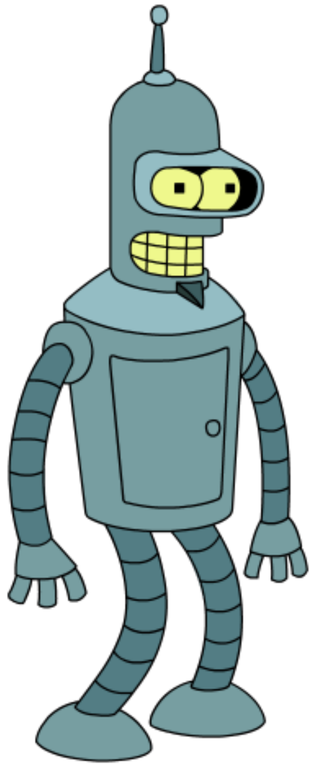
Reinforcement Learning

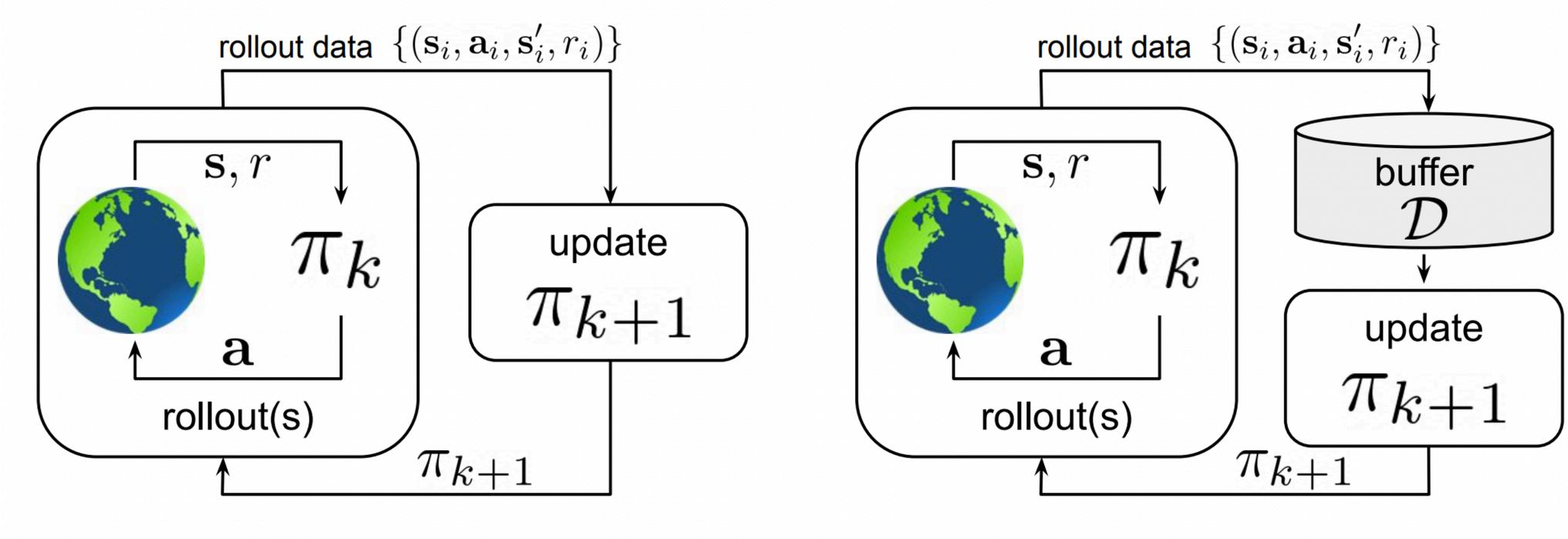
HSE, winter - spring 2025

Lecture 7: Offline RL

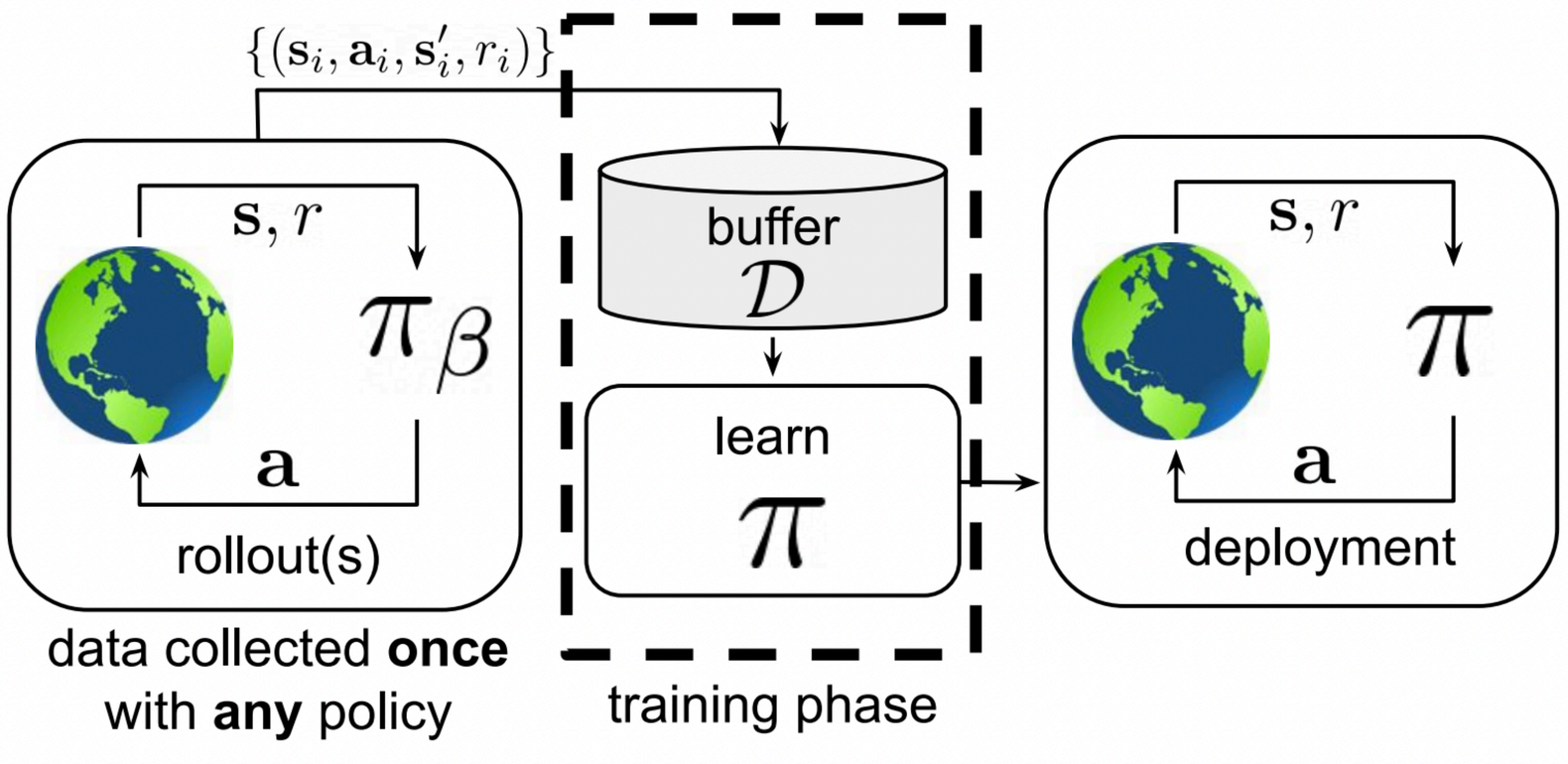


Sergei Laktionov
slaktionov@hse.ru
LinkedIn

Recap: On-policy vs Off-policy



Offline RL



$D = \{(s_i, a_i, r_i, s'_i)\}$

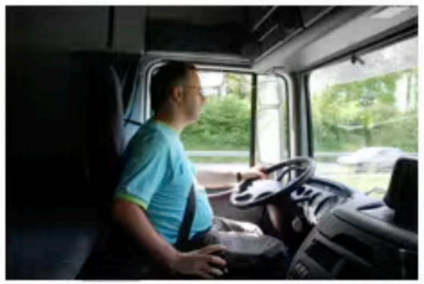
1. $a \sim \pi_\beta$
2. $s \sim d_{\pi_\beta}$
3. $s' \sim p(\,.\,|\,s, a)$
4. $r = r(s, a)$

