



Definitely not reinforcement learning

Today's menu

Imitation learning

a brief intro

Inverse reinforcement learning

a bit less brief

Problem formulation

You have a decision process,
but **no reward function**



Instead, there are examples
set by an “**expert**” agent



You want to learn
optimal policy by imitaton

Why bother

“natural” reward



toy tasks, videogames,
Robot gait @ race track,
Online advertising
Image captioning

No natural reward



real world problems,
Robot gait @ public space
Recommendation systems
Conversation systems

...

Why bother

“natural” reward



toy tasks, videogames,
Robot gait @ race track,
Online advertising
~~Image captioning~~

No natural reward



real world problems,
Robot gait @ public space
Recommendation systems
Conversation systems
Image Captioning, ...

Problem formulation

We have a dataset of sessions, $D: \{\tau_1, \tau_2, \tau_3\}$
under expert policy $\pi^*(a|s)$

$$\tau = \langle s, a, s', a', \dots s_T \rangle$$

You want to learn $\pi(a|s)$ that mimics $\pi^*(a|s)$

Q: how can we solve that?

Behavior cloning

We have a dataset of sessions, $D: \{\tau_1, \tau_2, \tau_3\}$
under expert policy $\pi^*(a|s)$

$$\tau = \langle s, a, s', a', \dots s_T \rangle$$

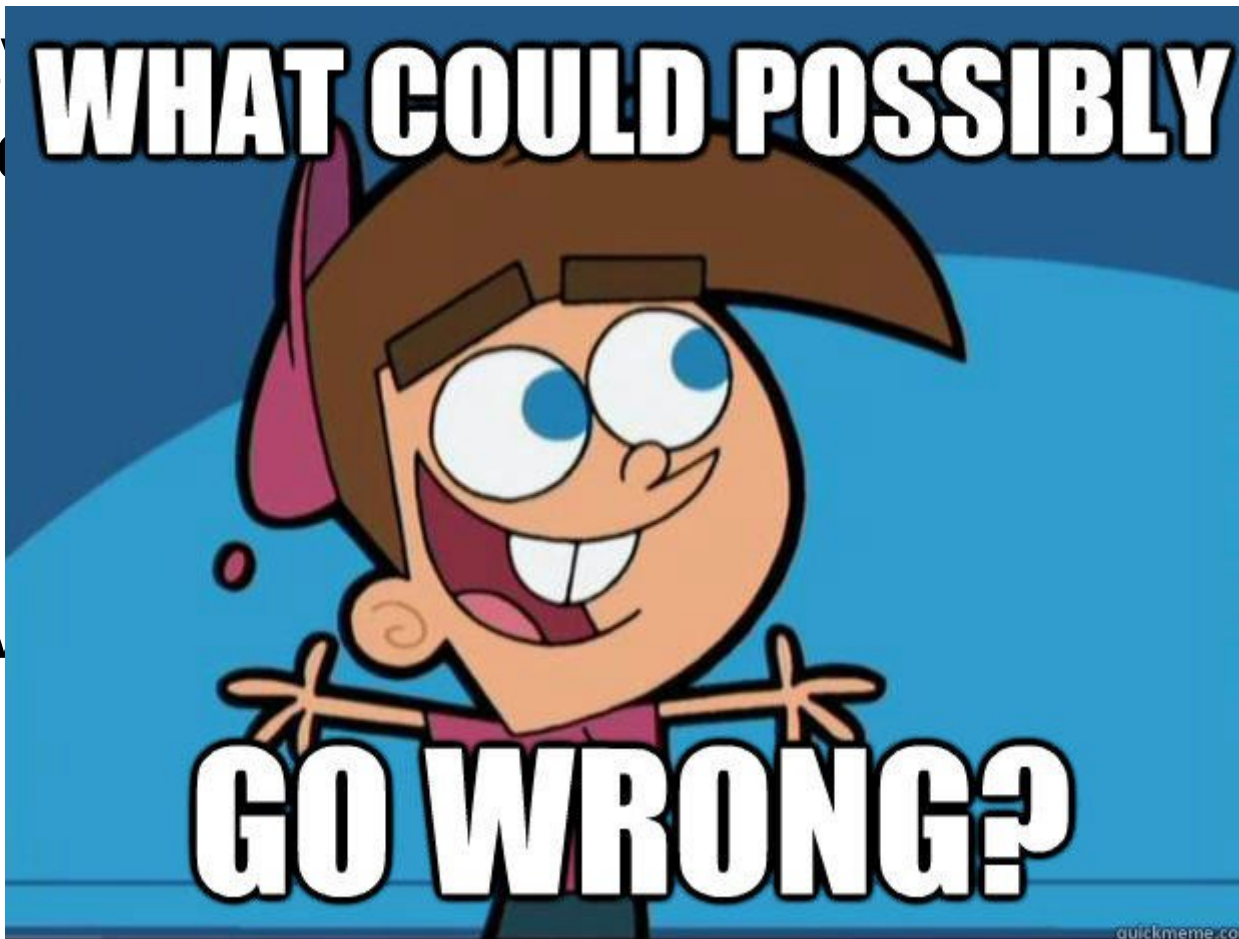
Supervised learning way:

$$\pi_{\theta}(a|s) = \underset{\pi_{\theta}}{\operatorname{argmax}} \sum_{\tau \in D} \sum_{s, a \in \tau} \log \pi(a|s)$$

