A Unified Framework for Alternating Offline Model Training and Policy Learning

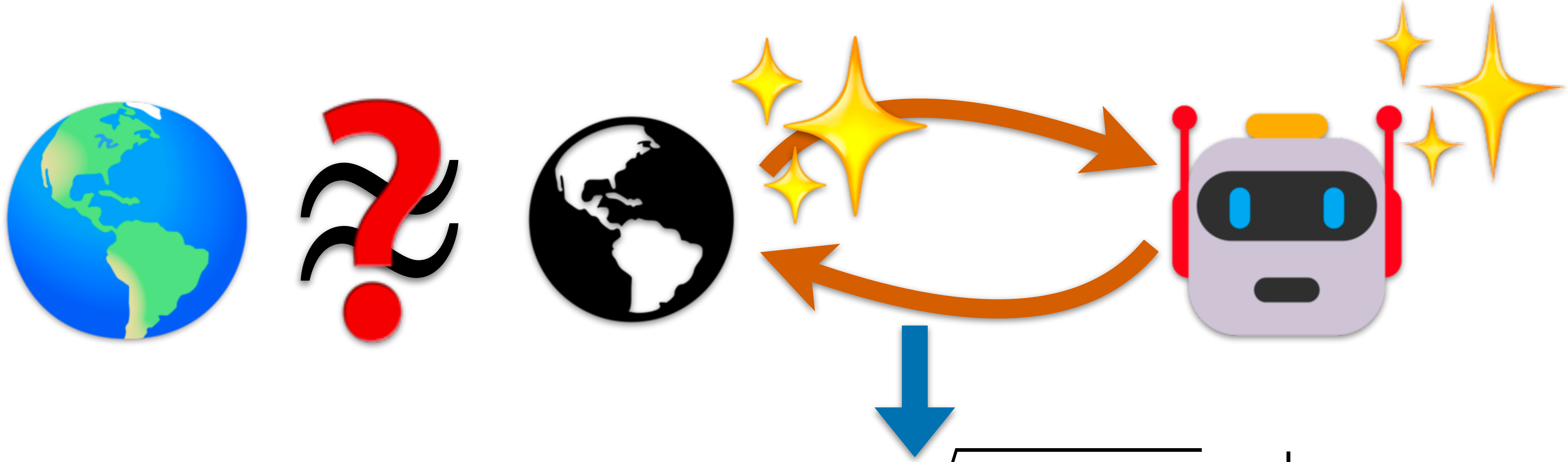
Shentao Yang[1], Shujian Zhang[1], Yihao Feng[2], Mingyuan Zhou[1]

[1]The University of Texas at Austin, [2]Salesforce Research

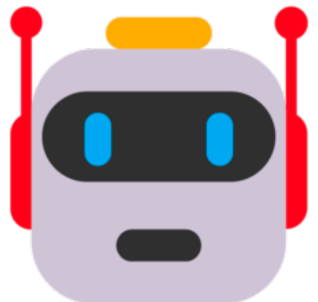
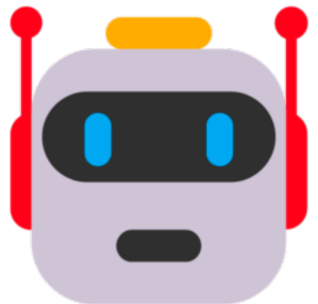
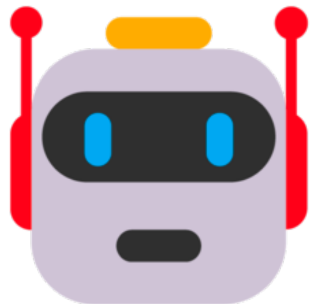
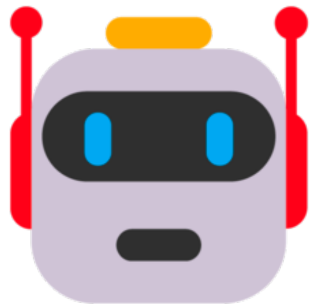
October, 2022

Proposed Method Sketch

- **Motivation**: model training = MLE ≠ improve policy = model usage.

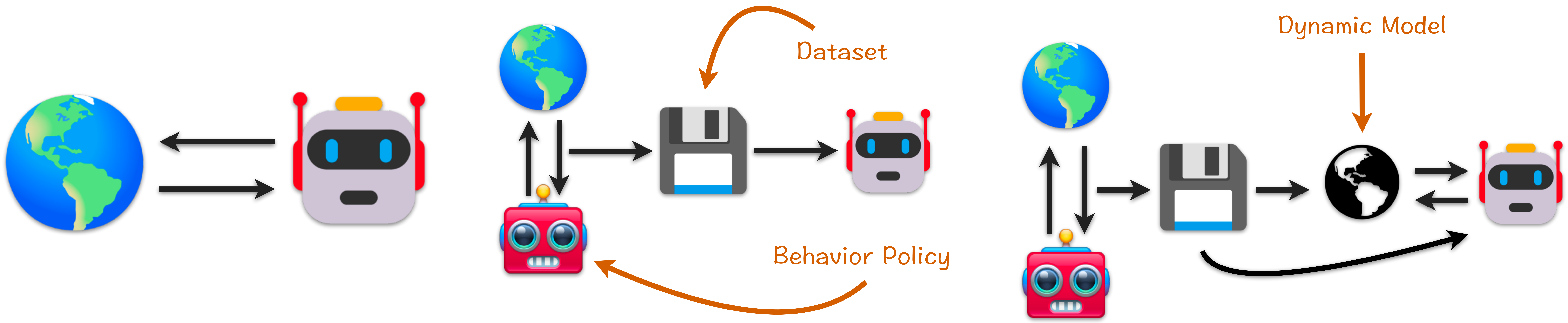


$$\min_{\pi, \widehat{P}} C \cdot \sqrt{D_\pi(P^*, \widehat{P})} \geq \left| J(\pi, P^*) - J(\pi, \widehat{P}) \right|$$

- Jointly train  and  to minimize an upper bound of the evaluation error.
  - Fixed ,  ≈  only on state-actions visited by .
  - Fixed , optimize  with a regularization based on .

