How to Select the Appropriate One From the Trained Models for Model-Based OPE : Appendix

Chongchong Li[1], Yue Wang[1], Zhi-Ming Ma[2], and Yuting Liu[1]

[1] Beijing Jiaotong University
{18118002,ytliu}@bjtu.edu.cn
[2] Academy of Mathematics and Systems Science, Chinese Academy of Sciences

Abstract. Offline Policy Evaluation (OPE) is a method for evaluating and selecting complex policies in reinforcement learning for decision-making using large, offline datasets. Recently, Model-Based Offline Policy Evaluation (MBOPE) methods have become popular because they are easy to implement and perform well. The model-based approach provides a mechanism for approximating the value of a given policy directly using estimated transition and reward functions of the environment. However, a challenge remains in selecting an appropriate model from those trained for further use. We begin by analyzing the upper bound of the difference between the true value and the approximated value calculated using the model. Theoretical results show that this difference is related to the trajectories generated by the given policy on the learned model and the prediction error of the transition and reward functions at these generated data points. We then propose a novel criterion inspired by the theoretical results to determine which trained model is better suited for evaluating the given policy. Finally, we demonstrate the effectiveness of the proposed method on both simulated and benchmark offline datasets.

Keywords: Reinforcement Learning · Model Based · Offline Policy Evaluation.

Table of Contents

1 Details for Experiments

1.1 Details for Synthetic Experiment

In this simulation experiment, we consider a one-dimensional problem of movement. The true model $s' = f(s, a)$ is defined as $s' = s+a$, where s' represents the next state, s denotes the current state, and a represents the action taken. The ground truth reward function $r(s, a)$ is defined as $r = -|s+a-5|^2-a^2$. We evaluate two policies: π_1, which is defined as $a = \max(\min(1, -0.5\times(s-(-7.5))), -1)$ and π_2, which is defined as $a = \max(\min(1, -0.5 \times (s - 7.5)), -1)$. Note that π_1 mostly makes the agent move towards the negative axis, while π_2 does not.

A series of synthetic models were designed, where the predictive accuracy varies on different sections of the axes for each model. To construct the synthetic models, we designed a fake model $s' = \hat{f}(s, a)$ with the following specifications: $s' = s + a + \epsilon^-$ if $s < 0$ and $s' = s + a + \epsilon^+$ if $s > 0$, where ϵ^- and ϵ^+ are noise terms drawn from normal distributions with different standard deviations, σ_1 and σ_2, respectively. For the convenience of describing the loss of the model, in this synthetic experiment, we use the negative log-likelihood instead of The MSE loss. To keep the total validation loss of the model constant, we set $\sigma_1 \times \sigma_2$ to a fixed value of 0.05. By varying σ_1 in the synthetic models, we were able to construct a range of models. It should be noted that σ_2 changes accordingly. In this study, we increased σ_1 gradually from 0.05 to 0.5. Then we can see how the error of the estimated value for each policy changes and whether the criterion can help pick up models with smaller estimated errors. Note that the validation loss is fixed for each model.

The Figure 1 displays the results of experiments conducted on synthetic data. The left column shows the calculated criterion, the validation loss for each model, and the estimated value for π_1 using each model. The models are identified by their prediction error on the negative half-axis. The right column displays the results for π_2. For policy π_1, which primarily moves the agent on the negative half-axis, the criterion calculated and the error of the estimated value increase simultaneously as the error of the model on the negative half-axis increases. The validation loss, however, remains constant. Furthermore, the model error on the positive half-axis decreases since the total model error is fixed. As a result, the error of the estimated value for policy π_2, which moves the agent mostly on the positive half-axis, decreases, and the corresponding criterion calculated decreases as well. These results indicate that the validation loss is insufficient for selecting a better model used for OPE for a given policy. The proposed criterion, however, can be beneficial.

1.2 Details for Benchmark Data Sets

In this paper we use the datasets from RL Unplugged [4], an offline RL benchmark suite based on Deepmind control-suite [7]. We show the basic information of RL Unplugged dataset in Table 1. The dataset is available[3].

