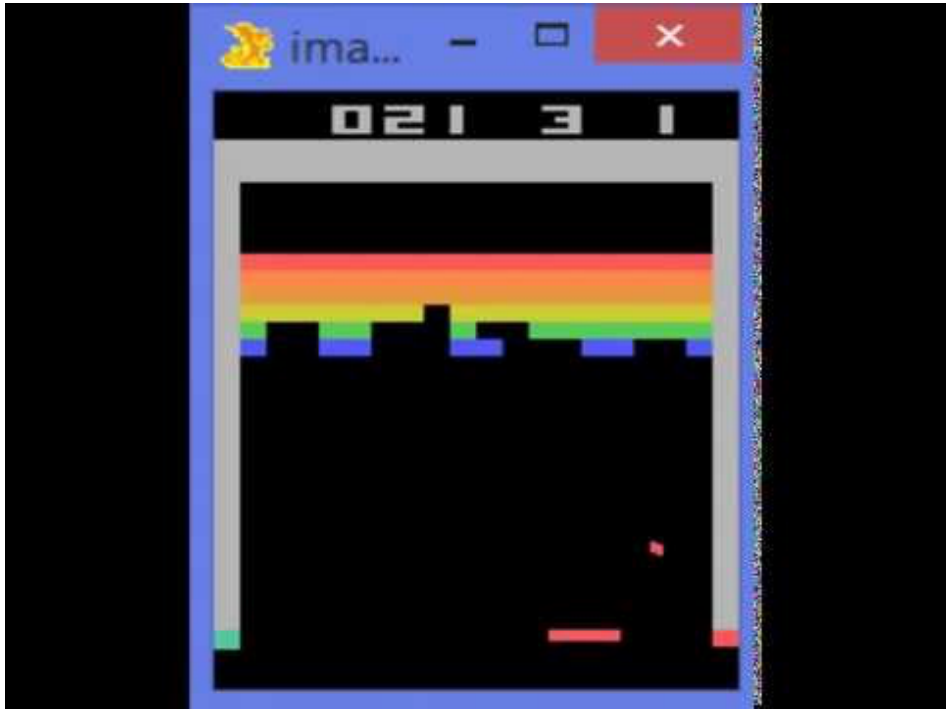
DeepMind

REINFORCEMENT LEARNING
Computational Modeling for Learning and Decision Making

Maria K. Eckstein
(mariaeckstein@deepmind.com)

BAMB summer school, 21/07/2023

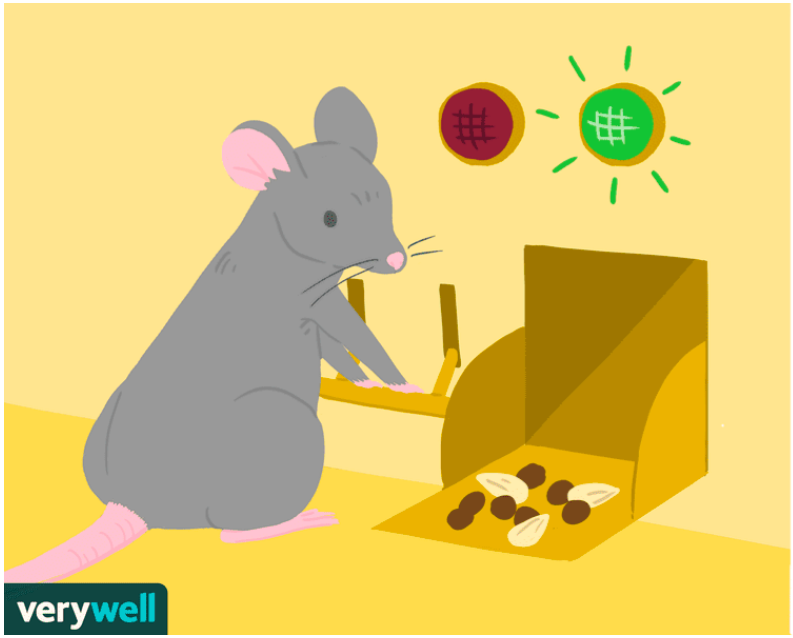
Reinforcement Learning (RL)



-> What do both videos have in common?

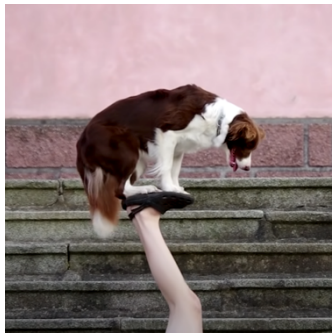
What is RL?

Learning from rewards;



and punishment.

How to Use RL (as a Cognitive Model)?

Goal	Reward	Ingredients	Algorithm
	+1	action = [→, ←] state = [] reward = [0, +1]	Q(s,a) <- Q(s,a) + α RPE RPE = r + γ Q(s',a') - Q(s,a)
		action = [jump, stand] state = [ ] reward = [0, ]	???

Questions?

DeepMind

Lecture Roadmap

Reinforcement Learning (RL)

1. **Introduction**
2. RL from a psychology perspective
3. RL from an AI perspective
4. RL from a neuroscience perspective
5. Bringing it all together: RL as a cognitive model
6. Conclusion

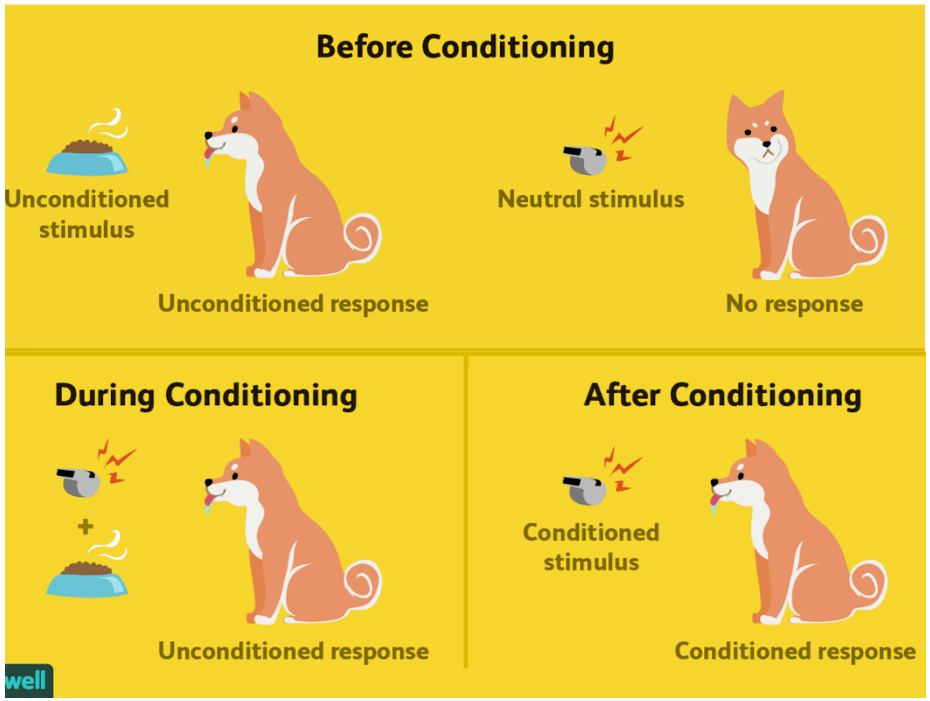
Reinforcement Learning (RL)

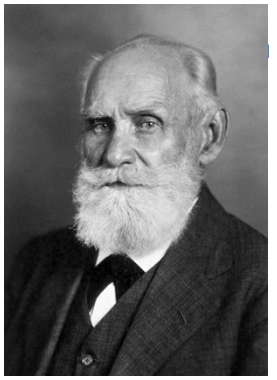
1. Introduction
2. **RL from a psychology perspective**
3. RL from an AI perspective
4. RL from a neuroscience perspective
5. Bringing it all together: RL as a cognitive model
6. Conclusion

DeepMind

RL from a psychology perspective

Classical Conditioning

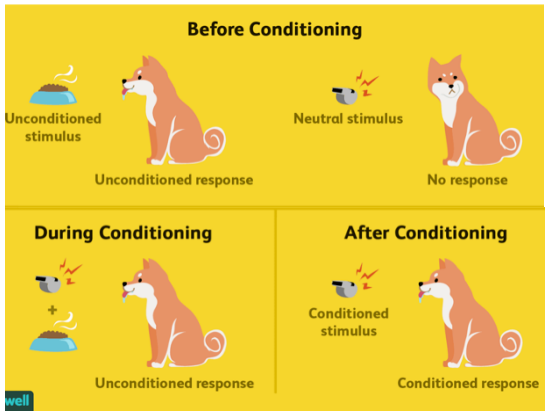




Ivan Pavlov
(1849-1936)

Animals learn associations between US (e.g., food) and neutral CS (e.g., bell) when they reliably co-occur.

