Advanced Topics in Reinforcement Learning

Lecture 26: Offline RL II

Josiah Hanna
University of Wisconsin — Madison

Announcements

- Next week: RL application
- Final projects due $<$ 1 week.
- Please complete the course evaluation! At 19% right now.
  - Due December 14!!
- Today:
  - Advanced offline RL challenges.
  - Off-policy Evaluation.

Offline RL Formalism

- Assume the target task can be described as an MDP.
- 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.

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?

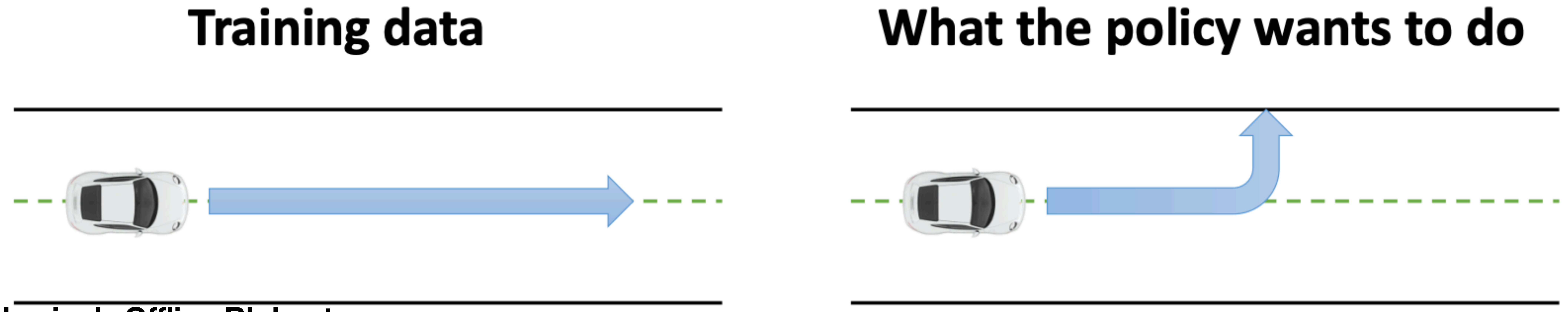
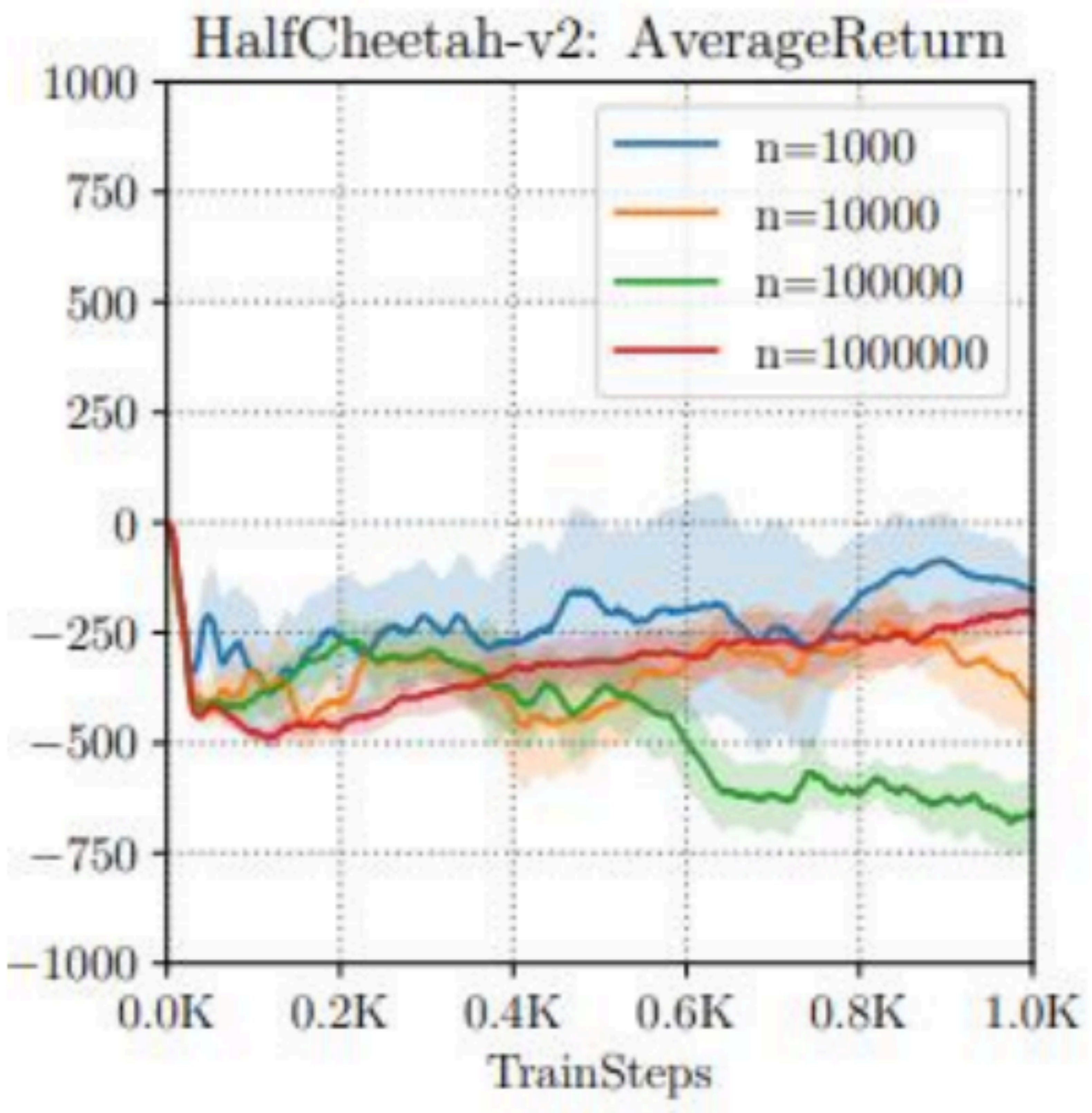


