

Model-based RL

**Systems Computational Neuroscience
16th of November 2022**

Philipp Schwartenbeck

Cognitive map seminar attendants: Sorry for the repetition..

Marrian Learning to Predict & Act

Ethology/computation

- Goal: maximise future pleasure; minimise future pain
- Logic: optimal control theory

normative
Bayesian
decision
theory

Psychology/algorithm

- Classical/instrumental conditioning
- Learning from rewards and punishments
- **Model based RL for learning and planning**

constraints of
the substrate;
heuristics;
approximations

Neuroscience/implementation

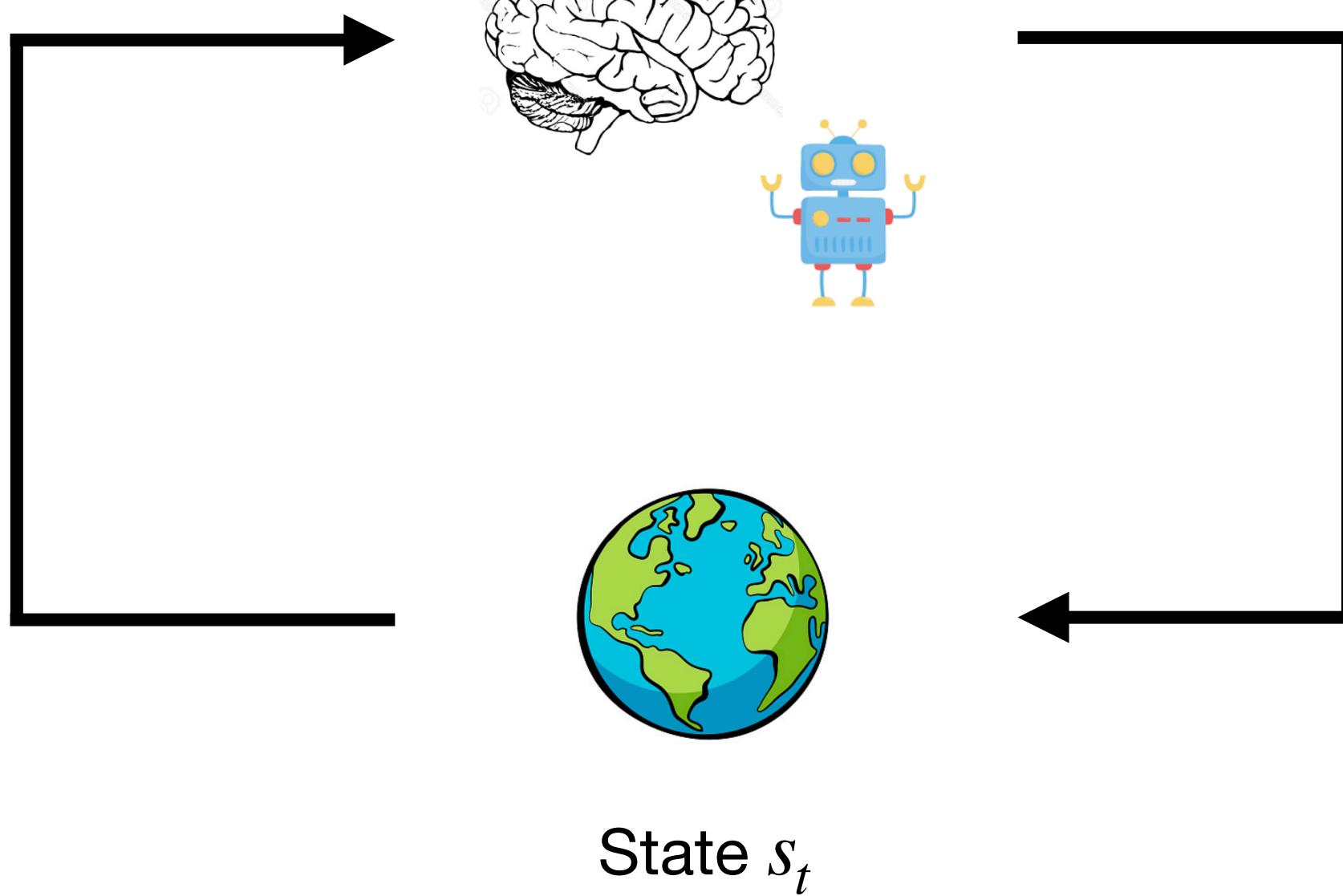
- Prefrontal cortex; hippocampus; entorhinal ctx; striatum
- Neuromodulation

Basic setup of RL?

Based on a reward signal, agents learn **values of actions/states**:

$$V_{\pi}(s) = \mathbb{E}_{\pi}[R | s_0 = s]$$

Reward r_t



Action is governed by a **policy**:

$$\pi(a, s) = P(a_t = a | s_t = s)$$

Agents can learn a **model of the environment** to make smarter decisions, e.g.:

$$P(s_{t+1} = s, r_{t+1} = r | s_t = s, a_t = a)$$

Overview

1. What happened previously..
 - Classical conditioning
 - Instrumental conditioning
2. Model-based reinforcement learning
 - De-valuation
 - Model-based RL for learning
 - Model-based RL for planning and action selection
 - Model-based RL and cognitive maps

Slides will be on slack, all materials here: <https://github.com/schwartenebeckph/Systems-Computational-Neuroscience>

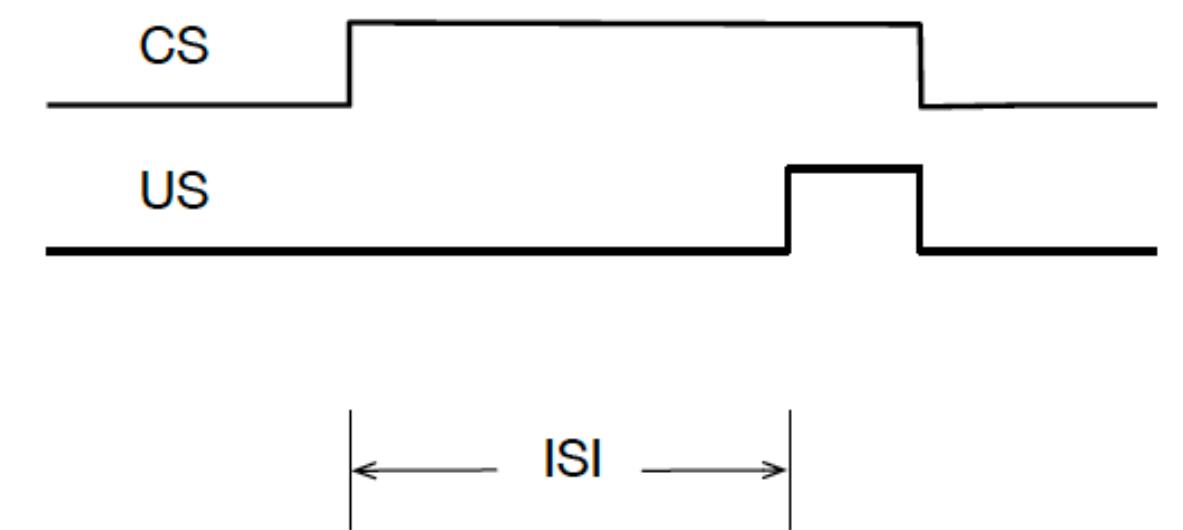
1. What happened previously

1.1 Classical Conditioning

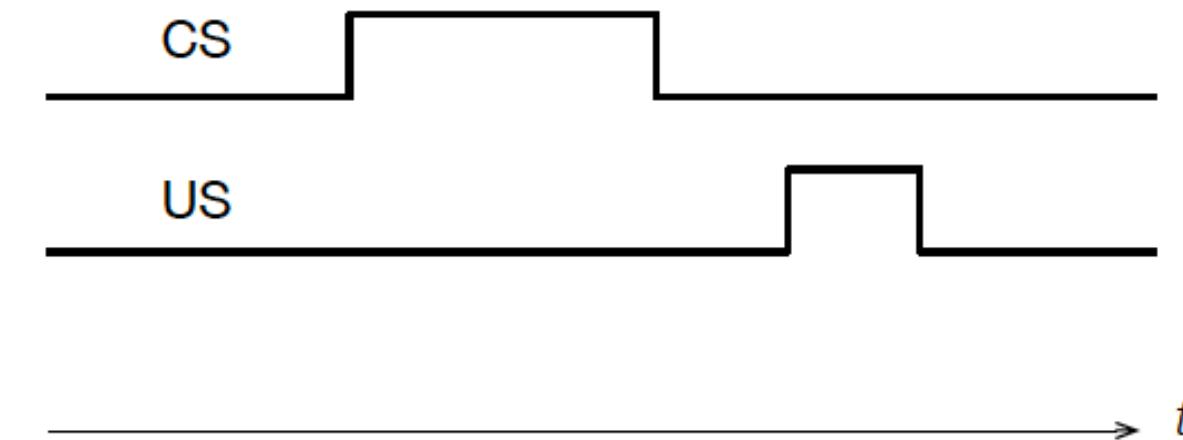
Classical Conditioning

- Two learning algorithms you should know about:
 - **Rescorla-Wagner (RW-)Learning**
 - Learn stimulus-outcome associations
 - **Temporal Difference (TD-)Learning**
 - Learn stimulus-outcome associations across time

Delay Conditioning



Trace Conditioning

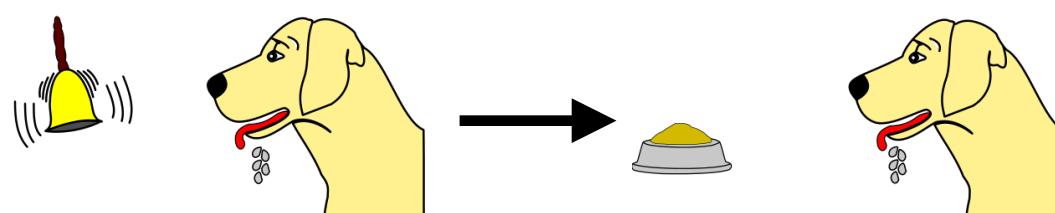


Basics of Learning: Rescorla-Wagner Learning

Learn associative strength between a CS and US

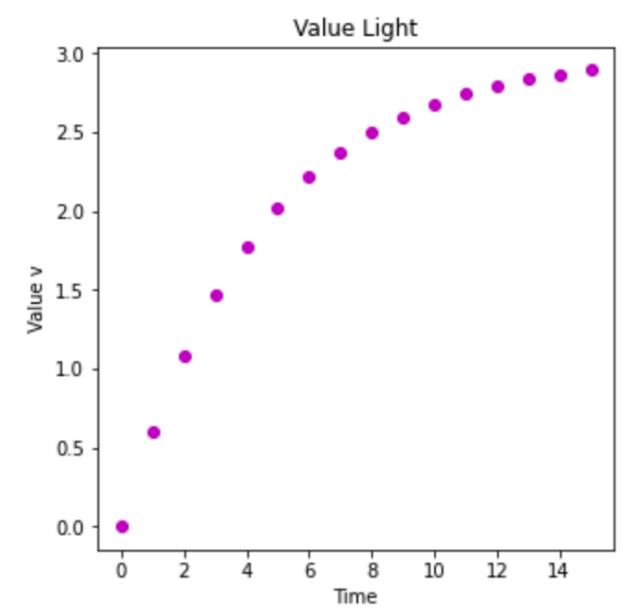
$$V(s) \leftarrow V(s) + \alpha \cdot (r - V(s))$$

↑
Learning rate
Prediction error



A diagram illustrating Rescorla-Wagner Learning. A lightbulb icon is shown with a blue question mark inside, followed by an arrow pointing to a dog's head icon. Below this, the learning rule is shown:

$$V(\text{Light}) \leftarrow V(\text{Light}) + \alpha \cdot (r - V(\text{Light}))$$



[Link to code here](#)

Temporal Difference Learning

- “If one had to identify one idea as **central** and **novel** to reinforcement learning, it would undoubtedly be temporal-difference (TD) learning.”
- Update based on other learned estimates, without waiting for final outcome (**bootstrap**)
 - Learn “a guess from a guess”

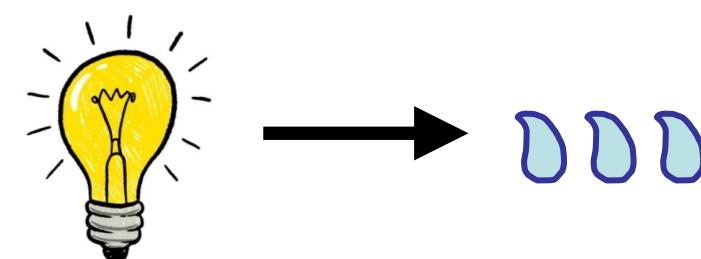
$$V(s_t) \leftarrow V(s_t) + \alpha \cdot (r + \gamma \cdot V(s_{t+1}) - V(s_t))$$

↑ ↑ ↓
Learning rate Discount rate Prediction error

Can RL tell us anything about the brain?

$$V(s_t) \leftarrow V(s_t) + \alpha \cdot (r + \gamma \cdot V(s_{t+1}) - V(s_t))$$

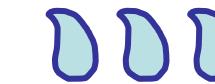
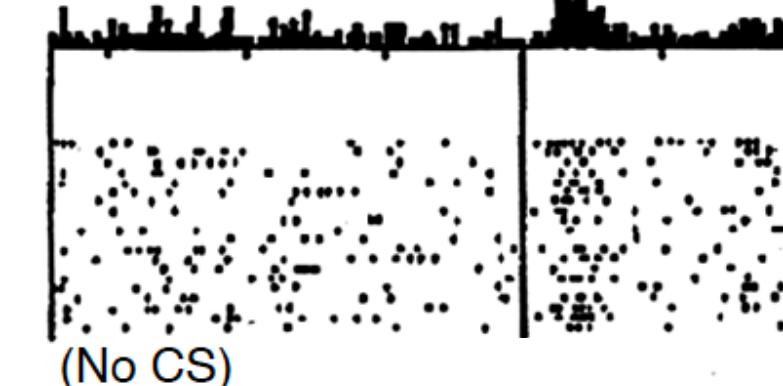
Yes, quite a lot - it looks like **dopamine** (DA) is a key neuromodulator for (TD) reward learning



Dopamine neurons signal
immediate reward

Do dopamine neurons report an error
in the prediction of reward?

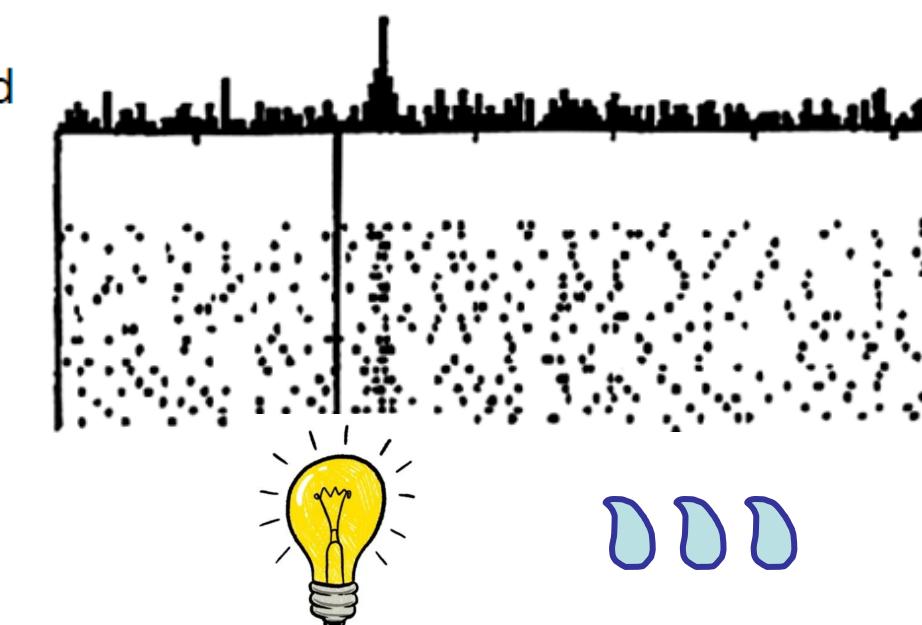
No prediction
Reward occurs



BUT: after training...

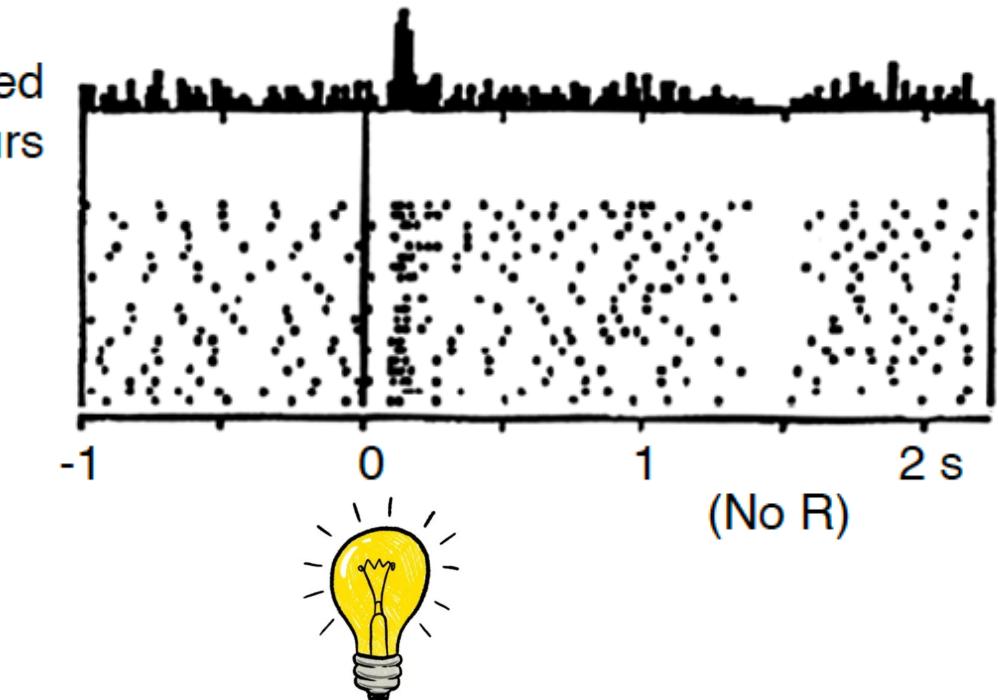
- DA signal reward prediction
- But not correctly predicted reward!

Reward predicted
Reward occurs



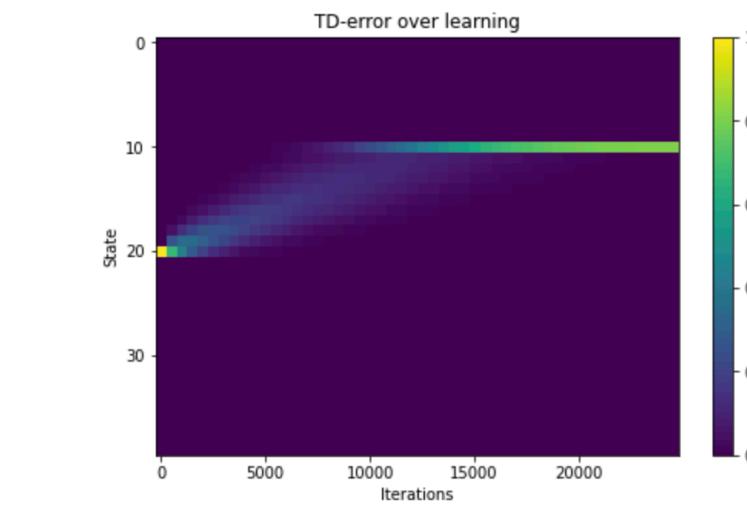
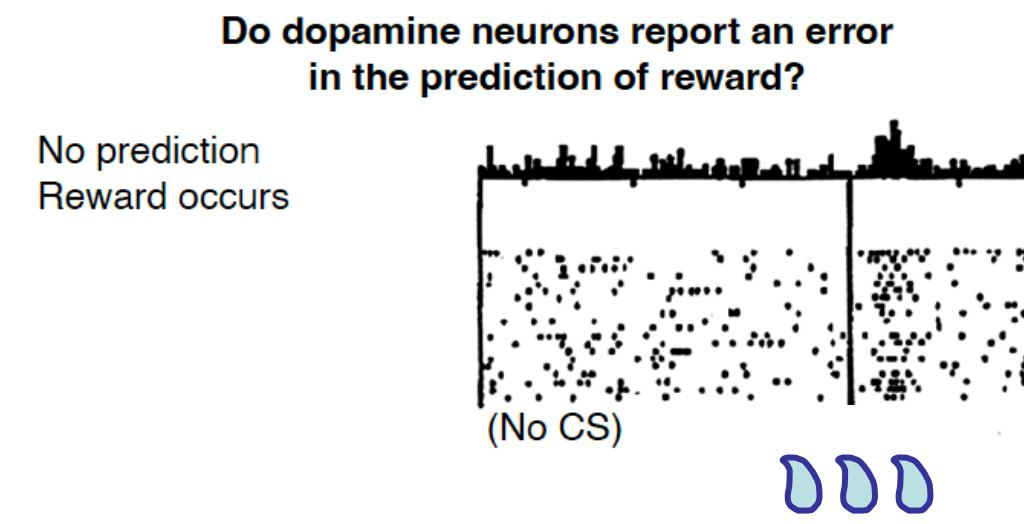
AND: it signals the unexpected
omission of a reward!

