Advanced Topics in Reinforcement Learning

Lecture 25: Offline RL I

Josiah Hanna
University of Wisconsin — Madison

Announcements

- Next week: RL application
- Final projects due ~ 1 week.
- Please complete the course evaluation! At 14% right now.
- Today:
  - Introduce offline RL problem setting, objectives, and challenges.
  - Describe 3 classes of offline RL methods.
- Thursday:
  - Advanced offline RL challenges.
  - Off-policy Evaluation.

Offline RL Introduction

- Online RL is what we have covered so far.
- Exploration makes online RL slow.
  - Have to collect data before any learning can begin and more data as the policy changes.
  - I.e., (s,a,s',r) tuples.
- What if we already have data available to us?
  - Offline RL is RL applied to a static dataset of (s,a,s',r) tuples without additional exploration.
  - Also called "Batch RL." Batch RL is the older term and offline RL has gained prominence recently along with much attention.

Offline RL Motivation

- Modern machine learning is being driven by 1) enormous data sets and 2) large neural networks.
- Collecting a large data set through task interaction often takes a long time.
- What if we have existing data from:
  - Previously used policies (possibly non-RL policies)?
  - Other tasks?
  - Data from humans (e.g., YouTube videos of people cooking dinner)?
- Many potential applications:
  - Self-driving cars (large amounts of data available).
  - Healthcare (exploration is limited or impossible).
  - Robotics (diverse data available).

Offline RL Formalism

- Assume the target task can be described as an MDP.
    - More on partial observability next class.
- A *behavior policy, $\pi_\beta(a \mid s)$,* has collected dataset $\mathscr{D} = \{(s_i, a_i, s'_i, r_i)\}_{i=1}^{m}$.
    - Possibly multiple behavior policies and possibly unknown to us.
- Goal: Use $\mathscr{D}$ to learn policy, π, that maximizes expected return when deployed on the target task.

Ransheng’s Presentation

- Slides

Challenges

- Distribution shift: distribution of data in $\mathscr{D}$ is different than it would be if $\mathscr{D}$ was collected with the current policy, π.
  - Similar challenge for any off-policy RL algorithm but more extreme in offline RL.
- Missing data for some actions.
  - Should we take or avoid those actions?

Image credit: Sergey Levine's Offline RL Lecture

Warm-up: Imitation learning

- Given $\mathscr{D}$, we can attempt to just mimic the behavior policy that generated the data.

  Supervised learning not reinforcement learning.

- $\pi \leftarrow \arg\max_{\pi} \sum_{i=1}^{m} \log \pi(a_i \mid s_i)$

- (Sort of) robust to distribution shift.

  Performance loss can be quadratic in the episode length.

- Limitations:
  - Cannot improve upon π_β (and may do worse).
  - Causal confusion.