Behavior cloning

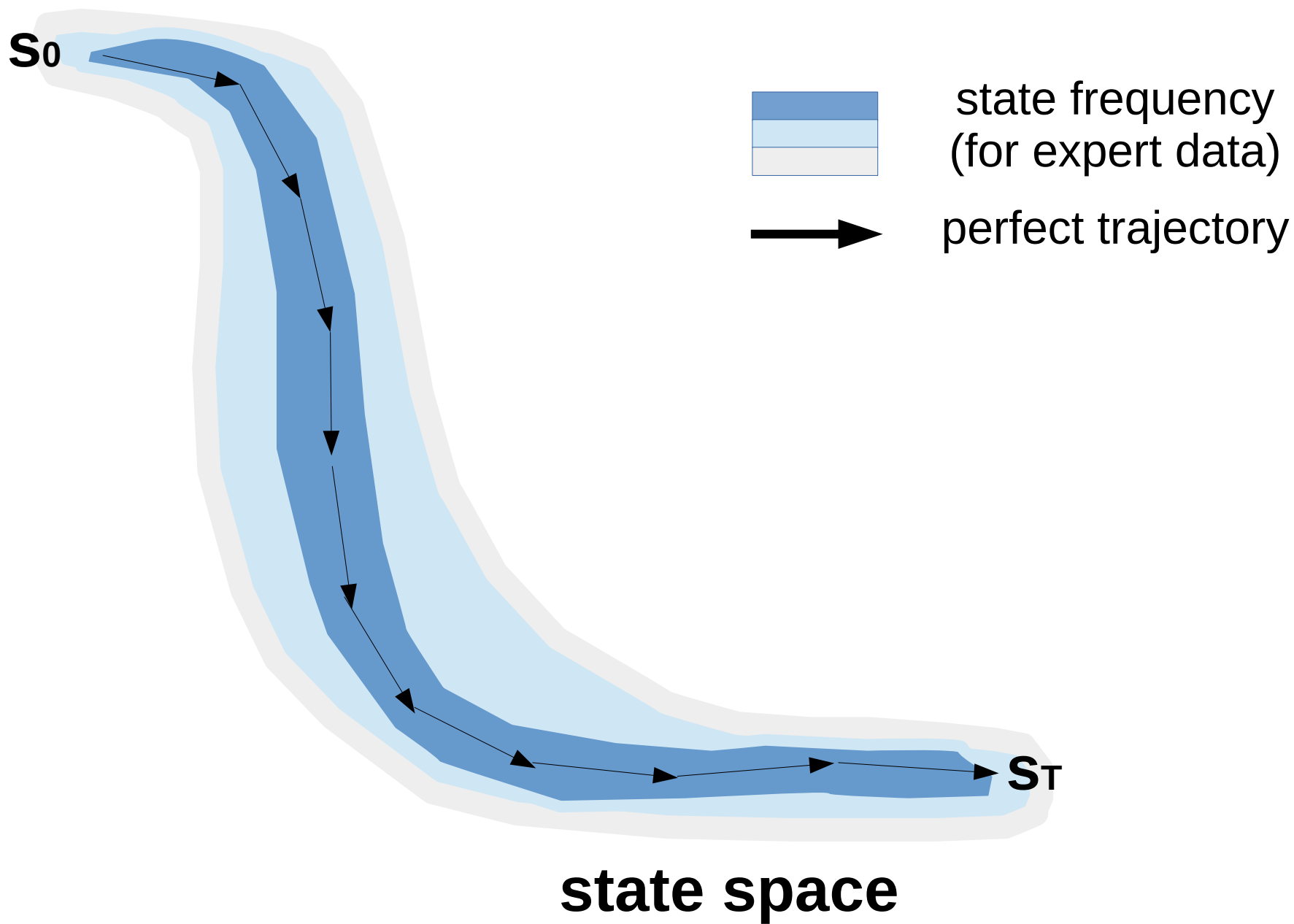
We have
understand

Superv

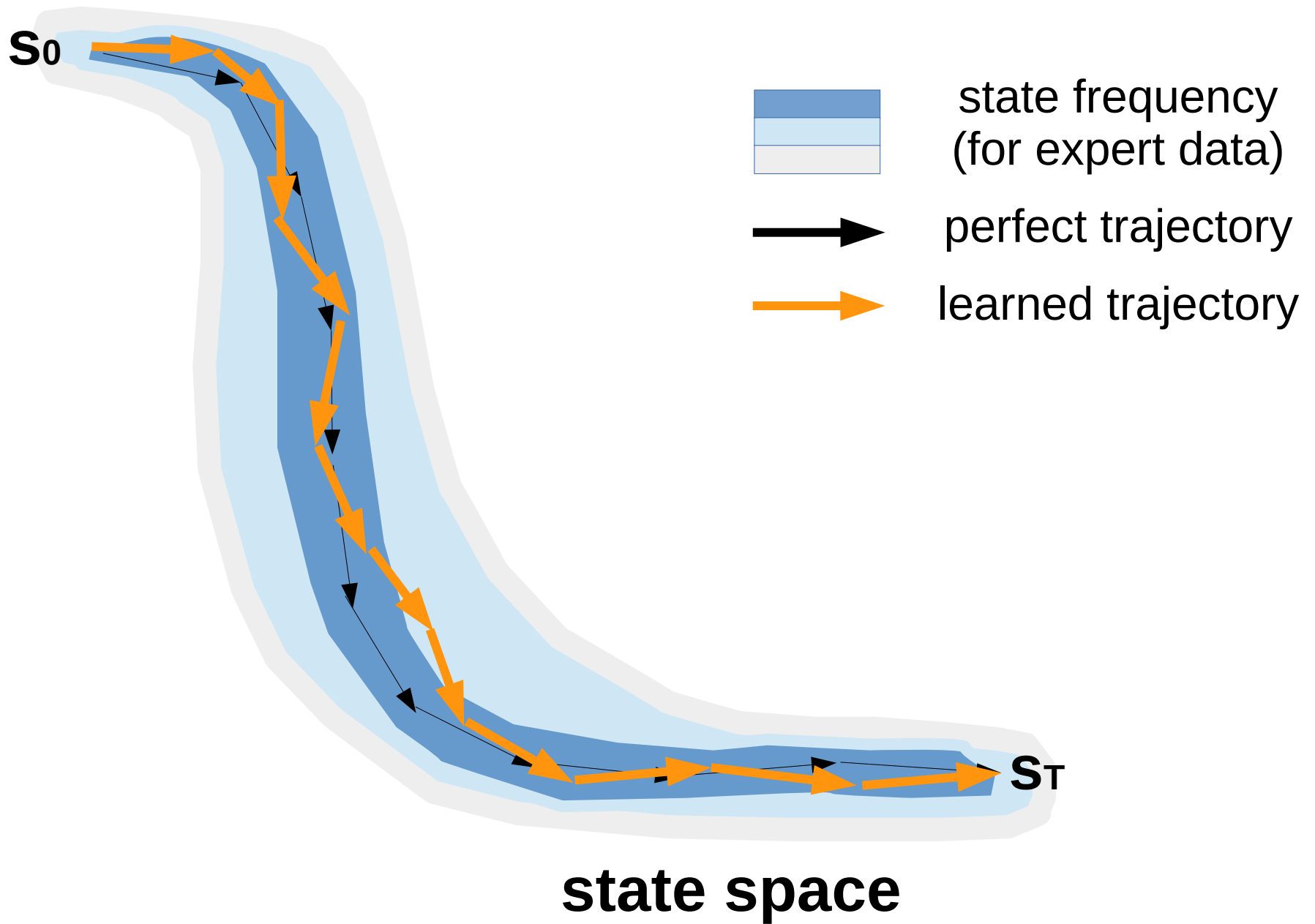


$\tau_2, \tau_3\}$

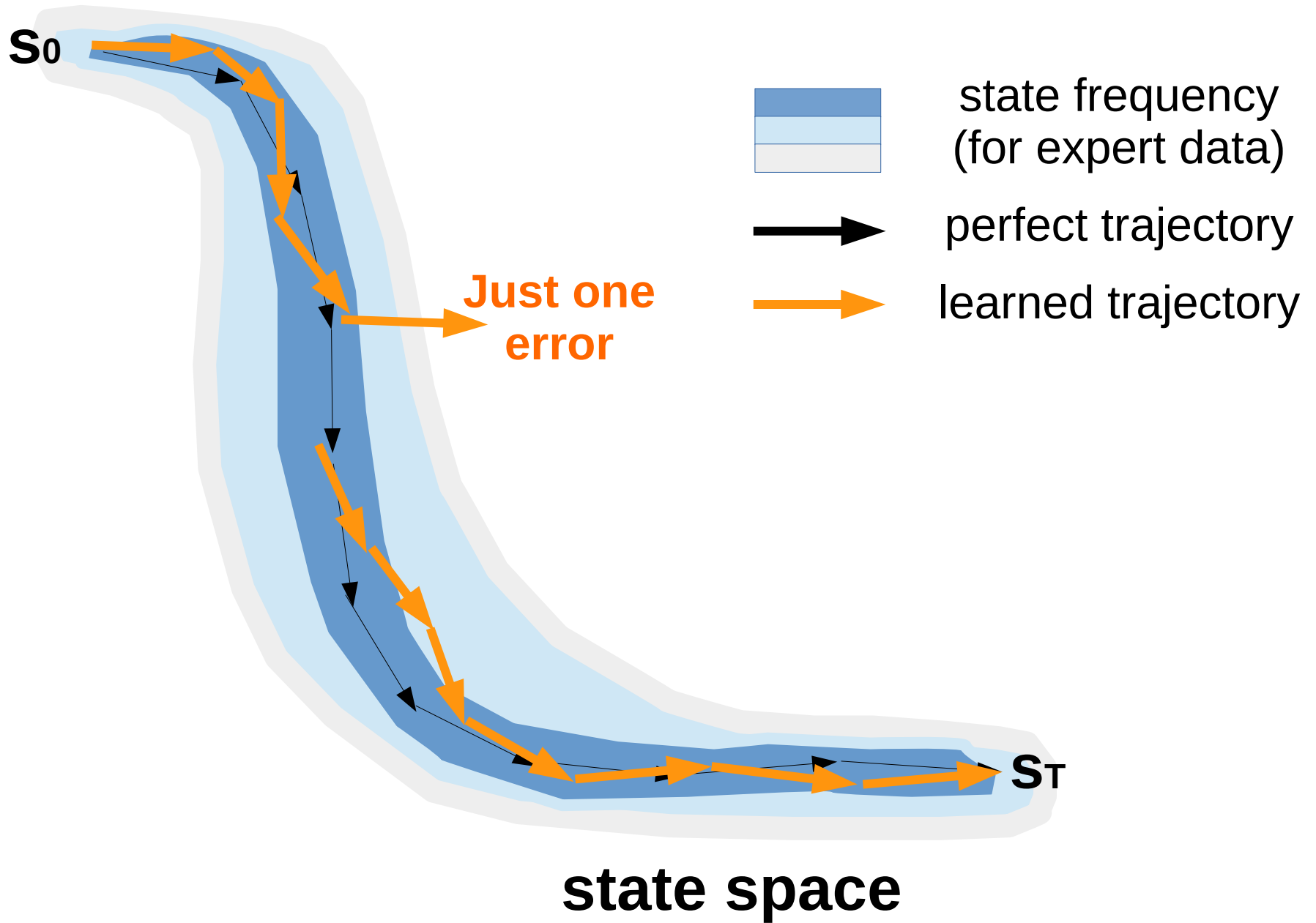
Behavior cloning



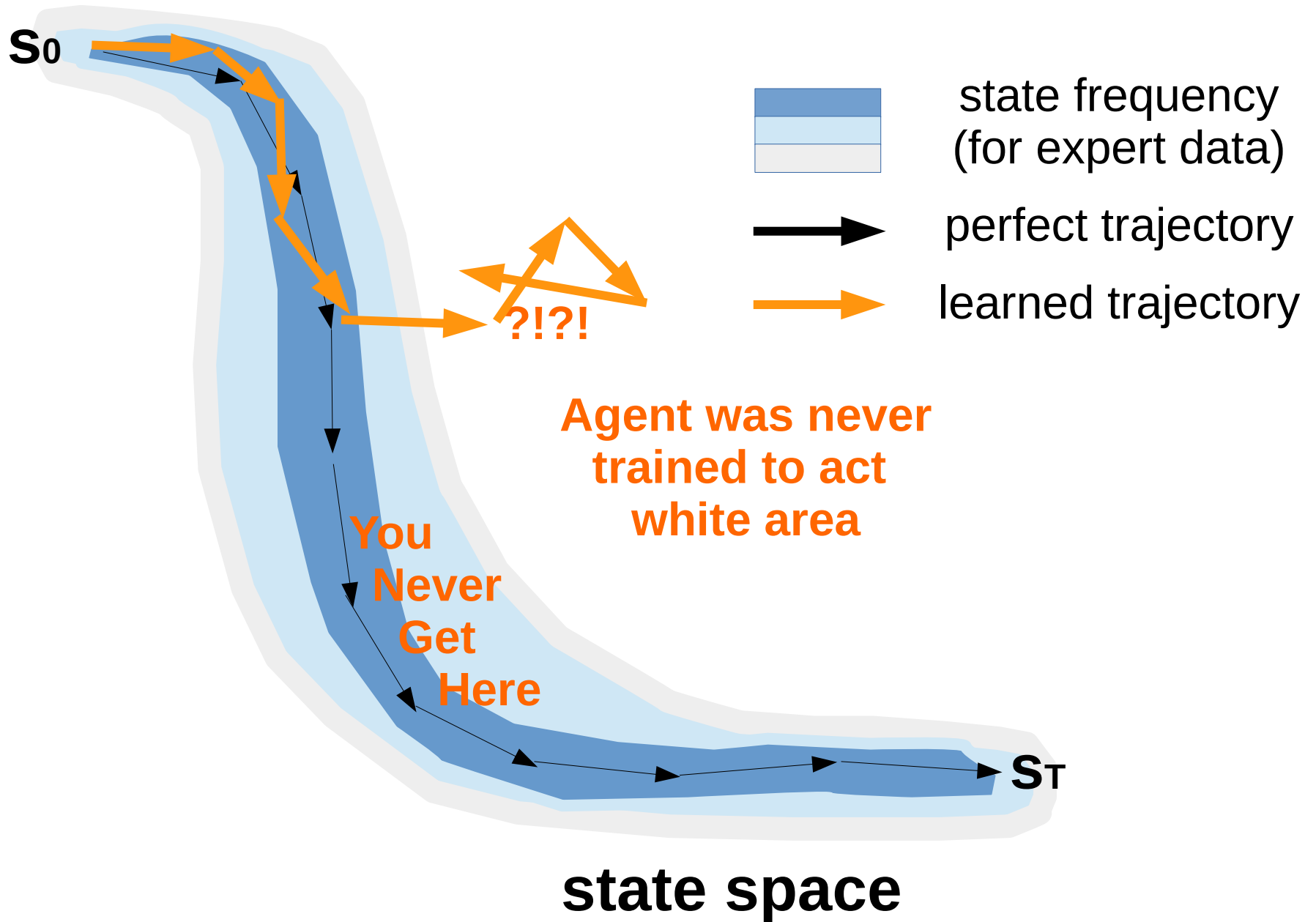
Behavior cloning



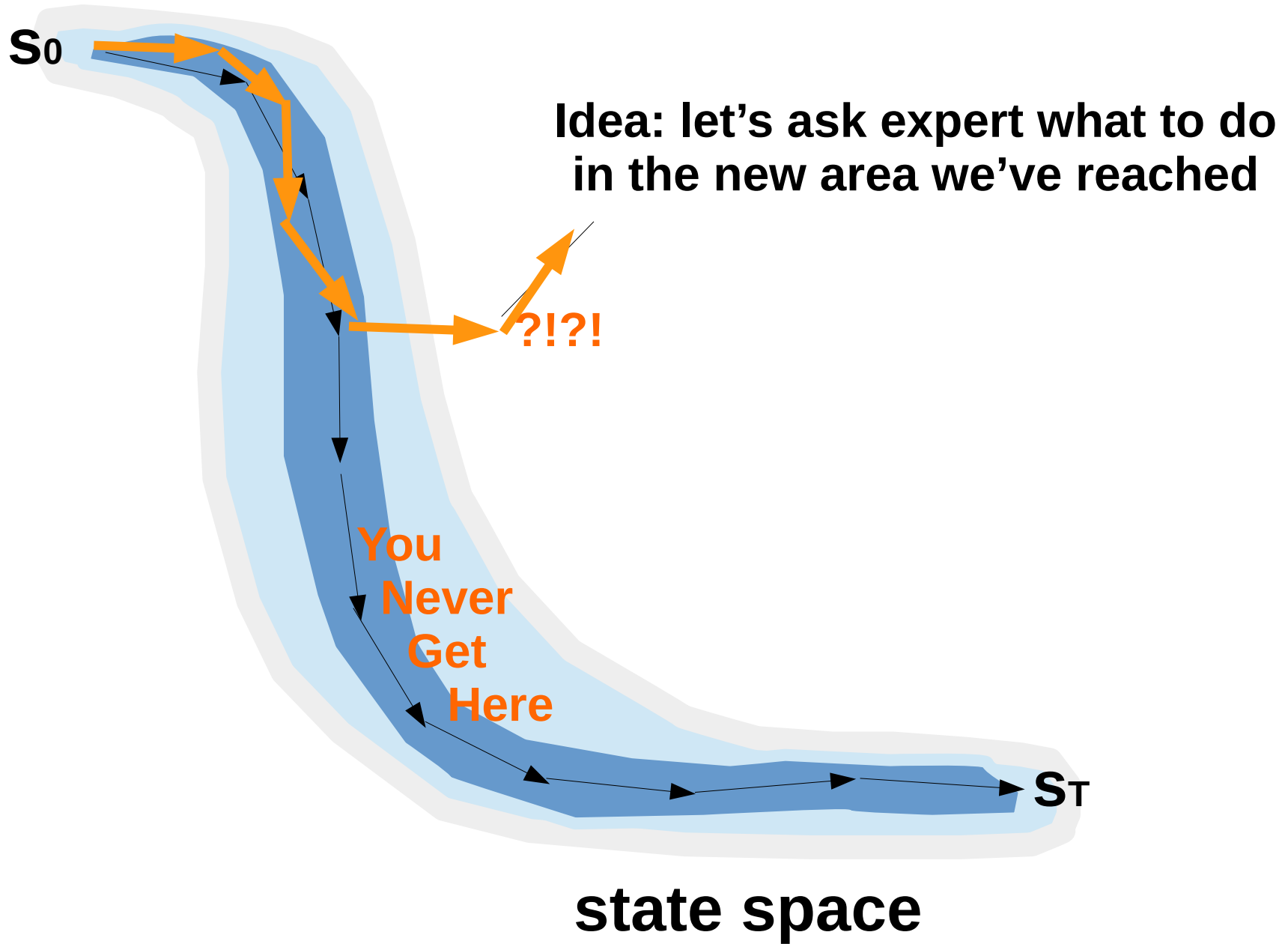
Behavior cloning



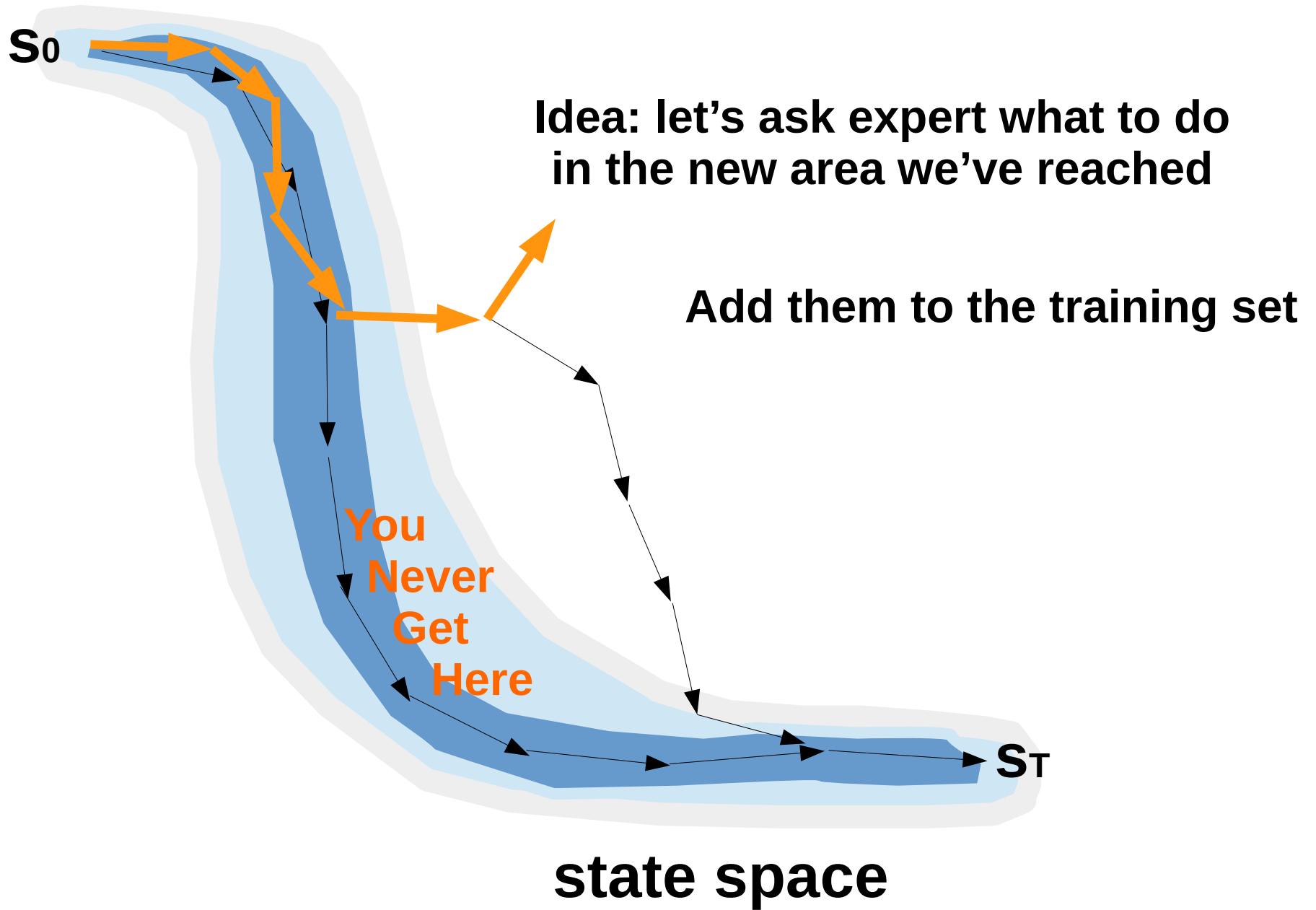
Behavior cloning



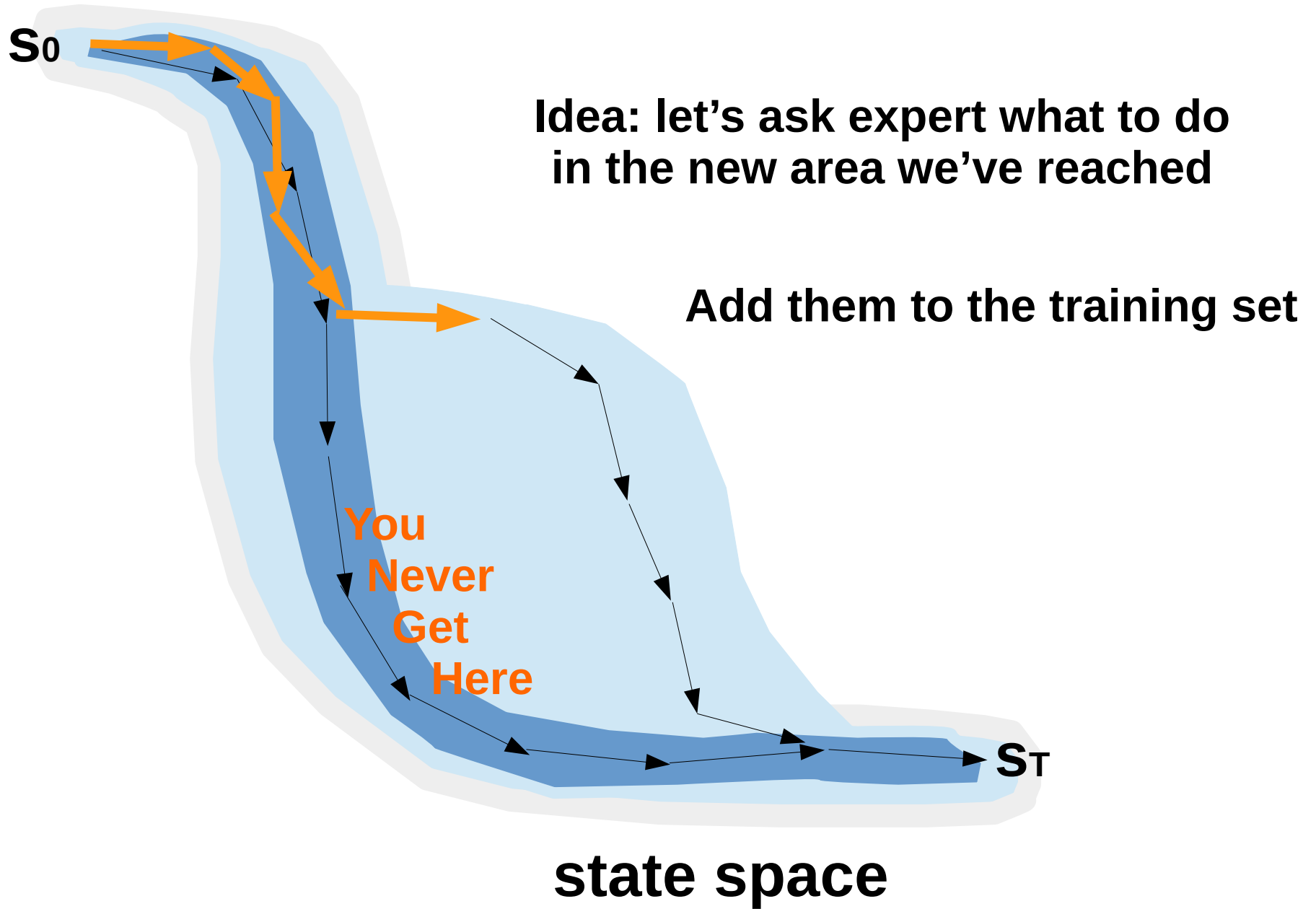
DAGGER



DAGGER



DAGGER



DAGGER

Initial dataset **D**: $\{\tau_1, \tau_2, \tau_3\}$,