Reward predicted
No reward occurs

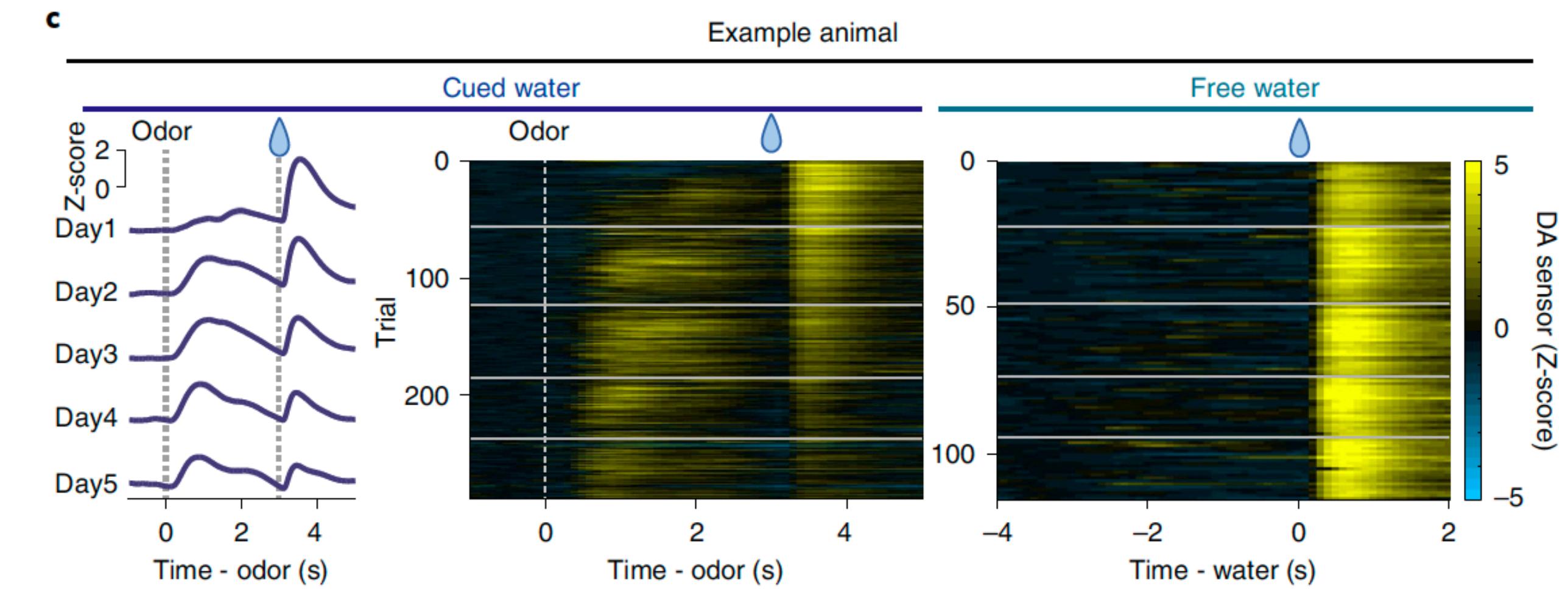
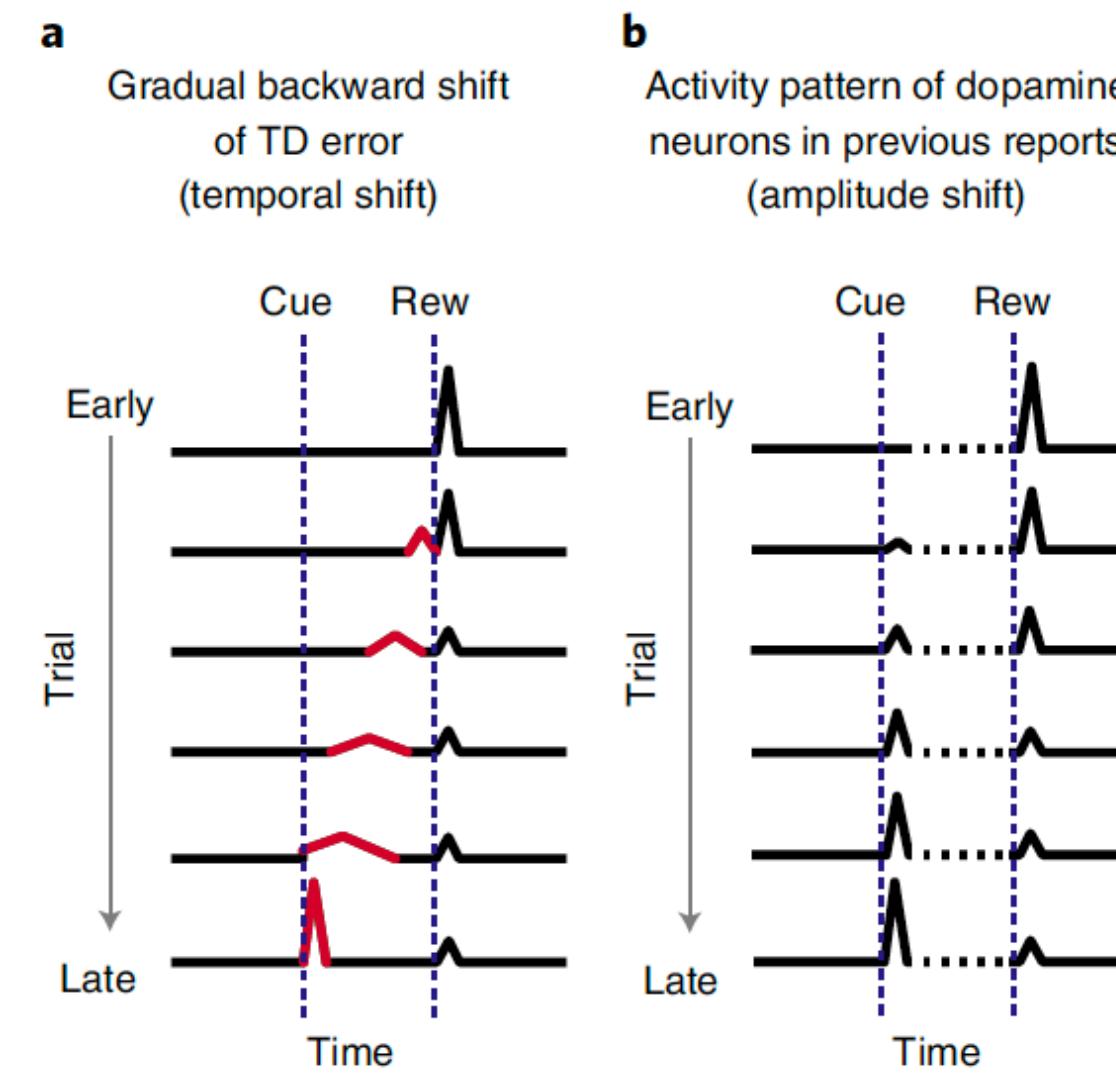


Schultz, Dayan & Montague (Science, 1997)

More TD-learning



Link to code [here](#)

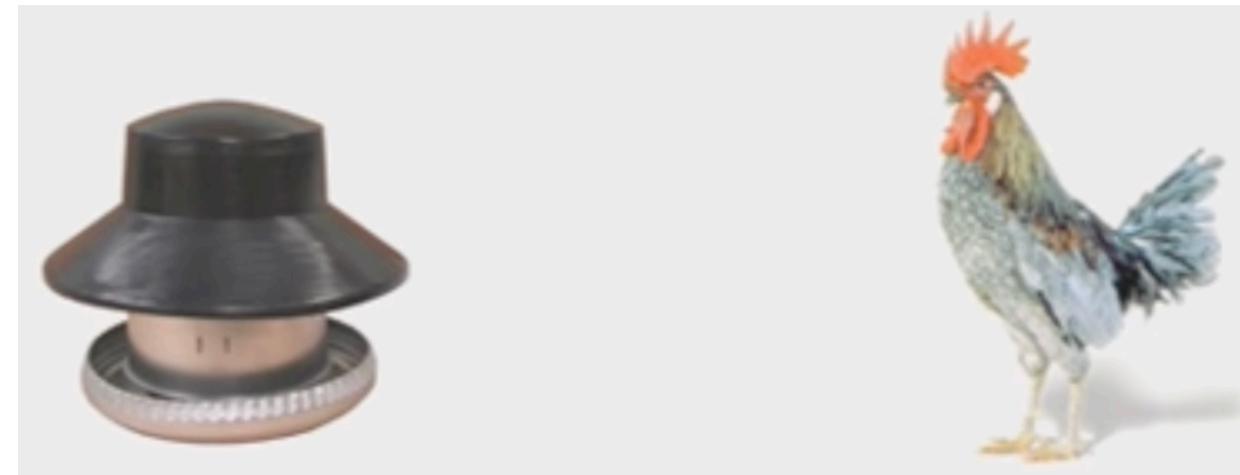


Amo, ..., Watabe-Uchida, Nature Neuroscience 2022

1. What happened previously

1.2 Instrumental Conditioning

Evolutionary Programming



Chicken feeder

Approach: move away



Avoid: move closer

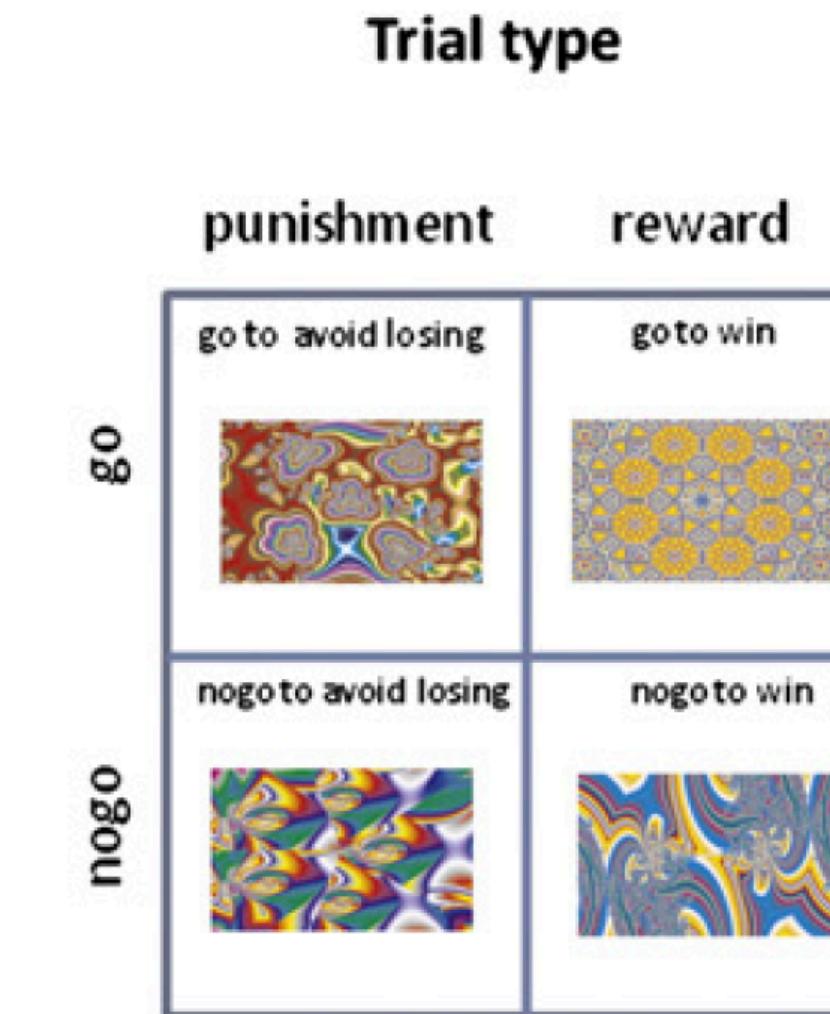
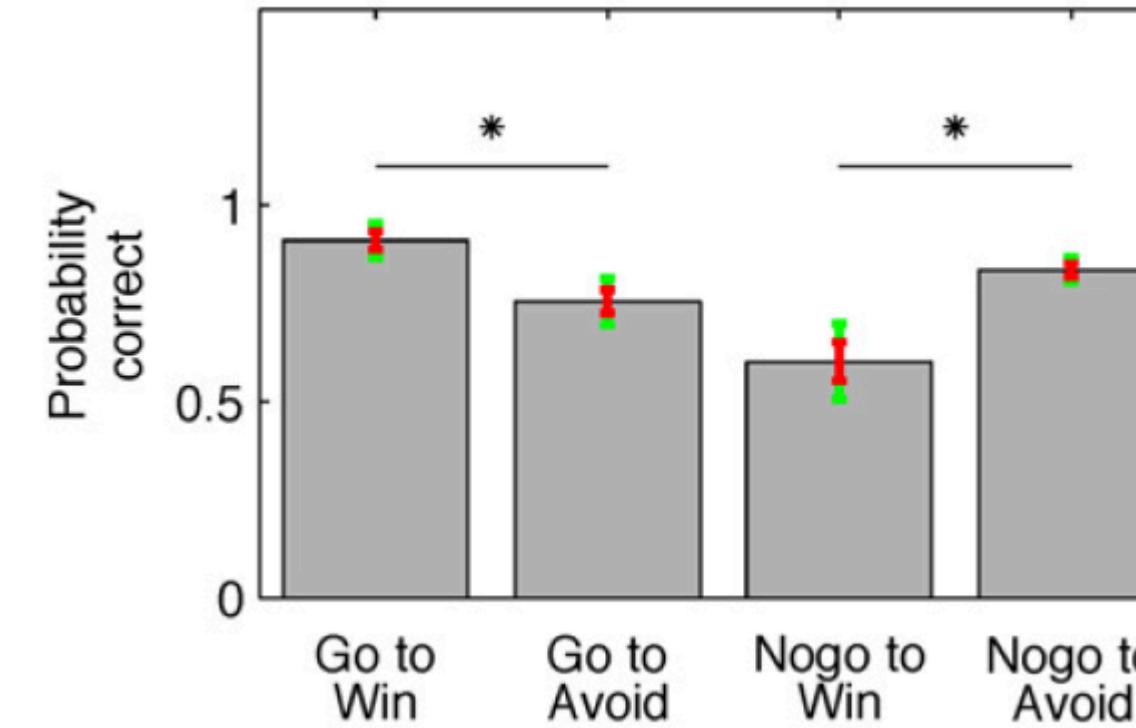


Chicken **never** learn this..

Hershberger, Animal Learning & Behavior, 1986

Evolutionary Programming

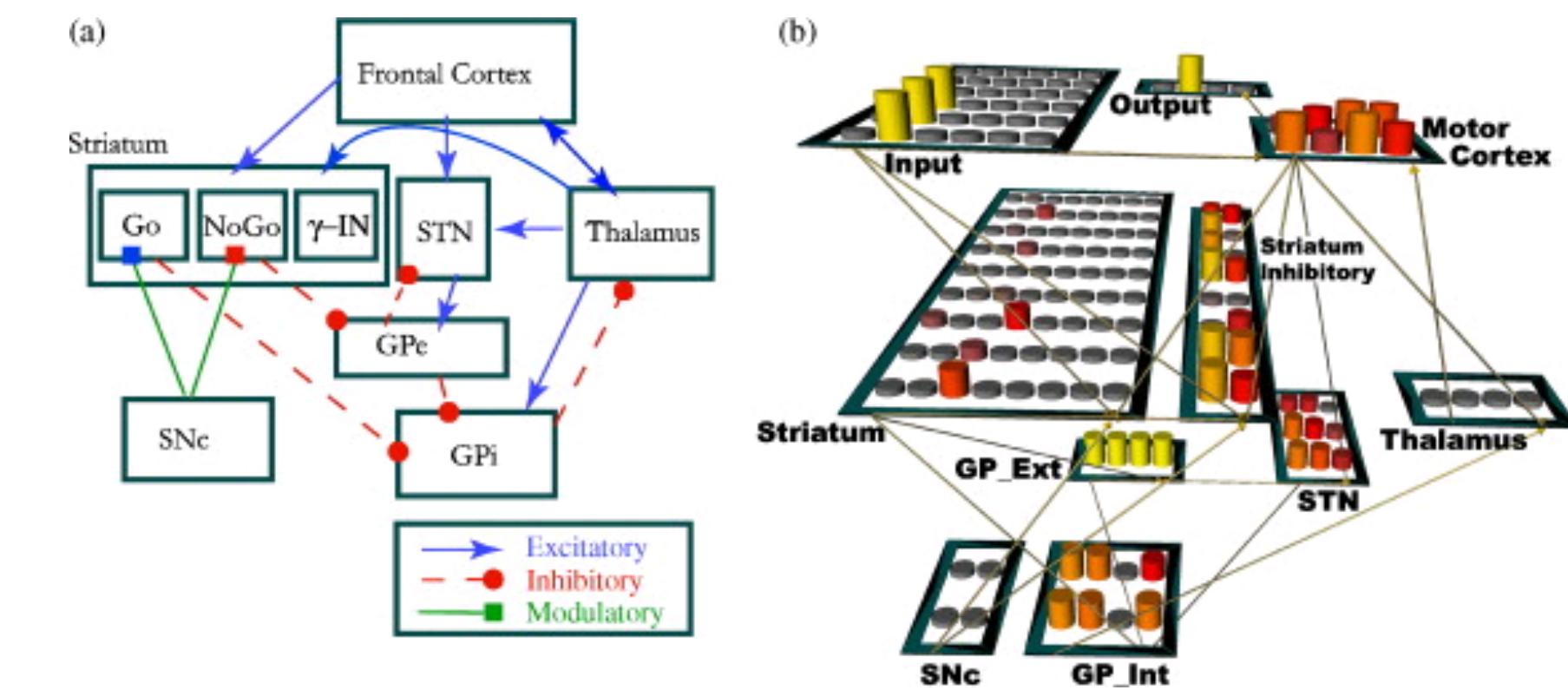
We see a strong signature of this in human behaviour:



Guitart-Masip et al. 2011, 2012

And have biological process models for these effects

- **Direct:** D1: GO; learn from DA increase
- **Indirect:** D2: noGO; learn from DA decrease