Background

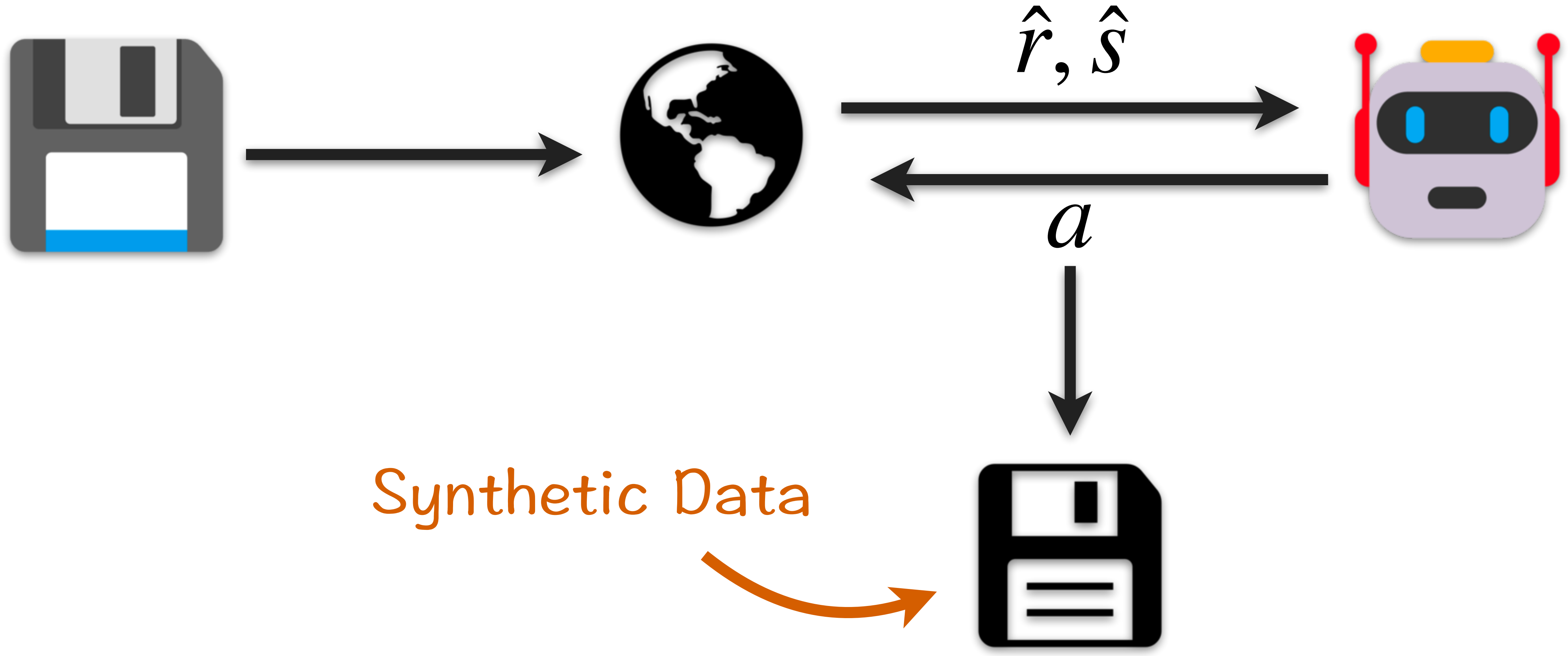
- Offline RL: learn policy from static datasets.
- Offline Model-Based RL (Offline MBRL): learn dynamic from static datasets.

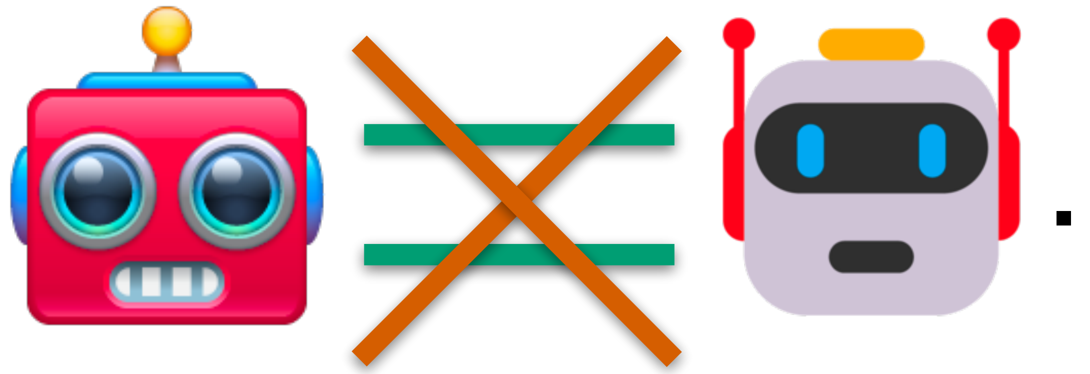


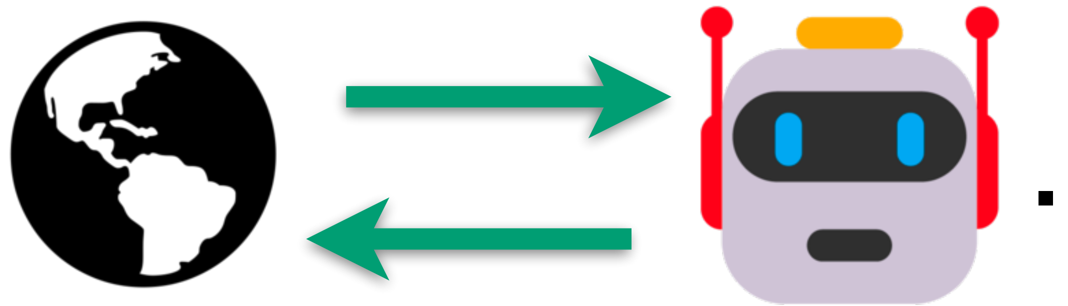
(a) Classical RL (b) Offline RL (c) Offline MBRL

Background

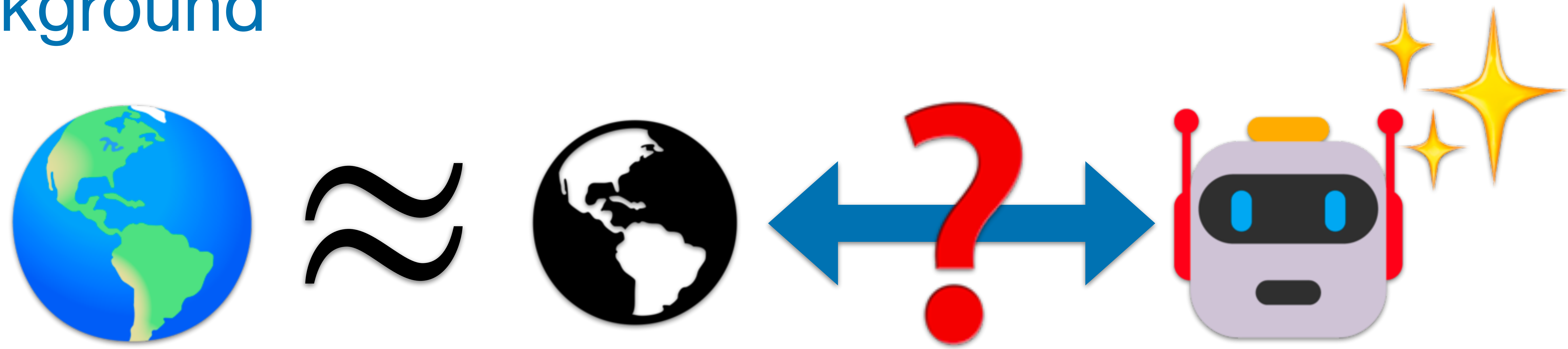
- Benefits of offline MBRL



- Offline model-free RL
  - Only know reward and next state at state-actions within the dataset.
  - Off-policy issue 