Initial policy $\pi_\theta(a|s)$

Forever:

- Update policy

$$\pi_\theta(a|s) := \underset{\pi_\theta}{\operatorname{argmax}} \sum_{\tau \in D} \sum_{s, a \in \tau} \log \pi(a|s)$$

- Sample session under current $\pi_\theta(a|s)$

$$\tau = \langle s, a, s', a', \dots s_T \rangle$$

- Update dataset

$$D := D \cup \{ s, \operatorname{get_optimal_action}(s), \forall s \in \tau \}$$

DAGGER

Benefits

- + fixes state space error**

Drawbacks

- you need expert feedback on new sessions**
- degrades if expert actions are noisy**

DAGGER

Benefits

- + fixes state space error**

Drawbacks

- you need expert feedback on new sessions**
- degrades if expert actions are noisy**

Idea: first infer what rewards does expert want
then train “regular” rl to maximize them

Inverse Reinforcement Learning

“regular” RL

inverse RL

given:

Environment,
Reward function

Environment,
Optimal policy

find out:

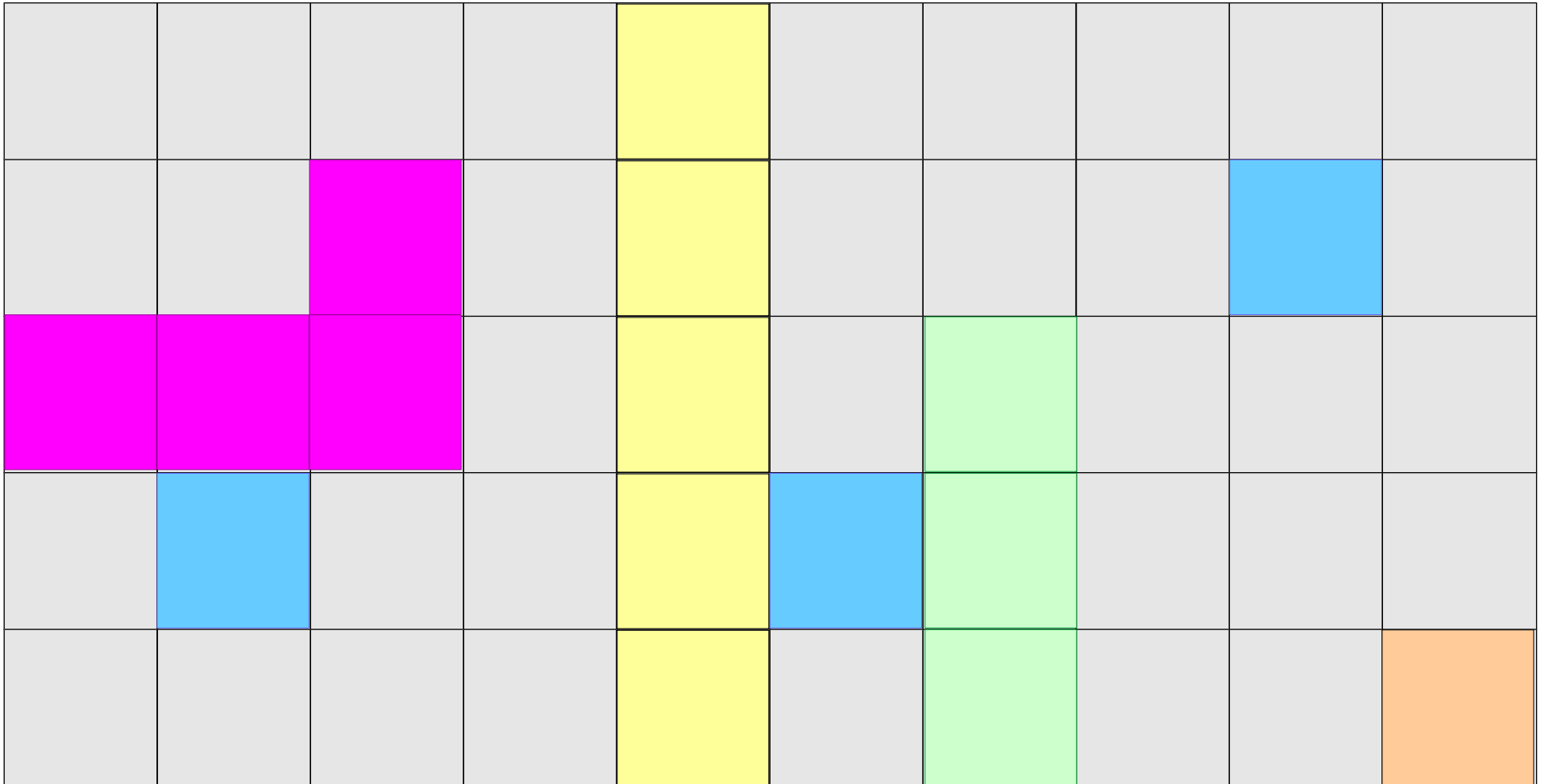
Optimal policy

Reward function

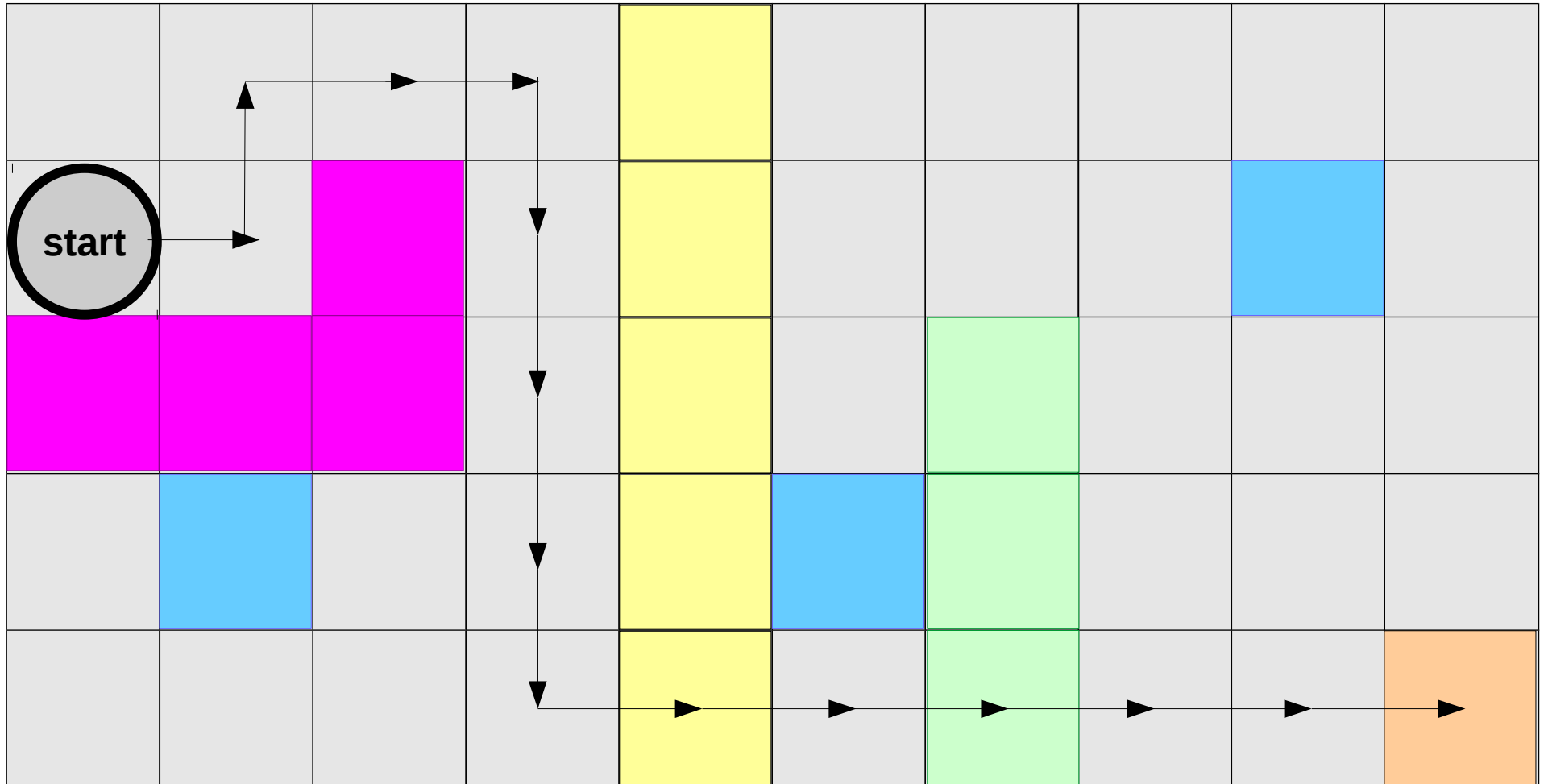
Is it even possible?