Frank & O'Reilly, Behavioural Neuroscience 2006

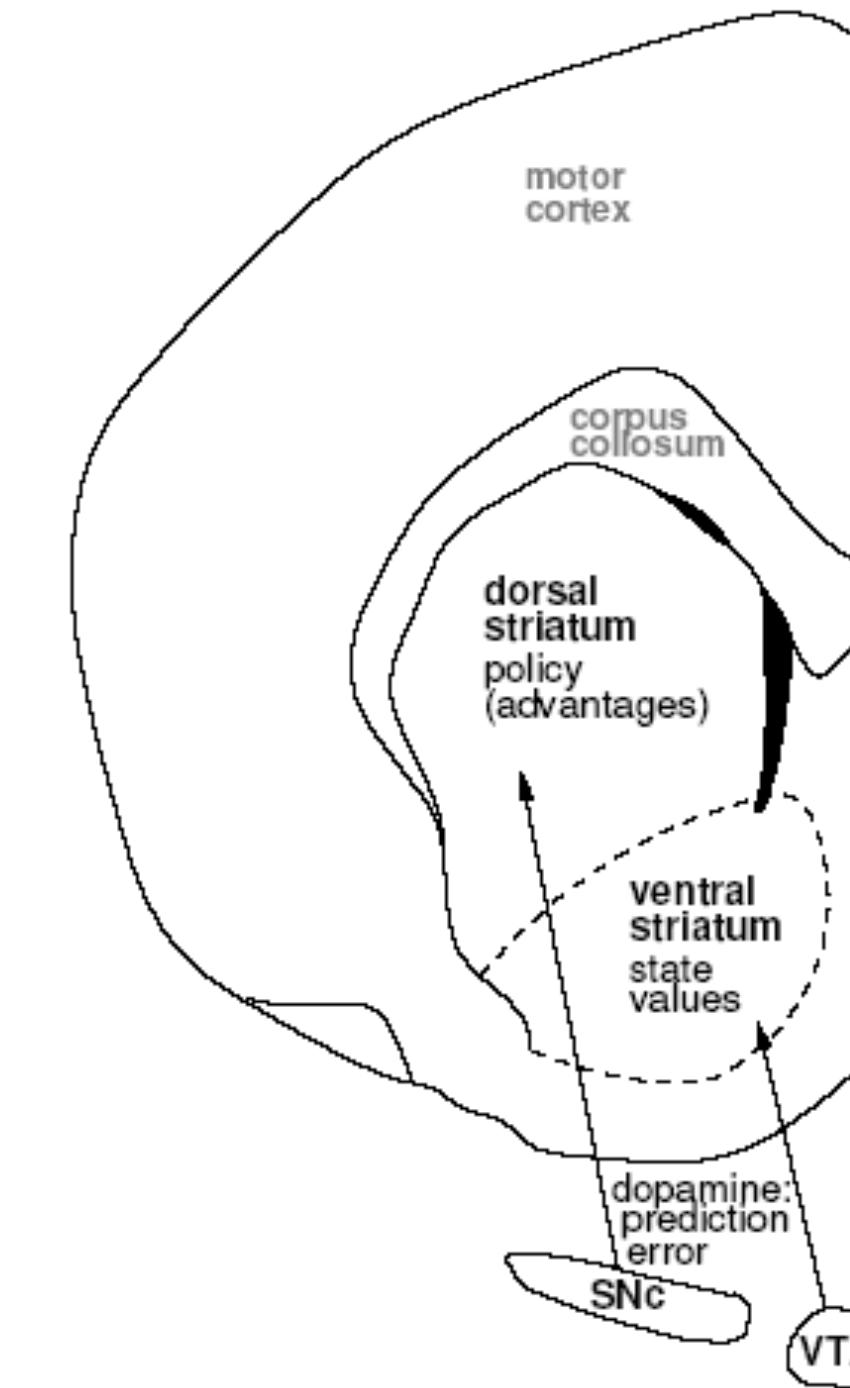
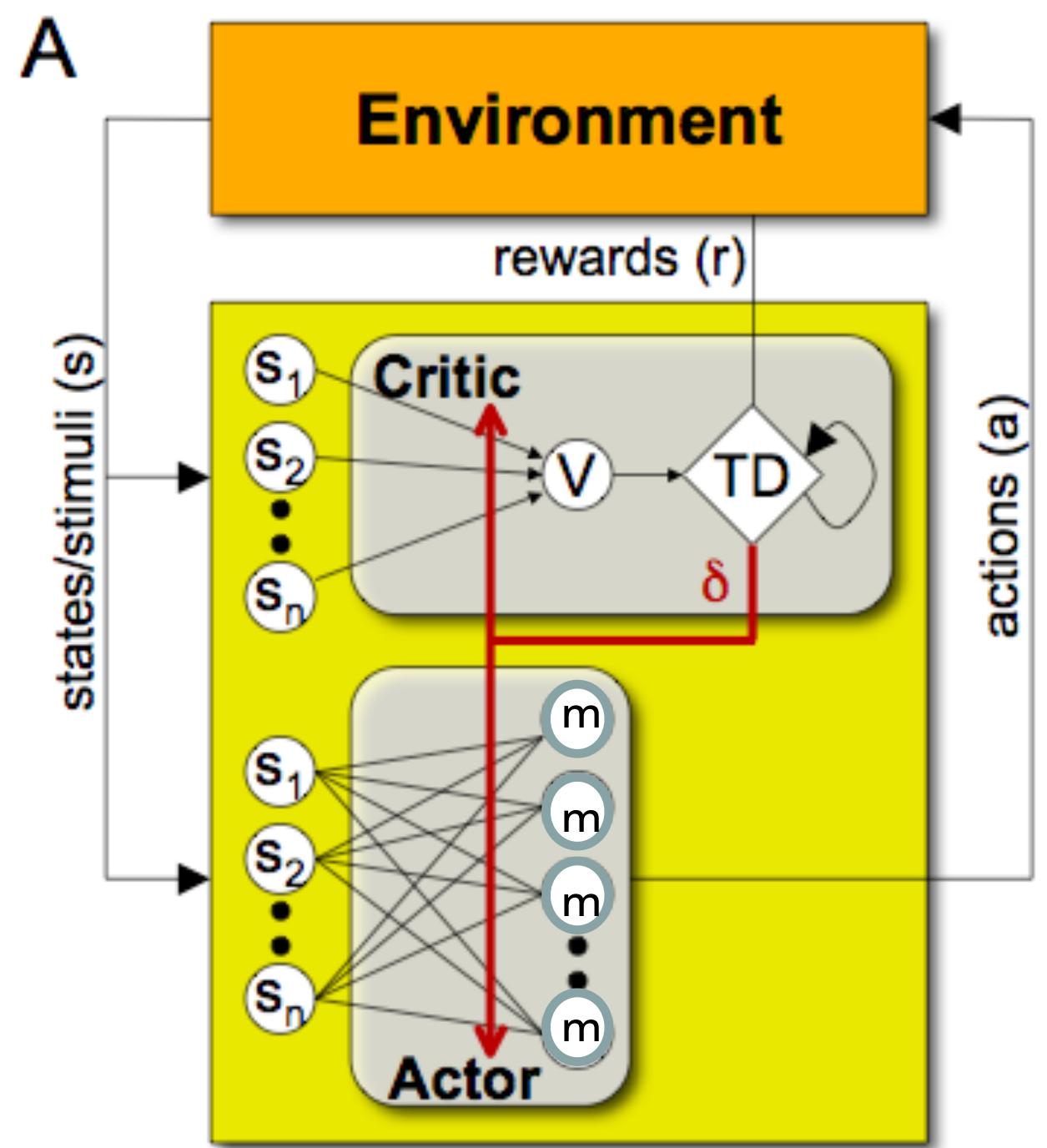
Actor/Critic

Learn values and policies iteratively based on the *same* teaching signal

$$\delta_t = r + \gamma \cdot V(s_{t+1}) - V(s_t)$$

$$V(s_t) \leftarrow V(s_t) + \alpha_v \cdot \delta_t$$

$$\pi(s, a) \leftarrow \pi(s, a) + \alpha_\pi \cdot \delta_t$$



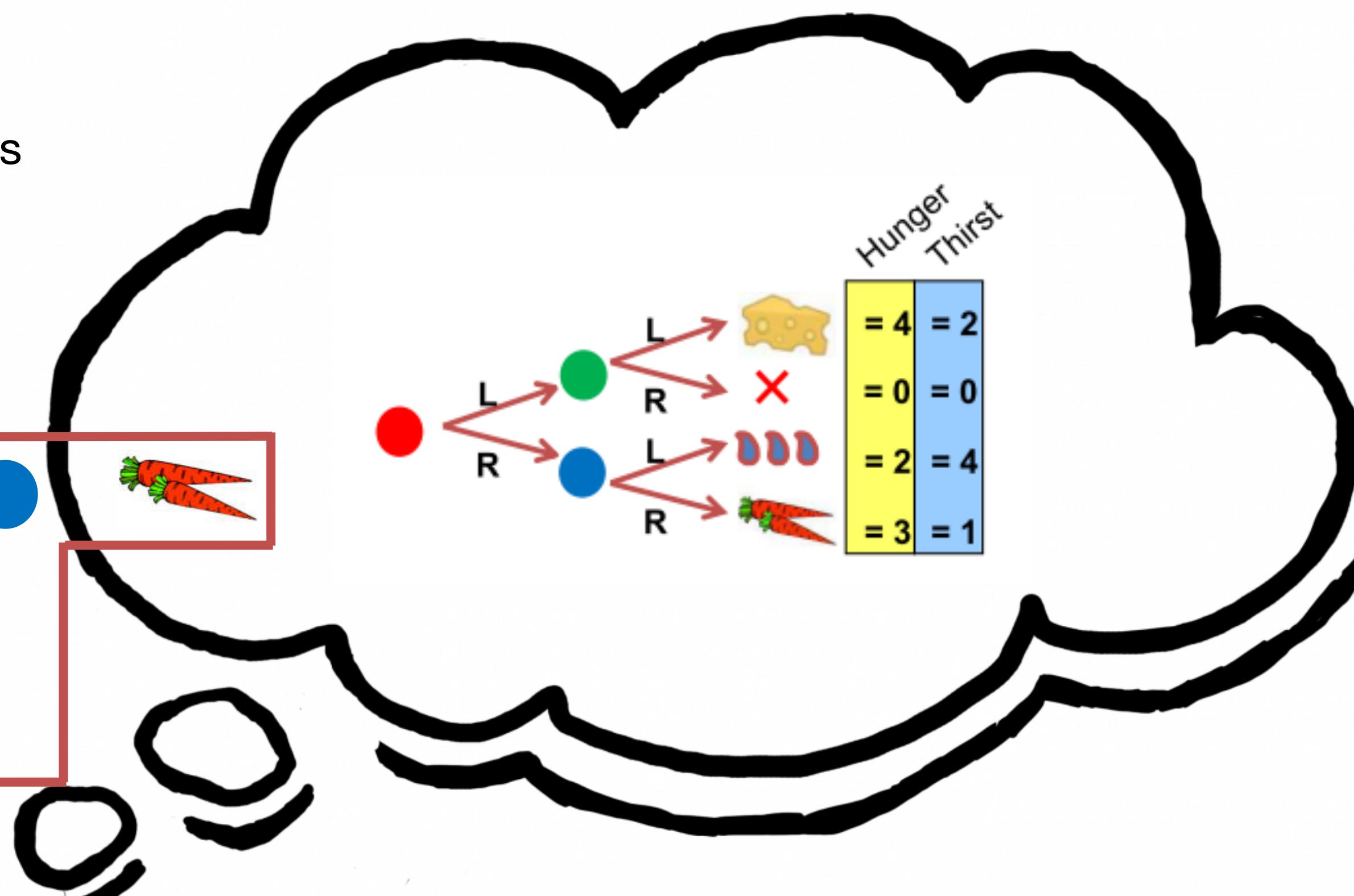
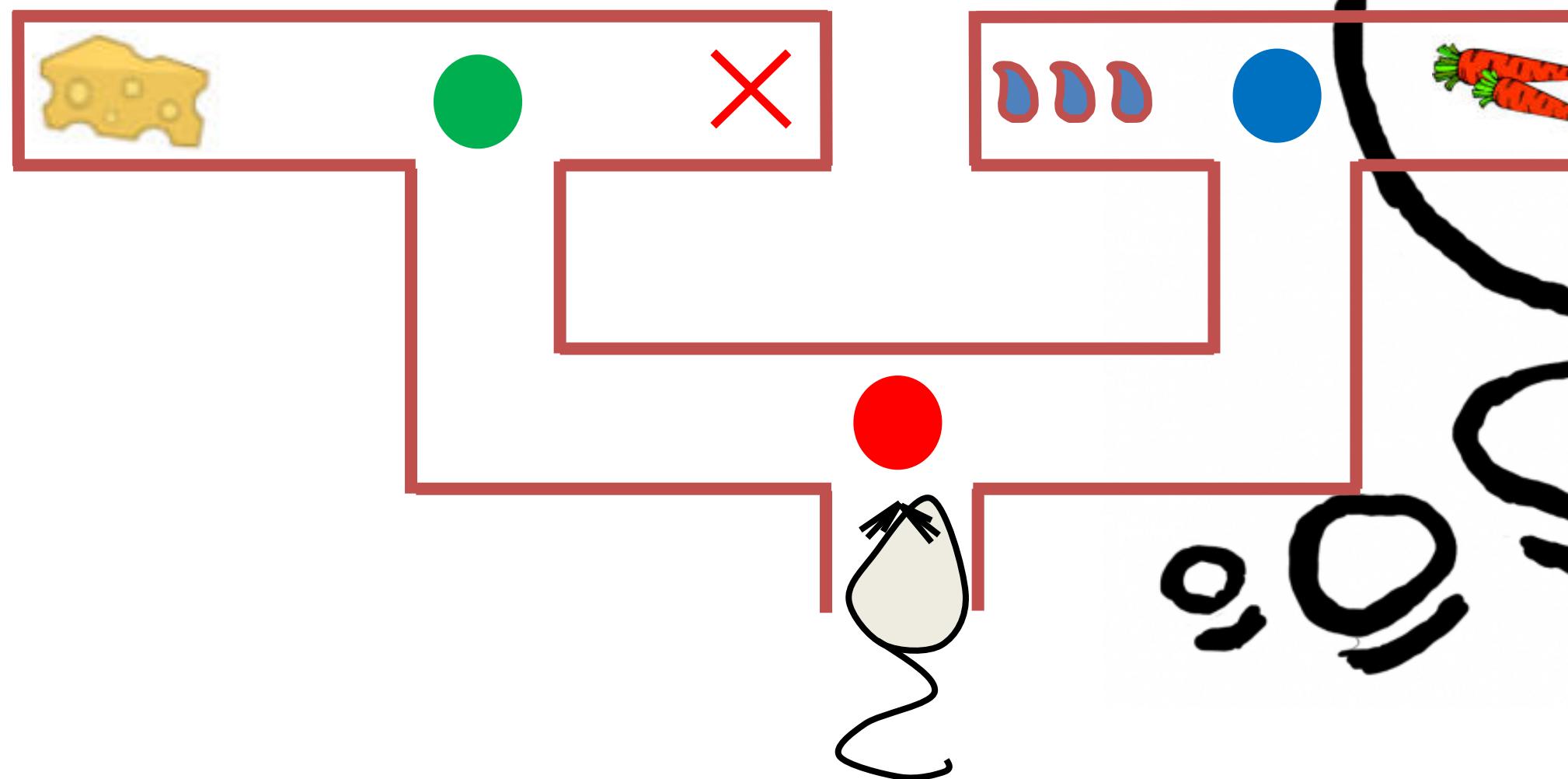
Dopamine signals to both **motivational & motor** regions

Suggestion: training both **values & policies**

Model-based Reinforcement Learning

What if we had a model to plan ahead?

- What should the agent learn if it is
- Hungry
 - Thirsty
 - Hungry first, then thirsty



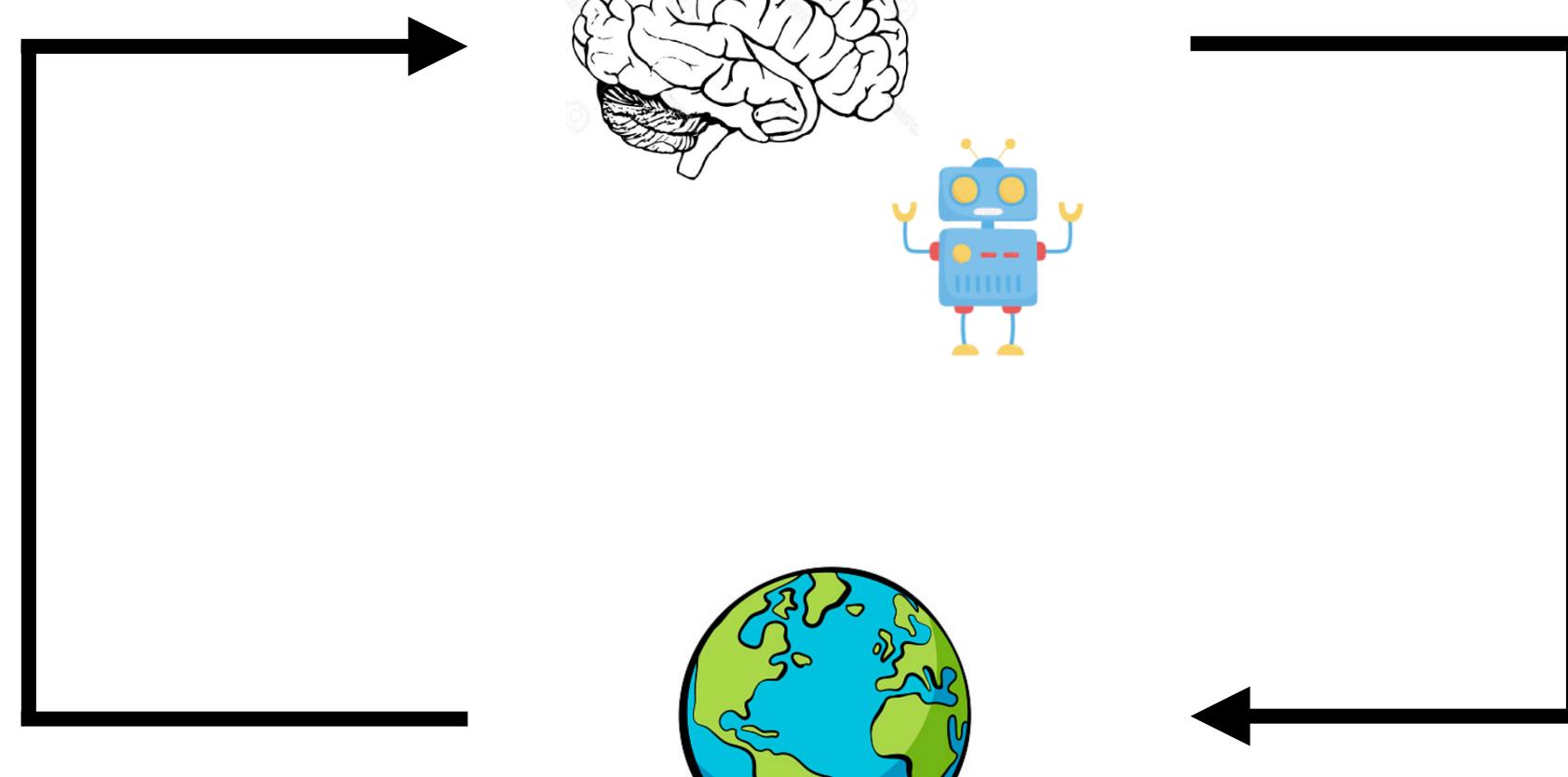
2. Model-based RL

Basic setup: how do agents learn to act?

Based on a reward signal, agents learn **values of actions/states**:

$$V_{\pi}(s) = \mathbb{E}_{\pi}[R \mid s_0 = s]$$

Reward r_t



Action is governed by a **policy**:

$$\pi(a, s) = P(a_t = a \mid s_t = s)$$

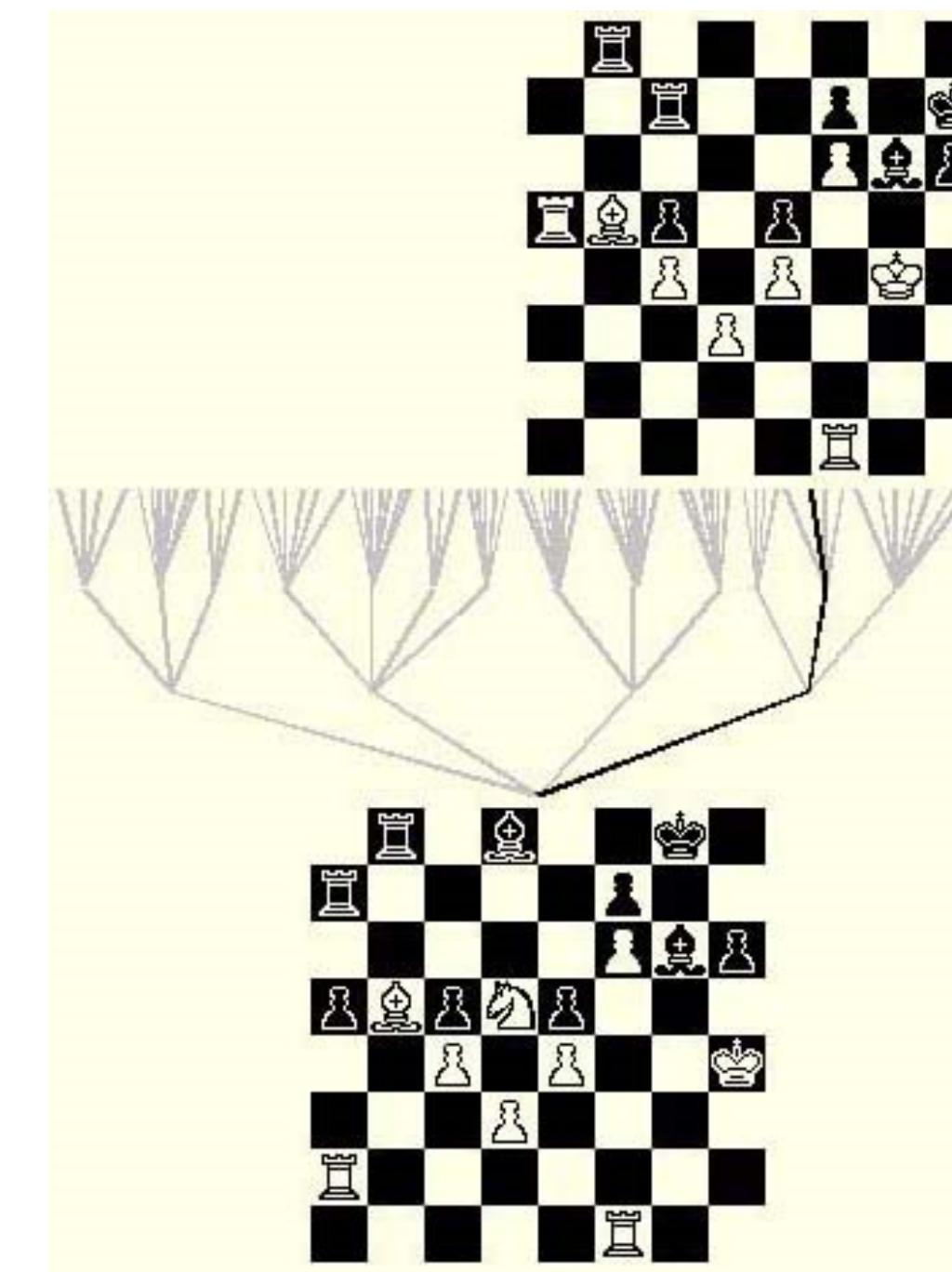
Agents can learn a **model of the environment** to make smarter decisions, e.g.:

$$P(s_{t+1} = s, r_{t+1} = r \mid s_t = s, a_t = a)$$

Different Decision-Makers

(At least) 3 ways to solve this:

1. Tree search
2. Position evaluation
3. [Situation memory]



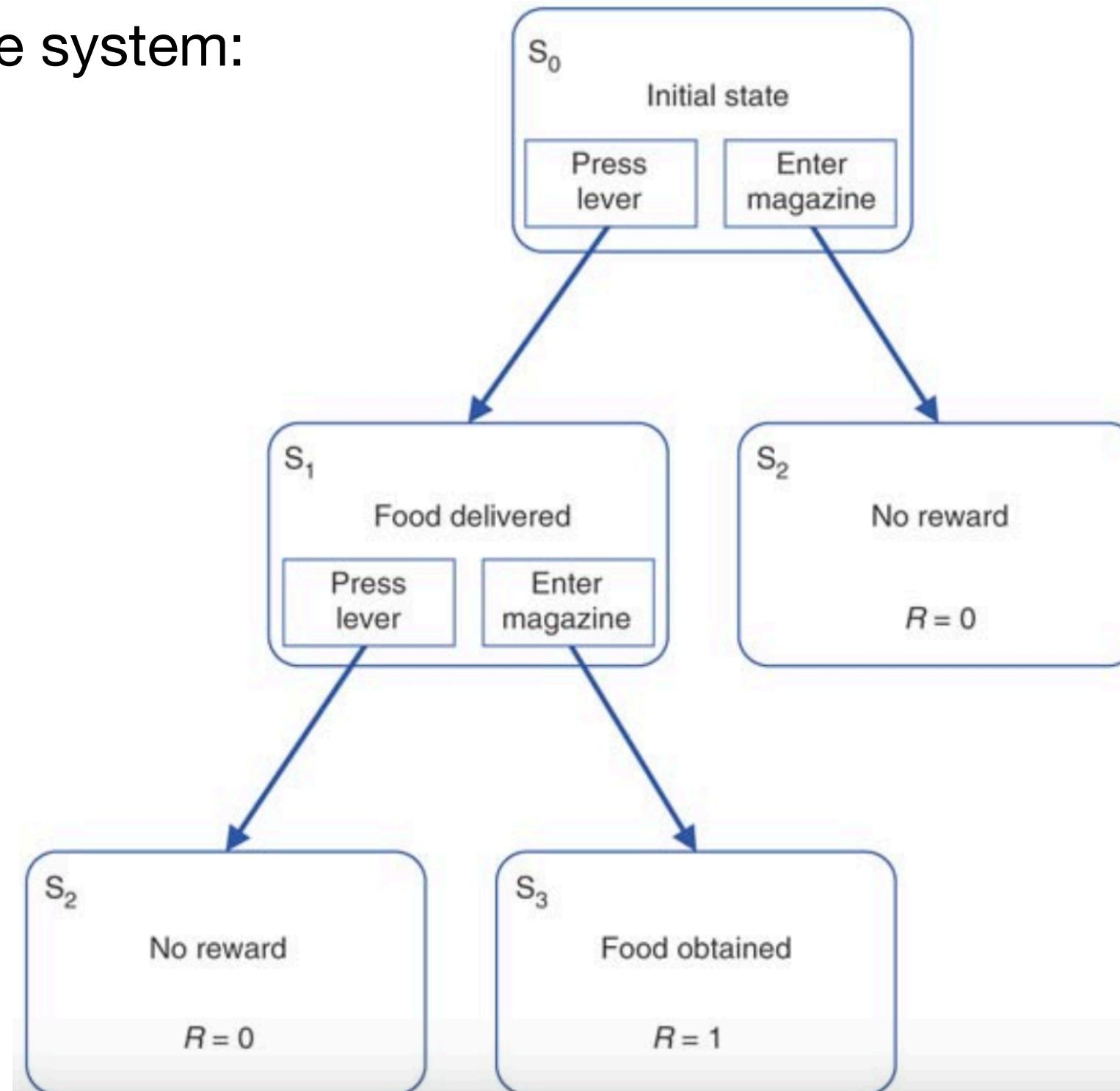
Multiple Systems in RL

Model-based RL

- Build a forward model of the task and outcomes
- Search in the forward model
- Optimal use of information, but computationally ruinous
- **Cached-based (model-free) RL**
 - Learn values, which summarise future worth
 - Computationally trivial but bootstrap-based - statistically inefficient
- Learn both – select according to **uncertainty**
 - Pro: **Robust** and **adaptive**
 - Con: **Competition** (and harder reverse engineering)

Multiple Systems in RL

Tree system:



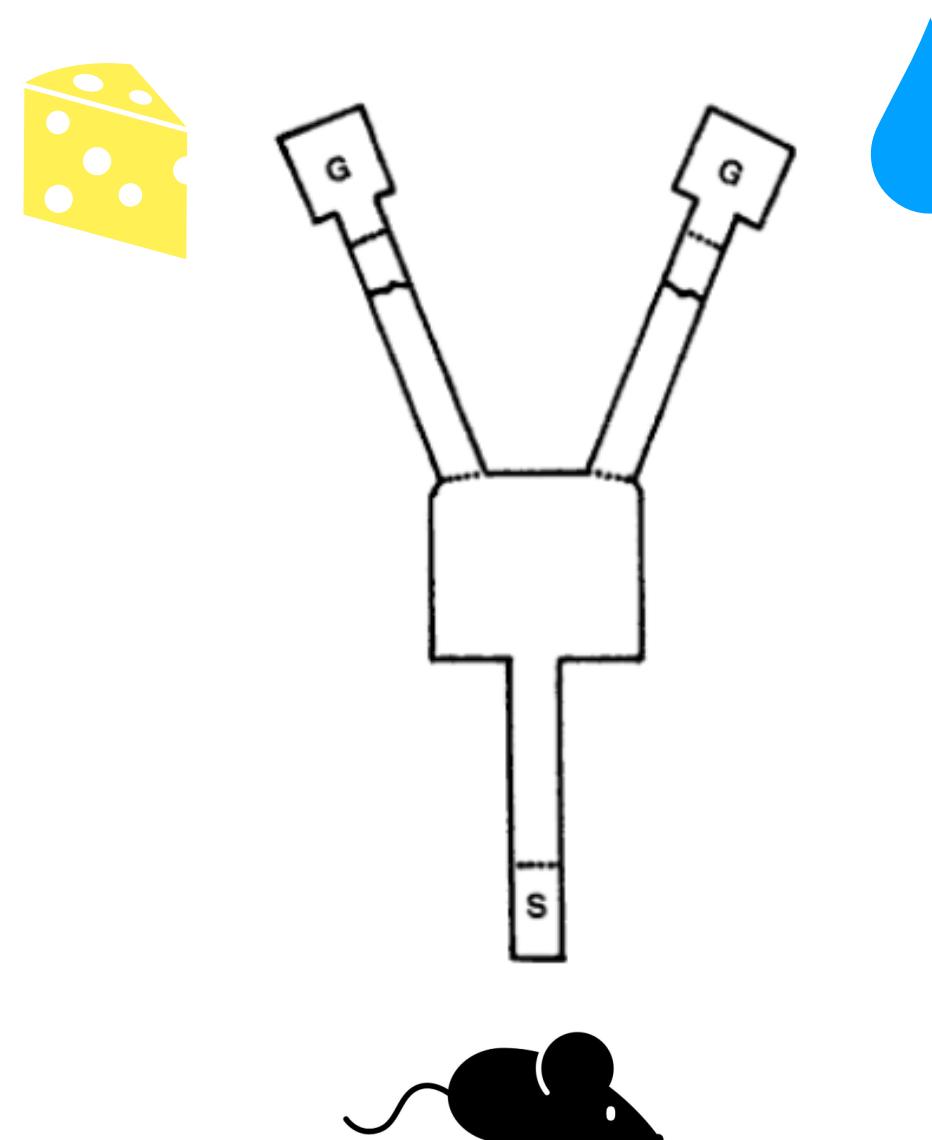
Cache system:



De-valuation

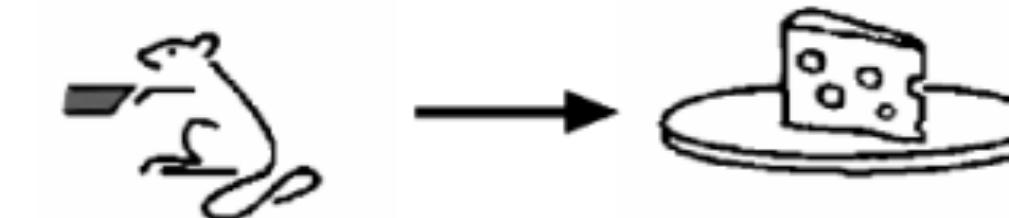
Outcome devaluation (*revaluation*): gold-standard test for forward model predicting outcomes of actions

Animal is trained to perform two different actions, with a different reward:

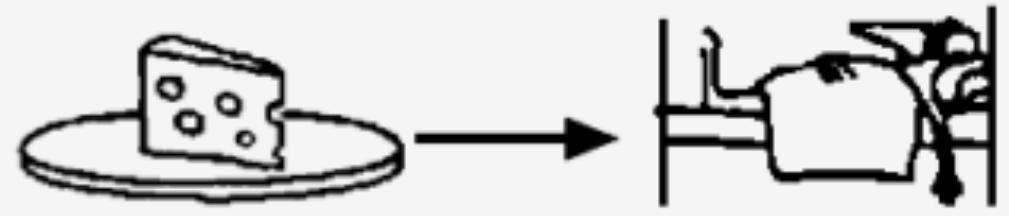


One reward is then **devalued**, for example by satiation.

1. **training**
(hungry)



2. **devaluation**



3. **test**

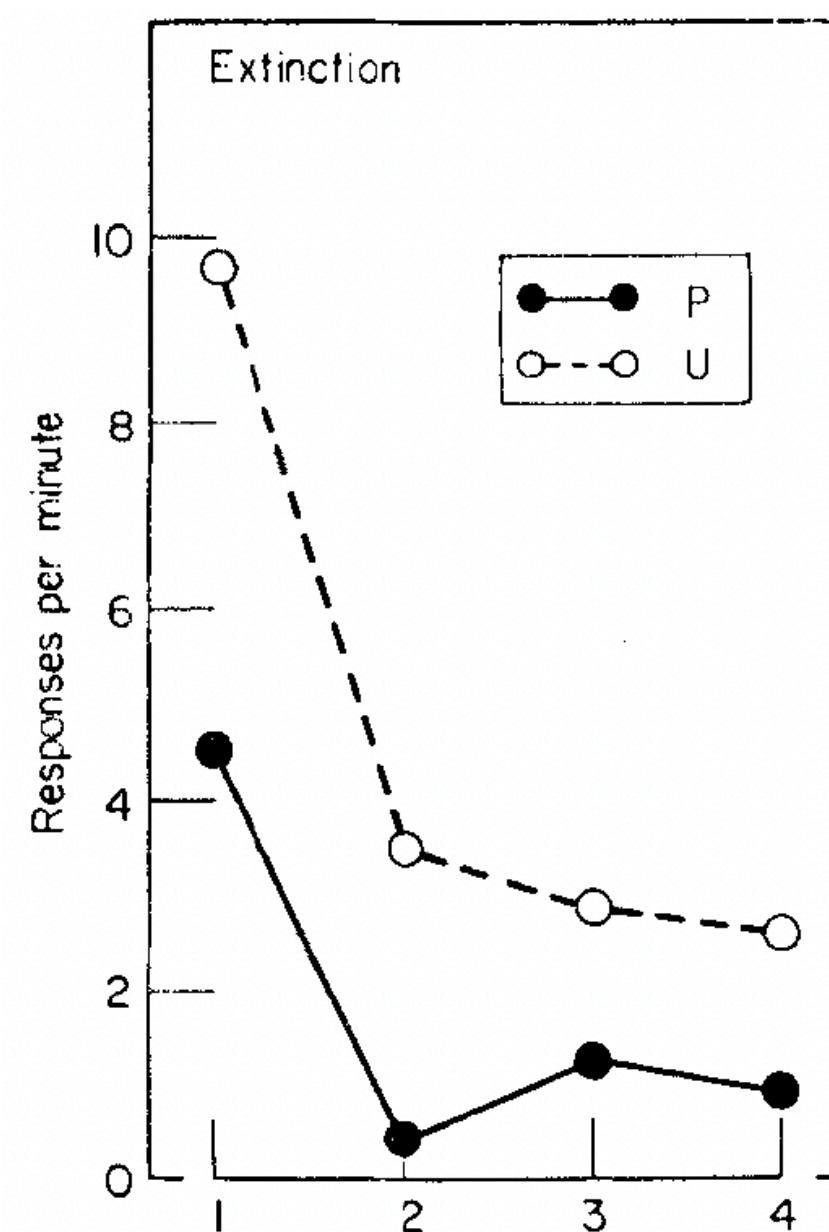


Drummond & Niv, Current Biology 2020

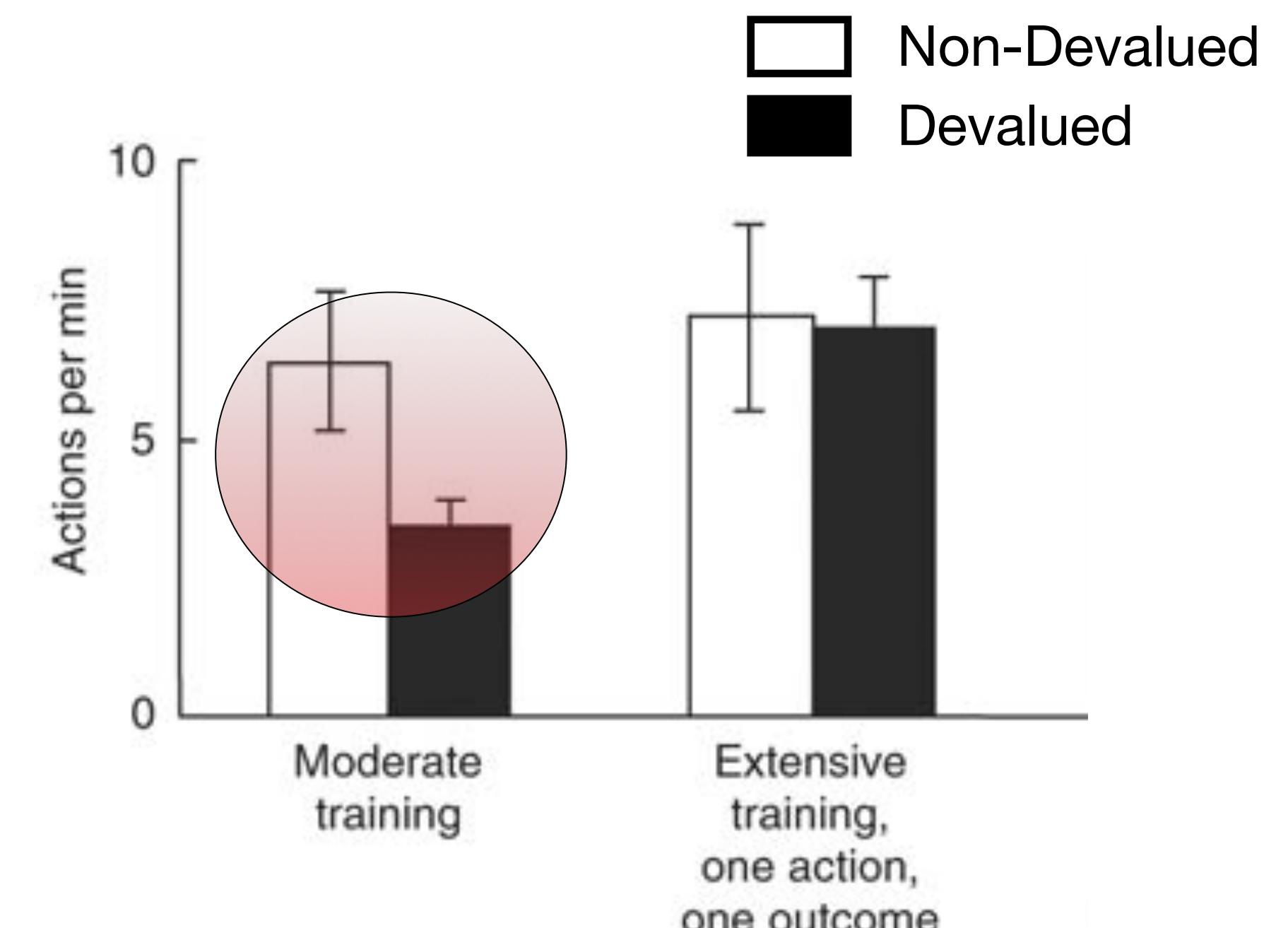
De-valuation

Outcome devaluation (*revaluation*): gold-standard test for forward model predicting outcomes of actions

Impact of this devaluation is tested in ‘extinction’, without providing outcomes.



Adams & Dickinson, Quarterly Journal of Experimental Psychology, 1981
Colwill & Rescorla, Journal of Experimental Psychology, 1985



Holland, Journal of Experimental Psychology, 2004
Dickinson & Balleine, Animal, Learning & Behaviour, 1994

Break?

What is the model in model-based RL?

$$P(s', r | s, a) = P(s_{t+1} = s', r_{t+1} = r | s_t = s, a_t = a)$$

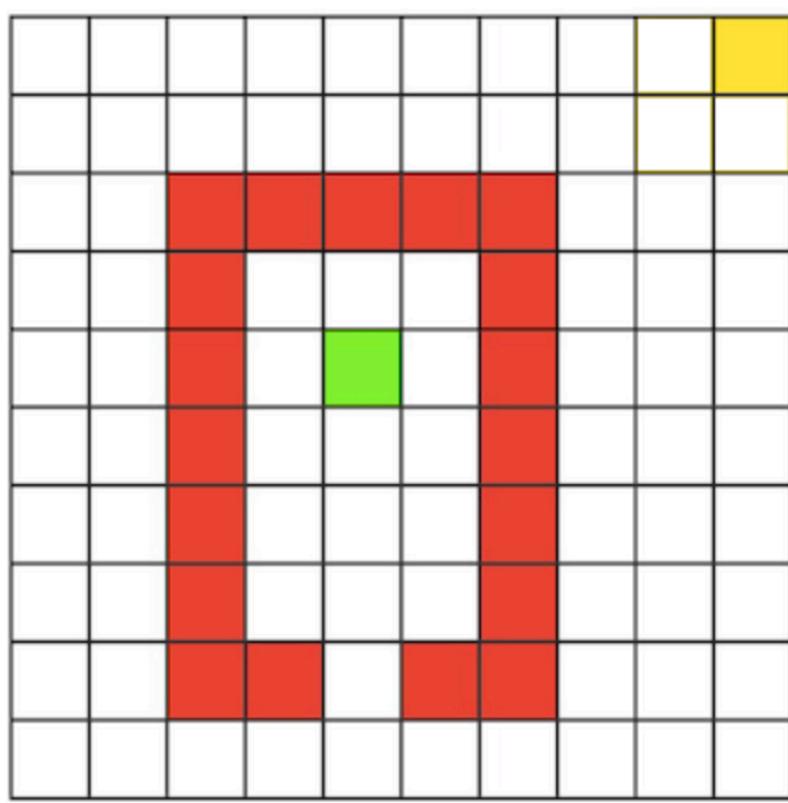
How can we make use of such models of the world?

