Offline Reinforcement Learning with Pessimistic Value Priors

Filippo Valdettaro[1]* and A. Aldo Faisal[1,2,3]

[1]Brain & Behaviour Lab, Dept. of Computing and Bioengineering, Imperial College London, UK
[2]Department of Digital Health & Data Science, University of Bayreuth, Germany
[3]Alan Turing Institute

Abstract

We mitigate the effect of distribution shift in offline reinforcement learning by regularisation through value function inference with a pessimistic prior as a mechanism to induce critic conservatism and avoid unsupported policies. By introducing a pessimistic prior on the value of the learned policy and carrying out inference in value function space, the resulting posterior will only have high action-values in regions where these are supported by the dataset. Regularisation through inference has the potential to be not as aggressively conservative as other forms of regularisation, such as those that try to be robust to worst-case outcomes given the data, while still avoiding out-of-distribution actions. We develop this approach for continuous control and propose a way to make it scalable and compatible with deep learning architectures. As a byproduct of this inference scheme we also obtain consistent Bayesian uncertainty for model-free off-policy evaluation from a non-episodic dataset of individual transitions. We develop this framework for control in continuous-action environments and present results on a toy environment with exact inference and preliminary results on a scalable, deep version of our framework on a D4RL benchmark robotics task. Our methods show potential for improved performance on such a task, and suggest that future experimental work on improving training stability of our methods could result in effective offline reinforcement learning algorithms coming from simple modifications of online algorithms.

1 Introduction

Offline reinforcement learning (RL) harnesses the information in static datasets from pre-collected interactions with an environment to find good policies without further direct interaction, thus vastly increasing the potential applicability of RL to tasks where direct agent interactions are costly or dangerous such as healthcare [Gottesman et al., 2019, Komorowski et al., 2018], robotics [Sinha et al., 2022, Kalashnikov et al., 2018, Dasari et al., 2020, Kendall et al., 2019] or recommender systems [Huang et al., 2022, Xiao and Wang, 2021]. Learning a policy from a static dataset, however, comes with the additional challenge that the agent cannot test its beliefs of what is a good policy during training. If unaddressed, the distributional shift due to the mismatch between learned and behaviour policies can lead to poorly performing, unsupported policies for datasets with insufficient coverage of the entire state-action space [Fujimoto et al., 2019].

One of the approaches suggested to combat this effect is to train a critic that learns a *conservative* or *pessimistic* estimate of the value function. For example, the learned value function can be augmented with a penalty for those state-actions with high uncertainty for instance as quantified

*filippo.valdettaro20@imperial.ac.uk

Workshop on Aligning Reinforcement Learning Experimentalists and Theorists (ARLET 2024).

through disagreement across an ensemble of critics [Ghasemipour et al., 2022, An et al., 2021] , which can itself incur heavy computational costs. Sidestepping uncertainty quantification, other methods attempt to train a critic that learns a lower bound of the value function given the observed data, thus being robust to the worst-case outcome given the observed data Kumar et al. [2020], Cheng et al. [2022]. Such forms of worst-case robustness, however, can be excessively pessimistic.

We investigate here an alternative pathway into conservative value function estimation for offline RL, building on the approach in Valdettaro and Faisal [2024], extending it to continuous control and scaling it to deep learning architectures. This approach is inspired by regularisation through inference in value-function space as a less aggressive form of critic regularisation than worst-case robustness that still avoids unsupported actions. We train a critic with a pessimistic value prior (PVP), so that after inference only those state-actions that are supported by the data will be assigned high values. We achieve this deterministic MDPs in an off-policy model-free way by setting up inference that respects the temporal-difference (TD) structure of a Markovian environment to get consistent value functions after conditioning on the observed data. In particular, we develop this framework for continuous control with exact inference showcasing results on a toy environment. We then propose a scalable implementation of PVPs for offline RL. We achieve this by adding a simple term to the critic's loss coming from the prior's regularising effect, which constitutes a small modification applicable to any online off-policy algorithm. We confirm the scalable version of PVPs maintains the key desirable behaviours displayed by the exact inference approach on the toy environment and show preliminary results on the halfcheetah-medium D4RL benchmark task [Fu et al., 2020].

2 Related work

We review here motivation for learning conservative value estimates and existing approaches that do so, as well as works relevant to carrying out model-free inference in RL. We also cover inference in function space, which enables Bayesian reasoning directly in terms of function outputs rather than parameters, which is necessary to introduce a pessimistic prior over value.

2.1 Offline RL and conservative value estimation

One key challenge of offline RL is to ensure that the distribution shift caused by differences in the data-generating and learned policies does not cause the algorithm to converge to an unsupported policy. One natural approach involves including some loss term during training that regularises the actor to be similar to the data-generating policy either explicitly [Fujimoto and Gu, 2021] or implicitly [Nair et al., 2020]. However, a fundamental challenge faced by this line of research is that it relies on having a dataset with a high proportion of good demonstrations, and the presence of suboptimal actions in the dataset will negatively affect the quality of the learned policy. An attempt to sidestep this limitation involves trying to learn an actor that picks actions that are merely in the support of the dataset policy. This can be done for example by learning a generative model trained to imitate the behaviour policy and work in it's latent space, so that any sampled actions will also be in support Fujimoto et al. [2019], Zhou et al. [2021]. However, this strongly relies on being able to accurately model the dataset's policy, which is in itself a non-trivial learning task.

