Reinforcement learning: an overview

The big picture

Model free | **Model based**

reward

$$Q(s,a) = \sum_{s'} p(s' \mid s, a) r(s')$$

value of (state, action)

probability of next states

Model free example: learning the mean with prediction errors

new prediction

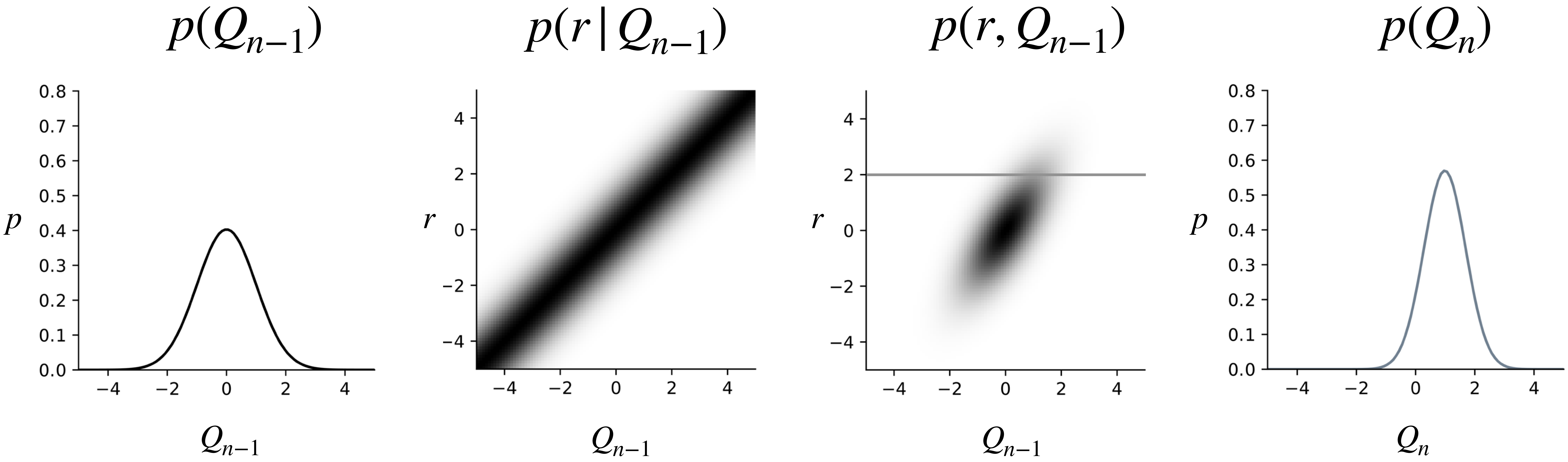
learning rate

$$Q_n = Q_{n-1} + \frac{1}{n}(r_n - Q_{n-1})$$

old prediction

prediction error

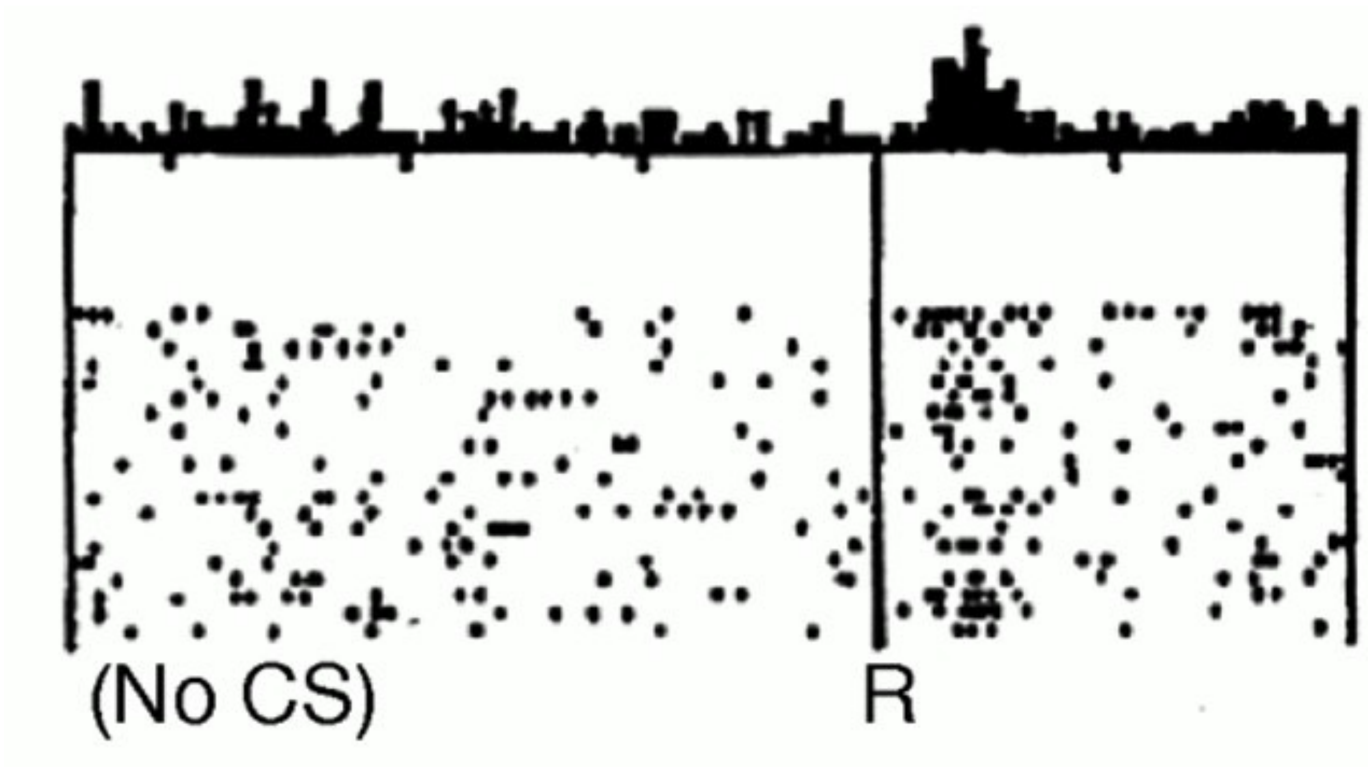
Want to make it Bayesian and talk precision?



