Reinforcement Learning Theory

Quentin Huys
MA MBBS PhD MBPsS

Translational Neuromodeling Unit, ETH and University of Zurich
Hospital of Psychiatry, University of Zurich

Friedrich Miescher Institute, Basel, 25/11/2013

Overview

- Reinforcement learning: rough overview
  - mainly following Sutton & Barto 1998
- Learning theory
  - classical & instrumental conditioning
- Dopamine
  - prediction errors and more

Types of learning

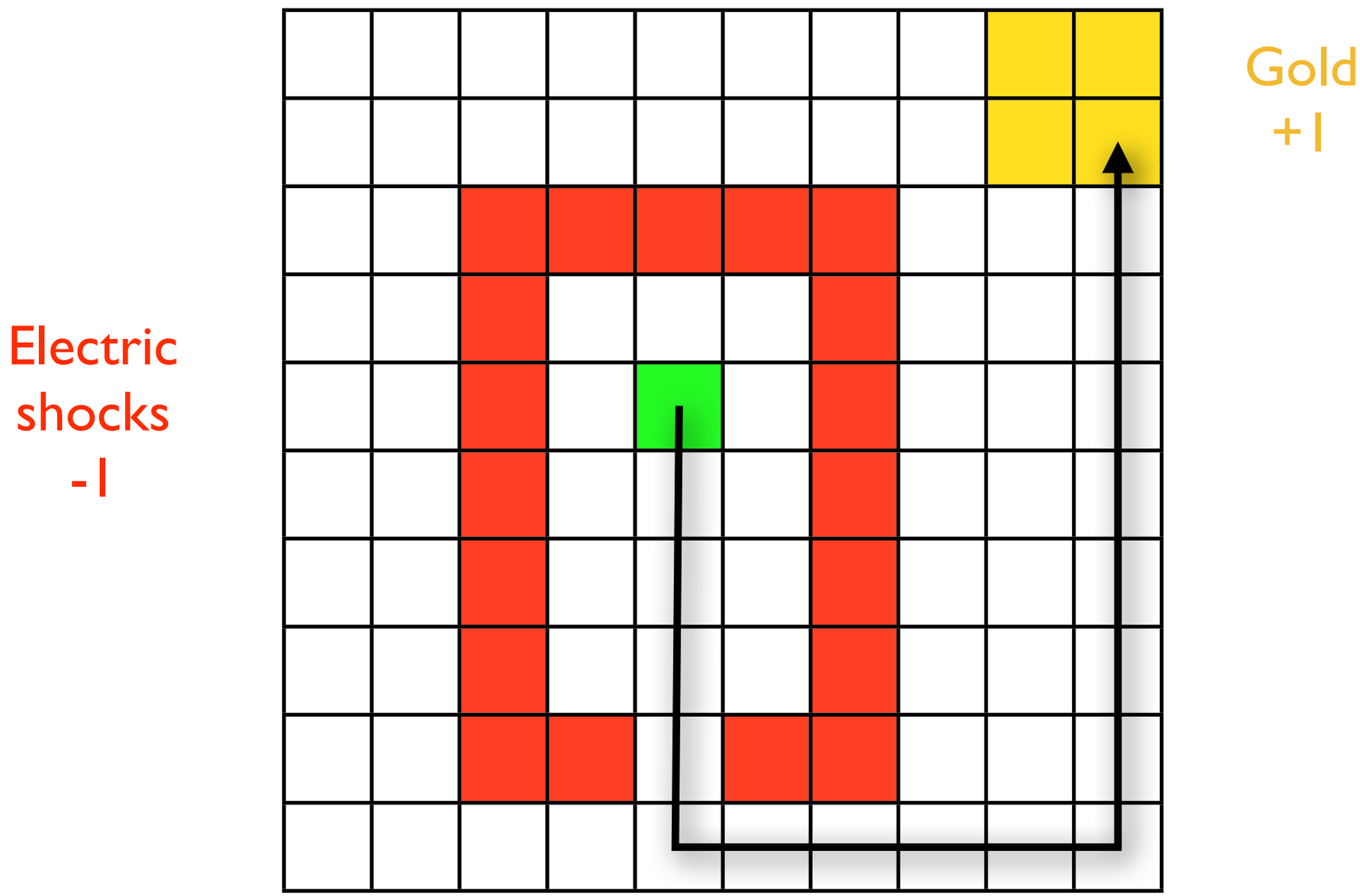
- Supervised
- Unsupervised
- Reinforcement learning

Setup

$$\{a_t\} \leftarrow \underset{\{a_t\}}{\operatorname{argmax}} \sum_{t=1}^{\infty} r_t$$

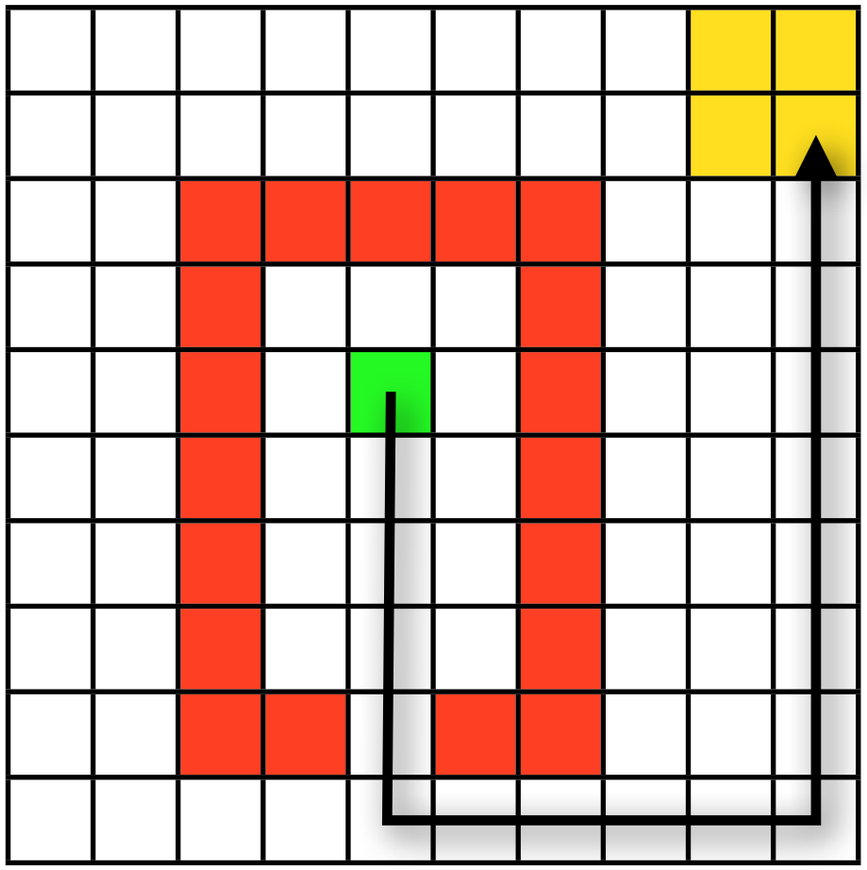
After Sutton and Barto 1998

State space



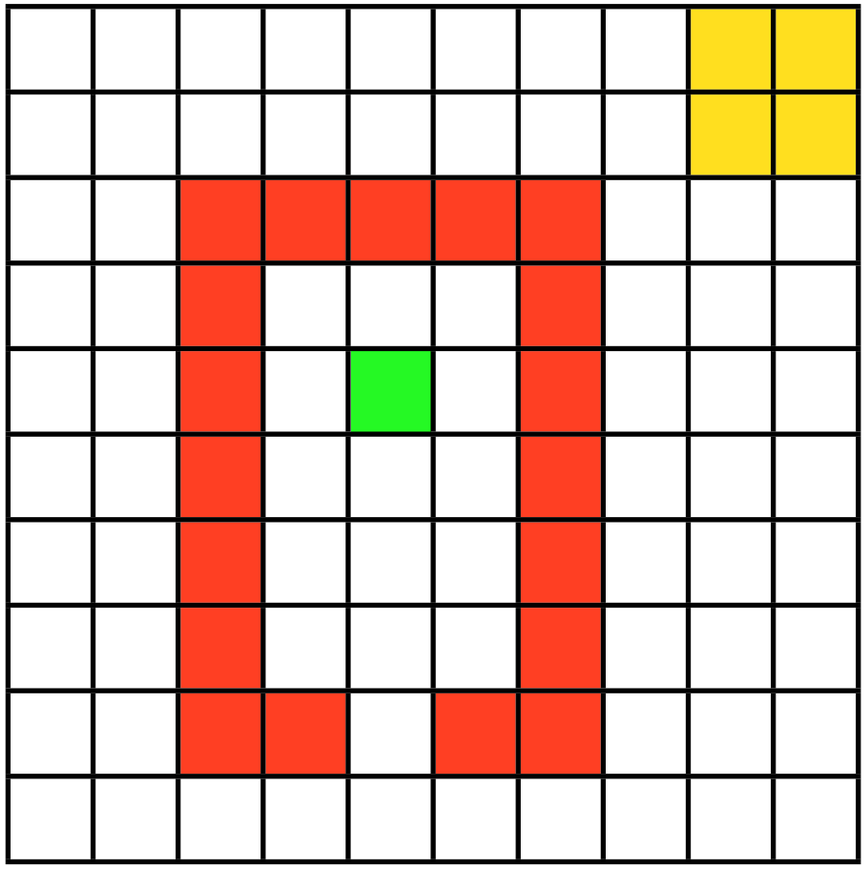
A Markov Decision Problem

$$
\begin{aligned}
s_t &\in \mathcal{S} \\
a_t &\in \mathcal{A} \\
\mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \\
r_t &\sim \mathcal{R}(s_{t+1}, a_t, s_t) \\
\pi(a|s) &= p(a|s)
\end{aligned}
$$



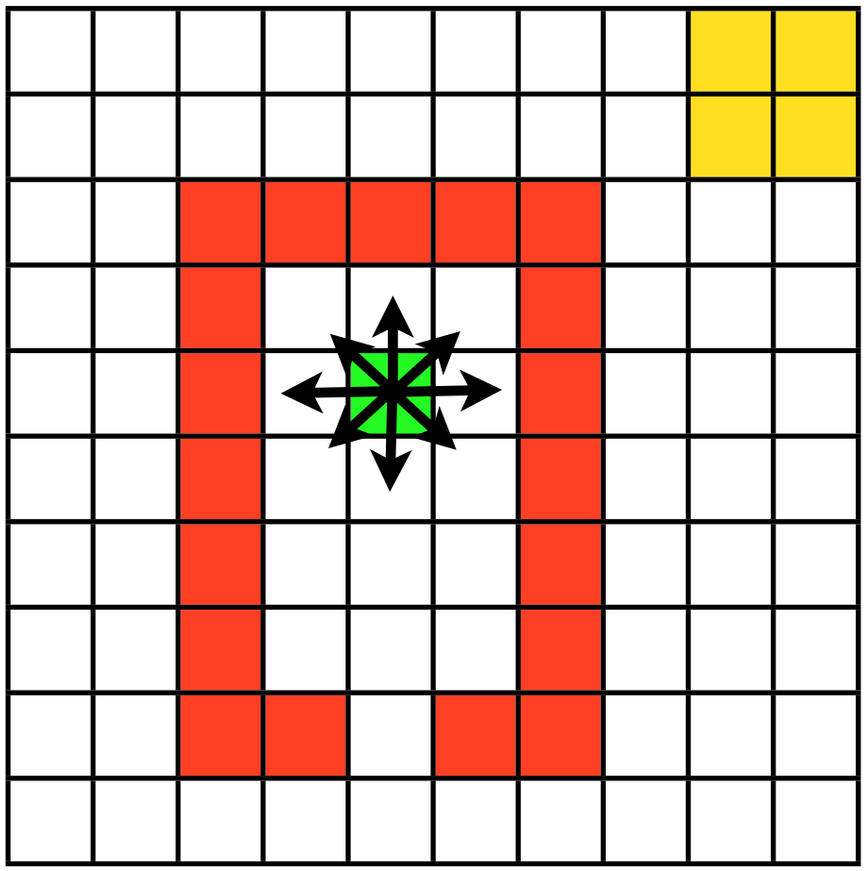
MDP

$$
\begin{aligned}
s_t &\in \mathcal{S} \\
a_t &\in \mathcal{A} \\
\mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \\
r_t &\sim \mathcal{R}(s_{t+1}, a_t, s_t) \\
\pi(a|s) &= p(a|s)
\end{aligned}
$$



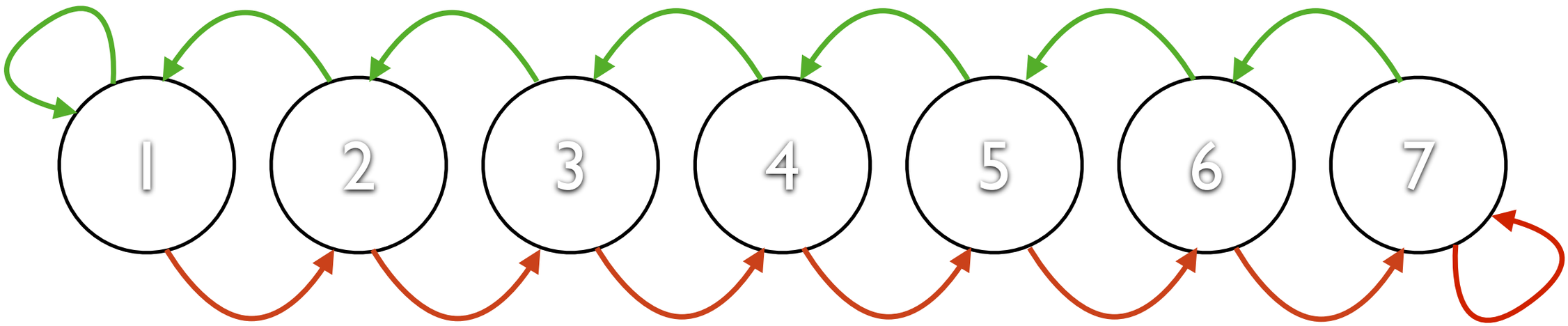
MDP

$$
\begin{aligned}
s_t &\in \mathcal{S} \\
a_t &\in \mathcal{A} \\
\mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \\
r_t &\sim \mathcal{R}(s_{t+1}, a_t, s_t) \\
\pi(a|s) &= p(a|s)
\end{aligned}
$$



Actions

Action left

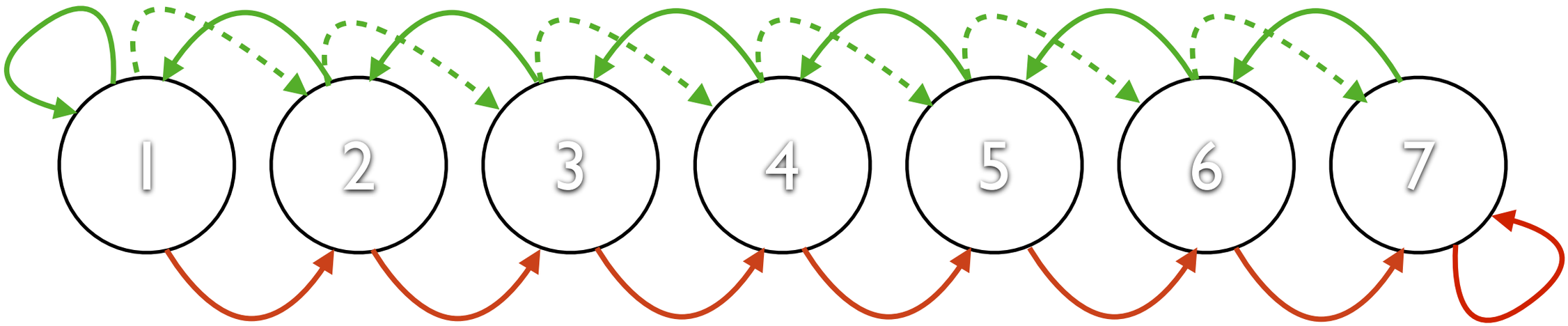


Action right

$$T^{\text{left}} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad T^{\text{right}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Actions

Action left

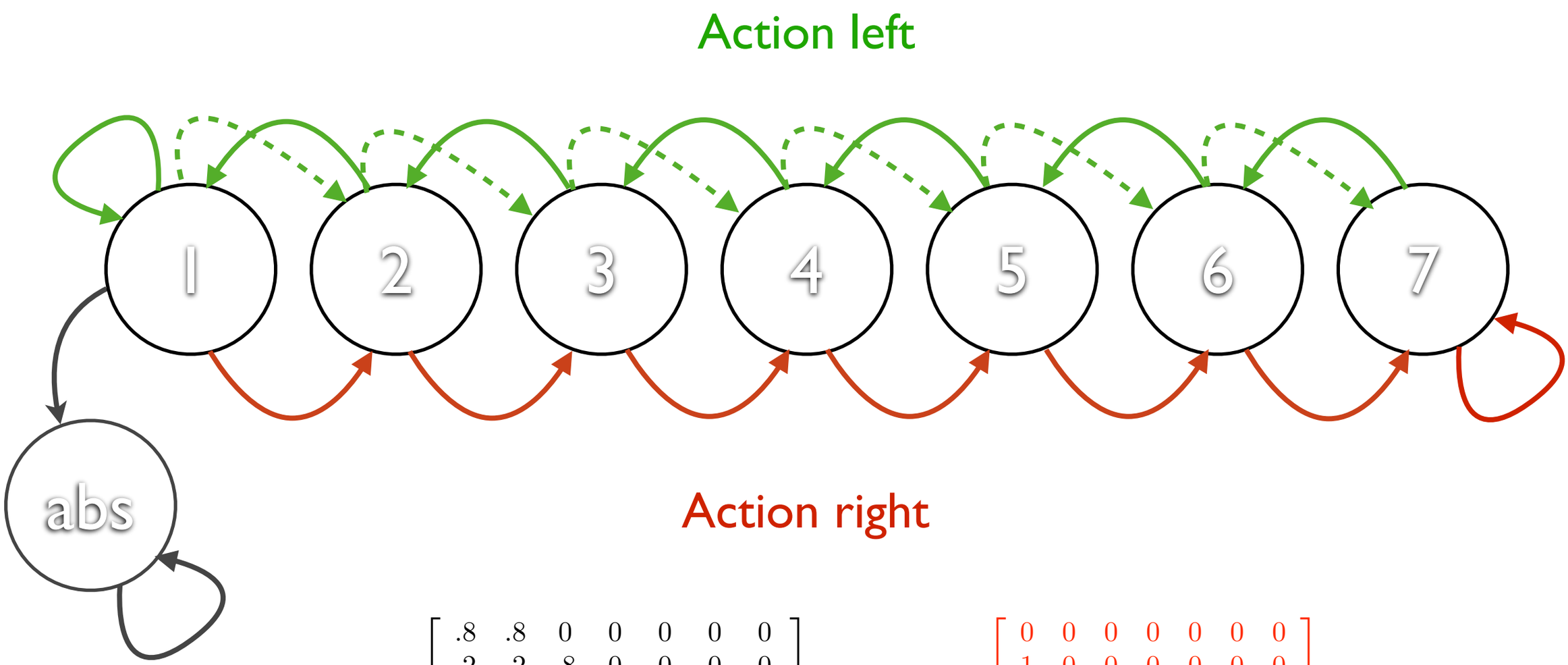


Action right

$$T^{\text{left}} = \begin{bmatrix} .8 & .8 & 0 & 0 & 0 & 0 & 0 \\ .2 & .2 & .8 & 0 & 0 & 0 & 0 \\ 0 & 0 & .2 & .8 & 0 & 0 & 0 \\ 0 & 0 & 0 & .2 & .8 & 0 & 0 \\ 0 & 0 & 0 & 0 & .2 & .8 & 0 \\ 0 & 0 & 0 & 0 & 0 & .2 & .8 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad T^{\text{right}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Noisy: plants, environments, agent

Actions



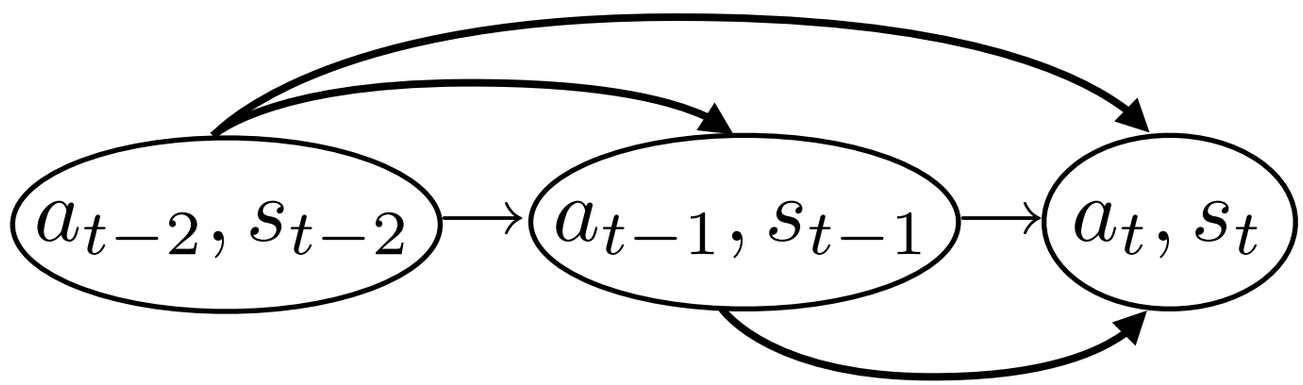
$$T^{\text{left}} = \begin{bmatrix} .8 & .8 & 0 & 0 & 0 & 0 & 0 \\ .2 & .2 & .8 & 0 & 0 & 0 & 0 \\ 0 & 0 & .2 & .8 & 0 & 0 & 0 \\ 0 & 0 & 0 & .2 & .8 & 0 & 0 \\ 0 & 0 & 0 & 0 & .2 & .8 & 0 \\ 0 & 0 & 0 & 0 & 0 & .2 & .8 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad T^{\text{right}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Noisy: plants, environments, agent

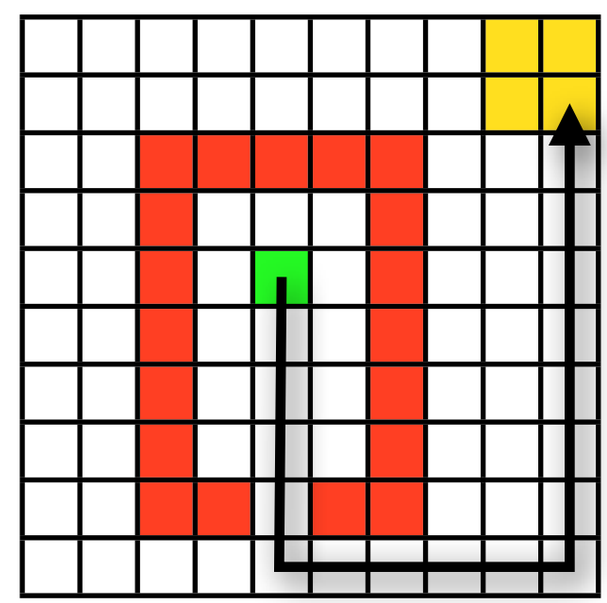
Absorbing state -> max eigenvalue < 1

Markovian dynamics

$$p(s_{t+1}|a_t, s_t, a_{t-1}, s_{t-1}, a_{t-2}, s_{t-2}, \cdots) = p(s_{t+1}|a_t, s_t)$$



Velocity

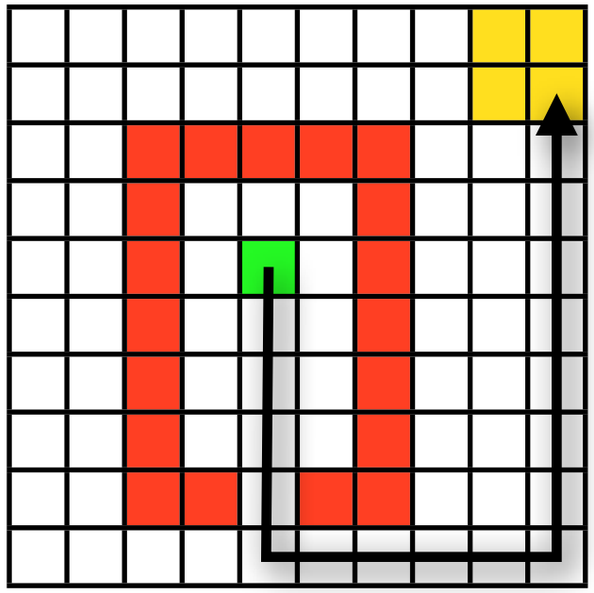


Markovian dynamics

$$p(s_{t+1}|a_t, s_t, a_{t-1}, s_{t-1}, a_{t-2}, s_{t-2}, \cdots) = p(s_{t+1}|a_t, s_t)$$

$$a_{t-2}, s_{t-2} \rightarrow a_{t-1}, s_{t-1} \rightarrow a_t, s_t$$

Velocity



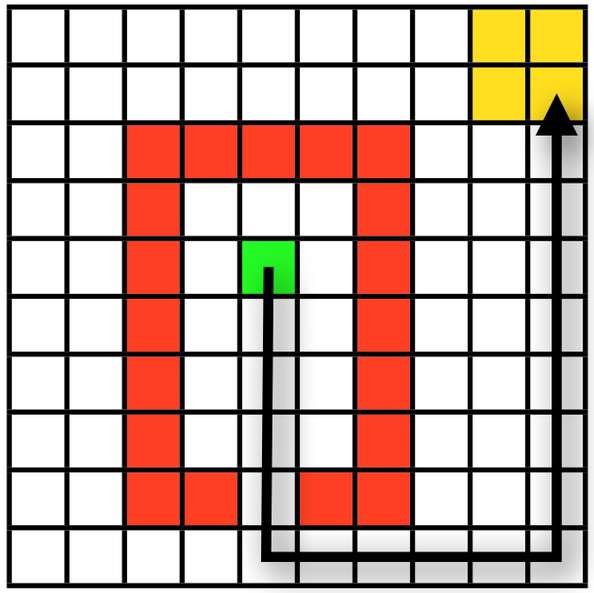
Markovian dynamics

$$p(s_{t+1}|a_t, s_t, a_{t-1}, s_{t-1}, a_{t-2}, s_{t-2}, \cdots) = p(s_{t+1}|a_t, s_t)$$

$$a_{t-2}, s_{t-2} \rightarrow a_{t-1}, s_{t-1} \rightarrow a_t, s_t$$

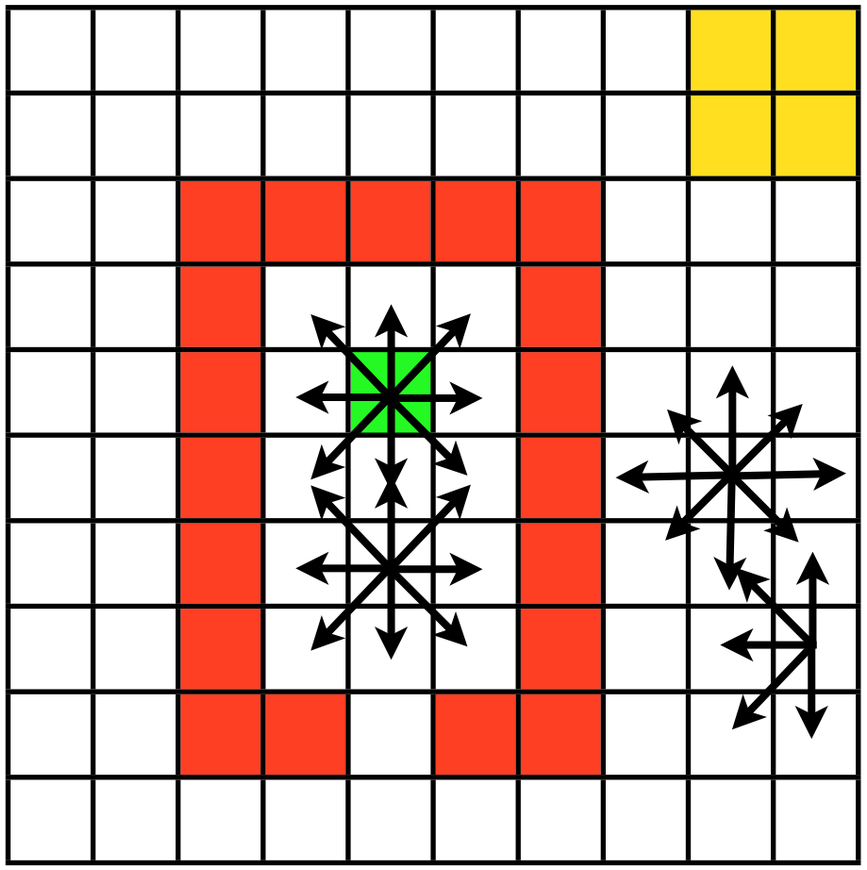
Velocity

$$s' = [\text{position}] \rightarrow s' = \begin{bmatrix} \text{position} \\ \text{velocity} \end{bmatrix}$$