In contrast to behaviour regularisation that explicitly constrains the learned policy to be by some measure similar to the dataset policy, methods that estimate conservative policy values are in principle not sensitive to the inclusion of poor quality demonstrations in the dataset. Uncertainty-based methods [An et al., 2021, Ghasemipour et al., 2022] regress the action-values against targets derived from some uncertainty-aware pessimistic value estimate of the ensemble to avoid unwarranted value overestimation in unsupported and therefore uncertain regions. However, large ensembles come with a significant computational burden and while empirically effective for uncertainty quantification in supervised learning [Lakshminarayanan et al., 2017], their immediate theoretical soundness is not guaranteed [D' Angelo and Fortuin, 2021].

Beyond uncertainty penalties, the regularisation for the critic's regression can come from other sources, such as the discrepancy between actions of the learned and behaviour policy [Wu et al., 2019, Tarasov et al., 2024, Kostrikov et al., 2021]. Alternatively, one can attempt to directly learn a conservative value lower bound with a corresponding policy that is robust to the worst-case realisation of the value function given the observed data [Kumar et al., 2020, Lyu et al., 2022, Cheng et al., 2022]. Finally, our work builds on Valdettaro and Faisal [2024], where the source of conservatism naturally

arises from carrying out inference with a pessimistic value prior. By employing a pessimistic prior on value and carrying out Bayesian inference on the observed data, the posterior will only have high expected value in those regions where the data supports it, with the natural consequence of avoiding out-of-distribution (OOD) state-actions without directly considering a worst-case value function.

2.2 Inference in RL

Bayesian inference in off-policy model-free RL presents the unique challenge that there are no ground-truth label, and the value function must instead be inferred by exploiting the Markovian structure of the environment and piecing together the information that individual observed transitions provide. Previous work that carries out Bayesian inference in value-function space includes GP-SARSA [Engel et al., 2005] which is online, on-policy and episodic and therefore must be adapted for use in offline RL. Techniques that have proven successful in supervised learning, such as ensembles, [Lakshminarayanan et al., 2017, Ovadia et al., 2019] have been ported over to RL for the purposes of uncertainty quantification [Osband et al., 2018] but it is unclear whether the quality of the uncertainty quantification is comparable in the RL case. Part of the additional difficulty comes from the lack of a ground-truth target, and what constitutes principled or consistent uncertainty quantification for model-free RL with temporal difference (TD) learning has been topic of debate [Osband et al., 2018, Touati et al., 2020].

2.3 Inference in function space

We consider Bayesian inference in function space rather than parameter space, which comes with a set of challenges. While some nonparametric approaches, such as Gaussian processes (GPs), naturally carry out inference in function space, these require approximations to be scalable to large datasets [Hensman et al., 2013, Titsias, 2009] and rely on hand-crafted features which is undesirable for complex tasks or environments. The field of functional variational inference [Burt et al., 2020, Ma and Hernández-Lobato, 2021] seeks to approximate inference in value-function space while still employing expressive parametric models. We employ the framework for functional variational inference proposed in Hafner et al. [2020], where the prior knowledge is injected into training by sampling pseudo-data points from the prior distribution.

3 Methods

We introduce here algorithms that make use of pessimistic value priors (PVPs) and off-policy evaluation with inference in value-function space to carry out offline RL. We present both an algorithm that relies on exact inference, suitable for simple environments and small datasets, as well as a scalable implementation of offline RL with PVPs and deep neural networks. Throughout this section we denote the state and action spaces of a Markov Decision Process (MDP) with $\mathcal{S}$ and $\mathcal{A}$ respectively, its discount factor with γ and unknown (but deterministic) transition P and reward R functions that depend on state-action. We consider a static, non-episodic dataset of transitions $(s_i, a_i, r_i, s'_i) \in \mathcal{D}$ from which we wish to learn a policy π that maximises cumulative discounted rewards.

3.1 Exact inference

One conceptual challenge that arises from doing probabilistic inference in off-policy RL is that, unlike supervised learning, the value at any state isn't directly observed. Rather, the value function has to be inferred through the effects that carrying out a policy over a number of timesteps. Thus, the observation from individual transitions must be aggregated with the knowledge that the environment is Markovian to carry out consistent inference. Usually, this is accounted for by employing iterative policy evaluation schemes off of bootstrapped approximations. However, since the bootstrapped labels are not ground-truth, employing usual inference techniques on these is problematic. We instead directly include the recursive Bellman equation into the inference process.

The algorithm we present here is a GP implementation that adapts the work in Valdettaro and Faisal [2024], which itself provides an off-policy adaptation of the line of work in Engel et al. [2005], to carry out TD off-policy inference in value-function space for continuous state and action spaces. For a deterministic policy π and a deterministic environment, the action-values at any s, a satisfy the

Bellman equation

$$Q(s,a) = R(s,a) + \gamma Q(s', \pi(s')) \tag{1}$$

where s' is the state deterministically reached from state s after taking action a. Hence, the probabilistic model we employ assumes that the observed (reward samples r) and latent (action-values Q) variables relevant to the transitions in the dataset $(s_i, a_i, r_i, s'_i) \in \mathcal{D}$ are related through

$$r_i = Q(s_i, a_i) - \gamma_i Q(s'_i, \pi(s'_i)) + \varepsilon_i, \tag{2}$$

where $\gamma_i = \gamma$ if s'_i is not terminal and 0 otherwise, and ε_i a small zero-mean independent Gaussian noise term. Next, we set a Gaussian (pessimistic) prior on Q. The prior covariance on Q is given by some hand-crafted kernel function that factorises across state and action spaces $k_Q((s_1, a_1), (s_2, a_2)) = k_s(s_1, s_2)k_a(a_1, a_2)$.