The Rescorla-Wagner Model (1972)



Combined predictive value of all stimuli

$$RPE = reward - \Sigma[value(CS)]$$

$$value(CS) \leftarrow value(CS) + \alpha_{CS} * \beta_{US} * RPE$$

Old value (before learning)

New value (after learning)

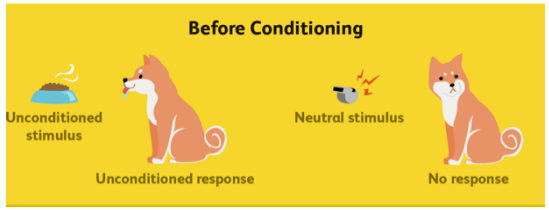
- Stimuli (CS) have “associative strength” (value)
  - Does the stimulus predict a US (reward)?
- When reward arrives, there might a “reward prediction error” (RPE)
  - Was the reward predicted by the present stimuli?
- RPEs trigger learning: update values to predict reward better
  - Learning speed depends on salience (α_{CS}) and “association value” (β_{US})

Rescorla-Wagner Example

$$RPE = reward - \Sigma[value(CS)]$$

$$value(CS) \leftarrow value(CS) + \alpha_{CS} * \beta_{US} * RPE$$

[[Assume $\alpha_{CS} * \beta_{US} = 0.5$]]

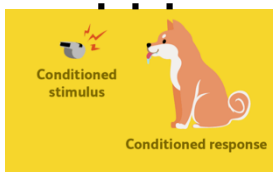



```
value(bell):          0
reward:               1
RPE:                  1
New value(bell):      0.5
```


```
value(bell):          0.5
reward:               1
RPE:                  0.5
New value(bell):      0.75
```


```
value(bell):          0.75
reward:               1
RPE:                  0.25
New value(bell):      0.865
```

???




```
value(bell):          1
```

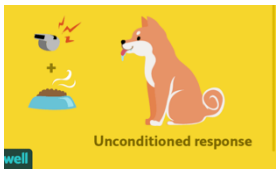
“Conditioned response”

Blocking Example

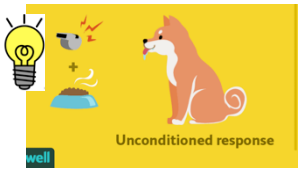
$$RPE = reward - \Sigma[value(CS)]$$

$$value(CS) \leftarrow value(CS) + \alpha_{CS} * \beta_{US} * RPE$$

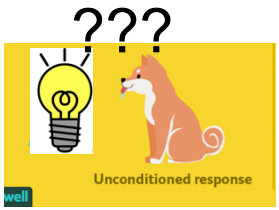
[[Assume $\alpha_{CS} * \beta_{US} = 0.5$]]




```
value(bell):          1
reward:               1
RPE:                  0
New value(bell):      1  (no change)
```




```
value(bell):          1
value(light):         0
Σ[value(CS)]:         1
reward:               1
RPE:                  0
New value(bell):      1  (no change)
New value(light):     0  (no change)
```



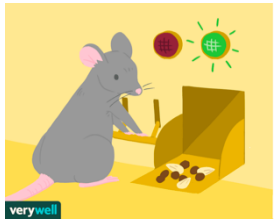

```
value(light):         0
```

No “Conditioned response”

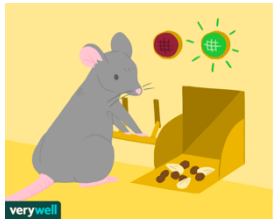
Operant conditioning

RPE = reward - value(action|state)

value(action|state) <-
value(action|state) + α * RPE

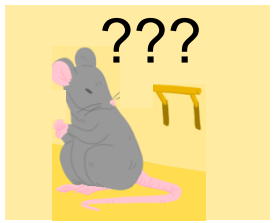



```
value(press|lev):        0
reward:                  1
RPE:                     1
New value(press|lev):    0.5
```




```
value(press|lev):        0.5
reward:                  1
RPE:                     0.5
New value(press|lev):    0.75
```

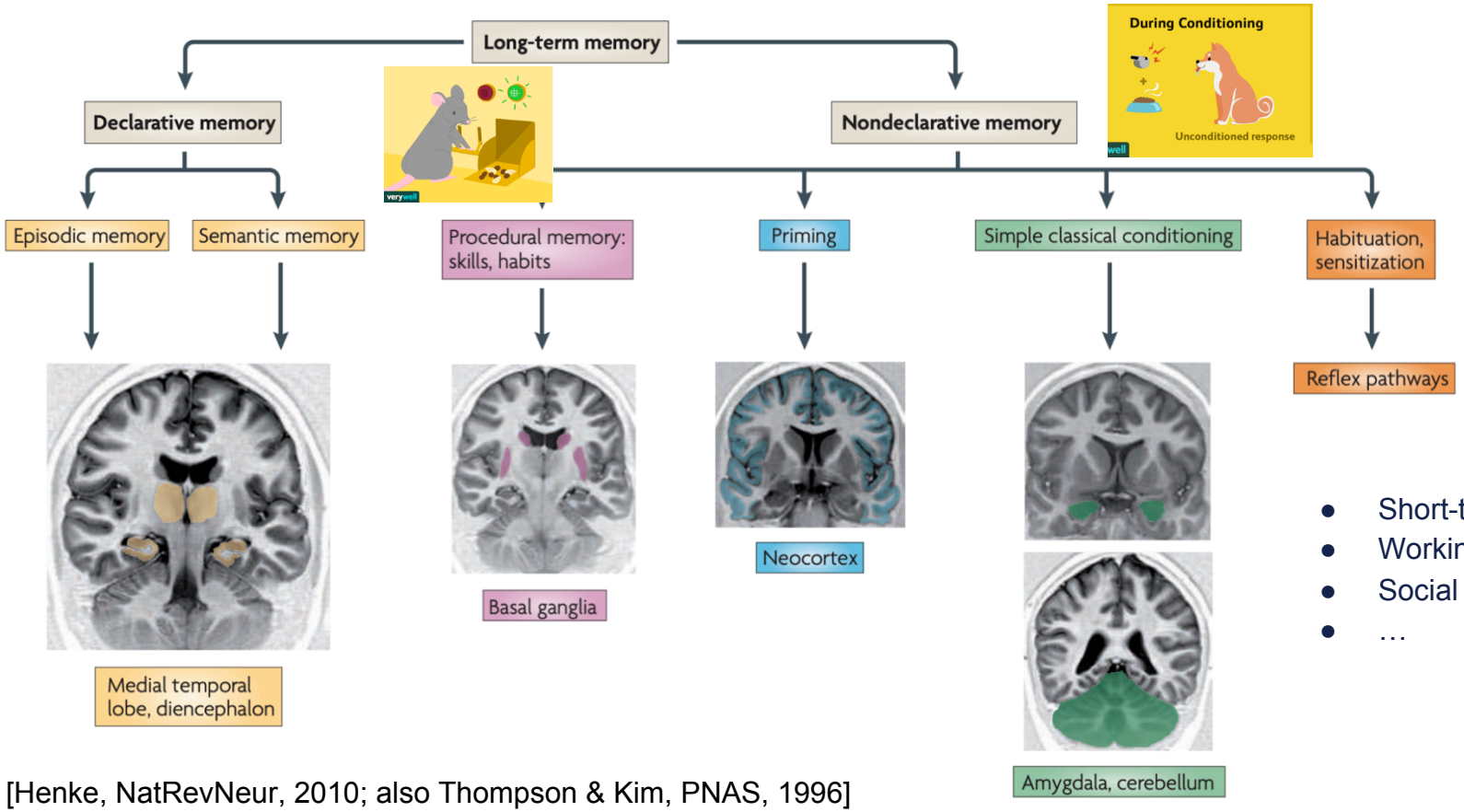
...




```
value(press|lev):        1
```

“Goal-directed response” or “Habit”?