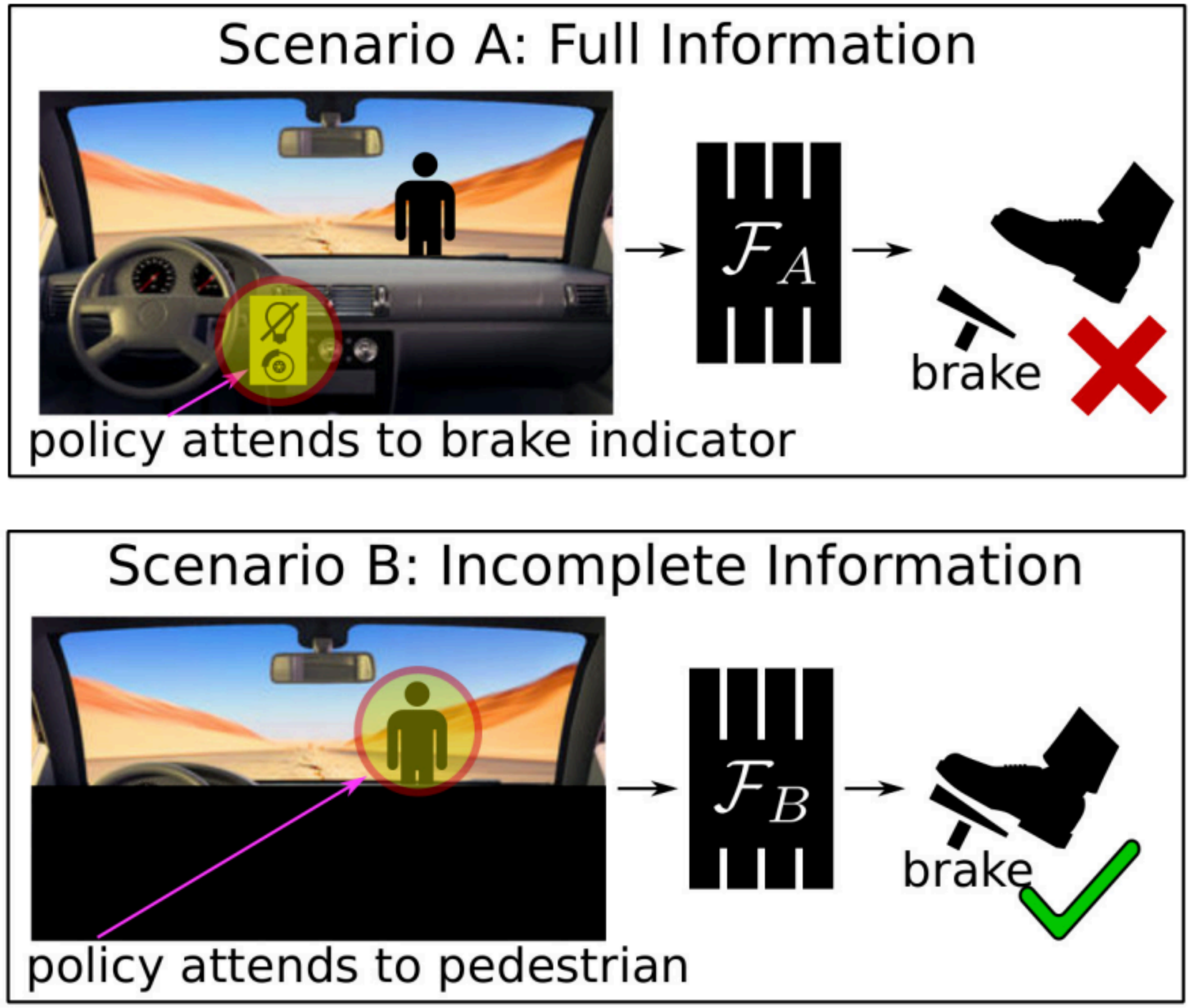
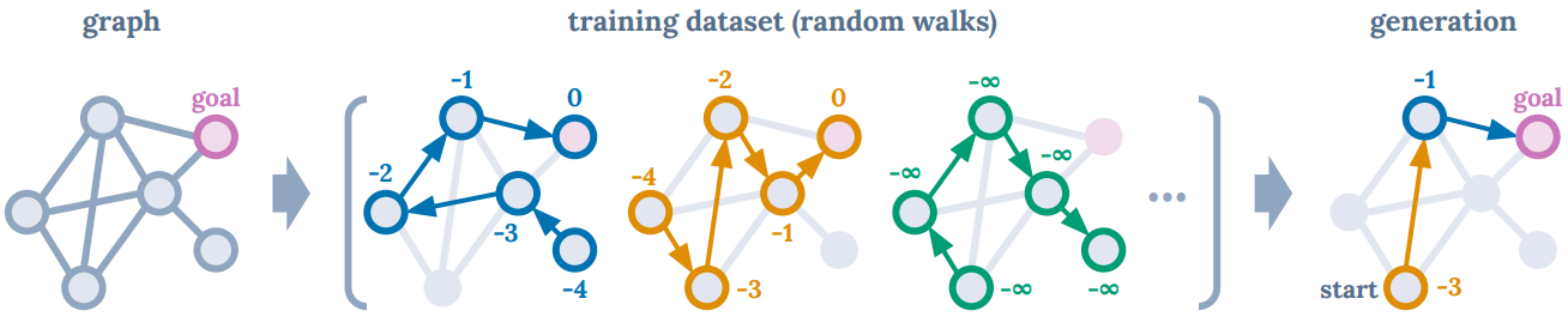


Image credit: Causal Confusion in Imitation Learning. De Haan et al. 2019.

