EFFICIENT EXPLORATION IN REINFORCEMENT LEARNING

JOHN LANGFORD

Yahoo! Research

Synonyms

PAC-MDP learning

Definition

An agent acting in a world makes observations, takes actions, and receives rewards for the actions taken. Given a history of such interactions, the agent must make the next choice of action so as to maximize the long term sum of rewards. To do this well, an agent may take suboptimal actions which allow it to gather the information necessary to later take optimal or near-optimal actions with respect to maximizing the long term sum of rewards. These information gathering actions are generally considered exploration actions.

Motivation

Since gathering information about the world generally involves taking suboptimal actions compared to a later learned policy, minimizing the number of information gathering actions helps optimize the standard goal in reinforcement learning. In addition, understanding exploration well is key to understanding reinforcement learning well, since exploration is a key aspect of reinforcement learning which is missing from standard supervised learning settings.

Efficient Exploration in Markov Decision Processes

One simplification of reinforcement learning is the Markov Decision Process setting. In this setting, an agent repeatedly takes an action a, resulting in a transition to a state according to a conditional probability transition matrix $P(s'|s,a)$ and a (possibly probabilistic) reward $R(s',a,s) \in [0,1]$. The goal is to efficiently output a policy π which is ϵ-optimal over T timesteps. The value of policy π in a start state s is defined as:

$$\eta(\pi, s) = E_{(a,s,r)^T \sim (\pi,P,R)^T} \sum_{t=1}^{T} r_t$$

Which should be read as the expectation over T-length sequences drawn from the interaction of the policy π with the world as represented by P and R. An ϵ-optimal policy π therefore satisfies:

$$\max_{\pi'} \eta(\pi', s) - \eta(\pi, s) \leq \epsilon$$

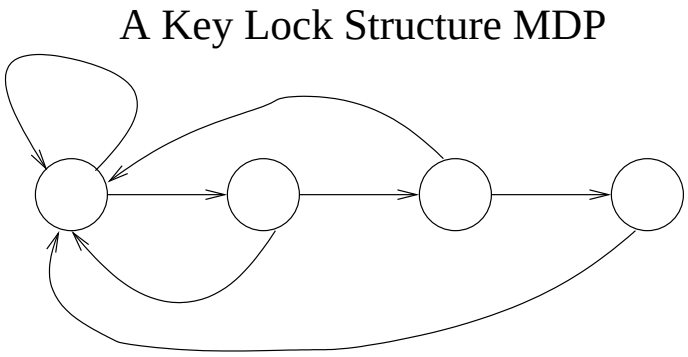

FIGURE 1. An example of a keylock MDP. The state are arranged in a chain. In each state, one of the two actions leads to the next state while the other leads back to the beginning. The only reward is in the transition to the last state in the chain. Keylock MDPs defeat simple greedy strategies, because the probability of randomly reaching the last transition is exponentially small in the length of the chain.

There are several notable results in this setting, typically expressed in terms of the dependence on the number of actions A, and the number of states S. The first is for the β-greedy strategy commonly applied when using Q-learning[9] which explores randomly with probability β.

Theorem. *There exists MDPs such that with probability at least* $1/2$*,* β*-greedy requires* $\Theta(A^S)$ *explorations to find an* ϵ*-optimal policy.*

This is essentially a negative result, saying that a greedy exploration strategy cannot quickly discover a good policy in some settings. The proof uses an MDP with a key-lock like structure where for each state all actions but 1 take the agent back to the beginning state, and the reward is at the end of a chain of states.

It turns out that there exists algorithms capable of finding a near-optimal policy in an MDP with only a polynomial number of transitions.

Theorem. *For all MDPs, for any* $\delta > 0$*, with probability* $1 - \delta$*, the algorithm Explicit-Explore-or-Exploit finds an* ϵ*-optimal policy after* $\tilde{O}(S^2A)$ *explorations.*

In other words, E^3[5] requires exploration steps at most proportional to the size of the probability table driving the dynamics of the agent's world. The algorithm works in precisely the manner which might be expected: it builds a model of the world based on it's observations and solves the model to determine whether to explore or exploit.

