

Using Ensembles to address Bootstrapping Error in Offline RL

Marco A. Gallo

Supervisor: Dr. Matthia Sabatelli

University of Groningen

29-06-2022

Outline

- 1 Background
- 2 Offline RL is hard
- 3 Possible solution: Ensembles
- 4 Experiments
- 5 Analysis
- 6 References

Reinforcement Learning - A schematic view

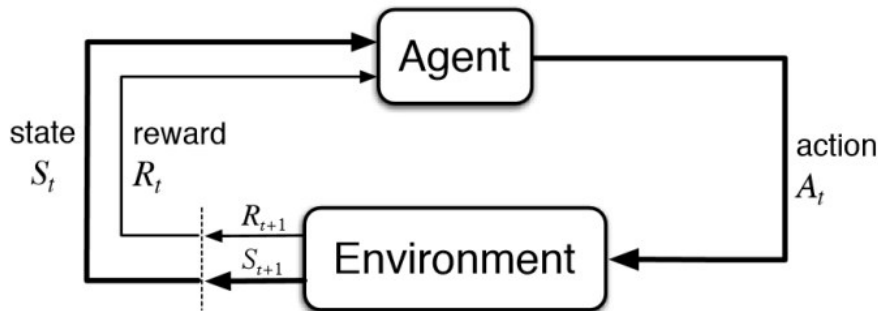


Figure 1: The agent-environment loop (Sutton and Barto, 2018)

Reinforcement Learning Problem Statement

- ▶ An agent seeking an optimal policy $\pi(s, a)$ - a mapping from states to action probabilities ($s \in \mathcal{S}$, $a \in \mathcal{A}$)
- ▶ Used in sequential decision making problems modeled as Markov decision process (*MDP*), enriched with a reward function $R(s, a): \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$
- ▶ Focus: *value-based*, *model-free* methods

RL Definitions

$$G_t = \sum_{k=0}^T \gamma^k r_{t+k} \quad (\text{Discounted cumulative reward})$$

$$Q^\pi(s, a) = \mathbb{E}[G_t | s_t = s, a_t = a, \pi] \quad (\text{State-action value function})$$

$$Q^* = Q^{\pi^*}$$

$$Q^*(s, a) = \mathbb{E} R(s, a) + \gamma \mathbb{E}_{s' \sim P} \max_{a' \in \mathcal{A}} Q^*(s', a') \quad (\text{Bellman optimality equations})$$

Reinforcement Learning (RL) - Offline

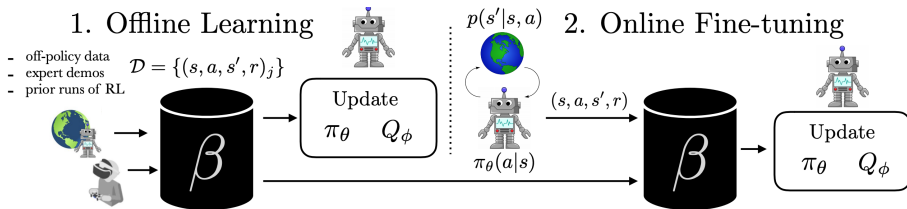


Figure 2: The learning loop in Offline RL, courtesy of Nair et al.

- ▶ Also called Batch Reinforcement Learning
- ▶ Behavior policy π_β generates dataset \mathcal{D}
- ▶ *Pure Batch* vs *Growing Batch* RL methods

Detrimental factors in Offline RL

- ▶ Function approximation errors in Deep RL (Neural Networks)
- ▶ Different state visitation frequencies under training and testing distributions
- ▶ **Bootstrapping error** (Kumar et al., 2019)

Bootstrapping Error

DQN objective function:

$$\mathcal{L}(\theta) = \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1} \rangle \sim D} \left[(r_t + \gamma \max_{a \in \mathcal{A}} Q(s_{t+1}, a; \theta) - Q(s_t, a_t; \theta))^2 \right]$$