Is it even possible?

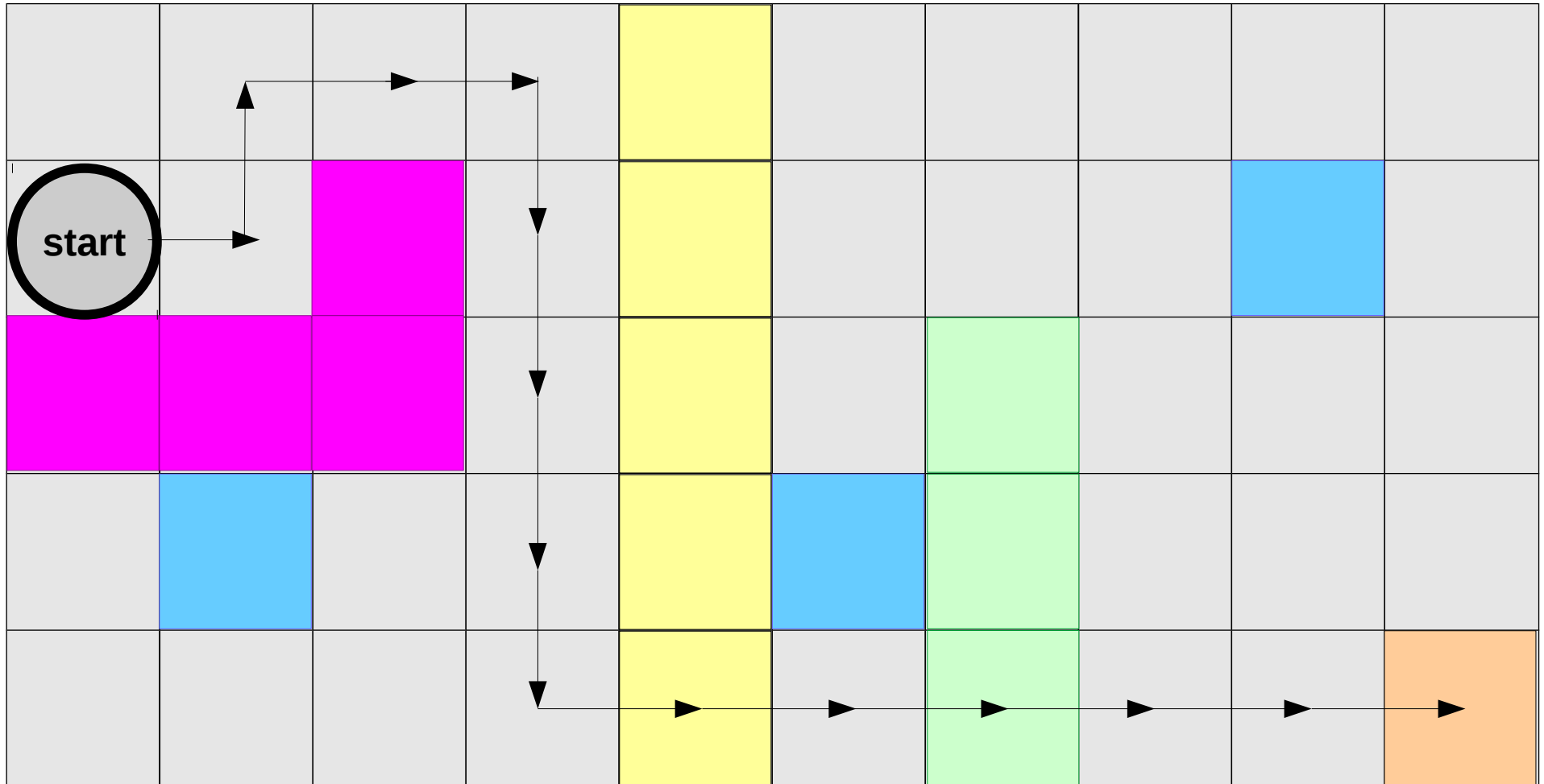


Is it even possible?



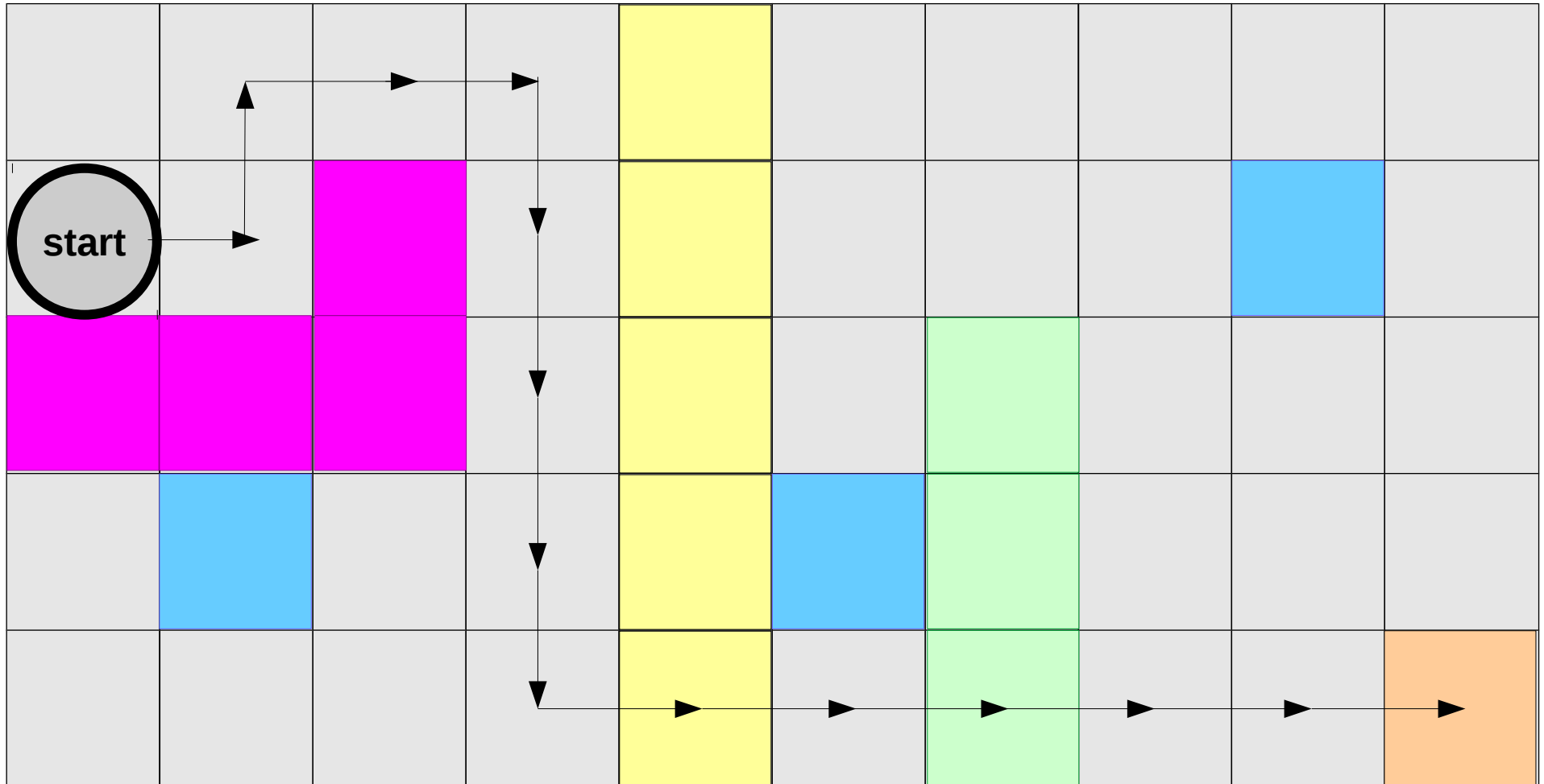
Agent is rewarded for the **first time** it enters a tile
It can exit the session at will. Also $R(\text{grey}) = 0$

Is it even possible?