Multiple memory systems



- Short-term memory
- Working memory
- Social memory
- …

[Henke, NatRevNeur, 2010; also Thompson & Kim, PNAS, 1996]

Questions?

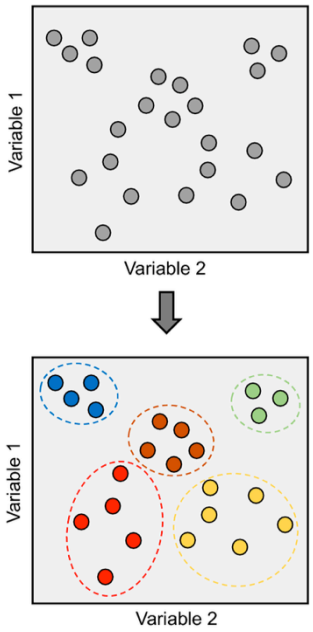
Reinforcement Learning (RL)

1. Introduction
2. RL from a psychology perspective
3. **RL from an AI perspective**
4. RL from a neuroscience perspective
5. Bringing it all together: RL as a cognitive model
6. Conclusion

DeepMind

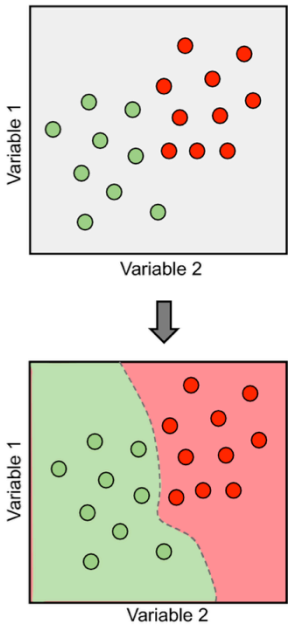
RL from an AI perspective

RL in the context of machine learning (ML)



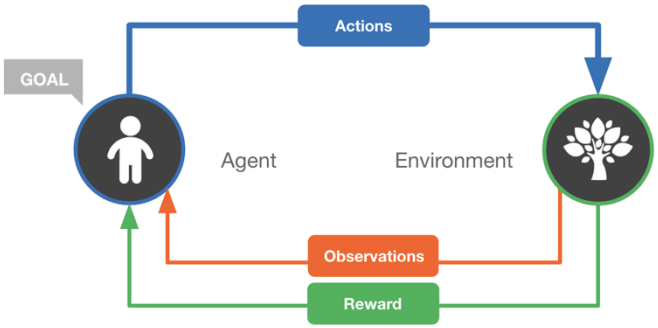
Unsupervised learning: Learn patterns or structure in data

(e.g., dimensionality reduction, clustering, …)



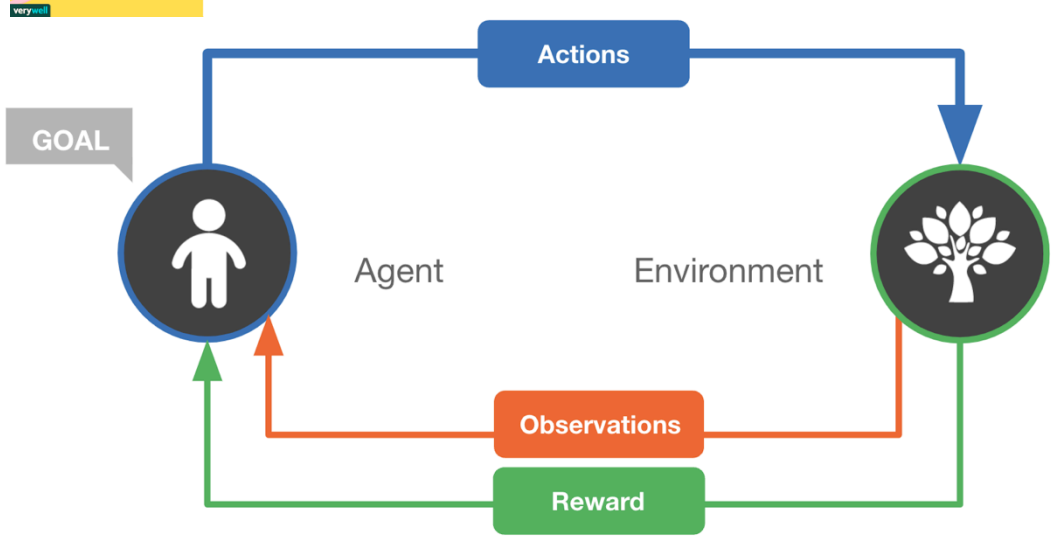
Supervised learning: Learn to predict target(s)

(e.g., regression, classification, …)



Reinforcement Learning: Learn from interactions in the world, through a scalar reward signal

RL Ingredients



Agent: Learns a policy π that maps observations to actions, in order to maximize rewards.

Environment: E.g., experimental task; game (chess, Starcraft); factory (robotics); fusion reactor; …

Reward:

- *Extrinsic* (food, water, hard-coded)
- *Intrinsic* (curiosity, novelty, empowerment, learning progress, compression, explanation, …)

The Markov Decision Process (MDP)

Markov Decision Processes allow us to *formalize* and *solve* the RL problem.

Finite set of **states**

Transition function

Discount factor

$$\langle S, \mathcal{A}, P, R, \gamma \rangle$$

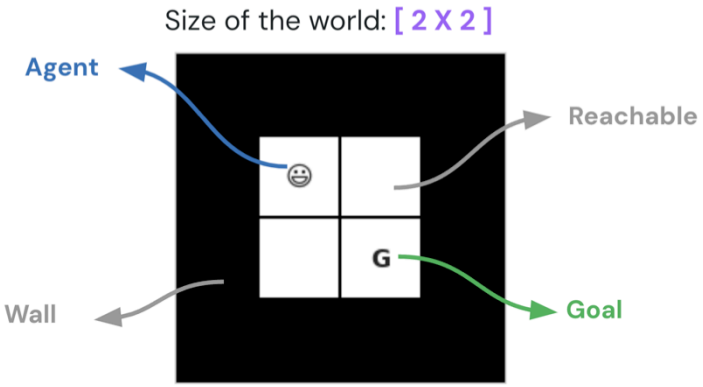
Finite set of **actions**

Reward function

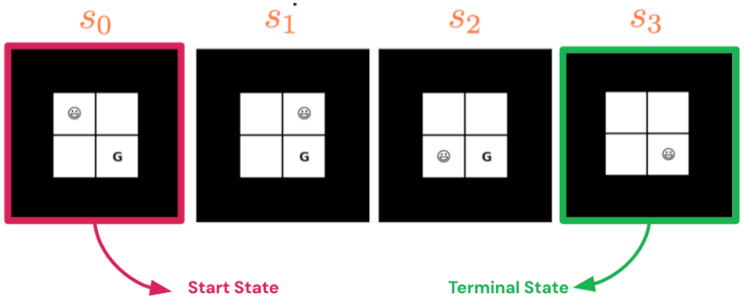
Markov Property: The next state depends only on the current state and action, not on the entire history (e.g., chess).

$$\mathbf{P}(\underbrace{s_{t+1}}_{\text{Future}} | \underbrace{s_t, a_t}_{\text{Present}}, \underbrace{s_{t-1}, a_{t-1}, \ldots, s_0}_{\text{Past}}) = \mathbf{P}(\underbrace{s_{t+1}}_{\text{Future}} | \underbrace{s_t, a_t}_{\text{Present}})$$