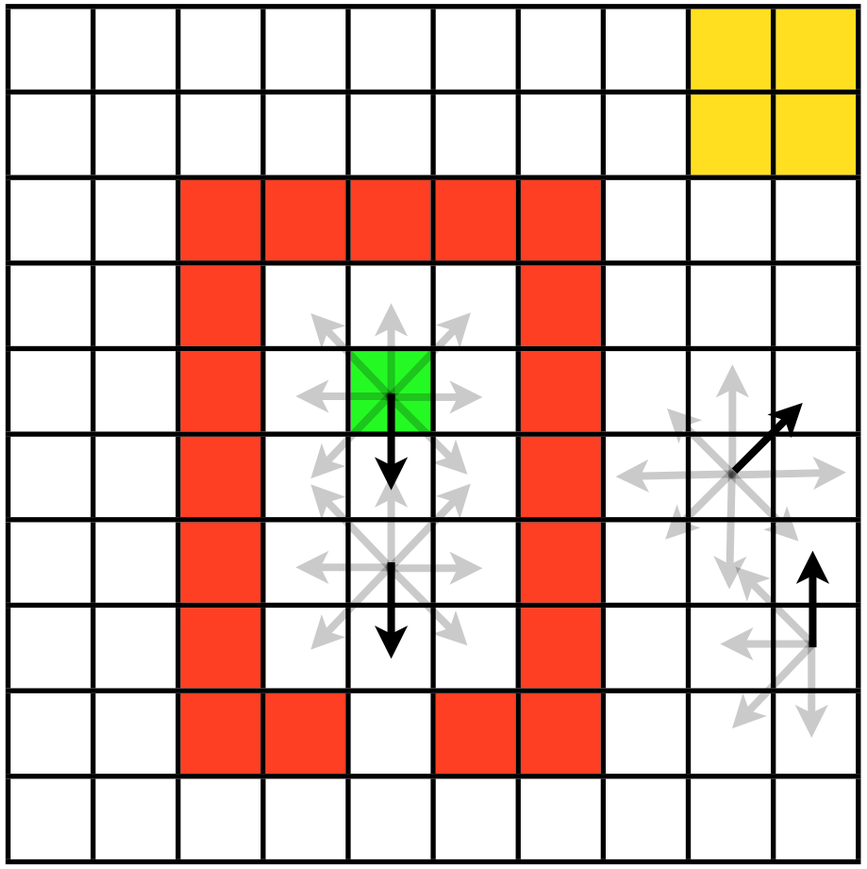
MDP

$$
\begin{aligned}
s_t &\in \mathcal{S} \\
a_t &\in \mathcal{A} \\
\mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \\
r_t &\sim \mathcal{R}(s_{t+1}, a_t, s_t) \\
\boxed{\pi(a|s) = p(a|s)}
\end{aligned}
$$



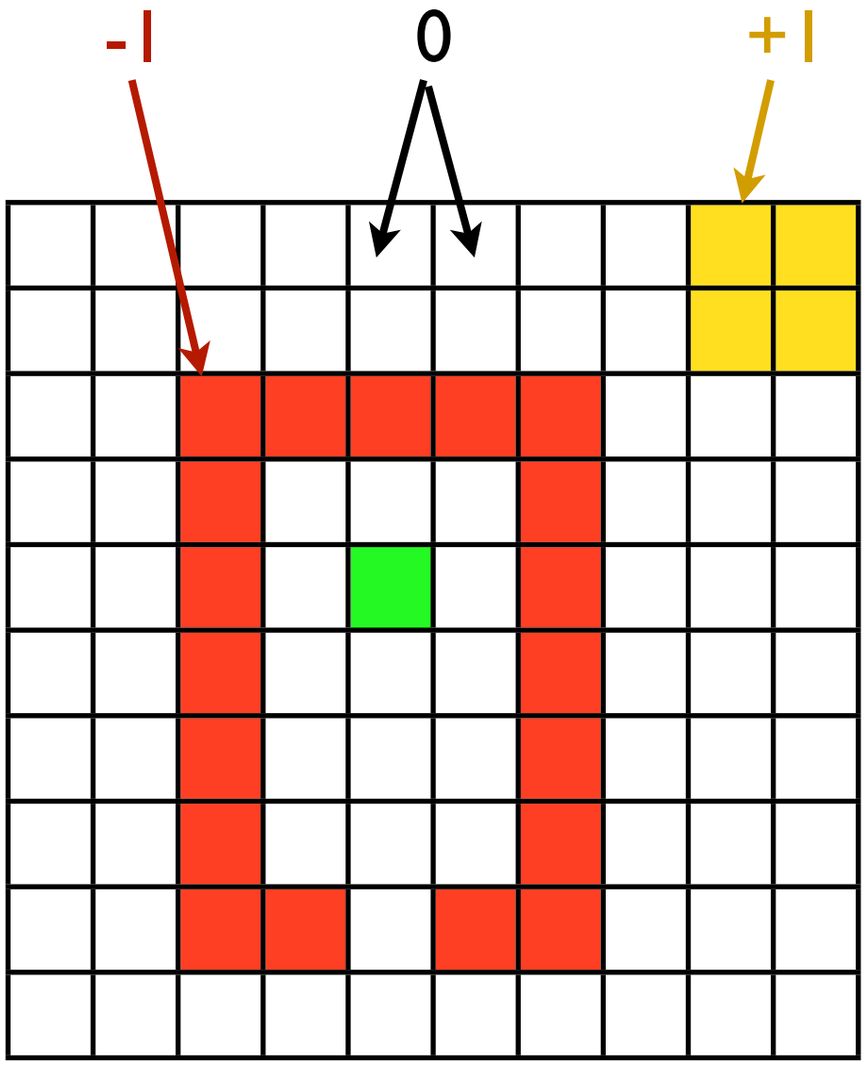
MDP

$$
\begin{aligned}
s_t &\in \mathcal{S} \\
a_t &\in \mathcal{A} \\
\mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \\
r_t &\sim \mathcal{R}(s_{t+1}, a_t, s_t) \\
\boxed{\pi(a|s) = p(a|s)}
\end{aligned}
$$



MDP

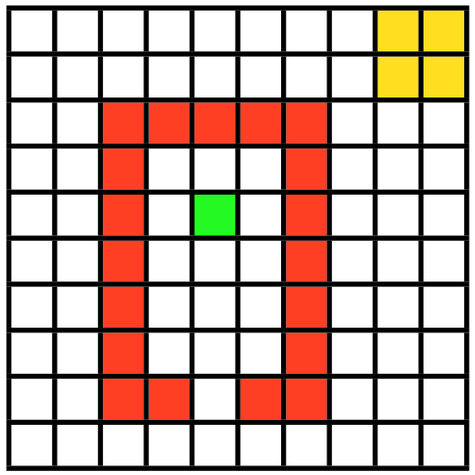
$$
\begin{aligned}
s_t &\in \mathcal{S} \\
a_t &\in \mathcal{A} \\
\mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \\
\boxed{r_t \sim \mathcal{R}(s_{t+1}, a_t, s_t)} \\
\pi(a|s) &= p(a|s)
\end{aligned}
$$



Tall orders

- Aim: maximise total future reward

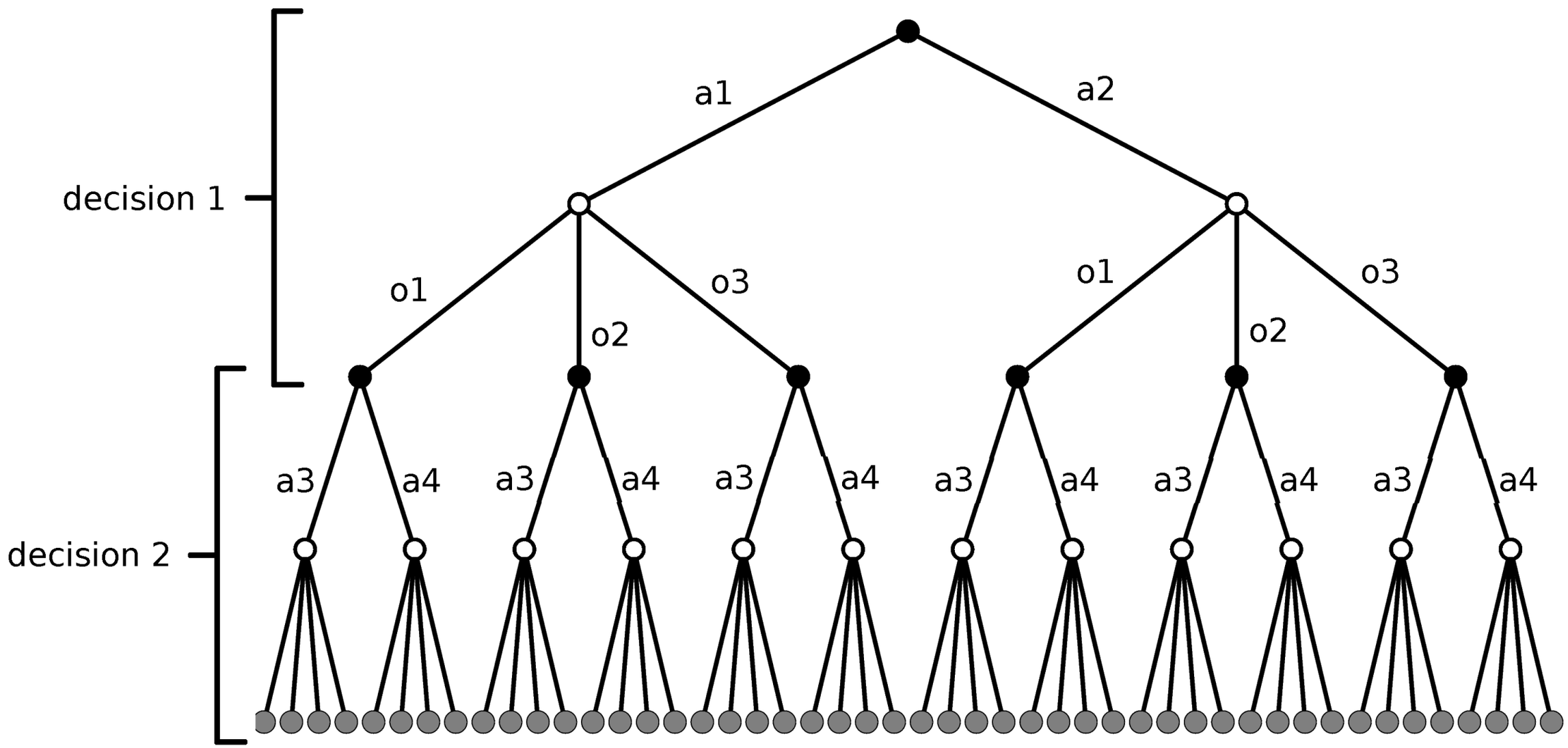
$$\sum_{t=1}^{\infty} r_t$$



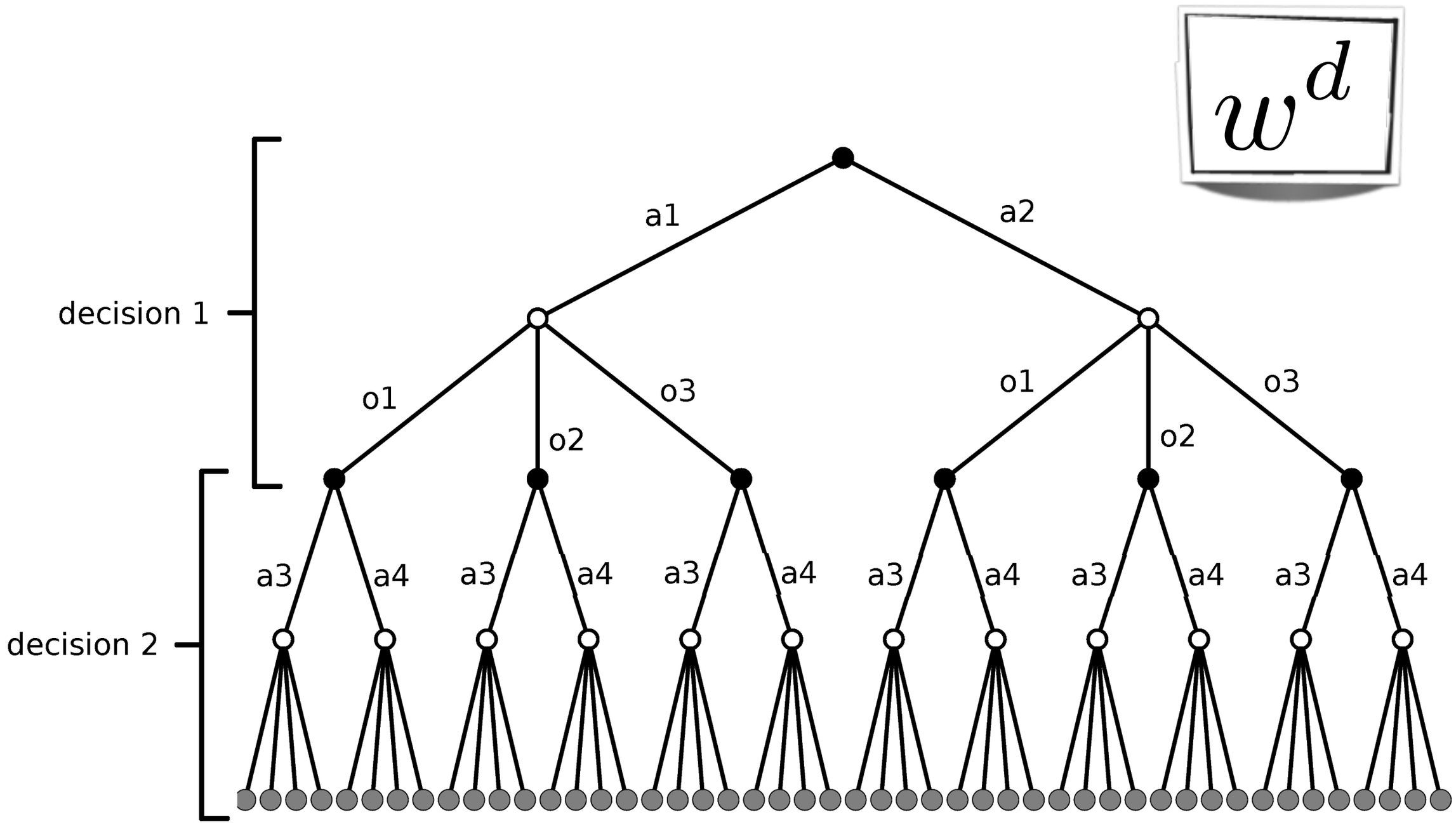
- i.e. we have to sum over paths through the future and weigh each by its probability