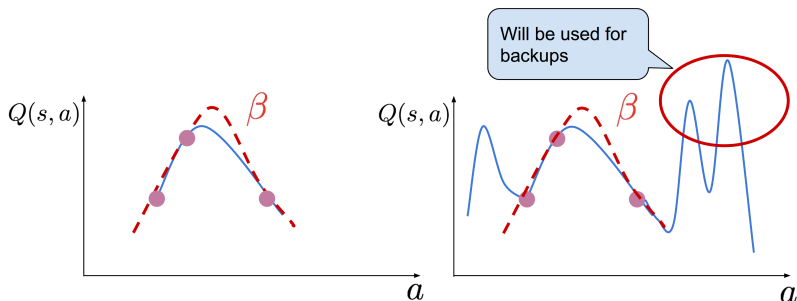


Figure 3: Incorrectly high Q-values for OOD actions may be used for backups, leading to accumulation of error. Figure and caption: Kumar, Aviral

Bootstrapping Error in the DQV⁹ algorithmic family

- ▶ We want to check if the DQV and DQV-Max deep RL algorithms suffer from the Bootstrapping Error in the *offline* setting
- ▶ DQV objective functions:

$$\mathcal{L}(\phi) = \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1} \rangle \sim D} \left[(r_t + \gamma V(s_{t+1}, a; \phi^-) - V(s_t, a; \phi))^2 \right] \quad (1)$$

$$\mathcal{L}(\theta) = \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1} \rangle \sim D} \left[(r_t + \gamma V(s_{t+1}, a; \phi^-) - Q(s_t, a; \theta))^2 \right] \quad (2)$$

- ▶ DQV-Max objective functions:

$$\mathcal{L}(\phi) = \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1} \rangle \sim D} \left[(r_t + \gamma \max_{a \in \mathcal{A}} Q(s_{t+1}, a; \theta^-) - V(s_t, a; \phi))^2 \right] \quad (3)$$

$$\mathcal{L}(\theta) = \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1} \rangle \sim D} \left[(r_t + \gamma V(s_{t+1}, a; \phi) - Q(s_t, a; \theta))^2 \right] \quad (4)$$

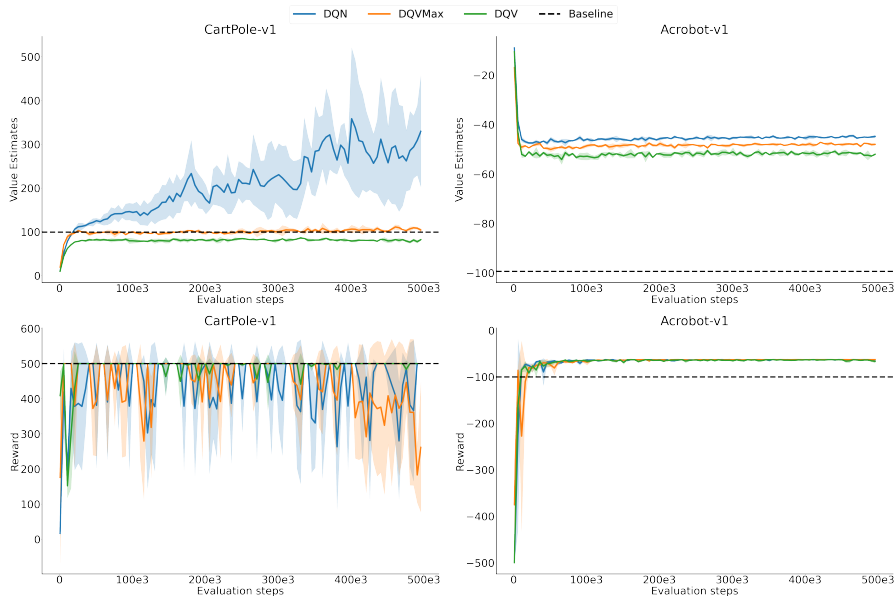
Experimental setup

- ▶ Classic control OpenAI Gym environments: CartPole-v1 and Acrobot-v1
- ▶ Data collection: log every trajectory $\langle s, a, r, s' \rangle$ of a DQN⁷ agent trained online for 500k steps
- ▶ Hyper-parameters and training scheme follow those of the Dopamine³ framework
- ▶ Record estimates of $\max_{a \in \mathcal{A}} Q(s_{t_0}, a)$ at each evaluation round to track the value estimates evolution, then compare against ground truth

$$G_{t_0} = \sum_{k=0}^T \gamma^k r_{t+k}$$

T is the environment's finite time horizon, and r_t is constant across environments