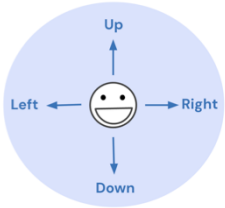
Grid Worlds



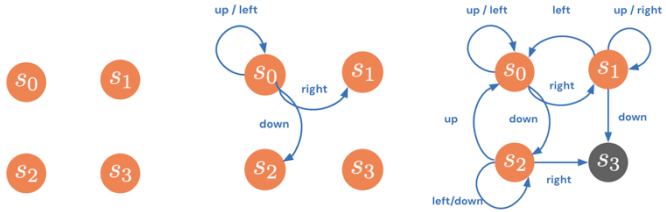
State space $\mathcal{S}$



Action Space $\mathcal{A}$



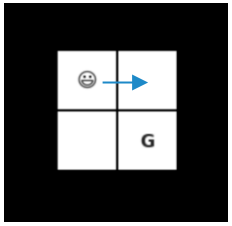
Transition model P

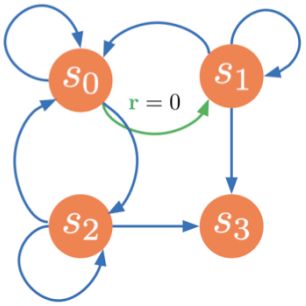


Rewards R

Empty cell: 0
Wall: -5
Goal: +10

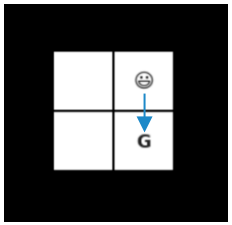
Policy and Values

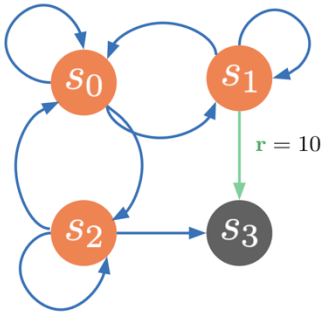




[assume γ = 0.9]

$Q^{\pi^*}(s_0, a_{right}) = 0 + 0.9 * 10 = 9$





$Q^{\pi^*}(s_1, a_{down}) = 10 + 0.9 * 0 = 10$

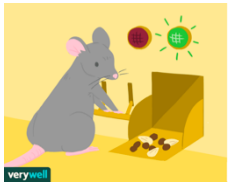
Agent's goal: Maximize (γ-discounted) sum of future rewards:

$$G_t = \underbrace{r_t + \gamma r_{t+1} + \gamma^2 r_{t+2}}_{\text{Return}} + \dots$$

To achieve this, the agent learns an action **policy π**:

$$a_t \sim \pi(a_t | s_t)$$

How do we find this policy?



Using values!

$$Q^{\pi}(s_t, a_t) = \mathop{\mathbb{E}}_{\pi}\left[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots | s_t, a_t\right]$$

Once we have values, finding the optimal policy is easy:

$$\pi^*(s) = \max_a Q^*(s, a)$$

Temporal Difference (TD) Learning

```
Value(s) += α * RPE
RPE = r - Value(s)
```

Learning rate

$$V_{t+1}(s_t) \leftarrow V_t(s_t) + \eta\delta_t$$

Reward prediction error

New estimate of value of current state

Old estimate of value of current state

$$\delta_t = R_t + \gamma V_t(s_{t+1}) - V_t(s_t)$$

Reward prediction error

Actual observed value of current state (written the recursive way)

Old estimate of value of current state

Learning the value function

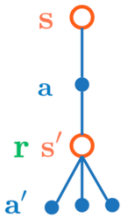
Problem: We can't predict the future! (And we don't want to…)

SARSA (on-policy control)

- Bootstrapping value updates based on "on-policy" (actual) experience

$$\underbrace{Q(s,a)}_{\text{new value}} \leftarrow \underbrace{Q(s,a)}_{\text{old value}} + \alpha(\overbrace{r + \gamma Q(s',a')}^{\text{reward + Value next state}} - \overbrace{Q(s,a)}^{\text{Old value}})$$

RPE

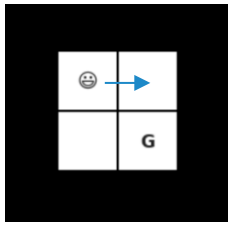


Q-learning (off-policy control)

- Bootstrapping value updates based on "off-policy" (hypothetical) experience

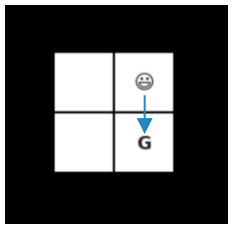
$$Q(s,a) \leftarrow Q(s,a) + \alpha(r + \gamma \overbrace{\max_{a'} Q(s',a')}^{\text{Best avail. action}} - Q(s,a))$$

$$\pi^*(s) = \max_a Q^*(s,a)$$

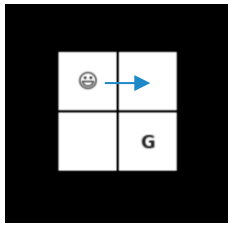


All Q's=0

Q(s0,right) <-
0+α(0+γ*0-0) = 0

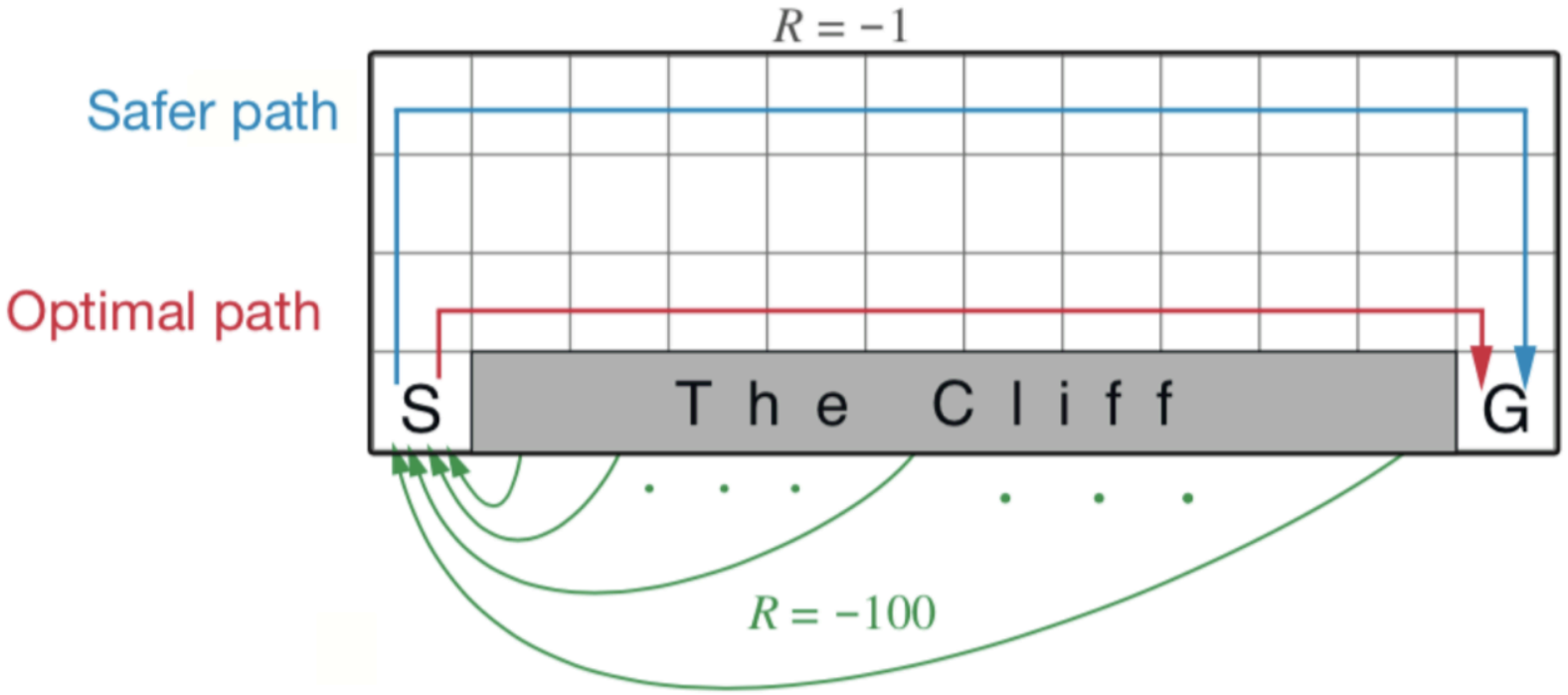


Q(s1,down) <-
0+α(10+γ*0-0) = 5



Q(s0,right) <-
0+α(0+γ*5-0) = 2.25

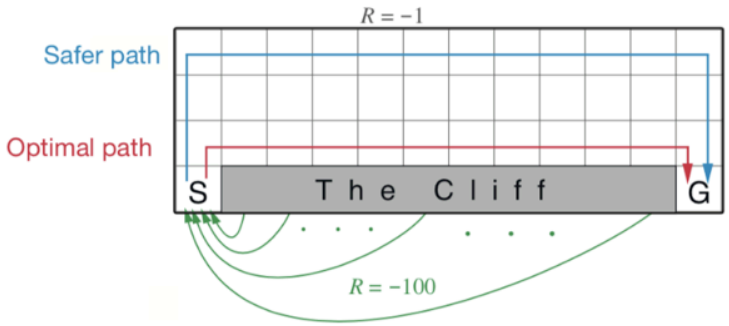
SARSA vs Q-Learning: The Cliff-walking Example