Learning

- Key idea: store experiences in world model $P(s', r | s, a)$
- Sample from this model to generate extra learning data
- This is called **DYNA-Q...**

DYNA-Q

Sample from world model $P(s', r | s, a)$ to generate additional learning data



$$P(s', r | s, a) = P(s_{t+1} = s', r_{t+1} = r | s_t = s, a_t = a)$$

$Model(S, A) \leftarrow R, S'$

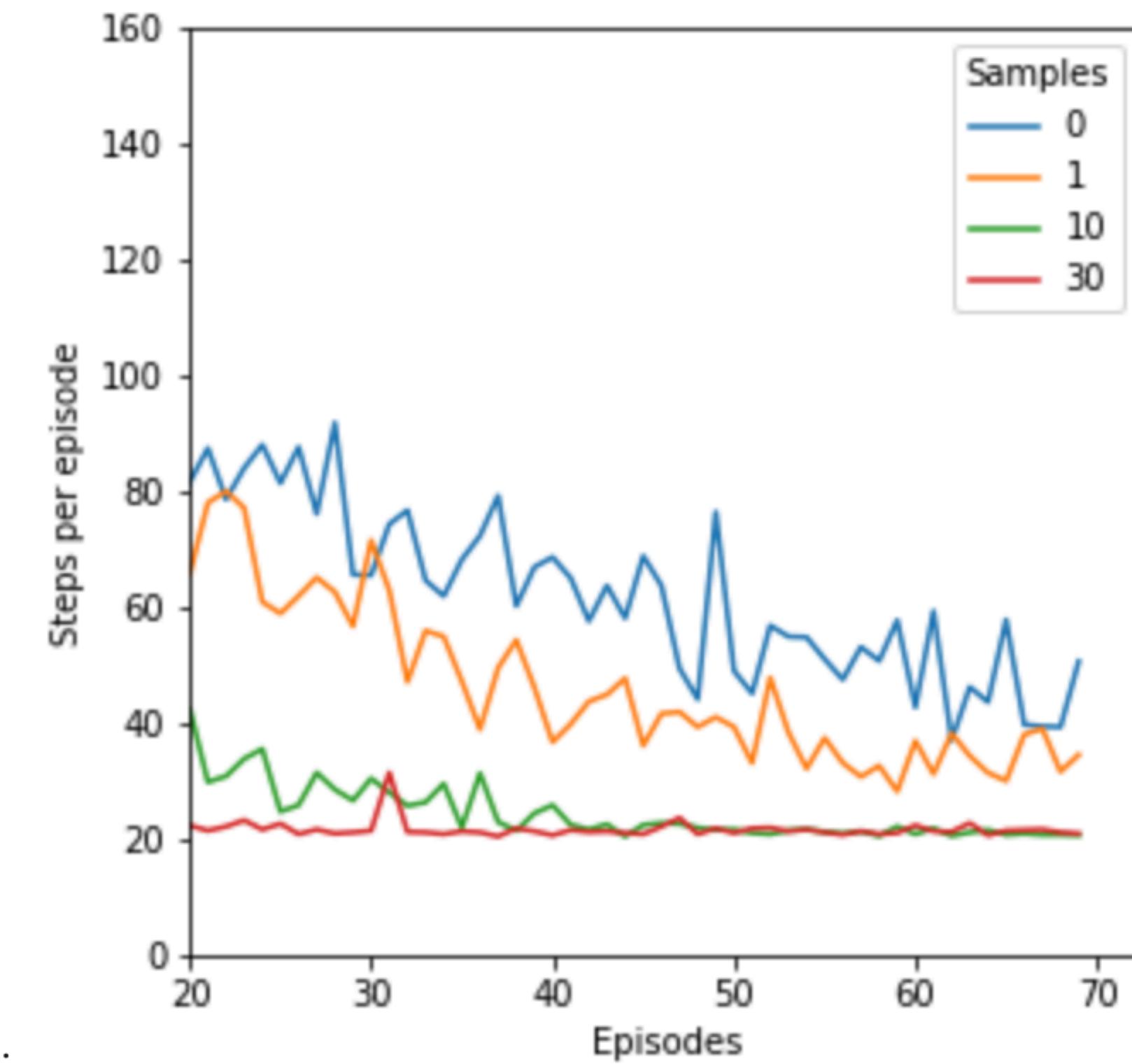
And during breaks ('offline rest'), they can sample from this experience and learn from these samples:

$S \leftarrow$ previously observed state

$A \leftarrow$ action previously taken in S

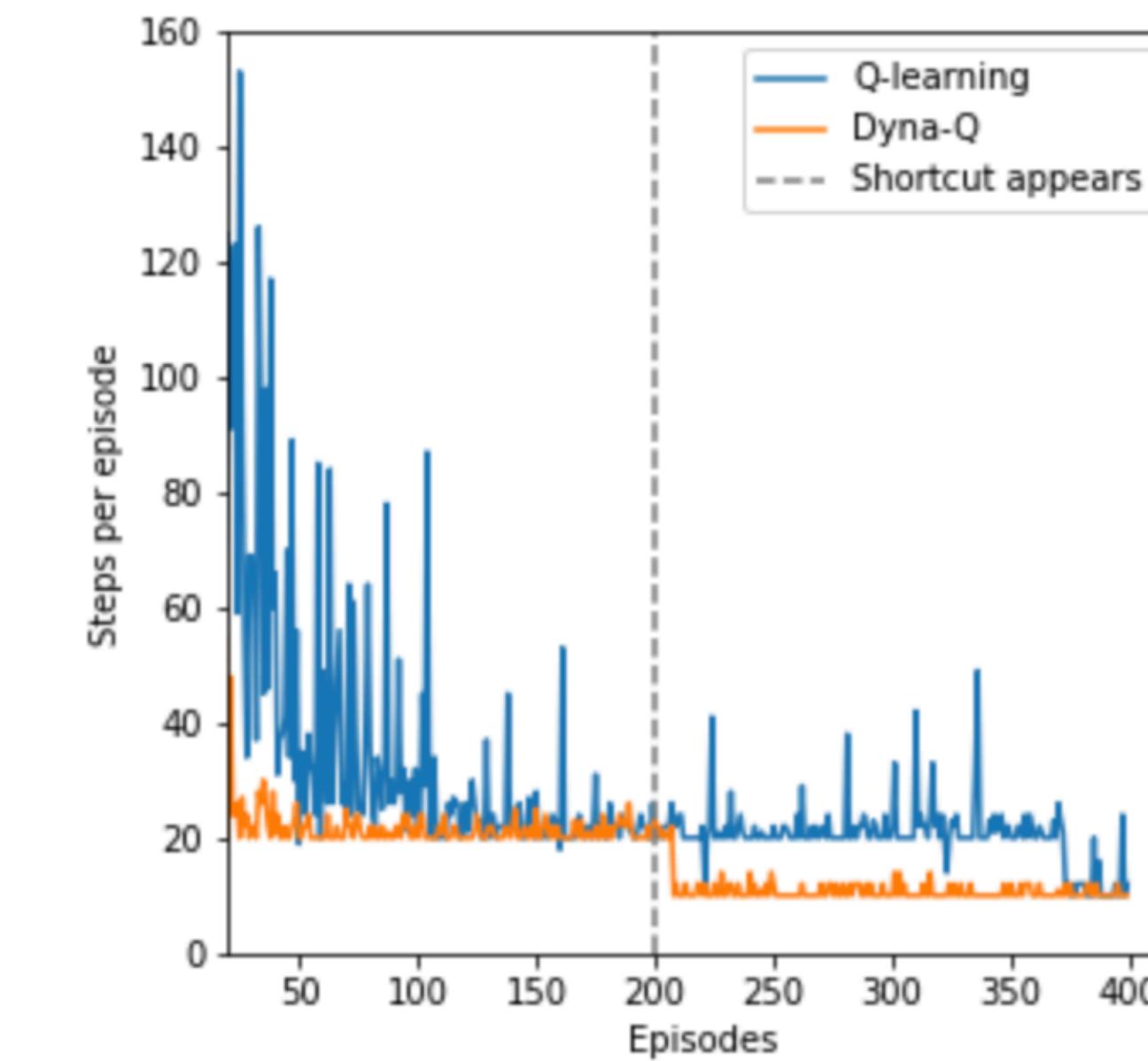
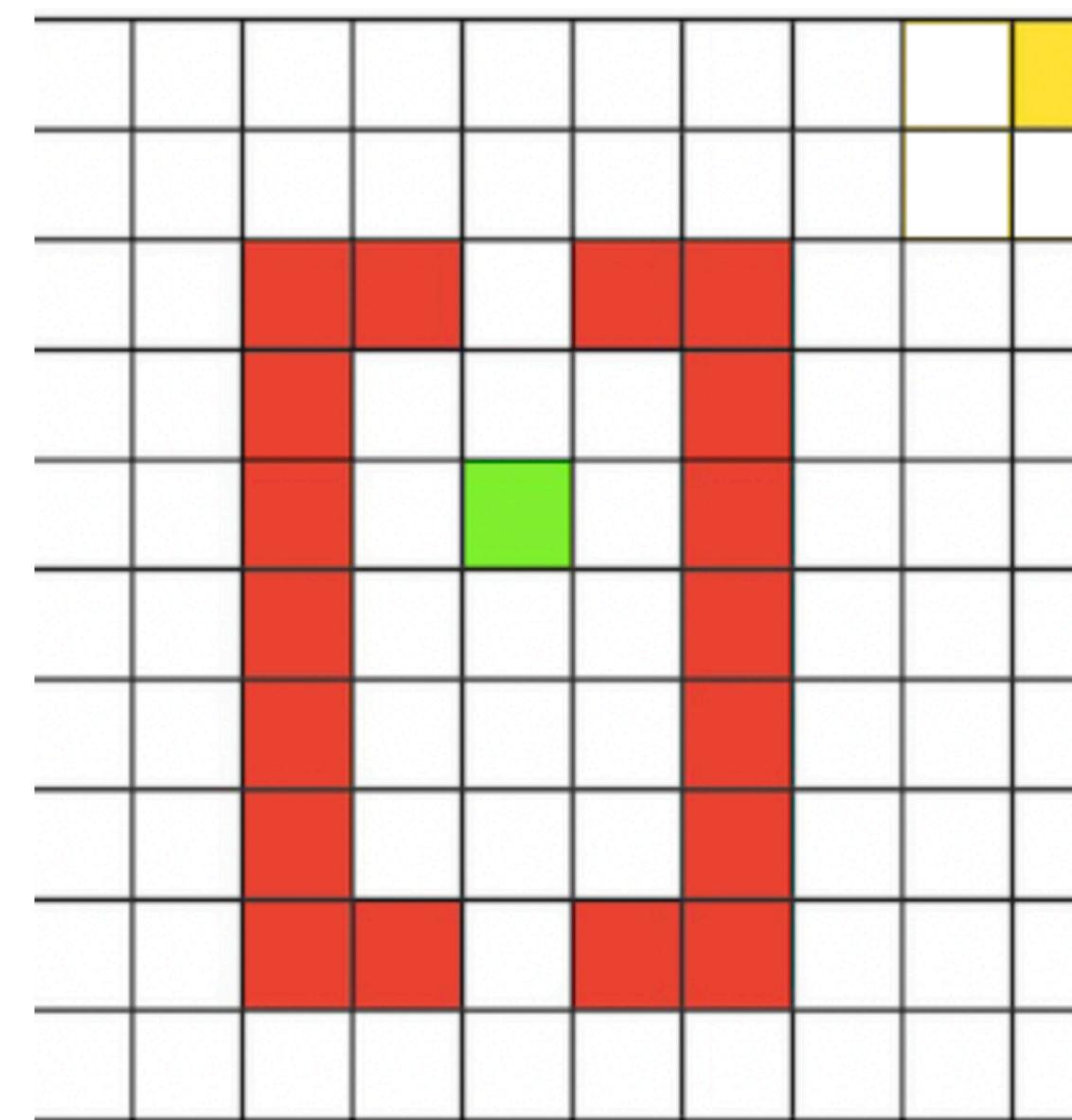
$R, S' \leftarrow Model(S, A)$

$Q(S, A) \leftarrow Q(S, A) + \alpha [R + \gamma \max_a Q(S', A) - Q(S, A)]$



DYNA-Q

This also helps with detecting shortcuts:

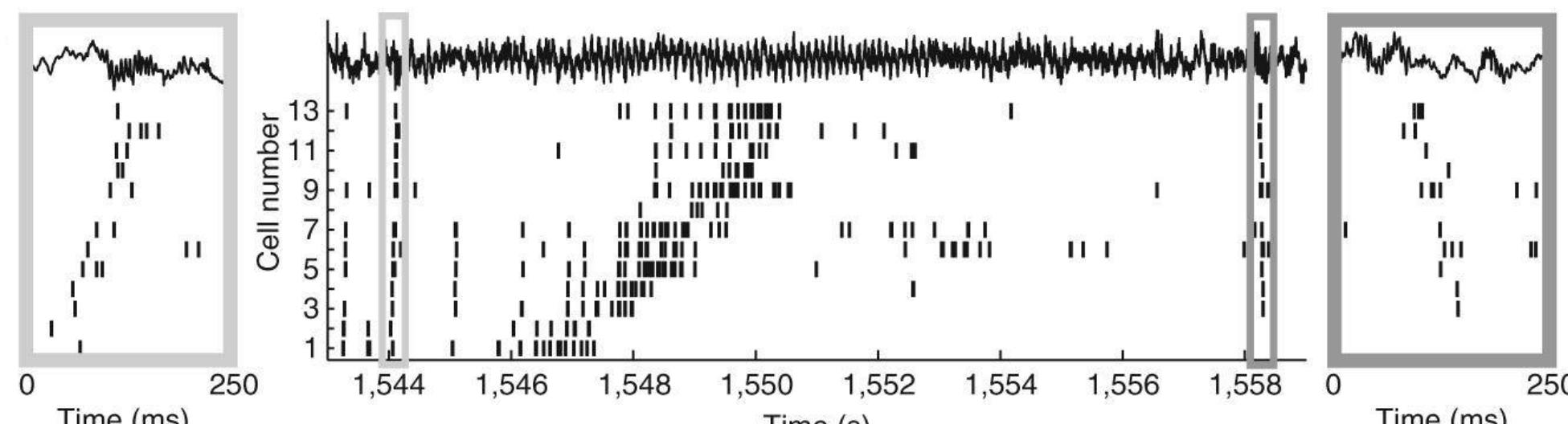


DYNA-Q - Replay as a candidate neural mechanism

DYNA-Q looks a lot like replay.

Replay as a computational mechanism in PFC and hippocampal formation

- i.e. fast reactivation of external states



Diba & Buzsaki (2007) Nature Neuroscience

Implicated in

- Learning from the *past* (credit assignment, Ambrose et al. (2016) Neuron)
- Planning *future* trajectories (Pfeiffer & Foster (2013) Nature)

What is the model in model-based RL?

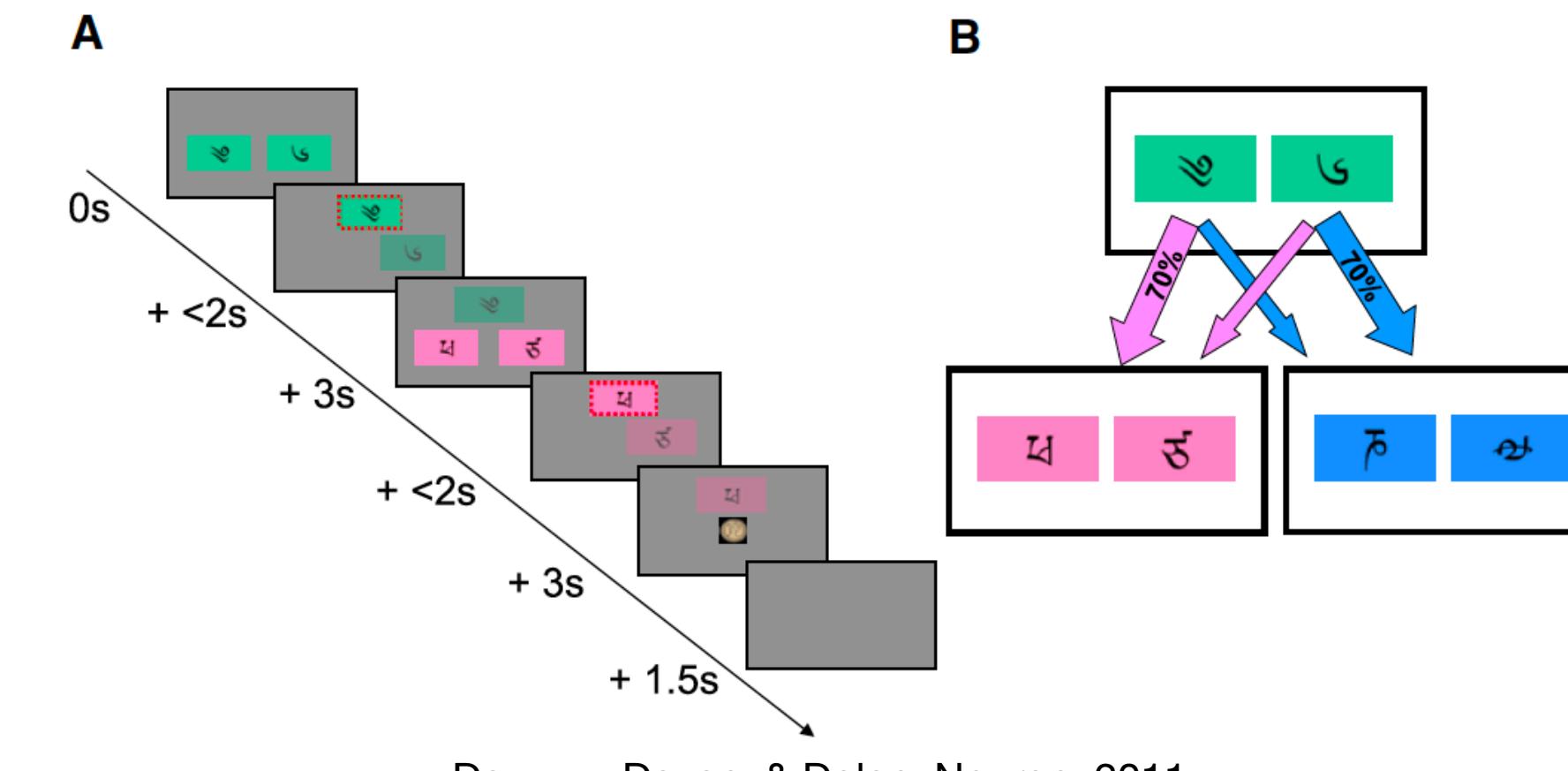
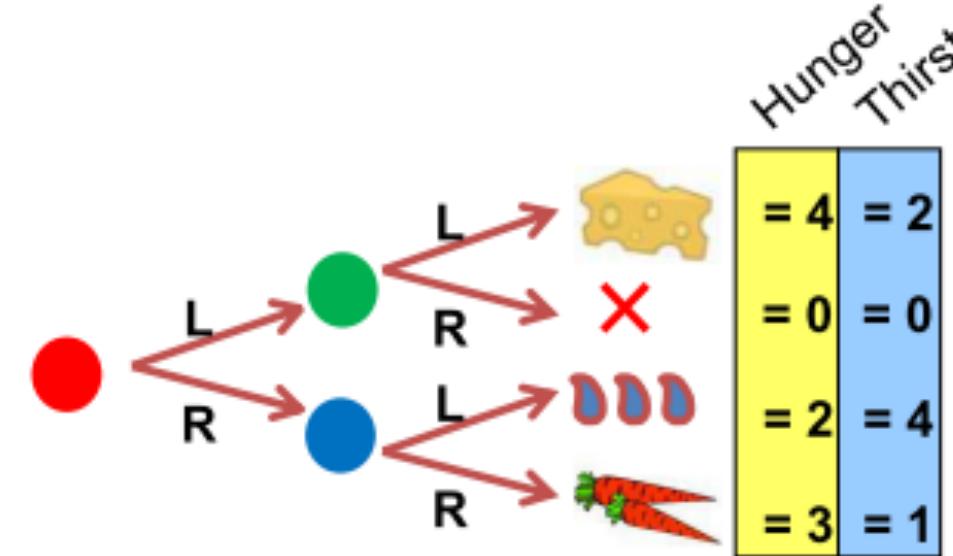
$$P(s', r | s, a) = P(s_{t+1} = s', r_{t+1} = r | s_t = s, a_t = a)$$

How can we make use of such models of the world?