Since each r_i is itself a linear combination of Gaussian random variables, we can establish their prior mean (which is 0) and the covariance between any two distinct observed rewards:

$$\text{cov}(r_i, r_j) = k(x_i, x_j) - \gamma_j k(x_i, x'_j) - \gamma_i k(x'_i, x_j) + \gamma_i \gamma_j k(x'_i, x'_j), \tag{3}$$

where we used the shorthand notation $x_k = (s_k, a_k)$ and $x'_k = (s'_k, \pi(s'_k))$. We can also compute the prior covariance between the observed rewards and the action-value of any arbitrary state-action:

$$\text{cov}(Q(s,a), r_i) = k((s,a), x_i) - \gamma_i k((s,a), x'_i). \tag{4}$$

We can now find the posterior value distribution (assuming zero prior mean) for any state-action s, a by using the Gaussian conditioning formulas for posterior mean μ and covariance $\mathbf{\Sigma}$ [Rasmussen et al., 2006]:

$$\mu^* = \mathbf{K}_{QR}^\top (\mathbf{K}_R + \sigma_r^2 \mathbf{I})^{-1} \mathbf{r} \tag{5}$$

$$\mathbf{\Sigma}^* = \mathbf{K}_Q - \mathbf{K}_{QR}^\top (\mathbf{K}_R + \sigma_r^2 \mathbf{I})^{-1} \mathbf{K}_{QR}, \tag{6}$$

where we can populate $\mathbf{K}_R, \mathbf{K}_{QR}$ and $\mathbf{K}_Q$ with the entries $\text{cov}(r_i, r_j), \text{cov}(Q(s,a), r_i)$ and $k((s,a),(s,a))$ respectively. Notice that the action-value posterior's dependence on policy is present through the terms containing $x' = (s', \pi(s'))$.

Finally, we can carry out policy improvement by using the posterior mean as the critic's regularised evaluation, and summarise the algorithm in Alg.1. As is common in RL, when implementing the algorithm we do not pass gradients through the terms used to evaluate action-values at the next-states $Q(s', \pi(s'))$ in the actor's parameter optimisation step. In principle, the posterior variance could also be used for action selection.

Algorithm 1 Pessimistic value priors with exact inference

Require: Dataset $\mathcal{D}$, Gaussian prior on Q, parametric actor $\pi_\theta : \mathcal{S} \to \mathcal{A}$, learning rate η
  while not converged **do**
    $\hat{Q} \leftarrow \mathbb{E}\left(p(Q|\mathcal{D}, \pi_\theta)\right)$ ▷ Evaluate $p(Q|\mathcal{D}, \pi_\theta)$ using Eq. 5
    $J(\theta) \leftarrow \sum_{s \in \mathcal{D}} \hat{Q}(s, \pi_\theta(s))$
    $\theta \leftarrow \theta + \eta \nabla_\theta J(\theta)$
  end while

3.2 Scalable inference

While exact Bayesian inference is suitable for small datasets and simple environments where a hand-crafted kernel is easily specified, sequential decision making from larger datasets on complex environments requires a more scalable approach. One option is to approximate exact inference with variational inference [Titsias, 2009], but this still leaves the difficult task of finding adequate prior kernels, and methods that try to learn these [Wilson et al., 2016, Ober et al., 2021, van Amersfoort et al., 2021] require introducing a highly non-trivial layer of algorithmic complexity on top of the already challenging task of offline RL. Therefore, we opt to instead build on the approach presented in Hafner et al. [2020], where the inductive bias of inference in function space with a pessimistic prior is induced by sampling pseudo-data points from the prior. We choose the simplest implementation that is able to recover an approximate maximum a-posteriori (MAP) mean. The loss function proposed by

Hafner et al. [2020] adapted to the RL context learns a MAP mean $\hat{Q}$ critic trained with deep neural network parameters θ with loss

$$L(\theta) = L_B(\theta) + \alpha_0 \mathbb{E}_{(s,a)\sim p_{\text{prior}}(s,a)}(D_{\text{KL}}\left(p(Q(s,a))||p(Q(s,a)|\theta)\right)) \tag{7}$$

$$= L_B(\theta) + \alpha \mathbb{E}_{(s,a)\sim p_{\text{prior}}(s,a)}(\hat{Q}(s,a|\theta) - \mu_Q)^2, \tag{8}$$

where L_B is the negative log-likelihood loss (here corresponding to the standard Bellman TD critic loss), μ_Q is the pessimistic prior mean on Q and we have used the property that the KL between two Gaussians is proportional to the squared distance of their means, absorbing the constant of proportionality into an overall hyperparameter α. p_{prior} is the prior pseudo-data distribution used to regularise the learned value function. While the exact inference approach implicitly carries out regularisation everywhere in state-action space, this is not feasible with the pseudo-data approach, so one must choose p_{prior} so that it appropriately regularises the most salient state-actions relevant towards decision making. For a dataset state distribution ρ, a natural pseudo-data distribution for a stochastic policy would be $\rho(s)\pi(a|s)$ whereas for a deterministic policy we can sample so as to ensure regularisation around the learned policy's boundary

$$p_{\text{prior}}(s,a) = \rho(s)\mathcal{N}(a|\pi_\theta(s), \sigma_a^2), \tag{9}$$

with similar intuition that this prioritises regularising the most relevant actions, which are those closest to the ones considered by the learned policy.

