Environment-related differences of Deep Q-learning and Double Deep Q-learning

Santhosh Rajamanickam, David Rau, Dennis Ulmer & Christina Winkler

University of Amsterdam, December 2018

Introduction

- Q-Learning can suffer from maximization bias
- **Remedy**: Use two independent Q-functions [1]!

Research Questions

- In **what environments exists a difference** between (single) Deep Q-learning and Double Deep Q-Learning?
- Are there environments where **(Single) Deep Q-Networks are better**?

Background

Deep Q-Networks

- Parameterize value function $Q(s, a; \boldsymbol{\theta}_t)$ using Deep Neural Networks; Update $\boldsymbol{\theta}_t$ with

$$\boldsymbol{\theta}_{t+1} \leftarrow \boldsymbol{\theta}_t + \eta \left(Y_t^Q - Q(S_t, A_t; \boldsymbol{\theta}_t)\right) \nabla_{\boldsymbol{\theta}_t} Q(S_t, A_t; \boldsymbol{\theta}_t) \quad (1)$$

- η - learning rate
- Y_t^Q - target value at time step t

Stabilizing training

- Target value computed by network with weights $\boldsymbol{\theta}_t^-$:

$$Y_t^Q \equiv R_{t+1} + \gamma \max_a Q(S_{t+1}, a; \boldsymbol{\theta}_t^-) \quad (2)$$

- $\boldsymbol{\theta}_t^-$ not trained but copy from $\boldsymbol{\theta}_t$ every τ time steps
- Experience replay to decorrelate transitions

Double Deep Q-Networks

- Target value *Double Deep Q-Network* (DDQN):

$$Y_t^{Q^{\text{double}}} \equiv R_{t+1} + \gamma Q(S_{t+1}, \arg\max_a Q(S_{t+1}, a; \boldsymbol{\theta}_t); \boldsymbol{\theta}_t^-) \quad (3)$$

- Online network ($\boldsymbol{\theta}_t$) used to select action
- Target network ($\boldsymbol{\theta}_t^-$) used to evaluate chosen action