Learning

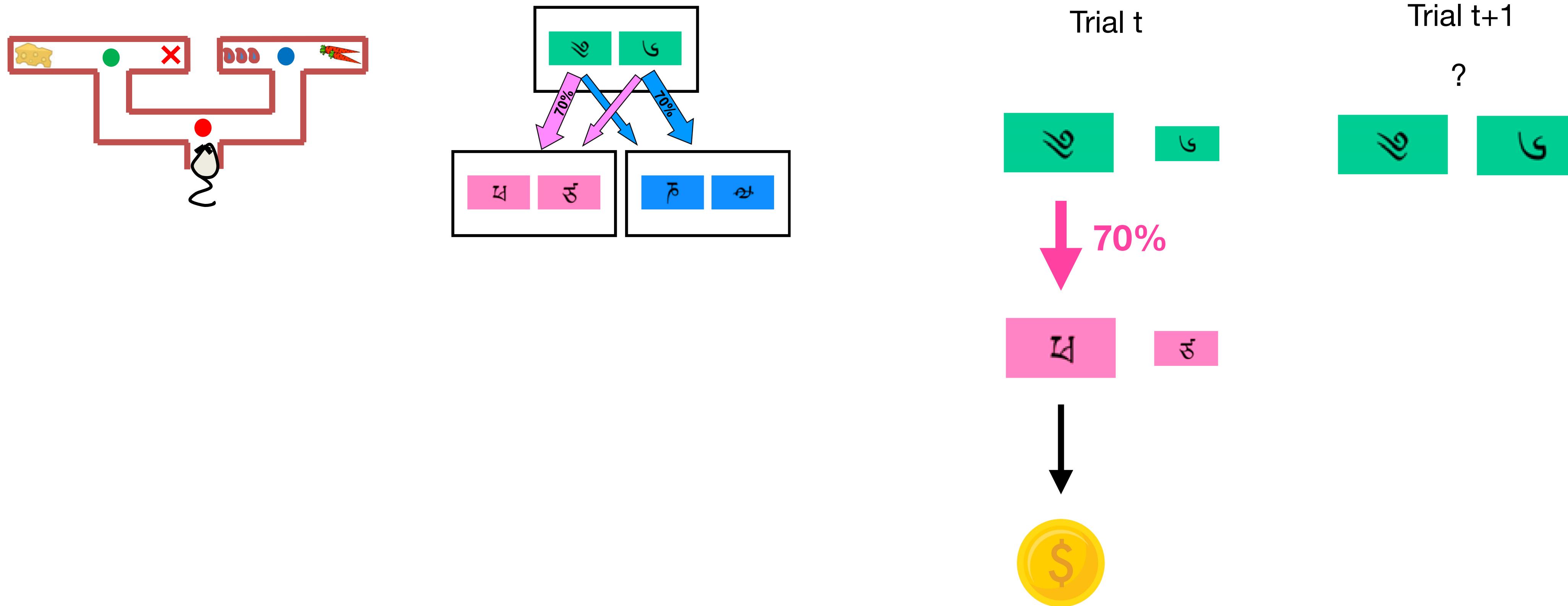
- Key idea: sample from $P(s', r | s, a)$

Planning and action selection



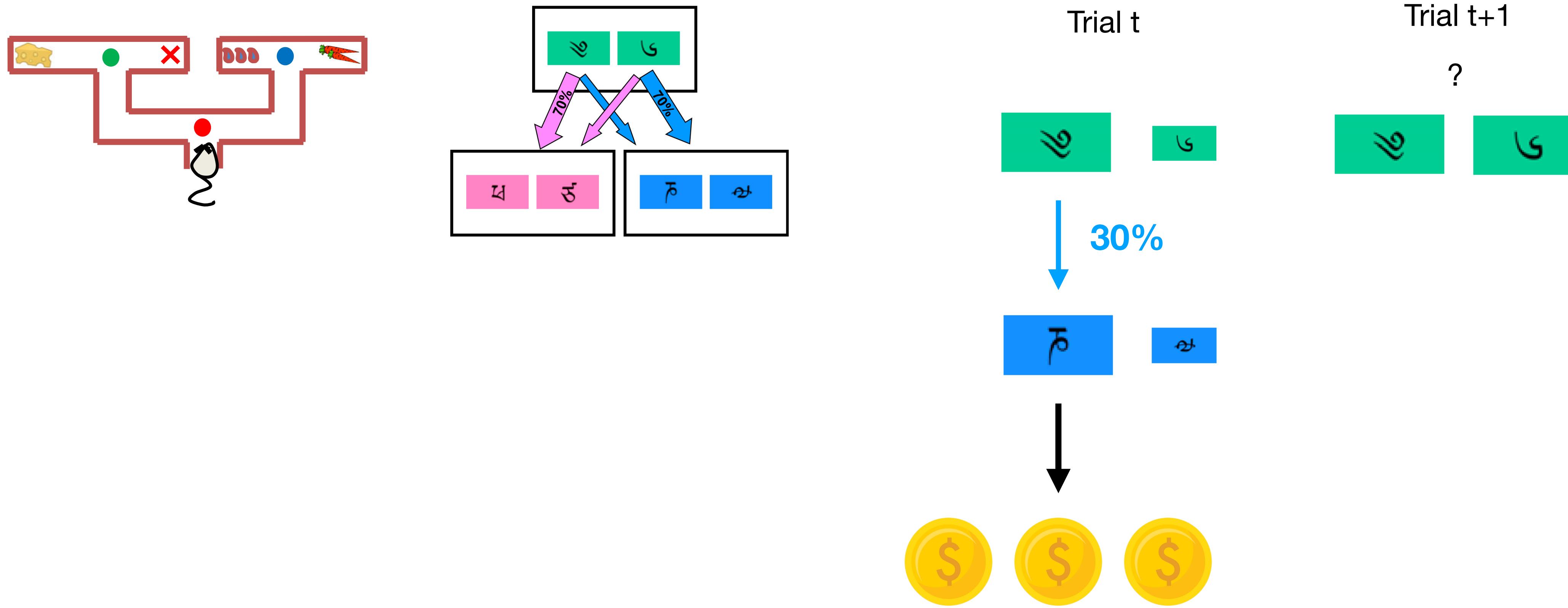
‘Two-step task’

Two-step task: one of the most iconic RL tasks



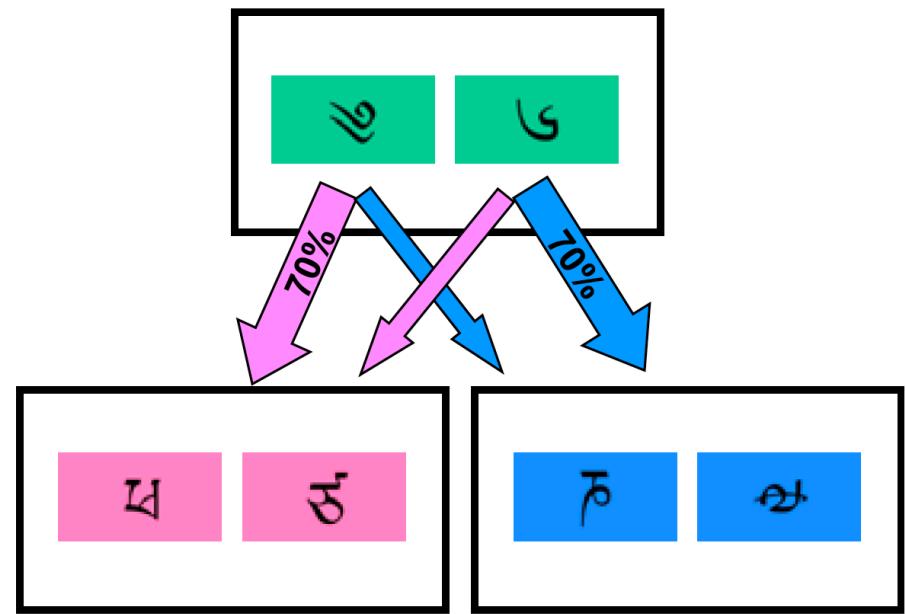
Which green option should the agent choose again at trial t+1?

Two-step task: one of the most iconic RL tasks



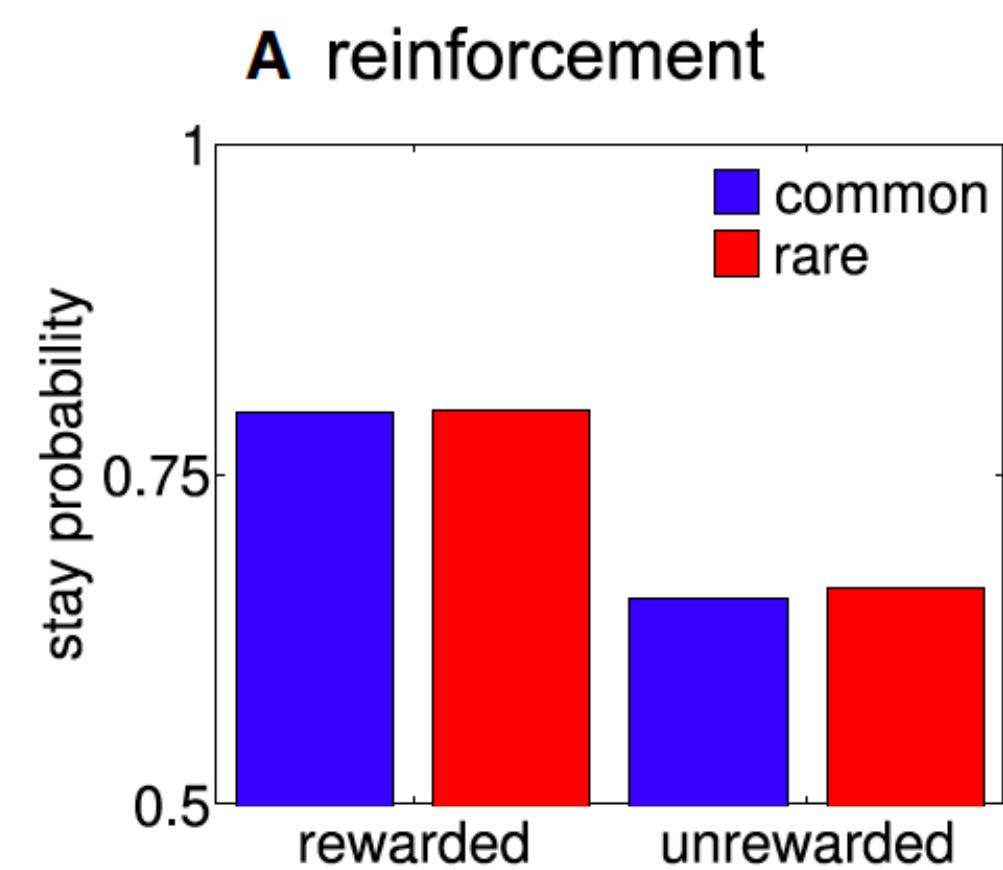
Which green option should the agent choose again at trial t+1?

Two-step task: one of the most iconic RL tasks

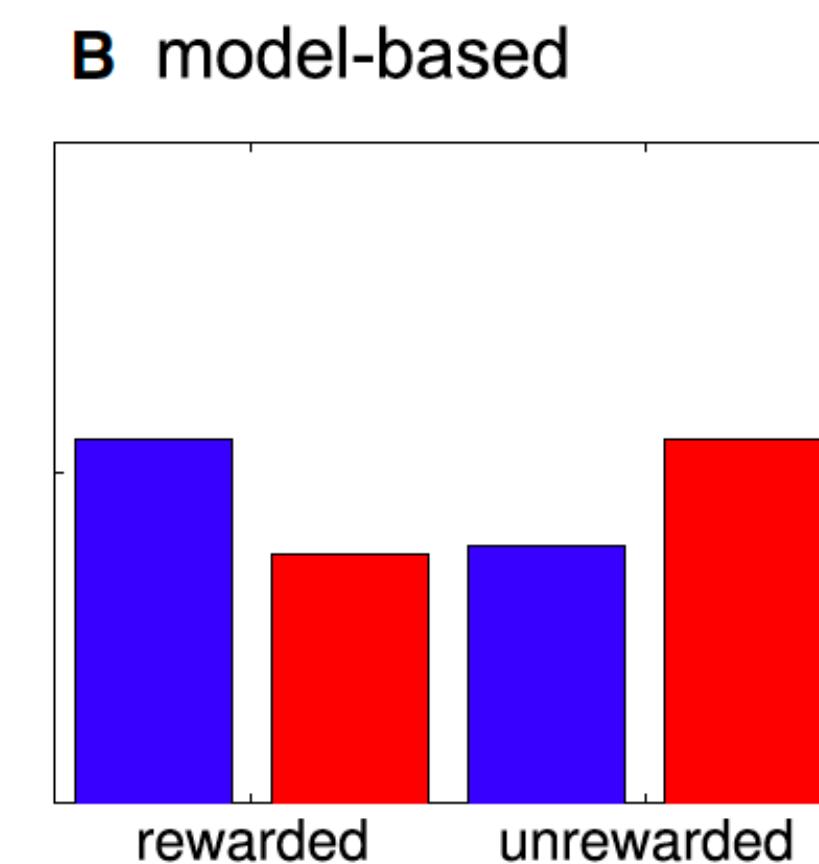


Trial t+1

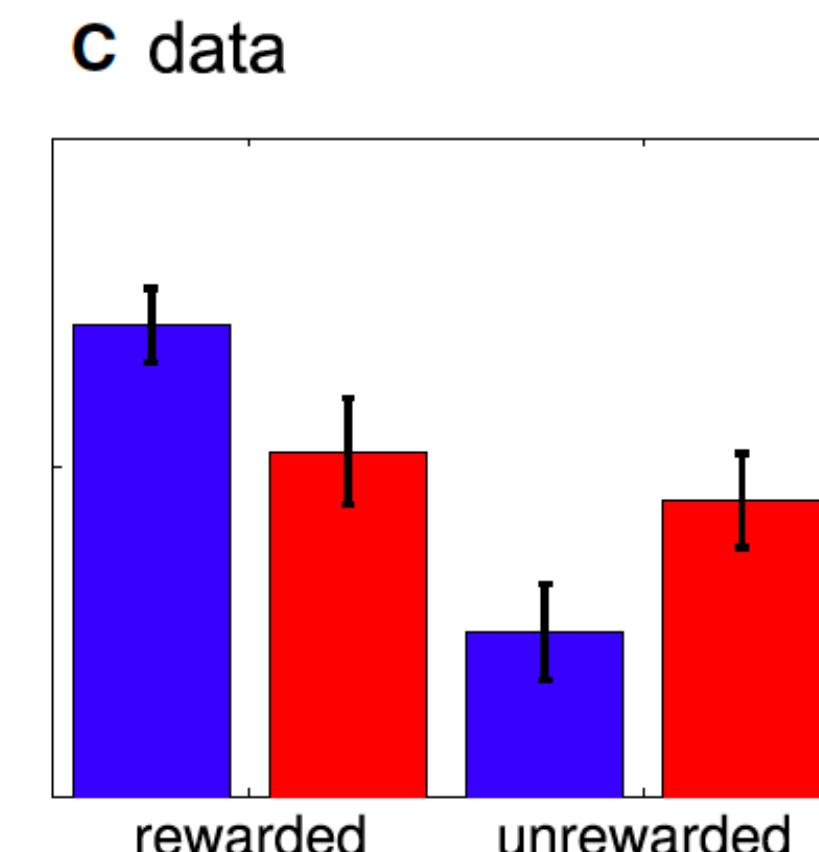
?



Model-free RL agent: repeat what is rewarding



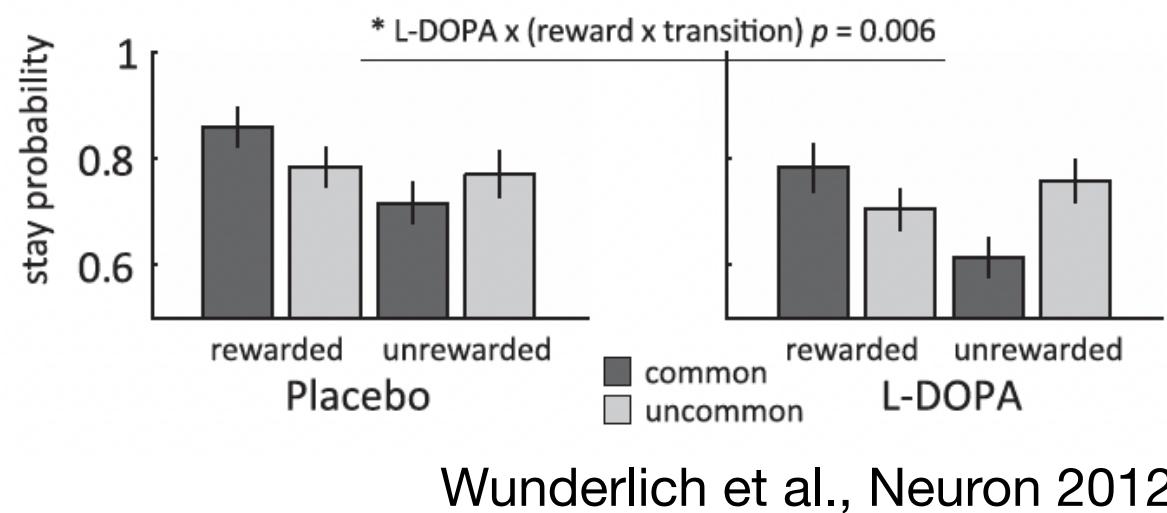
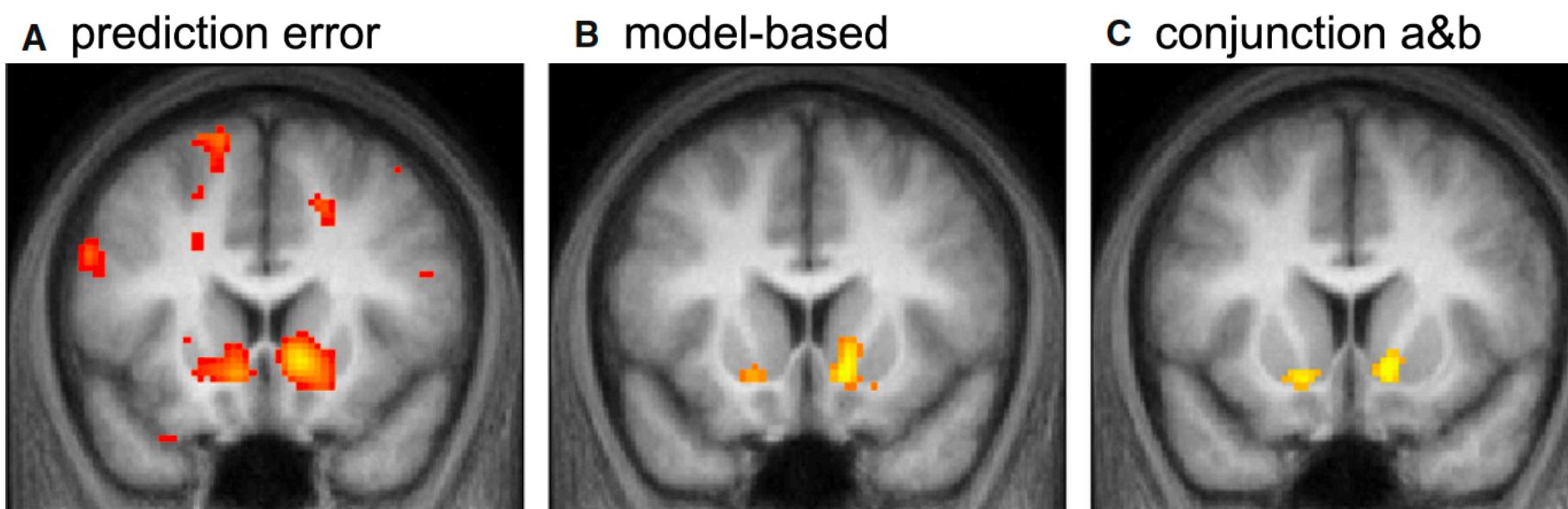
Model-based RL agent: repeat what is rewarding, but be clever