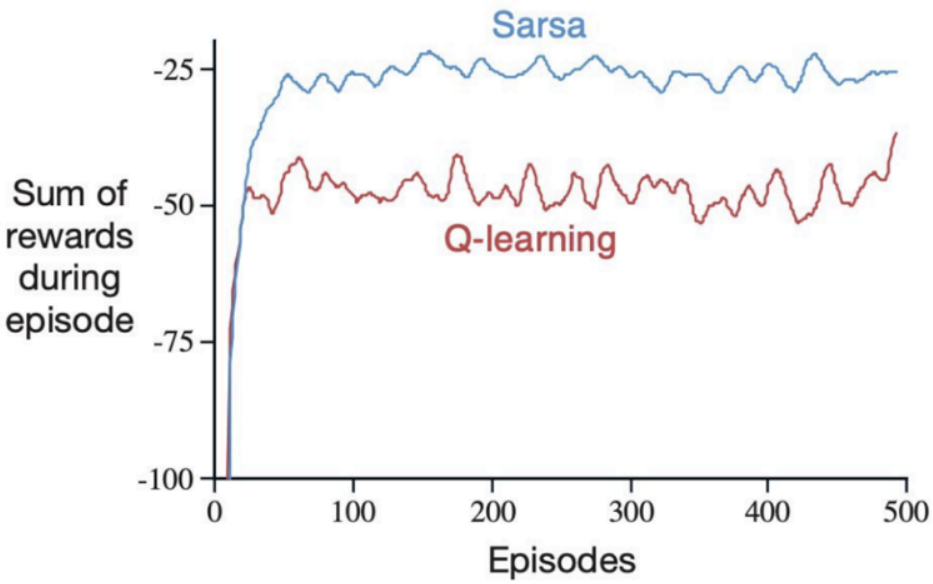


Sutton & Barto. Reinforcement Learning: An Introduction. (Chapter 6)

SARSA vs Q-Learning: The Cliff-walking Example

- **Q-learning** learns the **optimal path** while its online performance is worse than **SARSA**.
- **SARSA** learns the **safer path**.

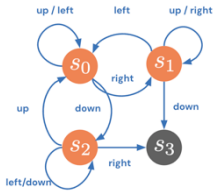




Cheat Sheet

Rescorla Wagner: keep track of reward expectations

TD Learning: +over time

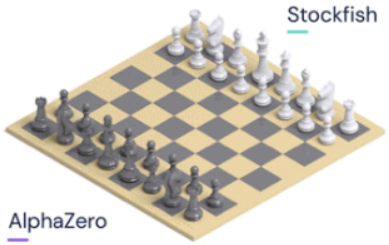


SARSA: +control (on-policy)

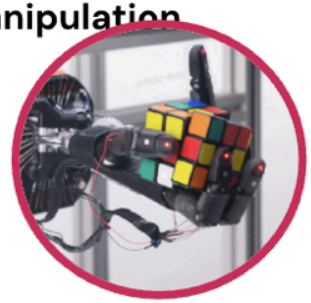
Q-Learning: +control (off-policy)

Real-world Reinforcement Learning: **Examples**

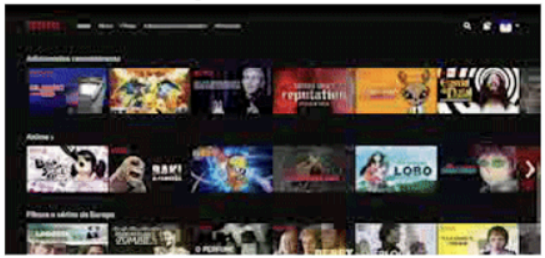
Game playing



Robotics / Manipulation



User personalization



Self-driving cars

Managing energy usage

Questions?

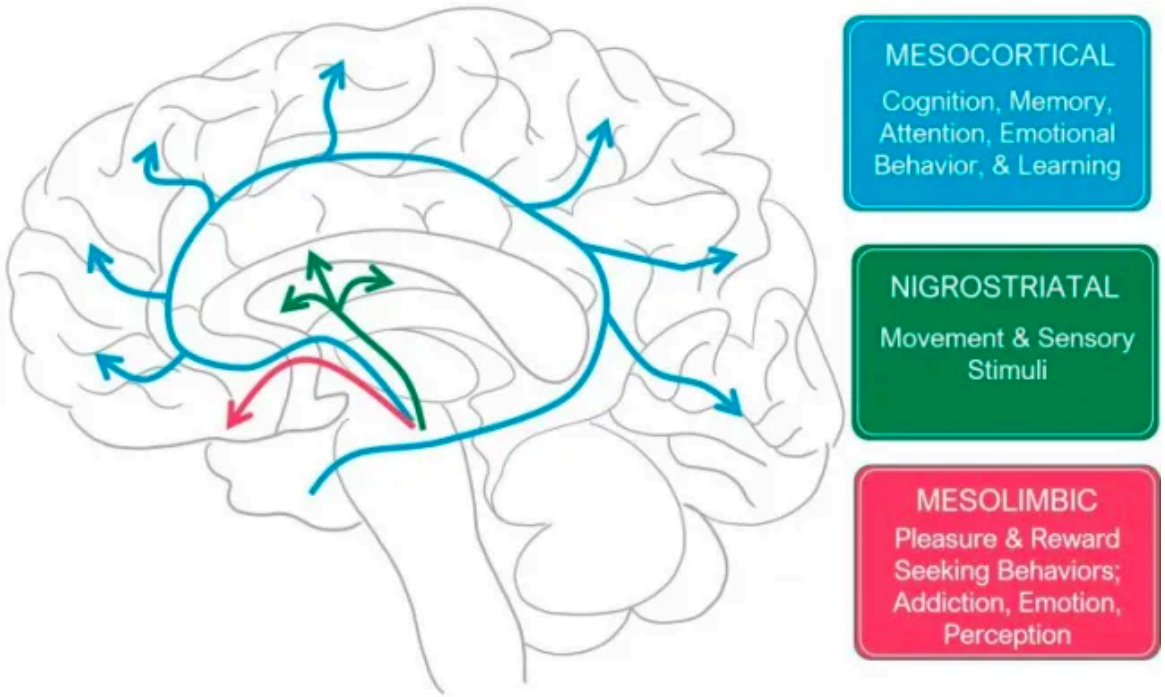
Reinforcement Learning (RL)

1. Introduction
2. RL from a psychology perspective
3. RL from an AI perspective
4. **RL from a neuroscience perspective**
5. Bringing it all together: RL as a cognitive model
6. Conclusion