Having established an approach to include the pessimistic prior's effect in the critic's evaluation, we can adapt any off-policy online algorithm by adding the new term in Eq. 8 to obtained a suitably regularised offline RL critic. We summarise how the critic update changes for any base online off-policy algorithm in Algorithm 2.

Algorithm 2 Scalable PVP critic update loss

Require: Dataset $\mathcal{D}$, online off-policy algorithm with deterministic or stochastic actor π, critic Q_θ and critic loss $L_B(\theta)$, pessimistic prior mean μ_Q, prior regularisation strength α, pseudo-data number of samples n and noise σ_a.
Sample batch of transitions $\mathcal{B}$ from $\mathcal{D}$
for $(s,a,r,s') \in \mathcal{B}$ **do**

Sample n actions, $a_p^{(s)} \leftarrow \begin{cases} a_p^{(s)} = \pi(s) + \sigma_r\varepsilon, \varepsilon \sim \mathcal{N}(\cdot|0,\sigma_a^2\mathbf{I}_n) & \text{if } \pi \text{ deterministic} \\ a_p^{(s)} \sim \pi(\cdot|s) & \text{if } \pi \text{ stochastic} \end{cases}$

end for

$$L \leftarrow L_B(\theta) + \alpha \frac{1}{|\mathcal{B}|n} \sum_{s\in\mathcal{B}} \sum_{a_p^{(s)}} (Q_\theta(s, a_p^{(s)}) - \mu_Q)^2$$

return loss L

4 Results

We present here results that illustrate the behaviour of our form of critic regularisation on a toy experiment as well as preliminary results on the halfcheetah-medium D4RL benchmark task.

4.1 Toy experiment

With the toy experiment we consider here we seek to highlight three key principled behaviours of our algorithm. First, it is able to stitch information from different transitions in a dataset to obtain policies that lead to in-distribution trajectories with high rewards. Secondly, it correctly assigns low values to policies that rely on out-of-distribution state-actions. Thirdly, it leads to consistent Bayesian off-policy value uncertainty quantification from a non-episodic dataset.

We consider here an MDP corresponding to a continuous-space maze-like environment where only a portion of its state-action space is adequately covered by the dataset. The state-space we consider is $\mathcal{S} = \mathbb{R}^2$ and the action space is $\mathcal{A} = [-1,1]^2$, visualised in Fig. 1. At each step, the agent

deterministically moves in state space by adding the action vector to its current state vector. The agent observes a reward of 1 if it reaches a particular goal region $\{(x, y) \in \mathbb{R}^2 : 2 < x < 3, 0 < y < 1\}$ and 0 otherwise. Episodes terminate on the transition that reaches the goal state.

The static dataset we consider covers a limited part of the environment's state-space, and is formed by individual (s, a, r, s') transitions rather than full episodic traces. The datasets contains transitions starting from 100 uniformly spaced states in the region $\{(x, y) \in \mathbb{R}^2 : 2 < x < 3, 0 < y < 1)\} \cup \{(x, y) \in \mathbb{R}^2 : 0 < x < 3, 1 < y < 2)\}$. All actions in the dataset are in either the x or y directions and have size 0.6, with the action observed at each state being the one that makes progress towards the goal while remaining in the in/support region (see Fig. 1a).

We first apply Algorithm 1 to this toy dataset, with RBF kernels with length-scale 0.25 for both state and actions and an actor parametrised by an MLP with two hidden layers of 256 neurons trained for 1000 steps. In Fig 1b, we observe that the resulting policy reaches the goal while remaining in the in-distribution data regions. The values learned are displayed in Fig. 1c, where we observe that only the in-distribution regions are assigned high values, which gradually decrease with number of steps required to reach the goal (as expected due to the discount factor $\gamma = 0.9$). Similarly, the value uncertainty displayed in Fig. 1d is high in the out-of-distribution regions and low where the agent can confidently reach the goal while remaining in-distribution.

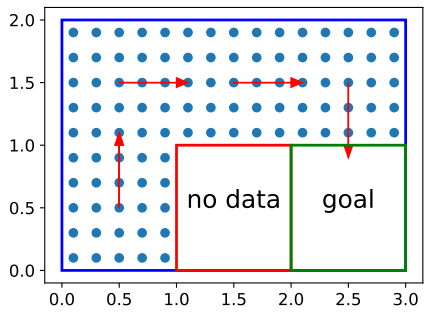
We also apply scalable versions as described in Section 3.2 and Algorithm 2, with two base online algorithms, one learning a deterministic (Twin Delayed DDPG [Fujimoto et al., 2018]), and one a stochastic (Soft Actor-Critic [Haarnoja et al., 2018a,b]) policy, with offline PVP-based algorithms called PVP-TD3 and PVP-SAC respectively. As to be consistent with the GP implementation, we use prior mean $\mu_Q = 0$ and $\alpha = 0.001$ for both methods. We implement our algorithm by modifying the base implementations provided by Tarasov et al. [2022].