- Best policy achieves best long-term reward

Exhaustive tree search

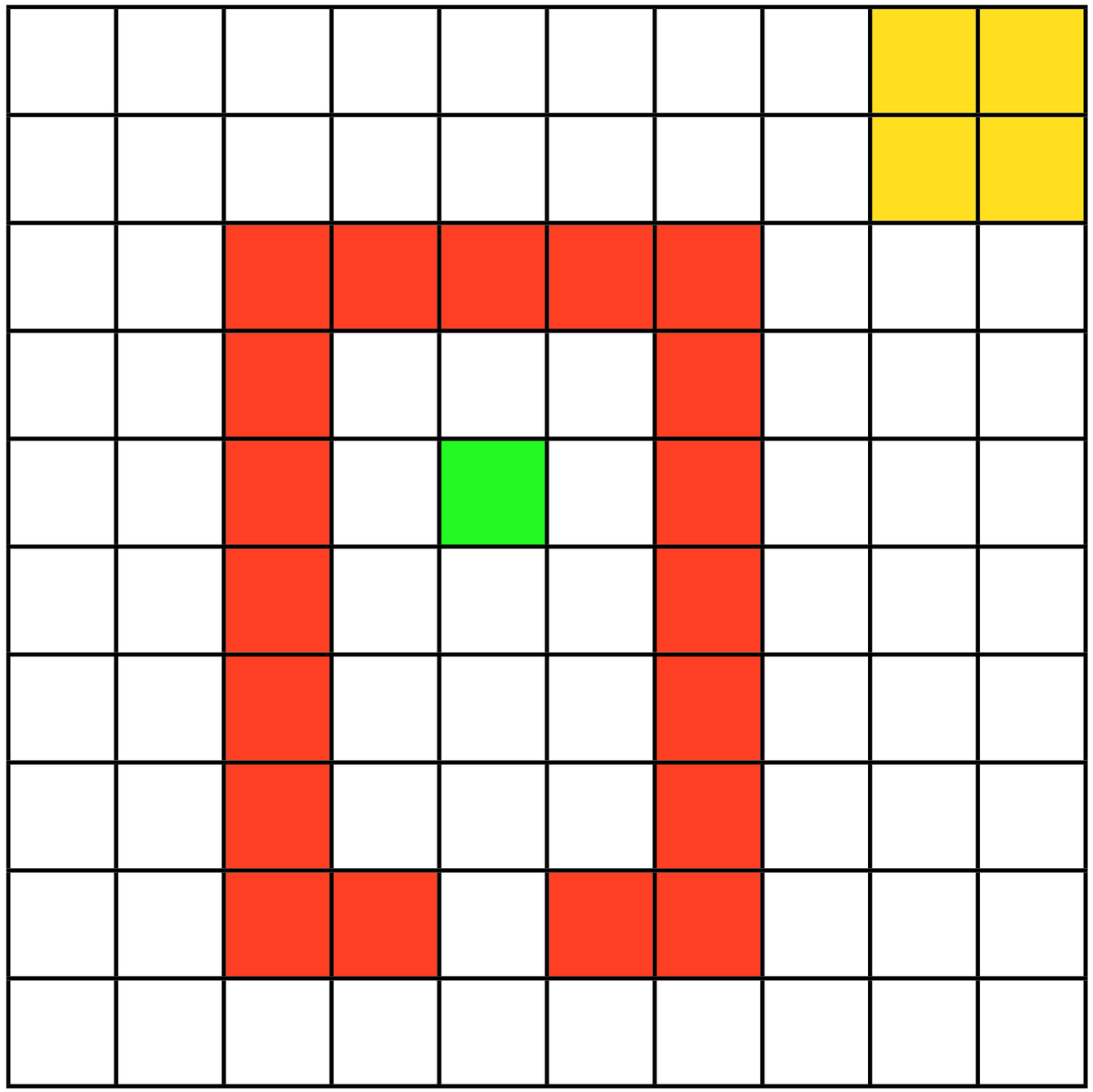


Exhaustive tree search



Decision tree

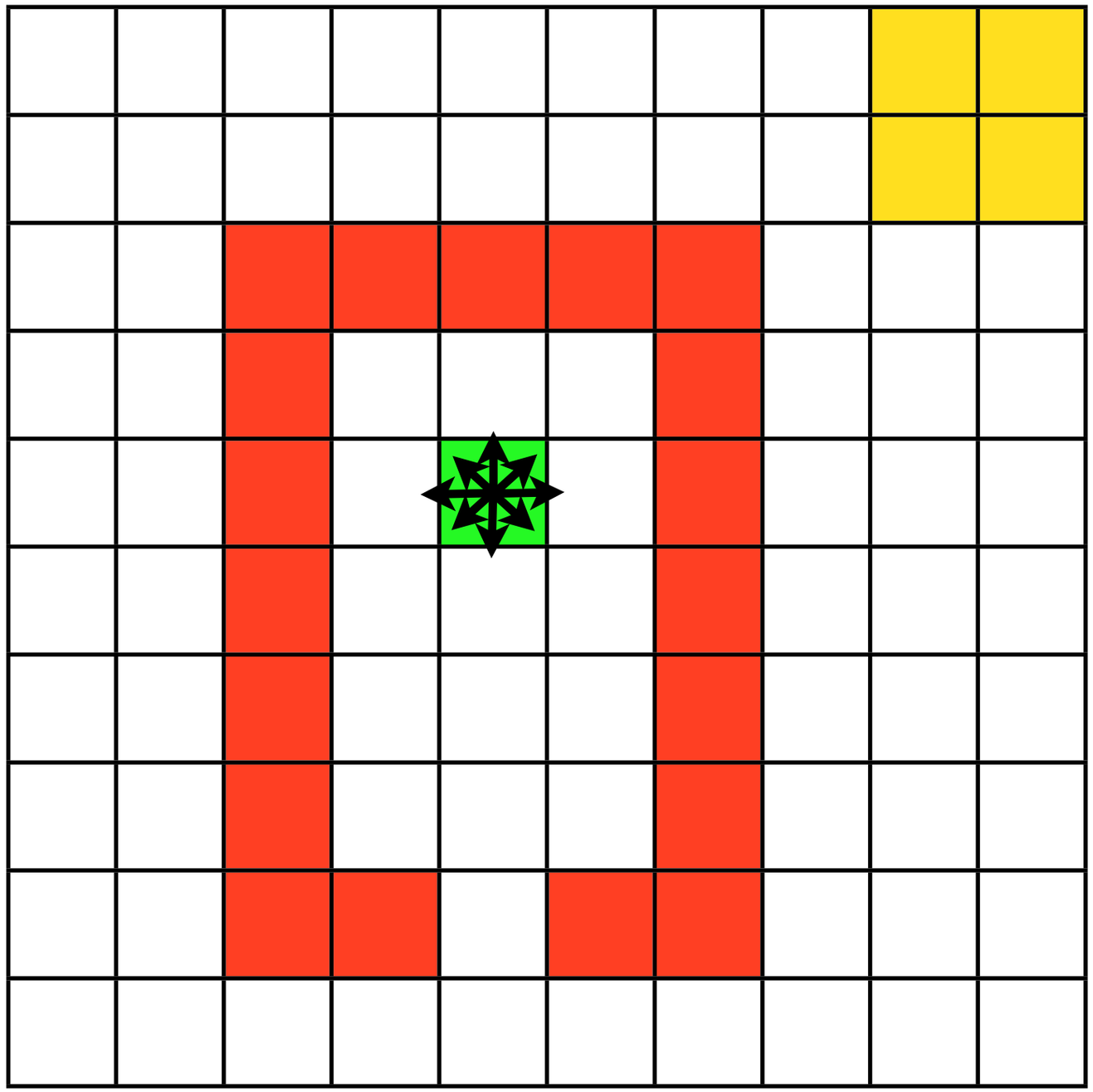
$$\sum_{t=1}^{\infty} r_t$$



Decision tree

$$\sum_{t=1}^{\infty} r_t$$

8

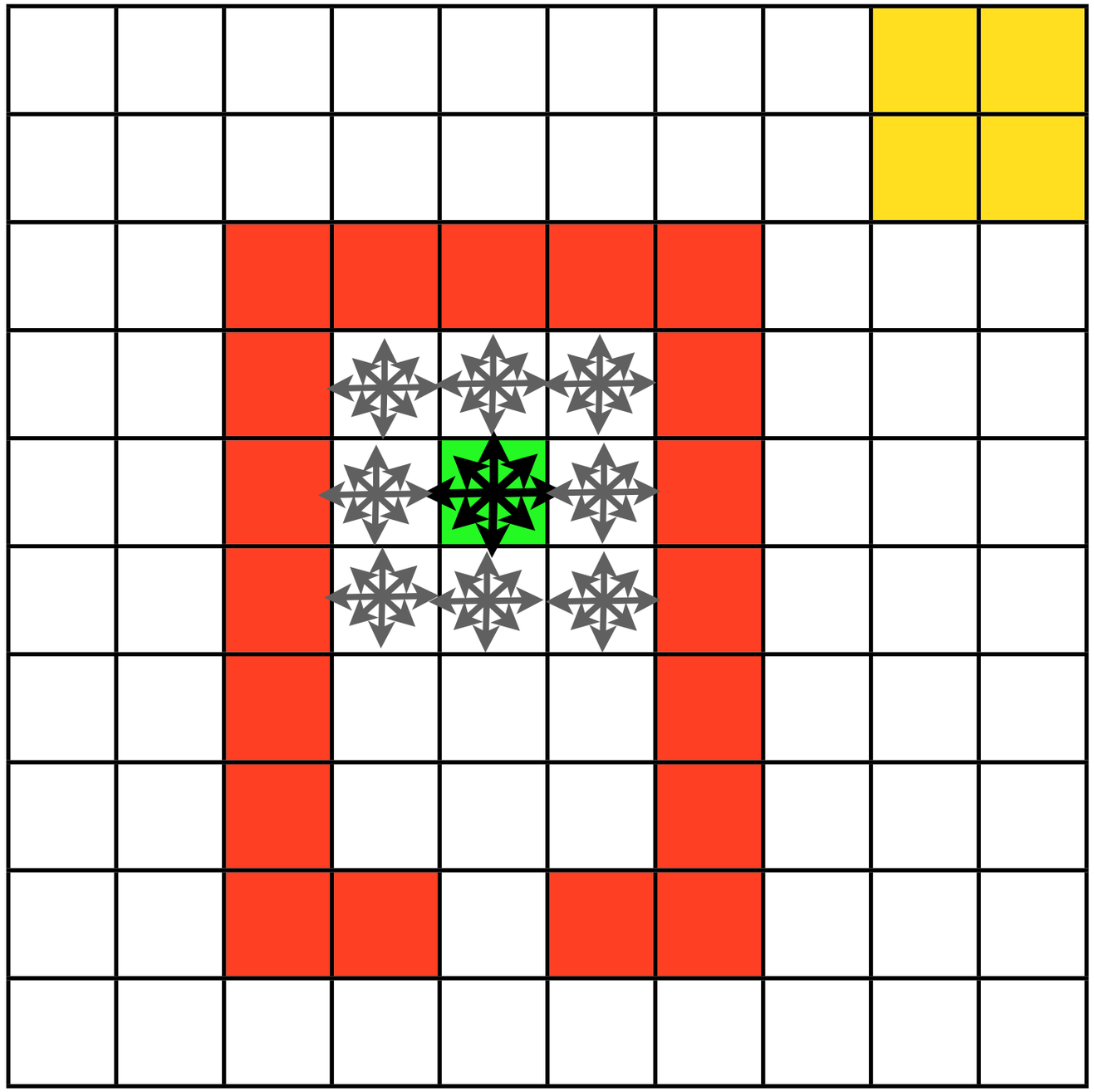


Decision tree

$$\sum_{t=1}^{\infty} r_t$$

8

64



Decision tree

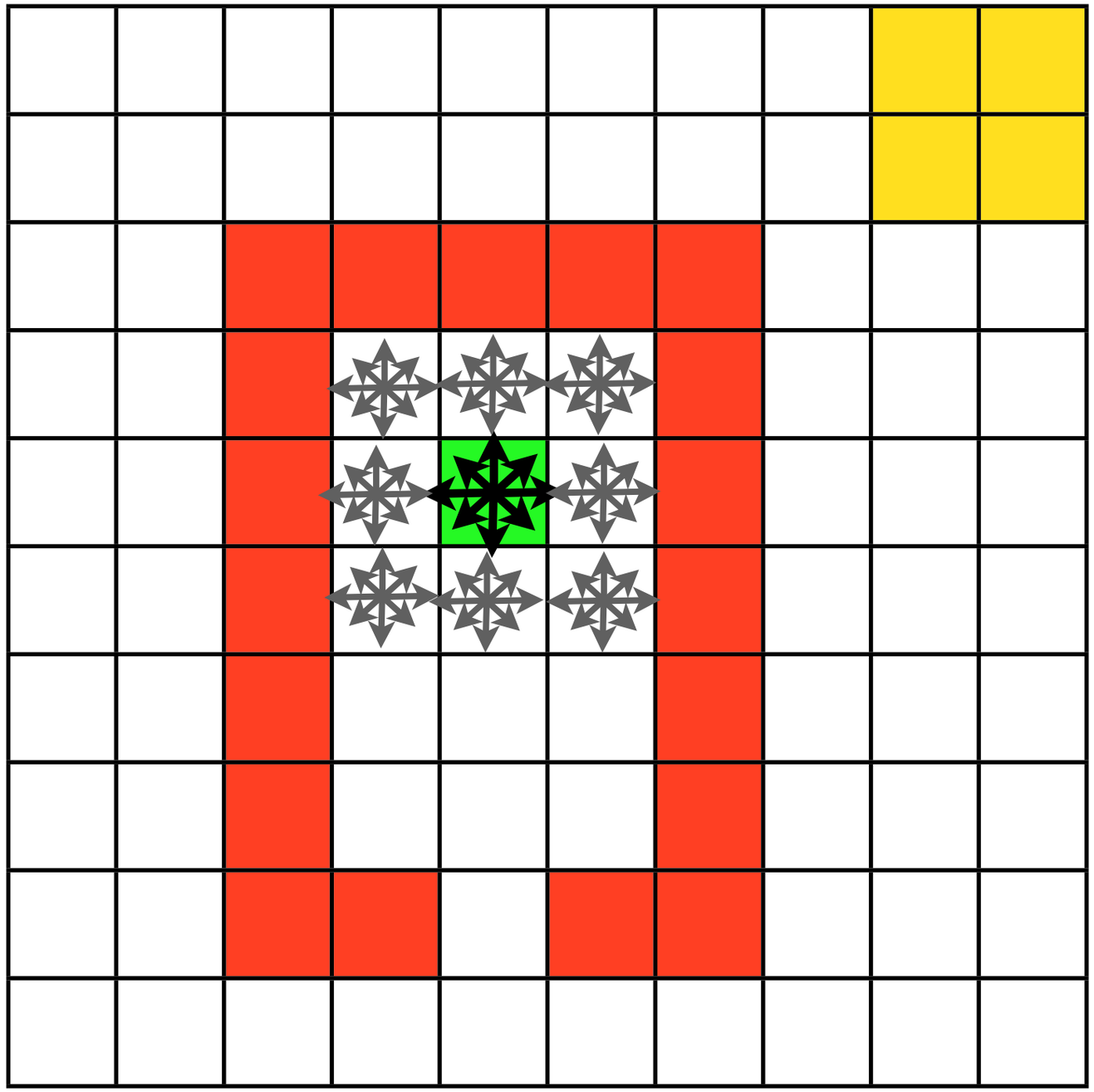
$$\sum_{t=1}^{\infty} r_t$$

8

64

512

...



Policy for this talk

- Pose the problem mathematically
- Policy evaluation
- Policy iteration
- Monte Carlo techniques: experience samples
- TD learning

Policy

Evaluate $\rightleftarrows$ Update

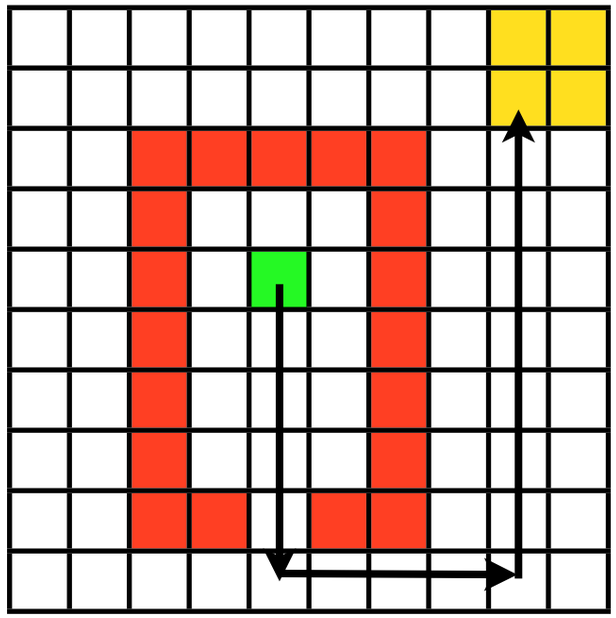
Evaluating a policy

- Aim: maximise total future reward

$$\sum_{t=1}^{\infty} r_t$$

- To know which is best, evaluate it first
- The policy determines the expected reward from each state

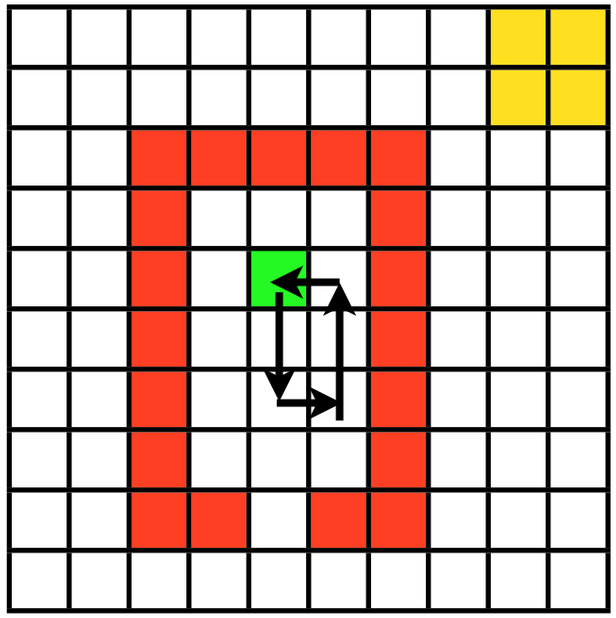
$$\mathcal{V}^{\pi}(s_1) \quad = \quad \mathbb{E}\left[\sum_{t=1}^{\infty} r_t | s_1 = 1, a_t \sim \pi\right]$$



Discounting

- Given a policy, each state has an expected value

$$\mathcal{V}^{\pi}(s_1) \quad = \quad \mathbb{E}\left[\sum_{t=1}^{\infty} r_t | s_1 = 1, a_t \sim \pi\right]$$



- But: $\sum_{t=0}^{\infty} r_t = \infty$

- Episodic $\sum_{t=0}^{T} r_t < \infty$

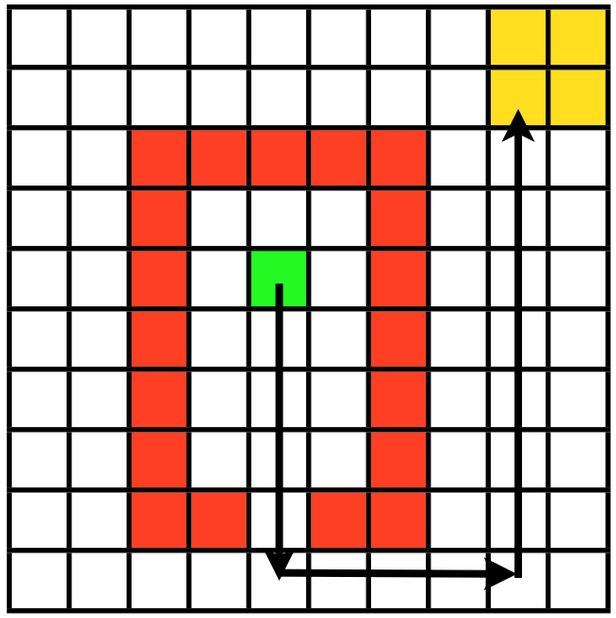
- Discounted
  - infinite horizons $\sum_{t=0}^{\infty} \gamma^t r_t < \infty$
  - finite, exponentially distributed horizons

$$\sum_{t=0}^{T} \gamma^t r_t \qquad\qquad T \sim \frac{1}{\tau} e^{t/\tau}$$

Discounting

- Given a policy, each state has an expected value

$$\mathcal{V}^\pi(s_1) \quad = \quad \mathbb{E}\left[\sum_{t=1}^{\infty} r_t | s_1 = 1, a_t \sim \pi\right]$$



- But: $$\sum_{t=0}^{\infty} r_t = \infty$$

- Episodic $$\sum_{t=0}^{T} r_t < \infty$$

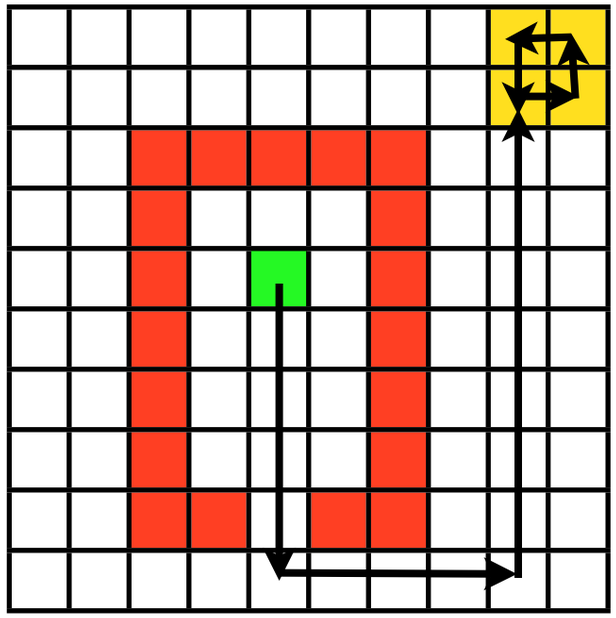
- Discounted
  - infinite horizons $$\sum_{t=0}^{\infty} \gamma^t r_t < \infty$$
  - finite, exponentially distributed horizons

$$\sum_{t=0}^{T} \gamma^t r_t \qquad\qquad T \sim \frac{1}{\tau} e^{t/\tau}$$

Discounting

- Given a policy, each state has an expected value

$$\mathcal{V}^\pi(s_1) \quad = \quad \mathbb{E}\left[\sum_{t=1}^{\infty} r_t | s_1 = 1, a_t \sim \pi\right]$$



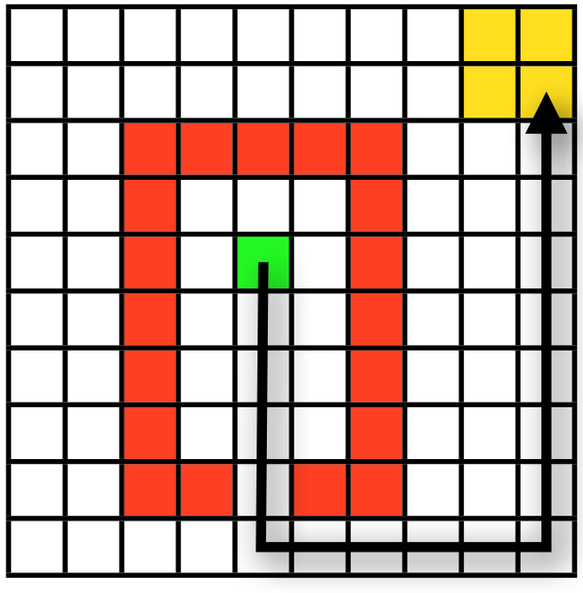
- But: $$\sum_{t=0}^{\infty} r_t = \infty$$

- Episodic $$\sum_{t=0}^{T} r_t < \infty$$

- Discounted
  - infinite horizons $$\sum_{t=0}^{\infty} \gamma^t r_t < \infty$$
  - finite, exponentially distributed horizons

$$\sum_{t=0}^{T} \gamma^t r_t \qquad\qquad T \sim \frac{1}{\tau} e^{t/\tau}$$

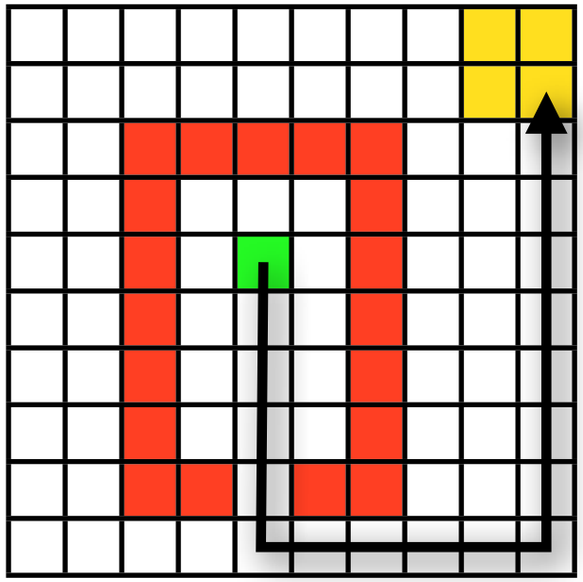
Markov Decision Problems



$$
\begin{aligned}
V^{\pi}(s_t) &= \mathbb{E}\left[\sum_{t'=1}^{\infty} r_{t'} | s_t = s, \pi\right] \\
&= \mathbb{E}\left[r_1 | \, s_t = s, \pi\right] + \mathbb{E}\left[\sum_{t=2}^{\infty} r_t | s_t = s, \pi\right] \\
&= \mathbb{E}\left[r_1 | \, s_t = s, \pi\right] + \mathbb{E}\left[V^{\pi}(s_{t+1}) | s_t = s, \pi\right]
\end{aligned}
$$

This dynamic consistency is key to many solution approaches.
It states that the value of a state s is related to
the values of its successor states s'.

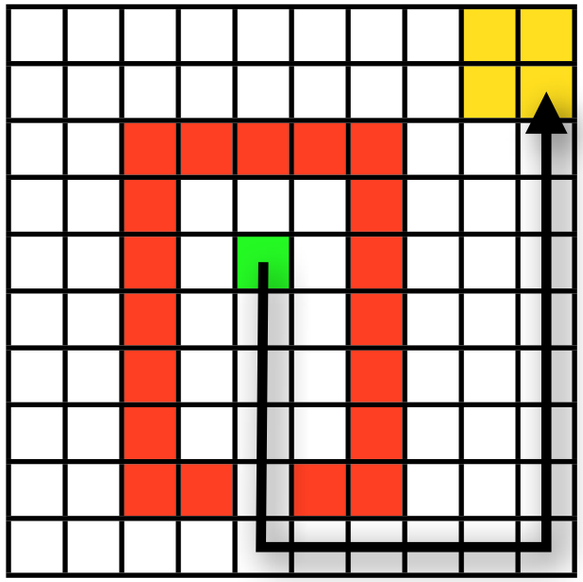
Markov Decision Problems



$$
\begin{aligned}
V^{\pi}(s_t) &= \boxed{\mathbb{E}\left[r_1 \middle| s_t = s, \pi\right]} + \mathbb{E}\left[V(s_{t+1}), \pi\right] \\
r_1 &\sim \mathcal{R}(s_2, a_1, s_1)
\end{aligned}
$$

$$
\begin{aligned}
\mathbb{E}\left[r_1 | s_t = s, \pi\right] &= \mathbb{E}\left[\sum_{s_{t+1}} p(s_{t+1}|s_t, a_t)\mathcal{R}(s_{t+1}, a_t, s_t)\right] \\
&= \sum_{a_t} p(a_t|s_t)\left[\sum_{s_{t+1}} p(s_{t+1}|s_t, a_t)\mathcal{R}(s_{t+1}, a_t, s_t)\right] \\
&= \sum_{a_t} \pi(a_t, s_t)\left[\sum_{s_{t+1}} \mathcal{T}^{a_t}_{s_t s_{t+1}} \mathcal{R}(s_{t+1}, a_t, s_t)\right]
\end{aligned}
$$

Bellman equation



$$
\begin{aligned}
V^{\pi}(s_t) &= \mathbb{E}\left[r_1 \middle| s_t = s, \pi\right] + \mathbb{E}\left[V(s_{t+1}), \pi\right] \\
\mathbb{E}\left[r_1 | s_t, \pi\right] &= \sum_a \pi(a, s_t) \left[\sum_{s_{t+1}} \mathcal{T}^a_{s_t s_{t+1}} \mathcal{R}(s_{t+1}, a, s_t)\right] \\
\mathbb{E}\left[V^{\pi}(s_{t+1}), \pi, s_t\right] &= \sum_a \pi(a, s_t) \left[\sum_{s_{t+1}} \mathcal{T}^a_{s_t s_{t+1}} V^{\pi}(s_{t+1})\right]
\end{aligned}
$$

$$
V^{\pi}(s) = \sum_a \pi(a|s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V^{\pi}(s')\right]\right]
$$

Bellman Equation

$$V^{\pi}(s) \quad = \quad \sum_{a} \pi(a|s) \left[\sum_{s'} \mathcal{T}^{a}_{ss'} \left[\mathcal{R}(s', a, s) + V^{\pi}(s') \right] \right]$$

Bellman Equation

All future reward from state s = E [Immediate reward + All future reward from next state s']

$$V^{\pi}(s) = \sum_{a} \pi(a|s) \left[\sum_{s'} \mathcal{T}^{a}_{ss'} \left[\mathcal{R}(s', a, s) + V^{\pi}(s') \right] \right]$$

Bellman Equation

$$V^{\pi}(s) = \sum_{a} \pi(a|s) \left[\sum_{s'} \mathcal{T}^{a}_{ss'} \left[\mathcal{R}(s', a, s) + V^{\pi}(s') \right] \right]$$

All future reward from state s = E [Immediate reward + All future reward from next state s']

Q values = state-action values

$$V^\pi(s) = \sum_a \pi(a|s) \underbrace{\left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V^\pi(s') \right] \right]}_{\mathcal{Q}^\pi(s,a)}$$

- so we can define state-action values as:

$$\begin{aligned} \mathcal{Q}(s,a) &= \sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V(s') \right] \\ &= \mathbb{E} \left[\sum_{t=1}^{\infty} r_t | s, a \right] \end{aligned}$$

- and state values are average state-action values:

$$V(s) = \sum_a \pi(a|s) \mathcal{Q}(s,a)$$

Bellman Equation

$$V^\pi(s) = \sum_a \pi(a|s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V^\pi(s') \right] \right]$$

- to evaluate a policy, we need to solve the above equation, i.e. find the self-consistent state values

- options for policy evaluation
  - exhaustive tree search - outwards, inwards, depth-first
  - linear solution in 1 step
  - value iteration: iterative updates
  - experience sampling

Solving the Bellman Equation

Option 1: turn it into update equation

Option 2: linear solution (w/ absorbing states)

$$
\begin{aligned}
V(s) &= \sum_a \pi(a, s_t) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V(s') \right] \right] \\
\Rightarrow \mathbf{v} &= \mathbf{R}^\pi + \mathbf{T}^\pi \mathbf{v} \\
\Rightarrow \mathbf{v}^\pi &= (\mathbf{I} - \mathbf{T}^\pi)^{-1} \mathbf{R}^\pi \qquad \mathcal{O}(|\mathcal{S}|^3)
\end{aligned}
$$

Solving the Bellman Equation

Option 1: turn it into update equation

$$V^{k+1}(s) = \sum_a \pi(a, s_t) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V^k(s') \right] \right]$$

Option 2: linear solution (w/ absorbing states)

$$\begin{aligned} V(s) &= \sum_a \pi(a, s_t) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V(s') \right] \right] \\ \Rightarrow \mathbf{v} &= \mathbf{R}^\pi + \mathbf{T}^\pi \mathbf{v} \\ \Rightarrow \mathbf{v}^\pi &= (\mathbf{I} - \mathbf{T}^\pi)^{-1} \mathbf{R}^\pi \qquad \mathcal{O}(|\mathcal{S}|^3) \end{aligned}$$

Policy update

Given the value function for a policy, say via linear solution

$$V^\pi(s) \quad = \quad \sum_a \pi(a|s) \underbrace{\left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V^\pi(s') \right] \right]}_{\mathcal{Q}^\pi(s,a)}$$

Given the values V for the policy, we can improve the policy by always choosing the best action:

$$\pi'(a|s) = \begin{cases} 1 \,\text{if}\, a = \text{argmax}_a \, \mathcal{Q}^\pi(s,a) \\ 0 \,\text{else} \end{cases}$$

It is guaranteed to improve:

for deterministic policy

$$\mathcal{Q}^\pi(s, \pi'(s)) = \max_a \mathcal{Q}^\pi(s,a) \geq \mathcal{Q}^\pi(s, \pi(s)) = \mathcal{V}^\pi(s)$$

Policy iteration

Policy evaluation

$$\mathbf{v}^{\pi} = (\mathbf{I} - \mathbf{T}^{\pi})^{-1}\mathbf{R}^{\pi}$$

$$\pi(a|s) = \begin{cases} 1 \text{ if } a = \text{argmax}_a \sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}^a_{ss} + V^{pi}(s')\right] \\ 0 \text{ else} \end{cases}$$

Policy iteration

Policy evaluation

$$\mathbf{v}^{\pi} = (\mathbf{I} - \mathbf{T}^{\pi})^{-1}\mathbf{R}^{\pi}$$

greedy policy improvement

$$\pi(a|s) = \begin{cases} 1 \text{ if } a = \text{argmax}_a \sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}^a_{ss} + V^{pi}(s')\right] \\ 0 \text{ else} \end{cases}$$

Policy iteration

Policy evaluation

$$\mathbf{v}^{\pi} = (\mathbf{I} - \mathbf{T}^{\pi})^{-1}\mathbf{R}^{\pi}$$

Value iteration

$$V^*(s) = \max_a \sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}^a_{ss} + V^*(s')\right]$$

greedy policy improvement

$$\pi(a|s) = \begin{cases} 1 \text{ if } a = \text{argmax}_a \sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}^a_{ss} + V^{pi}(s')\right] \\ 0 \text{ else} \end{cases}$$

Model-free solutions

- So far we have assumed knowledge of R and T
  - R and T are the 'model' of the world, so we assume full knowledge of the dynamics and rewards in the environment
- What if we don't know them?
- We can still learn from state-action-reward samples
  - we can learn R and T from them, and use our estimates to solve as above
  - alternatively, we can directly estimate V or Q

Solving the Bellman Equation

Option 3: sampling

$$V(s) = \sum_a \pi(a, s_t) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V(s') \right] \right]$$

So we can just draw some samples from the policy and the transitions and average over them:

$$a = \sum_k f(x_k) p(x_k)$$

$$x^{(i)} \sim p(x) \rightarrow \hat{a} = \frac{1}{N} \sum_i f(x^{(i)})$$

Solving the Bellman Equation

Option 3: sampling

So we can just draw some samples from the policy and the transitions and average over them:

$$a = \sum_k f(x_k)p(x_k)$$

$$x^{(i)} \sim p(x) \rightarrow \hat{a} = \frac{1}{N}\sum_i f(x^{(i)})$$

Solving the Bellman Equation

Option 3: sampling

this is an expectation over policy and transition samples.

So we can just draw some samples from the policy and the transitions and average over them:

$$a = \sum_k f(x_k)p(x_k)$$

$$x^{(i)} \sim p(x) \rightarrow \hat{a} = \frac{1}{N}\sum_i f(x^{(i)})$$

Solving the Bellman Equation

Option 3: sampling

this is an expectation over policy and transition samples.