Offline RL

Offline RL algorithms must do two things:

1. Maximise expected cumulative reward
2. Learn π_θ which is close to π_β

Imitation Learning



$\mathbf{s}_t$
$\mathbf{a}_t$

$\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$

$$\sum_{(s,a)\in D} \log \pi_\theta(a\,|\,s) \rightarrow \max_\theta$$

Imitation Learning

$$\sum_{(s,a) \in D} \log \pi_\theta(a \mid s) \rightarrow \max_\theta$$

+ Very stable

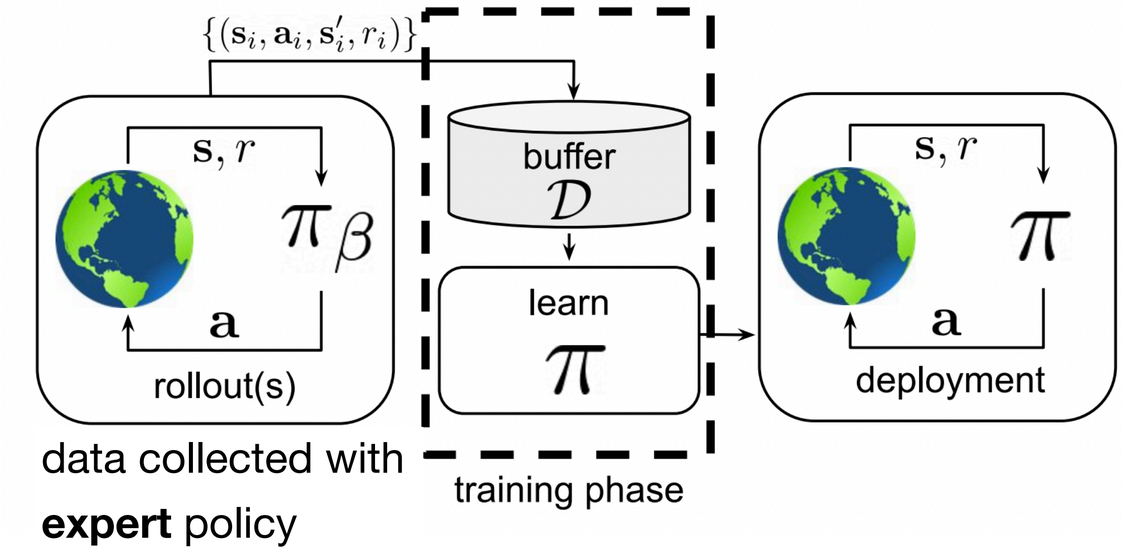
+ Scalable

+ Easy to implement and debug

— Does not maximise the reward