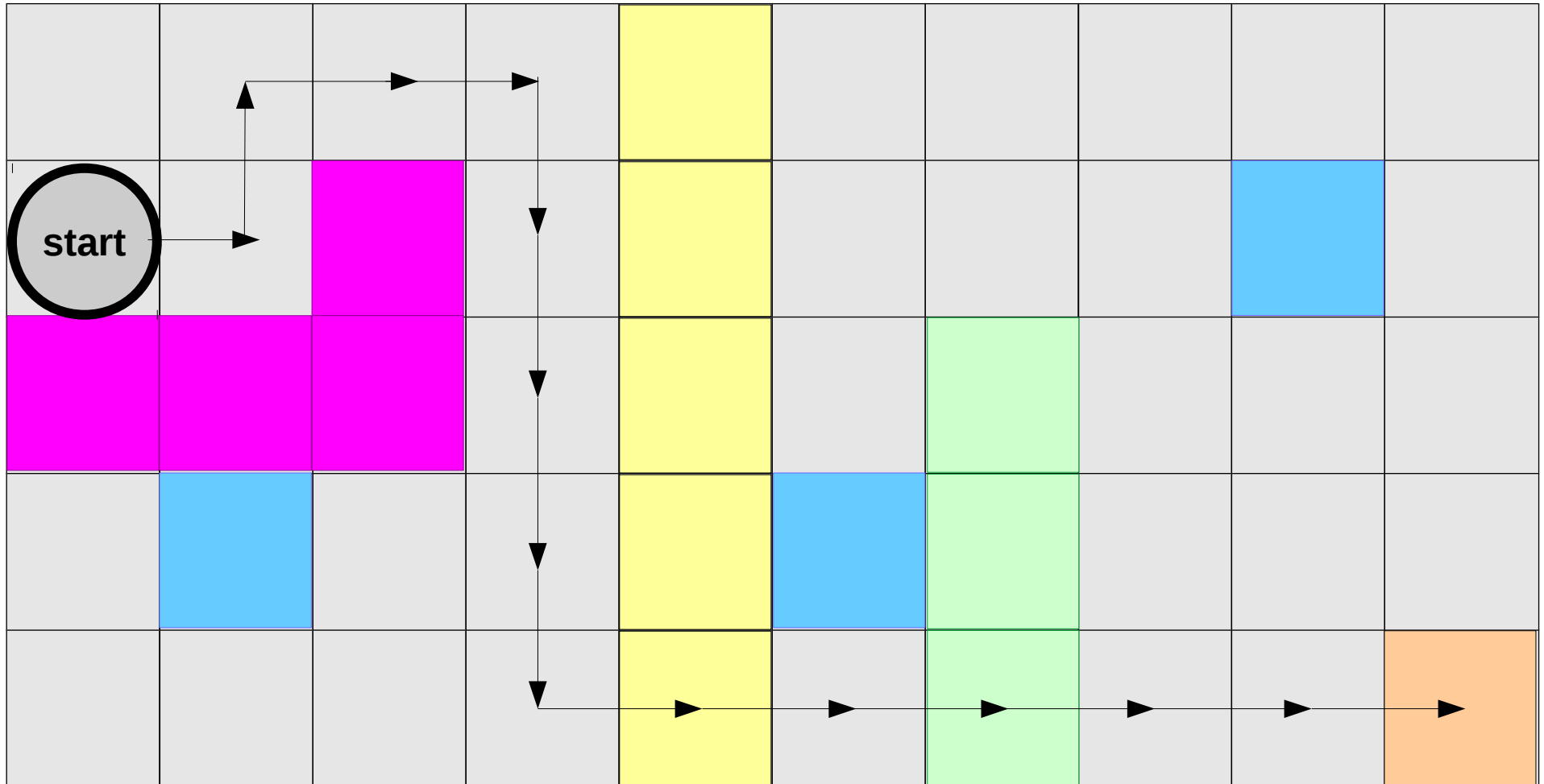
Agent is rewarded for the **first time** it enters a tile
Q: what is the “cost of living” for 1 step? (+1 / -1)

Is it even possible?



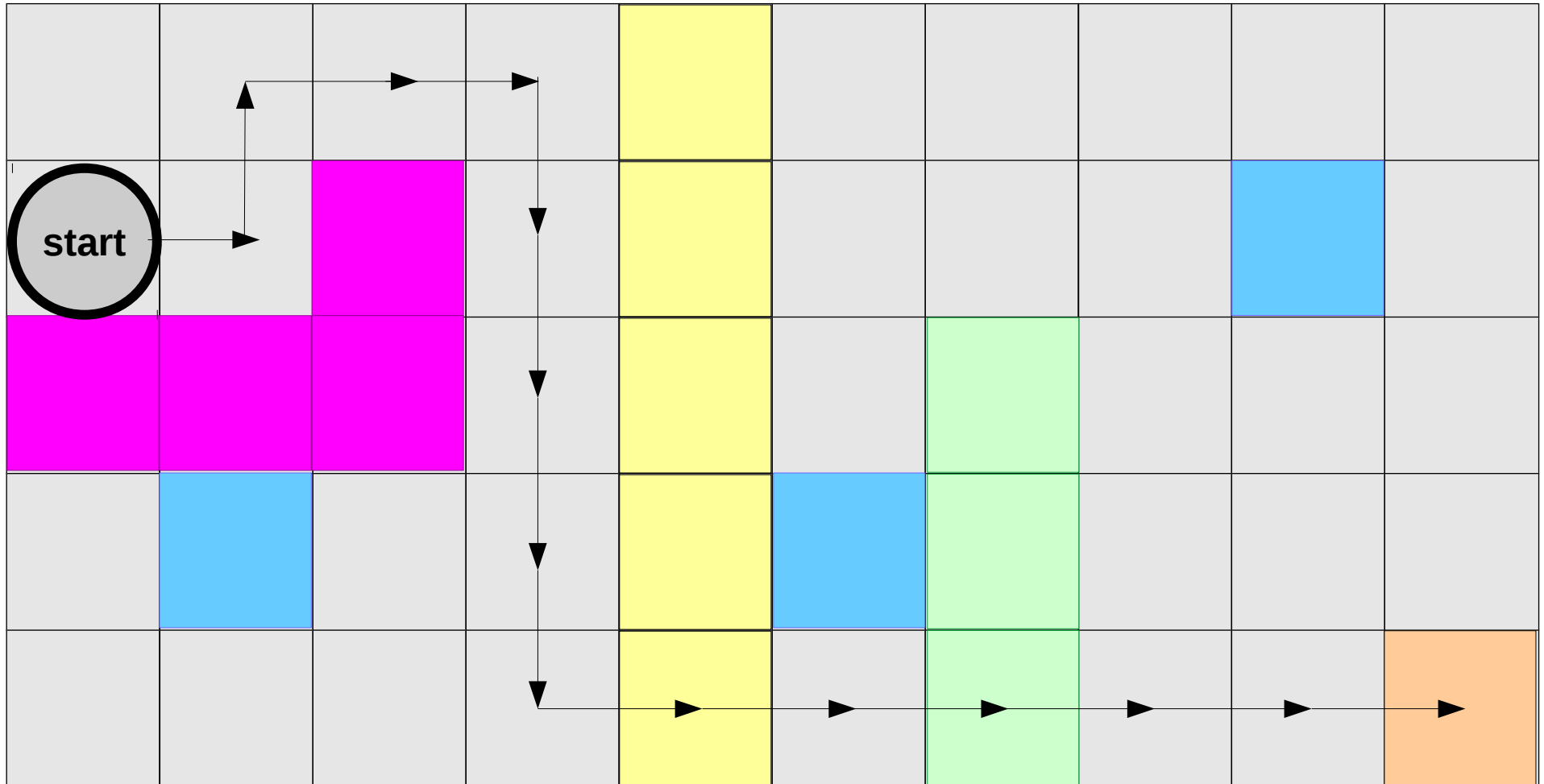
Agent is rewarded for the **first time** it enters a tile
Agent gets -1 for each turn (cost of living)

Is it even possible?



$R(\blacksquare) = ?$ $R(\blacksquare) = ?$ $R(\blacksquare) = ?$

Is it even possible?