Image credit: Sergey Levine's Offline RL Lecture

Conservative Q-Learning



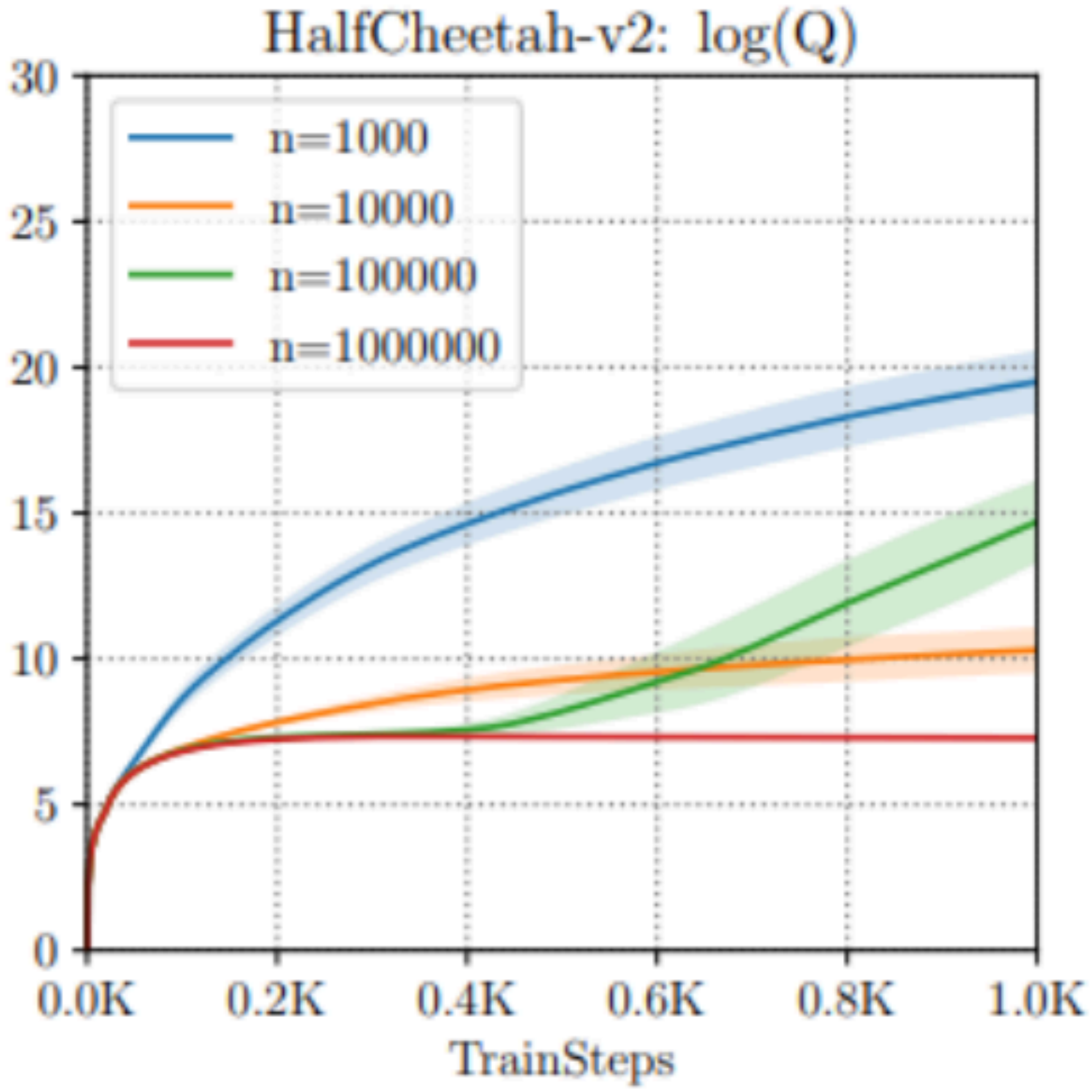


Image Credit: Conservative Q-Learning for Offline Reinforcement Learning. Kumar et al. 2019.

Conservative Q-Learning

- Be pessimistic with out-of-distribution action-values.

$$\mathcal{L}_{CQL} = \boxed{(Q(s,a) - (r + \gamma \mathbf{E}_\pi[Q(s',a')]))^2} - \alpha \boxed{\mathbf{E}_{(s,a)\sim\mathcal{D}}[Q(s,a)]} + \boxed{\max_\mu \mathbf{E}_{s\sim\mathcal{D},a\sim\mu(a|s)}[Q(s,a)]}$$

Expected SARSA $Q \rightarrow q_\pi$

In-distribution bonus $Q \rightarrow \infty$

OOD penalty $Q \rightarrow 0$

- Make π greedy w.r.t. Q and repeat.

- Limitations: when to stop training to avoid over-fitting? We lack offline RL workflows as we have with supervised learning.

