
Imitative Policies for Reinforcement Learning

Gary Cottrell, Dana Dahlstrom Charles Elkan and Eric Wiewiora

Department of Computer Science and Engineering
University of California, San Diego
La Jolla CA 92093-0114, USA
{gary,dana,elkan,wiewiora}@cs.ucsd.edu

Abstract

We discuss a reinforcement learning framework where learners observe experts interacting with the environment. Our approach is to construct from these observations exploratory policies which favor selection of actions the expert has taken. This imitation strategy can be applied at any stage of learning, and requires neither that information regarding reinforcement be conveyed from the expert to the learner nor that the learner have any explicit knowledge of its reinforcement structure. We show that learning with an imitative policy can be much faster than learning in isolation. We also show that our approach is robust to sub-optimal experts.

1 Introduction

In its standard formulation, reinforcement learning assumes a solipsistic environment where the learner’s task is to find a reward-maximizing control policy starting with no prior knowledge and receiving no outside help—the only available information is first-hand experience. In the real world, however, learners are usually part of social networks from which they can cull some knowledge pertaining to their tasks. For such social learners, “Learning is more often a transfer than a discovery” [1].

Most social machine learning algorithms take the form of a passive agent learning directly from the behavior of another agent [2, 3, 4, 5]. These methods require that the learner is able to derive information regarding the actor’s state, goals and actions. The learner makes no judgement on the competence of the actor *a priori*; instead the observations are evaluated as if they were performed by the learner itself. This approach has been called *learning by watching* (LBW). While LBW is robust to ineffective teachers, if the benefit of observed actions are not immediately apparent, potentially useful observations are ignored.

Another branch of social learning in machine learning makes use of an expert that provides feedback on a learner’s actions [2, 6, 7]. This approach can be called *learning by instruction*. In order for learning by instruction to be effective, an attentive and skilled expert must be present.

A relatively simple way to learn from others is by imitating them. If their behavior is expedient, imitation may yield better performance much faster than lone experimentation. Of course others may not be perfect, or they may have different goals entirely. Thus it is generally unwise to merely imitate; one should pay attention to first-hand experience

as well. When imitated behavior is not perfect, it may be possible to improve upon it by experimentation.

The approach we propose is to use observations of others to guide the learner’s exploration. We assume the learner can observe the states and actions of others but not necessarily their rewards. We also assume the learner can identify its own states with those of others, and that the actions it observes others taking are available to it. These assumptions are consistent with a model in which the learner and those it observes are of the same species but may have different goals.

In an environment where many agents of the same species are present, some agents may be older and more experienced, and thus worthy of being imitated by less experienced agents. There is some evidence that primates are able to identify the actions of others in terms of their own actions through mirror neurons.

2 Background

Our imitation learning framework makes use of the standard approach to reinforcement learning. The task is to learn a policy for interacting with a Markov Decision process in order to maximize expected reward. Q-learning is a popular mechanism for learning this task. In order to learn effectively, an agent must be exposed to a wide variety of states and actions, as well as their consequences. The Imitation learning framework focuses the learner’s attention on actions it has seen an expert perform, and are thus more promising than an aimless action.

2.1 Markov decision processes

Most reinforcement learning techniques model the learning environment as a Markov decision process (MDP) [8]. An MDP is a quadruple (S, A, T, R) , where

- S is the set of states,
- A is the set of actions,
- $T(s'|s, a)$ is the probability of transitioning to state s' when performing action a in state s , and
- $R(s, a, s')$ is the reinforcement received when action a is performed in state s and there is a transition to state s' .

The reinforcement learning task is to find a policy $\pi : S \rightarrow A$ that maximizes the total discounted reinforcement $\sum_{t=0}^{\infty} \gamma^t r_t$ where r_t is the reinforcement received at time t and γ is the discount rate determining the relative importance of future versus immediate reinforcement.

2.2 Q-Learning

Q-learning is a reinforcement learning algorithm based on estimating the expected total discounted reinforcement $Q(s, a)$ received when taking action a in state s [8]. An *experience* is a quadruple (s, a, r, s') where action a is taken in state s , resulting in reinforcement r and a transition to next state s' . We consider an implementation of Q-learning which stores Q values in a tabular format; for each experience, a table entry is updated according to the rule

$$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + \alpha[r + \gamma \operatorname{argmax}_{a'} Q(s', a')]$$

where α is the learning rate. The greedy policy $\pi_g(s) = \operatorname{argmax}_a Q(s, a)$ is optimal when the Q values are accurate.

To guarantee the entries in the table converge to true Q values, all state-action pairs must be explored infinitely often. This can be ensured by using an exploration strategy such as ϵ -greedy: with probability ϵ , choose an action uniformly at random; otherwise choose the greedy action $\pi_g(s)$.