- Offline model-based RL
  - Estimate reward and next state at new state-actions.
  - $\approx$ on-policy 

Background



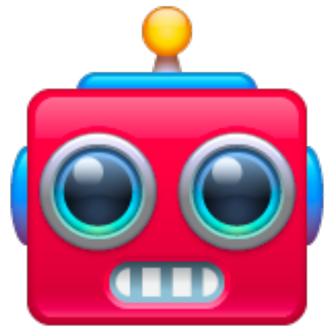
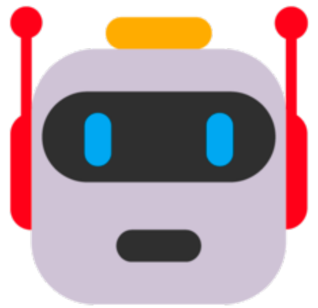
- Most offline MBRL: pre-train a fixed dynamic model on  .
  - Objective: MLE — “simply a mimic of the world.”
  - Usage: improve the policy.
- Objective mismatch: model training $\neq$ model usage.
  - Especially when  is limited and  is hard to learn.

Proposed Method: Bounding the Evaluation Error

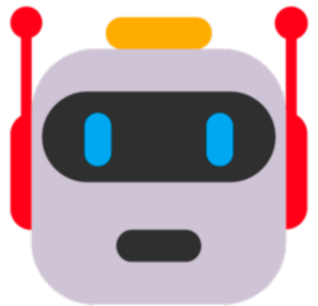
- A tractable upper bound for the evaluation error

$$\left| J(\pi, P^*) - J(\pi, \widehat{P}) \right| \leq C \cdot \sqrt{D_\pi(P^*, \widehat{P})}, \quad \text{with}$$

$$D_\pi(P^*, \widehat{P}) \triangleq \mathbb{E}_{(s,a)\sim d^{P^*}_{\pi_b,\gamma}} \left[\omega(s,a)\, \mathrm{KL}\left(P^*(s' \mid s,a)\, \pi_b(a' \mid s') \,||\, \widehat{P}(s' \mid s,a)\, \pi(a' \mid s') \right) \right],$$

- π_b is the behavior policy  .
- $d^{P^*}_{\pi_b,\gamma}$ is the offline-data distribution  .
- $\omega(s,a) \triangleq \dfrac{d^{P^*}_{\pi,\gamma}(s,a)}{d^{P^*}_{\pi_b,\gamma}(s,a)}$ is the density ratio between  and visitation freq. of  .

Proposed Method: Model Training

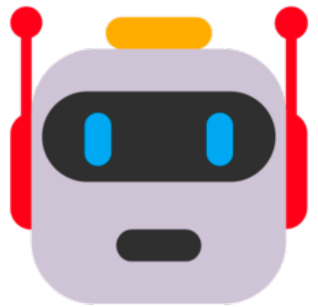
- Fix , we train the model  by

$$\ell(\widehat{P}) \triangleq -\mathbb{E}_{(s,a,s')\sim d^{P^*}_{\pi_b,\gamma}}\left[\omega(s,a)\log\left\{\widehat{P}(s' \mid s,a)\right\}\right] = D_\pi(P^*, \widehat{P}) - C', \quad \text{with } C' \text{ a constant to } \widehat{P}.$$

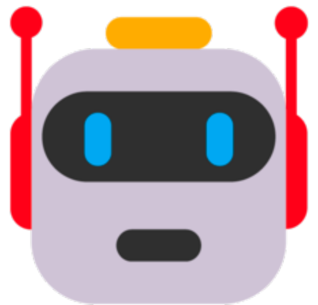
- (s, a, s') is one transition in .

- Given $\omega(s, a)$, a stable weighted MLE objective.

Proposed Method: Policy Learning

- A lower-bound of  performance: $J\left(\pi, \widehat{P}\right) - C \cdot \sqrt{D_\pi(P^*, \widehat{P})}$.
- Fix , empirically helpful to construct the regularizer by:
  - Removing the $\sqrt{\cdot}$.
  - Applying a further relaxation

$$D_\pi(P^*, \widehat{P}) \leq C'' \cdot \mathrm{KL}\left(P^*(s' \mid s, a)\, \pi_b(a' \mid s')\, d^{P^*}_{\pi_b,\gamma}(s, a) \,||\, \widehat{P}(s' \mid s, a)\, \pi(a' \mid s')\, d^{P^*}_{\pi_b,\gamma}(s)\, \pi(a \mid s)\right)$$

    - Stronger regularizer: regularizes  at both s and s'.
  - Changing KL-divergence to Jensen-Shannon divergence.