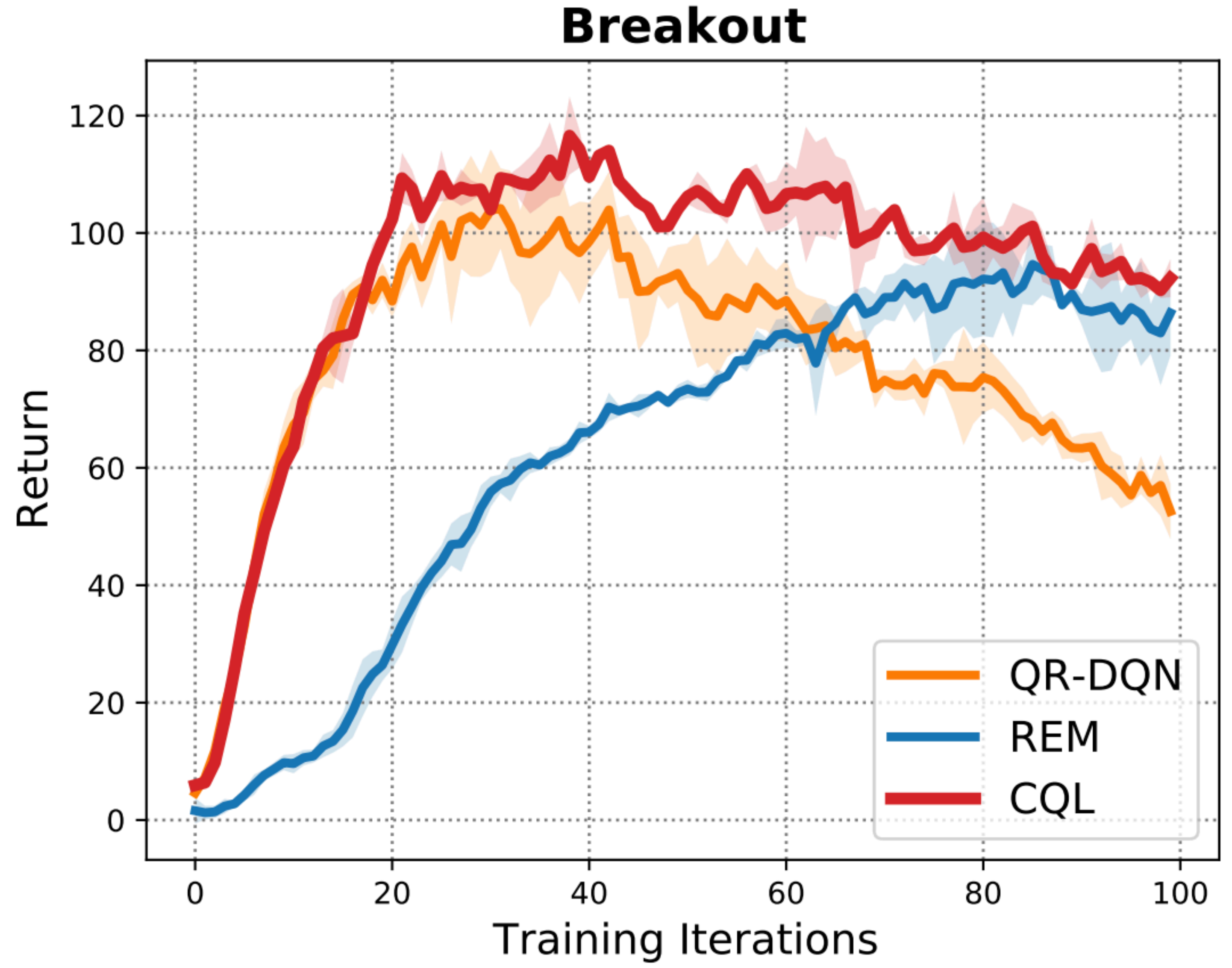


Image Credit: Conservative Q-Learning for Offline Reinforcement Learning. Kumar et al. 2019.

John’s Presentation

- Slides

Advanced Challenges

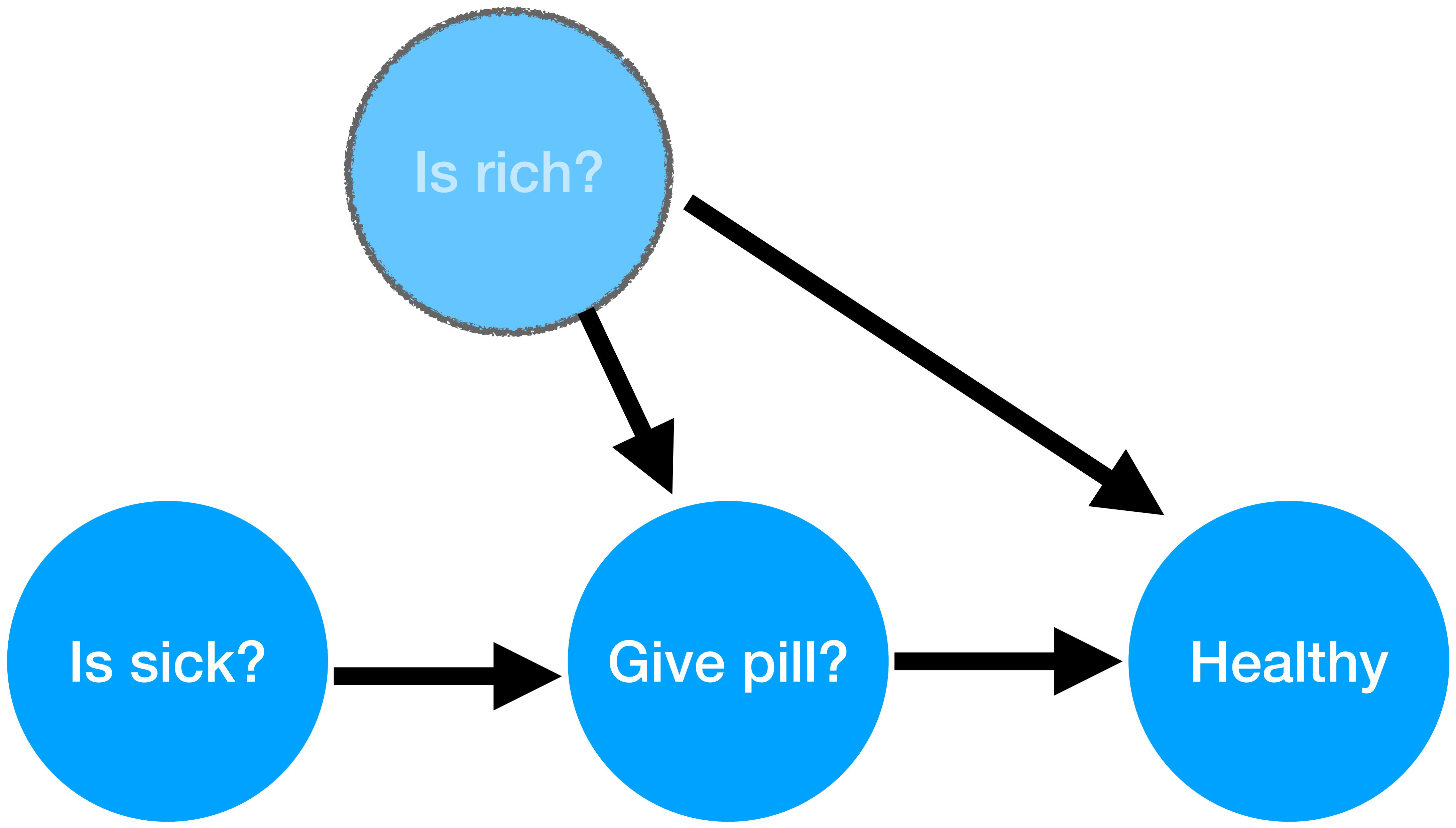
- Non-stationarity: offline data was collected in the past and the target MDP may have changed.
- Offline data may lack rewards or actions.
    - Example: videos of a task show you **what** happened but not **how** done.
- Partial observability:
    - Markov assumption might be violated.
    - Unobserved confounders.