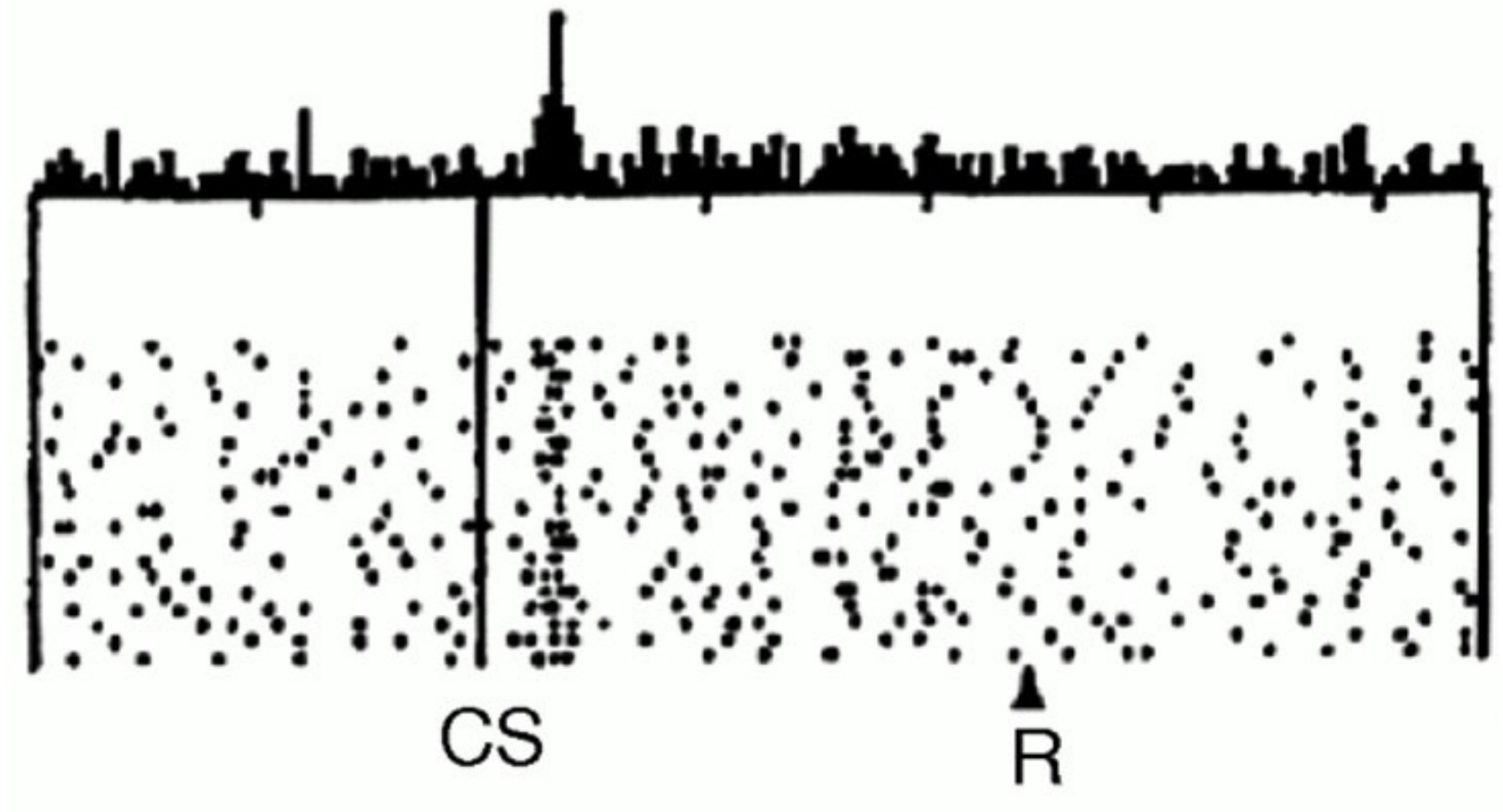
$$Q_n = Q_{n-1} + \frac{\sigma_Q}{\sigma_Q + \sigma_r}(r_n - Q_{n-1})$$

Do dopamine neurons report errors in the prediction of reward?