3 Imitative policies

We present two imitative policies for reinforcement learning and show how they integrate with Q-learning. *Action biasing* modifies the greedy objective so observed actions are more likely to be taken; *estimated policy exploration* probabilistically takes actions directly from an estimation of the expert’s policy. Both strategies are applicable whenever observation is available—be it prior to direct experience, in parallel with it, or intermittent.

Both mechanisms use the count $c(s, a)$ of the times the expert has been observed taking action a in state s and the count $c(s)$ of the times it has been in s . For the purpose of decaying expert influence as the learner gains experience, both mechanisms also use and the number of visits $v(s)$ the learner has made to s .

3.1 Action biasing

In action biasing, the policy is defined with respect to the bias function

$$B(s, a) = \begin{cases} +b & : c(s, a) > 0 \\ -b & : \text{otherwise} \end{cases}$$

where the nominal bias b is the magnitude of the bias toward taking an action the expert has taken, and away from taking one it hasn’t. The learner uses a biased ϵ -greedy policy

$$\pi^B(s) = \begin{cases} a \sim U(A) & \text{with probability } \epsilon \\ \operatorname{argmax}_a [(1-w)Q(s, a) + wB(s, a)] & \text{otherwise} \end{cases}$$

where $U(A)$ is a uniform distribution over the actions and w is the bias weight, which decays according to

$$w = \delta^{v(s)}$$

where $\delta \in [0, 1)$ is the decay rate.

Though similar in form, action biasing is different from Whitehead’s BB-LEC in several respects. There is no external critic; the bias function is derived from observation of an expert rather than from criticism of the learner’s actions. Also, unlike in BB-LEC, the influence of the bias approaches zero as the learner gains experience, and thus π^B converges to a simple ϵ -greedy policy.

The simple bias function described here should work in most environments, but a more sophisticated function that biases actions based on their semantic similarity to the expert’s observed actions may have advantages.

3.2 Estimated policy imitation

Another approach is to estimate the expert’s policy and choose actions probabilistically according to this estimation. For discrete MDPs the expert’s policy can be estimated as a multinomial distribution: given a count of the times the expert has taken each action in each state, we can make a maximum likelihood estimation of the expert policy¹

$$\hat{\pi}_e(a|s) = \begin{cases} \frac{c(s,a)}{c(s)} & c(s) > 0 \\ \frac{1}{|A|} & \text{otherwise} \end{cases}$$

¹The estimated policy $\hat{\pi}_e$, like any nondeterministic policy, is a probability distribution rather than a simple mapping from states to actions; we designate this by use of conditional probability notation.

In terms of $\widehat{\pi}_e$, the learner's policy under imitation is

$$\pi^\iota(s) = \begin{cases} a \sim \widehat{\pi}_e(\cdot|s) & \text{with probability } \iota \\ \pi(s) & \text{otherwise} \end{cases}$$

where $\iota \in [0, 1]$ is the *imitation rate*, the probability with which the learner chooses an action from the estimated expert policy. The default policy π could simply be greedy, but in our experiments we use an ϵ -greedy strategy to ensure exploration.

When $\iota = 0$, the policy π^ι is identical to the default policy. To make this true in the limit we let $\iota = \delta^{v(s)}$ where $\delta \in [0, 1)$ is the decay rate, exactly as in action biasing. The imitation rate ι and the bias weight w are analogous: both specify the extent of the expert's influence on the learner's policy. Decreasing this influence is particularly well suited to learning from imperfect experts: the learner begins by simply imitating and then refines its understanding through direct experience, correcting for any errors inherited from the expert through imitation.

4 Experiments with pong

We have experimented with imitative policies in learning a control policy for a pong simulation. This game presents a natural environment for imitation learning because it involves two players.

The pong board is a 20×12 rectangle in which a point-sized ball bounces. The learner controls a 2-unit-wide paddle which can move left or right 1 unit per time step along the bottom of the board. An expert controls a paddle in a similar manner along the top of the board. The goal is to position the paddle to intercept the ball whenever it reaches the top or bottom of the board.

When the game starts, the paddles are placed 6 units from the left wall and the ball is launched from one of 12 positions at the bottom of the board. The ball is launched with an upwards velocity of 2 units per time step. The ball's horizontal velocity is uniformly chosen from -2, -1, 0, 1 or 2 units per time step.

When the learner or expert hit the ball, a reward of 1 is given to that agent. The ball is then reflected back at the other agent with a randomly perturbed position and velocity. If the agent misses, both paddles positions are reset, and the ball is randomly served from the agent who missed's side of the board.