Unobserved Confounders

- So far we have assumed the data was generated by $\pi_\beta(a \mid s)$ meaning that the behavior policy based its action on the state s that we observe in the data.
- What if the behavior policy had access to information not recorded in the data?
- Example:
  - We have medical data that records a patient's vital signs, a treatment prescribed by a doctor, and whether the patient recovered or not.
  - Doctor observes — but does not record — the wealth of the patient.

Unobserved Confounders

Data Generating Process



π_β: **if rich and sick, give pill else don't.**

The Data

{sick, pill, healthy}
{sick, no pill, not healthy}
{not sick, no pill, healthy}
{not sick, no pill, healthy}

Assume wealth leads to recovery (e.g., better diet) and affects doctor's decision.

Even if the pill is useless, an online RL algorithm will conclude that it is beneficial!

Off-Policy Evaluation

- In offline RL, the learned policy does not interact with the real world until deployment time.
  - How do we know that a learned policy will perform well?
  - How do we select hyper-parameters for RL algorithms?
- Answer: use $\mathscr{D}$ to estimate $J(\pi)$ for learned policy π.

What would the expected return be had we ran π instead of π_β?

Importance Sampling Policy Evaluation

- Assume $\mathscr{D}$ consists of full episodes,$\mathscr{D} = \{(S_0, A_0, R_0, S_1, \ldots, S_T, A_T, R_t)\}$.
- If $\mathscr{D}$ had been generated by target policy π then $\frac{1}{m}\sum_{i=1}^{m}\sum_{t=0}^{T}\gamma^t R_t^i$ is an unbiased estimator of $J(\pi)$.
- Since $\mathscr{D}$ was generated by π_β, we instead use importance sampling to adjust for distribution shift:
- $\widehat{J}(\pi) \approx \frac{1}{m}\sum_{i=1}^{m}\rho_i\sum_{t=0}^{T}\gamma^t R_t^i$

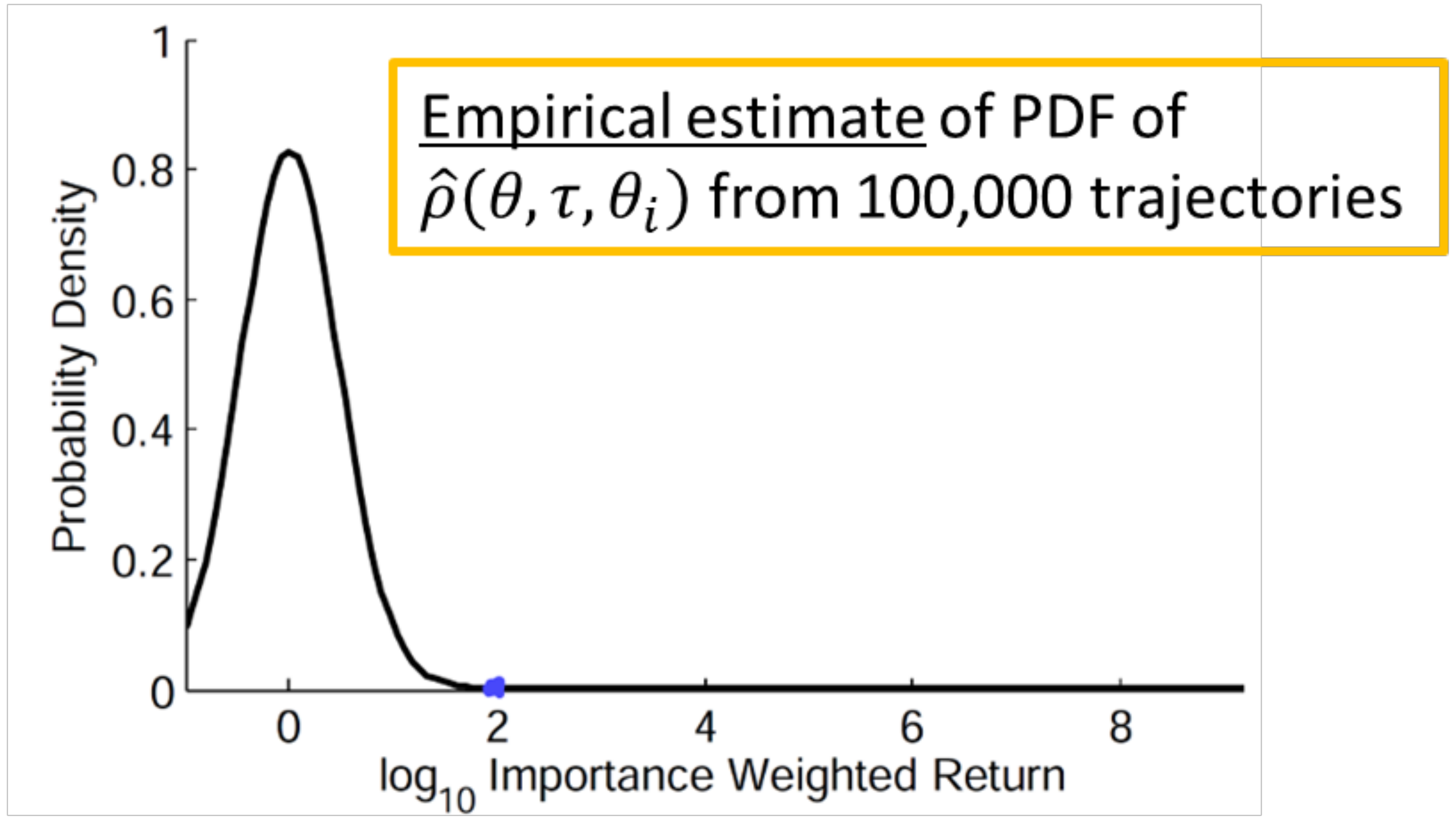
$$\rho_i = \prod_{t=0}^{T}\frac{\pi(A_t^i \mid S_t^i)}{\pi_\beta(A_t^i \mid S_t^i)}$$