Visualisations of the learned policies and values from applying Algorithm 2 are shown in Figs. 1e and 1f. We observe that the scalable version of the algorithms is also able to produce a policy that avoid the no-data region and reach the goal and a value estimate that is roughly consistent with the exact-inference case in the in-distribution region. We note that there is no training signal to regularise the out-of-distribution states for PVP-TD3 or PVP-SAC, as the pseudo-data distribution is chosen to regularise OOD actions rather than OOD states (so OOD states don't explicitly receive a signal to have value close to the prior). In contrast, Figs. 1g and 1h show that applying the base off-policy online methods naively to the offline data (equivalent to setting $\alpha = 0$) leads to poor policies that do not remain in the support of the data and wander into regions of state-space that the agent has no knowledge of, leading to unsatisfactory offline policies.

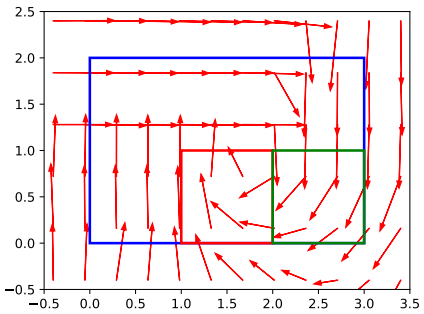
Finally, we observe the pessimistic prior directly in action in Fig. 2, where we visualise the action-values after training in the *action* space at state $(2.5, 1.1)$ just above the goal region. There is an action $(0, -0.6)$ in the dataset at this state that leads directly to the goal, which we expect the agent to follow. We observe that for all three methods, the neighbourhood around the observed action that leads to the goal is regularised as desired, and the algorithms correctly learn to choose the action similar to the one in the dataset that leads to the goal. As can be seen in Fig. 2a, due to the exact nature of the inference employed, the GPs smoothly regularise the whole action space around the observed action. Similarly, the actions that maximise value in Figs. 2b and 2c correctly correspond to following the observed action leading to the goal. While the extrapolation in regions far from the action the agent is considering can vary significantly across methods, this is because our approximate methods must focus on regularising those actions most towards the actor's decision making, and we do observe that values learned close to the actor's action are similar across methods, therefore ensuring that the appropriate action is learned. In contrast, Fig. 2d shows how a naive application of TD3 (setting $\alpha = 0$ in TD3-PVP) causes the actor to choose an unsupported action, where the increase in value is not supported by observed actions but rather entirely caused by unwarranted extrapolation.

4.2 D4RL

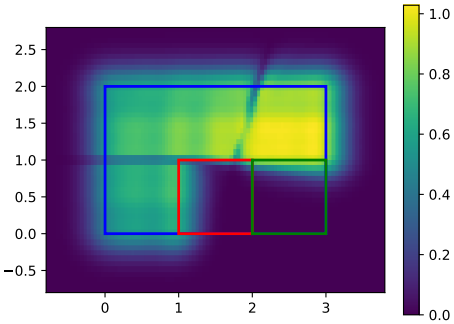
We present here preliminary results of the performance of the scalable version of PVPs as described in Section 3.2 on the D4RL halfcheetah-medium benchmark dataset, containing 10^6 transitions. We preprocessed the rewards by translating them so that the smallest reward in the dataset is 0, once again making a prior of $\mu_Q = 0$ a pessimistic prior. For both methods, we use a default number ($n = 10$) of



