
- Feedback reflects the degree to which a transition was useful.
- Rewards come from the environment, as part of the problem specification.

Examples of rewards

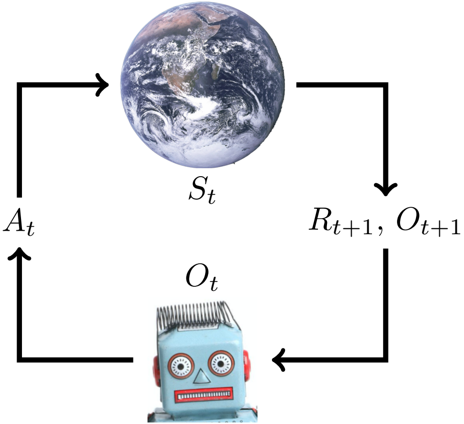
- Autonomous navigation: +1 for reaching goal, -1 for each time step.
- Play Atari games: the game score.
- Goal-directed animal behavior: release of dopamine.



Tolman and Honzik, 1932

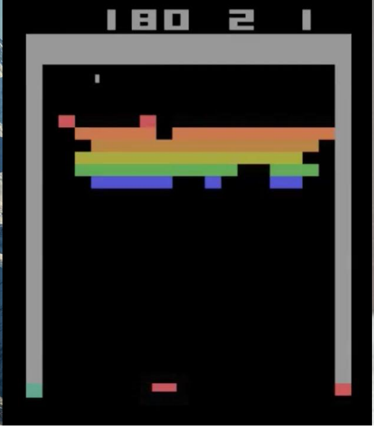
The reinforcement learning objective

- The learner wants actions that maximize the totality of its future reward.
- The *return* is a summary of future reward: $G_t \triangleq R_{t+1} + R_{t+2} + R_{t+3} + \cdots$.
- Captures temporal effect of delayed credit: short / long term consequences.



Thinking about observations

- Observations are the input that a learning system experiences.
- Observations carry information about the environment state.
- In general, measurable indicators of relevant phenomena.



Examples of observations

- Autonomous navigation: altitude sensor, inertial sensors, and GPS.
- Game bot: Pixels of a video game screen.
- Animal: Tactile information from whiskers and hands, also sight and scent.

Thinking about environment state

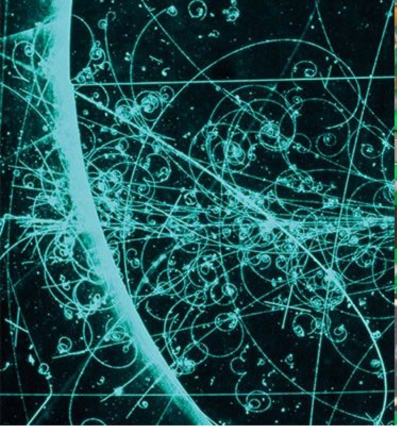
- Environment state is an external representation used to generate the observation-reward process.
- In general, the learner will not have access to the entire state.
- Potentially, this can be very complex, redundant, and not altogether relevant.

Examples of environment state

- Autonomous navigation: a full account of physical phenomena driving motion.
- Game bot: binary values of computer RAM.
- Animal: a total description of its nervous system and surrounding world.

Thinking about agent state

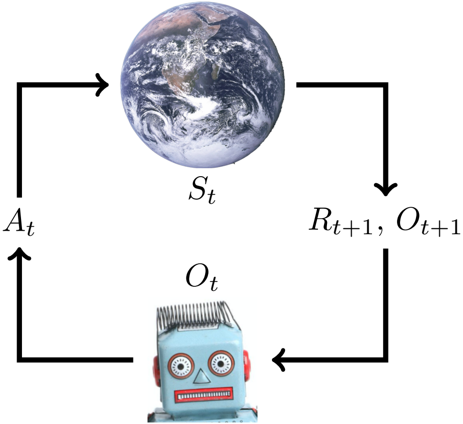
- Agent state is the learner's internal representation of environment state.
- This reflects patterns of the observation stream.
- The learner uses this information used to select actions.