- Limitations: high variance; requires π_β is known or estimated.
- Can be improved with different variance reduction techniques: weighted IS, control variates.

Approach – generating unbiased estimates of $\rho(\theta)$

- Unbiased estimate $\hat{\rho}(\theta,\tau,\theta_i)$ generated using importance sampling

$$\hat{\rho}(\theta,\tau,\theta_i) = R(\tau)\frac{\Pr(\tau|\theta)}{\Pr(\tau|\theta_i)} := \underbrace{R(\tau)}_{\text{return}} \underbrace{\prod_{t=1}^{T} \frac{\pi(a_t|s_t,\theta)}{\pi(a_t|s_t,\theta_i)}}_{\text{importance weight}}$$



- $\hat{\rho}(\theta,\tau,\theta_i)$ is bounded from below by zero
  - Since returns are normalized to $[0,1]$
- Upper bound:
  - Probability of selection of a specific action could be low under behavior policy and high under evaluation policy - makes the importance weighted return be large
  - $\hat{\rho}(\theta,\tau,\theta_i)$ has expected value in [0, 1] and has a long tail (large upper bound)
- Hence need to account for large range and high-variance to produce a tight bound on $\rho(\theta)$

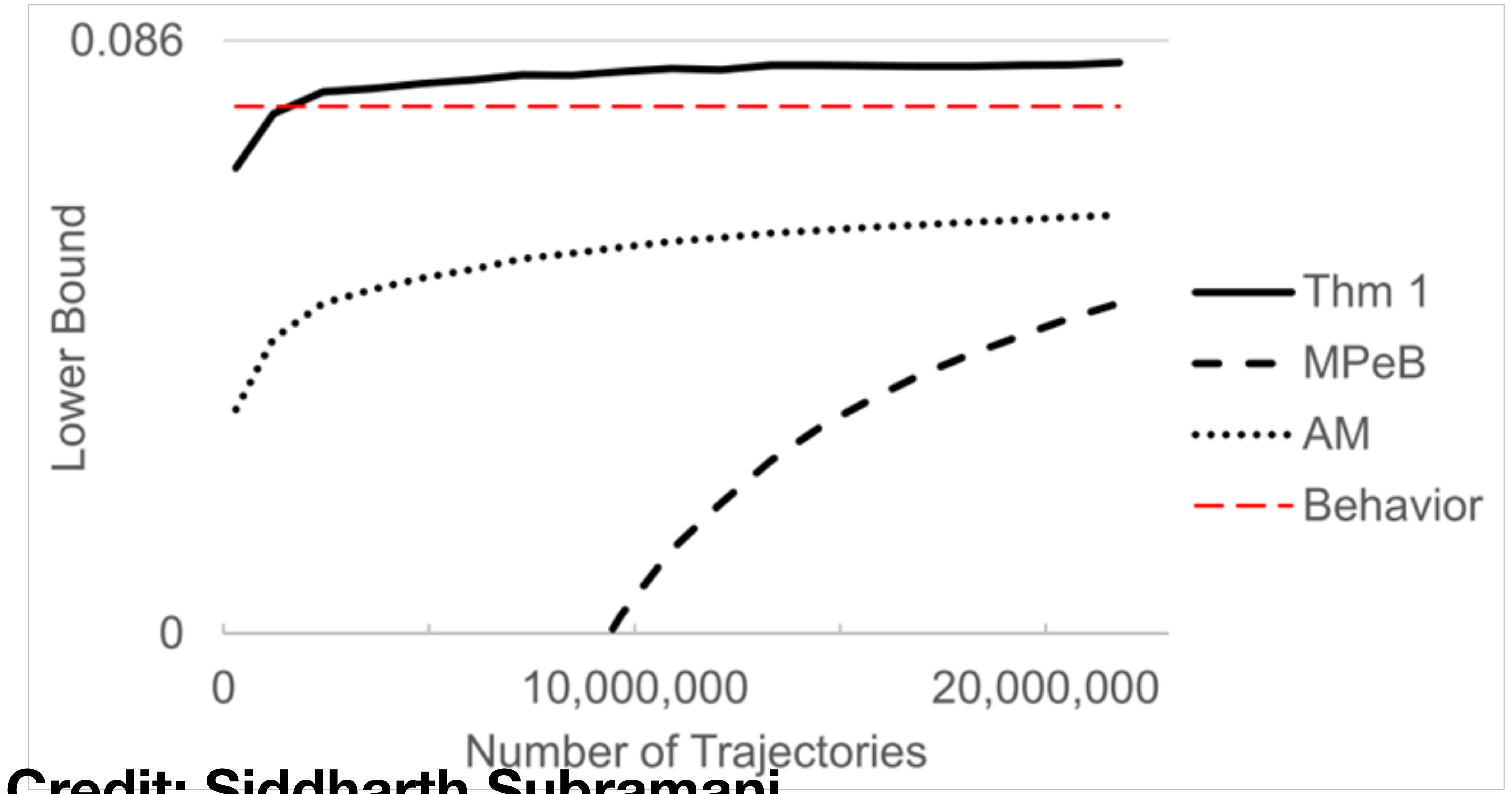
Slide Credit: Siddharth Subramani

Experiments and results

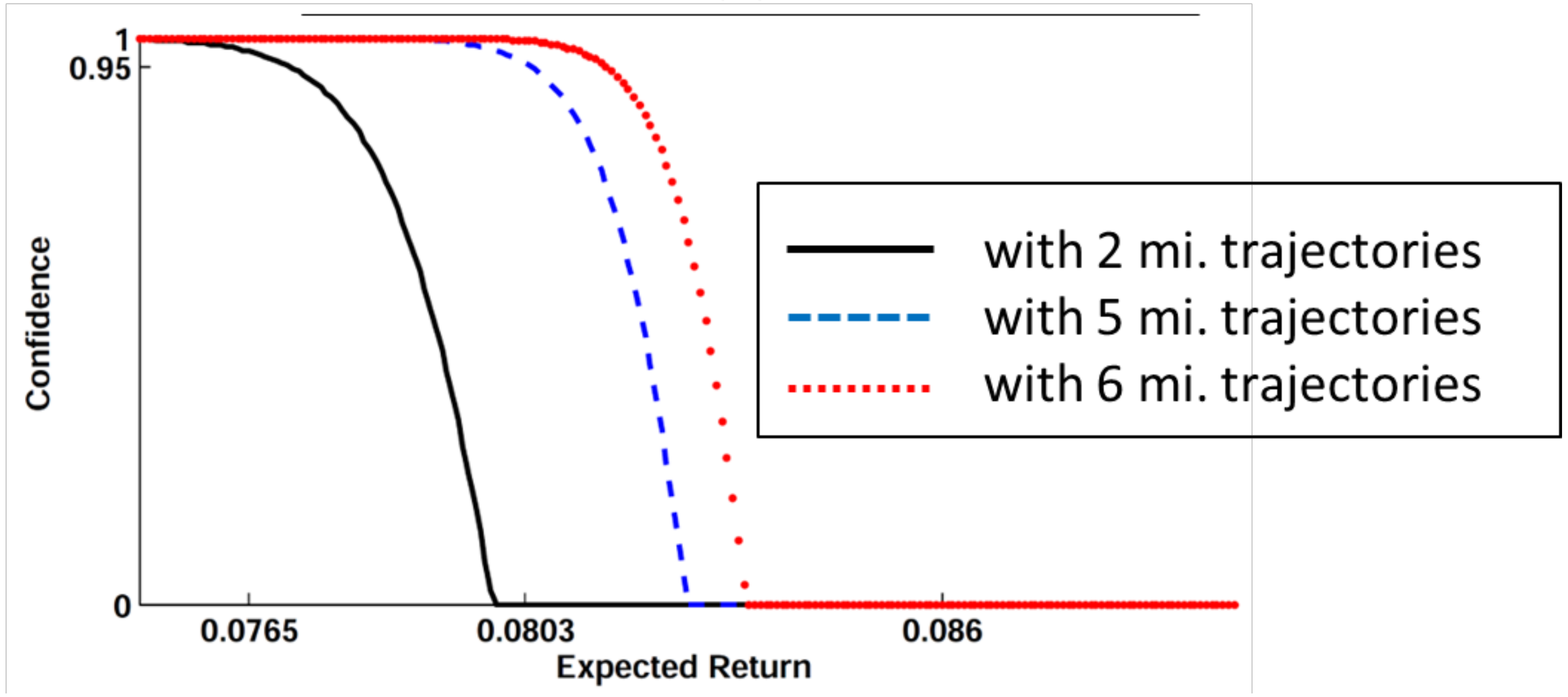
Targeting digital advertisement

- Ads shown on a webpage is based on known features of a user
  - Problem that attempts to maximize the probability of user clicking an ad
  - Sparse reward problem – returns have high variance since most trajectories provide none to less feedback
- This paper uses data from Adobe simulator
  - 31 features representing each user, $+1$ reward when ad is clicked, 0 when ad is overlooked, $T = 20, \gamma = 1$