DeepMind

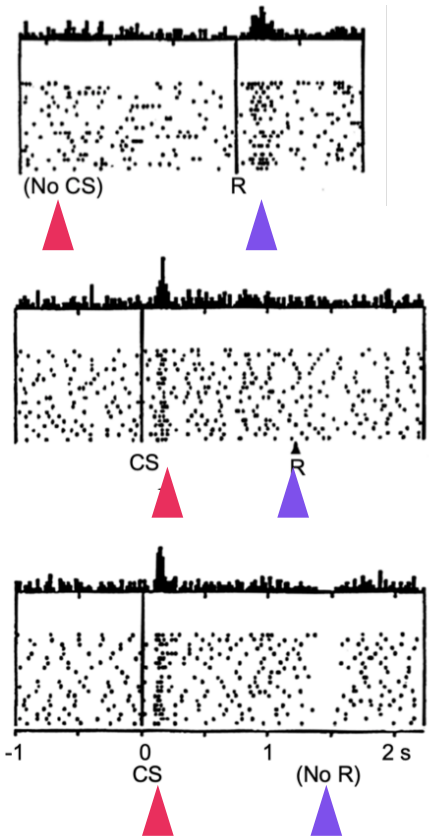
RL in neuroscience

The Neurotransmitter Dopamine



Dopamine Reward Prediction Errors

$$RPE = r + \gamma\, Q(s',a') - Q(s,a)$$



$RPE = 0 + \gamma\, 0 - 0 = 0$

$RPE = 1 + \gamma\, 0 - 0 = 1$

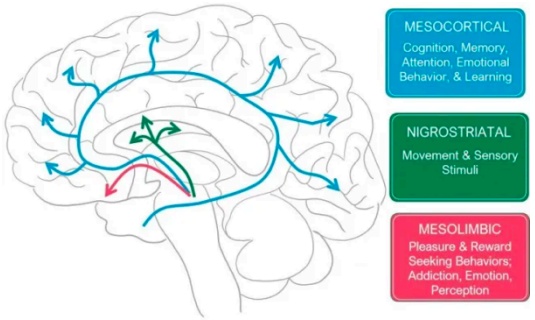
$RPE = 0 + \gamma\, 1 - 0 = 1$

$RPE = 1 + \gamma\, 0 - 1 = 0$

$RPE = 0 + \gamma\, 1 - 0 = 1$

$RPE = 0 + \gamma\, 0 - 1 = -1$

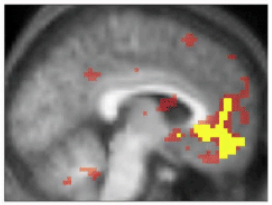
[Montague et al., 1996; Schultz et al., 1997]



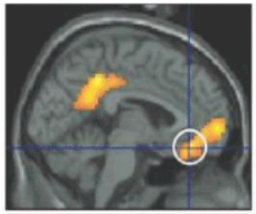
- Converging evidence across studies and species
- Mostly in simple conditioning paradigms

[Niv, 2009]

Human fMRI

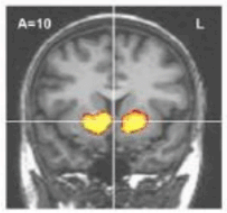


money
value predicted
(Daw et al 2006)

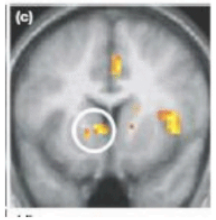


faces
attractiveness
(O' Doherty et al 2003)

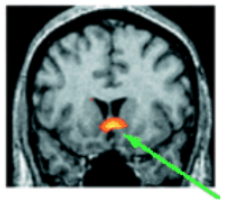
Coke or Pepsi
degree favored
(McClure et al. 2004)



money
gain vs loss
(Kuhnen & Knutson
2005)



food odors
valued vs devalued
(Gottfreid et al 2003)



juice
unpredictable vs
predictable
(Berns et al 2001)

Rewards / reward anticipation activate:

- Ventromedial prefrontal cortex
- Orbitofrontal cortex
- Striatum

➢ *Generalized appetitive function?*

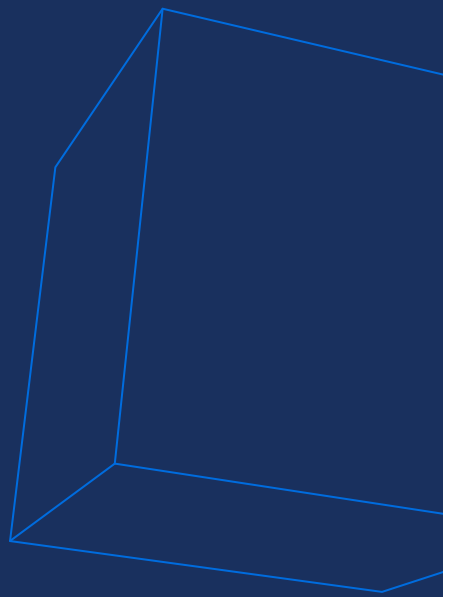
Questions?

Reinforcement Learning (RL)

1. Introduction
2. RL from a psychology perspective
3. RL from an AI perspective
4. RL from a neuroscience perspective
5. **Bringing it all together: RL as a cognitive model**
6. Conclusion

DeepMind

RL for Cognitive Modeling



What is Cognitive Modeling?

Goal: Understand behavior, cognitive process

Method:

- Find model (e.g., RL, Regression, DDM, …)
- “Fit” model (find best parameters)
- Expand model
  - e.g., reward vs punishment [Frank et al., 2004]; WM [Collins & Frank, 2012]; counterfactuals [Boorman et al., 2011]; …
- Model comparison (AIC, BIC, WAIC, …)

Result:

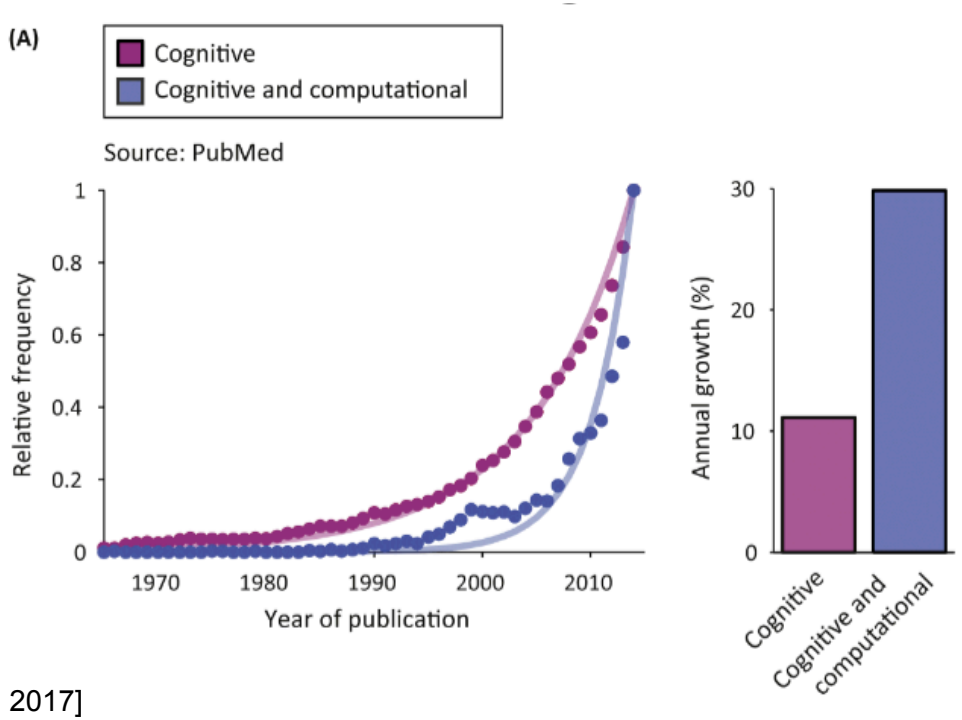
- “Cognitive process”
- Fitted parameters (individual differences)
- Normative understanding (optimality)
- Quantitative methods, statistics
- Complex, multi-step processes
- Precise prediction

RL

$$RPE = r + \gamma\, Q(s',a') - Q(s,a)$$

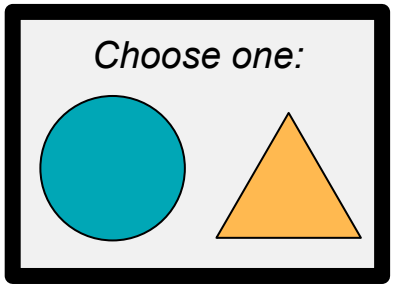
$$Q(s,a) \leftarrow Q(s,a) + \alpha * RPE$$

Computational modeling is on the rise!



[Palminter et al., 2017]

What is RL Modeling?