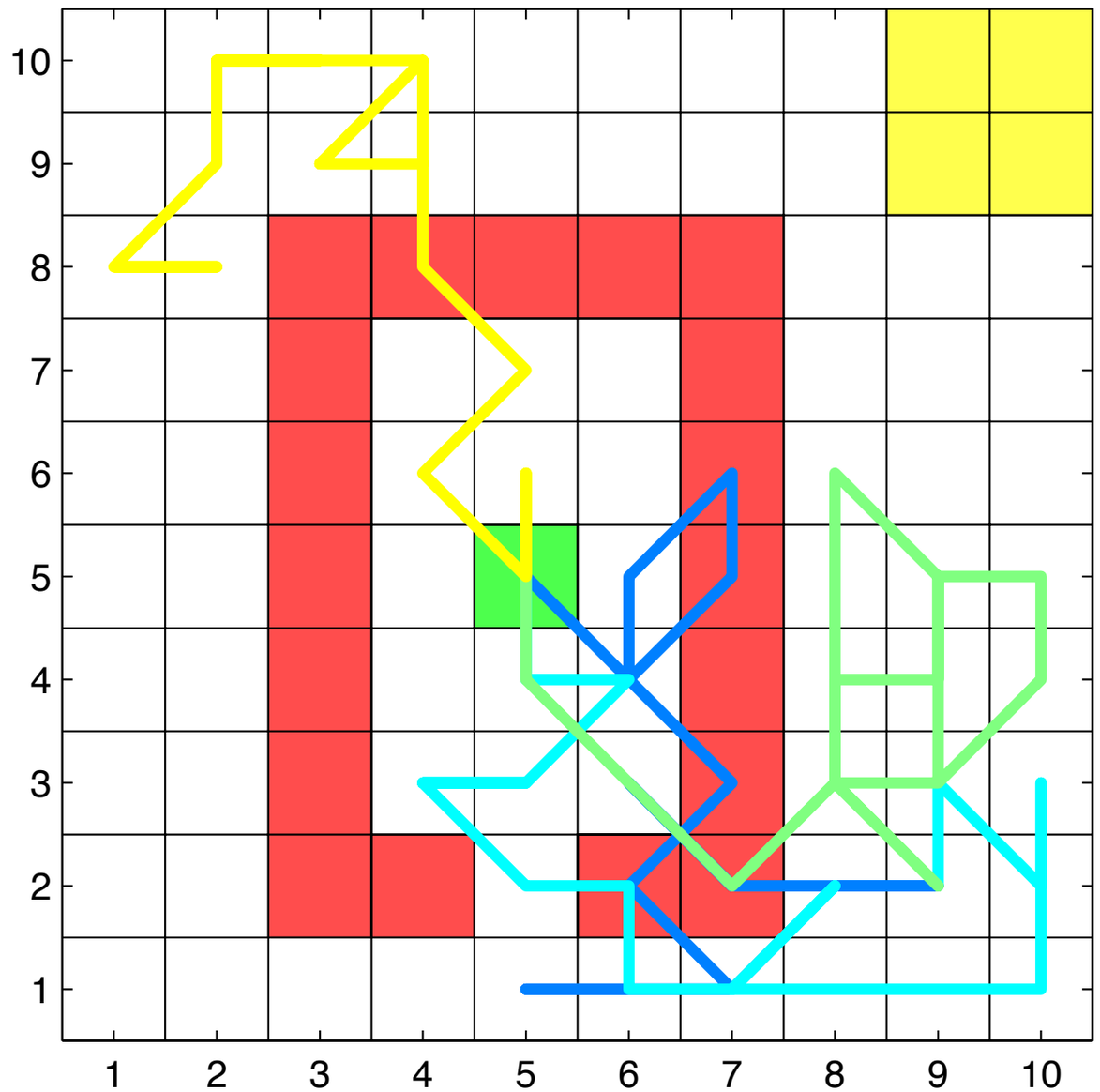
So we can just draw some samples from the policy and the transitions and average over them:

$$a = \sum_k f(x_k)p(x_k)$$

$$x^{(i)} \sim p(x) \rightarrow \hat{a} = \frac{1}{N}\sum_i f(x^{(i)})$$

more about this later...

Learning from samples



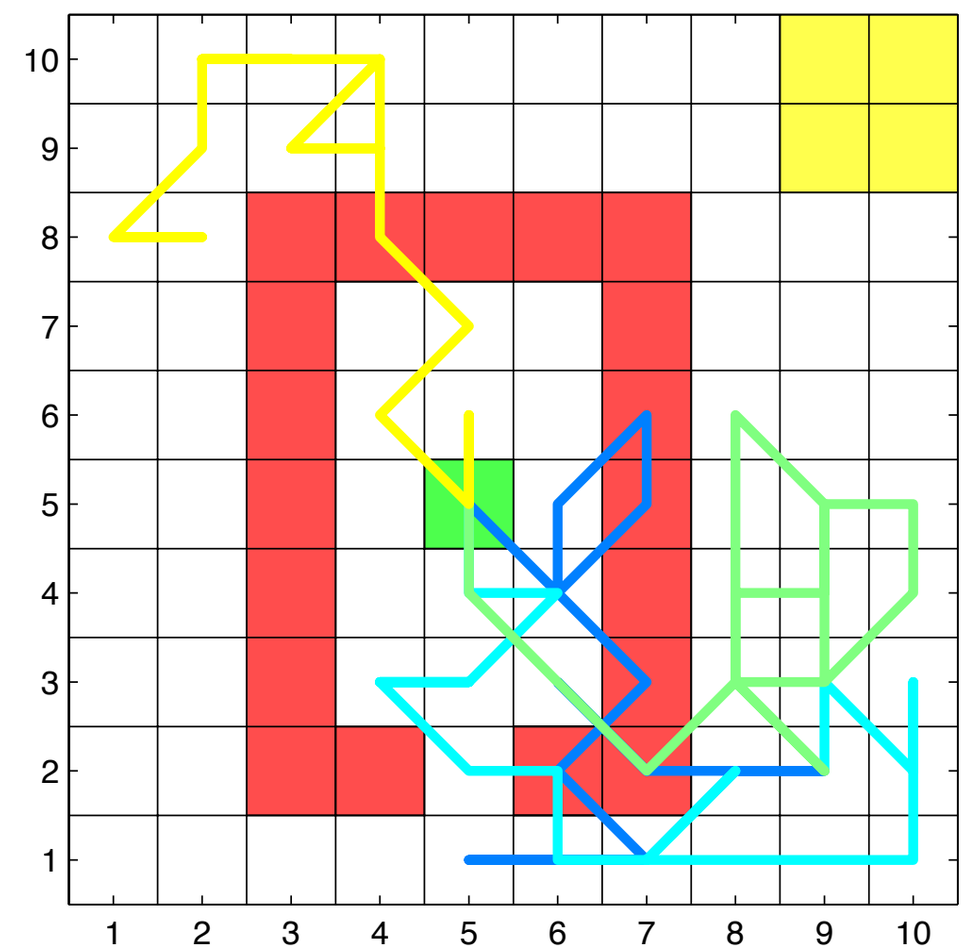
A new problem: exploration versus exploitation

Monte Carlo

- First visit MC
  - randomly start in all states, generate paths, average for starting state only

$$\mathcal{V}(s) = \frac{1}{N} \sum_i \left\{ \sum_{t'=1}^{T} r_{t'}^i | s_0 = s \right\}$$

- More efficient use of samples
  - Every visit MC
  - Bootstrap: TD
  - Dyna
- Better samples
  - on policy versus off policy
  - Stochastic search, UCT...



Update equation: towards TD

Bellman equation

$$V(s) = \sum_a \pi(a,s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s') \right] \right]$$

Not yet converged, so it doesn't hold:

$$dV(s) = -V(s) + \sum_a \pi(a,s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s') \right] \right]$$

And then use this to update

$$V^{i+1}(s) = V^i(s) + dV(s)$$

TD learning

$$dV(s) = -V(s) + \sum_a \pi(a,s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s') \right] \right]$$

TD learning

$$dV(s) = -V(s) + \sum_a \pi(a,s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s') \right] \right]$$

Sample

$$\begin{aligned} a_t &\sim \pi(a|s_t) \\ s_{t+1} &\sim \mathcal{T}^{a_t}_{s_t,s_{t+1}} \\ r_t &= \mathcal{R}(s_{t+1}, a_t, s_t) \end{aligned}$$

TD learning

$$dV(s) = -V(s) + \sum_a \pi(a,s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s') \right] \right]$$

Sample

$$\begin{aligned} a_t &\sim \pi(a|s_t) \\ s_{t+1} &\sim \mathcal{T}^{a_t}_{s_t,s_{t+1}} \\ r_t &= \mathcal{R}(s_{t+1},a_t,s_t) \end{aligned}$$

$$\delta_t = -V_{t-1}(s_t) + r_t + V_{t-1}(s_{t+1})$$

TD learning

$$dV(s) = -V(s) + \sum_a \pi(a,s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s') \right] \right]$$

Sample

$$\begin{aligned} a_t &\sim \pi(a|s_t) \\ s_{t+1} &\sim \mathcal{T}^{a_t}_{s_t,s_{t+1}} \\ r_t &= \mathcal{R}(s_{t+1}, a_t, s_t) \end{aligned}$$

$$\delta_t = -V_{t-1}(s_t) + r_t + V_{t-1}(s_{t+1})$$

$$V^{i+1}(s) = V^i(s) + dV(s) \qquad V_t(s_t) = V_{t-1}(s_t) + \alpha\delta_t$$

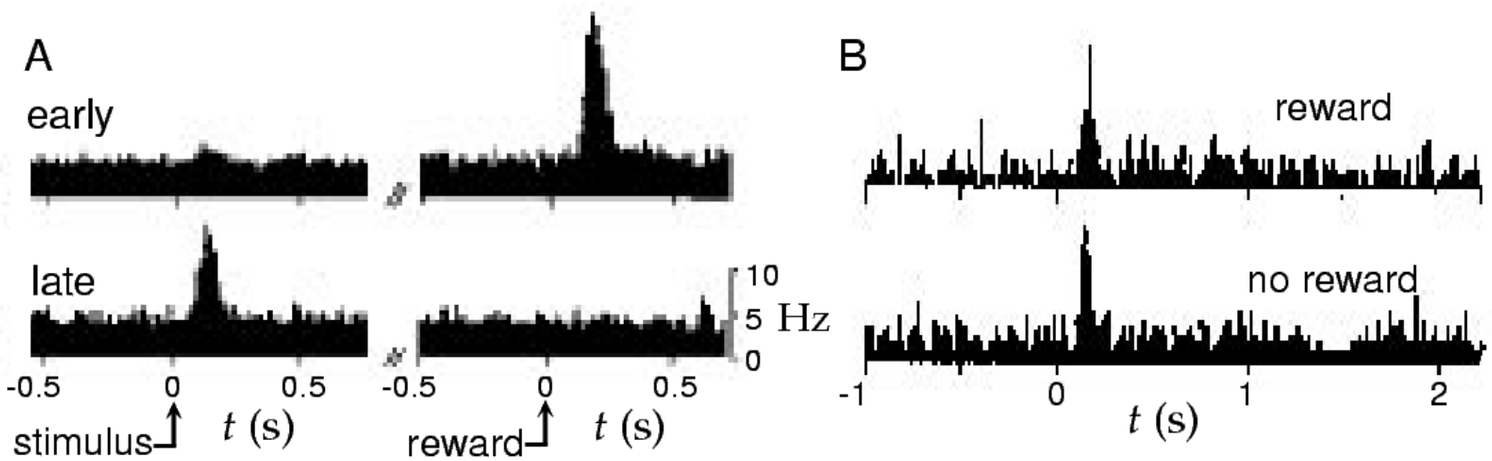
TD learning

$$
\begin{aligned}
a_t &\sim \pi(a|s_t) \\
s_{t+1} &\sim \mathcal{T}^{a_t}_{s_t,s_{t+1}} \\
r_t &= \mathcal{R}(s_{t+1}, a_t, s_t)
\end{aligned}
$$

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

$$V_{t+1}(s_t) = V_t(s_t) + \alpha\delta_t$$

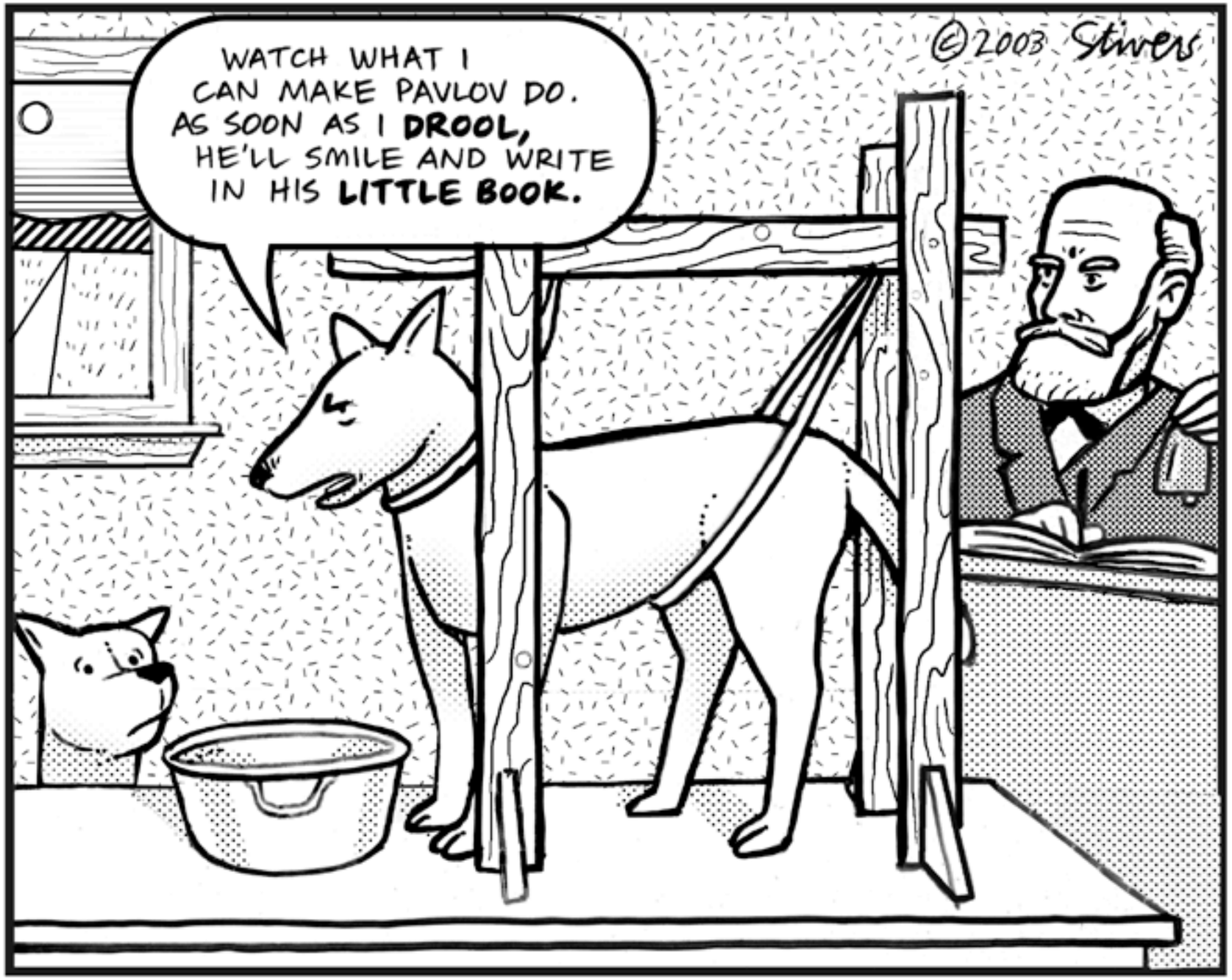
Aside: what makes a TD error?



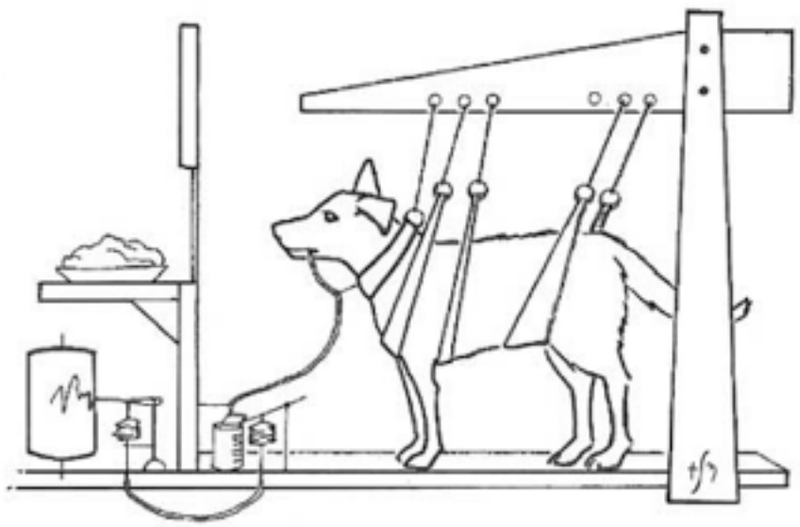
- unpredicted reward expectation change
- disappears with learning
- stays with probabilistic reinforcement
- sequentiality
  - TD error vs prediction error
- see Huys et al. 2013 Prog. Neurobiol.

Schultz et al.

Pavlovian conditioning



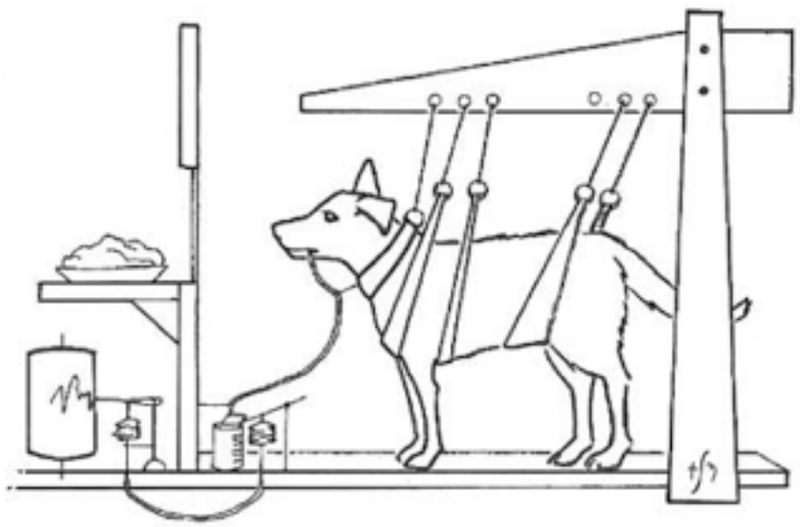
Pavlovian conditioning



prior to training

CS: bell -> no response
US: food -> salivation & consumption

Pavlovian conditioning



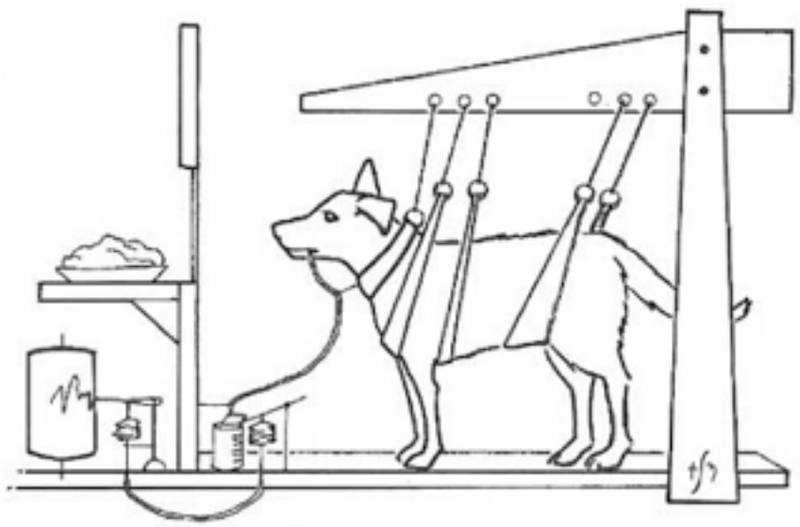
prior to training

CS: bell -> no response
US: food -> salivation & consumption

training

CS: bell ⟶ US: food

Pavlovian conditioning

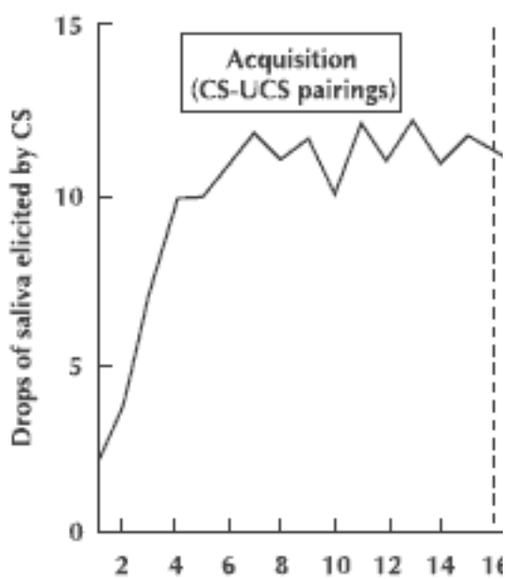


prior to training

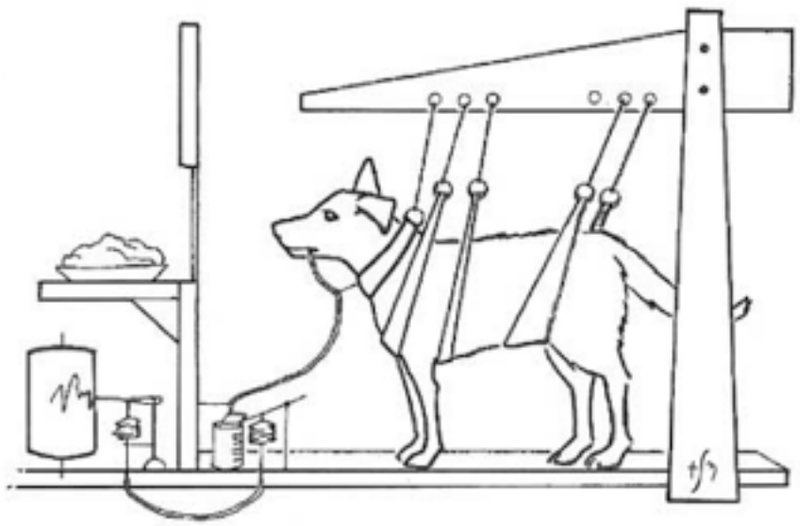
CS: bell -> no response
US: food -> salivation & consumption

training

CS: bell ⟶ US: food



Pavlovian conditioning



prior to training

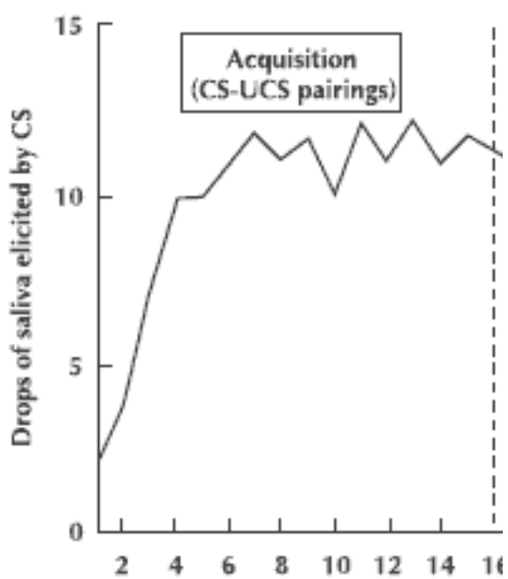
CS: bell -> no response
US: food -> salivation & consumption

training

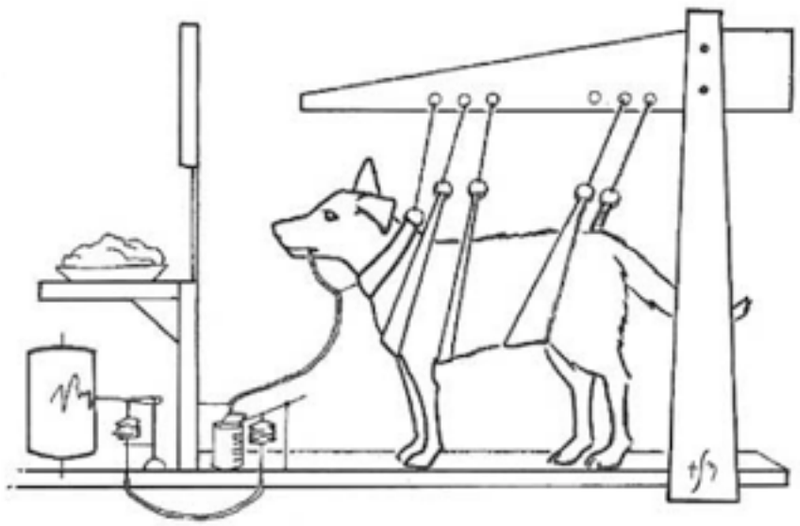
CS: bell ⟶ US: food

after training:

CS: bell -> salivation
US: food -> salivation & consumption



Pavlovian conditioning



prior to training

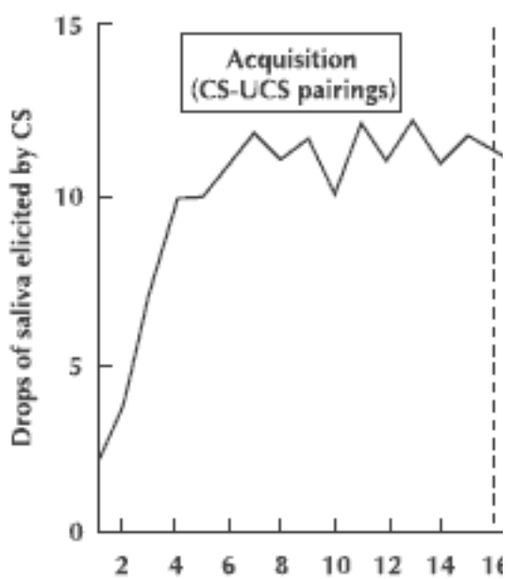
CS: bell -> no response
US: food -> salivation & consumption

training

CS: bell ⟶ US: food

after training:

CS: bell -> salivation
US: food -> salivation & consumption



omission training:

CS: bell —salivation→ US: omitted
CS: bell —no salivation→ US: food

Pavlovian and instrumental conditioning

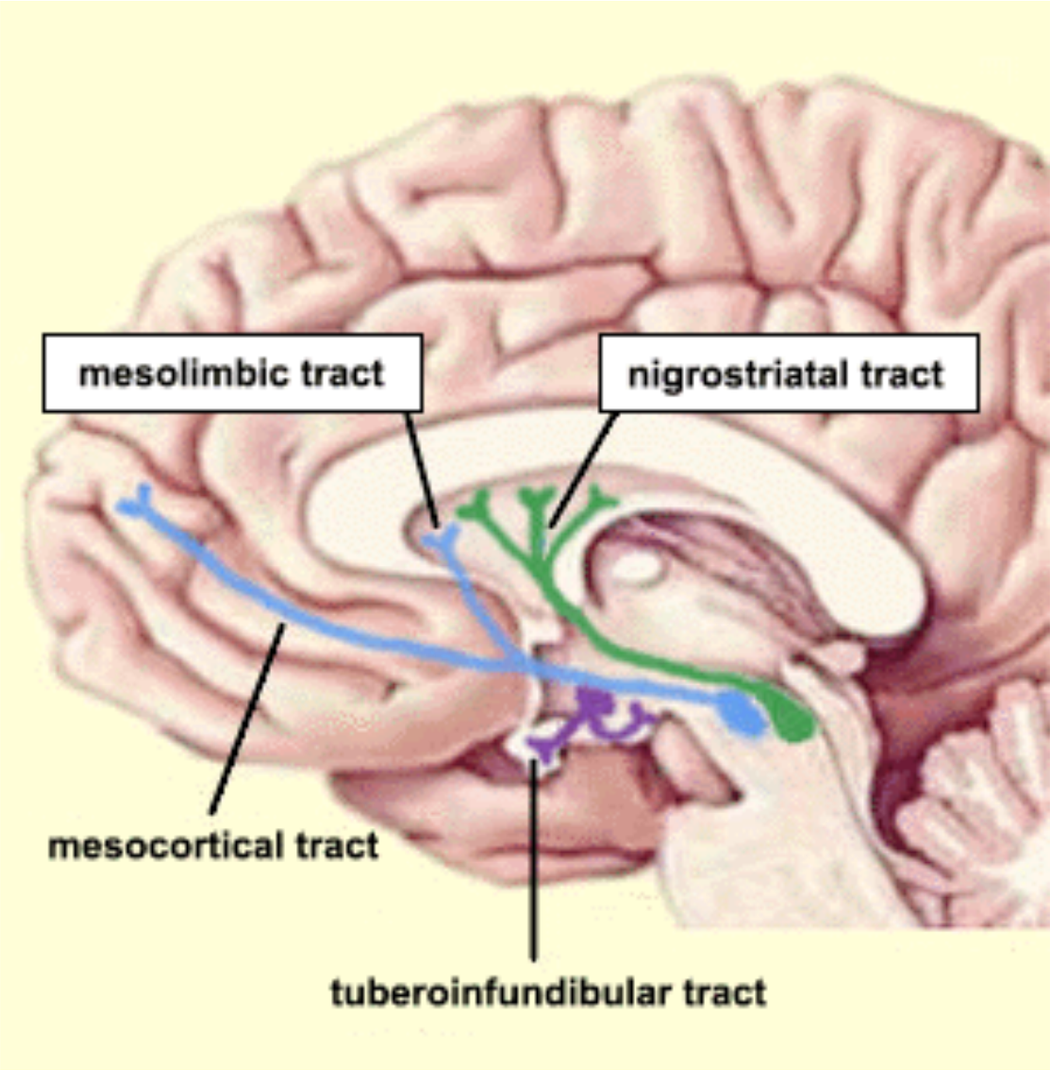
- Pavlovian model-free learning:

$$\mathcal{V}_t(s) = \mathcal{V}_{t-1}(s) + \epsilon(r_t - \mathcal{V}_{t-1}(s))$$

e.g. Rescorla & Wagner 1972

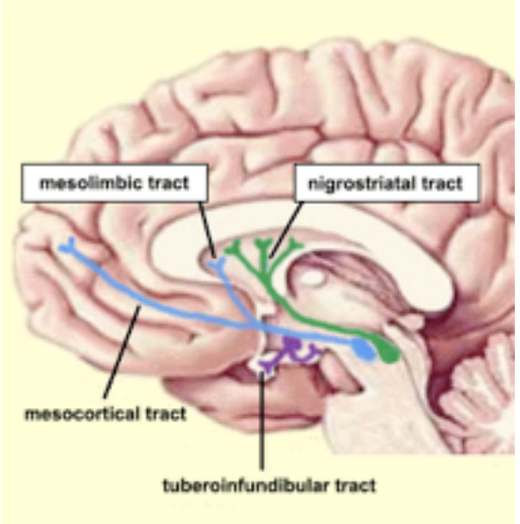
Pavlovian values in the brain?