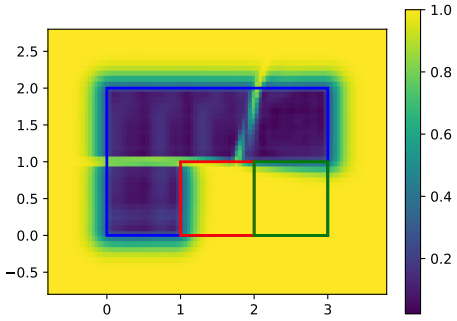
(a) Dataset visualisation



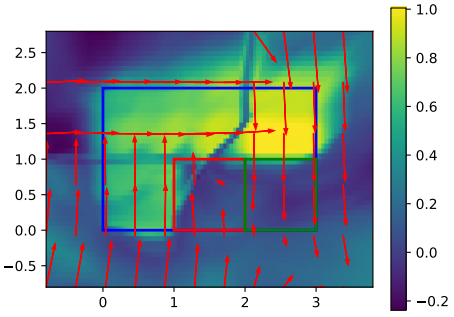
(b) Learned policy



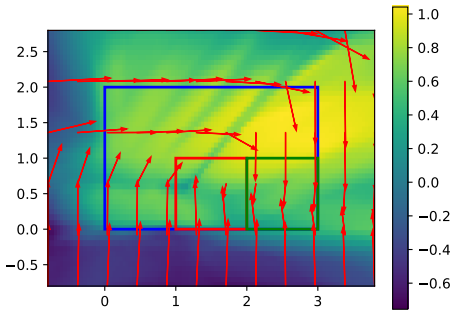
(c) Posterior expected value



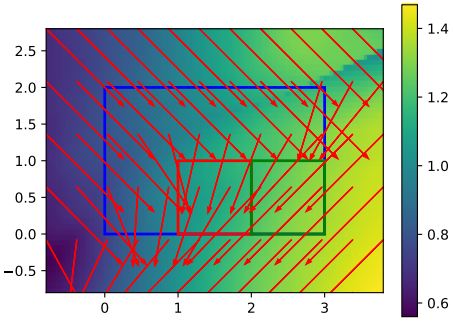
(d) Posterior value standard deviation



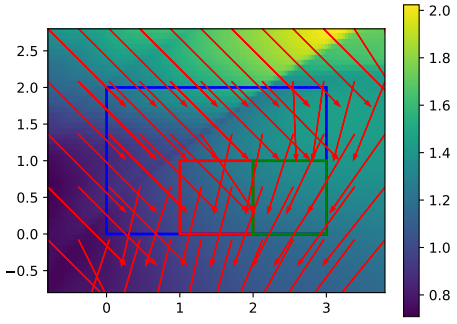
(e) PVP-TD3 learned policy and MAP value



(f) PVP-SAC learned policy and MAP value

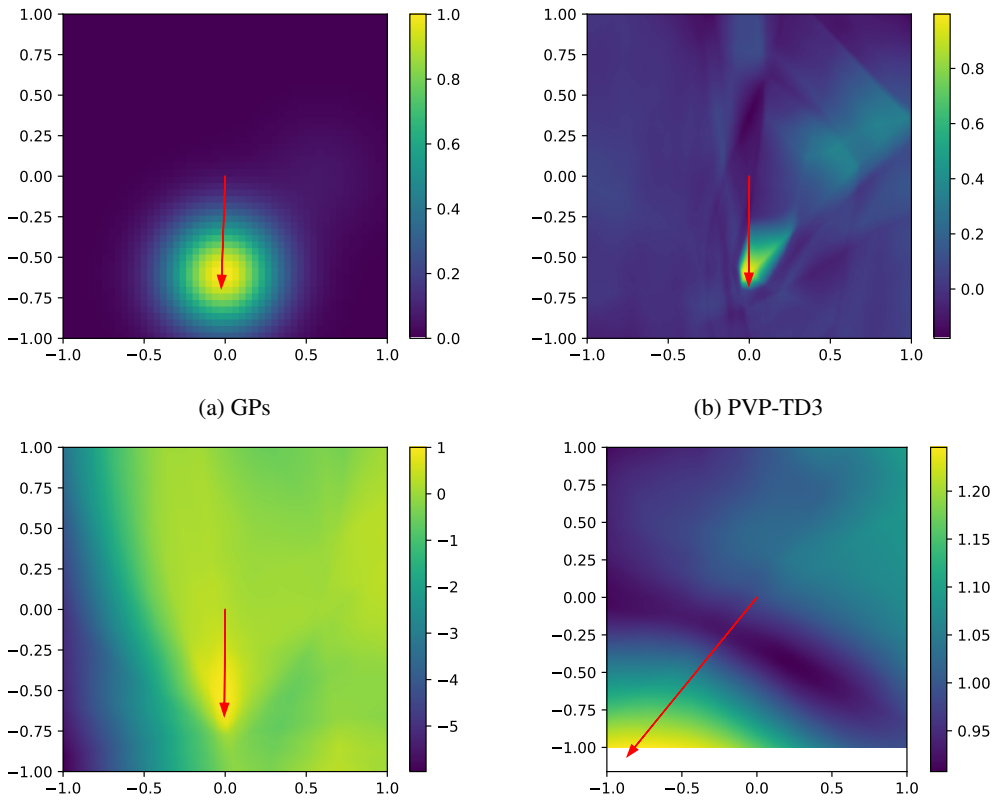


(g) Naive TD3 learned policy and MAP value



(h) Naive SAC learned policy and MAP value

Figure 1: Toy environment for navigation task. The agent receives a reward of 1 for transitioning into the goal region (where the episode terminates) and 0 otherwise. The dataset consists of steps of size 0.6 in the cardinal direction that leads towards the goal while remaining in the supported (blue) region. Learned policies and value functions are visualised as arrows and colour-maps respectively.



(a) GPs

(b) PVP-TD3

(c) PVP-SAC

(d) Naive TD3

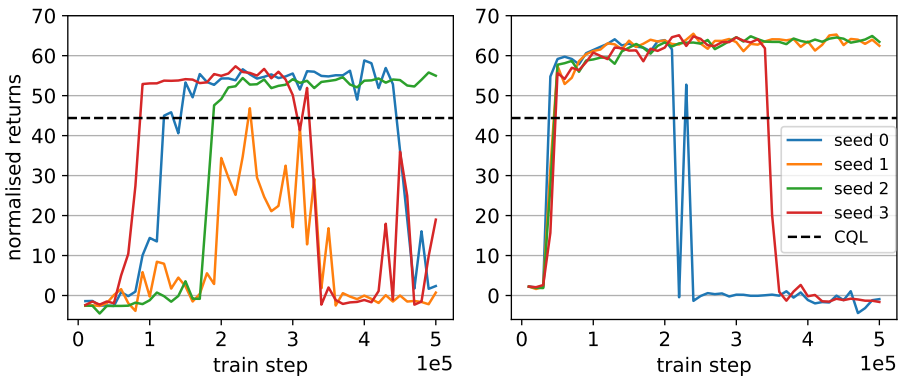
Figure 2: Learned action-values at state $(2.5, 1.1)$, located just above the goal. The red arrow shows the learned action. The dataset contains an action $(0, -0.6)$ leading to the goal at this state.