What do we want in offline RL?

- Offline RL should improve upon π_β.
- Combine the best parts of sub-optimal behaviors.



Decision Transformer: Reinforcement Learning via Sequence Modeling. Chen et al. 2021.

Offline RL Method Classes

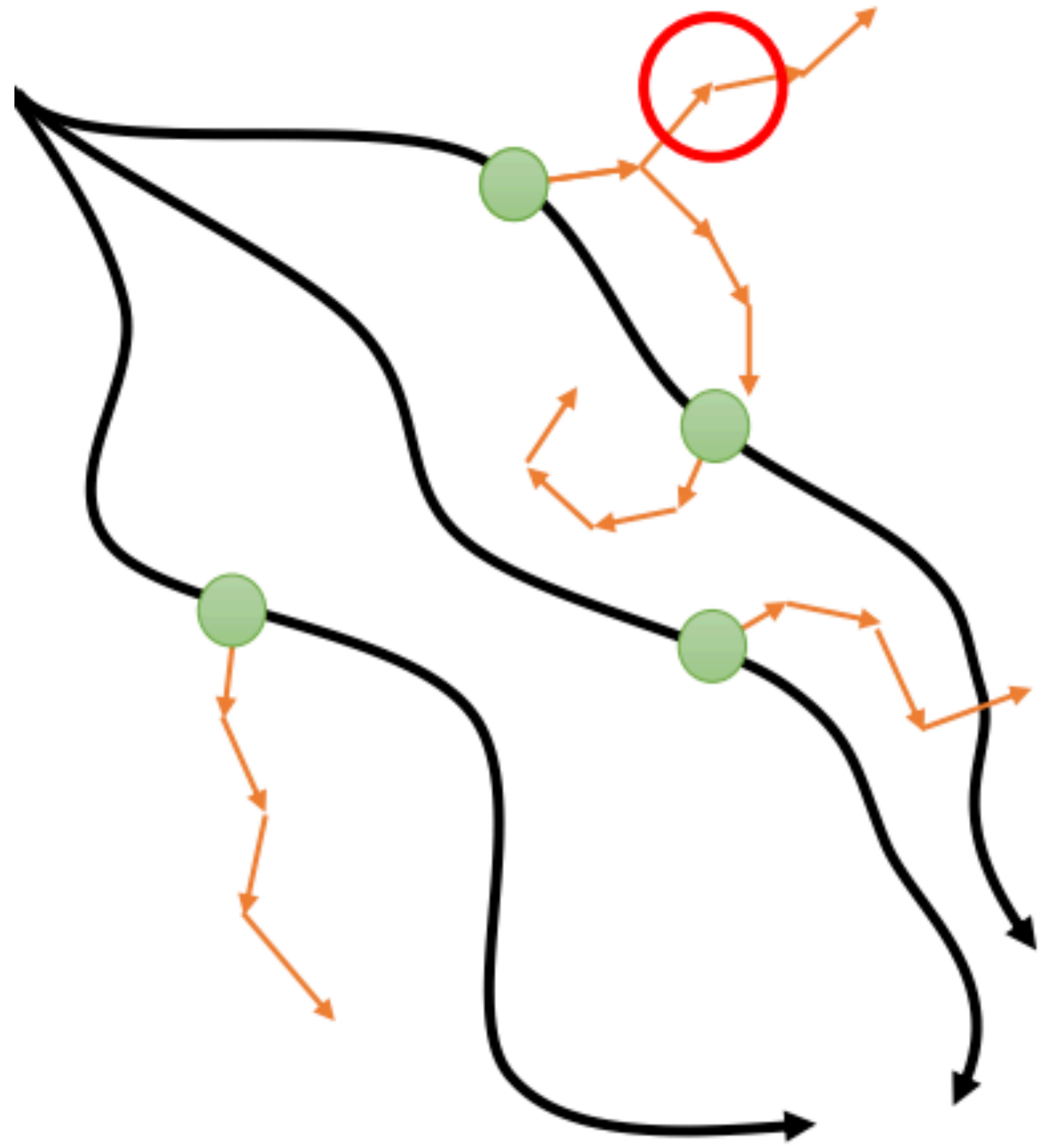
- Importance sampling for policy gradient methods.
- Model-based policy optimization.
- Action-value offline RL methods.
- Decision transformers.

