Regularizing a Model-based Policy Stationary Distribution to Stabilize Offline Reinforcement Learning

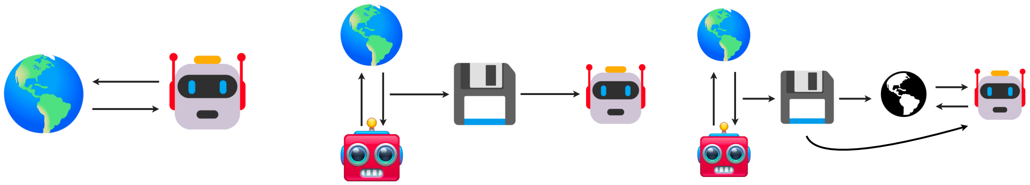
Shentao Yang, Yihao Feng, Shujian Zhang, Mingyuan Zhou

The University of Texas at Austin

June 19, 2022

Background

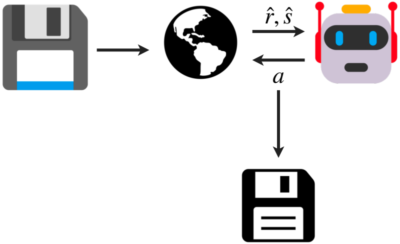
- Offline RL: learn policy from static datasets.
    - Offline model-based RL (offline MBRL): use the static datasets to learn the dynamic.



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

Background

Benefits of offline MBRL.



- Offline model-free RL
  - Only know the reward and next state at state-action pairs within the dataset.
  - Dataset size can be small.
- Offline model-based RL
  - Estimate the reward and next state at new state-action pair.
  - Augment the static dataset.

Background

Need proper regularization in offline MBRL.

- Limited dataset $\rightarrow$ estimated model is only accurate nearby.
- Regularization in policy learning $\rightarrow$ avoid bad predictions and model exploitation.

Proposed Method Sketch

- Constrain the learned policy to visit state-action pairs similar to the offline dataset.

Proposed Method Sketch

- Constrain the learned policy to visit state-action pairs similar to the offline dataset.

- Technically: regularize the undiscounted stationary state-action distribution of the learned policy towards the dataset during policy learning.

  - Why undiscounted? The offline dataset is just the rollouts of the data-collecting policy.

Proposed Method Sketch

- Constrain the learned policy to visit state-action pairs similar to the offline dataset.

- Technically: regularize the undiscounted stationary state-action distribution of the learned policy towards the dataset during policy learning.
    - Why undiscounted? The offline dataset is just the rollouts of the data-collecting policy.

- More technically: add a tractable regularizer into the policy optimization objective.
    - Only requires samples from the offline dataset, learned policy, and estimated dynamic.
    - The dynamic can be simply learned by the maximum likelihood estimation.

Practical Implementation: SDM–GAN

- Using augmented dataset $\mathcal{D} := f \cdot \mathcal{D}_{\text{env}} + (1 - f) \cdot \mathcal{D}_{\text{model}}$, with $f = 0.5$ as default.

Practical Implementation: SDM–GAN

- Using augmented dataset $\mathcal{D} := f \cdot \mathcal{D}_{\text{env}} + (1-f) \cdot \mathcal{D}_{\text{model}}$, with $f = 0.5$ as default.

- Regularizer construction: needs samples from the true data distribution ($\mathcal{B}_{\text{true}}$), and samples from the policy's distribution ($\mathcal{B}_{\text{fake}}$).

  - $\mathcal{B}_{\text{true}}$ is just samples from the offline dataset.

  - $\mathcal{B}_{\text{fake}}$ is constructed as

$$\mathcal{B}_{\text{fake}} := \begin{bmatrix} (\overset{s \sim \mathcal{D}}{s}, & \overset{a \sim \pi_\phi(\cdot \mid s)}{a}) \\ (\underset{s' \sim \widehat{P}(\cdot \mid s, a)}{s'}, & \underset{a' \sim \pi_\phi(\cdot \mid s')}{a'}) \end{bmatrix}$$

  - Minimize the Jensen–Shannon divergence via the GAN structure.