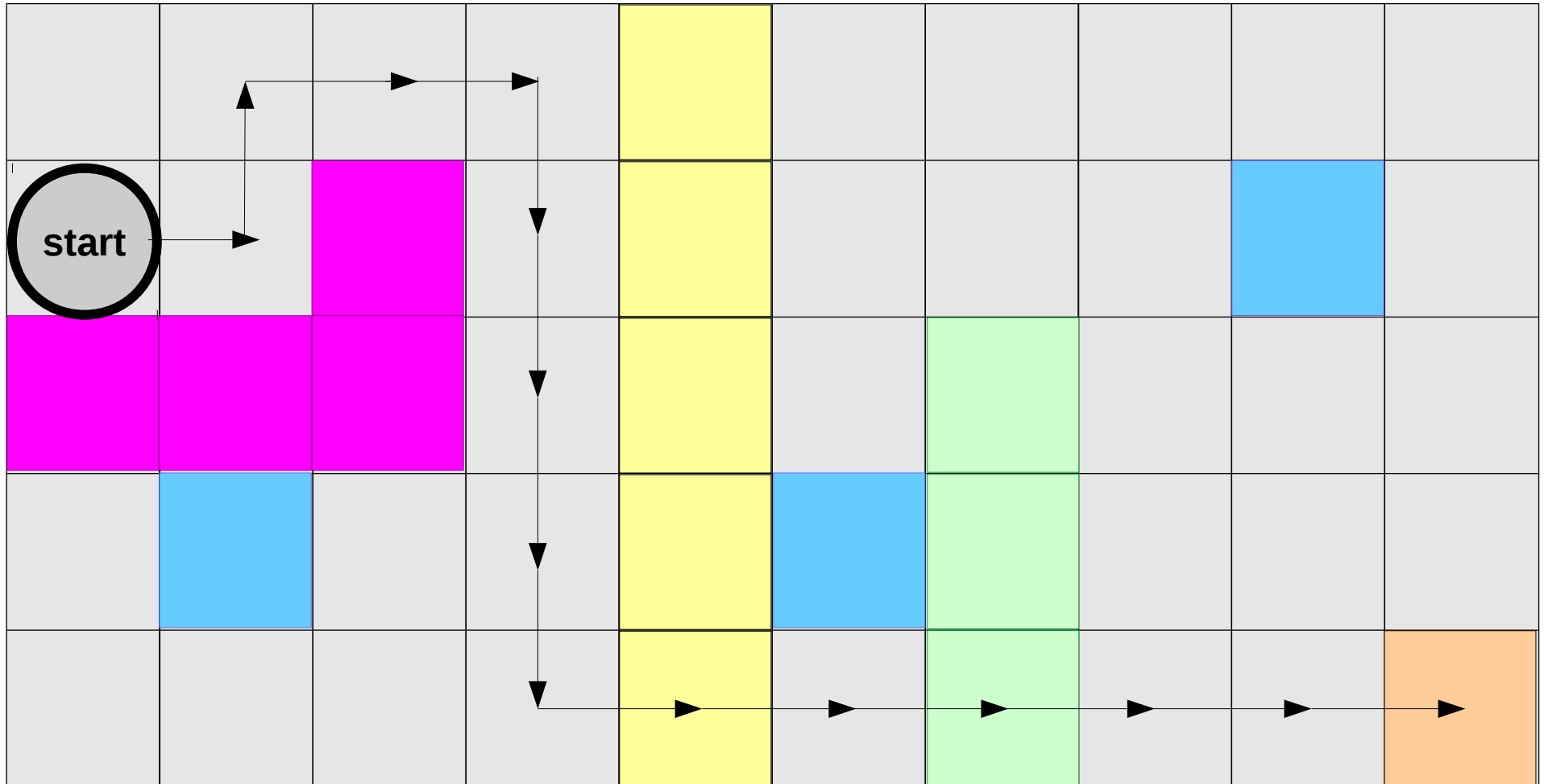


$$R(\text{red}) \gg 0$$

$$0 < R(\text{green}) \leq 2$$

$$R(\text{magenta}) \ll 0$$

Is it even possible?



$R(\text{yellow}) = ?$ $R(\text{blue}) = ?$

Is it even possible?

Yes, to some extent

Maximum Entropy Inverse RL

D. Ziebart et al.

We have a dataset of sessions, $D: \{\tau_1, \tau_2, \tau_3\}$
under expert policy $\pi^*(a|s)$

$$\tau = \langle s, a, s', a', \dots s_T \rangle$$

Assumption: assume that $\pi^*(\tau) \sim e^{R(\tau)}$
where

$$R(\tau) = \sum_{s_\tau, a_\tau} r(s_\tau, a_\tau)$$

(alt: use gamma)

Maximum Entropy Inverse RL

We have a dataset of sessions, $D: \{\tau_1, \tau_2, \tau_3\}$
under expert policy $\pi^*(a|s)$

$$\tau = \langle s, a, s', a', \dots s_T \rangle$$

Assumption: assume that $\pi^*(\tau) \sim e^{R(\tau)}$
where

$$R(\tau) = \sum_{s_\tau, a_\tau} r(s_\tau, a_\tau)$$

(alt: use gamma)

Sketch: learn $r(s_\tau, a_\tau)$ to maximize likelihood of D

Maximum Entropy Inverse RL

How it works: $\pi^*(\tau) \sim e^{R(\tau)}$

$$\log P(D|\theta) = \sum_{\tau \in D} \log \pi^*(\tau; \theta) = \sum_{\tau \in D} \log \frac{e^{R_\theta(\tau)}}{\sum_{\tau'} e^{R_\theta(\tau')}} =$$

Do you see the problem?

Maximum Entropy Inverse RL

How it works: $\pi^*(\tau) \sim e^{R(\tau)}$

$$\log P(D|\theta) = \sum_{\tau \in D} \log \pi^*(\tau; \theta) = \sum_{\tau \in D} \log \frac{e^{R_\theta(\tau)}}{\sum_{\tau'} e^{R_\theta(\tau')}} =$$



sum over all trajectories

Maximum Entropy Inverse RL

Let's simplify:

$$llh = \sum_{\tau \in D} \log \frac{e^{R_{\theta}(\tau)}}{\sum_{\tau'} e^{R_{\theta}(\tau')}} =$$

$$= \sum_{\tau \in D} [R_{\theta}(\tau) - \log \sum_{\tau'} e^{R_{\theta}(\tau')}] = \sum_{\tau \in D} R_{\theta}(\tau) - N \cdot \log \sum_{\tau'} e^{R_{\theta}(\tau')}$$

$$\nabla_{\theta} llh = ?$$

Maximum Entropy Inverse RL

Let's simplify:

$$llh = \sum_{\tau \in D} \log \frac{e^{R_{\theta}(\tau)}}{\sum_{\tau'} e^{R_{\theta}(\tau')}} =$$

$$= \sum_{\tau \in D} [R_{\theta}(\tau) - \log \sum_{\tau'} e^{R_{\theta}(\tau')}] = \sum_{\tau \in D} R_{\theta}(\tau) - N \cdot \log \sum_{\tau'} e^{R_{\theta}(\tau')}$$

$$\nabla_{\theta} llh = \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \nabla_{\theta} \log \sum_{\tau'} e^{R_{\theta}(\tau')} =$$

Maximum Entropy Inverse RL

Let's simplify:

$$llh = \sum_{\tau \in D} \log \frac{e^{R_{\theta}(\tau)}}{\sum_{\tau'} e^{R_{\theta}(\tau')}} =$$

$$= \sum_{\tau \in D} [R_{\theta}(\tau) - \log \sum_{\tau'} e^{R_{\theta}(\tau')}] = \sum_{\tau \in D} R_{\theta}(\tau) - N \cdot \log \sum_{\tau'} e^{R_{\theta}(\tau')}$$

$$\nabla_{\theta} llh = \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \nabla_{\theta} \log \sum_{\tau'} e^{R_{\theta}(\tau')} =$$

$$= \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \frac{1}{\sum_{\tilde{\tau}} e^{R_{\theta}(\tilde{\tau})}} \sum_{\tau'} e^{R_{\theta}(\tau')} \cdot \nabla_{\theta} R_{\theta}(\tau') =$$


Maximum Entropy Inverse RL

Let's simplify:

$$llh = \sum_{\tau \in D} \log \frac{e^{R_{\theta}(\tau)}}{\sum_{\tau'} e^{R_{\theta}(\tau')}} =$$

$$= \sum_{\tau \in D} [R_{\theta}(\tau) - \log \sum_{\tau'} e^{R_{\theta}(\tau')}] = \sum_{\tau \in D} R_{\theta}(\tau) - N \cdot \log \sum_{\tau'} e^{R_{\theta}(\tau')}$$

$$\nabla_{\theta} llh = \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \nabla_{\theta} \log \sum_{\tau'} e^{R_{\theta}(\tau')} =$$

$$= \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \frac{1}{\sum_{\tilde{\tau}} e^{R_{\theta}(\tilde{\tau})}} \sum_{\tau'} e^{R_{\theta}(\tau')} \cdot \nabla_{\theta} R_{\theta}(\tau') =$$


Maximum Entropy Inverse RL

Let's simplify:

$$llh = \sum_{\tau \in D} \log \frac{e^{R_{\theta}(\tau)}}{\sum_{\tau'} e^{R_{\theta}(\tau')}} =$$

$$= \sum_{\tau \in D} [R_{\theta}(\tau) - \log \sum_{\tau'} e^{R_{\theta}(\tau')}] = \sum_{\tau \in D} R_{\theta}(\tau) - N \cdot \log \sum_{\tau'} e^{R_{\theta}(\tau')}$$

$$\nabla_{\theta} llh = \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \nabla_{\theta} \log \sum_{\tau'} e^{R_{\theta}(\tau')} =$$

Reminds of sth?

$$= \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \frac{1}{\sum_{\tilde{\tau}} e^{R_{\theta}(\tilde{\tau})}} \sum_{\tau'} e^{R_{\theta}(\tau')} \cdot \nabla_{\theta} R_{\theta}(\tau') =$$

Maximum Entropy Inverse RL

Let's simplify:

$$llh = \sum_{\tau \in D} \log \frac{e^{R_{\theta}(\tau)}}{\sum_{\tau'} e^{R_{\theta}(\tau')}} =$$

$$= \sum_{\tau \in D} [R_{\theta}(\tau) - \log \sum_{\tau'} e^{R_{\theta}(\tau')}] = \sum_{\tau \in D} R_{\theta}(\tau) - N \cdot \log \sum_{\tau'} e^{R_{\theta}(\tau')}$$

$$\nabla_{\theta} llh = \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \nabla_{\theta} \log \sum_{\tau'} e^{R_{\theta}(\tau')} =$$

$$= N \cdot \left[\mathbb{E}_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - \mathbb{E}_{\tau' \sim \pi^*(\tau'; R_{\theta})} \nabla_{\theta} R_{\theta}(\tau') \right]$$

Maximum Entropy Inverse RL

Let's simplify:

$$llh = \sum_{\tau \in D} \log \frac{e^{R_{\theta}(\tau)}}{\sum_{\tau'} e^{R_{\theta}(\tau')}} =$$

$$= \sum_{\tau \in D} [R_{\theta}(\tau) - \log \sum_{\tau'} e^{R_{\theta}(\tau')}] = \sum_{\tau \in D} R_{\theta}(\tau) - N \cdot \log \sum_{\tau'} e^{R_{\theta}(\tau')}$$

$$\nabla_{\theta} llh = \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \nabla_{\theta} \log \sum_{\tau'} e^{R_{\theta}(\tau')} =$$

$$= N \cdot \left[\underbrace{E_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau)}_{\text{where}} - E_{\tau' \sim \pi^*(\tau'; R_{\theta})} \nabla_{\theta} R_{\theta}(\tau') \right]$$

where $\pi^*(\tau'; R_{\theta}) \sim e^{R_{\theta}(\tau')}$

Tabular, model-based

Replace sum over trajectories...