Policy Gradients via Importance Sampling

- Recall policy gradient learning:
  - $\nabla_\theta J(\pi_\theta) = \mathbf{E}[q_\pi(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t \mid s_t) \mid s_t \sim d_{\pi_\theta}, a_t \sim \pi_\theta]$
- Gradient is an expectation w.r.t. on-policy distribution.
  - Approximation with $\mathscr{D}$ provides a biased estimate of the gradient.
- One solution: correct with importance sampling.
  - $\nabla_\theta J(\pi_\theta) = \mathbf{E}[\frac{\pi_\theta(a_t \mid s_t)}{\pi_\beta(a_t \mid s_t)} q_\pi(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t \mid s_t) \mid s_t \sim d_{\pi_\theta}, a_t \sim \pi_\theta]$.
- Limitations:
  - Requires π_β is known or first estimated, e.g., with maximum likelihood $\hat{\pi}_\beta = \arg\max_\pi \sum_{i=1}^{m} \log \pi(a_t \mid s_t)$.
  - High variance unless $\pi_\beta \approx \pi_\theta$.

Model-based Offline RL

- Use $\mathscr{D}$ to build a simulator of the target MDP.
  - Use $\mathscr{D}$ to learn transition dynamics, p.
  - Learn $\pi^\star$ in the simulator.
- Limitations
  - Learning accurate models from scratch is hard.
  - What should the model predict when an action has not been observed?
- One solution: penalize the policy learned in simulation to avoid out of distribution actions, e.g., with a reward penalty $\tilde{r}(s, a) = r(s, a) - \lambda u(s, a)$.