Examples of agent state versus environment state

- Autonomous navigation: output of aircraft sensors vs. reality.
- Game bot: output of a convolutional neural network vs. computer RAM.
- Animal: totality of nervous system activation patterns vs. reality.

Assembling the Pieces



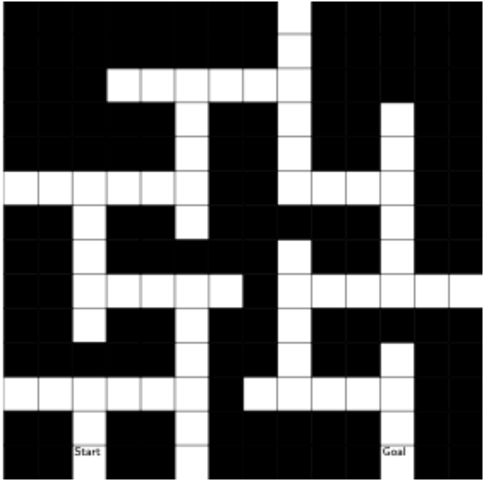
Markov Decision Processes (MDPs)

- Is a tuple with $\langle \mathcal{S}, \mathcal{A}, p, \gamma \rangle$ of observable states $\mathcal{S}$, actions $\mathcal{A}$, .
- A distribution $p(s', r | s, a)$ that describes the likelihood of possible transitions.
- An optional discount factor $\gamma \in [0, 1)$ to impose an effective decision horizon.

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 R_{t+4} + \cdots.$$

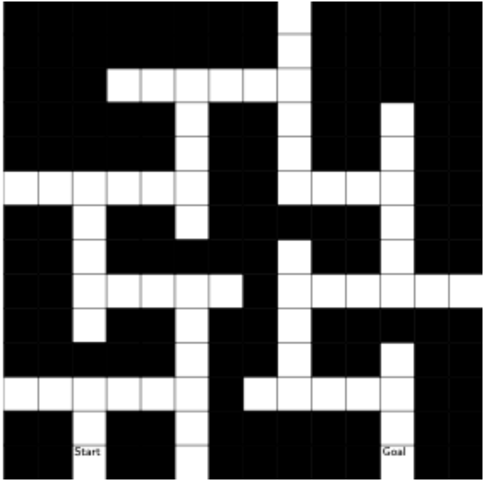
Markov Decision Processes (MDPs)

- Is a tuple with $\langle \mathcal{S}, \mathcal{A}, p, \gamma \rangle$ of observable states $\mathcal{S}$, actions $\mathcal{A}$, .
- A distribution $p(s', r|s, a)$ that describes the likelihood of possible transitions.
- An optional discount factor $\gamma \in [0, 1)$ to impose an effective decision horizon.



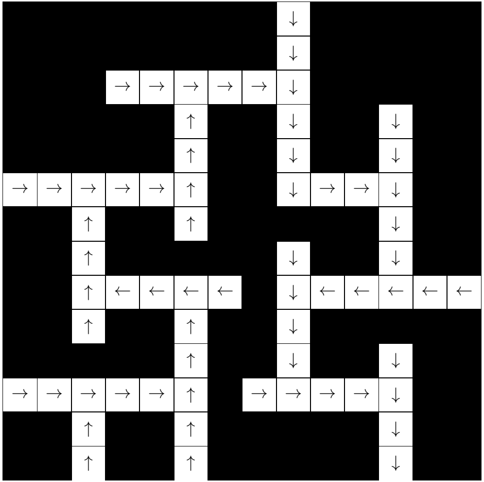
The Canonical Gridworld Domain

- Fully-observable states $\mathcal{S} = \{1, \cdots, n\}$, and actions $\mathcal{A} = \{\leftarrow, \uparrow, \downarrow, \rightarrow\}$.
- Reward is typically $+1$ at the goal and 0 everywhere else.
- Transitions can be deterministic or stochastic.



