Supplementary Material

This document accompanies the ALA'24 submission "Bayesian Ensembles for Exploration in Deep Q-learning".

1 Supervised Learning Experiment

This experiment tests posterior approximation methods in an ideal setting. The data is exactly drawn from the assumed likelihood and prior. The neural network is a fully connected network with two hidden layers of size 100 and 10. The data is generated by sampling x i.i.d. from a uniform mixture of three normal distributions with means -1, 0, and 1 and standard deviation 0.25. The labels y are generated by randomly sampling a neural network $\theta_{\texttt{prior}}$ from the prior $p(\theta)$ and setting

$$y_i = f_{\theta_{\texttt{prior}}}(x_i) + \epsilon_i,$$

where $\epsilon_i \sim \mathcal{N}(0, 0.25)$.

The hyperparameters for SMC are shown in Table 1. For randomized prior functions, the prior function was drawn from the same prior as used for SMC and HMC.

2 Reinforcement Learning Experiments

For every agent, the Q-value network is a fully connected neural network with two layers of size 50. For Deep Sea, the observation is flattened before the first layer. The hidden units have leaky ReLU activations. Action selection is unchanged and done by picking one ensemble member uniformly at random, and acting greedily with respect to that network for the rest of the episode. Each ensemble member has access to all data collected by other ensemble members. No noise is added to the TD errors. We use an ensemble size of 10 for every agent.

The hyperparameters used for SMC-DQN are shown in Table 2, and the hyperparameters for BootDQN with and without priors are shown in Table 3.

The hyperparameters for BootDQN + prior and BootDQN are the default hyperparameters in BSuite. In addition to the default BSuite hyperparemeters, Figure 1 also shows results for our baseline on Cartpole with prior scales 15 and 30 to incentize more exploration, but we did not observe significantly better results than reported in the main text.

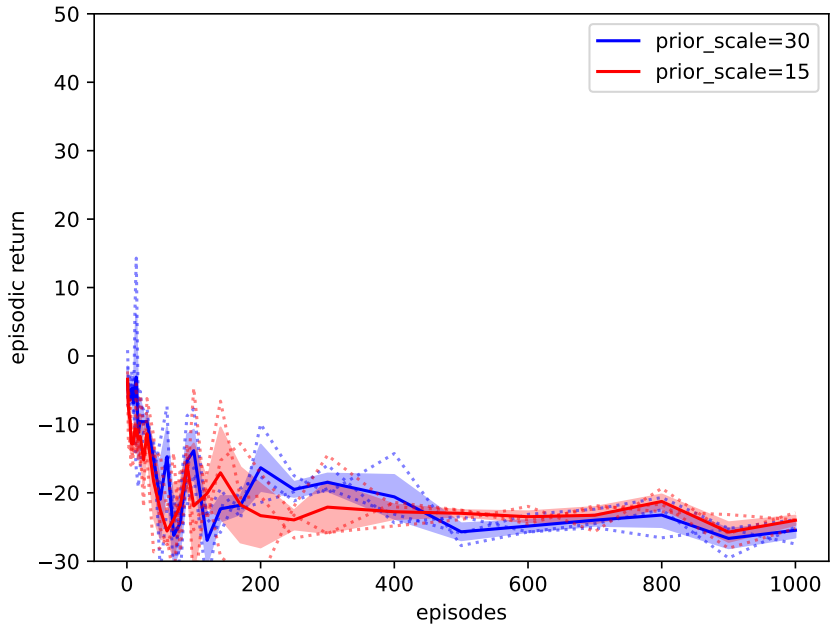

Figure 1: BootDQN with prior functions evaluated on Cartpole for several values of prior scale. The agent fails to find a performing policy.

Sequential Monte Carlo hyperparameters	
$\pi(\theta)$ (prior)	i.i.d. $\mathcal{N}(0,1)$
σ (likelihood std)	0.1
B (batch size)	32
MCMC steps (main)	10
MCMC steps (target)	10
Symplectic Euler Langevin Scheme hyperparameters	
ℓ (learning rate)	10^{-3}
cycle length	20
β	0.98
h (step size)	$\sqrt{\frac{\ell}{\mathtt{n_{data}}}}$
γ	$\frac{1}{h}$

Table 1: Hyperparameters for SMC in the supervised learning experiment.

Sequential Monte Carlo hyper-parameters	
$\pi(\theta)$ (prior)	i.i.d. $\mathcal{N}(0,1)$
σ (likelihood std)	0.1 (1.0 in cartpole)
B (batch size)	128
MCMC steps (main)	100
MCMC steps (target)	100
Target ESS	10%
Reinforcement Learning hyper-parameters	
Buffer size	∞
Main update frequency	B
Target update frequency	B
Symplectic Euler Langevin Scheme hyper-parameters	
ℓ (learning rate)	10^{-3}
cycle length	20
β	0.98
h (step size)	$\sqrt{\frac{\ell}{\mathtt{n_{data}}}}$
γ	$\frac{1}{h}$

Table 2: Hyper-parameters of SMC-DQN in the RL experiments.

BootDQN Hyper-parameters	
Learning rate	0.001
Optimizer	Adam ($\beta_1 = 0.9, \beta_2 = 0.999$)
Update frequency	every step
Target update frequency	every 4 steps
batch size	32
prior scale (if +priors)	5

Table 3: Hyper-parameters of BootDQN(+prior) in the RL experiments.