$$\nabla_{\theta} llh = \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \cdot \mathbb{E}_{\tau' \sim \pi^*(\tau'; \theta)} \nabla_{\theta} R_{\theta}(\tau') =$$

... with sum over states

$$= \sum_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau) - N \cdot \mathbb{E}_{s \sim d_{\theta}(s)} \mathbb{E}_{a \sim \pi_{\theta}^*(a|s)} \nabla_{\theta} r_{\theta}(s, a)$$


**state visitation freq;
(stationary distribution)**

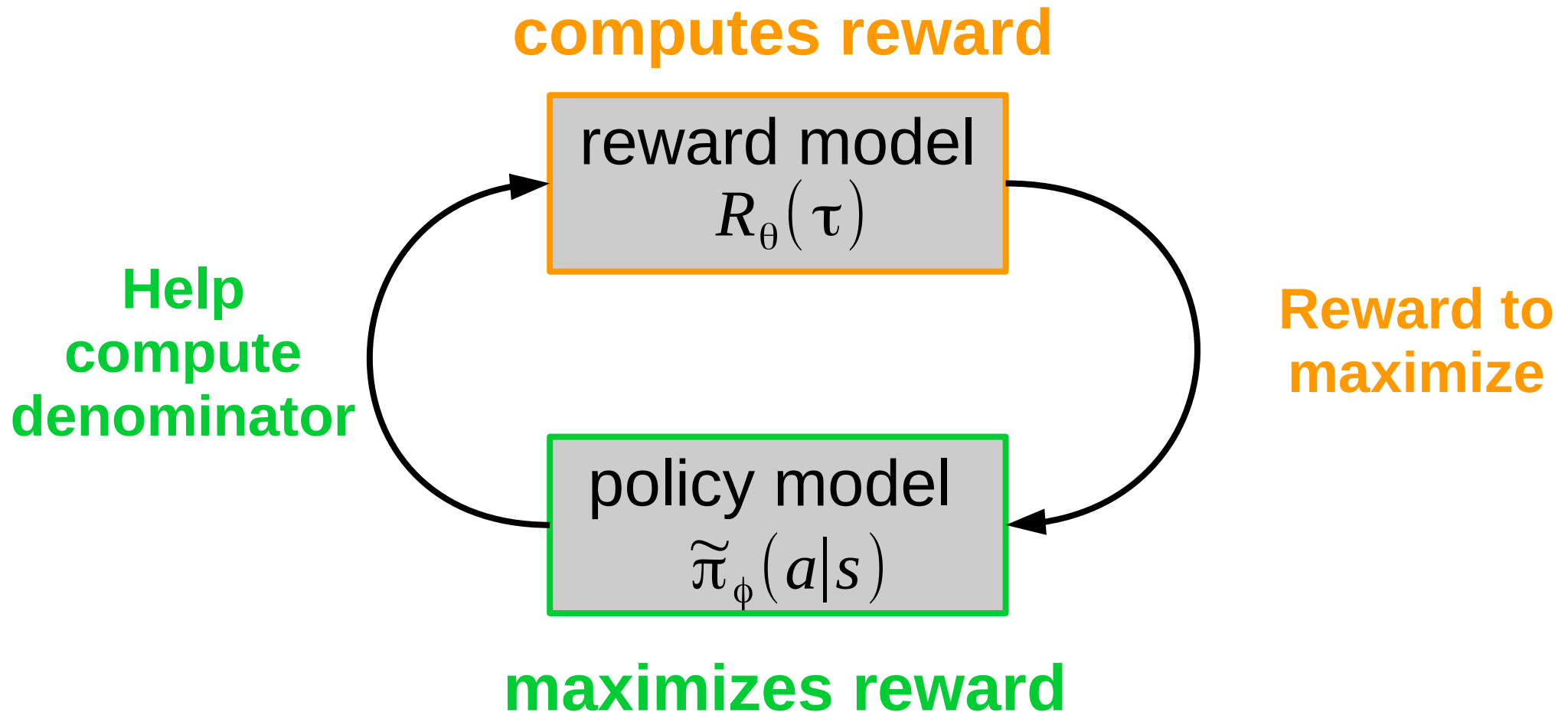
Model-free case

$$\nabla_{\theta} llh = N \cdot \left[\underbrace{E_{\tau \in D} \nabla_{\theta} R_{\theta}(\tau)}_{\text{sample from data}} - \underbrace{E_{\tau' \sim \pi^*(\tau'; R_{\theta})} \nabla_{\theta} R_{\theta}(\tau')}_{\text{hard to even sample}} \right]$$

To sample from $\pi^*(\tau'; R_{\theta}) \sim e^{R_{\theta}(\tau')}$

We need to estimate $\sum_{\tau'} e^{R_{\theta}(\tau')}$

Guided Cost Learning



Training policy

$$L_{\tilde{\pi}_{\phi}} = KL(\tilde{\pi}_{\phi}(\tau) \parallel \pi^*(\tau; R_{\theta})) =$$

Training policy

$$L_{\tilde{\pi}_\phi} = KL(\tilde{\pi}_\phi(\tau) \parallel \pi^*(\tau; R_\theta)) = E_{\tilde{\pi}_\phi(\tau)} \frac{\log \tilde{\pi}_\phi(\tau)}{\log \pi^*(\tau; R_\theta)} =$$

$$= E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \tilde{\pi}_\phi(\tau) - E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \pi^*(\tau; R_\theta) =$$

Training policy

$$L_{\tilde{\pi}_\phi} = KL(\tilde{\pi}_\phi(\tau) \parallel \pi^*(\tau; R_\theta)) = E_{\tilde{\pi}_\phi(\tau)} \frac{\log \tilde{\pi}_\phi(\tau)}{\log \pi^*(\tau; R_\theta)} =$$

$$= E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \tilde{\pi}_\phi(\tau) - E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \pi^*(\tau; R_\theta) =$$

$$= E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \tilde{\pi}_\phi(\tau) - E_{\tau \sim \tilde{\pi}_\phi(\tau)} R_\theta(\tau) - \log \sum_{\tau'} e^{R_\theta(\tau')}$$

Training policy

$$L_{\tilde{\pi}_\phi} = KL(\tilde{\pi}_\phi(\tau) \parallel \pi^*(\tau; R_\theta)) = E_{\tau \sim \tilde{\pi}_\phi(\tau)} \frac{\log \tilde{\pi}_\phi(\tau)}{\log \pi^*(\tau; R_\theta)} =$$

$$= E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \tilde{\pi}_\phi(\tau) - E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \pi^*(\tau; R_\theta) =$$

$\log(e^R)$
numerator denominator

$$= E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \tilde{\pi}_\phi(\tau) - E_{\tau \sim \tilde{\pi}_\phi(\tau)} R_\theta(\tau) - \log \sum_{\tau'} e^{R_\theta(\tau')}$$

Anything
peculiar?

???

???

Training policy

$$L_{\tilde{\pi}_{\phi}} = KL(\tilde{\pi}_{\phi}(\tau) \parallel \pi^*(\tau; R_{\theta})) = E_{\tau \sim \tilde{\pi}_{\phi}(\tau)} \frac{\log \tilde{\pi}_{\phi}(\tau)}{\log \pi^*(\tau; R_{\theta})} =$$

$$= E_{\tau \sim \tilde{\pi}_{\phi}(\tau)} \log \tilde{\pi}_{\phi}(\tau) - E_{\tau \sim \tilde{\pi}_{\phi}(\tau)} \log \pi^*(\tau; R_{\theta}) =$$

log(e^R)
numerator denominator

$$= E_{\tau \sim \tilde{\pi}_{\phi}(\tau)} \log \tilde{\pi}_{\phi}(\tau) - E_{\tau \sim \tilde{\pi}_{\phi}(\tau)} R_{\theta}(\tau) - \log \sum_{\tau'} e^{R_{\theta}(\tau')}$$

Anything
peculiar?

- entropy

main stuff

const(ϕ)!

Training them both

$$L_{\tilde{\pi}_\phi} = KL(\tilde{\pi}_\phi(\tau) \parallel \pi^*(\tau; R_\theta)) = E_{\tilde{\pi}_\phi(\tau)} \frac{\log \tilde{\pi}_\phi(\tau)}{\log \pi^*(\tau; R_\theta)} =$$

$$= E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \tilde{\pi}_\phi(\tau) - E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \pi^*(\tau; R_\theta) =$$

log(e^R)
numerator denominator

$$= E_{\tau \sim \tilde{\pi}_\phi(\tau)} \log \tilde{\pi}_\phi(\tau) - E_{\tau \sim \tilde{\pi}_\phi(\tau)} R_\theta(\tau) - \log \sum_{\tau'} e^{R_\theta(\tau')}$$

Anything
peculiar?

- entropy

main stuff

const(ϕ)!

Training them both

- Update $R_\theta(\tau)$ under current $\tilde{\pi}_\phi(a|s)$

$$\nabla_\theta \text{llh} = N \cdot \left[\mathbb{E}_{\tau \in D} \nabla_\theta R_\theta(\tau) - \mathbb{E}_{\tau' \sim \tilde{\pi}(\tau'; R_\theta)} \nabla_\theta R_\theta(\tau') \right]$$

- Update $\tilde{\pi}_\phi(a|s)$ under current $R_\theta(\tau)$

$$\nabla_\phi KL = - \mathbb{E}_{\tau \sim \tilde{\pi}_\phi(\tau)} \nabla \log \tilde{\pi}_\phi(\tau) \cdot (1 + R_\theta(\tau))$$

See also

- Generative Adversarial Imitation Learning
 - [arXiv:1606.03476](#) , Ho et al.
- Model-based Adversarial Imitation learning
 - [arXiv:1612.02179](#) , Baram et al.
- Cooperative Inverse Reinforcement Learning
 - [arXiv:1606.03137](#) , Hadfield-Menel et al.
 - Agent learns to understand human's goal and assist

a whole lot of other stuff, just google