Deep Quality-Value (DQV) Learning

Matthia Sabatelli[1], Gilles Louppe[1], Pierre Geurts[1], and Marco A. Wiering[2]

[1] Montefiore Institute, Department of Electrical Engineering and Computer Science, Université de Liège, Belgium
[2] Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence University of Groningen, The Netherlands

Abstract. We present Deep Quality-Value Learning (DQV), a novel *model-free* Deep Reinforcement Learning (DRL) algorithm which learns an approximation of the state-value function (V) alongside an approximation of the state-action value function (Q). We empirically show that simultaneously learning both value functions results in faster and better learning when compared to DRL methods which only learn an approximation of the Q function.

Keywords: Deep Reinforcement Learning · Model-Free Deep Reinforcement Learning · Temporal Difference Learning · Function Approximators

1 Preliminaries

We formulate a Reinforcement Learning (RL) setting as a Markov Decision Process (MDP) consisting of a finite set of states $\mathcal{S} = \{s^1, s^2, ..., s^n\}$, actions, $\mathcal{A}$, and a time-counter variable t [5]. In each state $s_t \in \mathcal{S}$, the RL agent can perform an action $a_t \in \mathcal{A}(s_t)$ after which it transits to the next state as defined by a transition probability distribution $p(s_{t+1}|s_t, a_t)$. At each transition from s_t to s_{t+1} the agent receives a reward signal r_t coming from the reward function $\Re(s_t, a_t, s_{t+1})$. The actions of the agent are selected based on its policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$ that maps each state to a particular action. For every state $s \in \mathcal{S}$, under policy π its *value function* is defined as: $V^{\pi}(s) = \mathbb{E}\Big[\sum_{k=0}^{\infty} \gamma^k r_{t+k} \Big| s_t = s, \pi\Big]$, while its *state-action* value function as: $Q^{\pi}(s,a) = \mathbb{E}\Big[\sum_{k=0}^{\infty} \gamma^k r_{t+k} \Big| s_t = s, a_t = a, \pi\Big]$. Both functions are computed with respect to the discount factor $\gamma \in [0, 1]$. The goal of an RL agent is to find a policy π^* that realizes the optimal expected return: $V^*(s) = \max_{\pi} V^{\pi}(s)$, for all $s \in \mathcal{S}$ and the optimal Q value function: $Q^*(s,a) = \max_{\pi} Q^{\pi}(s,a)$ for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$. Both value functions satisfy the *Bellman* optimality equation as given by $V^*(s_t) = \max_a \sum_{s_{t+1}} p(s_{t+1}|s_t, a_t)\Big[\Re(s_t, a_t, s_{t+1}) + \gamma V^*(s_{t+1})\Big]$ for the state-value function, and by $Q^*(s_t, a_t) = \sum_{s_{t+1}} p(s_{t+1}|s_t, a_t)\Big[\Re(s_t, a_t, s_{t+1}) + \gamma \max_{a_{t+1}} Q^*(s_{t+1}, a_{t+1})\Big]$,

for the state-action value function. In what follows we show how to learn an approximation of both value functions with deep learning methods [2].

2 The Deep Quality-Value (DQV) Learning Algorithm

Deep Quality-Value (DQV) Learning learns an approximation of the V function alongside an approximation of the Q function. This is done with two neural networks, respectively parametrized as Φ and θ, and two objective functions that can be minimized by gradient descent. Such objectives adapt two tabular RL update rules presented in [7] and result in the following losses for learning the Q and V functions: $L_{(\theta)} = \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1}\rangle \sim U(D)}\left[\left(r_t + \gamma V(s_{t+1}; \Phi^-) - Q(s_t, a_t; \theta)\right)^2\right]$; $L_{(\Phi)} = \mathbb{E}_{\langle s_t, a_t, r_t, s_{t+1}\rangle \sim U(D)}\left[\left(r_t + \gamma V(s_{t+1}; \Phi^-) - V(s_t; \Phi)\right)^2\right]$. Both losses are computed with respect to the same target $(r_t + \gamma V(s_{t+1}); \Phi^-)$, which uses an older version of the V network (Φ^-) for computing the temporal-difference errors. Minimizing these objectives is done over batches of RL trajectories ($\langle$ s_t, a_t, r_t, $s_{t+1}\rangle$) that get uniformly sampled from a memory buffer D. Our results [4], presented in Fig. 1, show that DQV learns significantly faster and better than DQN [3] and DDQN [6] on several DRL test-beds coming from the `Open-AI-Gym` environment [1]. This empirically highlights the benefits of learning two value functions simultaneously and makes DQV a new faster synchronous value-based algorithm present in DRL.

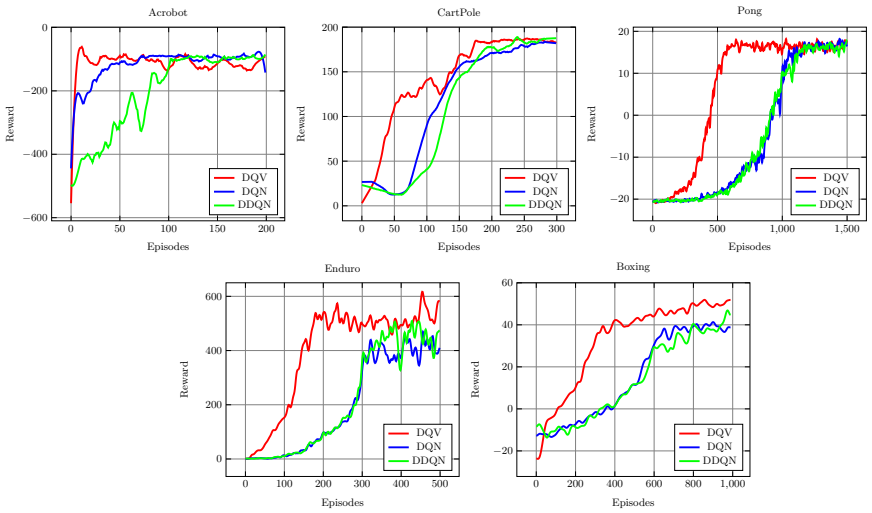

Fig. 1. The results obtained by DQV on several RL environments coming from the `Open-AI-Gym` [1] benchmark. DQV learns significantly faster than DQN and DDQN on all test-beds. Results adapted from [4].

References

1. Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. OpenAI gym. *arXiv preprint arXiv:1606.01540*, 2016.
2. Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436, 2015.
3. Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.
4. Matthia Sabatelli, Gilles Louppe, Pierre Geurts, and Marco Wiering. Deep quality value (dqv) learning. In *Advances in Neural Information Processing Systems, Deep Reinforcement Learning Workshop*. Montreal, 2018.
5. Richard S Sutton and Andrew G Barto. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.
6. Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double Q-learning. In *AAAI*, volume 16, pages 2094–2100, 2016.
7. Marco A Wiering. QV (lambda)-learning: A new on-policy reinforcement learning algorithm. In *Proceedings of the 7th European Workshop on Reinforcement Learning*, pages 17–18, 2005.