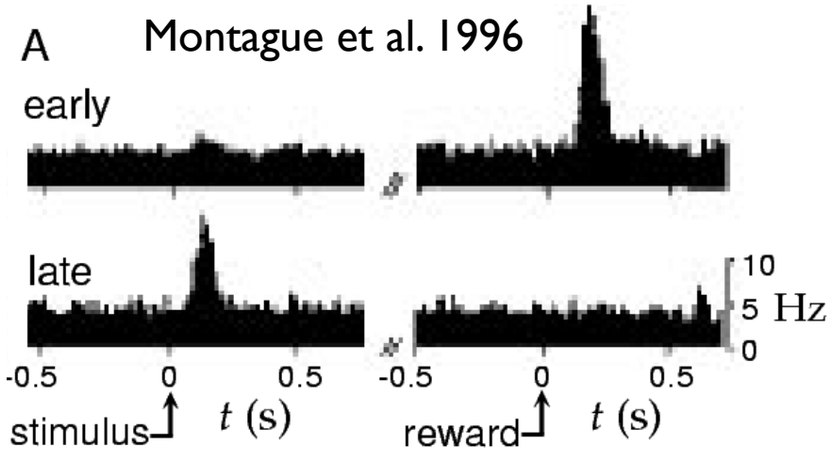
- Where the values V(s) are is not so clear
- Midbrain dopamine neurons report a TD error



Pavlovian values in the brain?

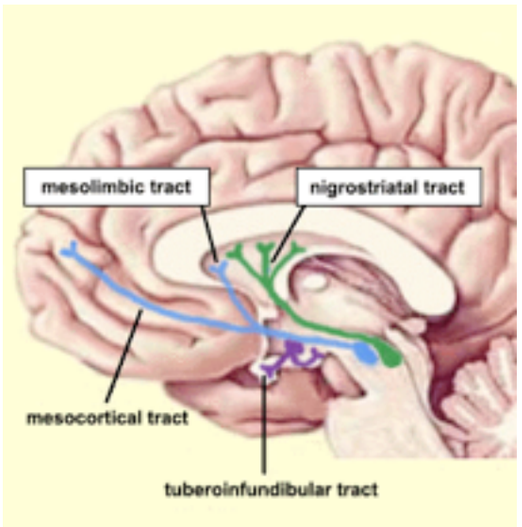
- Where the values V(s) are is not so clear
- Midbrain dopamine neurons report a TD error

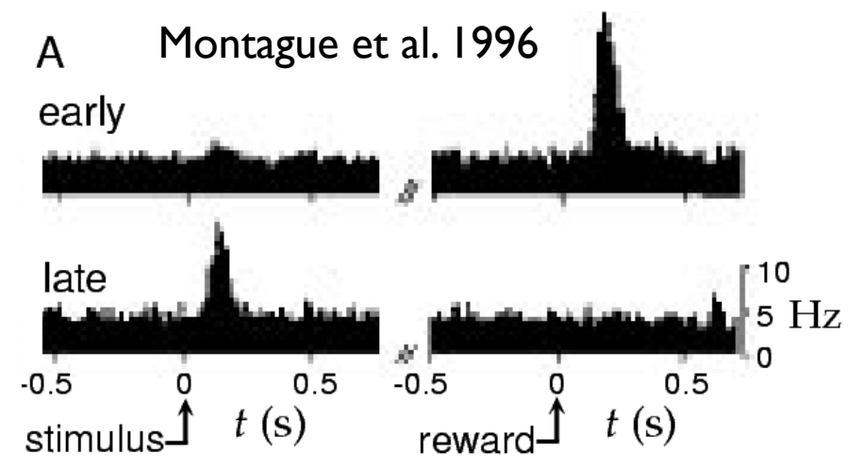


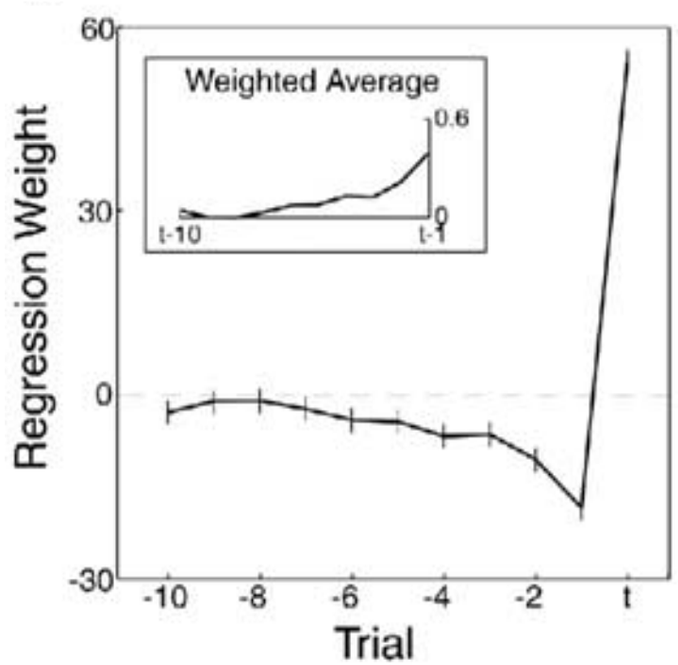


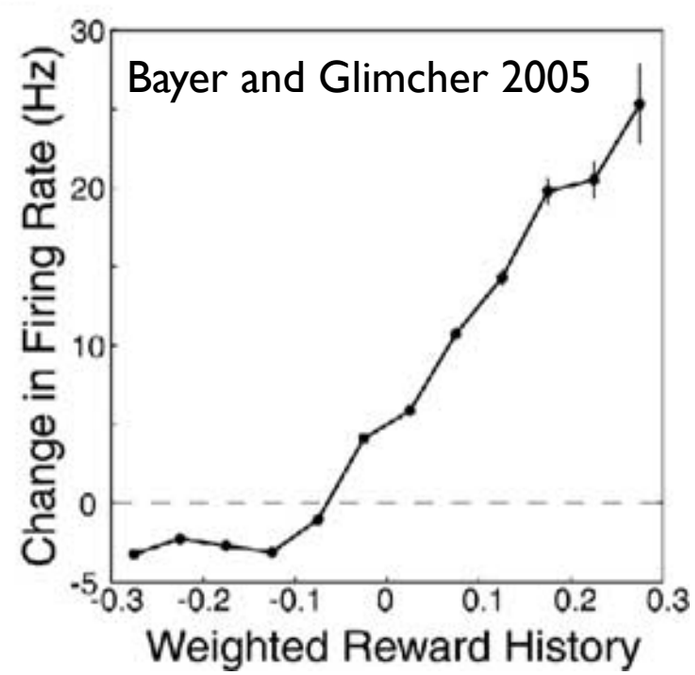
Pavlovian values in the brain?

- Where the values V(s) are is not so clear
- Midbrain dopamine neurons report a TD error

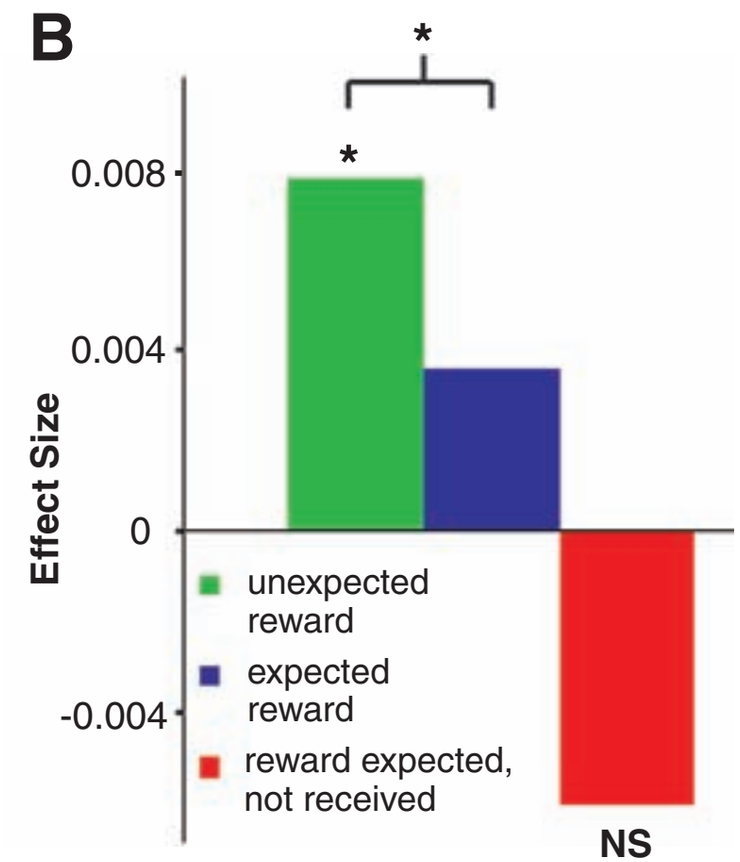




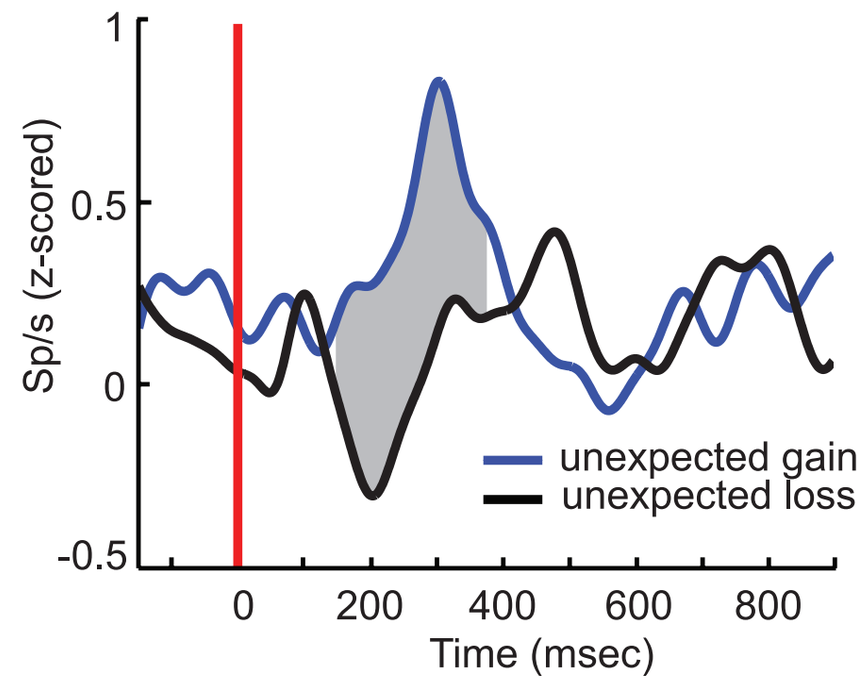




Phasic signals in humans



D'Ardenne et al. 2008 Science

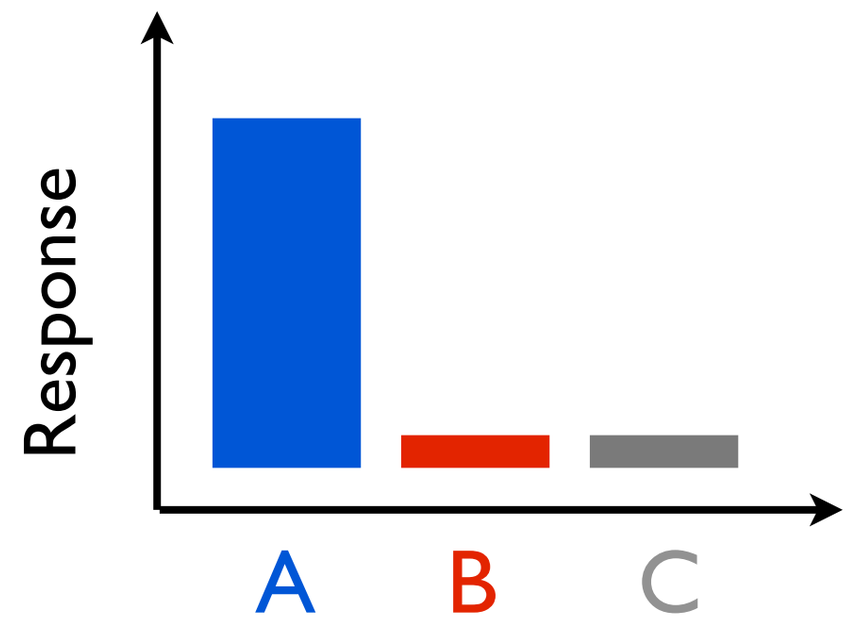


Zaghloul et al. 2009 Science

Blocking

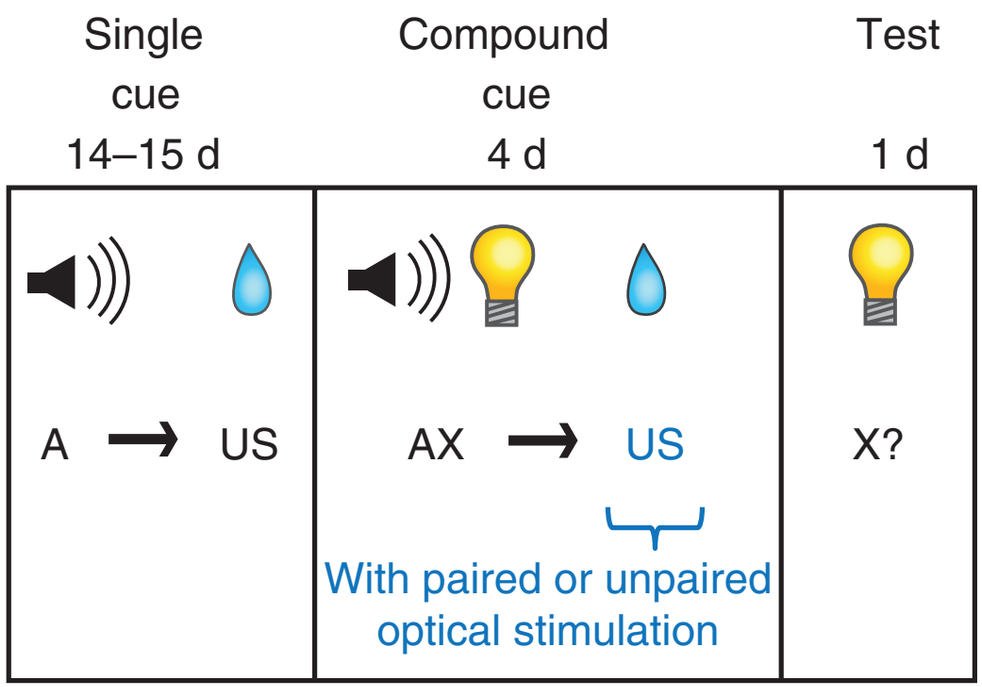
- Are predictions and prediction errors really causally important in learning?

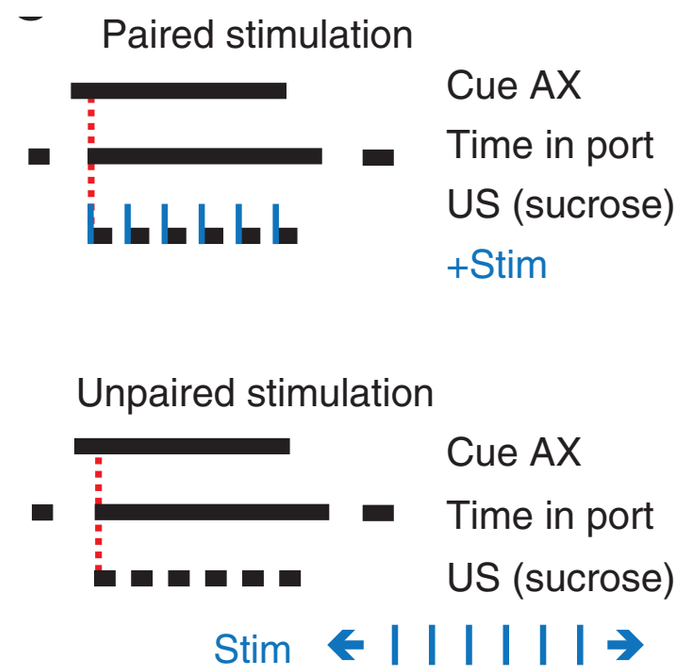
  - 1: A -> Reward
  - 2: A+B -> Reward
  - 3: A -> ? approach
  - B -> ? approach

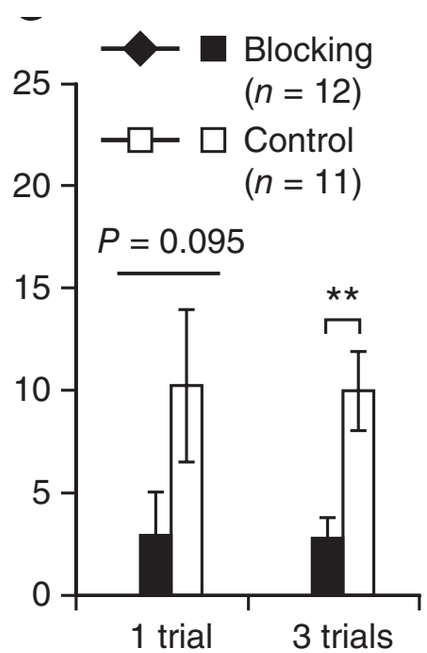


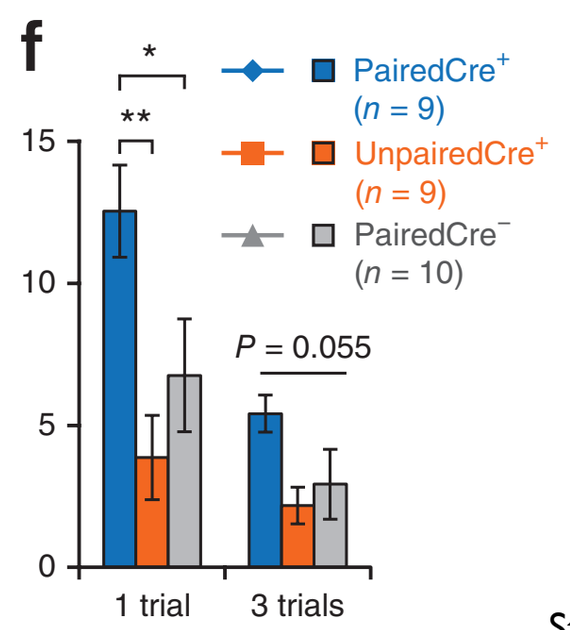
Kamin 1968

Causal role of phasic DA in learning







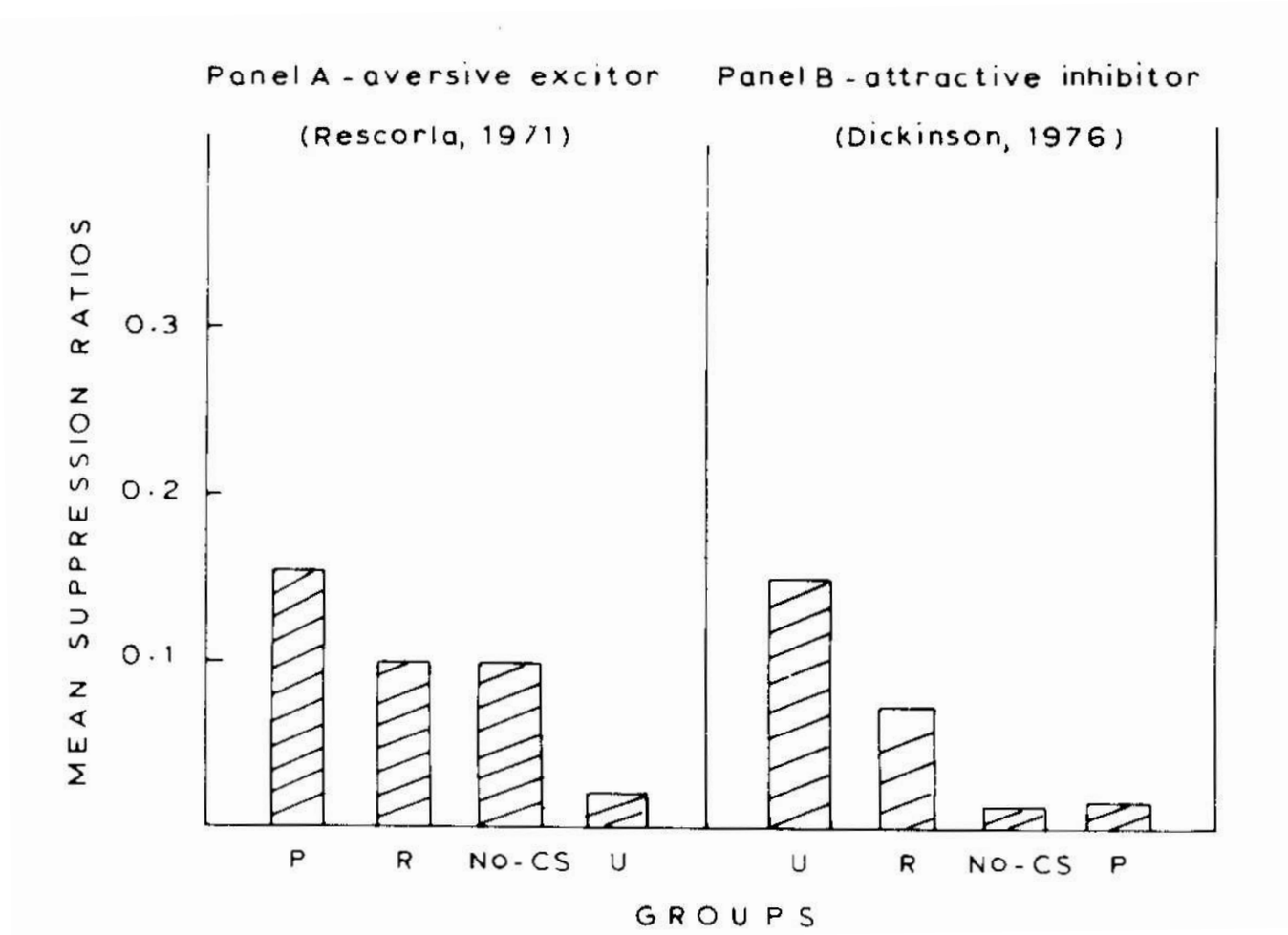


Steinberg et al. 2013 Nat. Neurosci.

Value matters - transreinforcer blocking

TABLE 8.4
Blocking of aversive conditioning

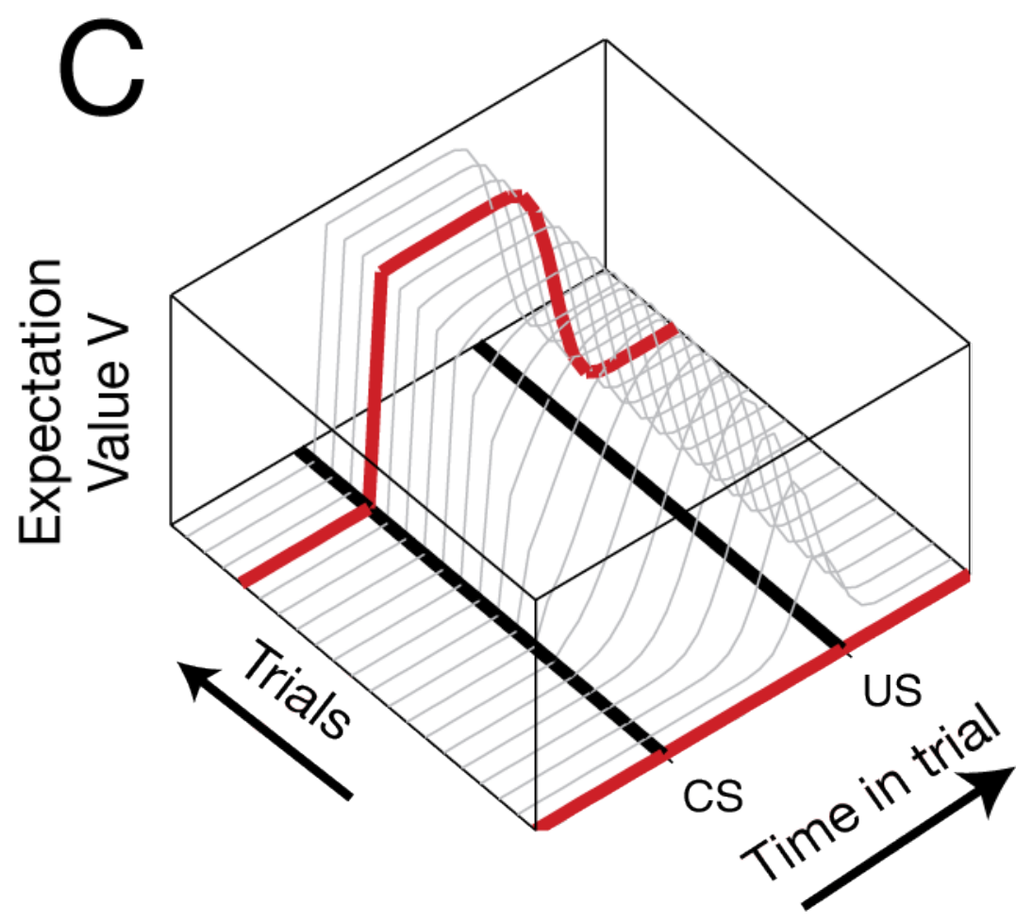
Conditions	*Stage 1*	*Stage 2*	*test*
1	A → shock	AX → shock	X
2	control treatments	AX → shock	X
3	A → food ommission	AX → shock	X



Dickinson & Dearing 1979

Pavlovian conditioning

- By being predictive of an affective outcome, a neutral stimulus can come to elicit the innate preparatory response usually evoked by the affective outcome.