95% confidence lower bounds



Confidence that $\rho(\theta)$ is the lower bound



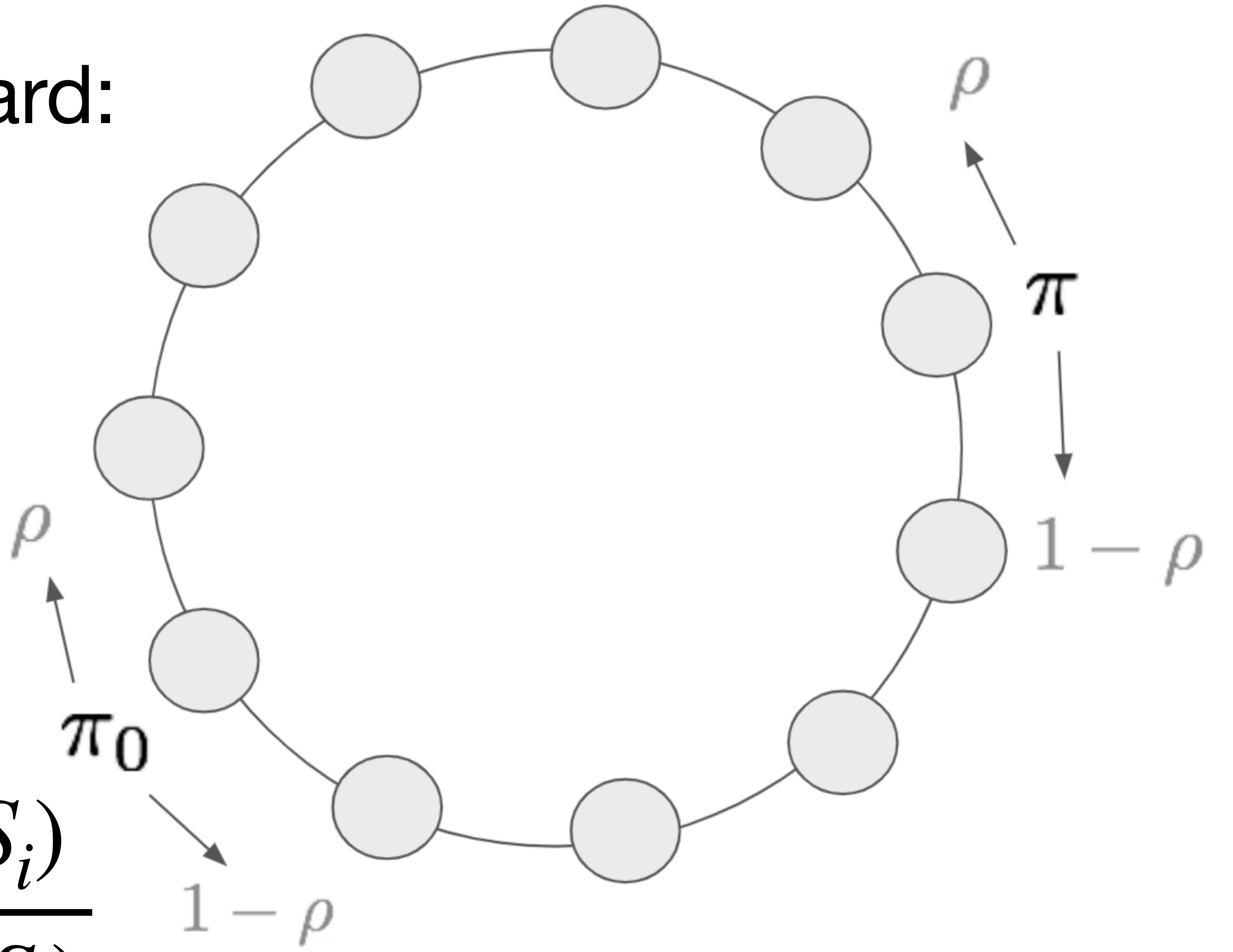
Slide Credit: Siddharth Subramani

Importance Sampling Policy Evaluation

- Importance sampling has variance that is exponential in the length of episodes.
- Alternatively, consider estimating average reward:
  - $J(\pi) = \frac{1}{1-\gamma}\mathbf{E}[R_t \mid S_t \sim d_\pi, A_t \sim \pi]$
  - $J(\pi) \approx \frac{1}{m}\sum_{i=1}^{m} w_i R_i$

Must be estimated from $\mathscr{D}$.
Many ways to do this.

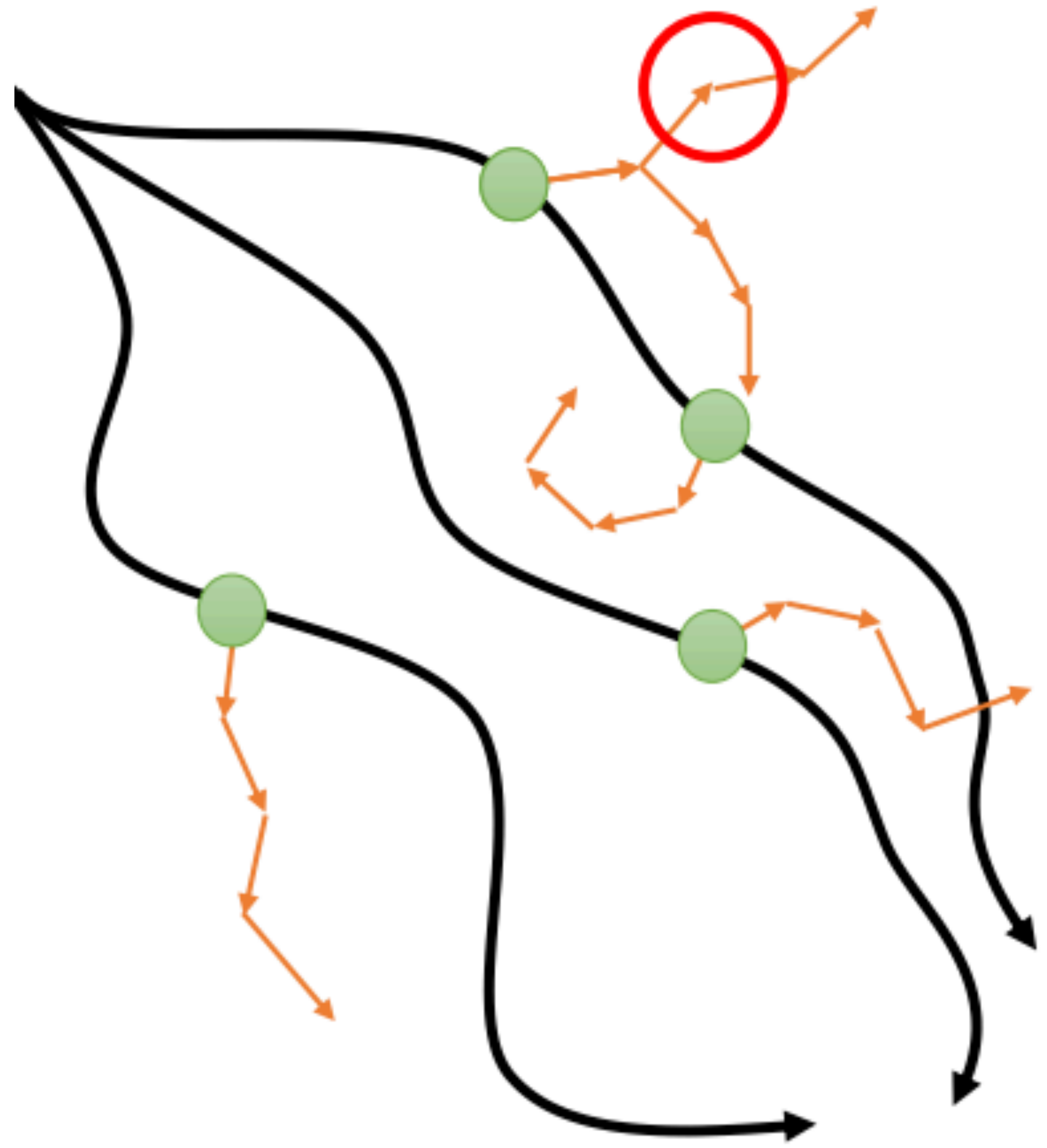
$$w_i = \frac{d_\pi(S_i, A_i)}{d_\beta(S_i, A_i)} = \frac{d_\pi(S_i)\pi(A_i \mid S_i)}{d_\beta(S_i)\pi_\beta(A_i \mid S_i)}$$