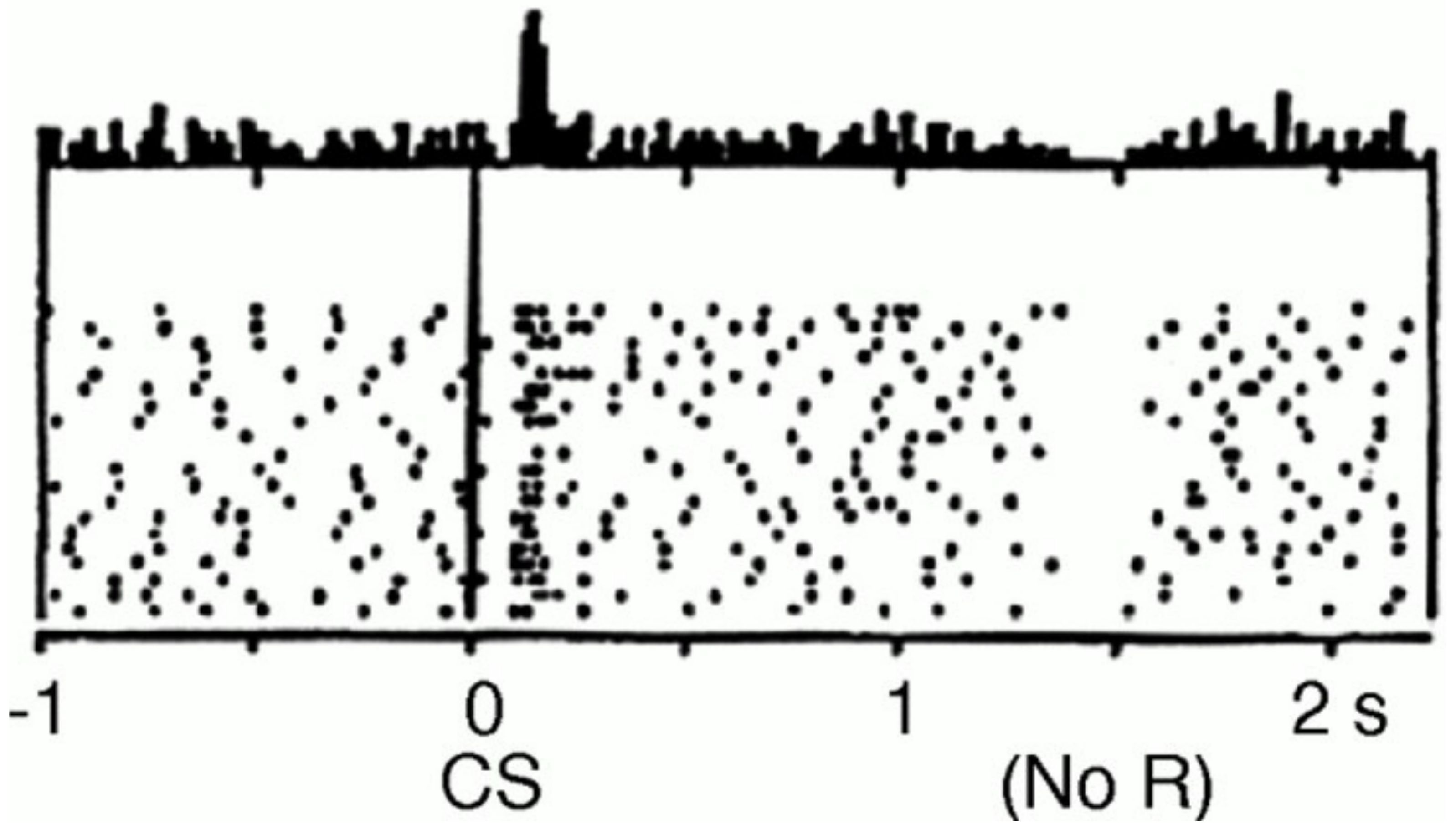
No prediction
Reward occurs



Reward predicted
Reward occurs



Reward predicted
No reward occurs



Schulz et al ‘97

Multistep decisions and temporal difference learning

now later

$$Q^{\pi}(s,a) = \sum_{s'} p(s' \mid s,a) r(s') + \gamma Q^{\pi}(s', \pi(s'))$$

Multistep decisions and temporal difference learning

Model free

now later

$$Q^{\pi}(s,a) = \sum_{s'} p(s' \mid s,a) r(s') + \gamma Q^{\pi}(s', \pi(s'))$$

$$\left(r(s') + \gamma \max_{a} Q(s',a') - Q(s,a)\right)$$

prediction error

Multistep decisions and temporal difference learning