We compare the estimated policy imitation and action bias methods to standard Q-learning. In our experiments, the imitative learners will observe the action the expert takes at every time step, as well as learn from its own actions. In order to properly observe the expert's actions in terms of the learner's side of the board, the learner must have access to a mirror-image representation of the current game state.

Our experiments are divided into two alternating phases:

Practice The agent observes expert experience while exploring the state space using an imitative policy.

Evaluation The agent neither observes nor learns; it merely exploits a greedy policy.

Practice phases last for 1000 time steps, which gives the learner 500 chances to hit the ball.

We measure performance during the evaluation phase to isolate the effects of the methods on the learned greedy policy without the noise introduced by random exploration. Our performance metric is the learner's accumulated score on ten chances to hit or miss.

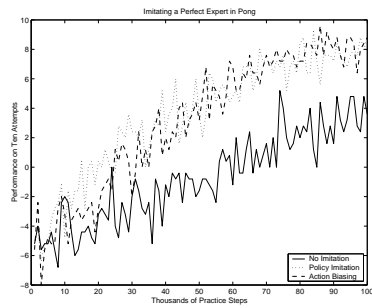


Figure 1: Observing a perfect expert.

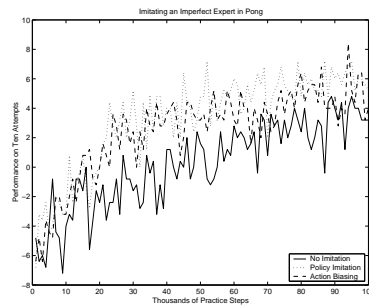


Figure 2: Observing an imperfect expert.

All the learners use a learning rate $\alpha = 0.5$ and a discount rate $\gamma = 0.95$. The action biasing learner uses a bias decay rate $\delta = 0.75$, and the estimated policy imitation learner's imitation rate decays at the same rate. The bias function used is ± 0.5 , which is a good approximate value for a learned Q-table entry. All the learners use a constant exploration rate $\epsilon = 0.25$.

We have created two expert pong agents. The perfect expert predicts where the ball will next contact the bottom edge and moves the paddle there directly. The imperfect expert attempts to keep the paddle directly beneath the ball at all times. When the ball's horizontal velocity is faster than the paddle can move, this strategy results in frequent misses.

4.1 Observing the perfect expert

Figure 1 shows that using an imitative policy is clearly beneficial. By the end of the trial, the imitative learners are performing nearly perfectly, while standard Q-learning is hitting eight balls out of ten. The difference between the learners becomes more pronounced as the learner observes the expert player more.

4.1.1 Observing the imperfect expert

Figure 2 demonstrates that imitative learning is still beneficial, even when the expert is not doing exceptionally well. The largest improvement the imitative learners achieve over standard Q-learning occur about one third through the experiment. At this point, the learner has accumulated enough observations for the teacher for them to be useful, but it still hasn't learned a better policy than the expert.

5 Experiments with grid world

A grid world is a simple environment with a state for each cell in a rectangular grid and actions for moving from one cell to another. In our 10×10 grid world, depicted in Figure 3, there are 100 states and 8 actions allowing movement to the adjacent cells in each of the 4 cardinal and 4 diagonal directions. The arrow represents the optimal 9-step path from the start state in the northwest corner to the goal state in the southeast corner. Upon entering the goal state the agent receives a +1 reward and begins again at the start state; in all other cases the reinforcement is 0.

We compare estimated policy imitation to standard Q-learning using the following three phases for each epoch:

observation The agent observes 180 steps of expert performance.

exploration The agent learns from 180 steps of direct experience.

evaluation The agent's performance is measured by how many steps it takes to reach the goal from the start state exploiting a greedy policy, neither observing nor learning.

Because the standard Q-learner can make no use of observed state-action pairs, it does nothing during the observation phase. The imitative learner is only updating its estimation of the expert's policy during the observation phase, though, so it does the same number of Q-table updates per epoch as does the standard Q-learner.

Since the environment is completely deterministic, the learners use a learning rate $\alpha = 1$. During the exploration phase they use an exploration rate $\epsilon = 0.75$; this is a much higher value than is traditionally used, but it makes sense because performance is not measured during this phase. The value $\epsilon = 0.75$ was empirically observed to approximately maximize the performance of both the imitative learner and the standard Q-learner.

There are two expert agents. The diagonal expert always moves southeast and always reaches the goal in exactly 9 steps. The cardinal expert always moves south or east; if there are cells in both directions, one of the two actions is selected uniformly at random. The cardinal expert always reaches the goal in exactly 18 steps.

The imitative learner uses a decay rate $\delta = 0.75$, which was empirically observed to approximately maximize its performance when observing the cardinal expert. This decay rate is not the best choice when observing the diagonal expert; since its policy is optimal, more imitation is better. One cannot always know whether an expert is perfect, however.