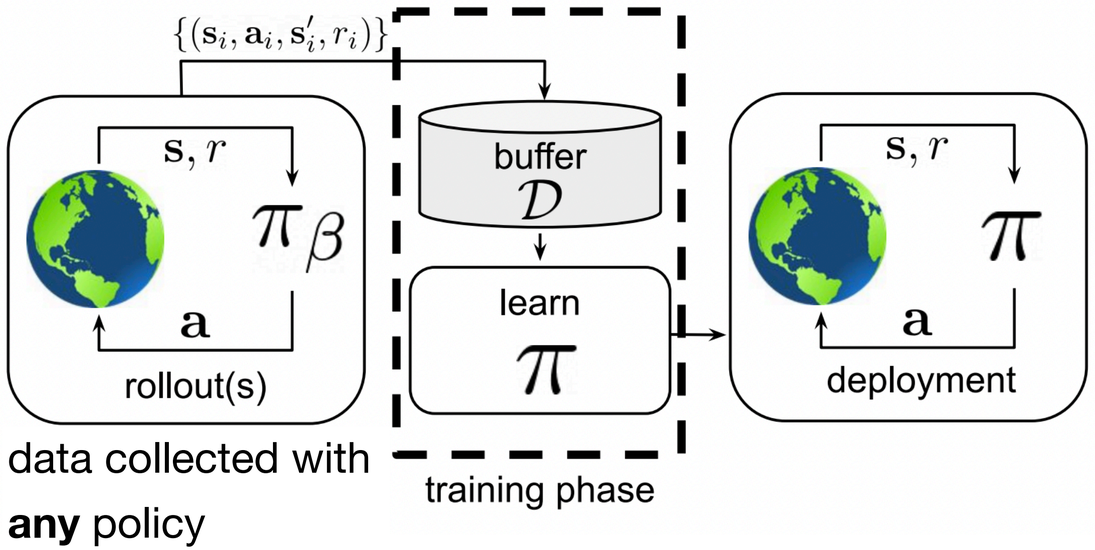
Imitation Learning vs Offline RL

Imitation Learning

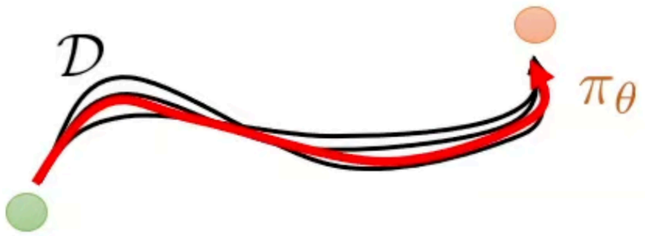


data collected with **expert** policy

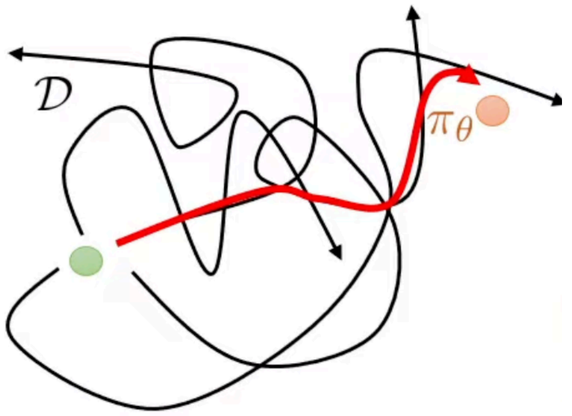
Offline RL



data collected with **any** policy

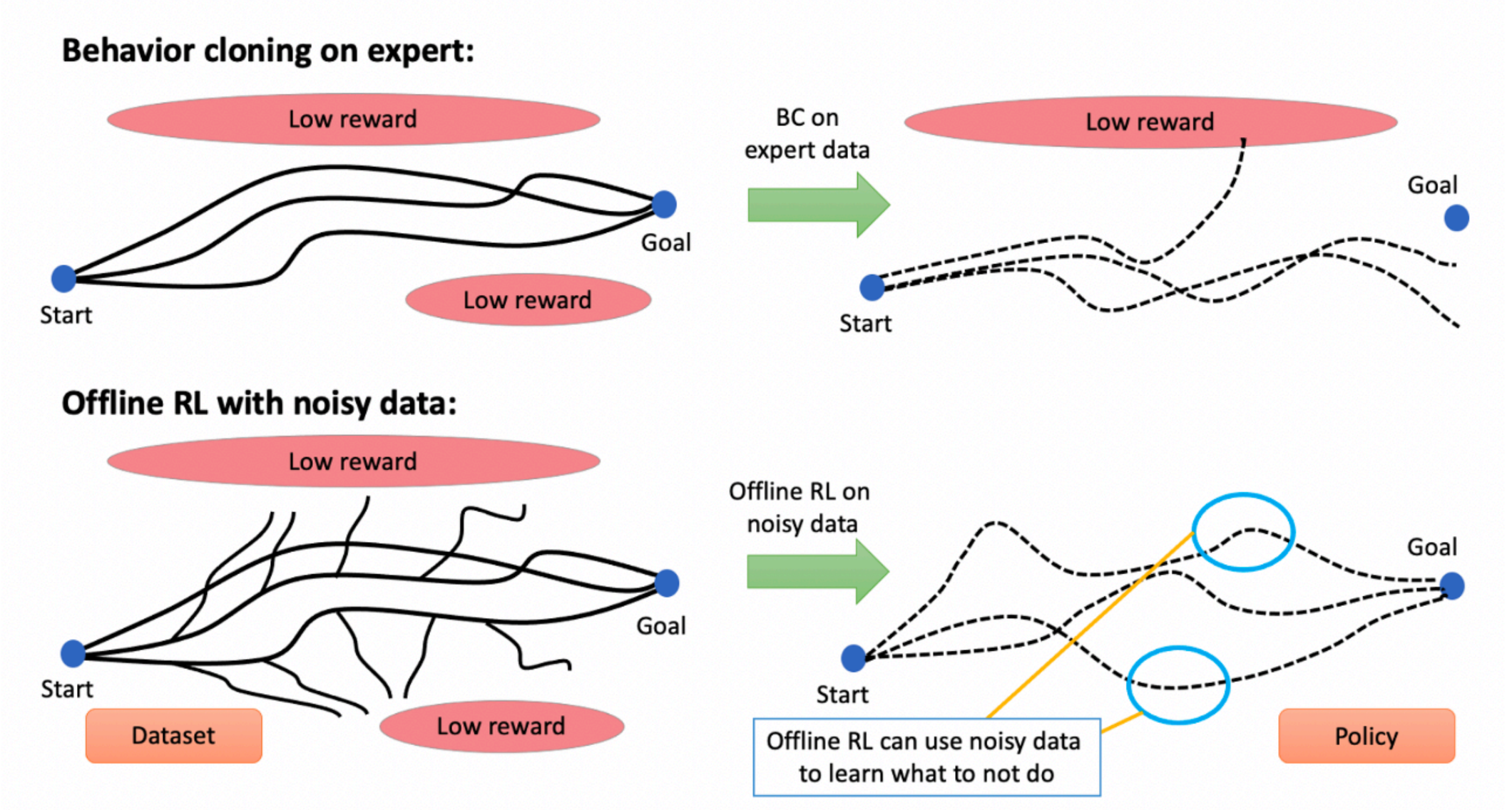


Imitate the data collected by experts



Extract the insights from noisy data

Imitation Learning vs Offline RL



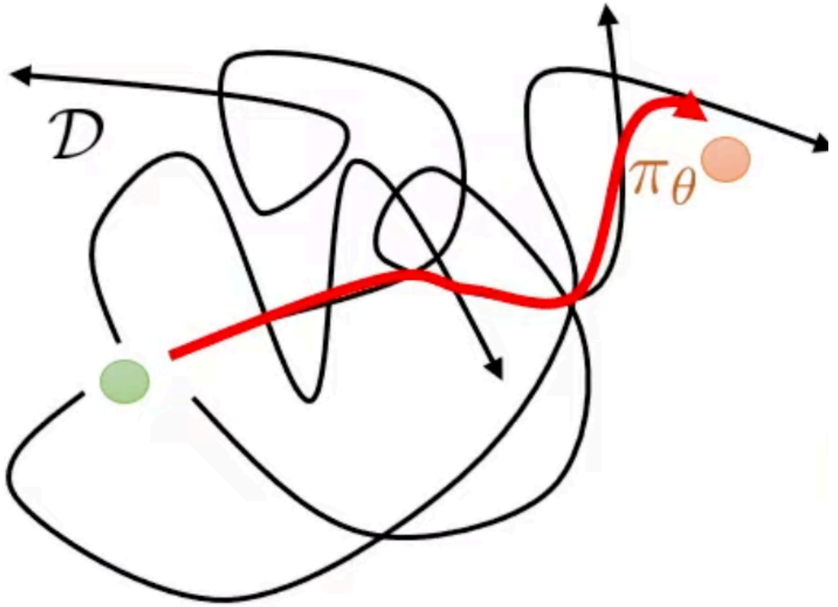
When should we prefer Offline RL over Behavioral Cloning?