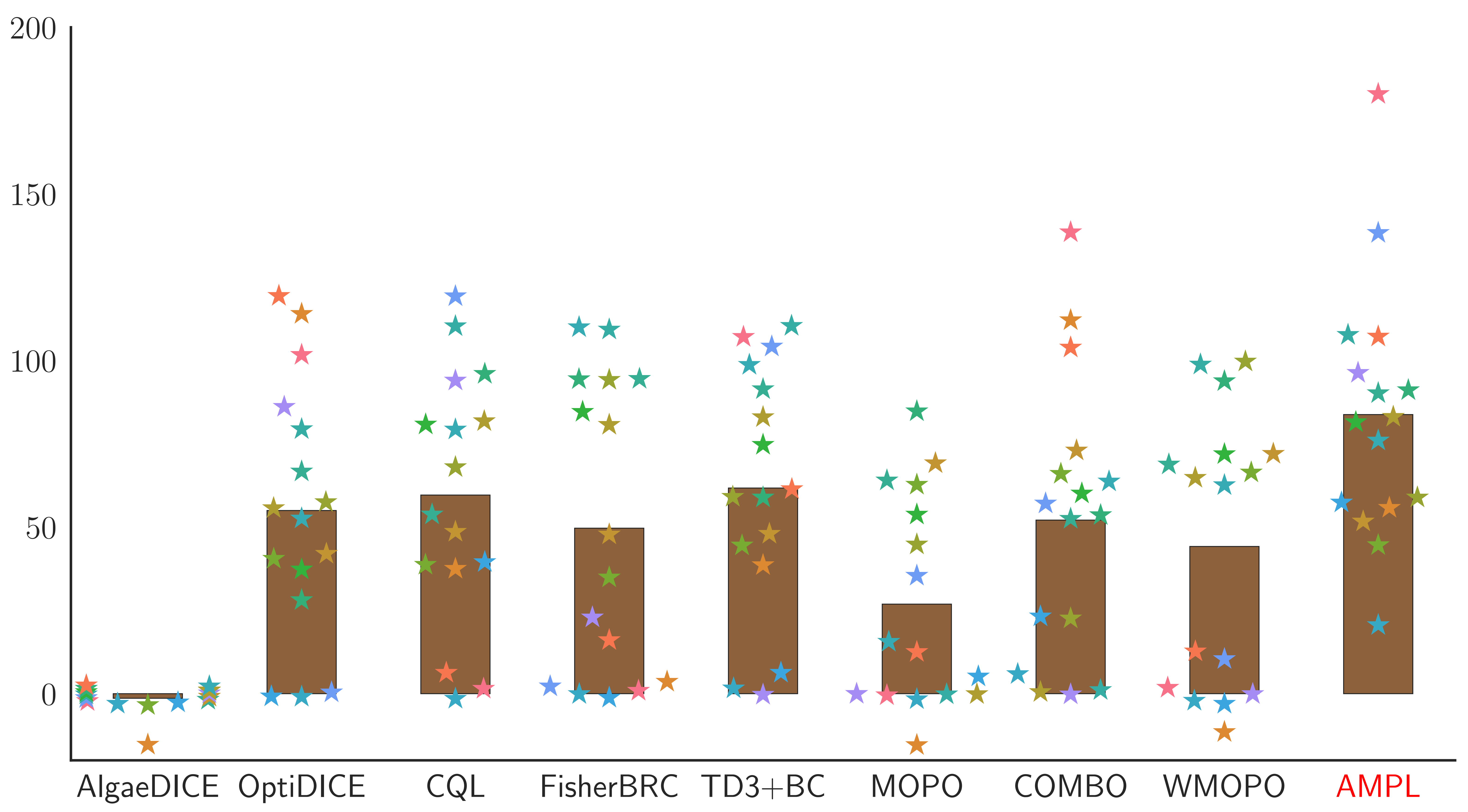
Proposed Method: Density-Ratio Training

- Fixed-point style method, ~~saddle-point optimization~~.
- A simple MSE objective:

$$\mathbb{E}_{(s,a)\sim d^{P^*}_{\pi_b,\gamma}}\left[\omega(s,a)\cdot Q^{\widehat{P}}_{\pi}(s,a)\right] = \gamma\, \mathbb{E}_{\substack{(s,\,a,\,s')\,\sim\, d^{P^*}_{\pi_b,\gamma} \\ a'\,\sim\,\pi(\,\cdot\,|\,s')}}\left[\omega(s,a)\cdot Q^{\widehat{P}}_{\pi}(s',a')\right] + (1-\gamma)\,\mathbb{E}_{\substack{s\,\sim\,\mu_0(\,\cdot\,) \\ a\,\sim\,\pi(\,\cdot\,|\,s)}}\left[Q^{\widehat{P}}_{\pi}(s,a)\right].$$

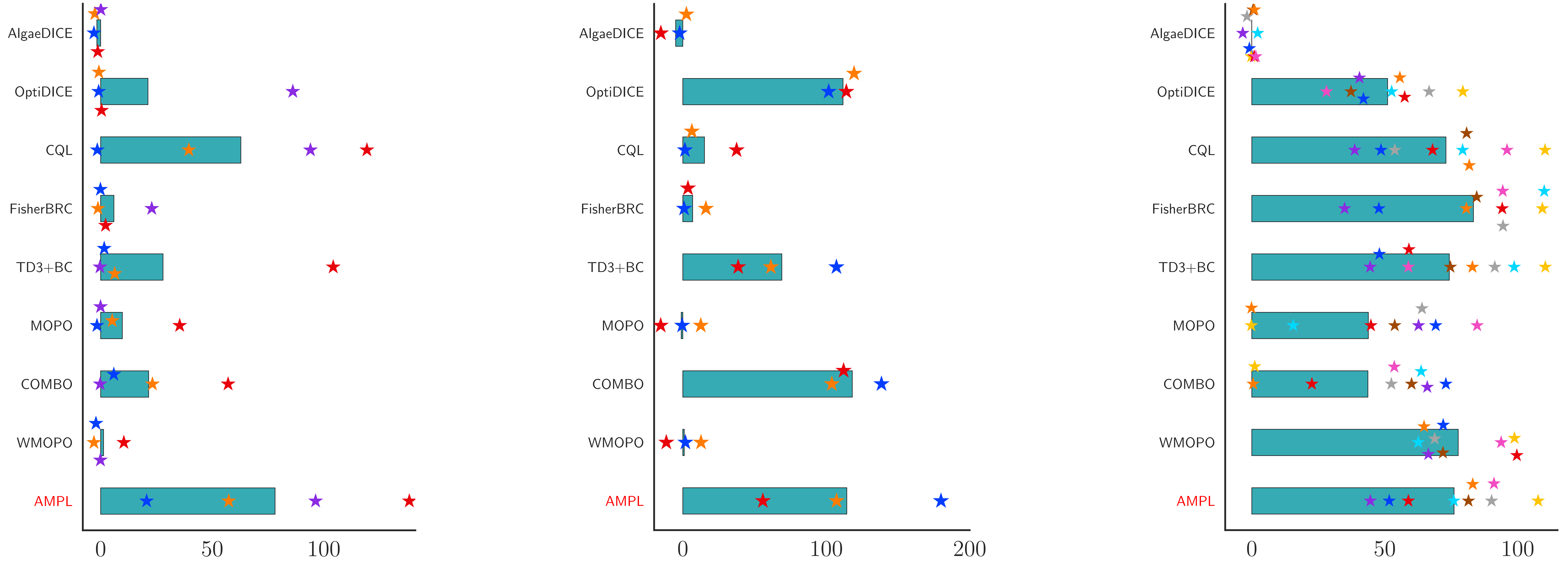
- Based on the "forward" Bellman equation for $\omega(s,a)$ —not tractable 😭 !
    - Use Q-function as test function and $\sum_{(s',a')}$ on both sides.
        - Primal-dual relation between $\omega(s,a)$ and Q-function in OPE.
- Only requires samples from 💾 and the initial state-distribution.

Results: Main Method



- Our offline Alternating Model-Policy Learning (AMPL) performs well on D4RL tasks.