[3] https://github.com/deepmind/deepmind-research/tree/master/rl_unplugged

Table 1. Summary of RL Unplugged datasets.

Environment	State dim.	Action dim.	Size
cartpole_swingup	5	1	40K
cheetah_run	17	6	300K
finger_turn_hard	12	2	500K
fish_swim	24	5	200K
walker_stand	24	6	200K
walker_walk	24	6	200K
humanoid_run	67	21	3M

As for the policies to be evaluated, we use the benchmark from a previous work [3]. For each environment, there are around 96 policies trained with behavioral cloning [1], D4PG [2], Critic Regularized Regression [8], and RABM [6] and snapshots along the training trajectory. The policies are available now [4].

1.3 Details for Experiments on Benchmark

We randomly split the data into two parts, allocating eighty percent for training and the remaining for validation. For each environment, we trained a series of models by adjusting the hyper-parameters, including the number of layers, the number of hidden nodes, learning rates, and weight decay, with a range of options. Specifically, we parameterized the models using fully connected neural networks, with 3 or 5 layers and 512 or 1024 hidden nodes. The learning rates are {1e-3,3e-4}. We use weight decay to avoid over-fitting and the weights are {0, 1e-6}. Models are trained by training data for a fixed number of epochs, the max epoch times are {100,200,500}. Overall, we obtained a total of 48 models for each environment. To select a model for OPE tasks, we estimated the true value of each policy using the Monte Carlo method, generating 256 real trajectories to approximate the value. To estimate the value using MBOPE and the learned model, we randomly sampled a batch of 256 initial states and calculated the averaged discounted return using a discount factor γ of 0.995. To evaluate a policy and a learned model, we set the horizon H to the default parameters for the environment, and the parameter N to 256.

1.4 More Results

We compare the performance of different methods by calculating Spearman's rank correlation between ground truth values and the estimated ones. Results are shown in Table 2. We can see that the rank correlation calculated using the proposed method is higher than using the validation loss in most environments.

[4] https://github.com/google-research/deep_ope

Table 2. Spearman's rank correlation between ground truth values and the estimated ones. The discount factor used is 0.995.

Environment	Validation loss	Ours w/o d	**Ours**
cartpole_swingup	0.71	0.72	**0.73**
cheetah_run	0.55	0.53	**0.60**
finger_turn_hard	**0.08**	-0.13	-0.03
fish_swim	0.35	0.37	**0.50**
walker_stand	0.32	0.14	**0.41**
walker_walk	0.20	**0.38**	0.24
humanoid_run	-0.58	**0.49**	0.24

2 Pseudocode

The algorithm pseudo-code of MBOPE is shown in Algorithm 1.

Algorithm 1 Model-based offline policy evaluation [9]

Require: the learned model $\hat{M}_\theta$, policy π, discount factor γ, horizon H, set of initial states S_0.

for $i = 1, 2, \ldots, n$ **do**
  $R_i \leftarrow 0$
  Sample initial state s_0 from S_0.
  $\hat{s}_0 = s_0$
  for $t = 0, 1, \ldots, H-1$ **do**
    Sample action using policy π, $a_t \sim \pi(\hat{s}_t)$
    Sample next state and reward using $\hat{M}_\theta$:
    $\hat{s}_{t+1}, \hat{r}_{t+1} \sim \hat{M}_\theta(\hat{s}_t, a_t)$
    $R_i \leftarrow R_i + \gamma^t \hat{r}_{t+1}$
  end for
end for
$R = 1/n \sum_{i=1}^{n} R_i$
return R

3 Proof of the Theorem

Theorem 1. *Let η_π be the true value of the policy π, $\hat{\eta}_\pi$ be the estimated value using the learned model, then we have*

$$|\eta_\pi - \hat{\eta}_\pi| \leq C \cdot \mathbb{E}_{t \sim Gemo(\gamma)} \mathbb{E}_{s' \sim P^t_{mix}, a' \sim \pi(s')} \mathcal{L}(s', a')$$

where $P^t_{mix} = \beta P^t_\pi + (1-\beta)\hat{P}^t_\pi$, $C = \frac{2\gamma r_{max}}{(1-\gamma)^2}$. $\mathcal{L}(s', a')$, the error of the model at (s', a'), is $D_{TV}\left(P(s|s', a') \| \hat{P}(s|s', a')\right)$ and $Gemo(\gamma)$ is a geometric distribution with parameter γ.

