Bayesian Model-Free Deep Reinforcement Learning

Doctoral Consortium

Pascal R. van der Vaart
Delft University of Technology
Delft, The Netherlands
p.r.vandervaart-1@tudelft.nl

ABSTRACT

Exploration in reinforcement learning remains a difficult challenge. In order to drive exploration, ensembles with randomized prior functions have recently been popularized to quantify uncertainty in the value model. However these ensembles have no theoretical reason to resemble the actual Bayesian posterior, which is known to provide strong performance in theory under certain conditions. In this thesis work, we view training ensembles from the perspective of Sequential Monte Carlo, a Monte Carlo method that approximates a sequence of distributions with a set of particles, and propose an algorithm that exploits both the practical flexibility of ensembles and theory of the Bayesian paradigm. We incorporate this method into a standard DQN agent and experimentally show qualitatively good uncertainty quantification and improved exploration capabilities over a regular ensemble. In the future, we will investigate the impact of likelihood and prior choices in Bayesian model-free reinforcement learning methods.

KEYWORDS

Bayesian, Reinforcement Learning, Exploration, Uncertainty

ACM Reference Format:
Pascal R. van der Vaart. 2024. Bayesian Model-Free Deep Reinforcement Learning: Doctoral Consortium. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 3 pages.

1 INTRODUCTION

Reinforcement learning (RL) algorithms are still notoriously sample inefficient. One pressing reason is the difficulty of exploring an environment efficiently while assuming little prior knowledge. A promising approach that is currently studied is to attempt to quantify epistemic uncertainty of an agent, which is the uncertainty caused specifically by a lack of data, as opposed to uncertainty caused by inherent randomness. If the epistemic uncertainty of the value models learned by the agent can be quantified accurately, then an approach that provides intrinsic reward or uses Thompson sampling [1, 2, 7, 8, 14, 15, 17] can effectively drive exploration. However, quantifying uncertainty for deep neural networks is in itself a difficult task [10, 13].

Ensembles of neural networks have been shown to provide better predictive accuracy over a single model in supervised learning tasks [4, 11], as well as suitable methods for uncertainty quantification for exploration in reinforcement learning [6, 14, 15]. While ensembles with independent models of identical architecture tend to collapse to the same predictive model [9], there are several techniques developed to prevent this, such as adversarial learning [11], bootstrapping the data [15], and adding additive priors [14]. Further, some techniques such as Stein Variational Gradient Descent [3, 12] alleviate this issue by interpreting the ensemble as an approximation to the Bayesian posterior and training it as such.

Bayesian neural networks can have desirable properties if the posterior can be inferred accurately. They have in theory optimal predictive accuracy given the correct likelihood and prior and also provide accurate uncertainty quantification. Many algorithms such as BootDQN [14, 15], NoisyNets [7], Epistemic Value Estimation [18], Monte Carlo Dropout [8] are motivated from a Bayesian point of view, but whether the posterior approximations are close to the true posterior is an important question. Performance of the algorithm does not relate one-to-one to quality of posterior approximation, since in model-free methods it is unlikely that the problem is properly described by the chosen likelihood, which is usually taken to be a normal distribution. For example, BootDQN with prior functions achieves state of the art performance on Deep Sea, a very difficult needle-in-haystack problem, but there is no inherent reason for the ensemble to be similar to the posterior distribution.

Unfortunately, exactly inferring the posterior is intractable already for some simple statistical models, and accurately approximating this posterior is very difficult for neural networks. Typically, posterior approximation methods fall into one of two categories: Markov Chain Monte Carlo, and Variational Inference.

In my thesis work, I aim to answer the following questions:

- Under standard likelihood and prior assumptions, does more accurate posterior approximation produce better model-free algorithms?
- Can we come up with more suitable likelihood and priors in order to make the resulting posterior more aligned to solving typical RL benchmarks?

2 BAYESIAN DEEP LEARNING

A Bayesian Neural Network (BNN) is any neural network f_θ parameterized by $\theta \in \Theta$ where the intent is to infer the posterior

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{\int p(\mathcal{D}|\theta)p(\theta)d\theta} = \frac{\prod_{i=1}^{n} \mathcal{L}(y_i|f_\theta(x_i))p(\theta)}{\int \prod_{i=1}^{n} \mathcal{L}(y_i|f_\theta(x_i))p(\theta)d\theta}, \quad (1)$$

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

where $(x_1, \ldots, x_n)$ and $(y_1, \ldots, y_n)$ are data points assumed to be i.i.d. from some likelihood $\mathcal{L}(y|f_\theta(x))$, and $\pi(\theta)$ is a prior distribution.

Unfortunately, especially in the case of large neural networks, the posterior is intractable to compute or sample from exactly. In reinforcement learning settings, Markov Chain Monte Carlo [5] and Variational Inference [7, 8, 18] have both seen use before, with varying model algorithms and model classes.

Another problem in the case of deep learning, is that the chosen likelihood and prior have to be the correct probability distributions in order for the posterior to enjoy the theoretical guarantees that the Bayesian paradigm provides.