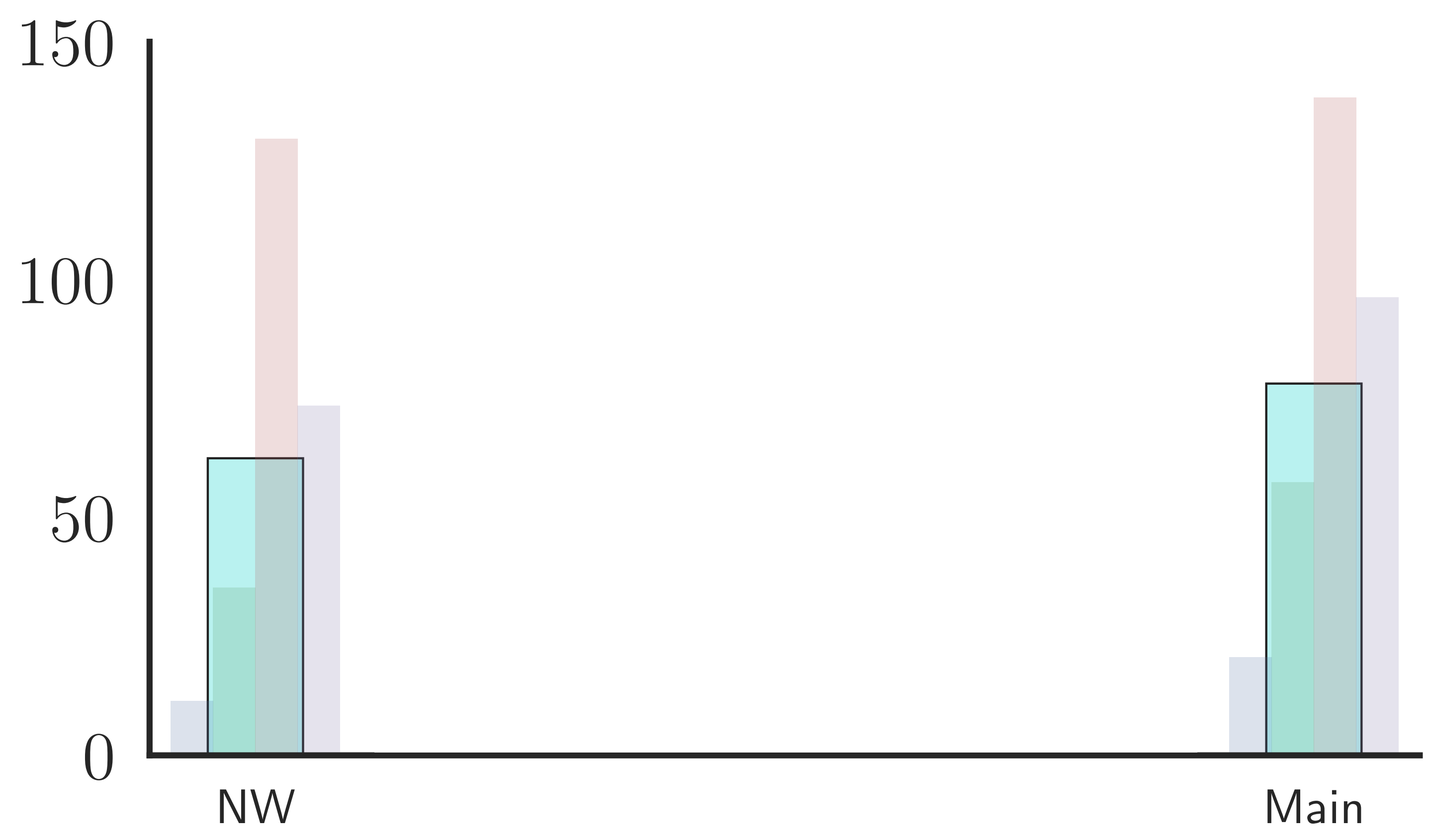
Results: Main Method



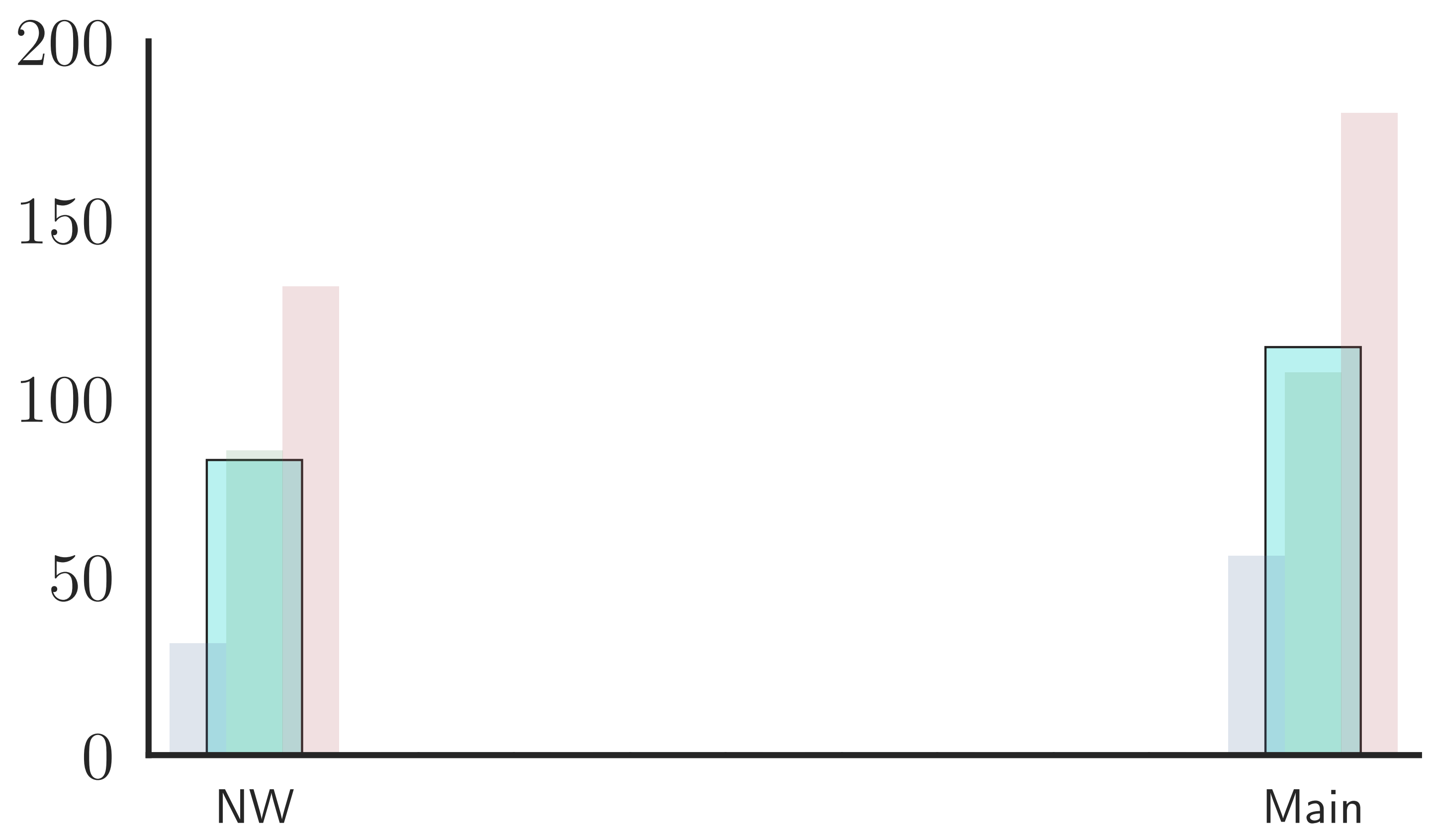
(a) Adroit (b) Maze2D (c) MuJoCo

- Learn well on the MuJoCo datasets.
- Robust and good results on the challenging Adroit and Maze2D datasets.

