Adaptive Policy Optimization for Model-based Offline Reinforcement Learning

Yijun Yang[1,2] Jing Jiang[1] Zhuowei Wang[1] Qiqi Duan[2] Yuhui Shi[2]
[1]University of Technology Sydney, Australia
[2]Southern University of Science and Technology, China

Abstract

Offline reinforcement learning (offline RL) aims to train an agent solely using a dataset of historical interactions with the environments without any further costly or dangerous active exploration. Model-based RL (MbRL) usually achieves promising performance in offline RL due to its high sample-efficiency and compact modeling of a dynamic environment. However, it may suffer from the bias and error accumulation of the model predictions. Existing methods address this problem by adding a penalty term to the model reward but require careful hand-tuning of the penalty and its weight. Instead in this paper, we formulate the model-based offline RL as a bi-objective optimization where the first objective aims to maximize the model return and the second objective is adaptive to the learning dynamics of the RL policy. Thereby, we do not need to tune the penalty and its weight but can achieve a more advantageous trade-off between the final model return and model's uncertainty. We develop an efficient and adaptive policy optimization algorithm equipped with evolution strategy to solve the bi-objective optimization, named as BiES. The experimental results on a D4RL benchmark show that our approach sets the new state of the art and significantly outperforms existing offline RL methods on long-horizon tasks.

Contribution

- We propose a bi-objective policy optimization algorithm, i.e., BiES, in which the first objective aims to maximize the model return, and the second one synchronously calibrates the learning bias of the policy. Our method achieves more stable policy improvement on offline MbRL tasks.
- To the best of our knowledge, our approach is the first to adopt evolution strategy (ES) to model-based offline RL problems and solves the optimization under uncertain and long-horizon RL tasks. We also theoretically establish an upper bound for the norm of a BiES-based gradient estimation.
- We conduct a large-scale empirical study on offline MuJoCo locomotion tasks from the D4RL benchmark. The experimental results show that our method attains state-of-the-art results compared to other offline RL algorithms.

Model-based Offline RL

Recent work has demonstrated that model-based RL is a promising paradigm for offline policy learning due to its high sample-efficiency and compact modeling of a dynamic environment. From the following flowchart, model-based offline RL usually consists of two stages: one is the model learning stage where we learn an environment model from a fixed dataset by supervised learning, and the other is the policy learning stage, in which we consider the learned environment model as a virtual environment and conduct online RL interacting with it. In this offline policy learning stage, we aim at solving the objective function $\max_\pi \hat{R}_{\hat{\rho}_0}(\pi, \widehat{\mathcal{M}}) = \mathbb{E}_{s_0\sim\hat{\rho}_0,\pi}\left[\sum_{h=0}^{H-1} \hat{r}\left(\hat{s}_h, a_h\right)\right]$ by existing online RL methods such as PPO or SAC.

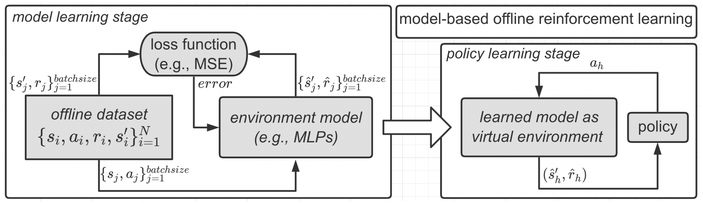

Figure 1:Illustration of model-based offline RL.

In practice, the uncertainty and accuracy of the environment model will drastically vary across different state-action pairs. In this case, an RL agent can easily exploit the model so that the objective function is biased by the overestimation of the agent's performance. In order to address this problem, existing works developed a conservative objective function that adds an uncertainty penalty term into the original reward at each time step: $\max_\pi \hat{R}_{\hat{\rho}_0}(\pi, \widehat{\mathcal{M}}) = \mathbb{E}_{s_0\sim\hat{\rho}_0,\pi}\left[\sum_{h=0}^{H-1}(\hat{r}\left(\hat{s}_h, a_h\right) - u(\hat{s}_h, a_h))\right]$. By carefully tuning the penalty weight λ, we can achieve preferred trade-offs between the model reward and model uncertainty. Despite being intuitive, this method's performance is very sensitive to the penalty weight. And tuning λ is usually challenging and costly for model-based offline RL.