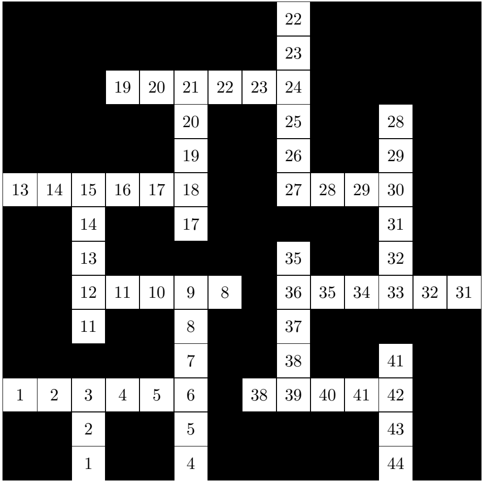
The Canonical Gridworld Domain

- Environment state is often an index: $S = i \in \mathcal{S} \subset \mathbb{N}$.
- Observations could be a one-hot vector of length $|\mathcal{S}|$, or an image of the maze.
- In the canonical setting, agent state is equivalent to environment state.



Policies

- A policy describes a way of behaving at every state.
- Mathematically, a policy is a stochastic mapping from states to actions.
- These can be distributions or deterministic functions.



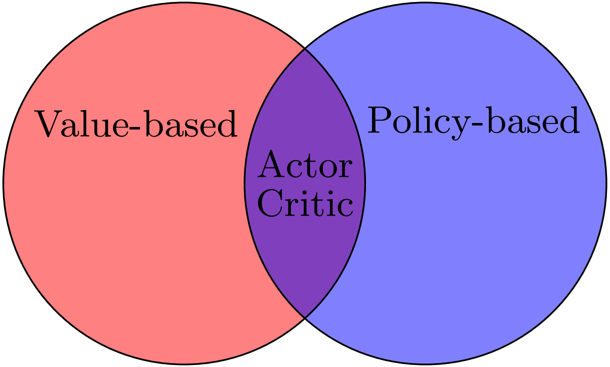
								22					
								23					
			19	20	21	22	23	24					
					20			25			28		
					19			26			29		
13	14	15	16	17	18			27	28	29	30		
		14			17						31		
		13						35			32		
		12	11	10	9	8		36	35	34	33	32	31
		11			8			37					
					7			38			41		
1	2	3	4	5	6		38	39	40	41	42		
		2			5						43		
		1			4						44		

Value functions

- The *value* is the expected return, defined from some state or state-action pair.
- State value $v_\pi(s) \triangleq \mathbf{E}[G_t | S_t = s]$, the predicted utility of being in s.
- Action value $q_\pi(s, a) \triangleq \mathbf{E}[G_t | S_t = s, A_t = a]$, the utility of taking a in s.

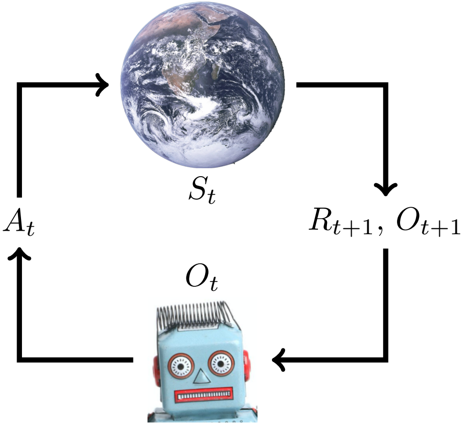
Exploration vs Exploitation

- Actions affect a learner's future experience.
- Exploring new / different actions could lead to better knowledge.
- Exploiting what is already known can be immediately useful.



Types of RL Systems

- Value-based: agent represents a value function as the learning target.
- Policy-based: agent represents a policy as the learning target.
- Actor-critic methods: represents both value and policy.