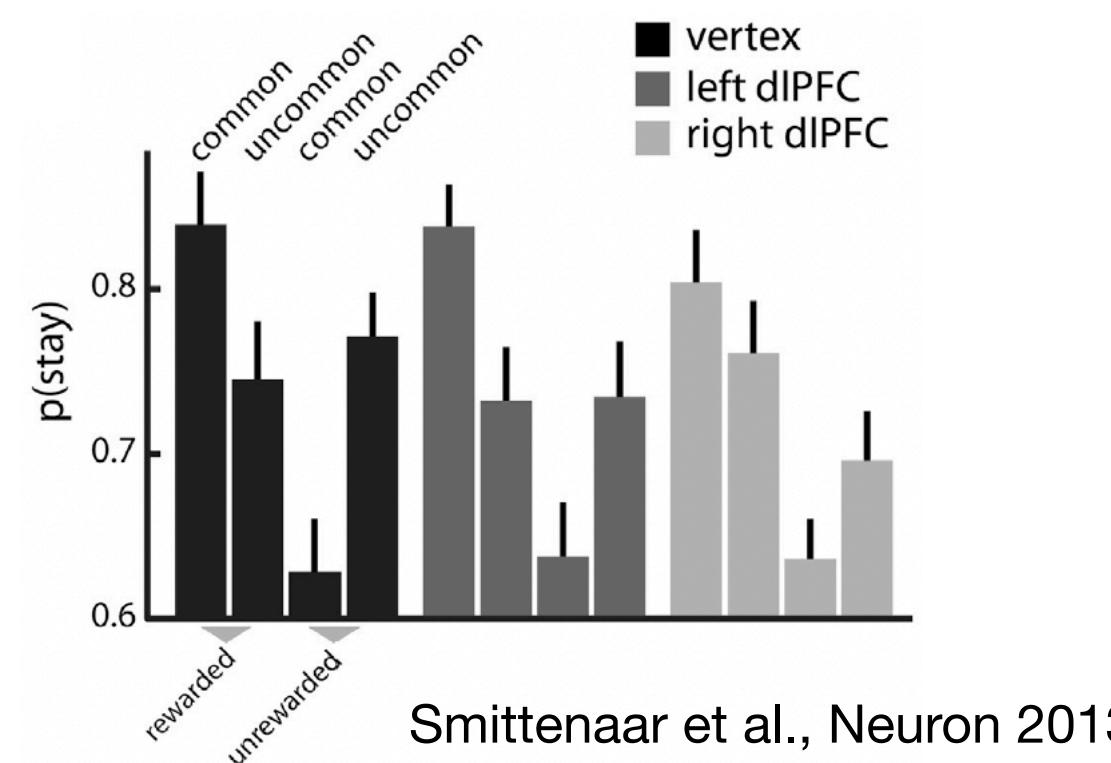
Real data: a mix of both

Two-step task: one of the most iconic RL tasks

Model-free and model-based prediction errors in ventral striatum



Increasing **dopamine**-levels makes participants more model-based



Disrupting **dIPFC** leads to more model-free behaviour

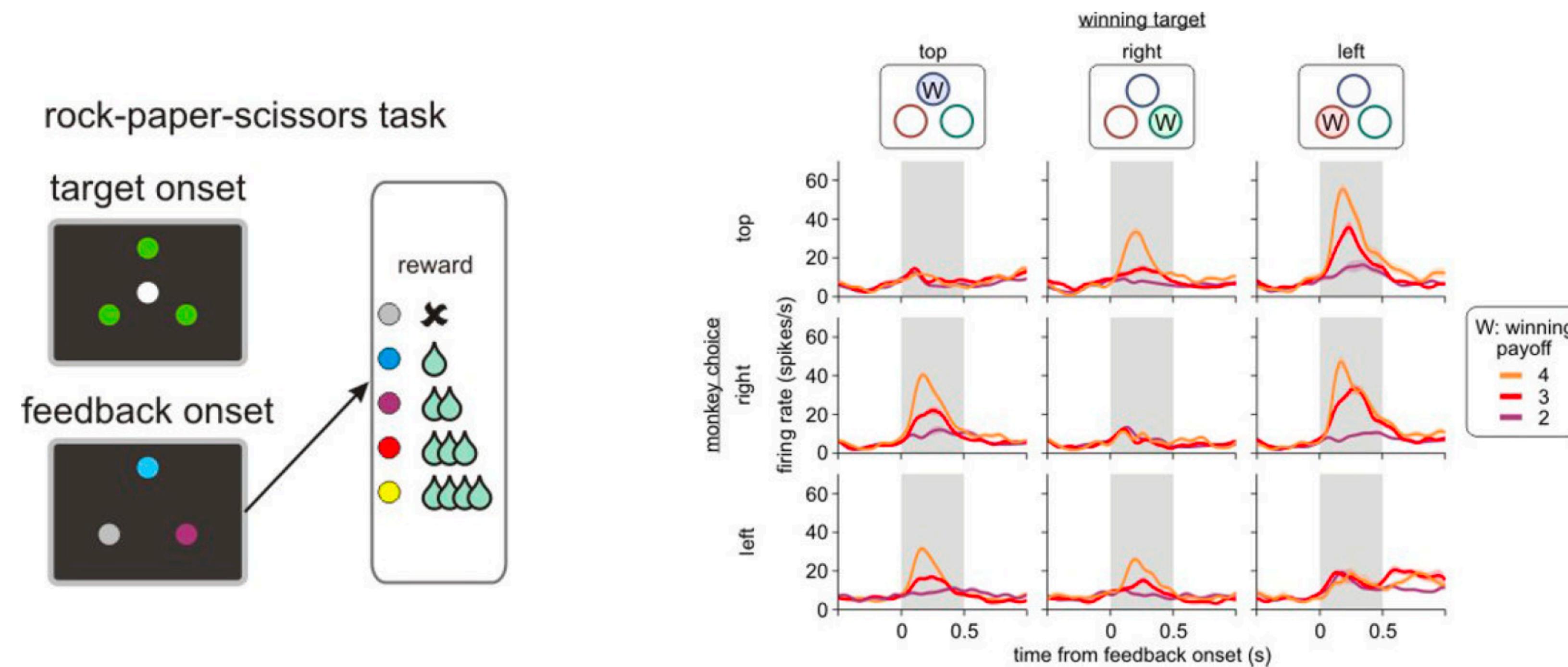
Model-based control...

- Depends on **cognitive control** (Otto et al., Journ Cogn Neuroscience 2014)
- Depends on the **dorsal hippocampus** (Miller et al., Nat Neurosci 2017)
- May sometimes be **model-free control in disguise** (Akam et al., PLOS CB 2015)
- May reflect how well people **understand the task** (Silva & Hare, NHB 2020)

Model-based reasoning: counterfactuals

Some neurons in orbitofrontal cortex encode hypothetical outcomes:

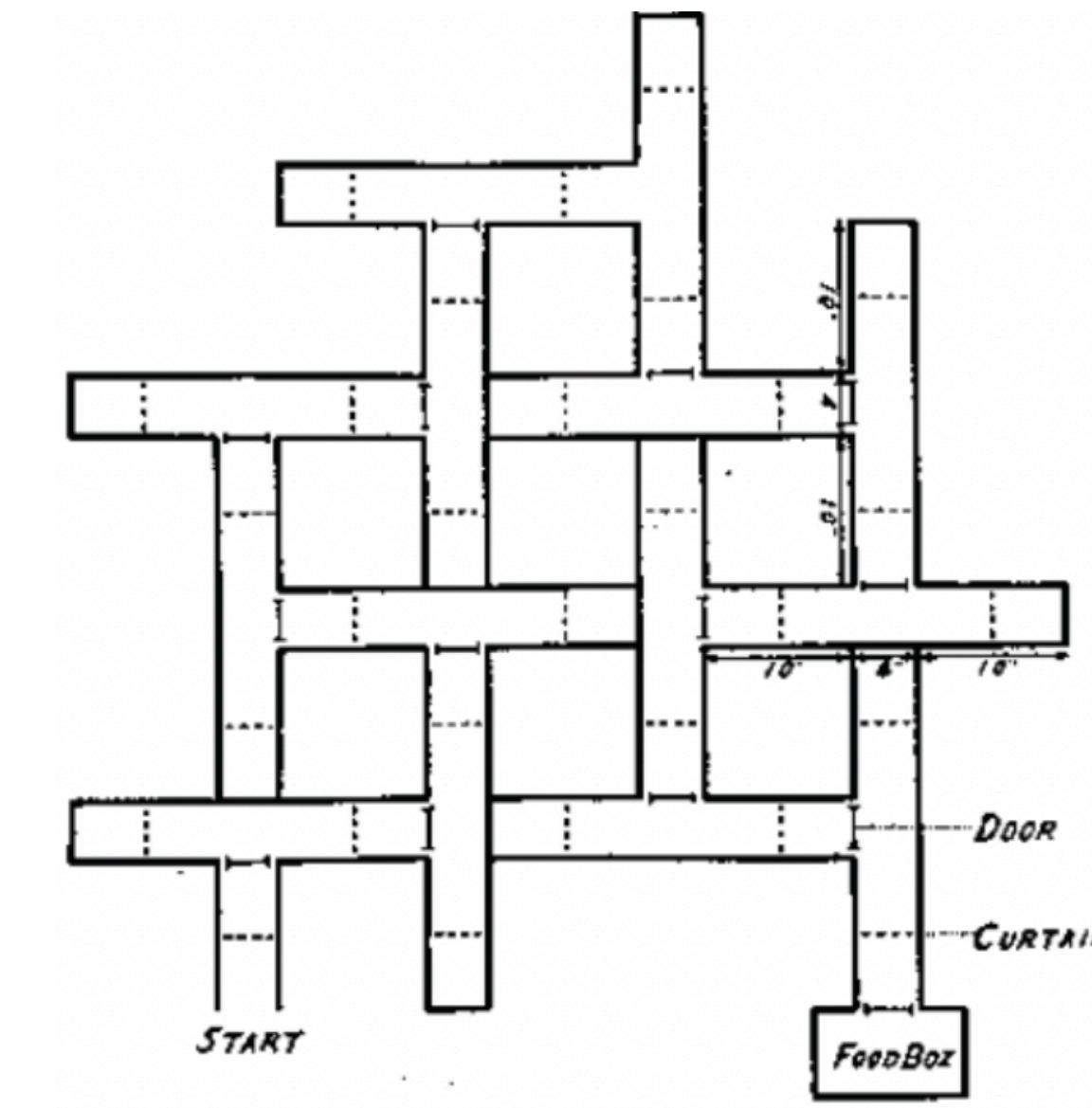
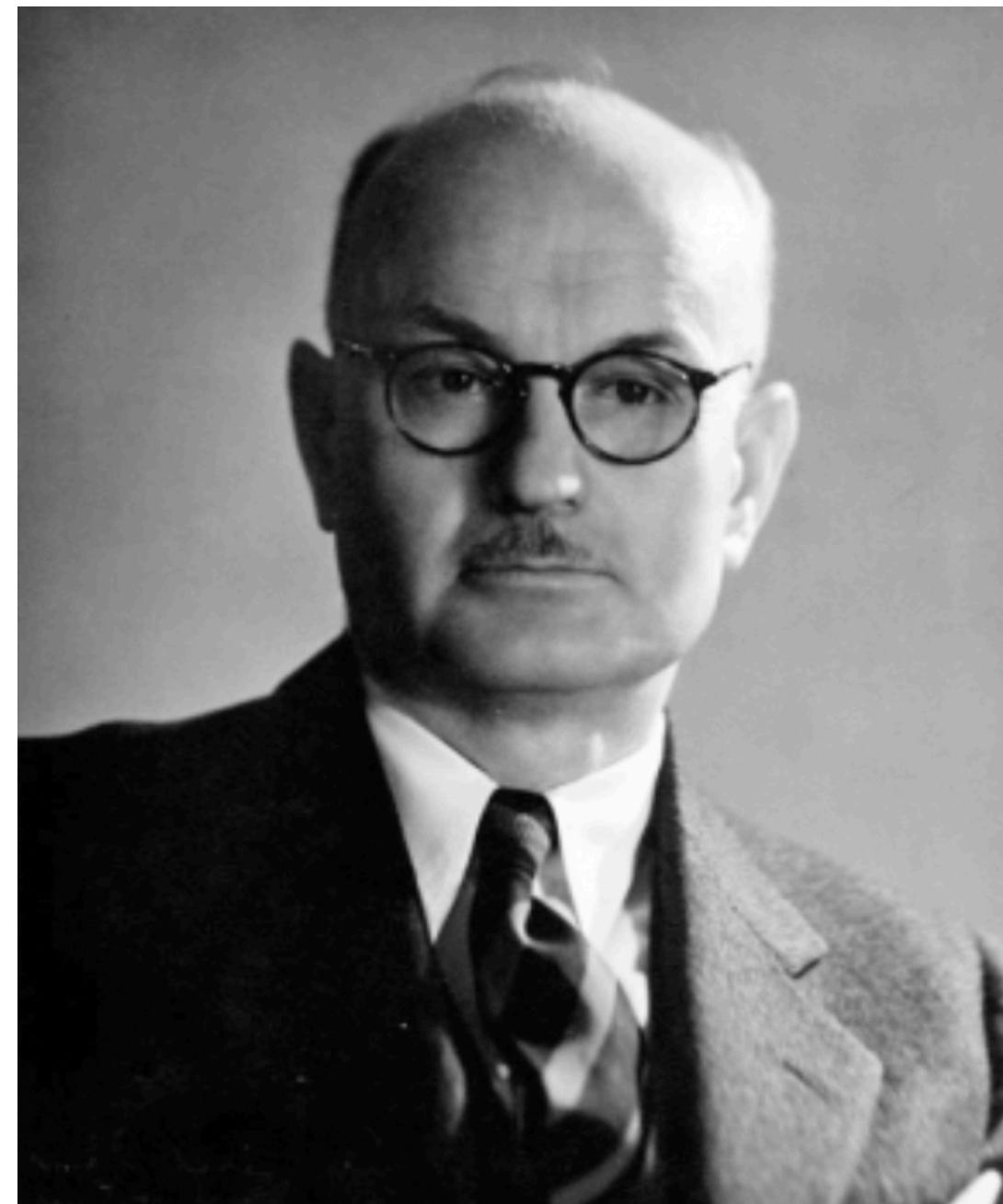
- Fire only if an *unchosen* option was rewarded



Abe & Lee, Neuron 2011

Lee et al., Annu Rev Neurosci. 2012

Cognitive maps for model-based RL?



Plan of maze
14-Unit T-Alley Maze

FIG. 1

(From M. H. Elliott, The effect of change of reward on the maze performance of rats. *Univ. Calif. Publ. Psychol.*, 1928, 4, p. 20.)

Tolman, Psychological Review, 1948

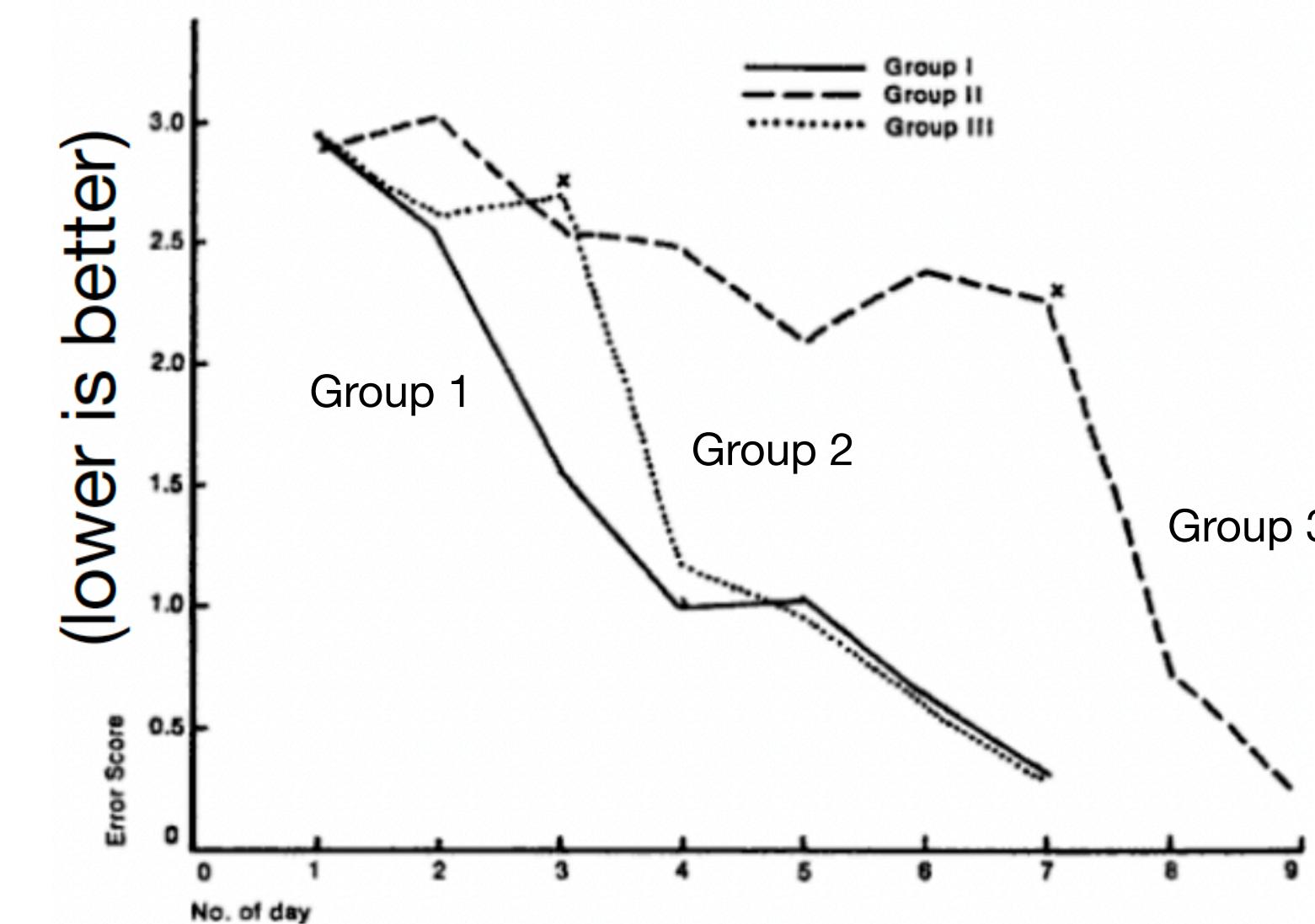
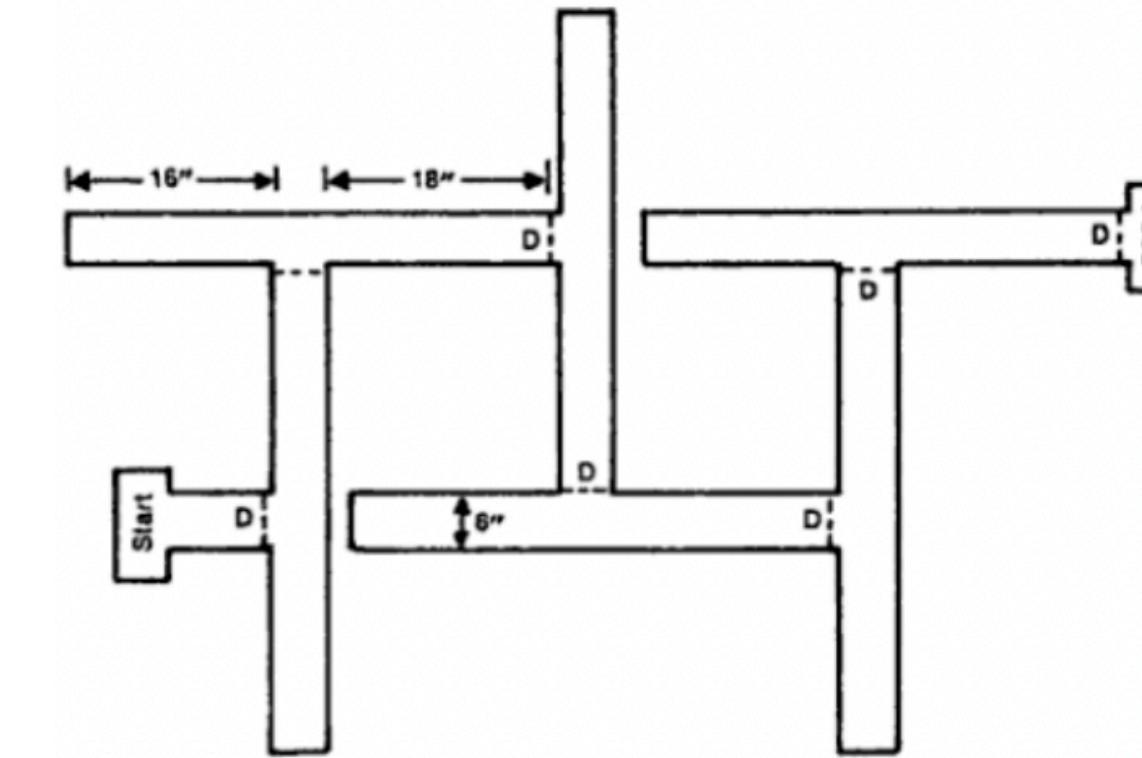
Cognitive maps for model-based RL?