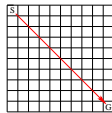


Figure 3: A simple 10×10 grid world

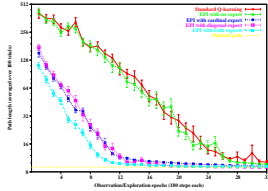


Figure 4: Estimated policy imitation vs. standard Q-learning in 10×10 grid world.

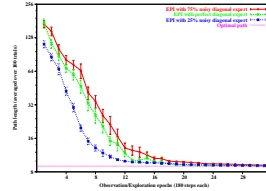


Figure 5: Imitating the perfect and noisy diagonal experts in 10×10 grid world.

5.1 Results

Figure 4 shows the performance of the standard Q-learner versus three estimated policy imitators. One observes the diagonal expert, another observes the cardinal expert, and a third observes each for half the observation phase. All the imitative learners outperform the standard Q-learner.

With the decay rate $\delta = 0.75$, imitating the cardinal expert is almost as effective as imitating the diagonal expert. Since the diagonal expert only ever takes actions in 9 states, it provides no guidance in the other 90 states in which the learner may have to act. The cardinal expert visits all states with non-zero probability, thereby providing more information, albeit of lower quality. The learner that observes both experts can benefit from ideal demonstration in the states along the diagonal as well as reasonably good demonstration in the other states, and so outperforms the learners observing only one of the two experts.

The cardinal expert is one kind of imperfect expert: one with a systematically suboptimal strategy. Another kind is a noisy optimal expert: with some probability it chooses an action uniformly at random; otherwise it takes an optimal action. Figure 5 shows the results of imitating the diagonal expert versus imitating a 25% noisy expert and a 75% noisy expert.

The 25% noisy expert visits all the states with non-zero probability; performance when imitating it is comparable to when imitating both the cardinal and diagonal experts as in Figure 4. Too much noise is problematic, however, because the learner imitates the random actions: imitation with the 75% noisy expert does worse than with the perfect expert. The conflict between covering the state space and taking consistently optimal actions could be resolved by starting an optimal expert at random states rather than always at the same state, but in general such control over the expert may not be available.

6 Discussion

When they can observe reasonably good experts, imitative learners can improve much faster than by direct experience alone. If the learner must trade off observation against direct experience, however, this benefit is not without cost. Probably the most problematic situation is imitating a bad expert, in which case the learner is biased *away* from more rewarding actions. Because our methods decay the influence of observations as the learner gains its own experience, imitating a bad expert will only slow convergence to an optimal policy rather than prevent it altogether.

Future work on imitation in reinforcement learning may include developing a principled way to decay imitation based upon the relative performance of the expert and the learner; our separate exploration and evaluation phases could facilitate this kind of comparison. If there is a cost for observation, another related problem is how to decide when it would be beneficial to observe rather than explore. It may also be productive to incorporate imitative policies into other reinforcement learning frameworks such as learning with eligibility traces, generalizing function approximators, or model-based methods.

7 Acknowledgements

We would like to thank Koji Morikawa and Panasonic for funding this research.

References

- [1] Steven D. Whitehead. A study of cooperative mechanisms for faster reinforcement learning. Technical Report 365, Department of Computer Science, University of Rochester, Mar 1991.
- [2] Steven D. Whitehead. A complexity analysis of cooperative mechanisms in reinforcement learning. In *Proceedings, Ninth National Conference on Artificial Intelligence (AAAI-91)*, volume 2, pages 607–613. AAAI Press / The MIT Press, Jul 1991.
- [3] Long-Ji Lin. Self-improving reactive agents based on reinforcement learning, planning, and teaching. *Machine Learning*, 8(3/4):293–321, May 1992.
- [4] Ming Tan. Multi-agent reinforcement learning: Independent versus cooperative. In *Machine Learning, Proceedings of the Tenth International Conference*, pages 330–337, Jun 1993.
- [5] Bob Price and Craig Boutilier. Implicit imitation in multiagent reinforcement learning. In *Machine Learning, Proceedings of the Sixteenth International Conference*, pages 325–334. Morgan Kaufmann, Jun 1999.
- [6] Paul E. Utgoff and Jeffery A. Clouse. Two kinds of training information for evaluation function learning. In *Proceedings, Ninth National Conference on Artificial Intelligence (AAAI-91)*, volume 2, pages 596–600. AAAI Press / The MIT Press, Jul 1991.
- [7] Jeffery A. Clouse and Paul E. Utgoff. A teaching method for reinforcement learning. In *Machine Learning, Proceedings of the Ninth International Workshop (ML92)*, pages 92–101. Morgan Kaufmann, Jul 1992.
- [8] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, 1998.