Breaking the Curse of Horizon: Infinite-Horizon Off-Policy Estimation. Liu et al. 2018

Model-based Policy Evaluation

- Use $\mathscr{D}$ to build a simulator of the target MDP.
  - Use $\mathscr{D}$ to learn transition dynamics, p.
  - Evaluate in the simulator.
- Limitations
  - Learning accurate models from scratch is hard.
  - What should the model predict when an action has not been observed?



Fitted Q-Evaluation

- Write policy performance in terms of action-values:
  - $J(\pi) = \mathbf{E}[q_\pi(S, A) \mid S \sim d_0, A \sim \pi]$
- Estimate q_π with DQN-like variant of expected SARSA:
  - $\mathscr{L}(Q_\theta) = \frac{1}{m} \sum_{i=1}^{m} \left(r_i + \gamma \sum_{a'} \pi(a' \mid s_i') Q_{\bar{\theta}}(s_i', a') - Q_\theta(s_i, a_i) \right)^2$

Like DQN except use expectation w.r.t. π instead of max

Which OPE method to use?

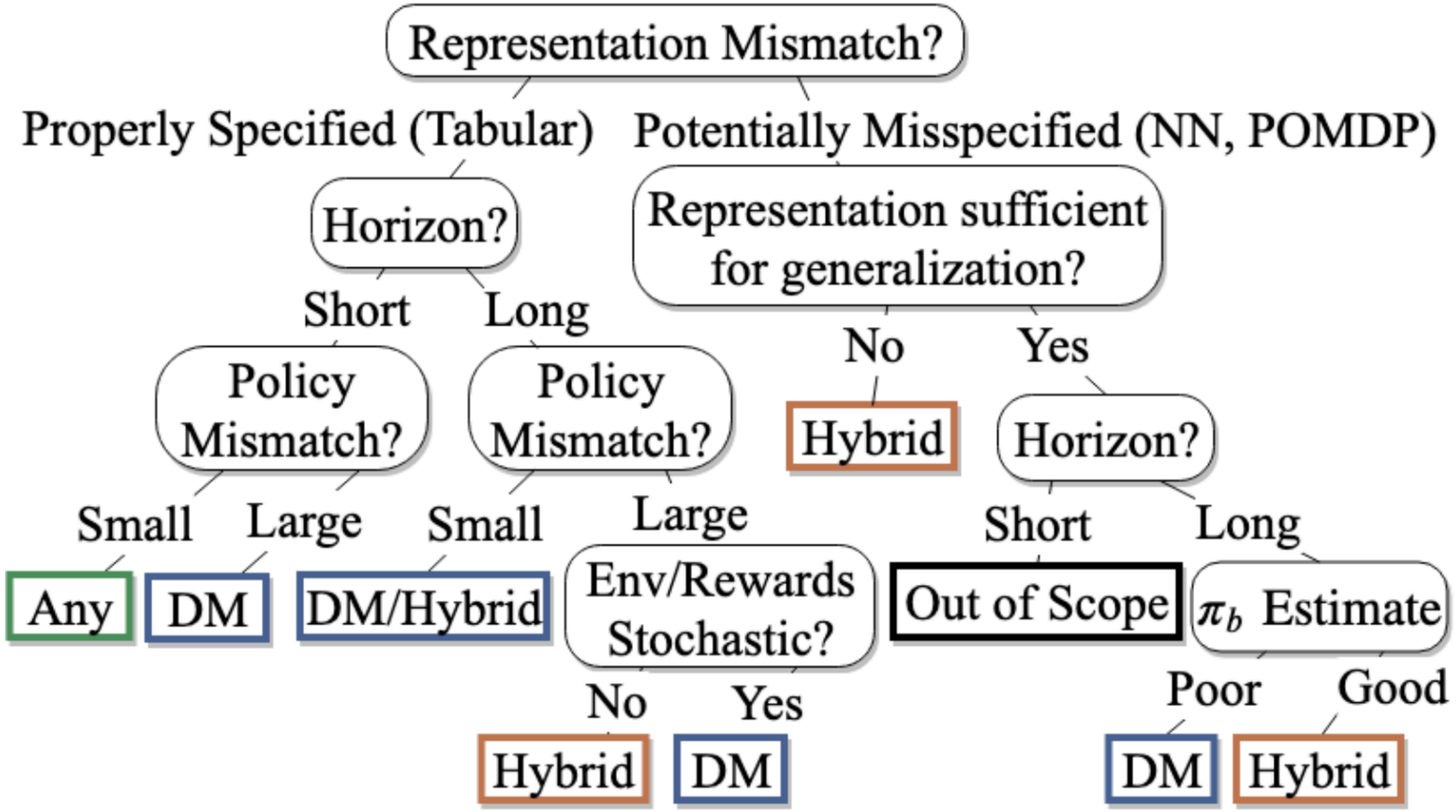


Figure 2: *General Guideline Decision Tree.*

Image Credit: Empirical Study of Off-Policy Policy Evaluation for Reinforcement Learning. Voloshin et al. 2021.

Summary

- Offline RL is RL with a static batch of data.
  - No exploration!
- Existing RL algorithms must be adapted for the offline setting to handle missing actions and distribution shift.
- Other challenges include: missing actions, non-stationarity, and partial observability that introduces unobserved confounders.
- Off-policy evaluation can mitigate the risk of deploying a sub-optimal policy but has many practical challenges.

Action Items

- Last reading on RL applications.
- Good luck on your final project.