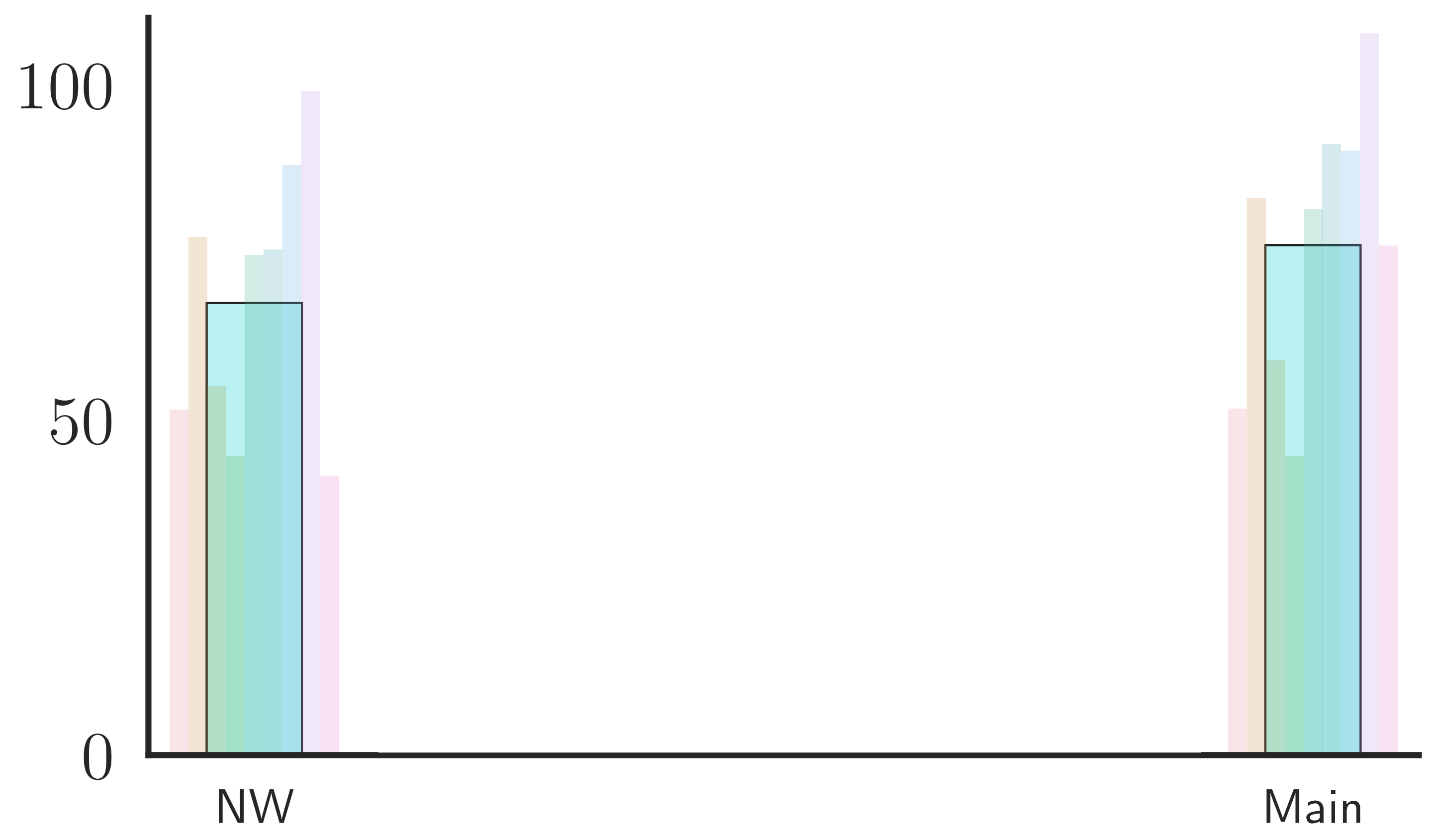
Ablation Study I: Does weighted model (re)training help?



(a) Adroit



(b) Maze2D

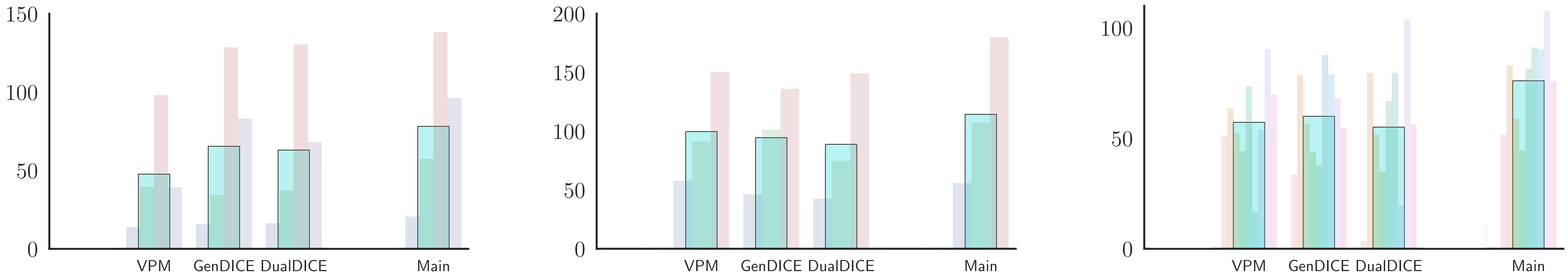


(c) MuJoCo

- Variant: training  only at the beginning using MLE — No Weights (NW).

- On all three domains, the NW variant generally underperforms the main method.

Ablation Study II: Other density-ratio estimation methods?