Practical Implementation: SDM–GAN

- Using augmented dataset $\mathcal{D} := f \cdot \mathcal{D}_{\text{env}} + (1 - f) \cdot \mathcal{D}_{\text{model}}$, with $f = 0.5$ as default.

- Regularizer construction: needs samples from the true data distribution ($\mathcal{B}_{\text{true}}$), and samples from the policy's distribution ($\mathcal{B}_{\text{fake}}$).

  - $\mathcal{B}_{\text{true}}$ is just samples from the offline dataset.

  - $\mathcal{B}_{\text{fake}}$ is constructed as

$s \sim \mathcal{D}$ $\qquad$ $a \sim \pi_\phi(\cdot \mid s)$

$$\mathcal{B}_{\text{fake}} := \begin{bmatrix} (s, & a) \\ (s', & a') \end{bmatrix}$$

$s' \sim \widehat{P}(\cdot \mid s, a)$ $\qquad$ $a' \sim \pi_\phi(\cdot \mid s')$

  - Minimize the Jensen–Shannon divergence via the GAN structure.

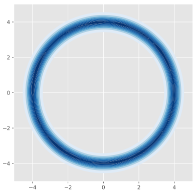
- Implicit policy:

s

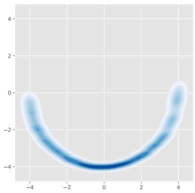
$$\pi_\phi(s, z) = a$$

$z \overset{iid}{\sim} p_z(z)$

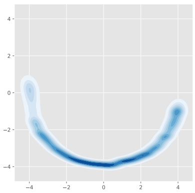
Results: Toy Experiments



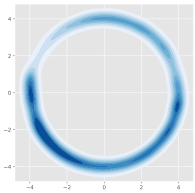
(a) Truth



(b) Deterministic policy



(c) Gaussian policy



(d) Implicit policy

- Behavior-cloning: clone the state (x-axis) action (y-axis) distribution in Fig. (a).
- Compare implicit (Fig. (d)) with deterministic (Fig. (b)) and Gaussian policy (Fig. (c)).
- Both the deterministic and the Gaussian policy fail to capture multiple action modes.
- The implicit policy does capture all the action modes at each state.

Results: Main Method

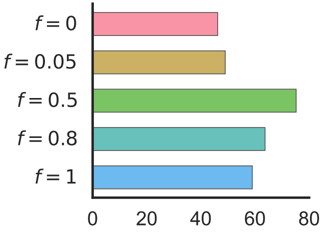
- SDM–GAN achieves competitive performance on the D4RL benchmark.