Bootstrapping Error in the DQV algorithmic family - Results



Preventing the Bootstrapping Error - Online

Two ways of addressing the Bootstrapping Error:

1. Obtain unbiased Q-values by decoupling *selection* and *evaluation*, e.g.

- ▶ **Double Q-Learning target**¹¹

$$Q^*(s, a) = r + \gamma Q(s', \operatorname{argmax}_{a \in \mathcal{A}} Q'(s', a))$$

- ▶ **DQV-Max targets** in Eq.(3)

2. Reducing the variance of the Target Approximation Error (TAE)²

- ▶ TAE: $Z_{s,a} = Q(s, a) - \mathbb{E}[r + \gamma \max_{a \in \mathcal{A}} Q(s', a) | s, a]$
- ▶ Anschel et al. show that the magnitude of the bootstrapping bias in Q-learning is related to the *variance* of the TAE

Preventing the Bootstrapping Error - Offline

- ▶ In the offline setting, algorithms such as BCQ⁴ and BEAR⁵ mitigate the Bootstrapping Error by *regularizing* the learned policy to be *close* to the *training trajectories*
- ▶ One exception: Random Ensemble Mixture (REM)¹
 - ▶ Dataset **size** and **diversity** are crucial for offline performance: DQN Replay Dataset on the Atari 2600 benchmark
 - ▶ REM idea: combining multiple noisy Q-functions creates a more robust Q-function

DQV and DQV-Max still incur in the Bootstrapping Error, but...

- ▶ Being an *on-policy* algorithm, DQV is less prone to it
- ▶ DQV-Max is *off-policy*, yet it uses multiple estimators to compute the expected Q-values → also more robust to the Bootstrapping Error
- ▶ **Idea**: can we use techniques for TAE reduction to improve resilience to the Bootstrapping Error in the DQV algorithmic family?
- ▶ Ensemble DQN²: training K Q-functions in parallel to obtain a $\frac{1}{K}$ variance reduction in Q-values
- ▶ Also motivated by REM's strong offline performance

Ensemble learning problem

- ▶ Ensemble DQN learning goal:

$$\mathcal{L}(\theta) = \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1} \rangle \sim D} \left[(r_t + \gamma \max_{a \in \mathcal{A}} Q(s_{t+1}, a; \theta_k^-) - Q(s_t, a_t; \theta_k))^2 \right] \quad (5)$$

- ▶ The learning goal for DQV becomes:

$$\mathcal{L}(\phi) = \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1} \rangle \sim D} \left[(r_t + \gamma V(s_{t+1}, a; \phi_k^-) - V(s_t, a; \phi_k))^2 \right] \quad (6)$$

$$\mathcal{L}(\theta) = \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1} \rangle \sim D} \left[(r_t + \gamma V(s_{t+1}, a; \phi_k^-) - Q(s_t, a_t; \theta))^2 \right] \quad (7)$$

- ▶ The learning goal for DQV-Max becomes:

$$\mathcal{L}(\phi) = \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1} \rangle \sim D} \left[(r_t + \gamma \max_{a \in \mathcal{A}} Q(s_{t+1}, a; \theta_k^-) - V(s_t, a; \phi_k))^2 \right] \quad (8)$$

$$\mathcal{L}(\theta) = \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1} \rangle \sim D} \left[(r_t + \gamma V(s_{t+1}, a; \phi_k) - Q(s_t, a_t; \theta_k))^2 \right] \quad (9)$$