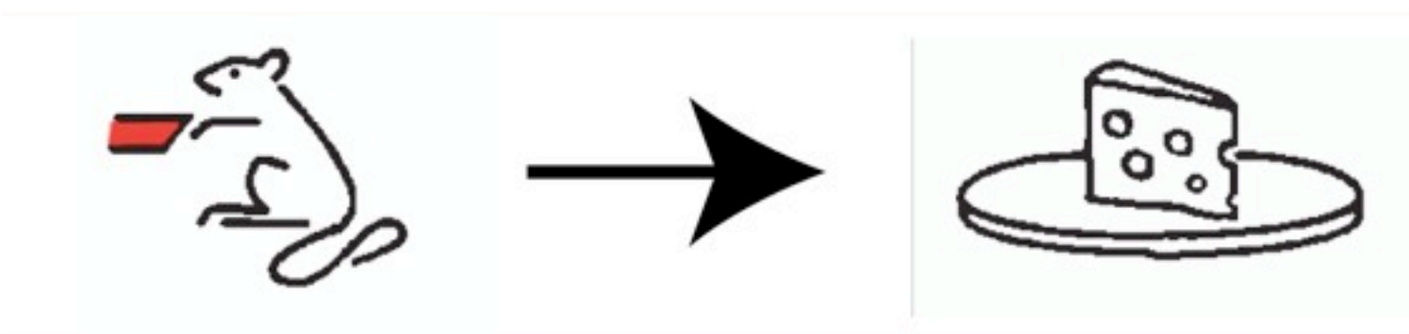
Pavlovian and instrumental conditioning

- Pavlovian model-free learning:

$$\mathcal{V}_t(s) = \mathcal{V}_{t-1}(s) + \epsilon(r_t - \mathcal{V}_{t-1}(s))$$

$$p(a|s, \mathcal{V}) \propto f(a, \mathcal{V}(s))\, p(a|s)$$

- Instrumental model-free learning:

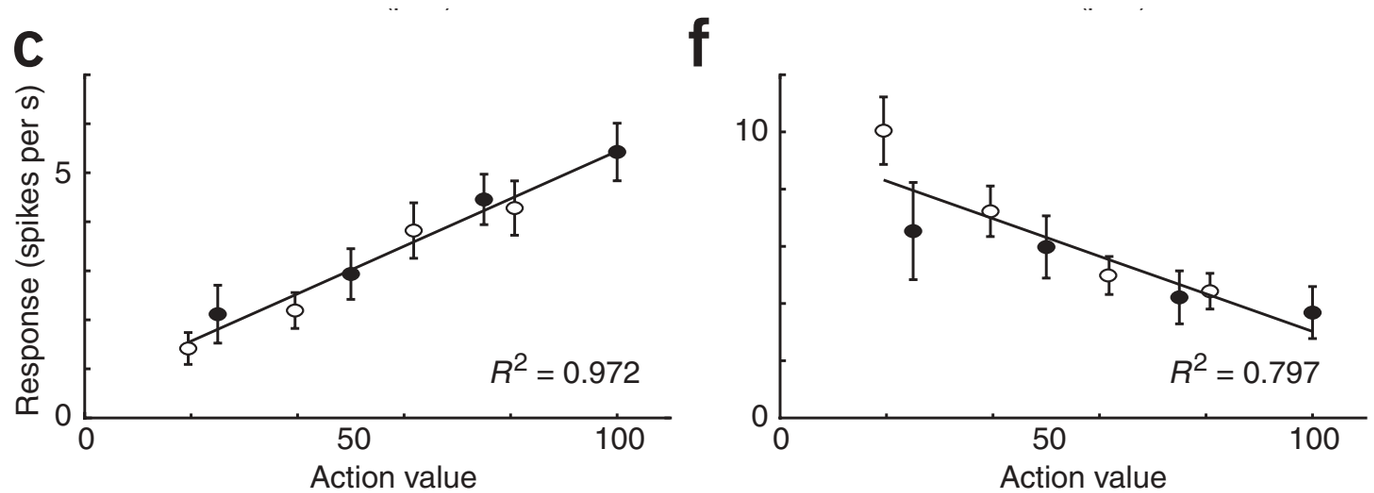


$$\mathcal{Q}_t(a, s) = \mathcal{Q}_{t-1}(a, s) + \epsilon(r_t - \mathcal{Q}_{t-1}(a, s))$$

e.g. Rescorla & Wagner 1972

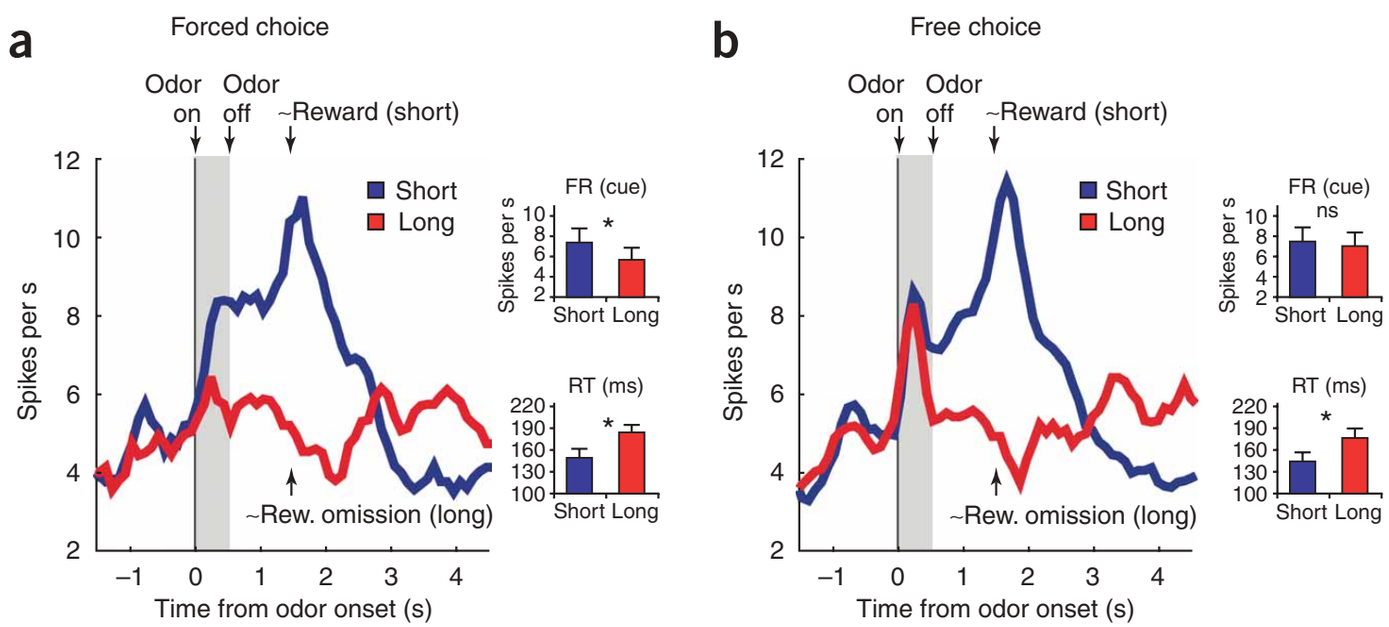
Prediction errors for habits

- SARSA



Morris et al. 2006

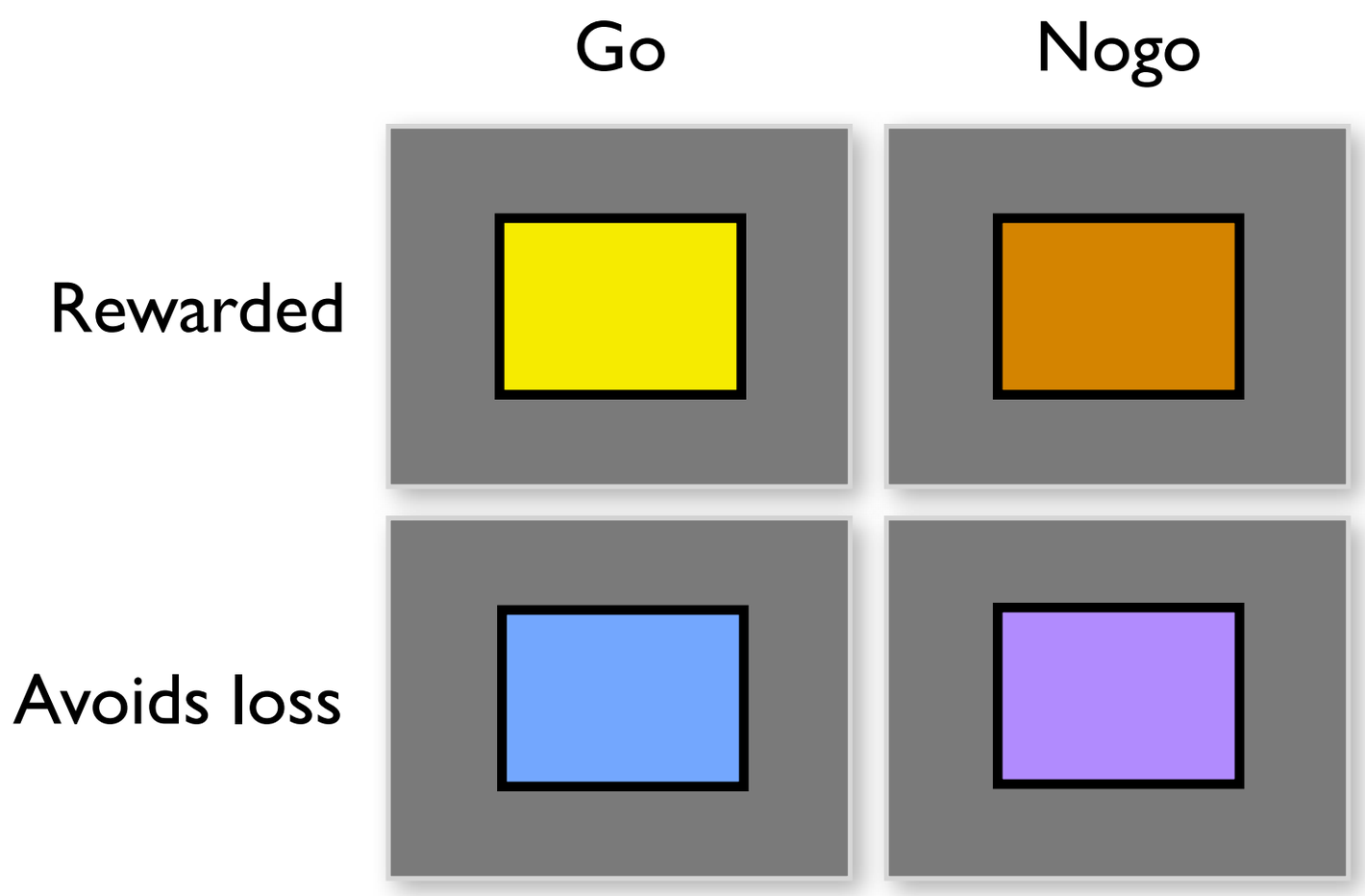
- Q learning



Roesch et al. 2007

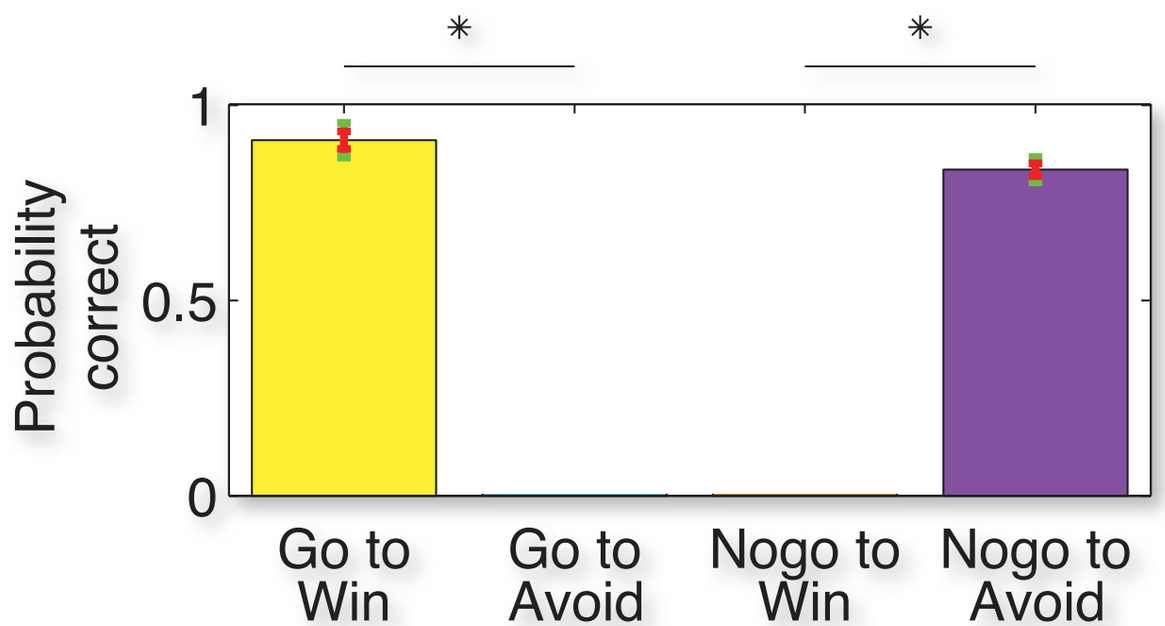
Cached learning ~ prediction errors
prediction errors ~ phasic dopamine
phasic dopamine ~ cached learning

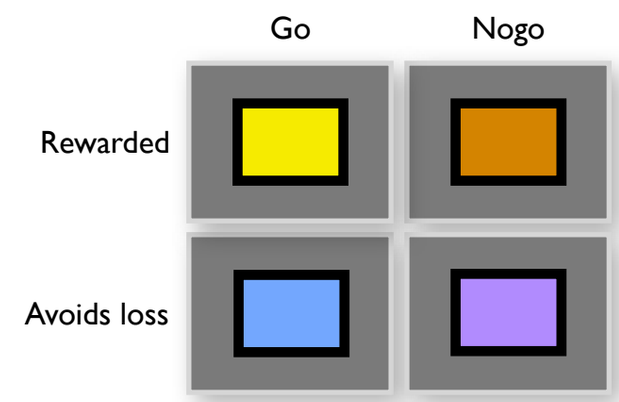
Pavlovian influences on instrumental learning

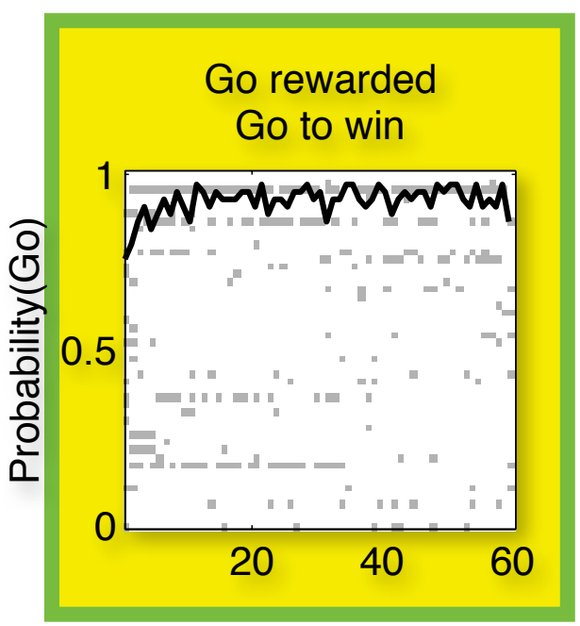


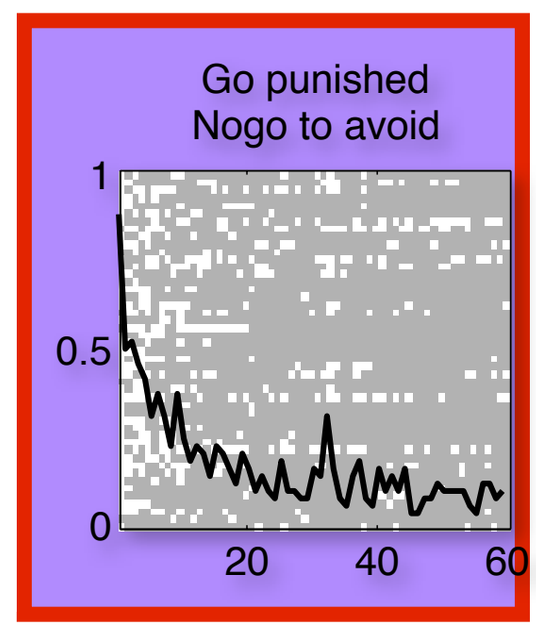
Guitart-Masip, Huys et al. 2011, 2012

Pavlovian influences on instrumental learning



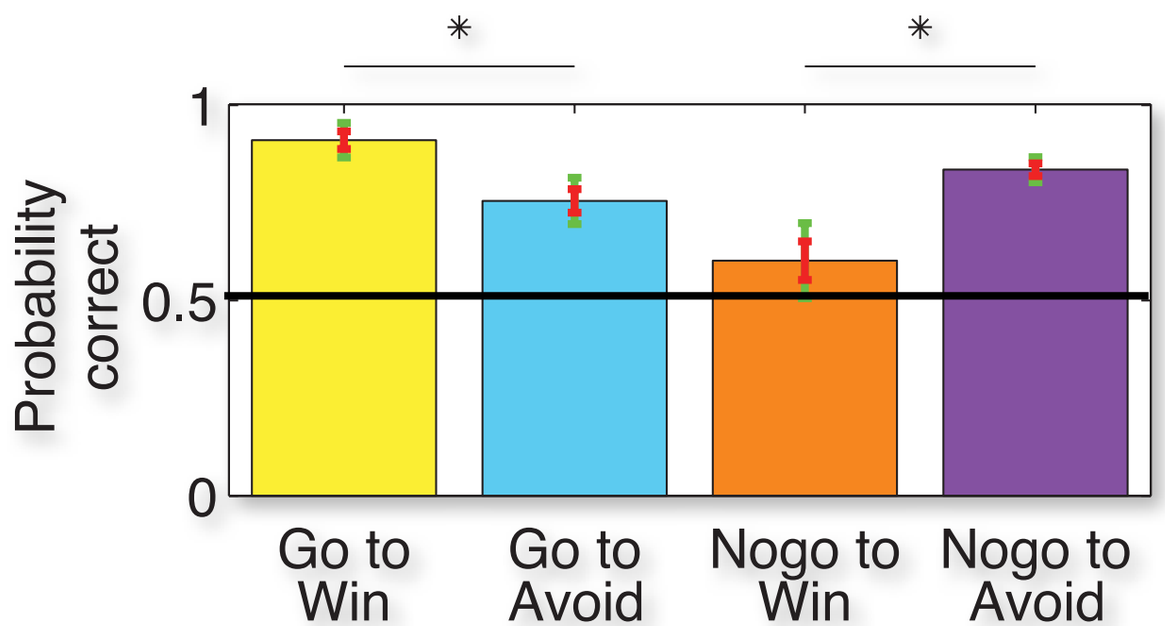


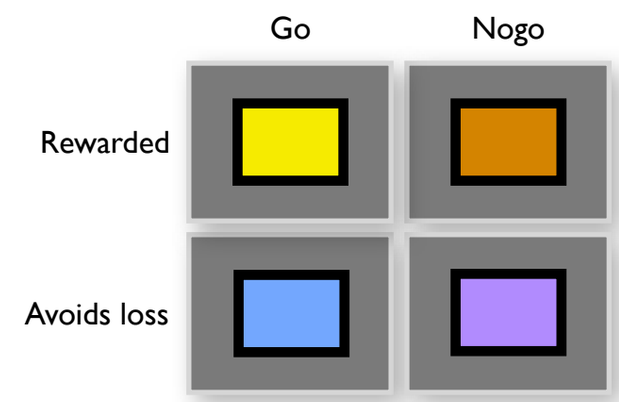


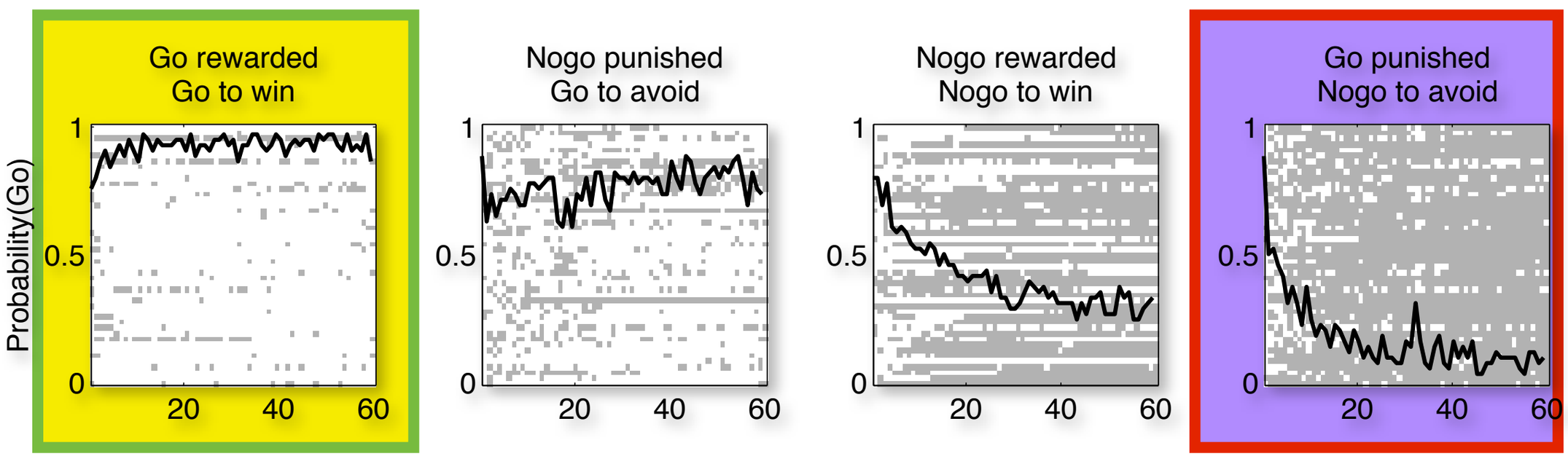


Guitart-Masip, Huys et al. 2011, 2012

Pavlovian influences on instrumental learning

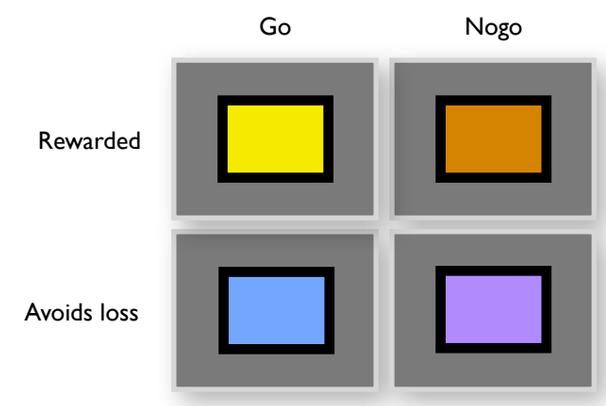


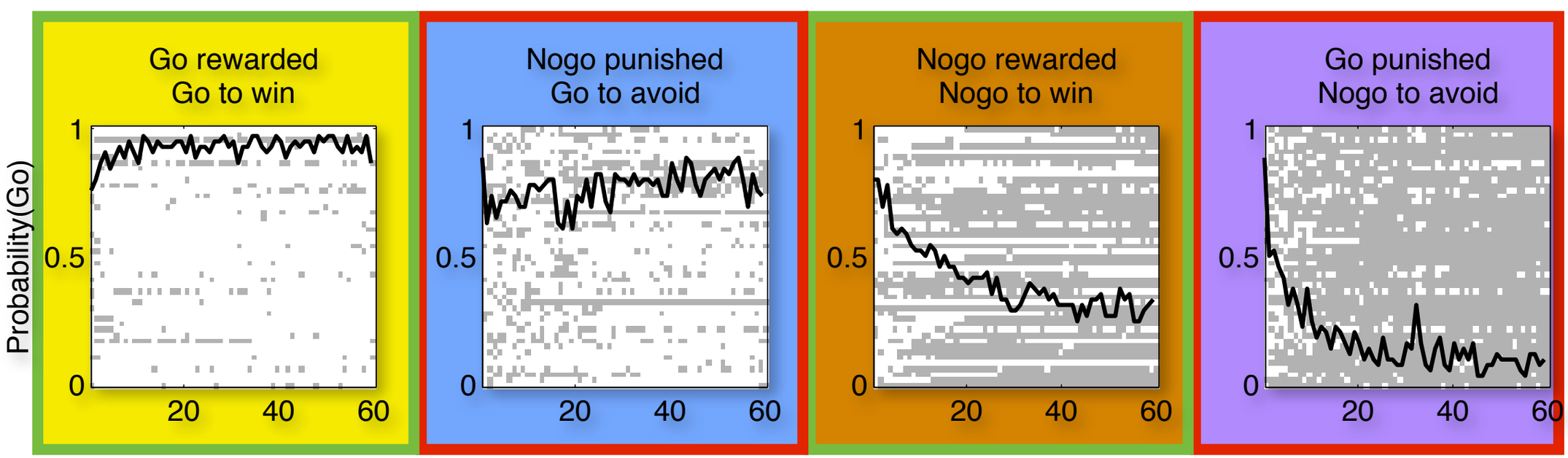




Guitart-Masip, Huys et al. 2011, 2012

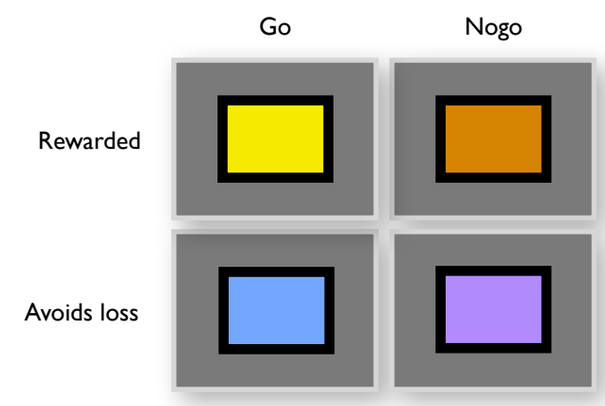
Model



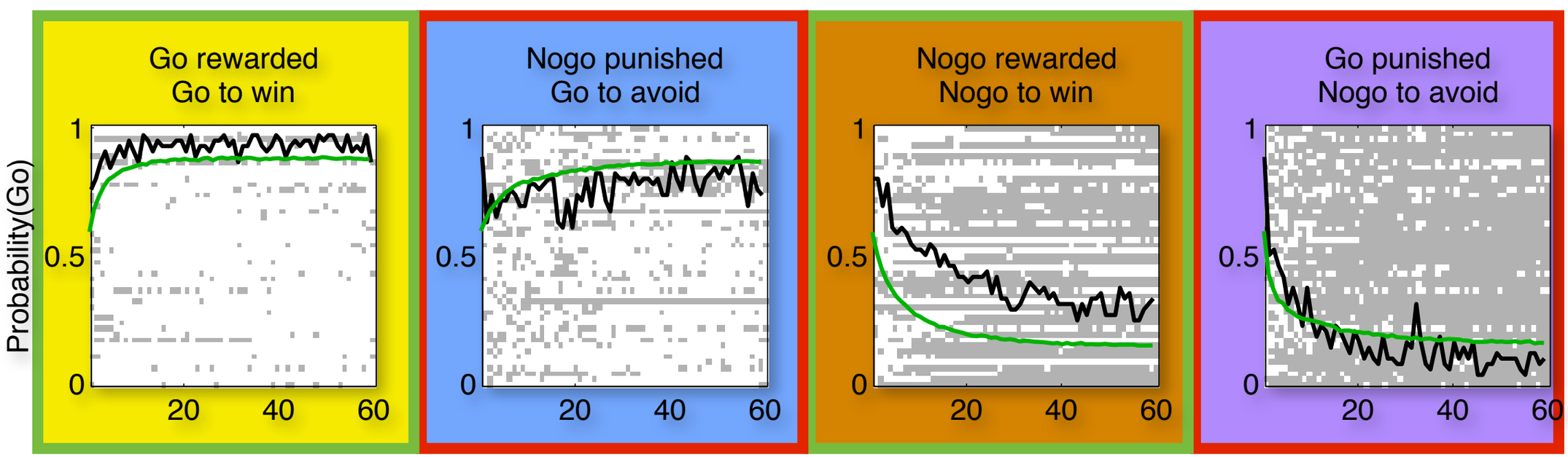


Guitart-Masip et al. 2011, Guitart-Masip, Huys et al. *2012*

Model

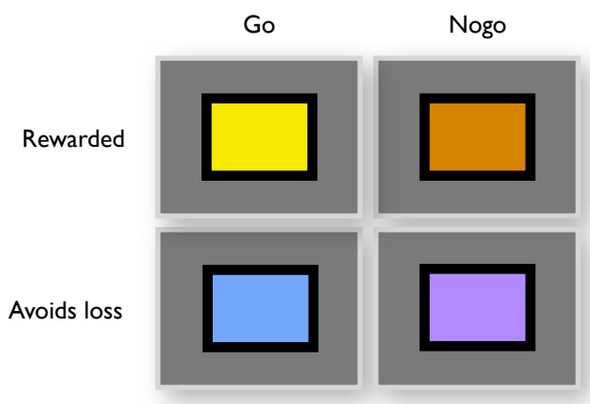


$$p(\text{go}|s_t) \quad \propto \quad \mathcal{Q}_t(\text{go}|s_t) + \text{bias}(\text{go})$$