Model free

now later

$$Q^{\pi}(s,a) = \sum_{s'} p(s' \mid s,a) r(s') + \gamma Q^{\pi}(s', \pi(s'))$$

new prediction

learning rate

$$Q(s,a) = Q(s,a) + \alpha(r(s') + \gamma \max_{a} Q(s',a') - Q(s,a))$$

old prediction

prediction error

Multistep decisions and thinking the future through

now later

$$Q^\pi(s,a) = \sum_{s'} p(s'|s,a)r(s') + \gamma Q^\pi(s',\pi(s'))$$

Model based

The lunch problem

Mensa

Bistro

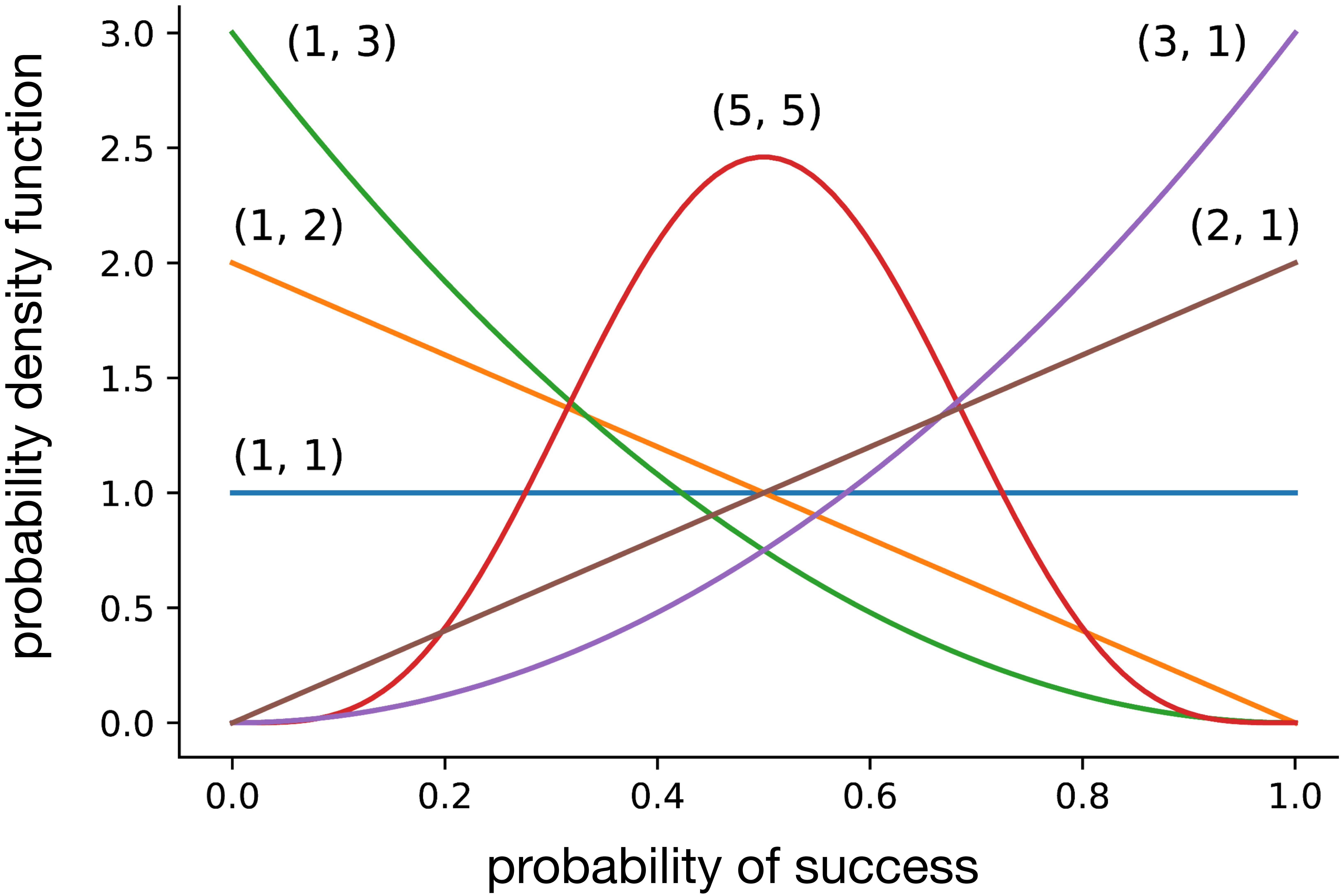
UKE’s Website

Let’s define a state as how many times we had (success, failure)