Ensemble Architecture

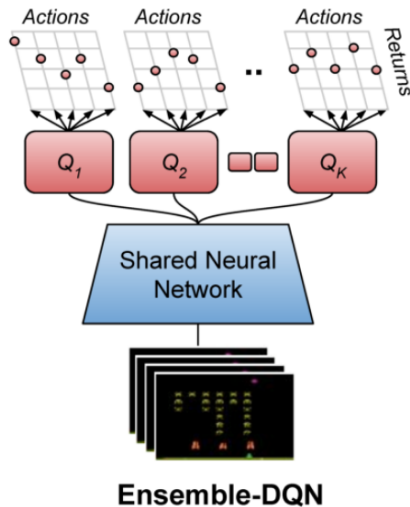
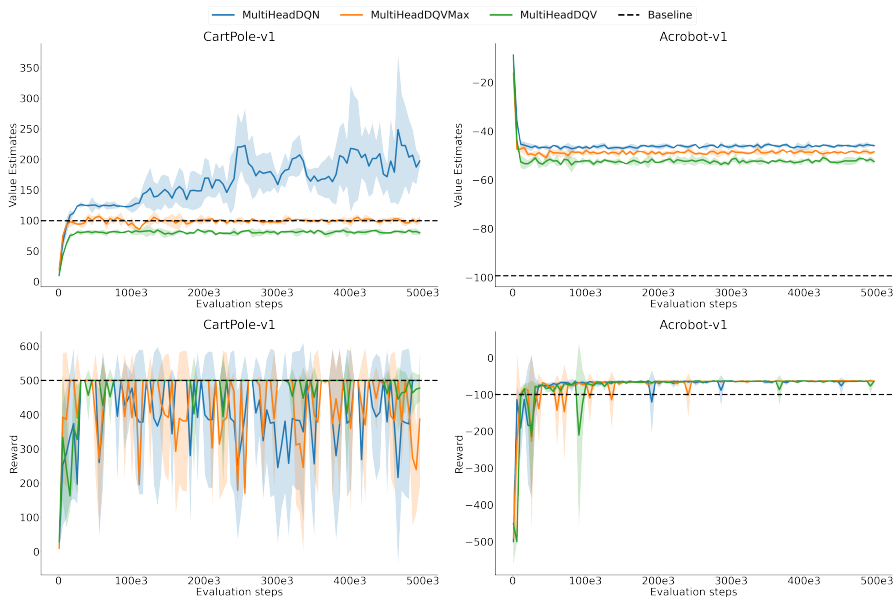


Figure 4: Multi-head Neural Network from Agarwal et al.

Bootstrapping Error with Multi-Headed DQV agents



- ▶ No real improvement over the traditional DQV algorithms
- ▶ The decoupling of estimation and update in the off-policy DQV-Max is stronger than the gains from multiple estimation observed with base DQN
- ▶ Rigorous analysis of the TAE for the DQV algorithms needed

- [1] Agarwal, R., Schuurmans, D., and Norouzi, M. (2020). An optimistic perspective on offline reinforcement learning. In *International Conference on Machine Learning*, pages 104–114. PMLR.
- [2] Anschel, O., Baram, N., and Shimkin, N. (2017). Averaged-dqn: Variance reduction and stabilization for deep reinforcement learning. In *International conference on machine learning*, pages 176–185. PMLR.
- [3] Castro, P. S., Moitra, S., Gelada, C., Kumar, S., and Bellemare, M. G. (2018). Dopamine: A Research Framework for Deep Reinforcement Learning.
- [4] Fujimoto, S., Meger, D., and Precup, D. (2019). Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*, pages 2052–2062. PMLR.
- [5] Kumar, A., Fu, J., Soh, M., Tucker, G., and Levine, S. (2019). Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in Neural Information Processing Systems*, 32.

- [6] Kumar, Aviral (2019). Data-Driven Deep Reinforcement Learning. <https://bair.berkeley.edu/blog/2019/12/05/bear/>. [Online; accessed 28-June-2022].
- [7] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.
- [8] Nair, A., Dalal, M., Gupta, A., and Levine, S. (2020). Accelerating online reinforcement learning with offline datasets. *CoRR*, abs/2006.09359.
- [9] Sabatelli, M., Louppe, G., Geurts, P., and Wiering, M. A. (2020). The deep quality-value family of deep reinforcement learning algorithms. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.
- [10] Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

- [11] Van Hasselt, H., Guez, A., and Silver, D. (2016). Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30.