Reducing Uncertainty Propagation in Markov Decision Processes

Davide Tateo, Alessandro Nuara, Carlo D'Eramo
Department of Electronics, Information and Bioengineering
Politecnico Di Milano, Milano, Italy
Email: {davide.tateo, alessandro.nuara, carlo.deramo}@polimi.it

***Abstract*—In many real-world problems stochasticity is a critical issue for the learning process. The sources of stochasticity come from the transition model, the explorative component of the policy or, even worse, from noisy observation of the reward function. For a finite number of samples, traditional reinforcement learning methods provide biased estimates of the action-value function. The presence of the bias leads to a poor estimation of the action-value function that is propagated to other action-values by the application of the Bellman operator. We propose an approach that significantly mitigates this issue avoiding the propagation of bad estimates of the action-value function.**

I. Introduction

It is well known that a key issue of reinforcement learning problems is the accuracy of the estimation of the action-value with a limited number of samples. While most algorithms guarantee the convergence of the estimates to the optimal action-value, in practice the presence of stochastic components lead to poor performance. In fact, the majority of real-world problems have significant sources of stochasticity: the environment could be have stochastic transition and this complicates the estimation of the effectiveness of an action; most of the times it is necessary to use stochastic policies to guarantee that all states are visited infinitely many times, that is required to guarantee the convergence of the algorithm; the policy could change during the learning process resulting in very different behaviors; the reward function is often corrupted by noisy observations and, in other cases, the reward function is stochastic itself. Moreover, it usually happens that some deterministic environments are partially observable and, thus, are perceived by the agent as stochastic decision processes (e.g. Blackjack).

Since Monte-Carlo estimates of action-values are affected by high variance of the returns, the most successful reinforcement learning algorithms are based on bootstrapping (e.g. Q-Learning [1]), that trades off the variance of the estimation with a consistent but biased estimator. However with a finite number of samples the bias of the estimation could be significantly relevant when propagating the action-values to the next state and, recursively, it propagates to all the other states. Recent works tried to deal with this issue, in particular focusing on the estimation of the maximum expected value. It is well known [2], [3] that the maximum operator (used in the Q-Learning update) is positively biased, thus it overestimates the optimal action-value. In highly stochastic environments, this overestimation leads to unstable learning and poor convergence rates. To avoid the presence of the positive bias, the Double Q-Learning algorithm [4] has been proposed. This algorithm uses the double estimator [5] that provides a negatively biased estimation of the action-values (i.e. it underestimates the optimal action-value) and this improves the performance when stochasticity is an issue. Another recently proposed approach is the Weighted Q-Learning [6] that balances between underestimation and overestimation.

However, the correlation between bias of the estimation and performance is still unclear; indeed, as also shown in the empirical section of this paper, Speedy Q-Learning algorithm [7] has very good performance despite its extremely poor estimation of the action-value. This is due to the fact that most of the policies are not dependent on the accuracy of the action-values, but instead they rely on their ordering. Starting from this consideration, we try to address this problem from another point of view. Indeed, we do not focus on the estimation of the maximum expected value, but we care about avoiding the propagation of the action-value estimates. In fact, we propose this work starting from the consideration that it is not relevant whether the approximations of the point estimates of the maximum expected value are good or not, but how these information is propagated to the other states. Our idea is to propagate the information about the value of the best action only when there is sufficiently certainty about the estimation.

The interesting aspect of this approach is that it is possible to use any maximum expected value estimator and, moreover, it can be easily extended in an on-policy scenario. Since we believe that in our approach the choice of the estimator is not a relevant issue, we choose the maximum operator as it is the simplest one.

II. Preliminaries

III. The proposed Method

A. Decomposition of the TD error

Decompose Q function:

$$\begin{aligned} Q(x,u) &= \mathbb{E}\left[R(x,u,x') + \gamma Q(x', \pi(x'))\right] \\ &= \mathbb{E}\left[R(x,u,x')\right] + \gamma \mathbb{E}\left[Q(x', \pi(x'))\right] \\ &= \tilde{R}(x,u) + \gamma \tilde{Q}(x,u) \end{aligned} \tag{1}$$

Decomposed TD update:

$$\tilde{R}(x,u) \leftarrow \tilde{R}(x,u) + \alpha(R(x,u,x') - \tilde{R}(x,u)) \quad (2)$$

$$\tilde{Q}(x,u) \leftarrow \tilde{Q}(x,u) + \beta(Q(x',\pi(x')) - \tilde{Q}(x,u)) \quad (3)$$

Update of the Q function:

$$\begin{aligned} Q(x,u) &\leftarrow \tilde{R}(x,u) + \alpha(R(x,u,x') - \tilde{R}(x,u)) \\ &+ \gamma\left(\tilde{Q}(x,u) + \beta(Q(x',\pi(x')) - \tilde{Q}(x,u))\right) \\ &= Q(x,u) + \alpha(R(x,u,x') - \tilde{R}(x,u)) \\ &+ \gamma\beta(Q(x',\pi(x')) - \tilde{Q}(x,u)) \end{aligned} \quad (4)$$

B. Analysis of the decomposed update

If $\alpha = \beta$

$$\begin{aligned} Q(x,u) &\leftarrow Q(x,u) + \alpha(R(x,u,x') + \gamma Q(x',\pi(x')) \\ &- Q(x,u)) \end{aligned} \quad (5)$$

That is the classical Q-Learning update

If $\beta = \delta\alpha$

$$\begin{aligned} Q(x,u) &\leftarrow Q(x,u) + \alpha(R(x,u,x') + \gamma\delta Q(x',\pi(x')) \\ &- (\tilde{R}(x,u) + \gamma\delta\tilde{Q}(x,u))) \\ &= Q(x,u) + \alpha(R(x,u,x') + \gamma' Q(x',\pi(x')) \\ &- (\tilde{R}(x,u) + \gamma'\tilde{Q}(x,u))) \\ &= Q(x,u) + \alpha((R(x,u,x') + \gamma' Q(x',\pi(x'))) \\ &- Q'(x,u)) \end{aligned} \quad (6)$$

With $\gamma' = \gamma\delta$. Notiche that $Q'(x,u)$ is the current Q function with a different learning rate.

C. Variance dependent learning rate

$$\alpha = \frac{\sigma^2}{\sigma^2 + 1} \quad (7)$$

IV. Experimental results

V. Conclusion

References

[1] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[2] J. E. Smith and R. L. Winkler, "The optimizer's curse: Skepticism and postdecision surprise in decision analysis," *Management Science*, vol. 52, no. 3, pp. 311–322, 2006.

[3] E. Van den Steen, "Rational overoptimism (and other biases)," *American Economic Review*, pp. 1141–1151, 2004.

[4] H. van Hasselt, "Double q-learning," in *Advances in Neural Information Processing Systems*, 2010, pp. 2613–2621.

[5] ——, "Estimating the maximum expected value: an analysis of (nested) cross-validation and the maximum sample average," *arXiv preprint arXiv:1302.7175*, 2013.

[6] C. DEramo, M. Restelli, and A. Nuara, "Estimating maximum expected value through gaussian approximation," in *International Conference on Machine Learning*, 2016, pp. 1032–1040.

[7] M. Ghavamzadeh, H. J. Kappen, M. G. Azar, and R. Munos, "Speedy q-learning," in *Advances in Neural Information Processing Systems 24*, J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2011, pp. 2411–2419. [Online]. Available: http://papers.nips.cc/paper/4251-speedy-q-learning.pdf