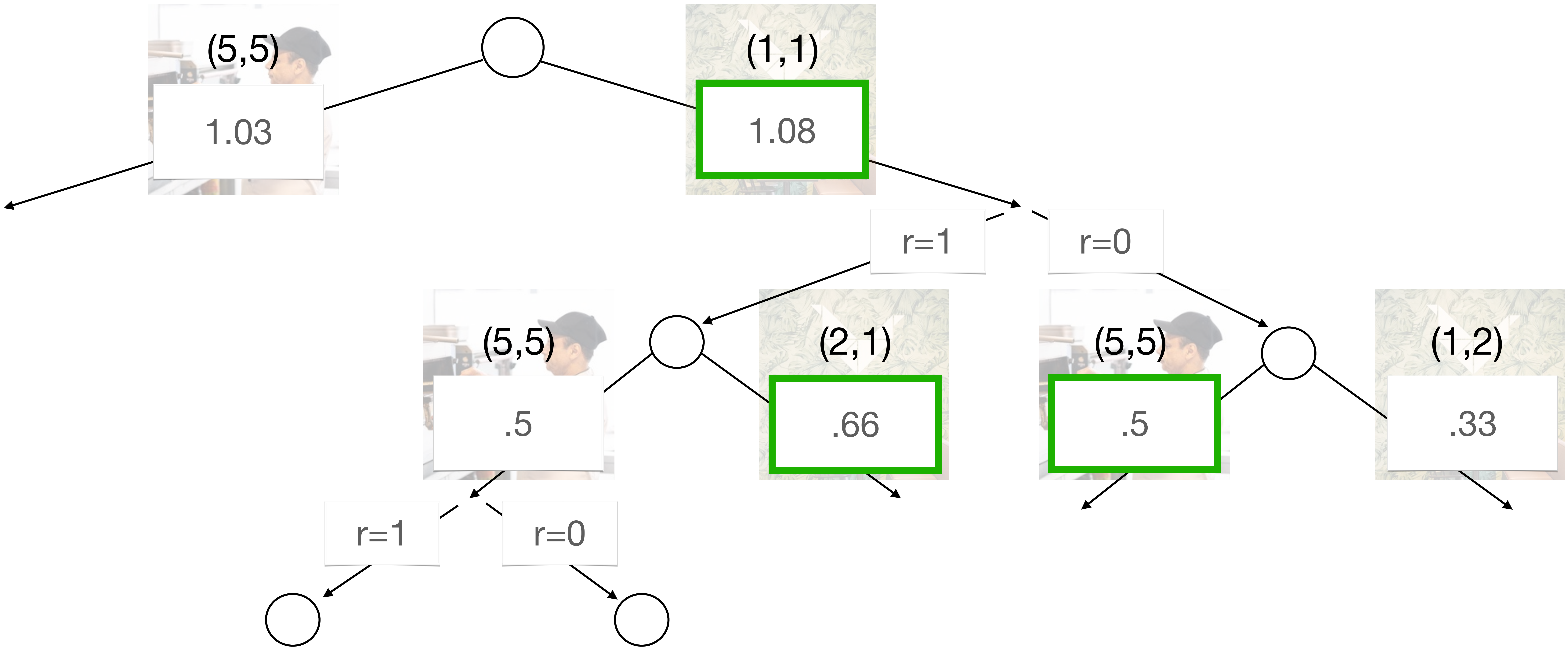
(5,5)

(1,1)

How would beliefs change



Where to go for lunch?