Order from Chaos

1. Extract good trajectories from both good and bad demonstrations

2. Generalise: good behaviour in one state may suggest good behaviour in another state

3. Recombine the parts of good behaviour in different scenarios



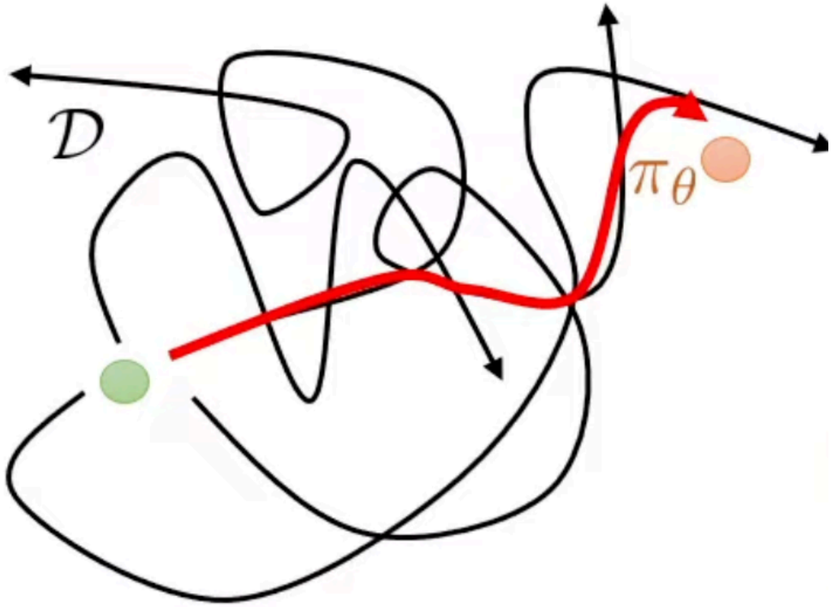
Order from Chaos

1. Extract good trajectories from both good and bad demonstrations

2. Generalise: good behaviour in one state may suggest good behaviour in another state

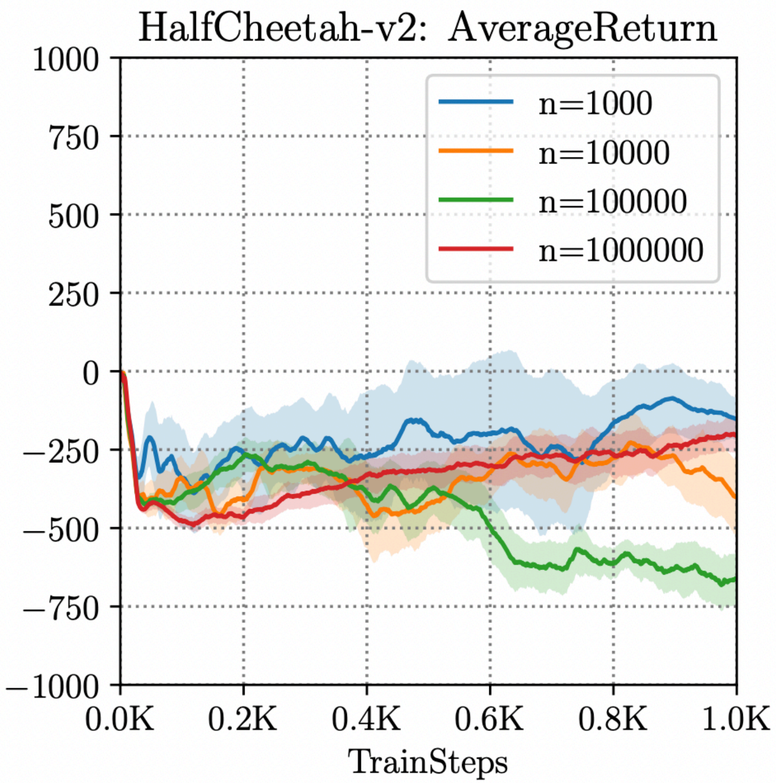
3. Recombine the parts of good behaviour in different scenarios

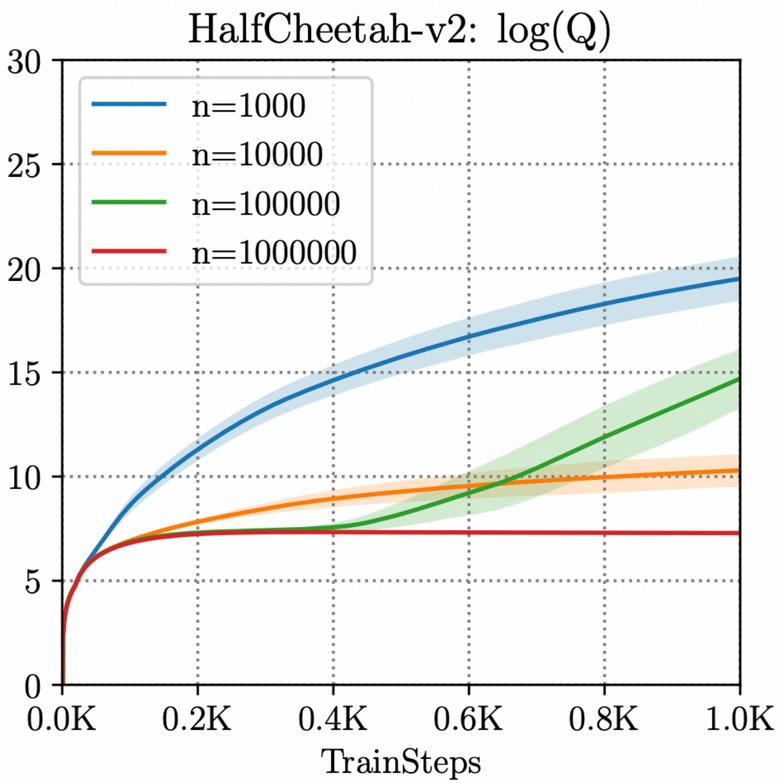
In theory, any off-policy method could be used. However they were devised under the assumption that further online interactions are possible and thus often fail.



Bootstrapping Error

Performance of SAC



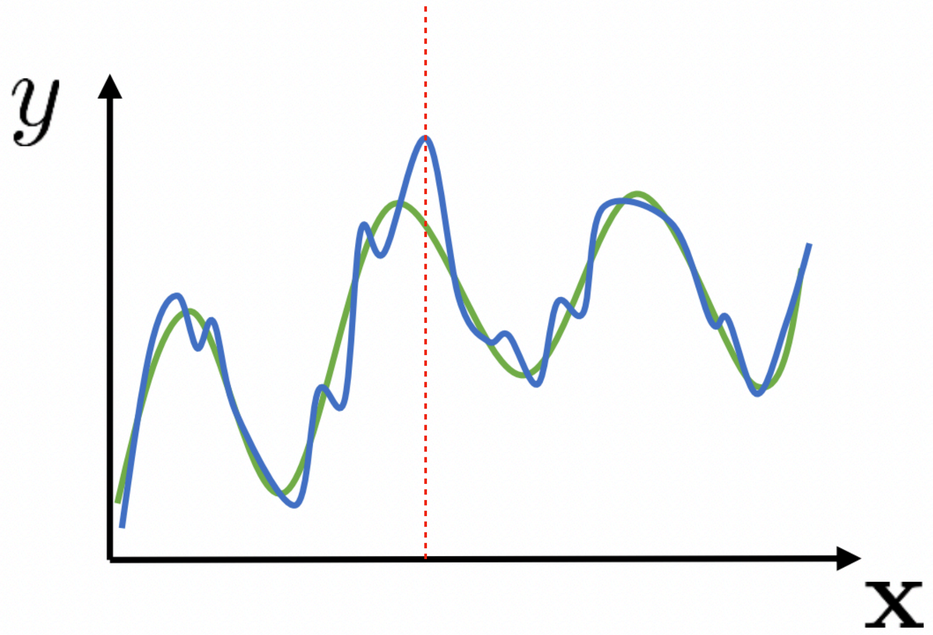


Stabilizing Off-Policy Q-Learning via Bootstrapping Error Reduction

Distribution Shift

$$y(s,a) = r + \gamma \mathbb{E}_{a' \sim \pi_{new}}[Q_{\phi^-}(s',a')]$$

$$\mathbb{E}_{s \sim d_{\pi_\beta}, a \sim \pi_\beta}[y(s,a) - Q_\phi(s,a)]^2 \rightarrow \min_\phi$$



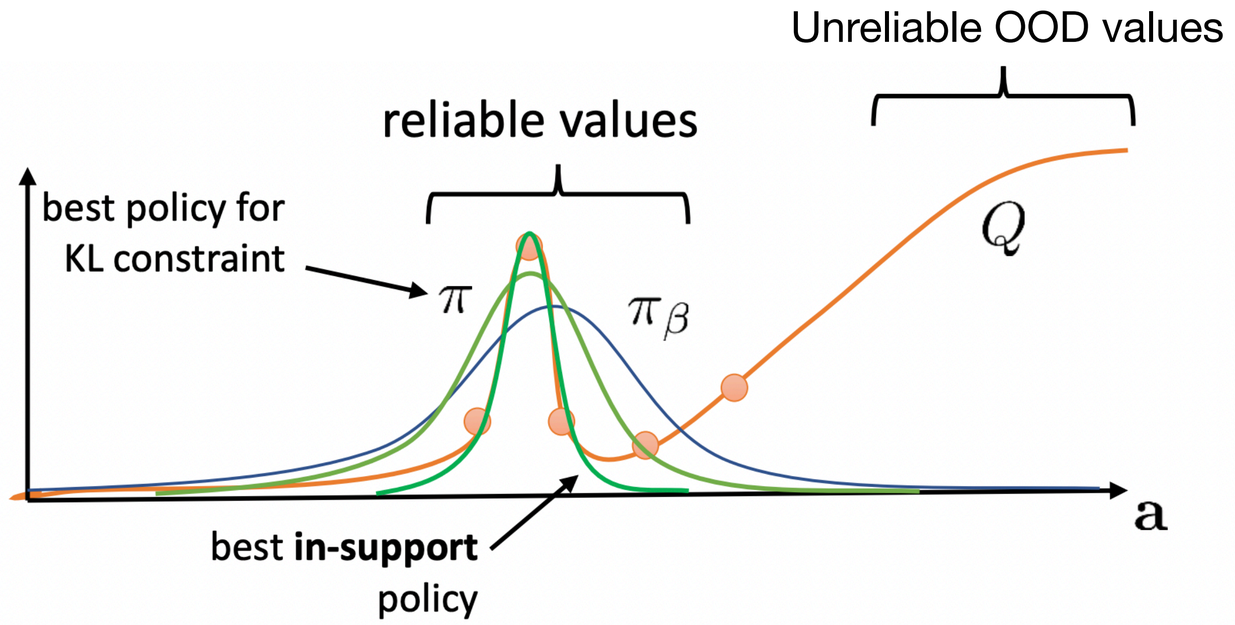
Explicit Policy Constraints

Actor objective:

$$\mathbb{E}_{s\sim D}\mathbb{E}_{a\sim\pi(.|s)}[A(s,a)] \rightarrow \max_{\pi}$$

KL constraint:

$$D_{KL}(\pi(\,.\,|\,s)\,||\,\pi_\beta(\,.\,|\,s)) \leq \varepsilon$$



Behavior Regularized Offline Reinforcement Learning

Explicit Policy Constraints

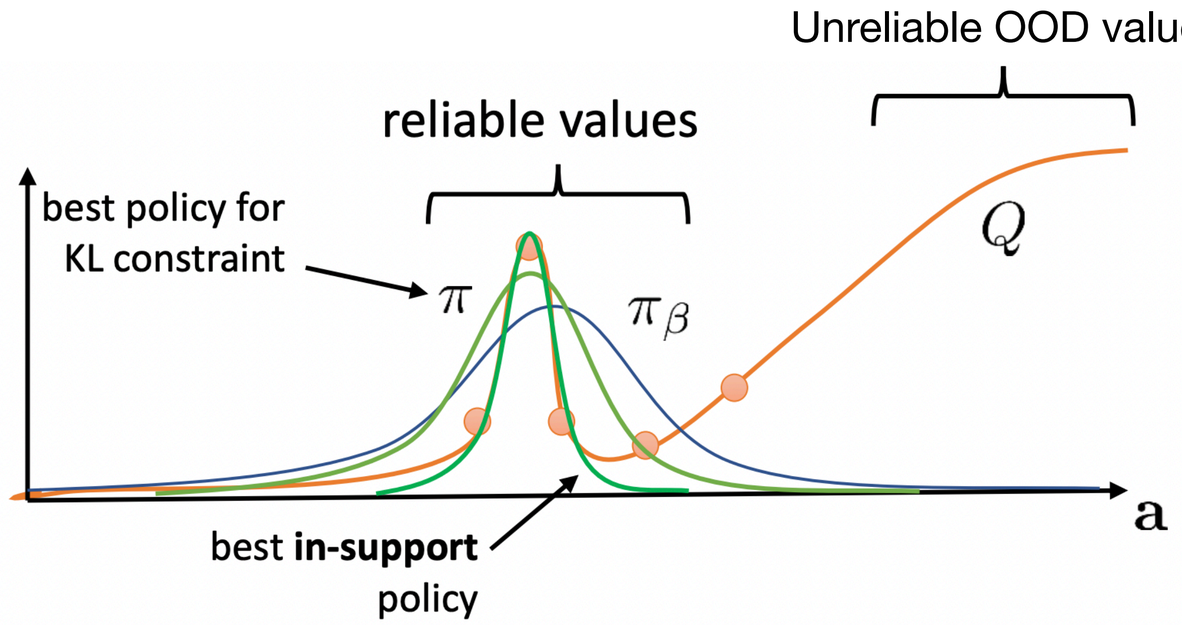
Actor objective:

$$\mathbb{E}_{s\sim D}\mathbb{E}_{a\sim\pi(.|s)}[A(s,a)] \rightarrow \max_{\pi}$$

KL constraint:

$$D_{KL}(\pi(\,.\,|\,s)\,||\,\pi_\beta(\,.\,|\,s)) \leq \varepsilon$$

Support constraint: $\pi(a\,|\,s) > 0$ only if $\pi_\beta(a\,|\,s) > 0$



Behavior Regularized Offline Reinforcement Learning

Implicit Policy Constraints

$$\mathbb{E}_{a\sim\pi(.|s)}[A^{\pi}(s,a)] - \lambda(D_{KL}(\pi(\,.\,|\,s)\,||\,\pi_{\beta}(\,.\,|\,s)) - \epsilon) - \mu(\int \pi(a\,|\,s)da - 1) \rightarrow \max_{\pi}$$

$$\pi^{*}(a\,|\,s) = \frac{1}{Z(s)}\pi_{\beta}(a\,|\,s)\exp\left(\frac{A^{\pi}(s,a)}{\lambda}\right)$$

Weighted max loglikelihood:

$$\pi^{new}(a\,|\,s) = argmax_{\pi}\mathbb{E}_{(s,a)\sim\pi_{\beta}}\left[\log\pi(a\,|\,s)\,\frac{1}{Z(s)}\,\exp\left(\frac{A^{\pi_{old}}(s,a)}{\lambda}\right)\right]$$

AWAC: Accelerating Online Reinforcement Learning with Offline Datasets

Advantage Weighted Actor Critic

Actor: $\mathbb{E}_{(s,a)\sim\pi_\beta}\Big[\log \pi_\theta(a\,|\,s)\,\dfrac{1}{Z(s)}\,\exp\Big(\dfrac{A_\phi(s,a)}{\lambda}\Big)\Big] \to \max_\theta,$

where $A_\phi(s,a) = Q_\phi(s,a) - Q_\phi(s,a'), a' \sim \pi_\theta(\,.\,|\,s).$

Critic: $\mathbb{E}_{(s,a,r,s')\sim D}\Big[r + \gamma \mathbb{E}_{a'\sim\pi_\theta(.|s')} Q_{\phi^-}(s',a') - Q_\phi(s,a)\Big]^2 \to \min_\phi,$

Advantage Weighted Actor Critic

Actor: $\mathbb{E}_{(s,a)\sim\pi_\beta}\left[\log \pi_\theta(a\,|\,s)\,\frac{1}{Z(s)}\,\exp\left(\frac{A_\phi(s,a)}{\lambda}\right)\right] \to \max_\theta,$

where $A_\phi(s,a) = Q_\phi(s,a) - Q_\phi(s,a'), a' \sim \pi_\theta(\,.\,|\,s).$

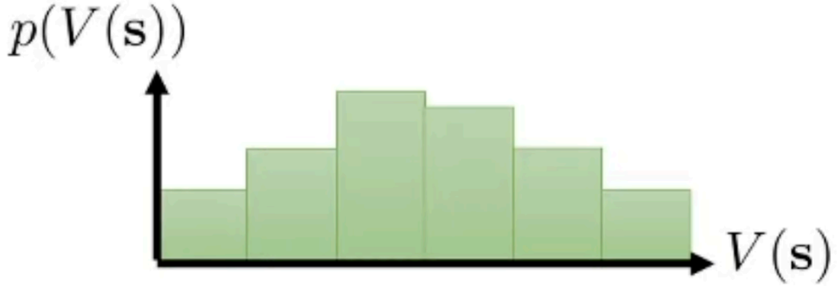
Critic: $\mathbb{E}_{(s,a,r,s')\sim D}\left[r + \gamma \mathbb{E}_{a'\sim\pi_\theta(.|s')} Q_{\phi^-}(s',a') - Q_\phi(s,a)\right]^2 \to \min_\phi,$

Can we also avoid OOD actions?