Sequential Monte Carlo (SMC) is a class of algorithms that aim to sample from a sequence of distributions $p_0(\theta), \ldots, p_m(\theta)$, lending itself very well to problems where data comes in sequentially.

For example, a posterior $p(\theta|\mathcal{D}) \propto p(\theta)p(\mathcal{D}|\theta)$ conditioned on one data set $\mathcal{D}$, can be updated to the posterior conditioned on newly obtained data $p(\theta|\mathcal{D} \cup \mathcal{B})$ by applying SMC to the sequence

$$\left(p(\theta)p(\mathcal{D}|\theta)p(\mathcal{B}|\theta)^{\lambda_t}\right)_{t=0,\ldots,T}, \tag{2}$$

where $0 = \lambda_0 < \lambda_1 < \cdots < \lambda_T = 1$.

3 SEQUENTIAL MONTE CARLO DQN

In my first paper [19], I propose to use Sequential Monte Carlo to train an ensemble of Q-networks, to create an agent named SMC-DQN that closely resembles the architecture of BootDQN, but uses a posterior approximation with stronger theoretical motivation.

In SMC-DQN, an agent keeps track of the posterior over parameters of a neural network $Q_\theta(s, a)$ that models the Q-values. The posterior is approximated by keeping a set of particles $\theta_1, \ldots, \theta_n$ and weights $w_1, \ldots, w_n$, that are updated by running SMC on a sequence of distributions interpolating between $p(\theta|\mathcal{D}_N)$ and $p(\theta|\mathcal{D}_{N+B})$, where $\mathcal{D}_N$ and $\mathcal{D}_{N+B}$ is the replay buffer at time N and $N + B$ respectively.

Specifically, we extend a standard DQN agent by replacing its point-wise estimator $Q_\theta(s, a)$ with an ensemble of neural networks $Q_{\theta_1}(s, a), \ldots, Q_{\theta_n}(s, a)$ and weights $w_1, \ldots w_n$ to maintain an approximation of the posterior $p(\theta|\mathcal{D}, \boldsymbol{\theta'})$, conditioned on the current replay buffer

$$\mathcal{D} = ((s_t, a_t, r_t, s_{t+1}))_{t=1\ldots N}$$

and current target parameters

$$\boldsymbol{\theta'} = (\theta'_1, \ldots, \theta'_n).$$

In line with the work by Schmitt et al. [18], a normal distribution

$$Q_\theta(s, a) - r(s, a) - \gamma \max_{a'} Q_{\theta'}(s', a') \sim \mathcal{N}(0, \sigma)$$

is used as a probabilistic interpretation of the squared temporal difference error, and to represent the uncertainty in the targets we define the likelihood to be a mixture distribution

$$\log \mathcal{L}(s, a, r, s'|\theta, \boldsymbol{\theta'}) = \log \sum_{i=1}^{n} \frac{1}{n} \exp\left(\frac{1}{2\sigma^2}[Q_\theta(s, a) - r(s, a) - \gamma \max_{a'} Q_{\theta'_i}(s', a')]^2\right), \tag{3}$$

contrasting BootDQN which shares no target values between ensemble members.

The log posterior distribution is defined as

$$\log p(\theta|\boldsymbol{\theta'}, \mathcal{D}) \propto \log p(\theta) + \log \mathcal{L}(\mathcal{D}|\theta, \boldsymbol{\theta'}), \tag{4}$$

where

$$\log \mathcal{L}(\mathcal{D}|\theta, \boldsymbol{\theta'}) = \sum_{(s,a,r,s') \in \mathcal{D}} \log \mathcal{L}(s, a, r, s'|\theta, \boldsymbol{\theta'}). \tag{5}$$

After collecting a batch of data, the agent updates its model of the posterior by running an SMC sampler on the sequence defined in Equation 2.

In our experiments, the agent uses Thompson Sampling to select their actions at train time, but any algorithm that can make use of a probabilistic model, such as UCB, could also be used.

We test our agent on the exploration environments in BSuite [16], which includes Deep Sea, Stochastic Deep Sea, and Cartpole-swingup. We also include Mountain Car since the states where the agent receives feedback on its performance here is also sparse.

We find strongly improved exploration capabilities over regular ensembles, and results competitive with ensembles using randomized prior functions. Especially on continuous state environments our agent performs well. We theorize that the discrepancy in performance between ensembles with randomized prior functions is due to issues with the likelihood. Since Deep Sea is one hot encoded and a deterministic environment, the likelihood has two issues:

(1) The assumption of normally distributed noise is surely violated, since the environment is deterministic for a deterministic policy.
(2) The parameterization of the neural network causes correlation between states that are in reality independent. This makes the likelihood incorrect, and the posterior with respect to this likelihood therefore does not accurately reflect the real uncertainty.

This experiment highlights the necessity of future work in devising good likelihoods for Bayesian model-free Q-learning approaches. Nonetheless, the agent does learn to solve the environment eventually.

