Reinforcement Learning: Q-Learning and SARSA

Yiying Zhang

Advanced Topics in Data Mining and Machine Learning
Mar. 28 2018

1. Readings

[1] Russell, Stuart J., and Peter Norvig. Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited,, 2016.

[2] Watkins, Christopher JCH, and Peter Dayan. "Q-learning." Machine learning 8.3-4 (1992): 279-292. https://link.springer.com/content/pdf/10.1007%2FBF00992698.pdf

[3] Rummery, Gavin A., and Mahesan Niranjan. On-line Q-learning using connectionist systems. Vol. 37. University of Cambridge, Department of Engineering, 1994. http://mi.eng.cam.ac.uk/reports/svr-ftp/auto-pdf/rummery_tr166.pdf

2. Abstract

Reinforcement Learning is learning what to do – how to map situations to actions – so as to maximize a numerical reward signal. The primary aim here is to cast learning as a problem involving agents that interact with an environment , and choose actions based on these interactions. The agent comes pre-equipped with goals that it seeks to satisfy. These goals are embodied in the influence of a ‘numerical reward signal’ on the way that the agent chooses actions, categorizes its sensations and changes its internal model of the environment.

Q-learning is a form of model-free reinforcement learning. It can also be viewed as a method of asynchronous dynamic programming. It provides agents with the capability of learning to act optimally in Markovian domains by experiencing the consequence of actions, without requiring them to build maps of the domains.   In Q-learning you start by setting all your state-action values to random and you go around and explore the state-action space. After you try an action in a state, you evaluate the state that it has led to. If it has led to an undesirable outcome you reduce the Q value (or weight) of that action from that state so that other actions will have a greater value and be chosen instead the next time you’re in that state. So you’re more likely to choose it again the next time you’re in that state.

SARSA (for State-Action-Reward-State-Action) is close to Q-learning. Q-learning backs up the best Q-value from the state reached in the observed transition while SARSA waits until an action is actually taken and backs up the Q-value for that action. Now, for a greedy agent that always takes the action with best Q-value, the two algorithms are identical. When exploration is happening, however, they differ significantly. Because Q-learning uses the best Q-value, it pays no attention to the actual policy being followed—it is an off-policy learning algorithm, whereas SARSA is an on-policy algorithm.

Q-learning is more flexible than SARSA, in the sense that a Q-learning agent can learn how to behave well even when guided by a random or adversarial exploration policy. On the other hand, SARSA is more realistic: for example, if the overall policy is even partly controlled by other agents, itis better to learn a Q-function for what will actually happen rather than what the agent would like to happen.

3. Spotlight Question

Is it better to learn a model and a utility function or to learn an action-utility function with no model?