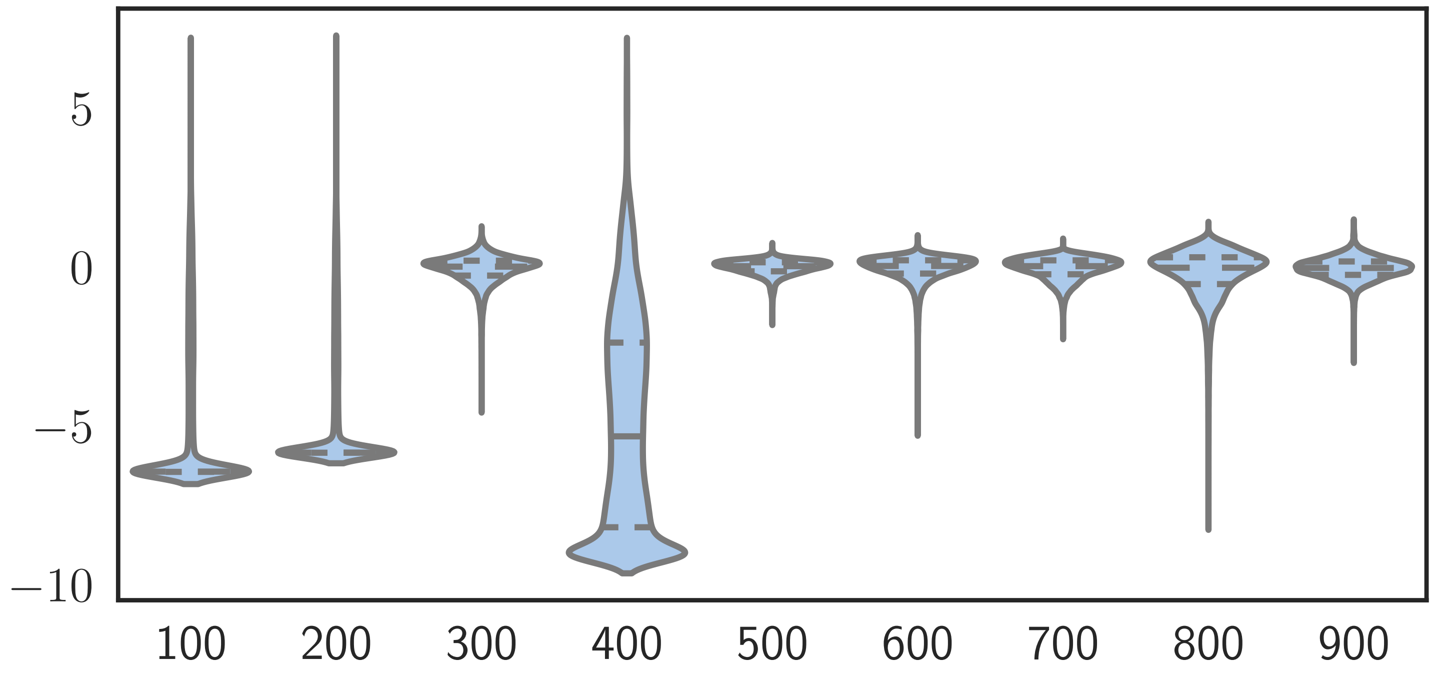


(a) Adroit (b) Maze2D (c) MuJoCo

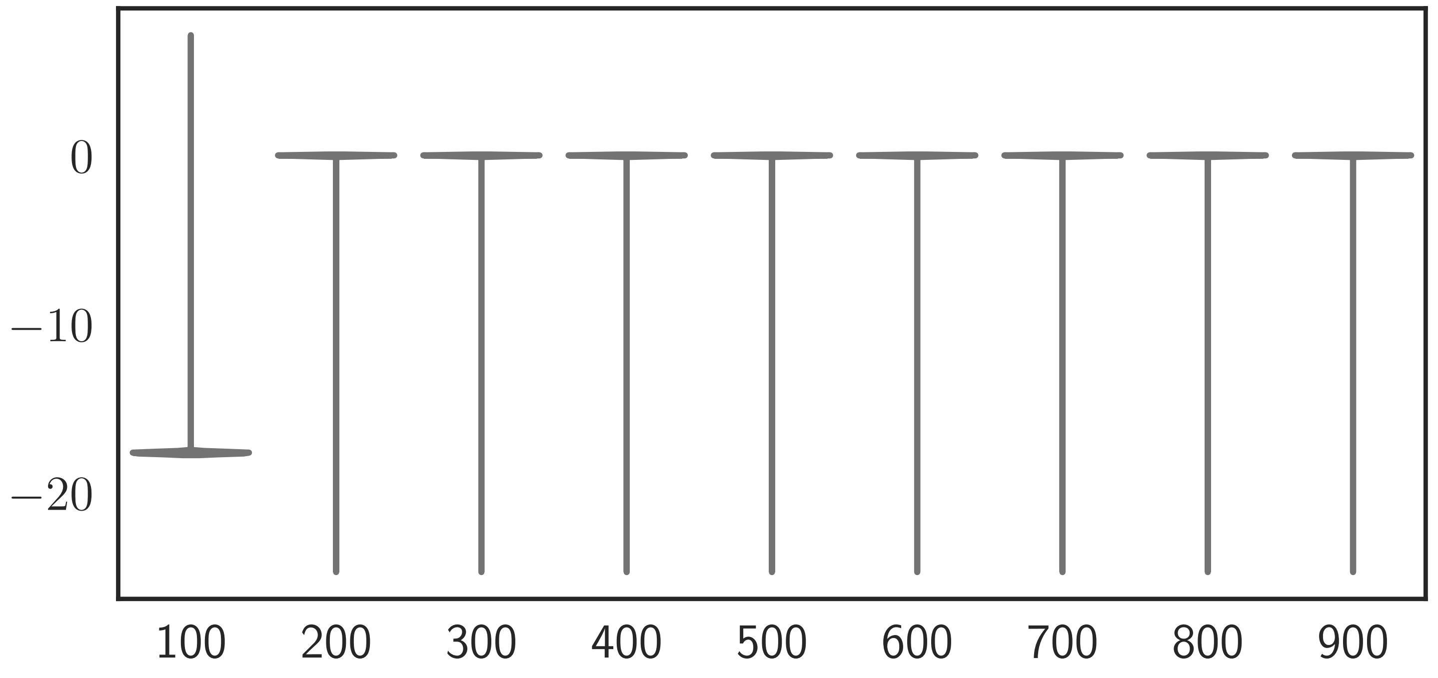
- Variant: $\omega(s, a)$ is estimated by VPM, GenDICE, and DualDICE.

- On all three domains, these three variants generally underperform our method.

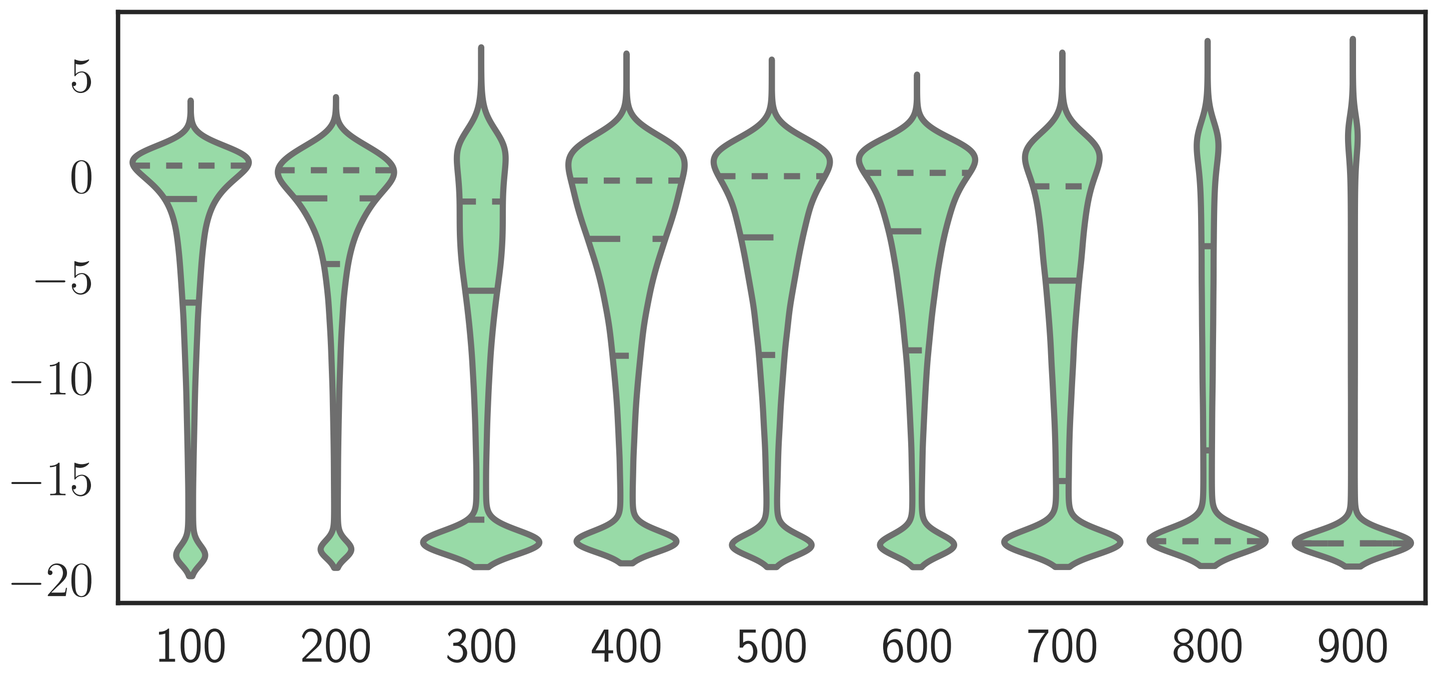
Ablation Study II: Other density-ratio estimation methods?