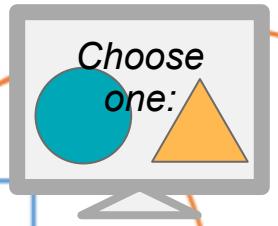
Goal	Reward	Ingredients	Algorithm
	+1	action = [→, ←] state =  reward = [0, +1]	RPE = r + γ Q(s',a') - Q(s,a) Q(s,a) <- Q(s,a) + α * RPE
	+1	action = [F H] state = [] reward = [0, +1]	RPE = r + γ Q(s',a') - Q(s,a) Q(s,a) <- Q(s,a) + α * RPE

A Recipe for Cognitive Modeling

Scientific question

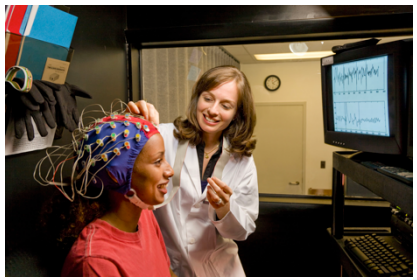
Design an experiment

Build competing models



Model predictions: Simulations

Model fitting Validate the procedure



Collect Data, Fit the models

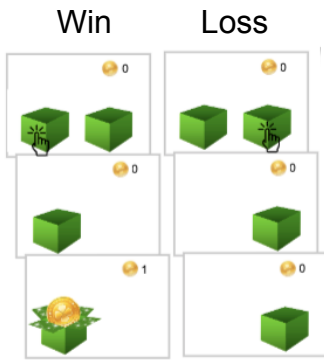
Validate the models

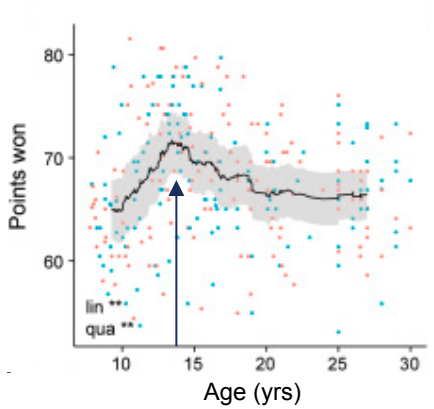
RL

[Wilson & Collins, 2019. *eLife*]

Learning to Reversal Learn

Goal: Understand age trajectory of reversal learning





- Best performance at ~13-15

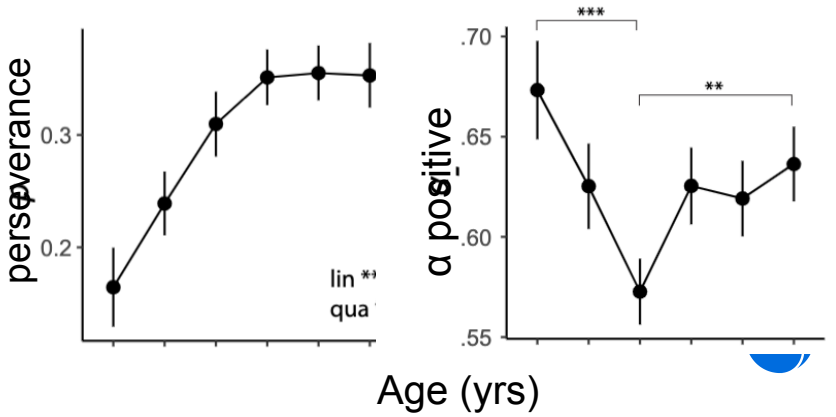
Why? Cognitive mechanism?

RL

[Eckstein, Master, Dahl, Wilbrecht & Collins, 2022. *DCN*]

RPE = r - Q(s,a)

Q(s,a) <- Q(s,a) + α * RPE



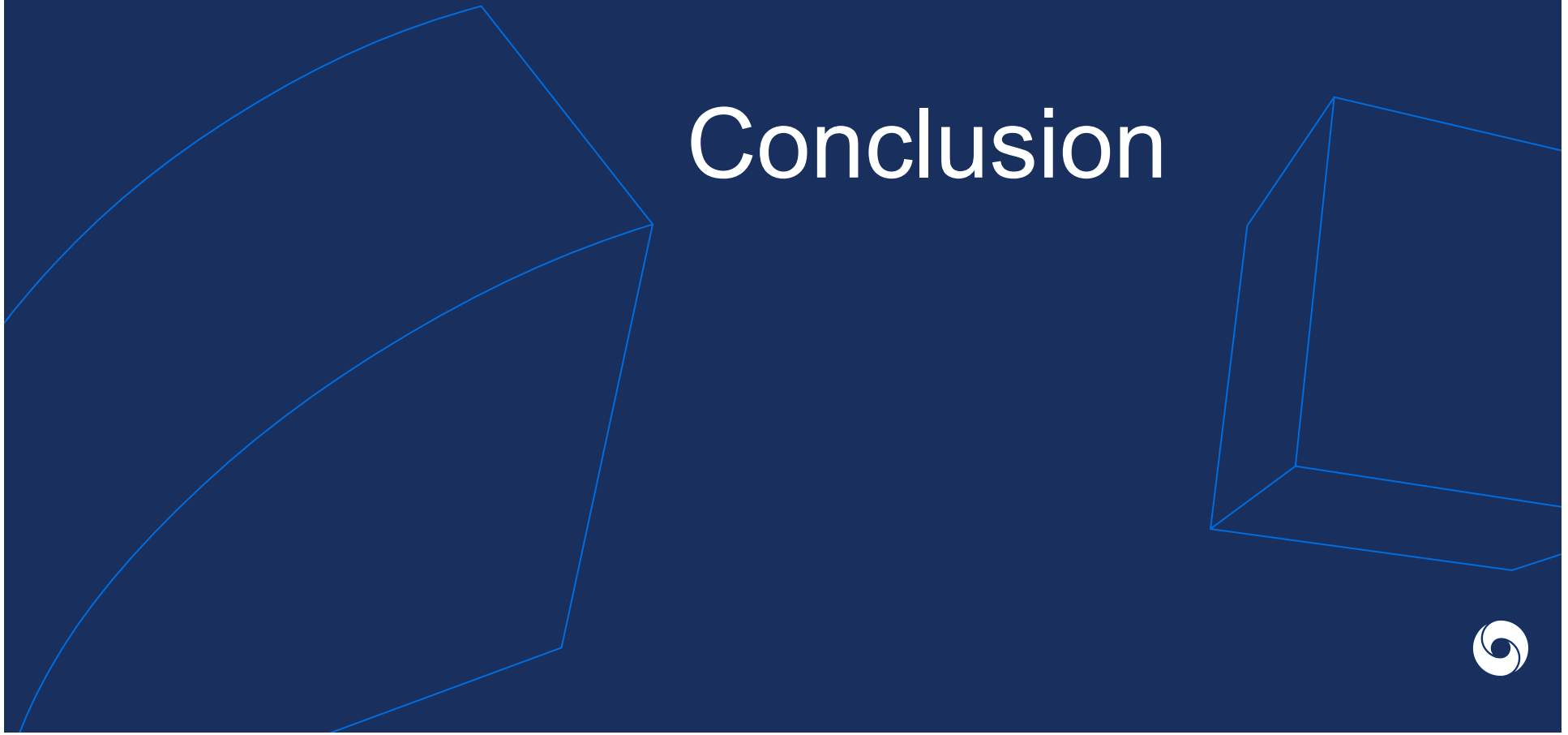
Questions?

Reinforcement Learning (RL)

1. Introduction
2. RL from a psychology perspective
3. RL from an AI perspective
4. RL from a neuroscience perspective
5. Bringing it all together: RL as a cognitive model
6. **Conclusion**

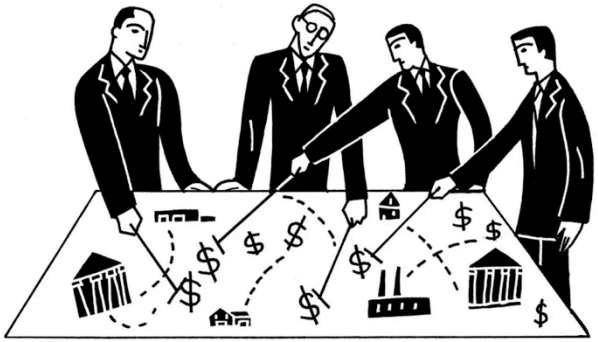
DeepMind

Conclusion



Where do rewards come from?