Implicit Q-Learning

$$\mathbb{E}_{a\sim\pi(.|s)}Q^{\pi}(s,a) = V^{\pi}(s)$$

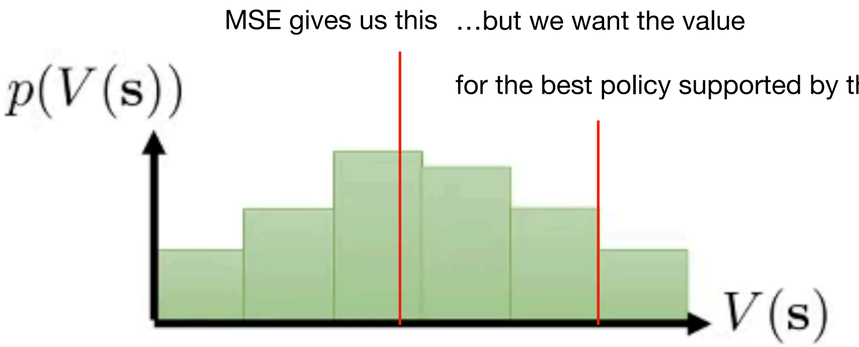
Approximate it with a neural network: $V^{\pi} \approx V_{\psi}$



Implicit Q-Learning

$$\mathbb{E}_{a\sim\pi(.|s)}Q^{\pi}(s,a) = V^{\pi}(s)$$

Approximate it with a neural network: $V^{\pi} \approx V_{\psi}$

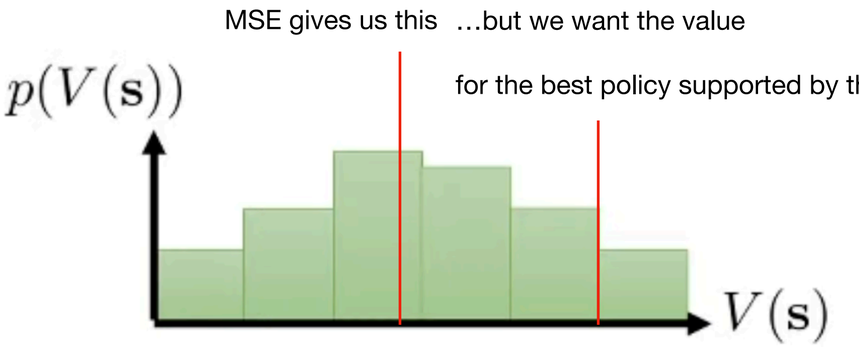


$$\mathbb{E}_{a\sim\pi_{\beta}}Q(s,a) \qquad \max_{a:\pi_{\beta}(a|s)\geq\epsilon} Q(s,a)$$

Implicit Q-Learning

$$\mathbb{E}_{a\sim\pi(.|s)}Q^{\pi}(s,a) = V^{\pi}(s)$$

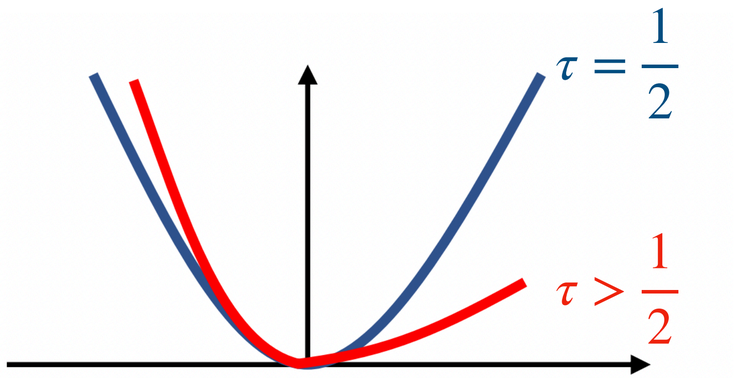
Approximate it with a neural network: $V^{\pi} \approx V_{\psi}$



$$\mathbb{E}_{a\sim\pi_{\beta}}Q(s,a) \qquad \max_{a:\pi_{\beta}(a|s)\geq\epsilon} Q(s,a)$$

Expectile loss:

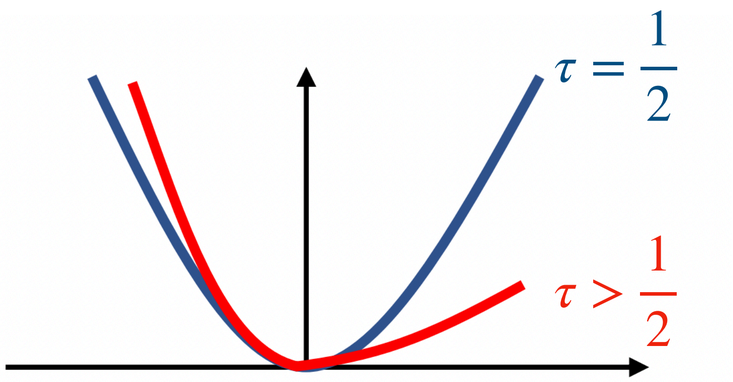
$$L_2^{\tau}(u) = |\tau - \mathbb{I}(u<0)|u^2$$



Implicit Q-Learning

Expectile loss:

$$L_2^\tau(u) = |\tau - \mathbb{I}(u < 0)|\, u^2$$



IQL:

$$L(\psi) = \mathbb{E}_{(s,a)\sim D}[L_2^\tau(Q_{\hat{\phi}}(s,a) - V_\psi(s))]$$

$$L(\phi) = \mathbb{E}_{(s,a,r,s')\sim D}[r + \gamma V_\psi(s') - Q_\phi(s,a)]^2$$

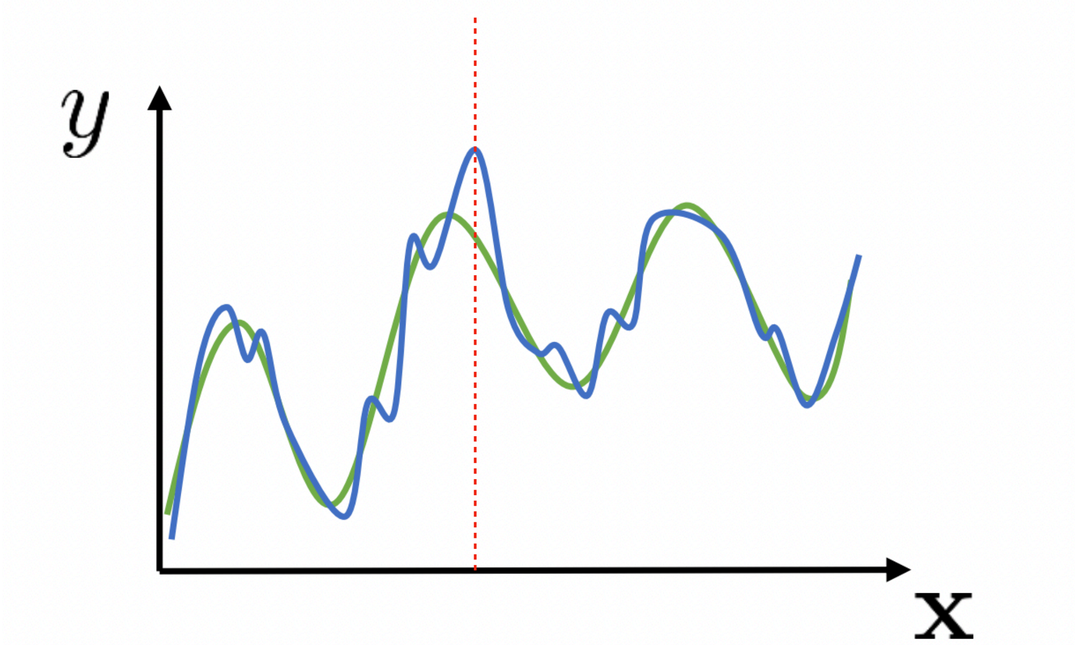
$$L(\theta) = -\mathbb{E}_{(s,a)\sim D}\Big[\log \pi_\theta(a\,|\,s)\, \frac{1}{Z(s)}\, \exp\Big(\frac{A(s,a)}{\lambda}\Big)\Big]$$