In this paper, P^t_{mix} represents the mixed distribution of P^t_π and $\hat{P}^t_\pi$, where P^t_π is the state distribution of the Markov process at time step t given policy π and the environment, while $\hat{P}^t_\pi$ is the state distribution given the learned model. C is a constant. r_{max} is the maximum of the reward. D_{TV} is the total variation distance and here portrays the error of the model.

Before proving Theorem 1, we will introduce some useful lemmas. Lemma 1 is used to derive Lemma 2, which upper-bounds the difference in the probabilities of observing state-action pairs between the true environment and the learned model. Lemma 3 upper-bounds the error in the estimated value as the sum of the differences between the environment and the learned model over time steps. Using Lemma 2 and Lemma 3, we can bound the error of the estimated value for a given policy as the sum of the model error over time steps. Finally, we use Lemma 4 and Lemma 5 to rewrite the bound in the form of expectations.

Lemma 1. *For $\forall$ policy π, we have*

$$\begin{aligned}\left|P^t_\pi(s,a)-\hat{P}^t_\pi(s,a)\right| \le \mathbb{E}_{s'\sim P^{t-1}_\pi, a'\sim\pi(s')}\left|P(s,a|s',a')-\hat{P}(s,a|s',a')\right| + \\ \sum_{s',a'}\hat{P}(s,a|s',a')\left|P^{t-1}_\pi(s',a')-\hat{P}^{t-1}_\pi(s',a')\right|, \end{aligned} \tag{1}$$

and

$$\begin{aligned}\left|P^t_\pi(s,a)-\hat{P}^t_\pi(s,a)\right| \le \mathbb{E}_{s'\sim \hat{P}^{t-1}_\pi, a'\sim\pi(s')}\left|P(s,a|s',a')-\hat{P}(s,a|s',a')\right| + \\ \sum_{s',a'}P(s,a|s',a')\left|P^{t-1}_\pi(s',a')-\hat{P}^{t-1}_\pi(s',a')\right|, \end{aligned} \tag{2}$$

where $P^t_\pi(s,a)$ is the probability of observing the state s and action a at time step t by interacting with the true environment using policy π, while $\hat{P}^t_\pi(s,a)$ induced by the learned model $\hat{M}$.

Proof of Lemma 1.

Proof.

$$\left|P^t_\pi(s,a)-\hat{P}^t_\pi(s,a)\right|$$

$$= \left|\sum_{s',a'}\left[P(s,a|s',a')P^{t-1}_\pi(s',a')-\hat{P}(s,a|s',a')\hat{P}^{t-1}_\pi(s',a')\right]\right|$$

$$\begin{aligned}= \Bigg|\sum_{s',a'}\Big[&P(s,a|s',a')P^{t-1}_\pi(s',a')-\hat{P}(s,a|s',a')P^{t-1}_\pi(s',a')+\hat{P}(s,a|s',a')P^{t-1}_\pi(s',a') \\ &-\hat{P}(s,a|s',a')\hat{P}^{t-1}_\pi(s',a')\Big]\Bigg|\end{aligned}$$

$$
\begin{aligned}
&\leq \mathbb{E}_{s'\sim P_\pi^{t-1}, a'\sim\pi(s')}\left|P(s,a|s',a') - \hat{P}(s,a|s',a')\right| \\
&\quad + \sum_{s',a'} \hat{P}(s,a|s',a')\left|P_\pi^{t-1}(s',a') - \hat{P}_\pi^{t-1}(s',a')\right|,
\end{aligned}
$$