4 FUTURE WORK

In supervised deep learning, it is already known that the posterior often does not lead to optimal performance even with accurate posterior approximation when using standard likelihoods and priors [20]. Similarly, our results tentatively suggest that the performance of a Bayesian model-free algorithm with a simple likelihood and prior depends on how applicable the likelihood is to the environment. To fully exploit the uncertainty quantification that the Bayesian paradigm provides, it is therefore an interesting future direction for my thesis to verify this claim and look into how better likelihoods can be constructed.

Furthermore, incorporating such uncertainty quantification and exploration methods into a wider range of RL algorithms and different return estimators, as opposed to standard one-step DQN algorithms, will also help to establish the applicability our methods.

ACKNOWLEDGMENTS

This work has received funding from the European Union's Horizon 2020 research and innovation programme, under grant agreements 964505 (Epistemic AI) and 952215 (TAILOR).

REFERENCES

[1] Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. 2016. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems*, Vol. 29.

[2] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. 2018. Exploration by random Network Distillation. In *International Conference on Learning Representations*.

[3] Francesco D'Angelo and Vincent Fortuin. 2021. Repulsive Deep Ensembles are Bayesian. In *Advances in Neural Information Processing Systems*, Vol. 34.

[4] Thomas G. Dietterich. 2000. Ensemble Methods in Machine Learning. In *Multiple Classifier Systems*. Springer Berlin Heidelberg, 1–15.

[5] Vikranth Dwaracherla and Benjamin Van Roy. 2021. Langevin DQN. arXiv:2002.07282 [cs.LG]

[6] Matthew Fellows, Kristian Hartikainen, and Shimon Whiteson. 2021. Bayesian Bellman Operators. In *Advances in Neural Information Processing Systems*, Vol. 34.

[7] Meire Fortunato, Mohammad Gheshlaghi Azar, Bilal Piot, Jacob Menick, Ian Osband, Alex Graves, Vlad Mnih, Remi Munos, Demis Hassabis, Olivier Pietquin, Charles Blundell, and Shane Legg. 2019. Noisy Networks for Exploration. arXiv:1706.10295 [cs.LG]

[8] Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a Bayesian Approximation: Representing Model Uncertainty in deep Learning. In *International Conference on Machine Learning*.

[9] Mario Geiger, Arthur Jacot, Stefano Spigler, Franck Gabriel, Levent Sagun, Stéphane d'Ascoli, Giulio Biroli, Clément Hongler, and Matthieu Wyart. 2020. Scaling description of generalization with number of parameters in deep learning. *Journal of Statistical Mechanics: Theory and Experiment* 2020, 2 (2020).

[10] Eyke Hüllermeier and Willem Waegeman. 2021. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning* 110, 3 (2021), 457–506.

[11] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In *Advances in Neural Information Processing Systems*, Vol. 30.

[12] Qiang Liu and Dilin Wang. 2016. Stein Variational Gradient Descent: A General Purpose Bayesian Inference Algorithm. In *Advances in Neural Information Processing Systems*, Vol. 29.

[13] Owen Lockwood and Mei Si. 2022. A Review of Uncertainty for Deep Reinforcement Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.

[14] Ian Osband, John Aslanides, and Albin Cassirer. 2018. Randomized Prior Functions for Deep Reinforcement Learning. In *Advances in Neural Information Processing Systems*, Vol. 31.

[15] Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. 2016. Deep Exploration via Bootstrapped DQN. In *Advances in Neural Information Processing Systems*, Vol. 29.

[16] Ian Osband, Yotam Doron, Matteo Hessel, John Aslanides, Eren Sezener, Andre Saraiva, Katrina McKinney, Tor Lattimore, Csaba Szepesvari, Satinder Singh, Benjamin Van Roy, Richard Sutton, David Silver, and Hado Van Hasselt. 2020. Behaviour Suite for Reinforcement Learning. In *International Conference on Learning Representations*.

[17] Georg Ostrovski, Marc G Bellemare, Aäron Oord, and Rémi Munos. 2017. Count-Based Exploration with Neural Density Models. In *International Conference on Machine Learning*.

[18] Simon Schmitt, John Shawe-Taylor, and Hado van Hasselt. 2023. Exploration via Epistemic Value Estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37.

[19] Pascal Van der Vaart, Neil Yorke-Smith, and Matthijs Spaan. 2024. Bayesian Ensembles for Exploration in Deep Reinforcement Learning. In *Proceedings of the 2024 International Conference on Autonomous Agents and MultiAgent Systems* (Auckland, New Zealand) *(AAMAS '24)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC. To appear.

[20] Florian Wenzel, Kevin Roth, Bastiaan Veeling, Jakub Swiatkowski, Linh Tran, Stephan Mandt, Jasper Snoek, Tim Salimans, Rodolphe Jenatton, and Sebastian Nowozin. 2020. How Good is the Bayes Posterior in Deep Neural Networks Really?. In *International Conference on Machine Learning*.