Motivation: How to find a "balanced" λ efficiently?

Our Solution: Bi-objective Optimization

$$\max_\theta \mathbf{J}_{\hat{\rho}_0}(\pi_\theta, \widehat{\mathcal{M}}) = \max_\theta (J^{\hat{r}}_{\hat{\rho}_0}(\pi_\theta, \widehat{\mathcal{M}}), J^{u}_{\hat{\rho}_0}(\pi_\theta, \widehat{\mathcal{M}}))^\mathsf{T}$$

$$(J^{\hat{r}}_{\hat{\rho}_0}(\pi_\theta, \widehat{\mathcal{M}}), J^{u}_{\hat{\rho}_0}(\pi_\theta, \widehat{\mathcal{M}}))^\mathsf{T} = \mathbb{E}_{s_0\sim\hat{\rho}_0,\pi}\left[\sum_{h=0}^{H-1}\left(\hat{r}\left(\hat{s}_h, a_h\right), -u\left(\hat{s}_h, a_h\right)\right)^\mathsf{T}\right]$$

Overview of the Proposed Method

Instead of formulating the problem as a single-objective optimization with regularization as in previous works, a primary contribution of this paper is to treat the model reward/return and model uncertainty as two separate objectives and develop an efficient bi-objective optimization method producing a set of diverse policies on the Pareto front, which correspond to different trade-offs (i.e., various selections of λ) between the two objectives. Thereby, when deployed to a new actual environment, we can accordingly choose the best policy from the set.

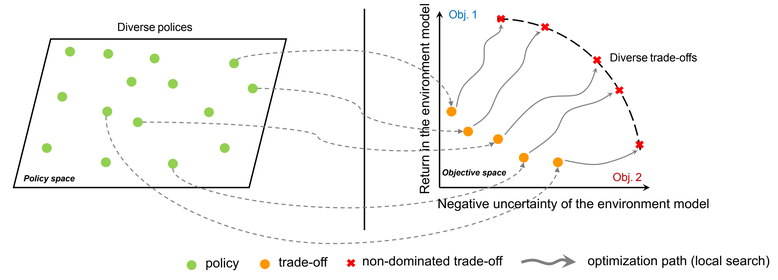

Figure 2:Illustration of our method. Our method contains two steps: (1) we randomly initialize a diverse population of polices that are uniformly distributed in the policy space. Each of them represents a different trade-off in the objective space (for better diversity); (2) for each of policies in the population, our method performs a gradient-based local search toward the Pareto front (for better efficiency).

Empirical Results

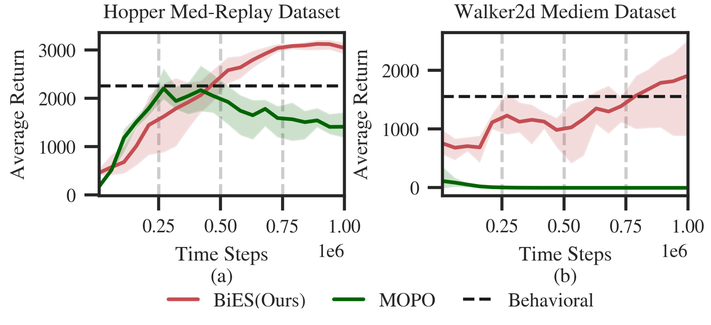

Figure 3:A proof-of-concept experiment on two offline RL tasks from the D4RL Gym benchmark. We evaluated BiES and compared it with a state-of-the-art offline MbRL algorithm MOPO. Our method achieves more stable policy improvement when using long-horizon model rollouts.