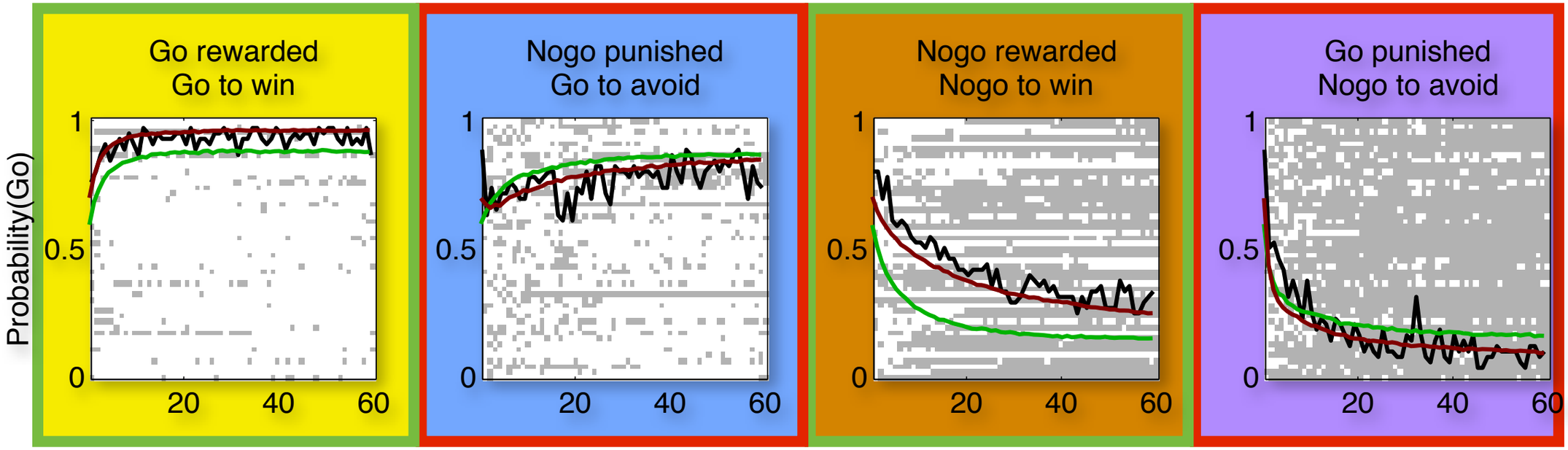


Guitart-Masip et al. 2011, Guitart-Masip, Huys et al. *2012*

Model



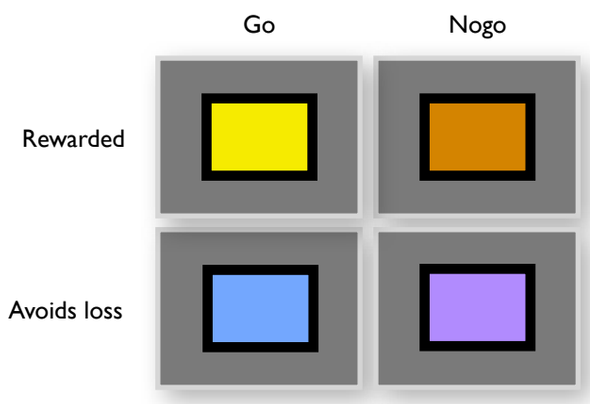
$$
\begin{aligned}
p(\text{go}|s_t) &\propto \mathcal{Q}_t(\text{go}|s_t) + \text{bias}(\text{go}) + \mathcal{V}_t(s_t) \\
\mathcal{V}_t(s_t) &= \mathcal{V}_{t-1}(s_t) + \epsilon(r_t - \mathcal{V}_{t-1}(s_t))
\end{aligned}
$$



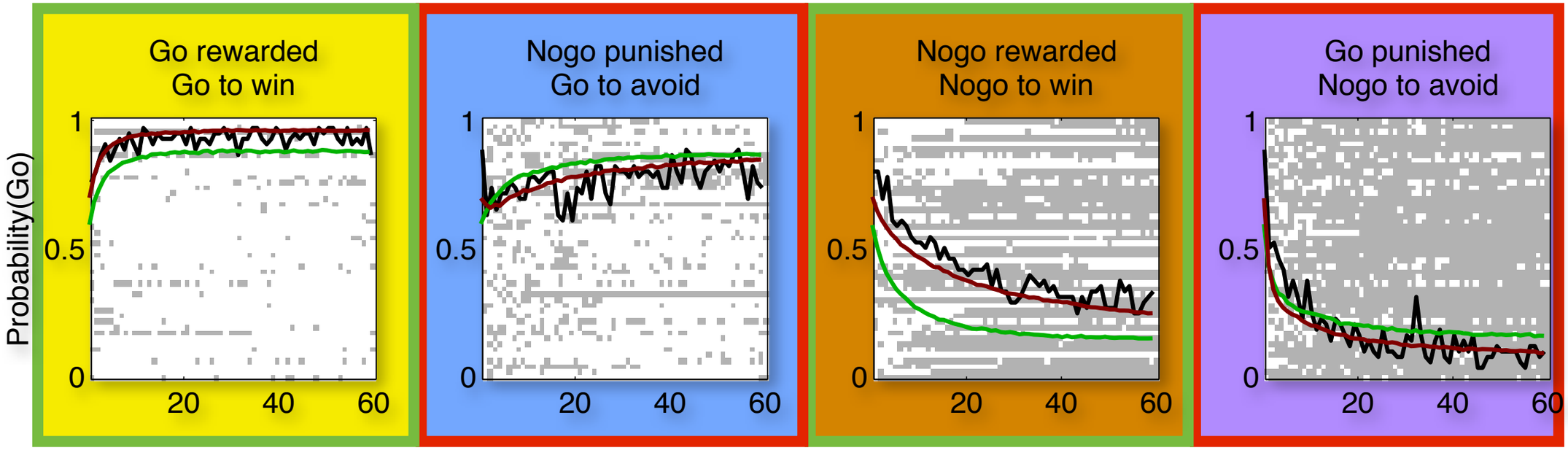
Guitart-Masip et al. 2011, Guitart-Masip, Huys et al. *2012*

Model

P(go) $\propto$ value of stimulus



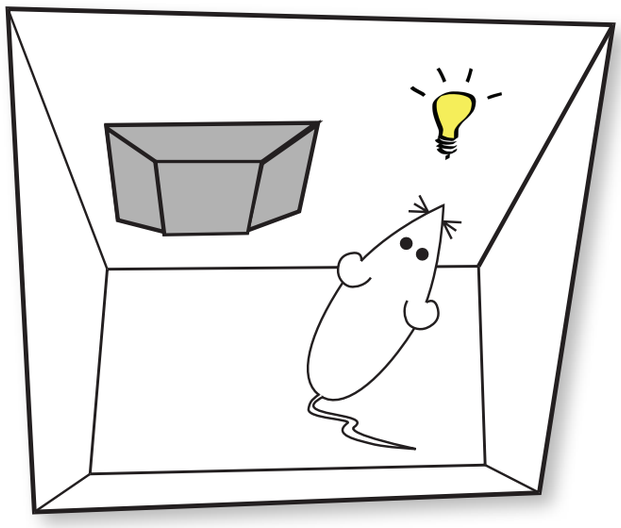
$$
\begin{aligned}
p(\text{go}|s_t) &\propto \mathcal{Q}_t(\text{go}|s_t) + \text{bias}(\text{go}) + \mathcal{V}_t(s_t) \\
\mathcal{V}_t(s_t) &= \mathcal{V}_{t-1}(s_t) + \epsilon(r_t - \mathcal{V}_{t-1}(s_t))
\end{aligned}
$$



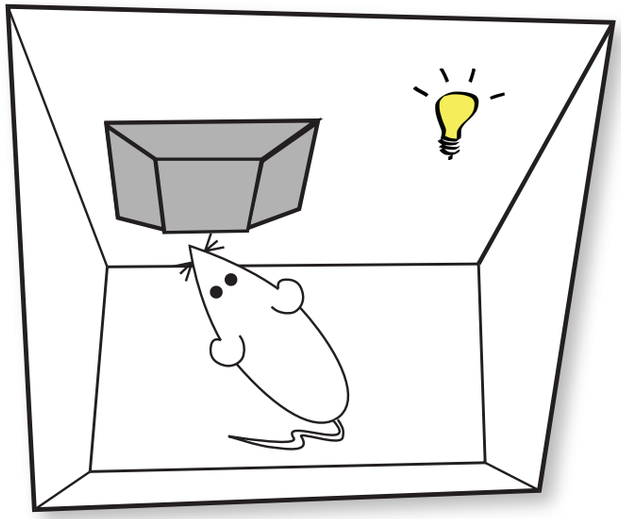
Guitart-Masip et al. 2011, Guitart-Masip, Huys et al. *2012*

Addictive Pavlovian values

A



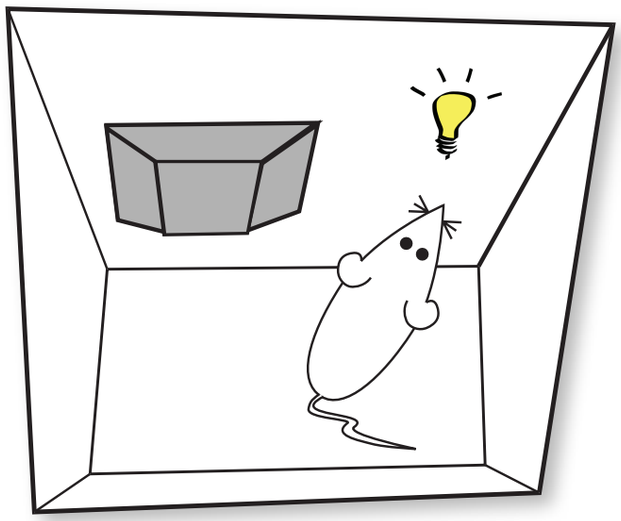
D



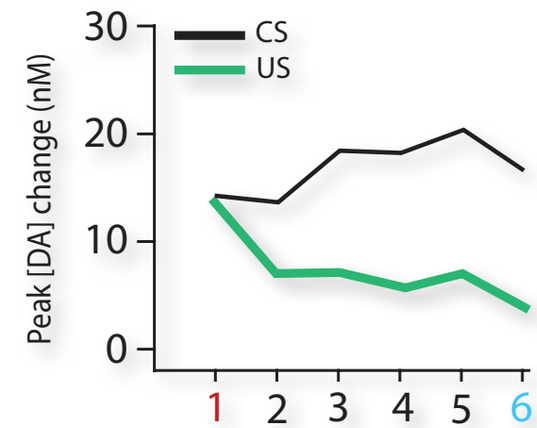
Huys et al. 2012 after Flagel et al. 2009, 2011

Addictive Pavlovian values

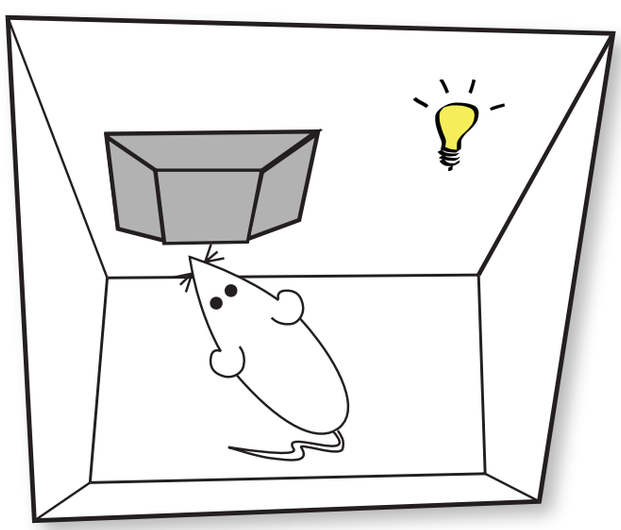
A



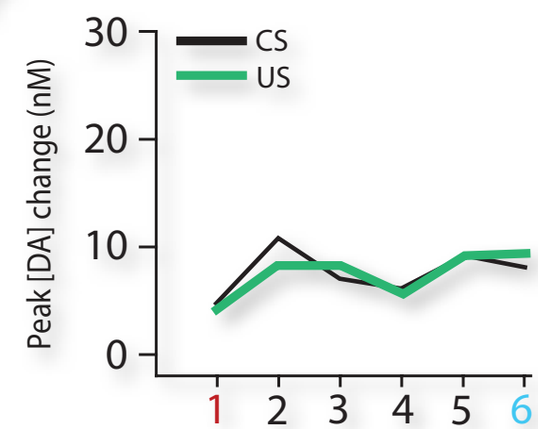
C



D



F

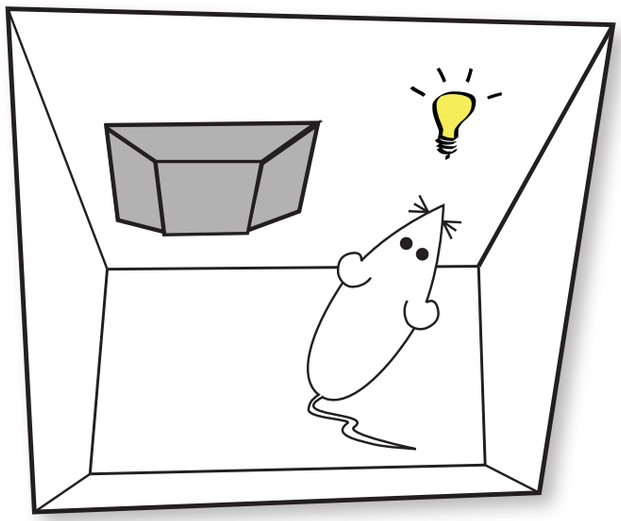


Huys et al. 2012 after Flagel et al. 2009, 2011

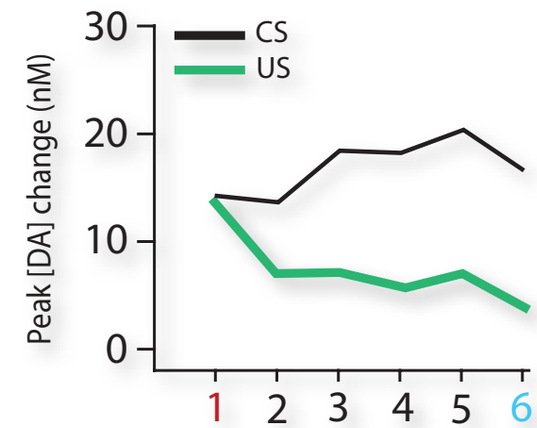
Addictive Pavlovian values

Pavlovian state value acquisition

A

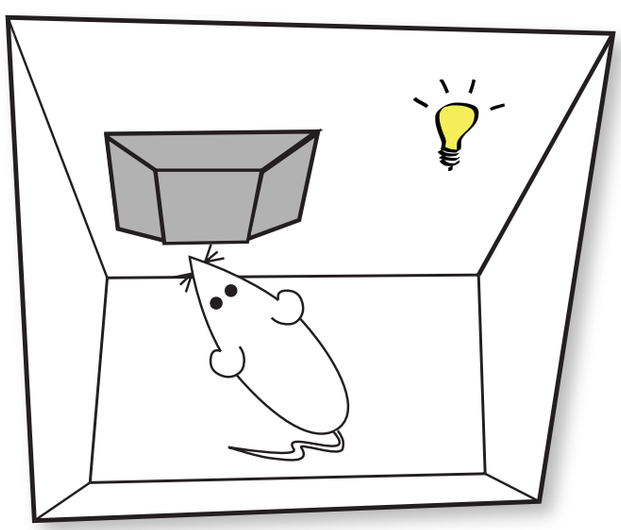


C

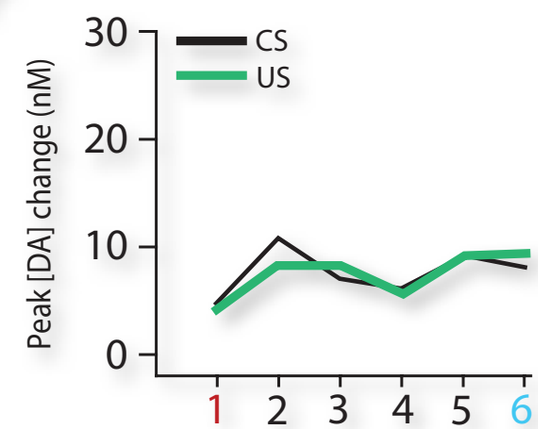


No Pavlovian state learning

D



F

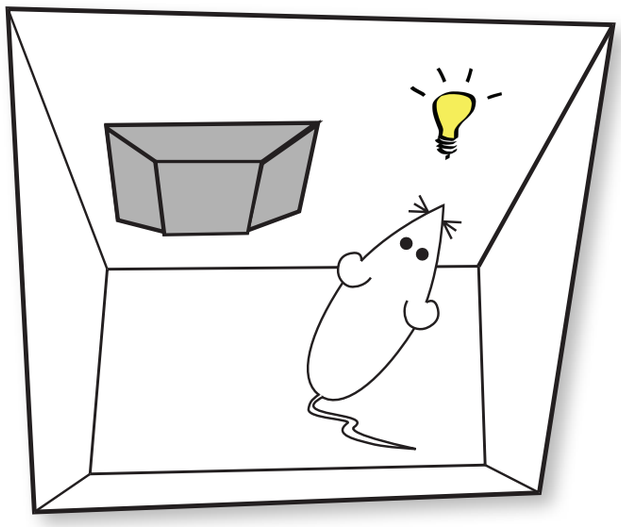


Huys et al. 2012 after Flagel et al. 2009, 2011

Addictive Pavlovian values

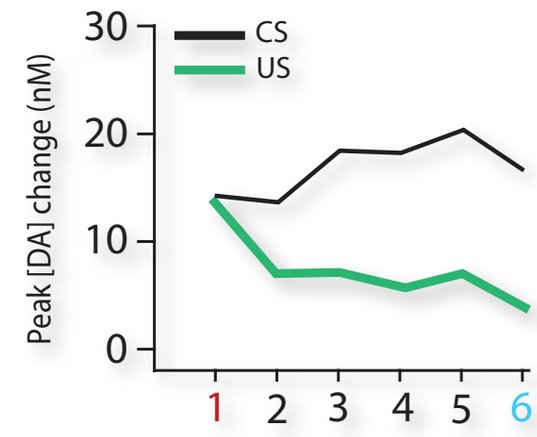
Pavlovian state value acquisition

A



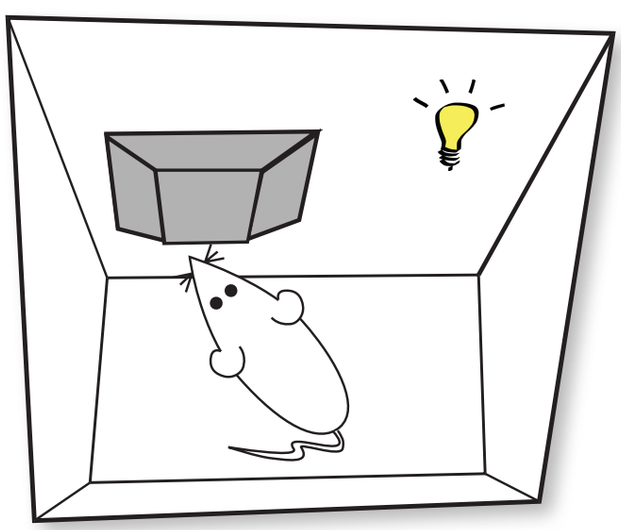
Impulsive
Amphetamine
DA antagonism
D2R<->phasic DA