sampled actions at each state for regularisation. Since PVP-TD3 learns a deterministic policy, we also need a further hyperparameter that determines the variance of the actions at which to sample prior pseudo-data, which we chose by default to be $\sigma_a = 0.2$. The only change made to the base algorithms was to reduce the policy noise in TD3 to 0.01. The rest of the architectures and hyperparameters of the base SAC and TD3 algorithms are unchanged from their adapted implementations in CQL and TD3-BC respectively. α values were found by sweeping through different orders of magnitude, and values of 0.001 and 0.0001 were chosen for PVP-TD3 and PVP-SAC respectively.

Fig. 3 shows the learning curves over 500k training steps corresponding to four random seeds on the halfcheetah-medium dataset, where the normalised score is obtained by rolling out the learned policy every 10k training steps. We compare the runs to the average returns obtained by CQL [Kumar et al., 2020] on this task, which is a relevant baseline that attempts to learn a conservative value estimate without requiring a generative model for the behaviour policy as done in Lyu et al. [2022]. In fact, although derived from significantly different theoretical starting points, CQL and PVP-SAC only differ by the regularisation terms they employ on top of the same online base algorithm (SAC), so comparison between the two directly compares the effect of the different critic regularisation terms. We observe that our methods have the potential to be able to converge to significantly higher-performing policies for substantial portions of training, but can suffer from instabilities that cause their performance to crash with additional training.

5 Conclusion

We have developed the framework of offline RL via pessimistic value priors [Valdettaro and Faisal, 2024] as a novel form of critic regularisation for continuous control. We presented a method with exact inference applicable to deterministic MDPs to accomplish this. We showed on a toy dataset



(a) PVP-TD3 (b) PVP-SAC

Figure 3: Learning curves of PVP-TD3 and PVP-SAC on halfcheetah-medium for four randomly initialised runs. The black dashed line corresponds to the average performance of CQL after training as reported in Kumar et al. [2020]. The normalised returns are obtained by rolling out the policy learned offline in the environment for 10 episodes every 10k training steps and averaging the obtained returns.

how Bayesian inference in value-function space with a pessimistic value prior can enable offline RL by stitching information in different transitions avoiding unsupported policies. Next, we used the same toy environment to verify that the core desirable behaviours of this framework are also realised in our scalable implementation with deep neural networks. We incorporated the pessimistic prior in a scalable way by employing the pseudo-data approach in Hafner et al. [2020]. This amounted to adding a new term to the critic loss, applicable to any online off-policy algorithm, and confirmed it lead to similar desirable behaviour, with good policies and consistent value posteriors on the same toy experiment. We were able to scale this approach to a complex robotic environment with a large (10^6 transitions) dataset and benchmark the performance of our method against one previous approach that carries out offline RL through critic regularisation. While the preliminary results are promising, further experimental work must go into enhancing the training stability of our methods, as this is it's current main limitation. Overall, we believe that having a probabilistic foundation for model-free policy evaluation can lead to principled approaches towards value estimation and uncertainty quantification for offline RL, with the potential for new successful offline RL algorithms.

References

Gaon An, Seungyong Moon, Jang-Hyun Kim, and Hyun Oh Song. Uncertainty-based offline reinforcement learning with diversified q-ensemble. *Advances in Neural Information Processing Systems*, 34:7436–7447, 2021.

David R Burt, Sebastian W Ober, Adrià Garriga-Alonso, and Mark van der Wilk. Understanding variational inference in function-space. *arXiv preprint arXiv:2011.09421*, 2020.

Ching-An Cheng, Tengyang Xie, Nan Jiang, and Alekh Agarwal. Adversarially trained actor critic for offline reinforcement learning. In *International Conference on Machine Learning*, pages 3852–3878. PMLR, 2022.

Sudeep Dasari, Frederik Ebert, Stephen Tian, Suraj Nair, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, Sergey Levine, and Chelsea Finn. Robonet: Large-scale multi-robot learning. In *Conference on Robot Learning*, pages 885–897. PMLR, 2020.

Francesco D' Angelo and Vincent Fortuin. Repulsive deep ensembles are Bayesian. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 3451–3465. Curran Associates, Inc., 2021.

Yaakov Engel, Shie Mannor, and Ron Meir. Reinforcement learning with Gaussian processes. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 201–208, 2005.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

Scott Fujimoto and Shixiang Shane Gu. A minimalist approach to offline reinforcement learning. *Advances in Neural Information Processing Systems*, 34:20132–20145, 2021.

Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International Conference on Machine Learning*, pages 1587–1596. PMLR, 2018.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International Conference on Machine Learning*, pages 2052–2062. PMLR, 2019.