$$Q^{\pi}(s,a) = \sum_{s'} p(s' \mid s,a) r(s') + \gamma Q^{\pi}(s', \pi(s'))$$

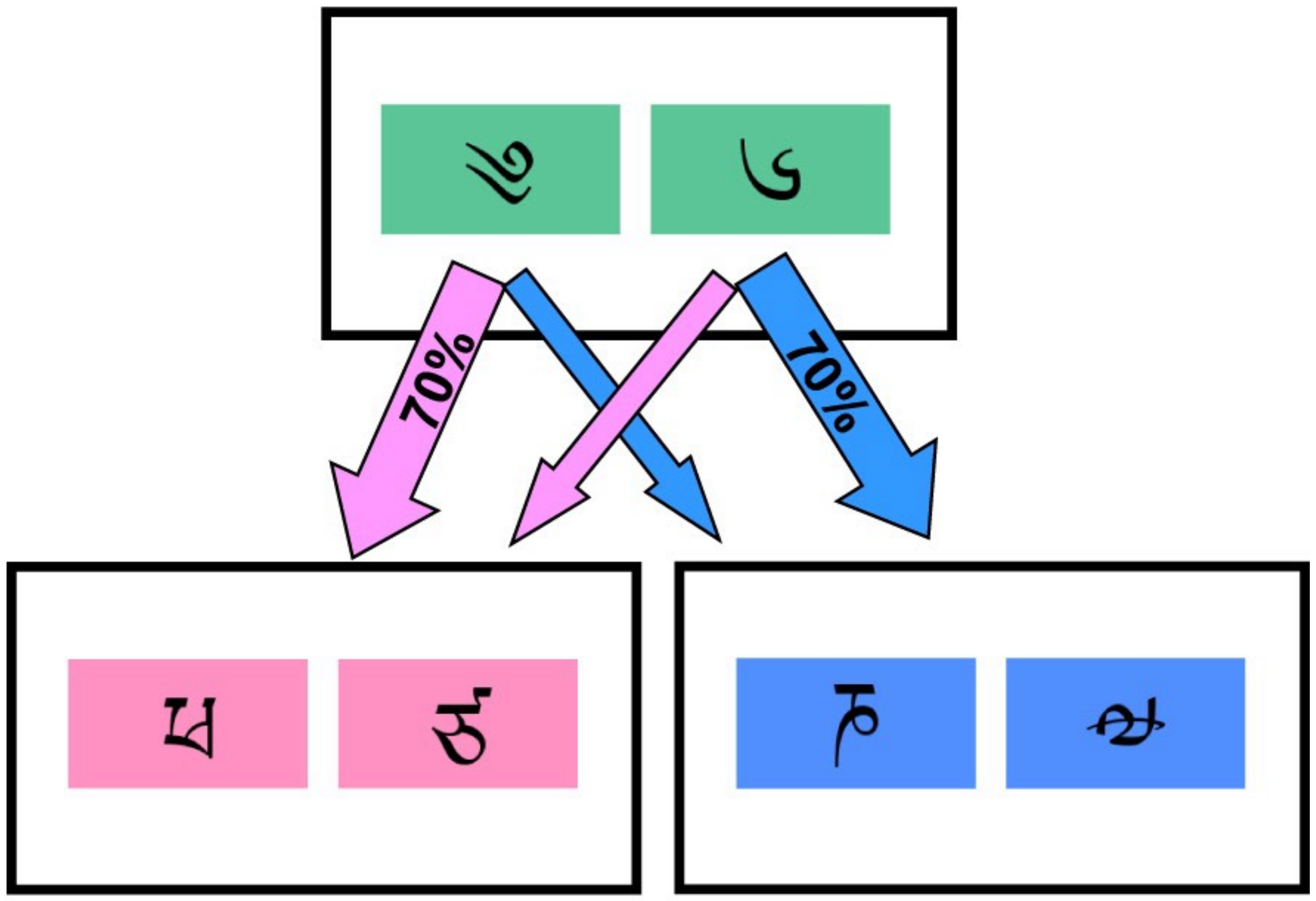
Bonus: restaurant choice table

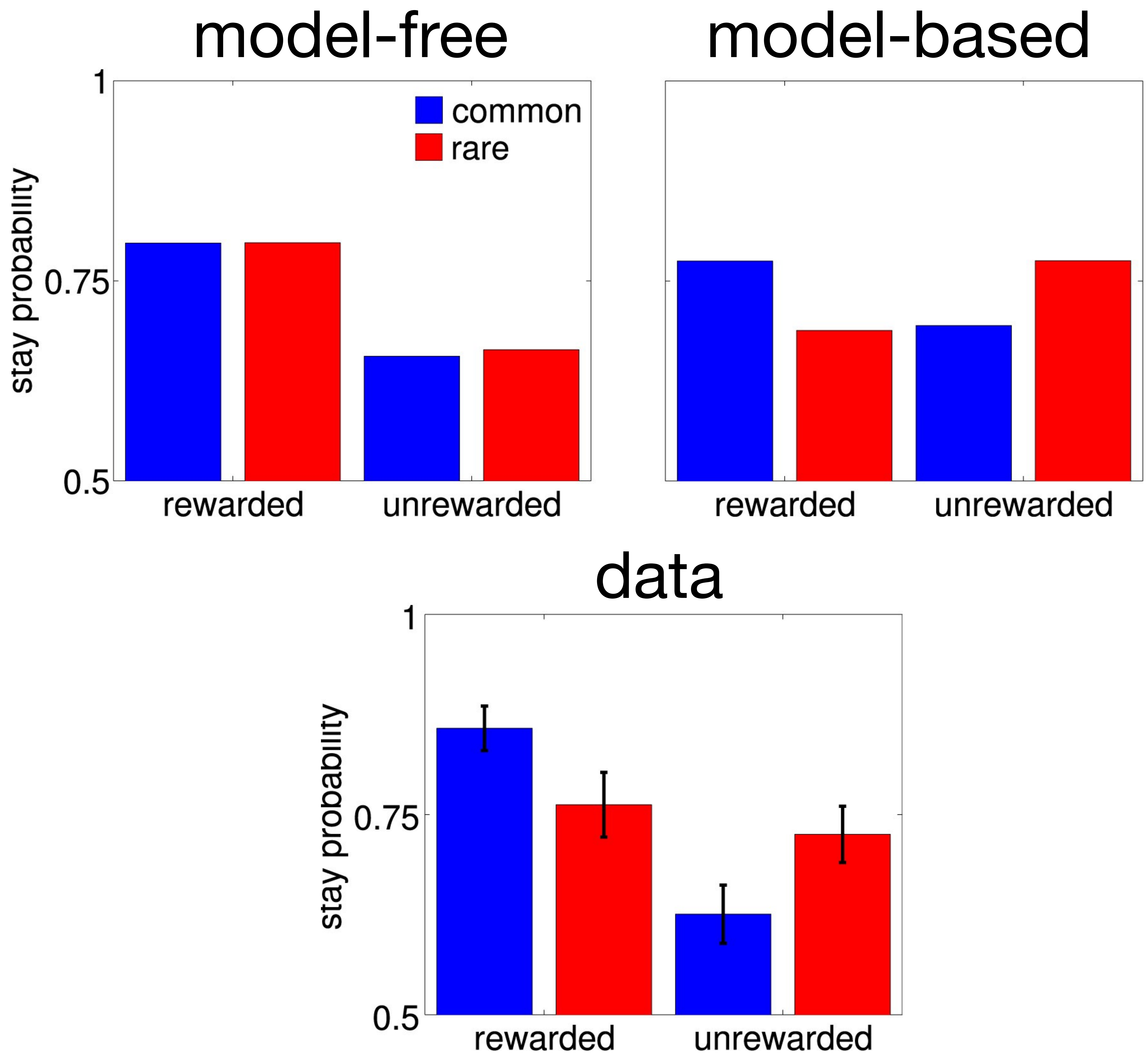
β \ α	1	2	3	4	5	6	7	8	9	10
1	.7029	.8001	.8452	.8723	.8905	.9039	.9141	.9221	.9287	.9342
2	.5001	.6346	.7072	.7539	.7869	.8115	.8307	.8461	.8588	.8695
3	.3796	.5163	.6010	.6579	.6996	.7318	.7573	.7782	.7956	.8103
4	.3021	.4342	.5184	.5809	.6276	.6642	.6940	.7187	.7396	.7573
5	.2488	.3720	.4561	.5179	.5676	.6071	.6395	.6666	.6899	.7101
6	.2103	.3245	.4058	.4677	.5168	.5581	.5923	.6212	.6461	.6677
7	.1815	.2871	.3647	.4257	.4748	.5156	.5510	.5811	.6071	.6300
8	.1591	.2569	.3308	.3900	.4387	.4795	.5144	.5454	.5723	.5960
9	.1413	.2323	.3025	.3595	.4073	.4479	.4828	.5134	.5409	.5652
10	.1269	.2116	.2784	.3332	.3799	.4200	.4548	.4853	.5125	.5373

Gittins 2011

Does our behaviour correspond to reinforcement learning predictions?