Latent learning in RL: three groups of animals

- Group 1: always rewarded
- Group 2: reward added early
- Group 2: reward added late

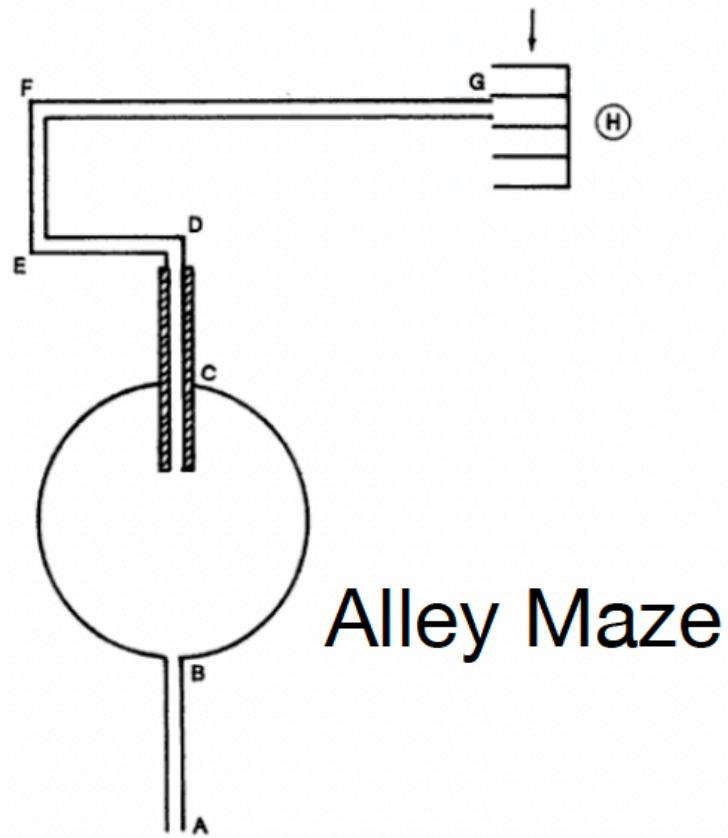
Key insight: whenever reward is added performance ‘catches up’ rapidly

- As if there is latent structural learning that can be used



Blodget, Univ. Calif. Publ. Psych., 1929

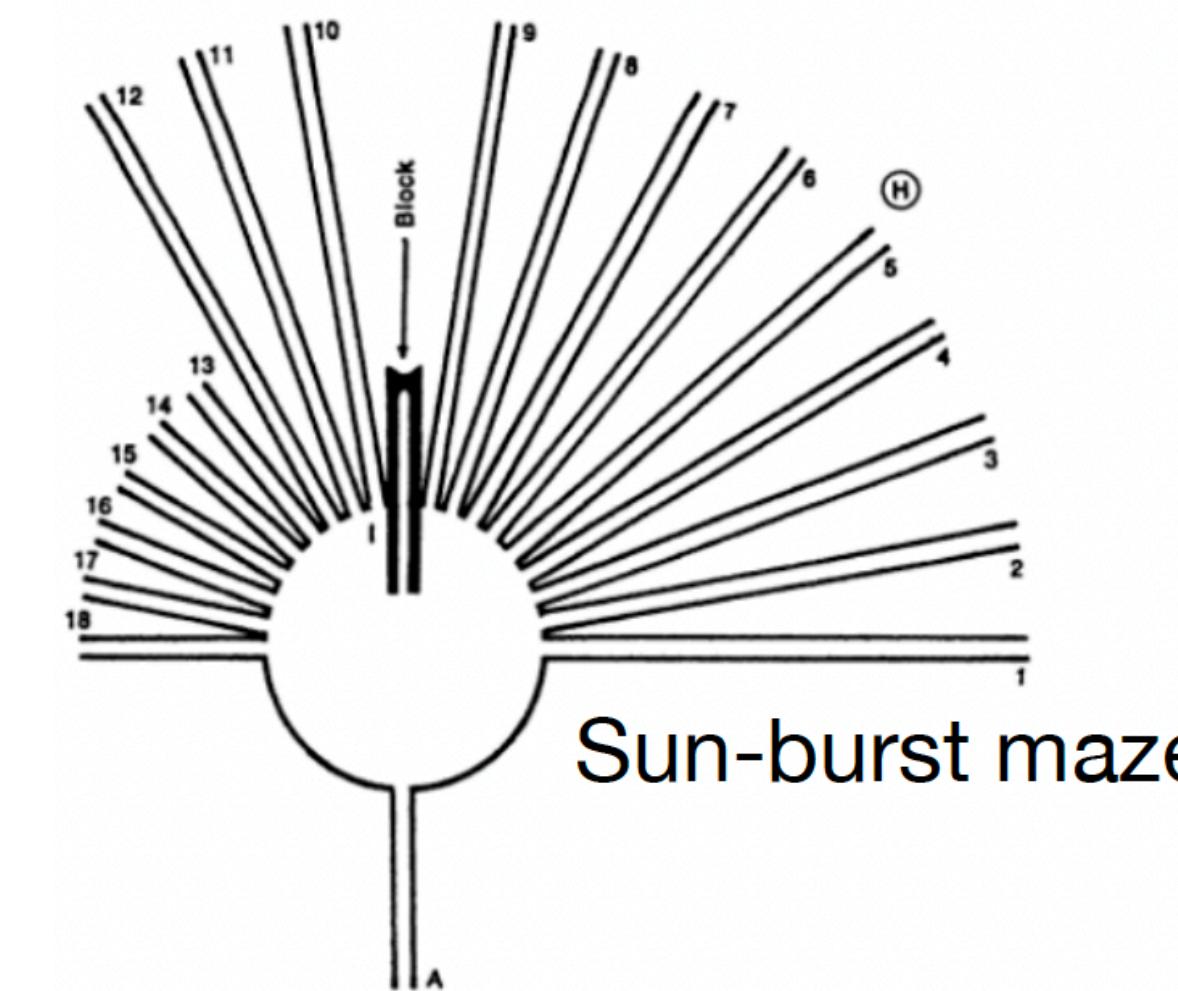
Cognitive maps for model-based RL?



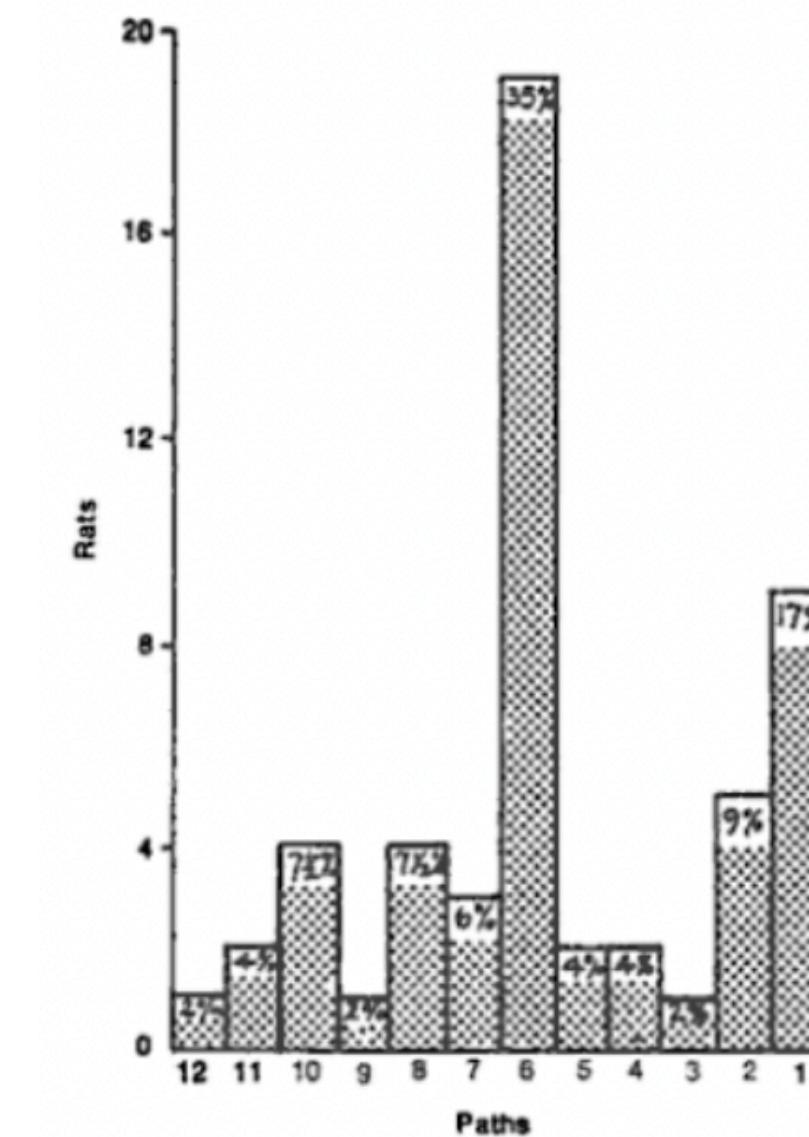
Rats are presented with a particular path to a goal G

When finding the original path blocked, rats choose the direct path to the goal in a 'sun-burst maze'

Key insight: rats have a map of 2D space, and use it for model-based RL



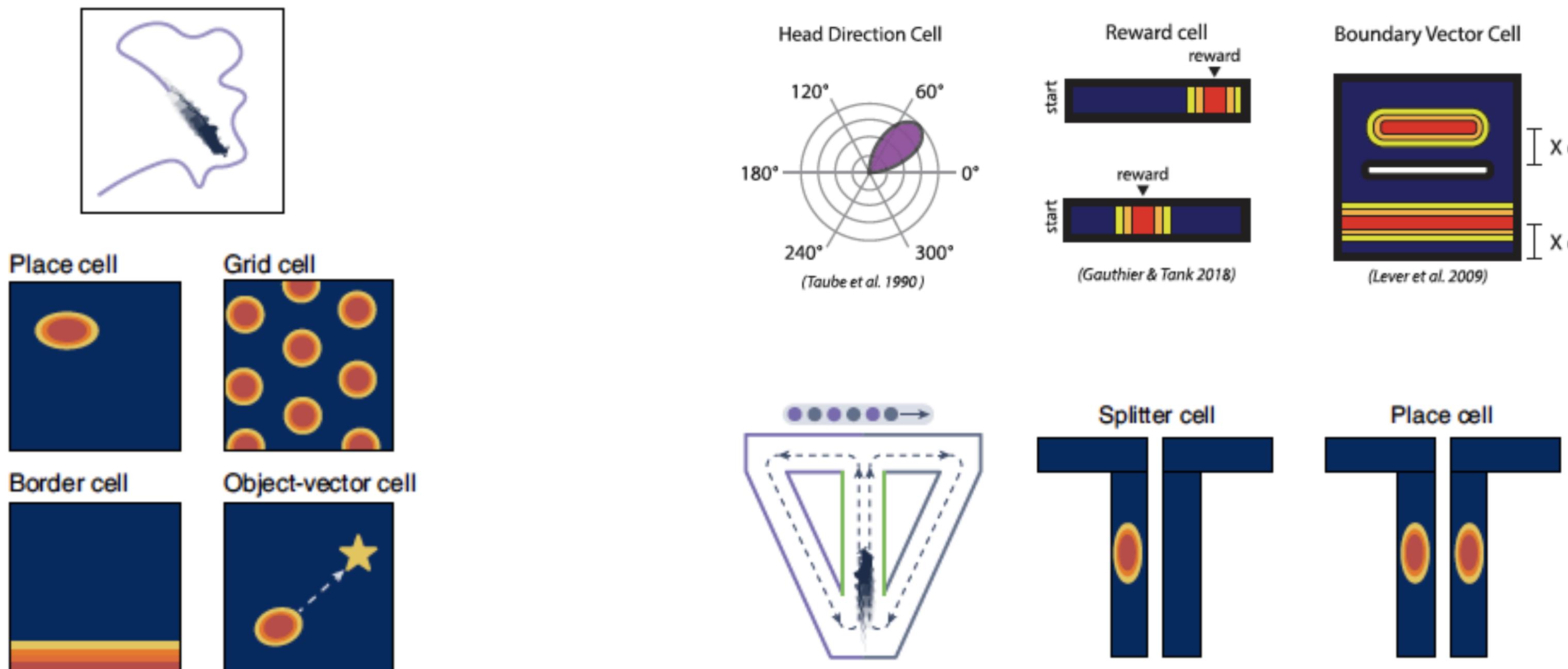
Sun-burst maze



Tolman et al., Journ. Exp. Psych., 1946

Cognitive maps for model-based RL?

We have gained much insight into cognitive maps used in goal-directed navigation



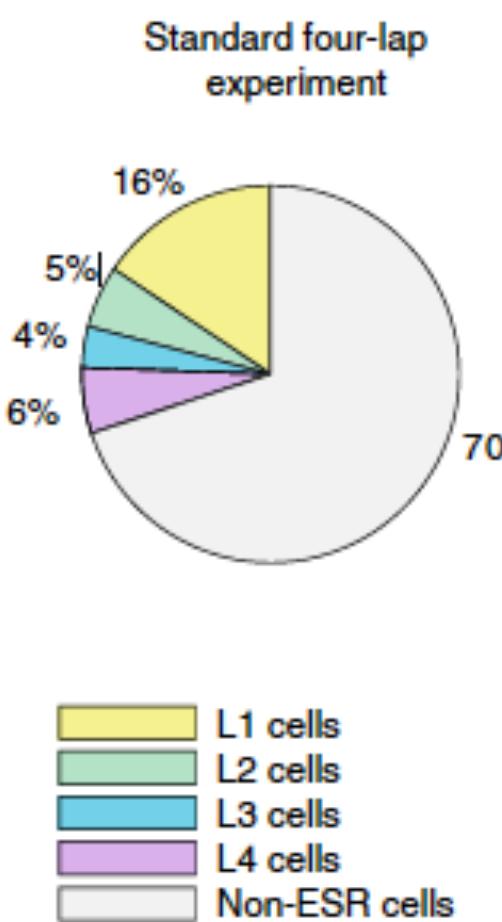
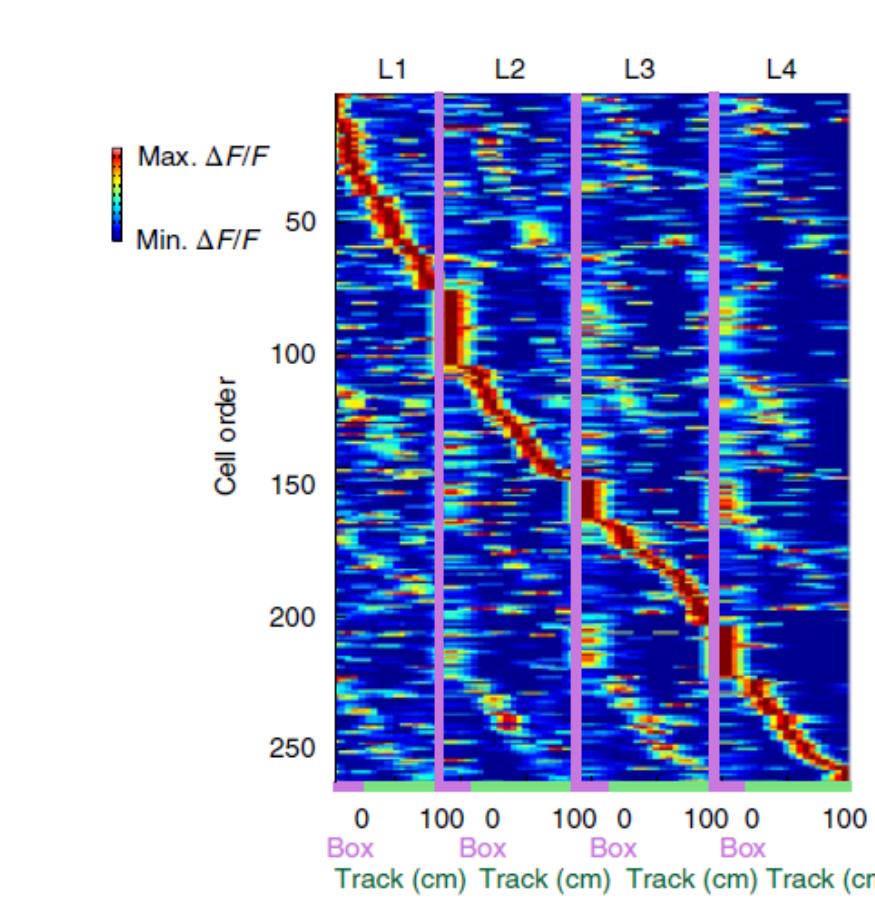
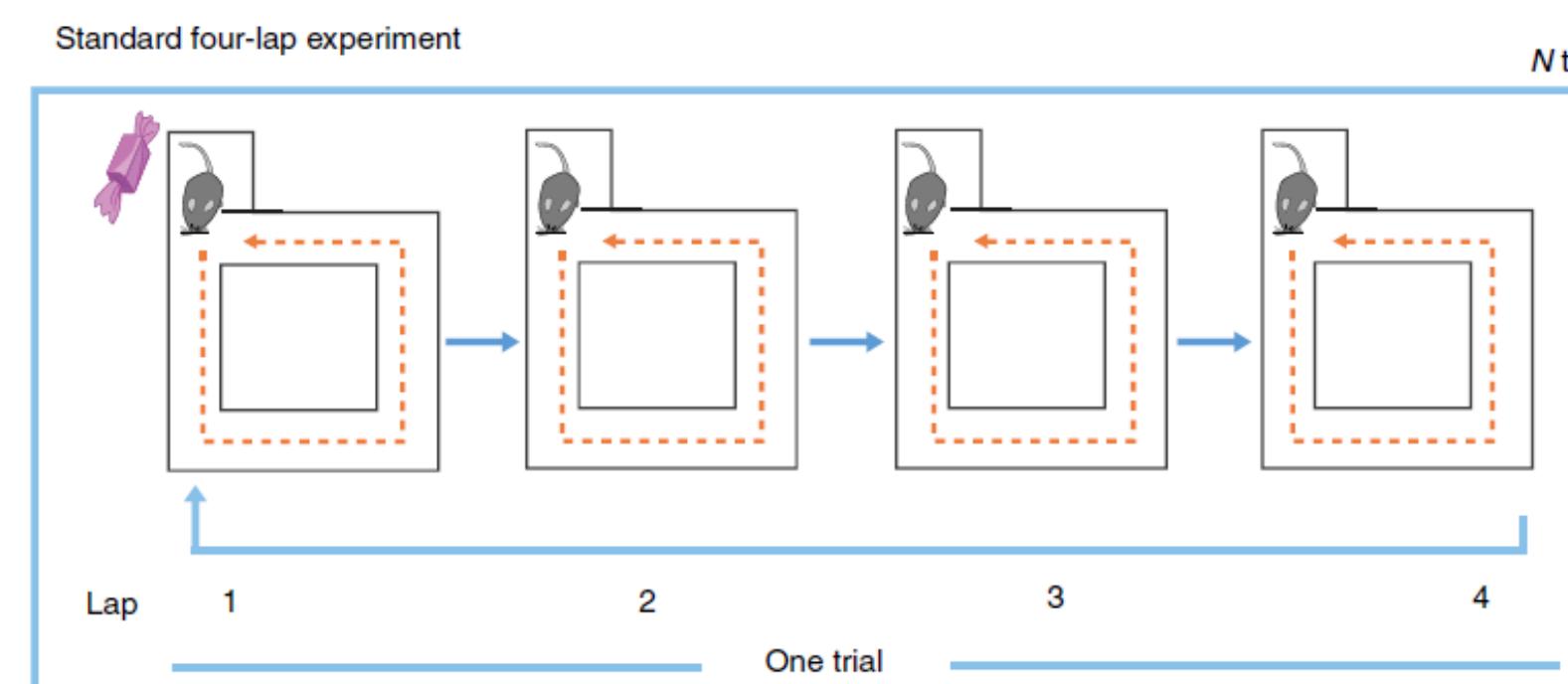
Is this a **basis set** over world structures?

Whittington et al. (2022). How to build a cognitive map. Nature Neuroscience

Behrens et al. (2018). What is a cognitive map? Organizing knowledge for flexible behavior. Neuron

A cognitive map for model-based RL - beyond space

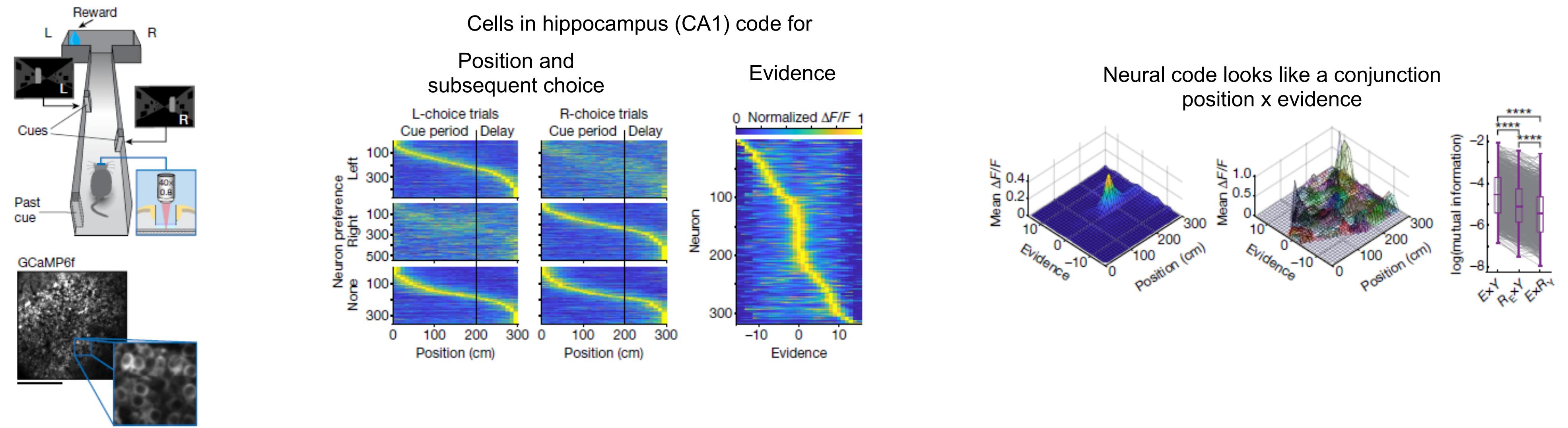
Place cells for lap counting - relevant for reward:



Important: cognitive maps might encode **task space**, rather than just physical space!

A cognitive map for model-based RL - beyond space

Place cells for evidence accumulation:



Important: cognitive maps might encode **task space**, rather than just physical space!

Summary

Model-free association learning powerful framework

- Remarkable success in explaining behaviour and neural phenomena
- Resource-efficient but inflexible

Model-based RL: high flexibility but also costly

- Gold-standard probe: **de-valution**
- Helps with **learning and planning/action selection**
- **Cognitive maps** as neural candidate for task model representation

Homework

Choose one of the following:

1. Reproduce a basic **DYNA-Q agent** in a maze task (with or without a shortcut)
 - [https://github.com/schwartenbeckph/Systems-Computational-Neuroscience/blob/main/
DYNA_Q_SysCompNeuro.ipynb](https://github.com/schwartenbeckph/Systems-Computational-Neuroscience/blob/main/DYNA_Q_SysCompNeuro.ipynb)
2. Reproduce the **two step task** and the **stay probabilities** for a model-free, model-based and mixed agent
 - [https://github.com/schwartenbeckph/Systems-Computational-Neuroscience/blob/main/
TwoStep_SysCompNeuro.ipynb](https://github.com/schwartenbeckph/Systems-Computational-Neuroscience/blob/main/TwoStep_SysCompNeuro.ipynb)

Please let me know if you are unfamiliar with Python notebooks!

Thank you