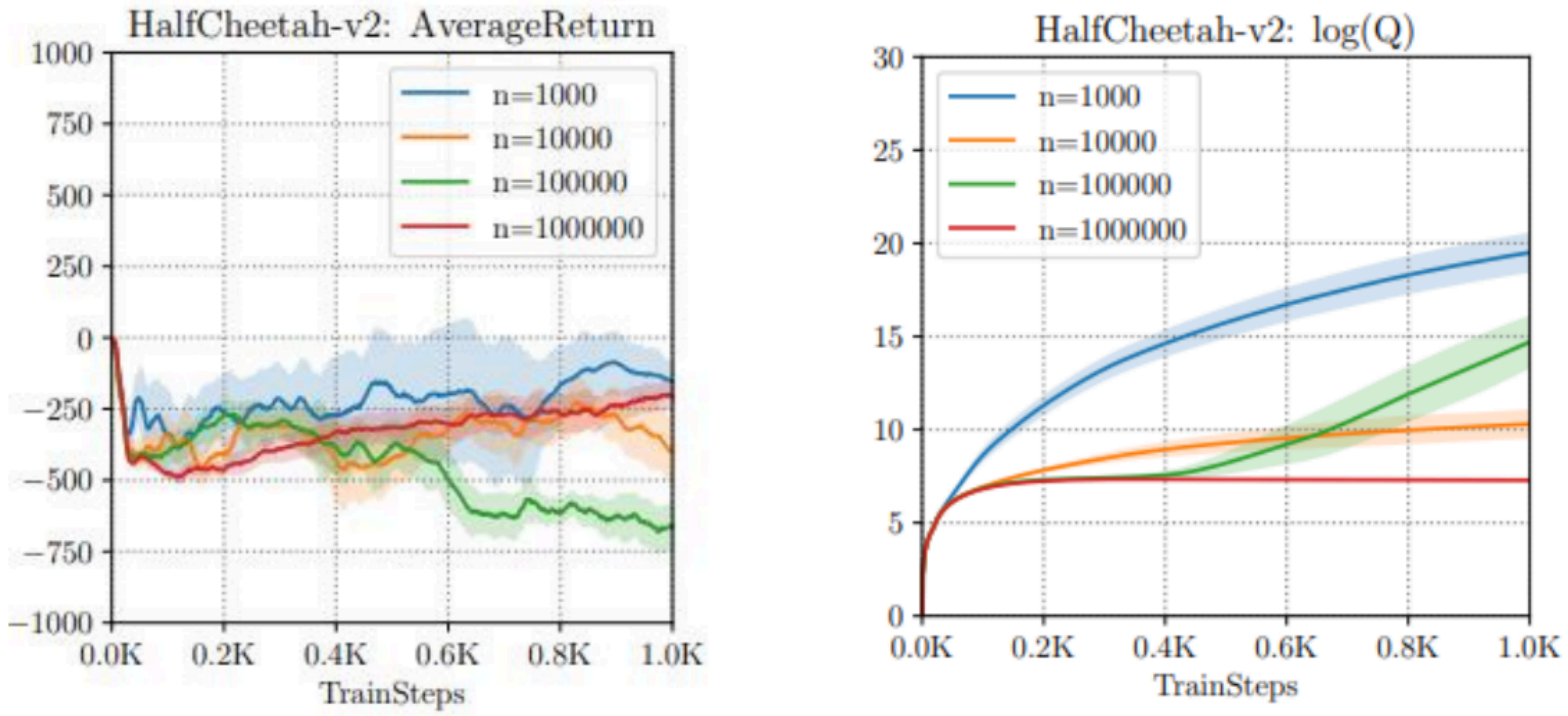


Action-value Based Offline RL

- Q-learning is already an off-policy algorithm.
  What if we just apply it directly to $\mathscr{D}$?

- $q_{k+1}(s_i, a_i) \leftarrow q_k(s_i, a_i) + \alpha(r_i + \gamma \max_{a'} q_k(s_i', a') - q_k(s_i, a_i))$

- $\max_{a'} q_k(s_i', a')$ might over-estimate value if a' is not in the data.



Constrained Policy Iteration

- Possible fix to over-estimation: keep current policy close to π_β.
    - Close in terms of KL-divergence [1] or maximal mean discrepancy [2].
- $\pi_{k+1} = \arg\max_\pi \mathbf{E}[q_{\pi_k}(s, a) \,|\, s \sim \mathscr{D}, a \sim \pi]$ such that $d(\pi_\beta \,||\, \pi) \leq \epsilon$.
- Intuition: make local improvement on top of π_β.
- Limitations: may not know π_β; unclear how to estimate it.
- One solution: use an implicit constraint.

[1] Behavior Regularized Offline Reinforcement Learning. Wu et al. 2019.
[2] Stabilizing Off-Policy Q-Learning via Bootstrapping Error Reduction. Kumar et al. 2019.

Conservative Q-Learning

- Instead of a constraint, we can just be pessimistic with out-of-distribution action-values.
- $$\mathscr{L}_{CQL} = \underbrace{\boxed{(Q(s,a) - (r + \gamma \mathbf{E}_\pi[Q(s',a')]))^2}}_{\substack{\text{Expected SARSA} \\ Q \to q_\pi}} - \underbrace{\boxed{\alpha \mathbf{E}_{(s,a)\sim\mathscr{D}}[Q(s,a)]}}_{\substack{\text{In-distribution bonus} \\ Q \to \infty}} + \underbrace{\boxed{\max_\mu \mathbf{E}_{s\sim\mathscr{D},a\sim\mu(a|s)}[Q(s,a)]}}_{\substack{\text{OOD penalty} \\ Q \to 0}}$$
- Make π greedy w.r.t. Q and repeat.
- Limitations: when to stop training? We lack offline RL workflows as we have with supervised learning.

Siddharth's Presentation

- Slides

Summary

- Offline RL is RL with a static batch of data.
  - No exploration!
- Existing RL algorithms must be adapted for the offline setting:
  - Policy gradient methods may require importance sampling.
  - Model-based methods may require a pessimism assumption.
  - Q-learning-based methods also require pessimism.