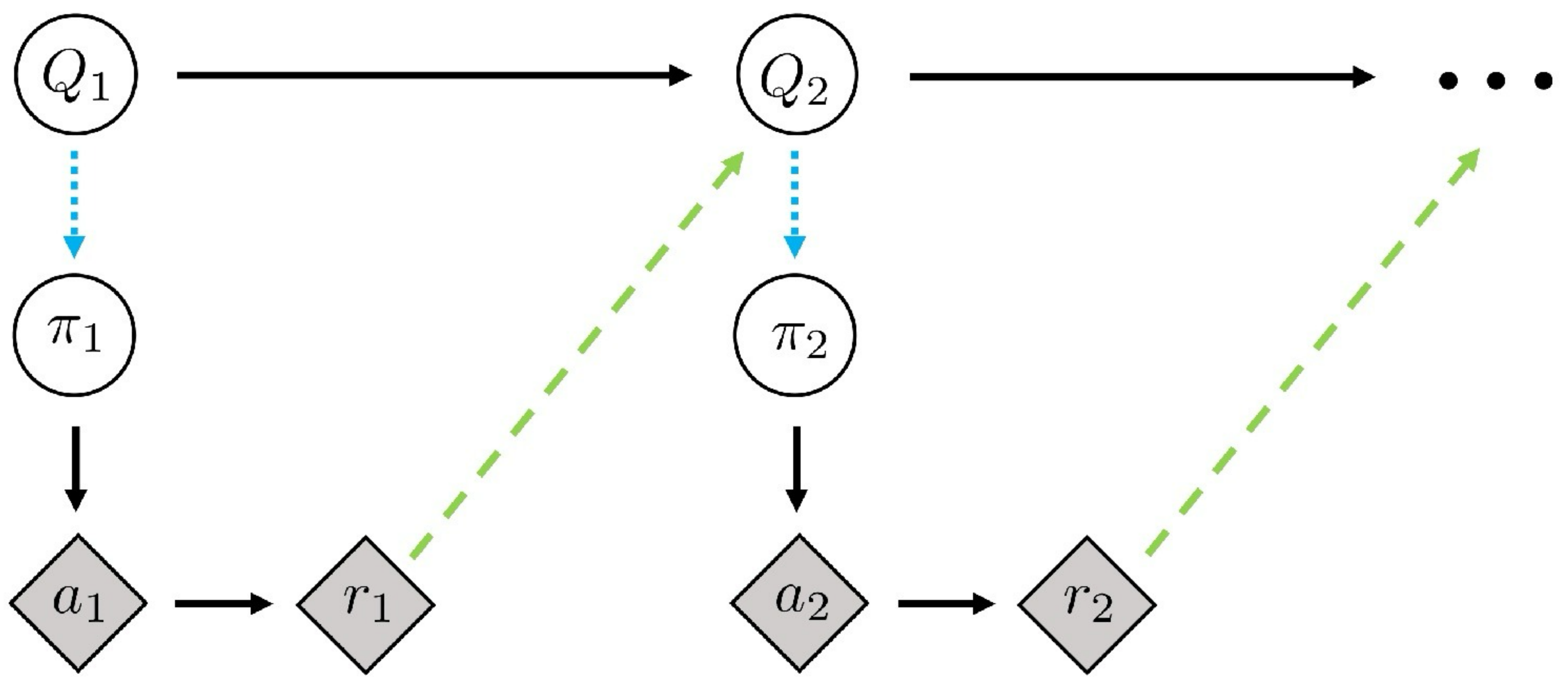
The 2-step task





Daw et al ‘12

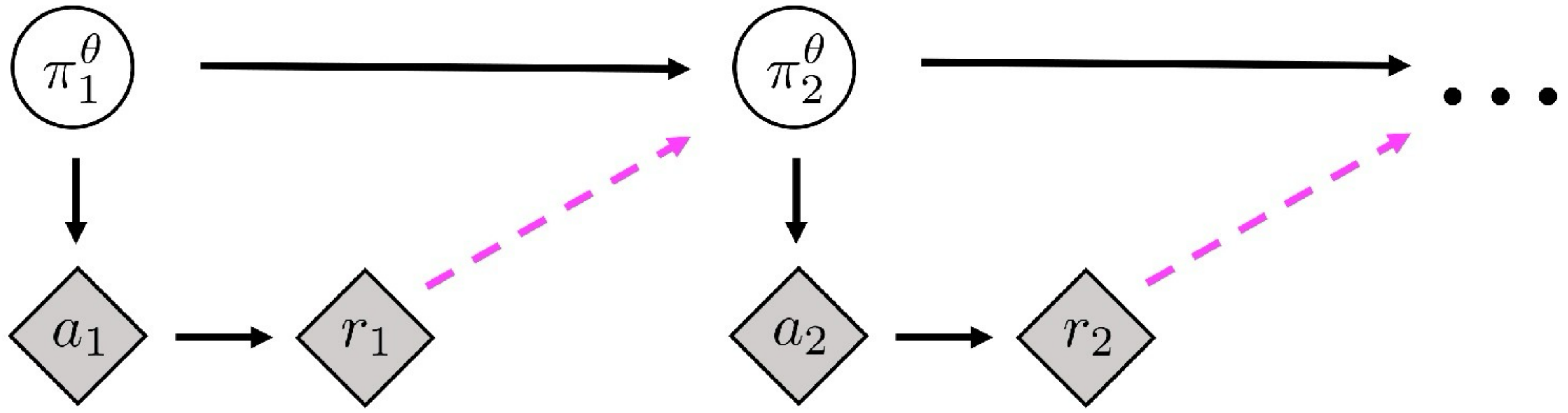
So far we have always computed values:



Compute policy from action-values: $$\pi(a) = \frac{e^{\beta \cdot Q(a)}}{\sum_{\hat{a} \in A} e^{\beta \cdot Q(\hat{a})}}$$

Update action-values: $$\Delta Q(a) = \alpha \left(r_t - Q(a) \right)$$

Another way: policy based reinforcement learning



Parametrised policy $$\pi^\theta(a) = \frac{e^{\theta_a}}{\sum_{\hat{a} \in A} e^{\theta_{\hat{a}}}}$$

Update parameters: $$\Delta\theta_a = \begin{cases} \alpha \cdot [1 - \pi^\theta(a)] \cdot r_t & \text{if } a \text{ was chosen} \\ -\alpha \cdot \pi^\theta(a) \cdot r_t & \text{if } a \text{ was unchosen} \end{cases}$$

Bennett, Niv, Langdon, ‘21

o.solopchuk@uke.de