The proof of another inequality in this lemma is similar.

$$
\begin{aligned}
&\left|P_\pi^t(s,a) - \hat{P}_\pi^t(s,a)\right| \\
&= \left|\sum_{s',a'}\left[P(s,a|s',a')P_\pi^{t-1}(s',a') - \hat{P}(s,a|s',a')\hat{P}_\pi^{t-1}(s',a')\right]\right| \\
&= \left|\sum_{s',a'}\left[P(s,a|s',a')P_\pi^{t-1}(s',a') - P(s,a|s',a')\hat{P}_\pi^{t-1}(s',a') + P(s,a|s',a')\hat{P}_\pi^{t-1}(s',a')\right.\right. \\
&\quad \left.\left. - \hat{P}(s,a|s',a')\hat{P}_\pi^{t-1}(s',a')\right]\right| \\
&\leq \mathbb{E}_{s'\sim \hat{P}_\pi^{t-1}, a'\sim\pi(s')}\left|P(s,a|s',a') - \hat{P}(s,a|s',a')\right| \\
&\quad + \sum_{s',a'} P(s,a|s',a')\left|P_\pi^{t-1}(s',a') - \hat{P}_\pi^{t-1}(s',a')\right|.
\end{aligned}
$$

Now we get the final conclusion.

Lemma 2. *Let D_{TV} be the total variation distance, we have*

$$
D_{TV}\left(P_\pi^t(s,a)\|\hat{P}_\pi^t(s,a)\right) \leq \sum_{i=1}^{t} \mathbb{E}_{s'\sim P_\pi^{i-1}, a'\sim\pi(s')} D_{TV}\left(P(s|s',a')\|\hat{P}(s|s',a')\right) \tag{3}
$$

Similarly, we have

$$
D_{TV}\left(P_\pi^t(s,a)\|\hat{P}_\pi^t(s,a)\right) \leq \sum_{i=1}^{t} \mathbb{E}_{s'\sim \hat{P}_\pi^{i-1}, a'\sim\pi(s')} D_{TV}\left(P(s|s',a')\|\hat{P}(s|s',a')\right). \tag{4}
$$

Proof of Lemma 2.

Proof. Now we prove Equation 3. Firstly, by definition of the total variation distance we can get that

$$
D_{TV}\left(P_\pi^t(s,a)\|\hat{P}_\pi^t(s,a)\right) = 1/2\sum_{s,a}\left|P_\pi^t(s,a) - \hat{P}_\pi^t(s,a)\right|.
$$

Then according to Equation 1 in Lemma 1, we have

$$
D_{TV}\left(P_\pi^t(s,a)\|\hat{P}_\pi^t(s,a)\right)
$$

$$\leq \frac{1}{2}\sum_{s,a}\mathbb{E}_{s'\sim P_\pi^{t-1},a'\sim\pi(s')}\left|P(s,a|s',a')-\hat{P}(s,a|s',a')\right|$$

$$+\frac{1}{2}\sum_{s,a}\sum_{s',a'}\hat{P}(s,a|s',a')\left|P_\pi^{t-1}(s',a')-\hat{P}_\pi^{t-1}(s',a')\right|$$

$$=\mathbb{E}_{s'\sim P_\pi^{t-1},a'\sim\pi(s')}D_{TV}\left(P(s,a|s',a')\|\hat{P}(s,a|s',a')\right)$$
$$+D_{TV}\left(P_\pi^{t-1}(s,a)\|\hat{P}_\pi^{t-1}(s,a)\right)$$

$$\leq\sum_{i=1}^{t}\mathbb{E}_{s'\sim P_\pi^{i-1},a'\sim\pi(s')}D_{TV}\left(P(s,a|s',a')\|\hat{P}(s,a|s',a')\right)$$

According to Lemma B.1 in [5], we have

$$D_{TV}\left(P(s,a|s',a')\|\hat{P}(s,a|s',a')\right)\leq D_{TV}\left(P(s|s',a')\|\hat{P}(s|s',a')\right),$$

and then arrive at the final conclusion. As for the Equation 4, it is similar just by applying Equation 2 in Lemma 1 instead.