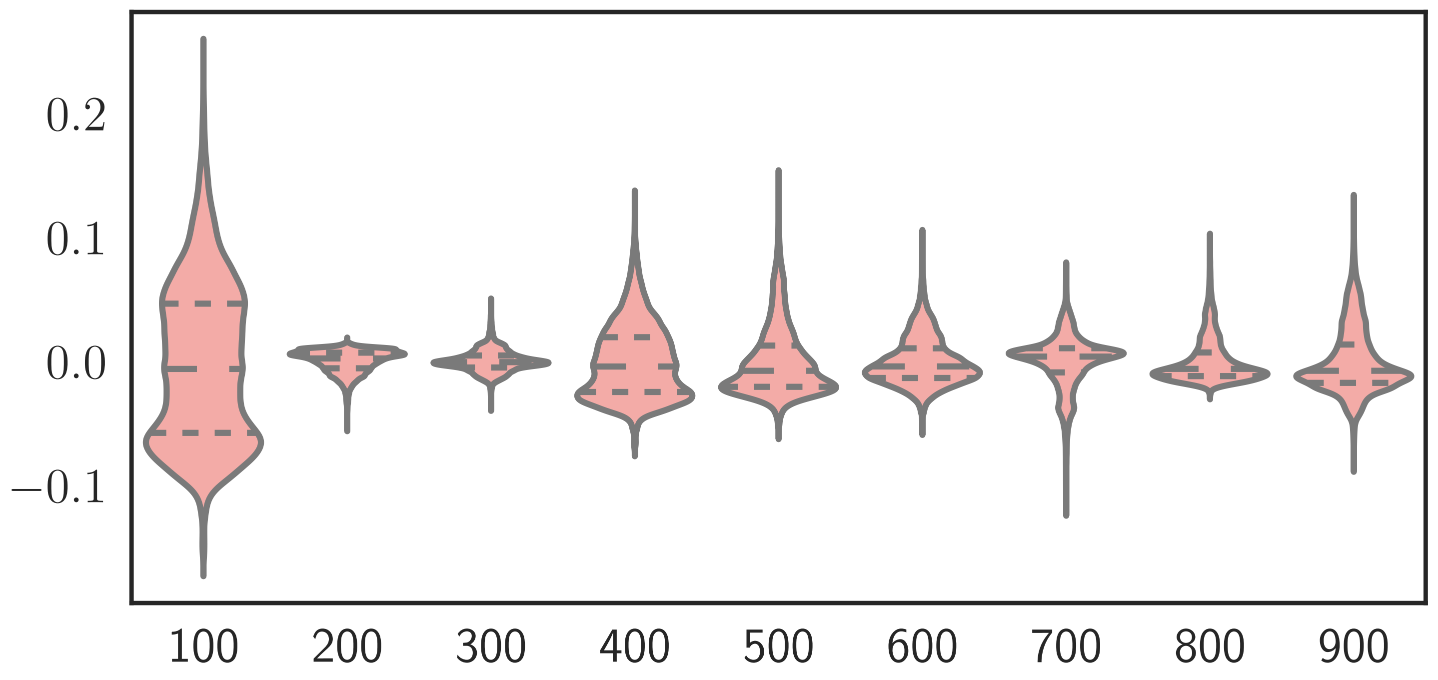
(a) VPM



(b) GenDICE



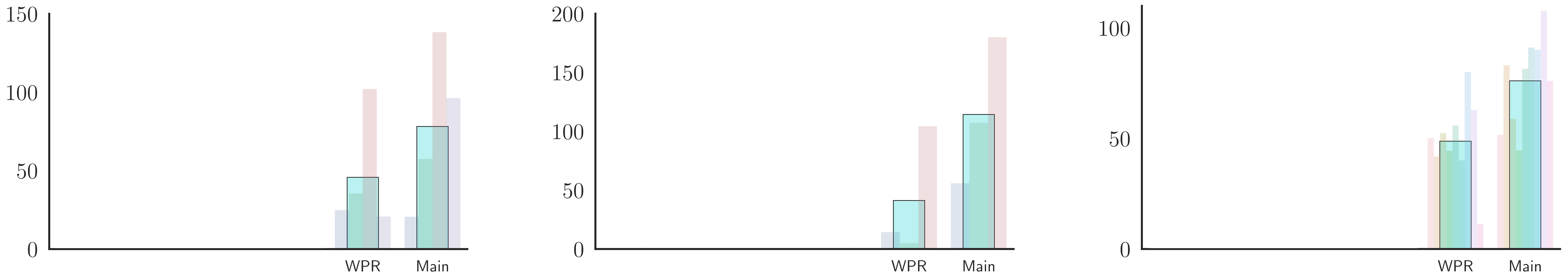
(c) DualDICE



(d) Ours

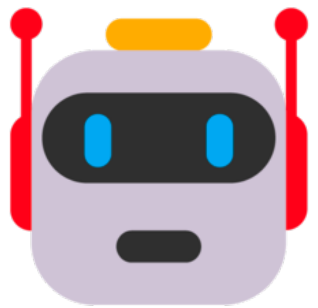
- Distribution plot of $\log(\omega(s, a))$ during the training process, on "walker2d-medium-replay."
- Three alternatives can be unstable to provide good density-ratio for 🌎 training.

Ablation Study III: A weighted policy regularizer?



(a) Adroit (b) Maze2D (c) MuJoCo

- Variant: policy regularizer is weighted by the density ratio $\omega(s, a)$ (WPR).

- Additional instability in training  $\Longrightarrow$ underperform!

Summary

- **Goal**: close the mismatched model objectives in offline MBRL.
- **Method**: offline Alternating Model-Policy Learning.

QR code for the full paper!

QR code for the GitHub Repo!