Kamyar Ghasemipour, Shixiang Shane Gu, and Ofir Nachum. Why so pessimistic? Estimating uncertainties for offline rl through ensembles, and why their independence matters. *Advances in Neural Information Processing Systems*, 35:18267–18281, 2022.

Omer Gottesman, Fredrik Johansson, Matthieu Komorowski, Aldo Faisal, David Sontag, Finale Doshi-Velez, and Leo Anthony Celi. Guidelines for reinforcement learning in healthcare. *Nature Medicine*, 25(1):16–18, 2019.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning*, pages 1861–1870. PMLR, 2018a.

Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018b.

Danijar Hafner, Dustin Tran, Timothy Lillicrap, Alex Irpan, and James Davidson. Noise contrastive priors for functional uncertainty. In *Uncertainty in Artificial Intelligence*, pages 905–914. PMLR, 2020.

James Hensman, Nicolo Fusi, and Neil D Lawrence. Gaussian processes for big data. *arXiv preprint arXiv:1309.6835*, 2013.

Zhiyu Huang, Jingda Wu, and Chen Lv. Efficient deep reinforcement learning with imitative expert priors for autonomous driving. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on Robot Learning*, pages 651–673. PMLR, 2018.

Alex Kendall, Jeffrey Hawke, David Janz, Przemyslaw Mazur, Daniele Reda, John-Mark Allen, Vinh-Dieu Lam, Alex Bewley, and Amar Shah. Learning to drive in a day. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8248–8254. IEEE, 2019.

Matthieu Komorowski, Leo A Celi, Omar Badawi, Anthony C Gordon, and A Aldo Faisal. The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. *Nature Medicine*, 24(11):1716–1720, 2018.

Ilya Kostrikov, Rob Fergus, Jonathan Tompson, and Ofir Nachum. Offline reinforcement learning with fisher divergence critic regularization. In *International Conference on Machine Learning*, pages 5774–5783. PMLR, 2021.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 33:1179–1191, 2020.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems*, 30, 2017.

Jiafei Lyu, Xiaoteng Ma, Xiu Li, and Zongqing Lu. Mildly conservative q-learning for offline reinforcement learning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=VYYf6S67pQc.

Chao Ma and José Miguel Hernández-Lobato. Functional variational inference based on stochastic process generators. *Advances in Neural Information Processing Systems*, 34:21795–21807, 2021.

Ashvin Nair, Abhishek Gupta, Murtaza Dalal, and Sergey Levine. Awac: Accelerating online reinforcement learning with offline datasets. *arXiv preprint arXiv:2006.09359*, 2020.

Sebastian W Ober, Carl E Rasmussen, and Mark van der Wilk. The promises and pitfalls of deep kernel learning. In *Uncertainty in Artificial Intelligence*, pages 1206–1216. PMLR, 2021.

Ian Osband, John Aslanides, and Albin Cassirer. Randomized prior functions for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in Neural Information Processing Systems*, 32, 2019.

Carl Edward Rasmussen, Christopher KI Williams, et al. *Gaussian processes for machine learning*, volume 1. Springer, 2006.

Samarth Sinha, Ajay Mandlekar, and Animesh Garg. S4rl: Surprisingly simple self-supervision for offline reinforcement learning in robotics. In *Conference on Robot Learning*, pages 907–917. PMLR, 2022.

Denis Tarasov, Alexander Nikulin, Dmitry Akimov, Vladislav Kurenkov, and Sergey Kolesnikov. CORL: Research-oriented deep offline reinforcement learning library. In *3rd Offline RL Workshop: Offline RL as a "Launchpad"*, 2022. URL https://openreview.net/forum?id=SyAS49bBcv.

Denis Tarasov, Vladislav Kurenkov, Alexander Nikulin, and Sergey Kolesnikov. Revisiting the minimalist approach to offline reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Michalis Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *Artificial Intelligence and Statistics*, pages 567–574. PMLR, 2009.

Ahmed Touati, Harsh Satija, Joshua Romoff, Joelle Pineau, and Pascal Vincent. Randomized value functions via multiplicative normalizing flows. In *Uncertainty in Artificial Intelligence*, pages 422–432. PMLR, 2020.

Filippo Valdettaro and A. Aldo Faisal. Towards offline reinforcement learning with pessimistic value priors. In Fabio Cuzzolin and Maryam Sultana, editors, *Epistemic Uncertainty in Artificial Intelligence*, pages 89–100, Cham, 2024. Springer Nature Switzerland. ISBN 978-3-031-57963-9.

Joost van Amersfoort, Lewis Smith, Andrew Jesson, Oscar Key, and Yarin Gal. On feature collapse and deep kernel learning for single forward pass uncertainty. *arXiv preprint arXiv:2102.11409*, 2021.

Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P Xing. Deep kernel learning. In *Artificial Intelligence and Statistics*, pages 370–378. PMLR, 2016.

Yifan Wu, George Tucker, and Ofir Nachum. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361*, 2019.

Teng Xiao and Donglin Wang. A general offline reinforcement learning framework for interactive recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4512–4520, 2021.

Wenxuan Zhou, Sujay Bajracharya, and David Held. Plas: Latent action space for offline reinforcement learning. In *Conference on Robot Learning*, pages 1719–1735. PMLR, 2021.