It turns out that an even better dependence is possible using the Delayed Q-learning [8] algorithm.

Theorem. *For all MDPs, for any* $\delta > 0$*, with probability* $1 - \delta$*, the algorithm Delayed Q-learning finds an* ϵ*-optimal policy after* $\tilde{O}(SA)$ *explorations.*

The Delayed Q-learning algorithm requires explorations proportional to the size of the solution policy rather than proportional to the size of world dynamics. At a

high level, Delayed Q-learning operates by keeping values for exploration and exploitation of observed state-actions, uses these values to decide between exploration and exploitation, and carefully updates these values. Delayed Q-learning does not obsolete E^3, because the (nonvisible) dependence on ϵ and T are worse[7].

This is a best possible result in terms of the dependence on S and A (up to log factors), as the following theorem [3] states:

Theorem. *For all algorithms, there exists an MDP such that with $\Omega(SA)$ explorations are required to find an ϵ optimal policy with probability at least $\frac{1}{2}$.*

Since even representing a policy requires a lookup table of size SA, this algorithm-independent lower bound is relatively unsurprising.

Variations on MDP learning. There are several minor variations in the setting and goal definitions which do not qualitatively impact the set of provable results. For example, if rewards are in a bounded range, they can be offset and rescaled to the interval $[0, 1]$.

It's also common to use a soft horizon where the policy evaluation is changed to:

$$\eta_\gamma(\pi, s) = E_{(a,s,r)^\infty \sim (\pi,P,R)^\infty} \sum_{t=1}^{\infty} \gamma^t r_t$$

for some value $\gamma < 1$. This setting is not precisely equivalent to the hard horizon, but since $\sum_{t=\frac{\ln \frac{1}{\epsilon} + \ln \frac{1}{1-\gamma}}{1-\gamma}}^{\infty} \gamma^t r_t \leq \epsilon$, similar results are provable with $\frac{1}{1-\gamma}$ taking the role of T and slightly altered algorithms.

One last variation changes the goal. Instead of outputting an ϵ-optimal policy for the next T timesteps, we could have an algorithm handle both the exploration and exploitation, then retrospectively go back over a trace of experience and mark a subset of the actions as "exploration actions", with a guarantee that the remainder of the actions are according to an ϵ-optimal policy[3]. Again, minor alterations to known algorithms in the above setting appear to work here.

Alternative Settings. There are several known analyzed variants of the basic setting formed by making additional assumptions about the world. This includes Factored MDPs [4], Metric MDPs [2], Continuous MDPs[1], and MDPs with a Bayesian prior [6].

See Also

Reinforcement Learning, Bandit learning

References

[1] Emma Brunskill, Bethany R. Leffler, Lihong Li, Michael L. Littman, and Nicholas Roy: CORL: A continuous-state offset-dynamics reinforcement learner. UAI-08, Helsinki, Finland, July, 2008.

[2] Sham Kakade, Michael Kearns, and John Langford Exploration in Metric State Spaces ICML2003.

[3] Sham Kakade, Thesis at Gatsby Computational Neuroscience Unit, 2003.

[4] Michael Kearns and Daphne Koller. Efficient Reinforcement Learning in Factored MDPs. Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, Morgan Kaufmann, 1999, pages 740–747.

[5] Michael Kearns and Satinder Singh, Near-Optimal Reinforcement Leanring in Polynomial Time, ICML 1998, pages 260-268.

[6] Pascal Poupart, Nikos Vlassis, Jesse Hoey, Kevin Regan. An analytic solution to discrete Bayesian reinforcement learning, ICML 2006.
[7] Alex Strehl, Thesis at Rutgers University, 2007.
[8] Alexander L. Strehl, Lihong Li, Eric Wiewiora, John Langford, and Michael L. Littman. PAC Model-Free Reinforcement Learning. In the proceedings of the 23rd International Conference on Machine Learning (ICML 2006), pages 881-888.
[9] Chris Watkins and Peter Dayan. Q-learning, Machine Learning Journal, 8, pages 279-292.