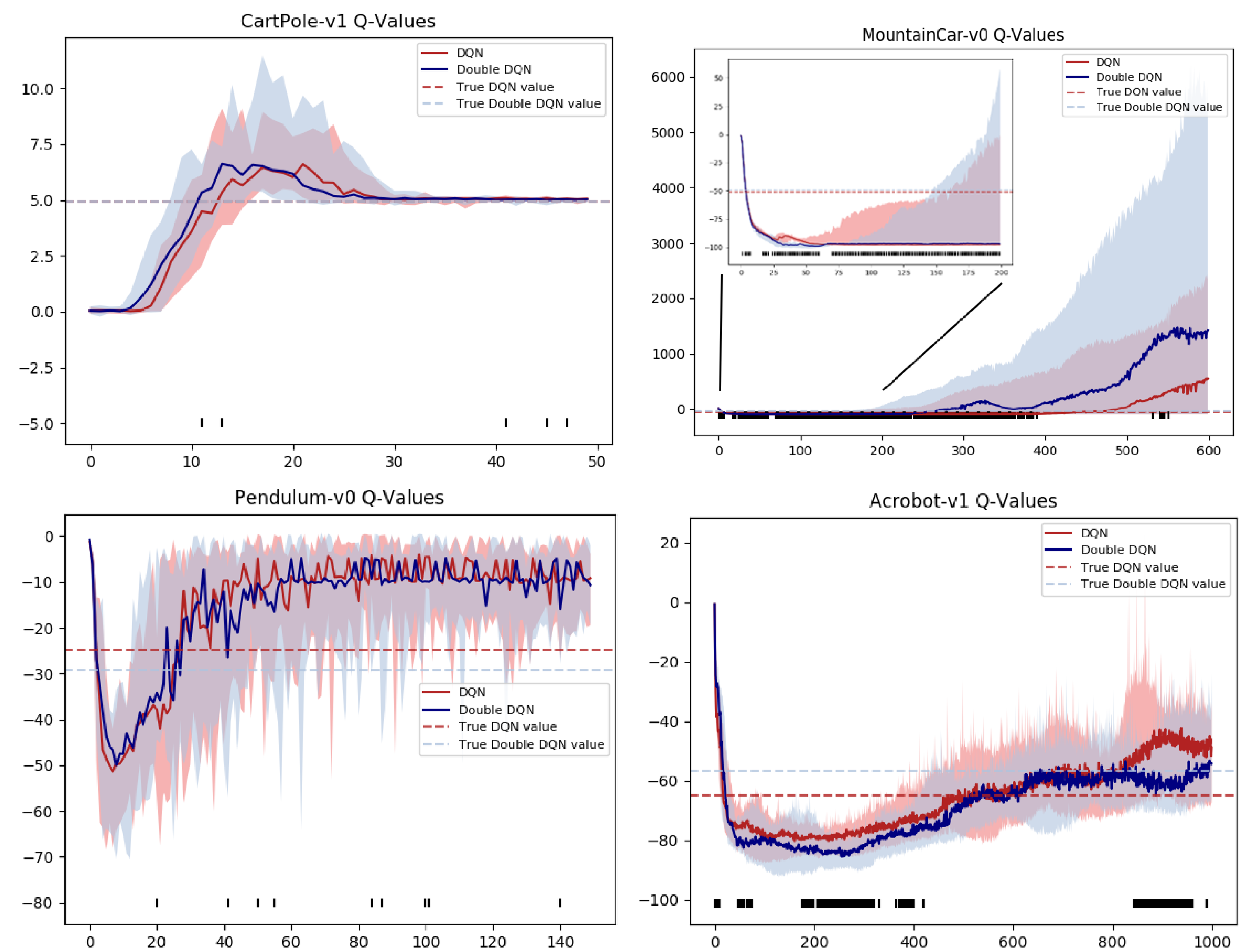

Figure 1: Average Q-Value estimates (y-axis) for 15 different models each during training (median curves) and real Q-Values during testing (dashed lines, obtained using one full Monte-Carlo rollout), number of episodes (x-axis). Intervals are determined by averaging the two extreme values. Markers (bottom of x-axis) indicate episodes with statistically significant differences between DQN and DDQN ($p = 0.05$).

Experiments

Setup

- *CartPole-v1*, *Acrobot-v1*, *MountainCar-v0*, *Pendulum-v0* from Open AI Gym [2]
- Discrete MountainCar and discretized Pendulum
- Stop environment after 1000 steps
- Joint hyperparameter grid search: Selected by highest reward using 2×10 random seeds per environment

Experiment

- Train $k = 15$ different models per environment
- Test for significant differences in Q-Values (Mann-Whitney U [3])

Results

- Both algorithms perform well on **CartPole-v1**; environment less challenging due to easy credit assignment (immediate, positive and constant rewards)
- **Pendulum-v0**: Similar Q-value estimates, but DQN performs better than DDQN: Reason might be due to the complex reward function requiring careful actions
- Confirming [4] for **Acrobot-v1**: DDQN performs better with better estimates
- Both algorithms solve **MountainCar-v0**, but Q-estimates likely influenced by "deadly triad" [5]

Conclusion

- Only Acrobot-v1 shows significant performance improvement when using DDQN
- DDQN performance improvement **depends on the reward structure** of the environment
- Function approximation (Neural Network), bootstrapping and off-policy learning ("deadly triad") can lead to **unstable Q-values** while **still achieving the objective**
- Bad Q-value estimates do **not necessarily imply bad performance** (cp. MountainCar-v0)

Code, demonstrations and more details about the experiments can be found online under `https://github.com/Kaleidophon/quirky-quokka`

References

[1] Hado V Hasselt. Double q-learning. In *Advances in Neural Information Processing Systems*, pages 2613–2621, 2010.

[2] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[3] Nadim Nachar et al. The mann-whitney u: A test for assessing whether two independent samples come from the same distribution. *Tutorials in Quantitative Methods for Psychology*, 4(1):13–20, 2008.

[4] Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 2094–2100. AAAI Press, 2016.

[5] Hado van Hasselt, Yotam Doron, Florian Strub, Matteo Hessel, Nicolas Sonnerat, and Joseph Modayil. Deep reinforcement learning and the deadly triad. *arXiv preprint arXiv:1812.02648*, 2018.

[6] 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.