S_t

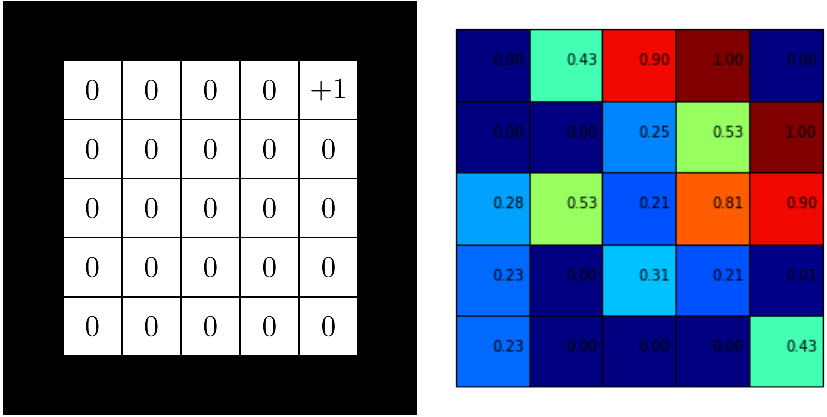
A_t

R_{t+1}, O_{t+1}

O_t

Two modes of reinforcement learning

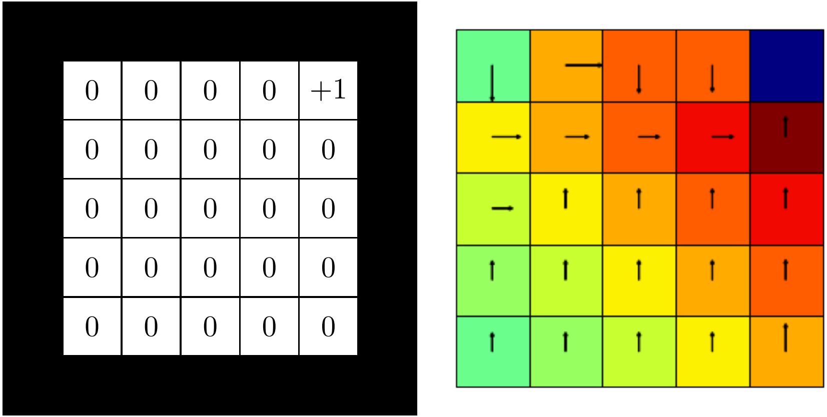
- Prediction: Estimate the return induced by a given policy.
- Control: Learn a useful policy.



0	0	0	0	+1
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

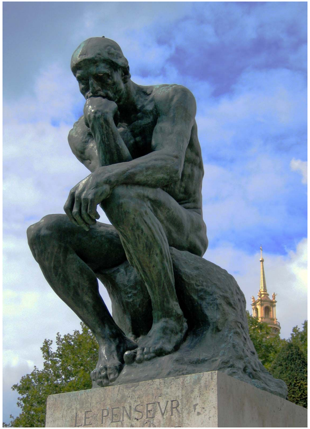
0.00	0.43	0.90	1.00	0.00
0.00	0.00	0.25	0.53	1.00
0.28	0.53	0.21	0.81	0.90
0.23	0.00	0.31	0.21	0.01
0.23	0.00	0.00	0.00	0.43

Prediction example with uniform random policy.



0	0	0	0	+1
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

Control example.



Reflecting on common assumptions

- Is a scalar reward sufficient for a machine to achieve human-level intelligence?
- Where do other agents fit into this picture?
- What about environments that persistently change?

A Preview of What's to Come

- RL day consists of four lectures
    1. A First Glimpse at Reinforcement Learning
    2. Essentials of RL
    3. RL in Modernity
    4. Applications of RL
- This lecture will cover
    - What RL is about.
    - Important concepts in RL.
    - What's goes into an RL system.
    - The RL problem.
- The course is intended to prepare you for RL research.

Additional Courses

- David Silver's RL Course [link].
- Emma Brunskill's RL Course [link].
- Adam White and Martha White RL Coursera course [link].

Additional Textbooks

- Csaba Szepesvari's book [link].
- RL Theory book [link].
- Deep RL book [link].
- Distributional RL book [link].