C

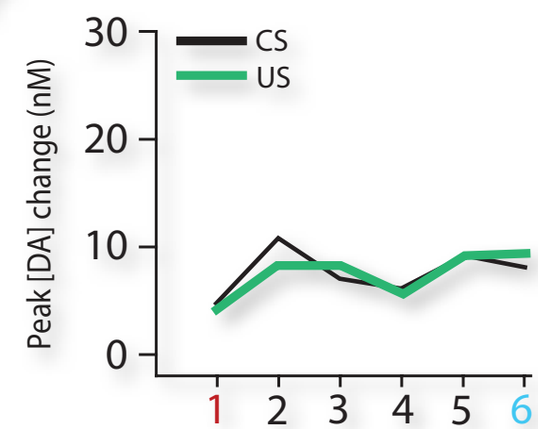


No Pavlovian state learning

D



F

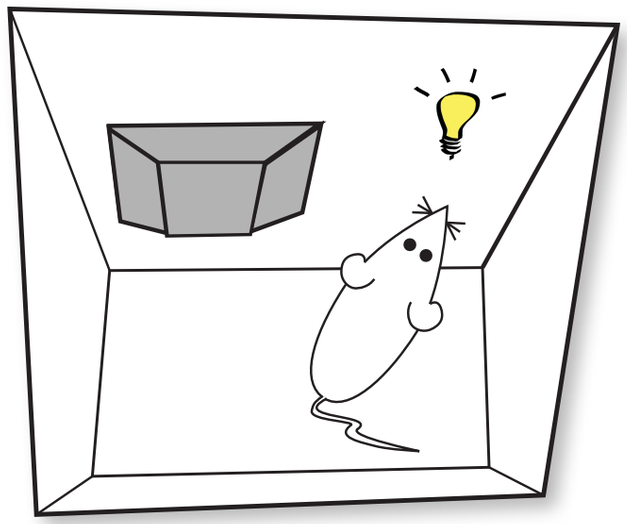


Huys et al. 2012 after Flagel et al. 2009, 2011

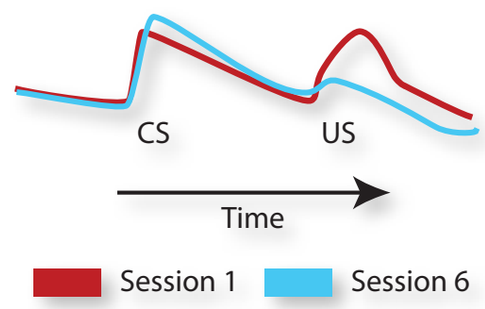
Pavlovian state values: sign tracking

Sign trackers

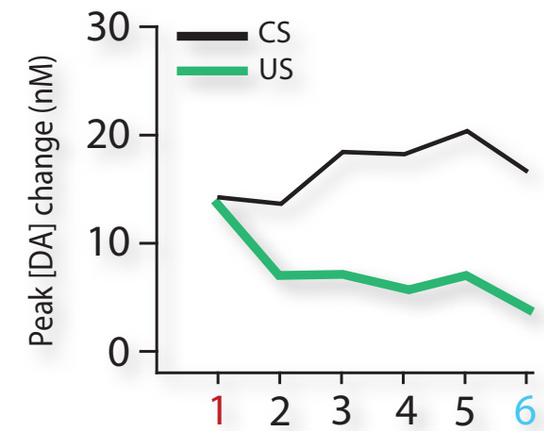
A



B

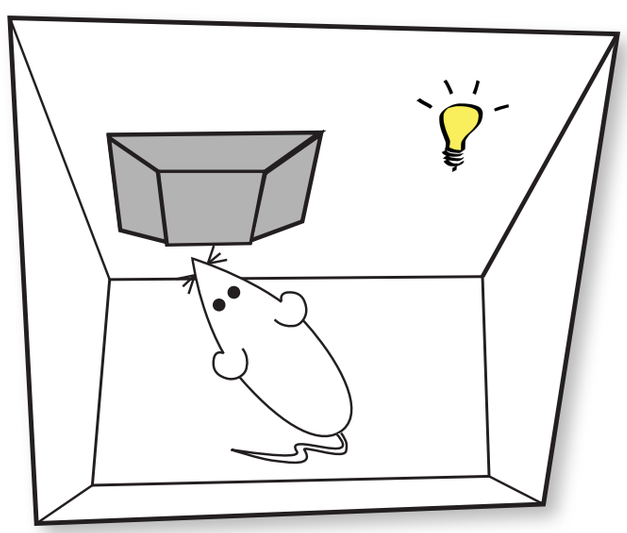


C

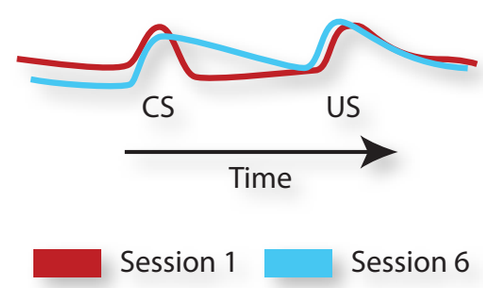


Goal trackers

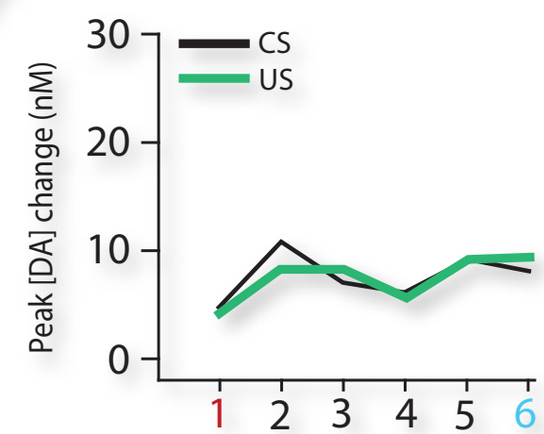
D



E

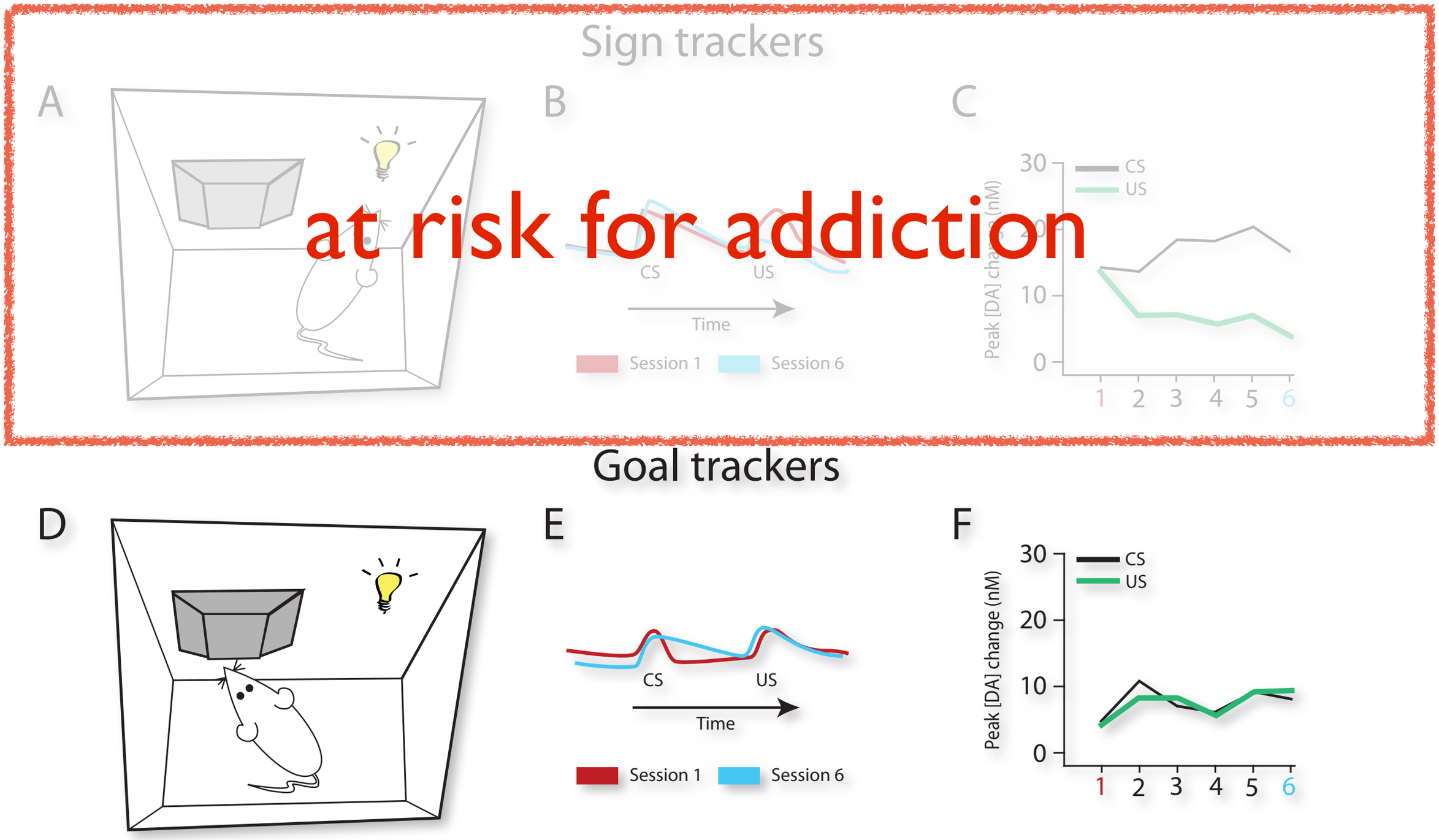


F



Flagel et al. 2011, Huys et al. 2011a

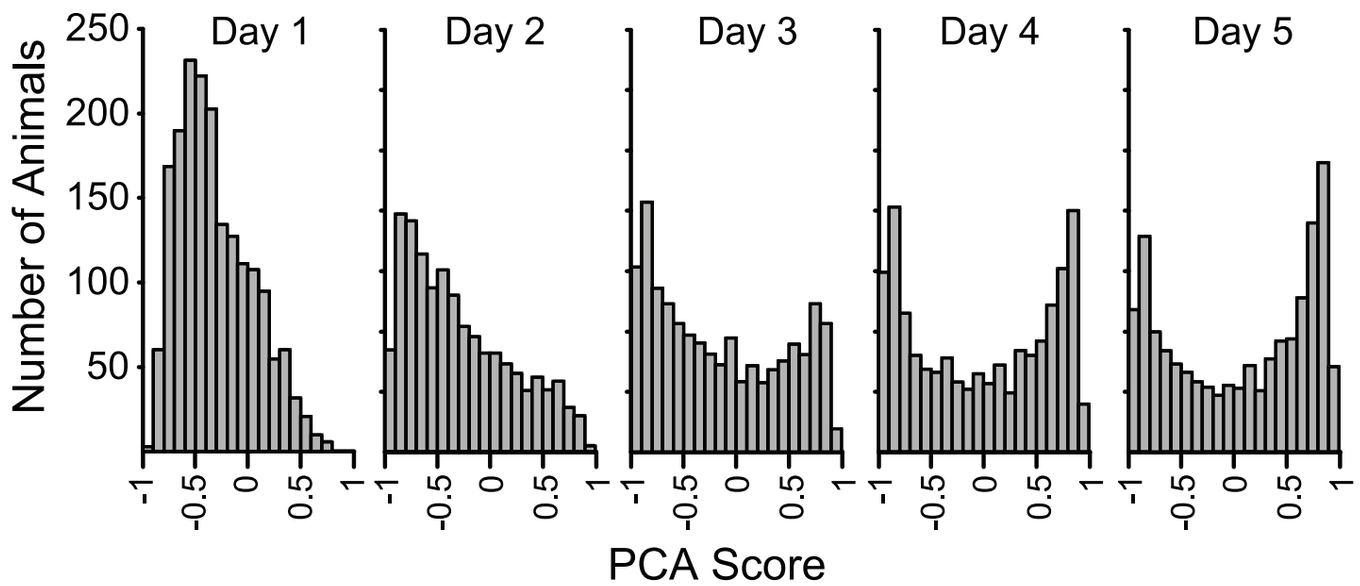
Pavlovian state values: sign tracking

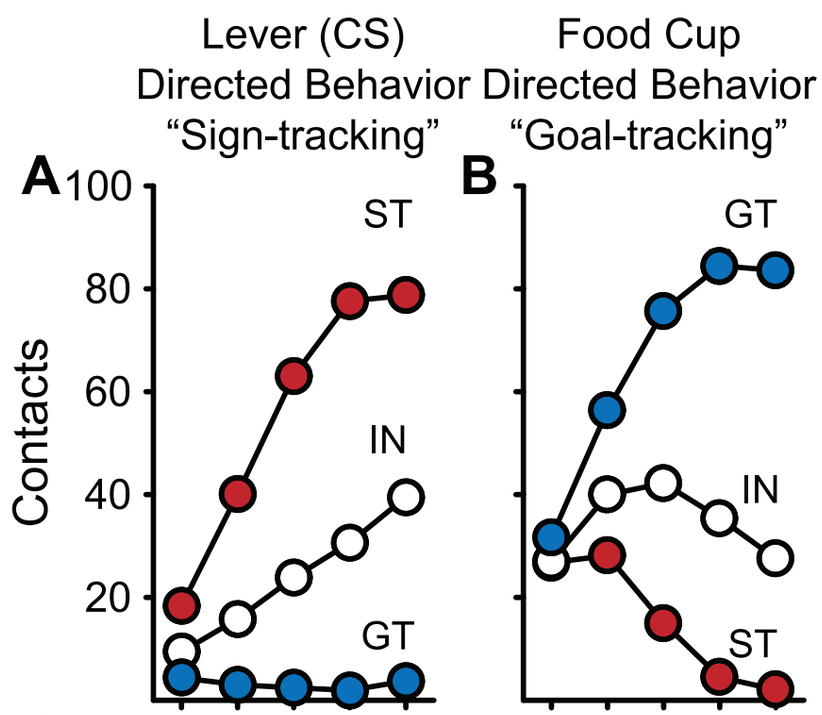


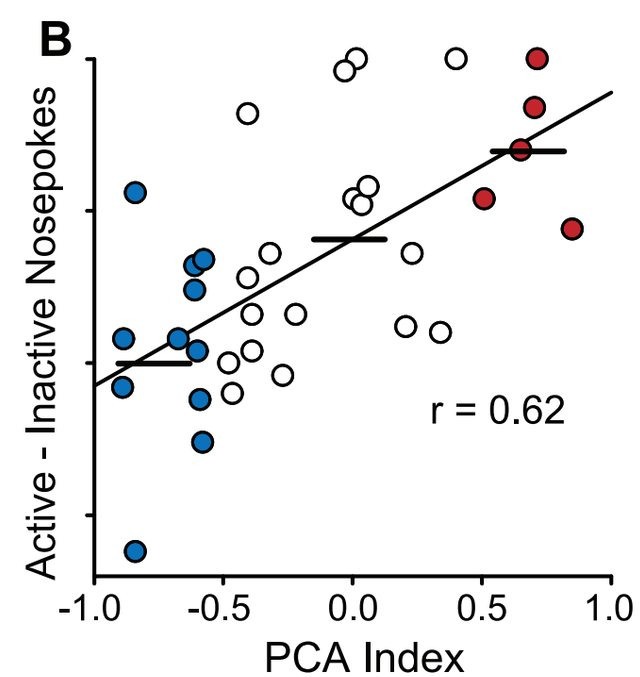
at risk for addiction

Flagel et al. 2011, Huys et al. 2011a

Sign-tracking vs goal-tracking

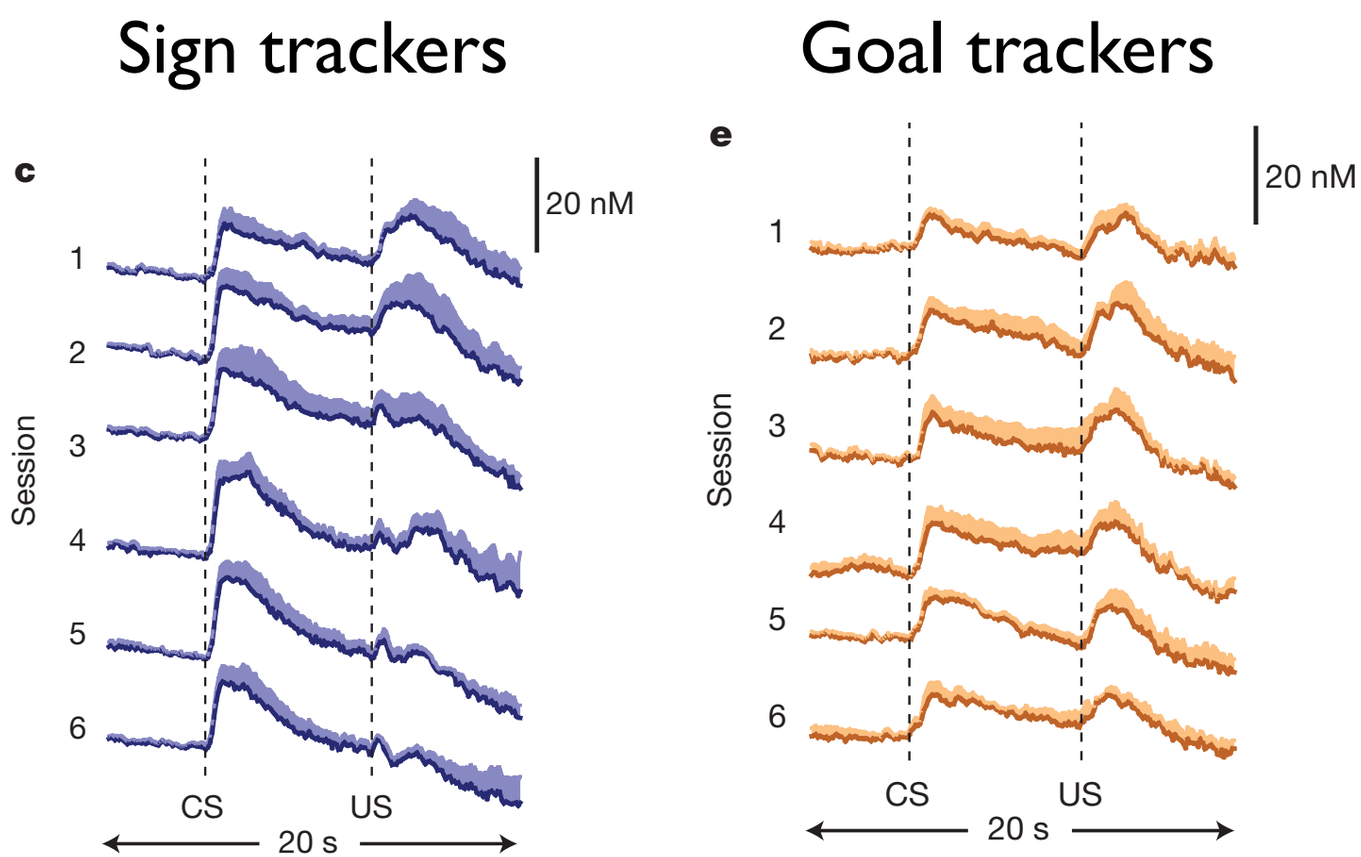






Meyer et al. 2012

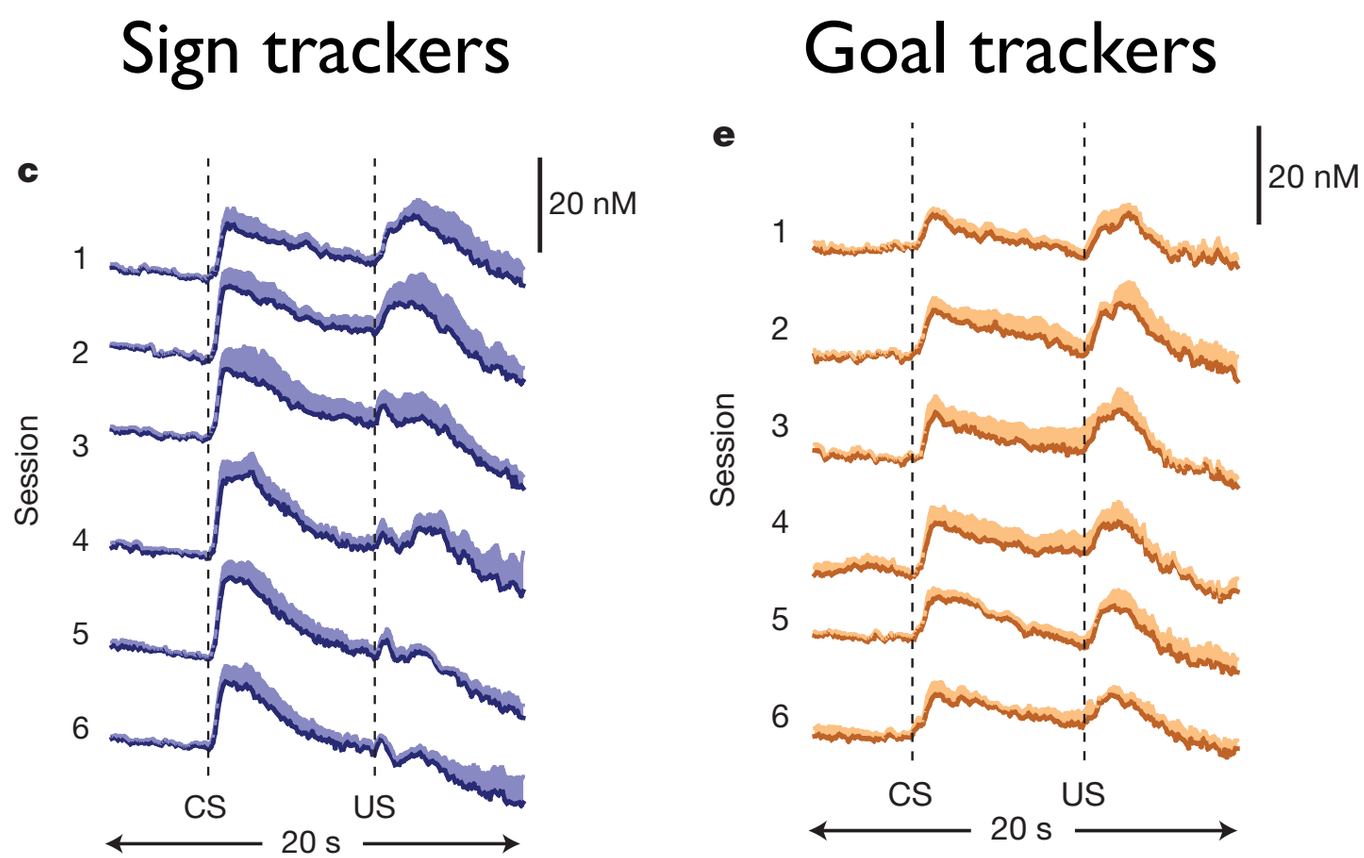
Absent model?



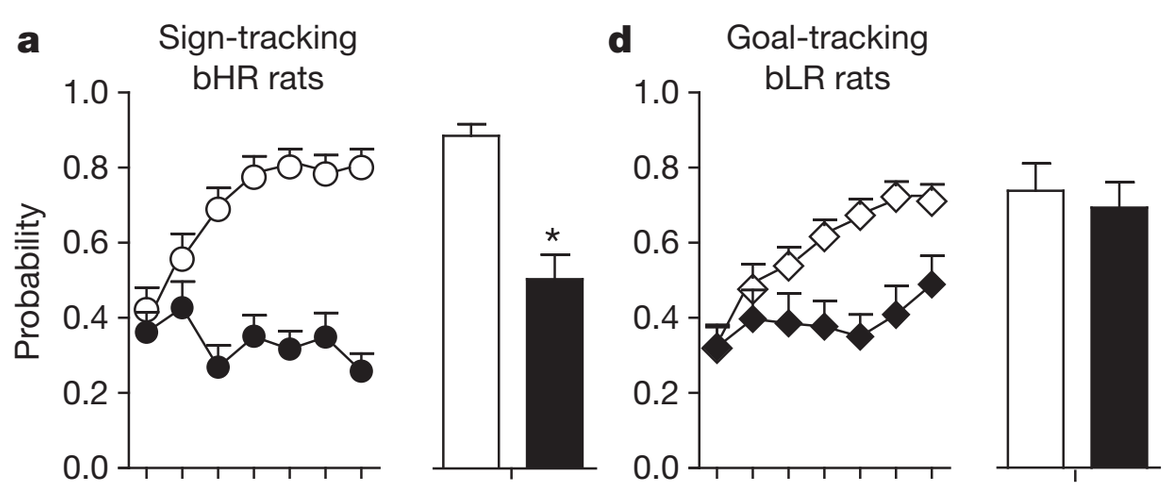
$$\delta = r - Q$$

Flagel et al. 2011, Nature

Absent model?

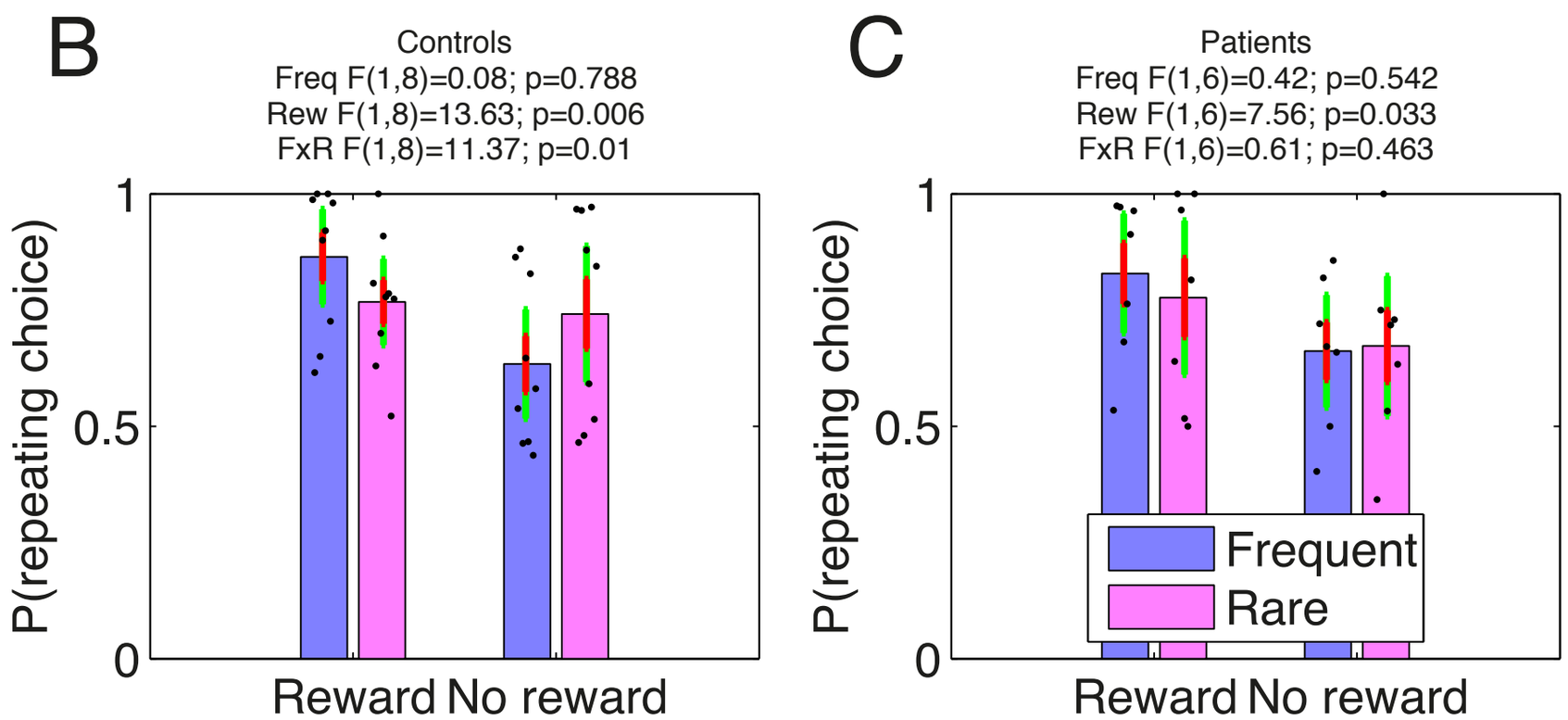


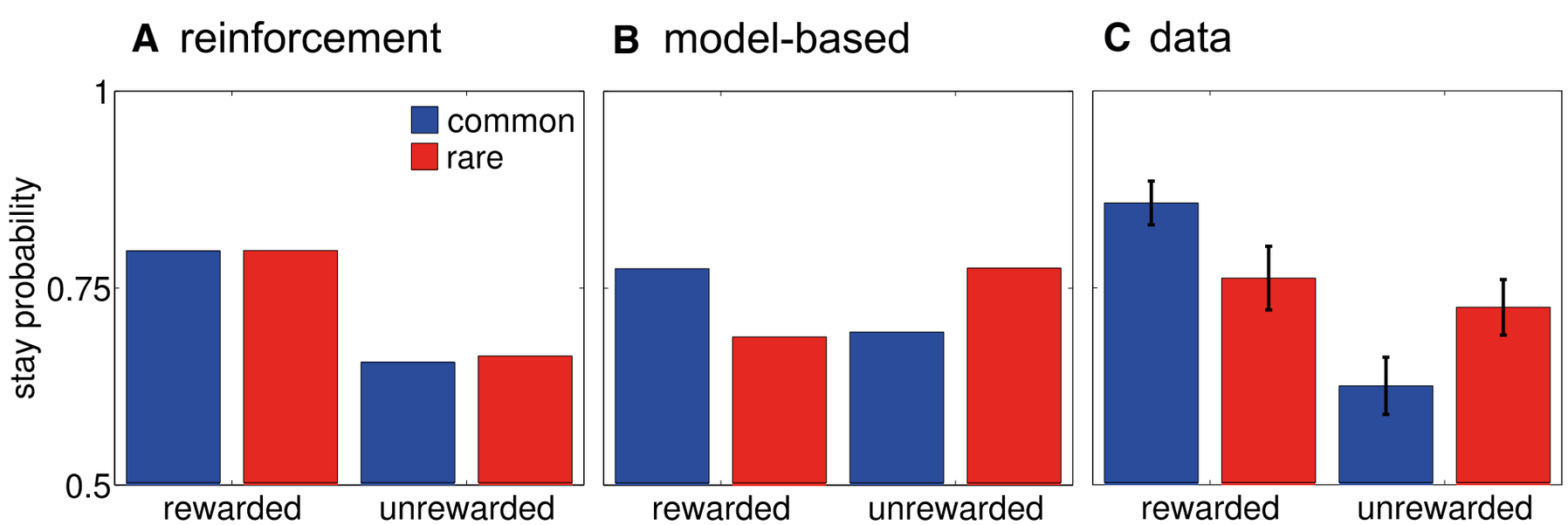
$$\delta = r - Q$$



Flagel et al. 2011, Nature

Shift towards model-free in addiction





Conclusion I

- Long-term rewards have internal consistency
- This can be exploited for solution
- Exploration and exploitation trade off when sampling
- Clever use of samples can produce fast learning
  - Brain most likely does something like this
- Model-free learning relies on phasic dopamine signals
  - iterative learning from experience
  - individual variability & addiction