Task Name	aDICE	CQL	FisherBRC	TD3+BC	OptiDICE	MOPO	COMBO	WMOPO	SDM-GAN
halfcheetah-medium	-2.2	39.0 ± 0.8	41.1 ± 0.6	43.0 ± 0.5	38.2 ± 0.5	47.2 ± 1.0	53.7 ± 2.1	**55.6** ± 1.3	42.5 ± 0.5
walker2d-medium	0.3	60.2 ± 30.8	**78.4** ± 1.8	77.3 ± 4.0	14.3 ± 15.0	0.0 ± 0.1	40.9 ± 28.9	22.7 ± 27.7	**66.7** ± 1.8
hopper-medium	1.2	34.5 ± 11.7	99.2 ± 0.3	**99.6** ± 0.6	92.3 ± 16.9	23.4 ± 7.2	51.8 ± 32.8	**66.5** ± 46.0	62.8 ± 14.3
halfcheetah-medium-replay	-2.1	43.4 ± 0.8	43.2 ± 1.3	41.9 ± 2.0	39.8 ± 0.8	**52.5** ± 1.4	51.8 ± 1.6	51.8 ± 5.6	41.7 ± 0.4
walker2d-medium-replay	0.6	16.4 ± 6.6	38.4 ± 16.6	24.6 ± 6.7	20.2 ± 5.8	51.9 ± 15.8	14.2 ± 11.9	**54.8** ± 12.3	20.3 ± 4.0
hopper-medium-replay	1.1	29.5 ± 2.3	33.4 ± 2.8	31.4 ± 2.7	29.0 ± 4.9	47.1 ± 16.2	34.5 ± 2.0	**93.9** ± 1.9	30.6 ± 2.8
halfcheetah-medium-expert	-0.8	34.5 ± 15.8	**92.5** ± 8.5	90.1 ± 6.9	91.2 ± 16.6	**92.1** ± 8.3	90.0 ± 10.5	42.7 ± 13.0	89.1 ± 6.6
walker2d-medium-expert	0.4	79.8 ± 22.7	**98.2** ± 13.1	96.1 ± 15.8	67.1 ± 30.2	36.0 ± 49.6	61.3 ± 36.1	48.6 ± 37.0	**97.9** ± 4.9
hopper-medium-expert	1.1	103.5 ± 20.2	112.3 ± 0.3	111.9 ± 0.3	101.8 ± 18.5	27.8 ± 3.6	**112.6** ± 1.8	97.8 ± 19.3	104.5 ± 5.4
maze2d-large	-0.1	43.7 ± 18.6	-2.1 ± 0.4	87.6 ± 15.4	130.7 ± 56.1	-	-	-	**207.7** ± 11.7
maze2d-medium	10.0	30.7 ± 9.8	4.6 ± 20.4	59.1 ± 47.7	**140.8** ± 44.0	-	-	-	115.4 ± 34.2
maze2d-umaze	-15.7	50.5 ± 7.9	-2.3 ± 17.9	13.8 ± 22.8	**107.6** ± 33.1	-	-	-	36.1 ± 28.4
pen-human	-3.3	2.1 ± 13.7	0.0 ± 3.9	-1.7 ± 3.8	-0.1 ± 5.6	-	-	-	**17.8** ± 1.7
pen-cloned	-2.9	1.5 ± 6.2	-2.0 ± 0.8	-2.4 ± 1.4	1.4 ± 6.8	-	-	-	**40.6** ± 6.1
pen-expert	-3.5	95.9 ± 18.1	31.6 ± 24.4	32.4 ± 24.3	-1.1 ± 4.7	-	-	-	**135.8** ± 11.7
door-expert	0.0	87.9 ± 21.6	57.6 ± 37.7	-0.3 ± 0.0	87.9 ± 25.8	-	-	-	**93.5** ± 6.7

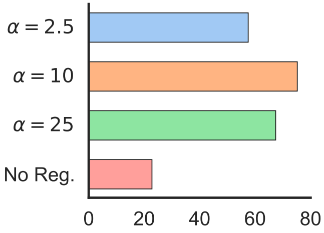
Learn well on the MuJoCo datasets

Robust and good performance on
the challenging Adroit and Maze2D datasets

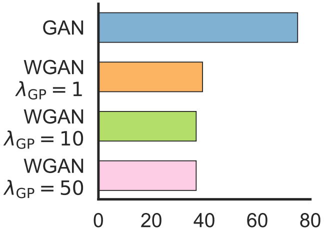
Results: Ablation Study



(a) Vary proportion of synthetic data



(b) Vary regularization strength



(c) JSD *v.s.* IPM (Wasserstein-1 dual)

- Synthetic data help learning, but too many can be harmful.
- SDM–GAN is relatively robust to the regularization strength, but cannot remove it.
- SDM–WGAN overall performs worse than SDM–GAN $\rightarrow$ future work on other IPM.

Summary

- Goal: match the undiscounted stationary state-action distribution of the learned policy with the dataset.
- Method: SDM–GAN, offline MBRL method + novel regularizer + flexible policy.

Please scan this QR code for the full paper!