where $A(s,a) = Q_\phi(s,a) - V_\psi(s)$

Conservative Q-Learning

$$L_{CQL} = \max_{\mu}[\alpha(\mathbb{E}_{s\sim D, a\sim\mu(.|s)}Q_\phi(s,a))]+$$

$$+\frac{1}{2}\mathbb{E}_{(s,a,r,s')\sim D}[r+\gamma\mathbb{E}_{a'\sim\pi}Q_{\phi^-}(s',a')-Q_\phi(s,a)]^2 \rightarrow \min_{\phi}$$



It can be shown that for large enough α

$$Q_\phi \leq Q^\pi$$

Conservative Q-Learning

$$L_{CQL} = \max_{\mu}[\alpha(\mathbb{E}_{s\sim D, a\sim\mu(.|s)}Q_\phi(s,a)\ ] - \alpha\mathbb{E}_{(s,a)\sim D}Q_\phi(s,a) +$$

$$+\frac{1}{2}\mathbb{E}_{(s,a,r,s')\sim D}[r + \gamma\mathbb{E}_{a'\sim\pi}Q_{\phi^-}(s',a') - Q_\phi(s,a)]^2 \rightarrow \min_{\phi}$$

It can be shown that for large enough α

$$\mathbb{E}_\pi Q_\phi \leq \mathbb{E}_\pi Q^\pi$$

Conservative Q-Learning

$$L_{CQL} = \max_{\mu}[\alpha(\mathbb{E}_{s\sim D, a\sim\mu(.|s)}Q_\phi(s,a)\ ) + R(\mu)] - \alpha\mathbb{E}_{(s,a)\sim D}Q_\phi(s,a) +$$

$$+\frac{1}{2}\mathbb{E}_{(s,a,r,s')\sim D}[r + \gamma\mathbb{E}_{a'\sim\pi}Q_{\phi^-}(s',a') - Q_\phi(s,a)]^2 \rightarrow \min_{\phi}$$

Conservative Q-Learning

Common choice $R(\mu) = \mathbb{E}_{s\sim D} H(\mu(\,.\,|\,s))$

$$\mathbb{E}_{s\sim D, a\sim\mu(.|s)}[Q_\phi(s,a) + H(\mu(\,.\,|\,s))] \to \max_\mu \implies \mu(a\,|\,s) \propto \exp(Q_\phi(s,a))$$

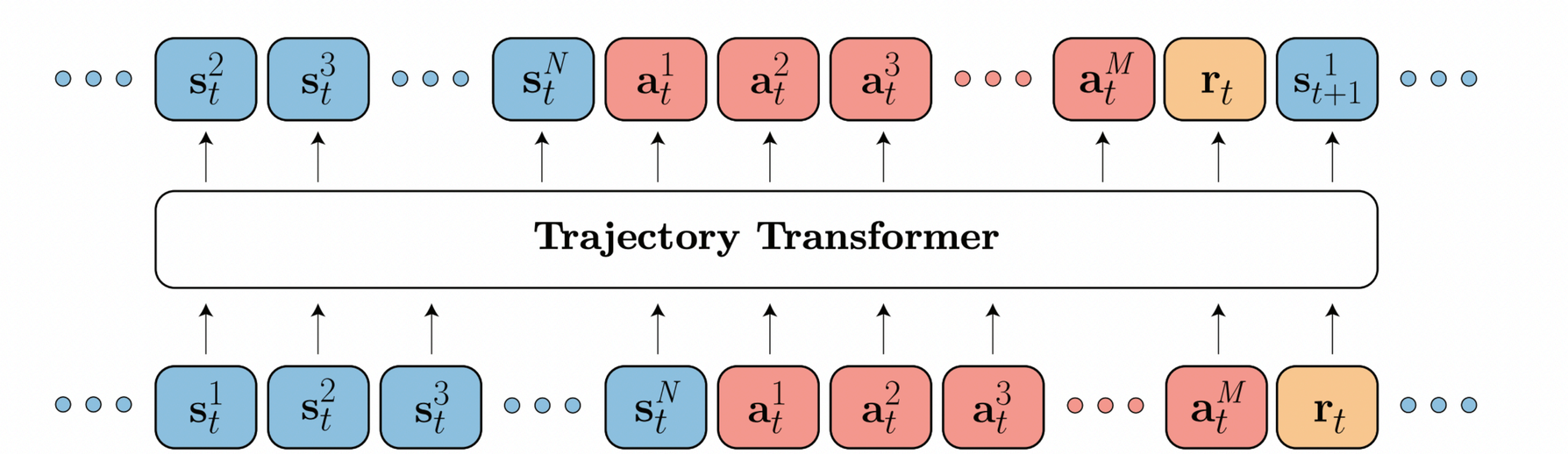
$$L_{CQL} = \alpha(\mathbb{E}_{s\sim D} \log \sum_a \exp(Q(s,a))) - \alpha \mathbb{E}_{(s,a)\sim D} Q_\phi(s,a)$$

$$+\frac{1}{2}\mathbb{E}_{(s,a,r,s')\sim D}[r + \gamma \mathbb{E}_{a'\sim\pi} Q_{\phi^-}(s',a') - Q_\phi(s,a)]^2 \to \min_\phi$$

1. Learn Q_ϕ optimising L_{CQL} on offline data

2. Update policy with SAC-style entropy regularization:
   $\mathbb{E}_{s\sim D, a\sim\pi_\theta}[Q_\phi(s,a) - \log \pi_\theta(a\,|\,s)] \to \max_\theta$