Lemma 3. *Let η_π be the true value of the policy π, $\hat{\eta}_\pi$ be the estimated value by using the learned model, r_{max} be the maximum of the rewards, then we have*

$$|\eta_\pi-\hat{\eta}_\pi|\leq 2r_{max}\sum_t\gamma^t D_{TV}\left(P_\pi^t(s,a)\|\hat{P}_\pi^t(s,a)\right).$$

Proof of Lemma 3.

Proof.

$$|\eta_\pi-\hat{\eta}_\pi|$$
$$=\left|\sum_{s,a}\left(P_\pi(s,a)-\hat{P}_\pi(s,a)\right)r(s,a)\right|$$
$$=\left|\sum_{s,a}\left(\sum_t\gamma^t\left(P_\pi^t(s,a)-\hat{P}_\pi^t(s,a)\right)r(s,a)\right)\right|$$
$$=\left|\sum_{t}\left(\sum_{s,a}\gamma^t\left(P_\pi^t(s,a)-\hat{P}_\pi^t(s,a)\right)r(s,a)\right)\right|$$
$$\leq\sum_t\sum_{s,a}\gamma^t\left|P_\pi^t(s,a)-\hat{P}_\pi^t(s,a)\right||r(s,a)|$$
$$\leq 2r_{max}\sum_t\gamma^t D_{TV}\left(P_\pi^t(s,a)\|\hat{P}_\pi^t(s,a)\right)$$

The last inequality is derived by applying the definition of the TV distance.

Lemma 4. *$\beta\mathbb{E}_{x\sim p_1}f(x)+(1-\beta)\mathbb{E}_{x\sim p_2}f(x)=\mathbb{E}_{x\sim p_3}f(x)$, where p_3 is the mixture of p_1 and p_2. The distribution is $p_3=\beta p_1+(1-\beta)p_2$.*

Lemma 5. *$\sum_{t=0}^{\infty}\gamma^t x_t = 1/(1-\gamma)\mathbb{E}_{t\sim Gemo(\gamma)}x_t$*

Proof of Lemma 5.

Proof. $\sum_{t=0}^{\infty}\gamma^t x_t = 1/(1-\gamma)\sum_{t=0}^{\infty}\gamma^t(1-\gamma)x_t$, and $\gamma^t(1-\gamma)$ is the probability of a geometric distribution with parameter γ.

Now we proof the Theorem 1 by using the previously proven lemmas.

Proof.

$$|\eta_\pi-\hat{\eta}_\pi| \le 2r_{max}\sum_t \gamma^t D_{TV}\left(P_\pi^t(s,a)\|\hat{P}_\pi^t(s,a)\right) \tag{5}$$

$$\begin{aligned}
&= 2r_{max}\sum_t \gamma^t\big[\beta D_{TV}\left(P_\pi^t(s,a)\|\hat{P}_\pi^t(s,a)\right)\\
&\quad + (1-\beta)D_{TV}\left(P_\pi^t(s,a)\|\hat{P}_\pi^t(s,a)\right)\big]\\
&\le 2r_{max}\cdot\sum_t \gamma^t\big[\sum_{i=1}^{t}\beta\mathbb{E}_{s'\sim P_\pi^{i-1},a'\sim\pi(s')}\mathcal{L}(s',a')\\
&\quad + (1-\beta)\mathbb{E}_{s'\sim \hat{P}_\pi^{i-1},a'\sim\pi(s')}\mathcal{L}(s',a')\big]
\end{aligned} \tag{6}$$

$$= 2r_{max}\sum_t \gamma^t\big[\sum_{i=1}^{t}\mathbb{E}_{s'\sim P_{mix}^{i-1},a'\sim\pi(s')}\mathcal{L}(s',a')\big] \tag{7}$$