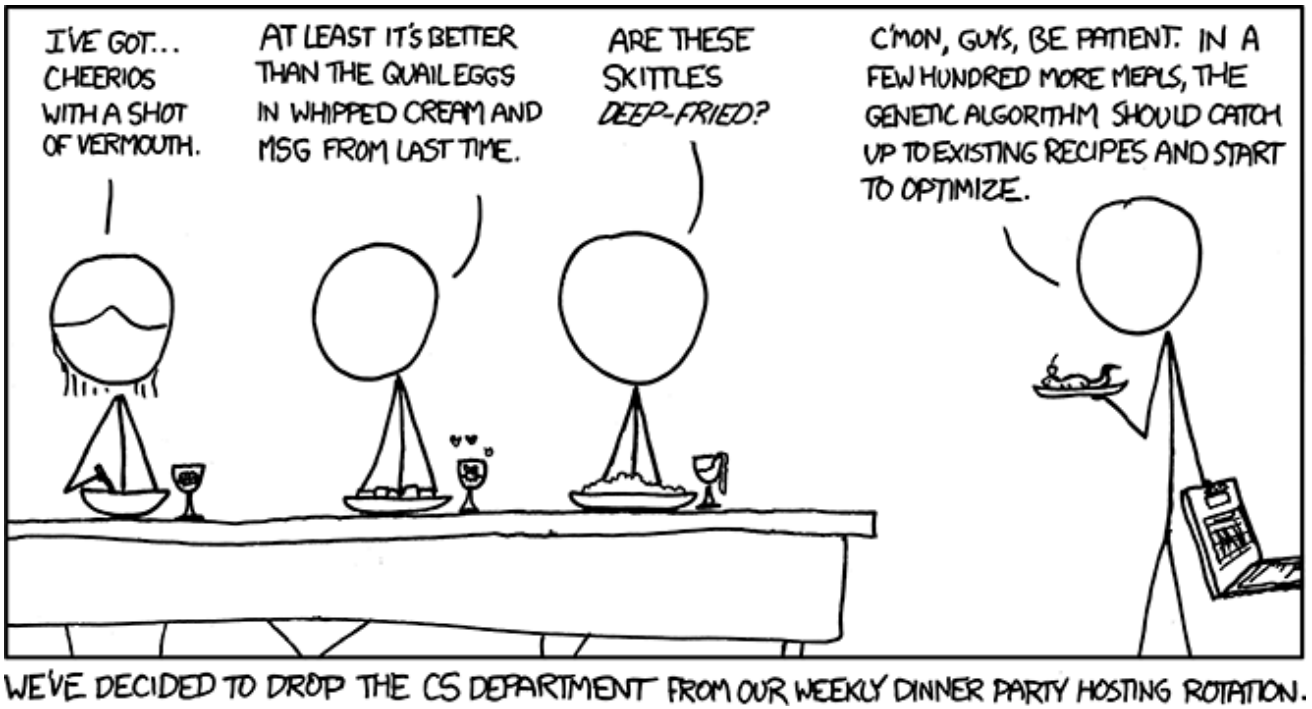
Evolution?



Economists?

- Intrinsic / extrinsic?
- Innate / learned?
- Context-dependent?
- Individual differences?

Exploration

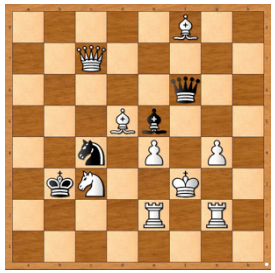


- Epsilon-greedy / softmax?
- Structured exploration?
- Intrinsic goals?
- Sparse rewards

Credit Assignment

Beginning

End

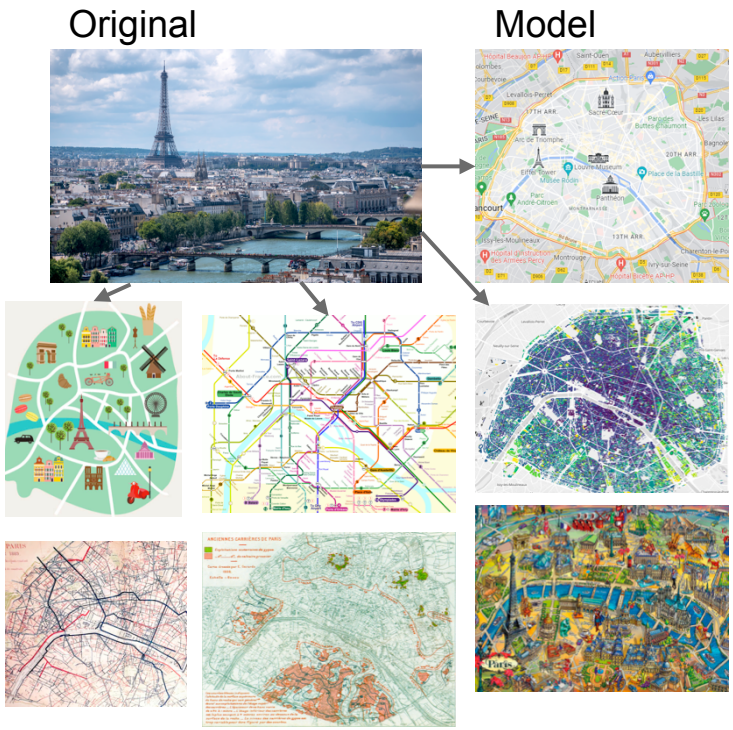


How to link distal outcomes to earlier causes despite many intervening events?

How to generalize over similar + different instances?

How to use knowledge of structure inform credit assignment?

Models as Maps



- Cognitive model = map
  - Smaller, more abstract
  - Loose information
- Different maps
  - Depending on the purpose
  - No one “true” map

Questions?

Want to Learn More?

Books

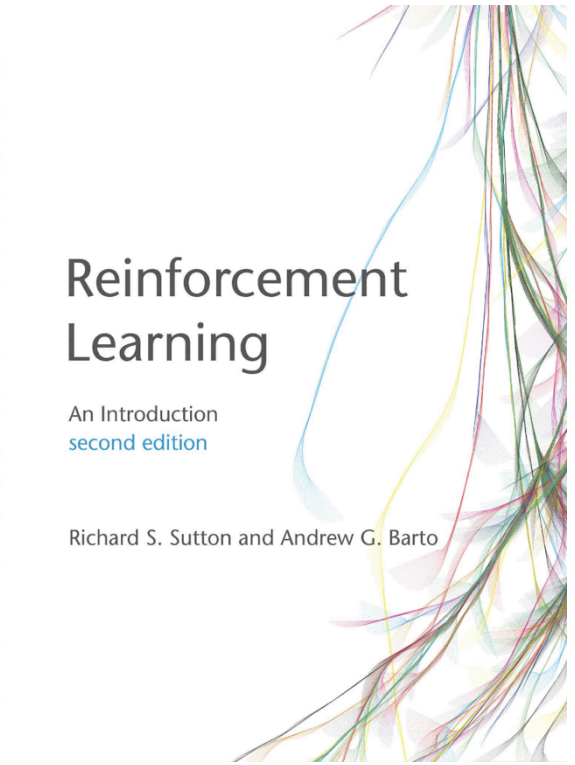
- Reinforcement Learning: an Introduction by Sutton & Barto
- Algorithms for Reinforcement Learning by Csaba Szepesvari

Lectures and course

- Neuromatch Lecture on RL by Jane Wang and Feryal Behbahani
- RL Course by David Silver
- Reinforcement Learning Course | UCL & DeepMind
- Emma Brunskill Stanford RL Course
- RL Course on Coursera by Martha White & Adam White

More practical

- Spinning Up in Deep RL by Josh Achiam
- Acme white paper & Colab tutorial
- OpenAI Gym



Acknowledgements

Kim Stachenfeld, Anne Collins, Jane Wang, Feryal Behbahani, Nathaniel Daw, Chris Knutsen, Kevin Miller, Zeb Kurth-Nelson, Matt Botvinick

Acknowledgements

Slides:


Anne Collins


Kim Stachenfeld

Collaborators at GDM:


Zeb Kurth-Nelson


Kevin Miller


Nathaniel Daw


Chris Summerfield


Dog
tricks
by
Justy