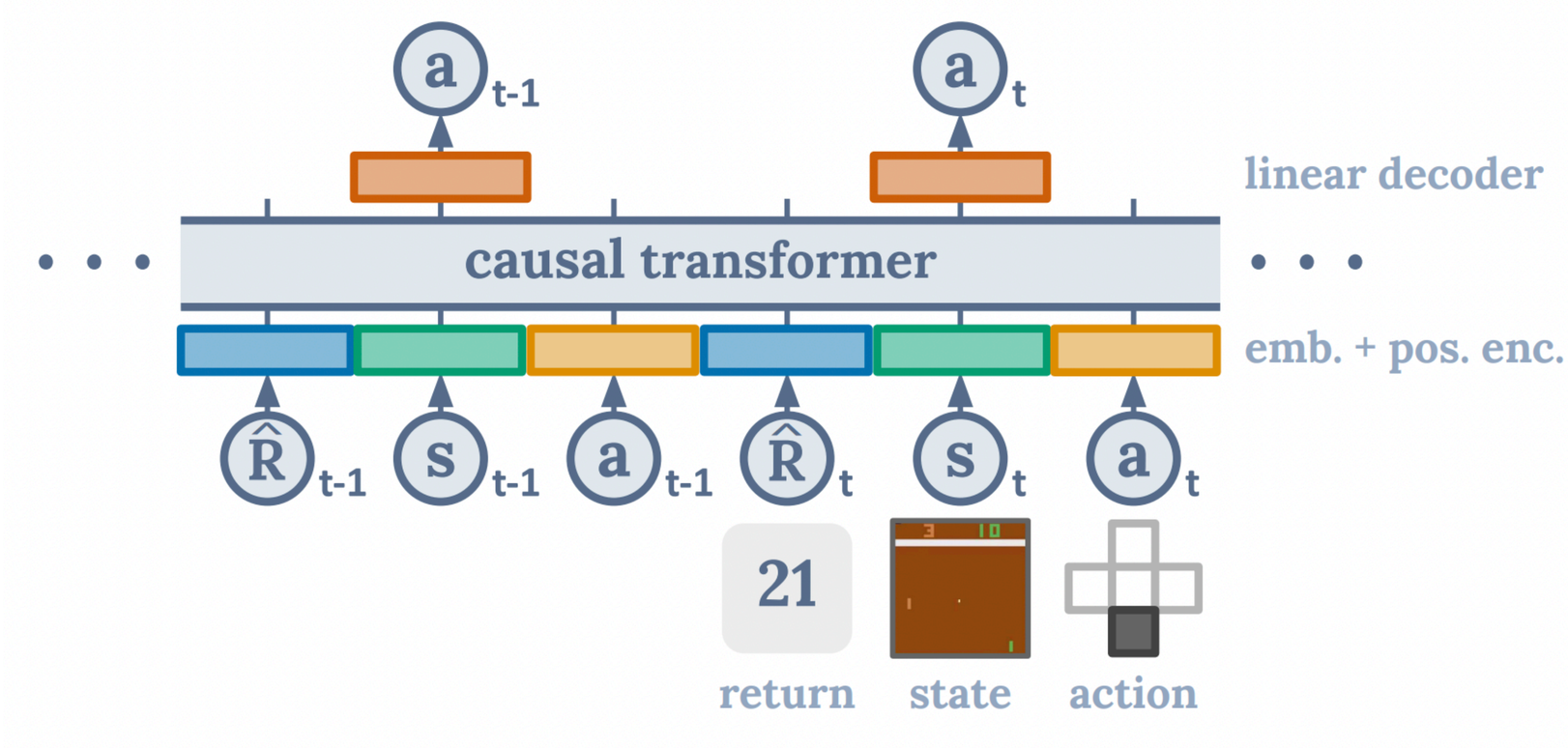
Trajectory Transformer



$$\mathcal{L}(\tau) = \sum_{t=1}^{T} \Big(\sum_{i=1}^{N} \log P_\theta \big(s_t^i \mid \mathbf{s}_t^{<i}, \boldsymbol{\tau}_{<t}\big) + \sum_{j=1}^{M} \log P_\theta \big(a_t^j \mid \mathbf{a}_t^{<j}, \mathbf{s}_t, \boldsymbol{\tau}_{<t}\big) + \log P_\theta \big(r_t \mid \mathbf{a}_t, \mathbf{s}_t, \boldsymbol{\tau}_{<t}\big) \Big),$$

Planning with Beam Search using sum of rewards

Decision Transformer



Recap: TD3

1. **Deterministic Policy Gradient:**

$$\mathbb{E}_{s\sim D}[Q_\phi(s, \pi_\theta(s))] \rightarrow \max_\theta$$

2. **Clipped Double-Q Learning:**

$$y = r + \gamma \min_{i=1,2} Q_{\phi_i^-}(s', \mu_{\theta^-}(s'))$$

3. **Delayed Policy Updates:** Update the policy (and target networks) less frequently than the Q-function.

$$y = r + \gamma \min_{i=1,2} Q_{\phi_i^-}(s', \mu_{\theta^-}(s'))$$

4. **Target Policy Smoothing:**

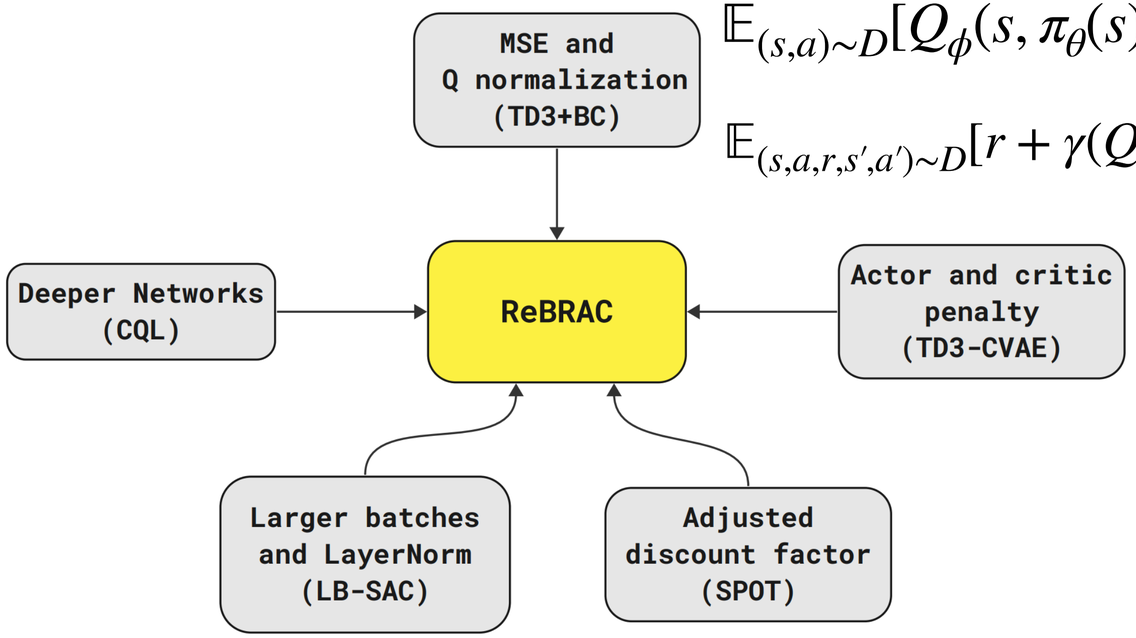
$$y = r + \gamma \min_{i=1,2} Q_{\phi_i^-}(s', a'), a' = \mu_{\theta^-}(s) + \varepsilon',$$
$$\varepsilon' \sim clip(\mathcal{N}(0,\sigma' I), -c, c)$$

The Minimalist Approach to Offline RL

TD3+BC:

$$\mathbb{E}_{(s,a)\sim D}[\lambda Q_\phi(s, \pi_\theta(s)) - (a - \pi_\theta(s))^2] \to \max_\theta$$

ReBRAC:



$$\mathbb{E}_{(s,a)\sim D}[Q_\phi(s, \pi_\theta(s)) - \beta_1(a - \pi(s))^2] \to \max_\theta$$

$$\mathbb{E}_{(s,a,r,s',a')\sim D}[r + \gamma(Q_{\phi^-}(s', a') - \beta_2(a' - \pi(s'))^2) - Q_\phi(s, a))^2] \to \min_\phi$$

Not Covered Yet

1. Uncertainty-Based Offline Reinforcement Learning with Diversified Q-Ensemble
2. Anti-Exploration by Random Network Distillation

Background

1. Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems

2. RL course by Sergey Levine, Lectures 15-16, Playlist

3. CORL: Research-oriented Deep Offline Reinforcement Learning Library

Thank you for your attention!