$$\begin{aligned}
&= 2r_{max}\sum_{t=0}^{\infty}\big[\mathbb{E}_{s'\sim P_{mix}^{t},a'\sim\pi(s')}\mathcal{L}(s',a')\cdot(\sum_{i=t+1}^{\infty}\gamma^i)\big]\\
&= 2r_{max}\sum_{t=0}^{\infty}\big[\mathbb{E}_{s'\sim P_{mix}^{t},a'\sim\pi(s')}\mathcal{L}(s',a')\cdot\frac{\gamma^{t+1}}{1-\gamma}\big]\\
&= \frac{2\gamma r_{max}}{1-\gamma}\cdot\sum_{t=0}^{\infty}\gamma^t\mathbb{E}_{s'\sim P_{mix}^{t},a'\sim\pi(s')}\mathcal{L}(s',a')\\
&= \frac{2\gamma r_{max}}{(1-\gamma)^2}\cdot\mathbb{E}_{t\sim Gemo(\gamma)}\mathbb{E}_{s'\sim P_{mix}^{t},a\sim\pi(s')}\mathcal{L}(s',a').
\end{aligned} \tag{8}$$

Note that Eq. 5 is derived using Lemma 3. And Eq. 6 is derived using Lemma 2. Eq. 7 is derived using Lemma 4 and Eq. 8 is derived using Lemma 5.

References

1. Bain, M., Sammut, C.: A framework for behavioural cloning. In: Machine Intelligence 15, Intelligent Agents. p. 103–129. Oxford University, GBR (1999)
2. Barth-Maron, G., Hoffman, M.W., Budden, D., Dabney, W., Horgan, D., TB, D., Muldal, A., Heess, N., Lillicrap, T.: Distributional policy gradients. In: ICLR (2018)

3. Fu, J., Norouzi, M., Nachum, O., Tucker, G., ziyu wang, Novikov, A., Yang, M., Zhang, M.R., Chen, Y., Kumar, A., Paduraru, C., Levine, S., Paine, T.: Benchmarks for deep off-policy evaluation. In: ICLR (2021)
4. Gulcehre, C., Wang, Z., Novikov, A., Paine, T., Gómez, S., Zolna, K., Agarwal, R., Merel, J.S., Mankowitz, D.J., Paduraru, C., Dulac-Arnold, G., Li, J., Norouzi, M., Hoffman, M., Heess, N., de Freitas, N.: RL Unplugged: A Suite of Benchmarks for Offline Reinforcement Learning. In: NeurIPS. vol. 33, pp. 7248–7259 (2020)
5. Janner, M., Fu, J., Zhang, M., Levine, S.: When to Trust Your Model: Model-Based Policy Optimization. In: NeurIPS. vol. 32 (2019)
6. Siegel, N., Springenberg, J.T., Berkenkamp, F., Abdolmaleki, A., Neunert, M., Lampe, T., Hafner, R., Heess, N., Riedmiller, M.: Keep doing what worked: Behavior modelling priors for offline reinforcement learning. In: ICLR (2020)
7. Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., Casas, D.d.L., Budden, D., Abdolmaleki, A., Merel, J., Lefrancq, A., Lillicrap, T., Riedmiller, M.: DeepMind Control Suite. arXiv:1801.00690 [cs] (Jan 2018)
8. Wang, Z., Novikov, A., Zolna, K., Merel, J.S., Springenberg, J.T., Reed, S.E., Shahriari, B., Siegel, N., Gulcehre, C., Heess, N., de Freitas, N.: Critic Regularized Regression. In: NeurIPS. vol. 33, pp. 7768–7778 (2020)
9. Zhang, M.R., Paine, T., Nachum, O., Paduraru, C., Tucker, G., ziyu wang, Norouzi, M.: Autoregressive dynamics models for offline policy evaluation and optimization. In: ICLR (2021)