Multi-Agent Learning II: Algorithms

Yoav Shoham and Rob Powers
Stanford University
{shoham,powers}@cs.stanford.edu

January 16, 2010

1 Definition

Multi-agent learning refers to settings in which multiple agents learn simultaneously. Usually defined in a game theoretic setting, specifically in repeated games or stochastic games, the key feature that distinguishes multi-agent learning from single-agent learning is that in the former the learning of one agent impacts he learning of others. As a result neither the problem definition for mutli-agent learning, nor the algorithms offered, follow in a straightforward way from the single-agent case. In this second of two entries on the subject we focus on algorithms.

2 Some MAL techniques

We will discuss three classes of techniques – one representative of work in game theory, one more typical of work in AI, and one that seems to have drawn equal attention from both communities.

2.1 Model-based approaches

The first approach to learning we discuss, which is common in the game theory literature, is the model-based one. It adopts the following general scheme:

1. Start with some model of the opponent's strategy.
2. Compute and play the best response.
3. Observe the opponent's play and update your model of her strategy.
4. Goto step 2.

Among the earliest, and probably the best-known, instance of this scheme is *fictitious play*. The model is simply a count of the plays by the opponent in the past. The opponent is assumed to be playing a stationary strategy, and

the observed frequencies are taken to represent the opponent's mixed strategy. Thus after five repetitions of the Rochambeau game in which the opponent played (R, S, P, R, P), the current model of her mixed strategy is $(R = .4, P = .4, S = .2)$.

There exist many variants of the general scheme, for example those in which one does not play the exact best response in step 2. This is typically accomplished by assigning a probability of playing each pure strategy, assigning the best response the highest probability, but allowing some chance of playing any of the strategies. A number of proposals have been made of different ways to assign these probabilities such as *smooth fictitious play* and *exponential fictitious play*.

A more sophisticated version of the same scheme is seen in *rational learning*. The model is a distribution over the repeated-game strategies. One starts with some prior distribution; for example, in a repeated Rochambeau game, the prior could state that with probability .5 the opponent repeatedly plays the equilibrium strategy of the stage game, and, for all $k > 1$, with probability 2^{-k} she plays R k times and then reverts to the repeated equilibrium strategy. After each play, the model is updated to be the posterior obtained by Bayesian conditioning of the previous model. For instance, in our example, after the first non-R play of the opponent, the posterior places probability 1 on the repeated equilibrium play.

2.2 Model-free approaches

An entirely different approach that has been commonly pursued in the AI literature is the model-free one, which avoids building an explicit model of the opponent's strategy. Instead, over time one learns how well one's own various possible actions fare. This work takes place under the general heading of *reinforcement learning*[1], and most approaches have their roots in the Bellman equations. We start our discussion with the familiar single-agent *Q-learning* algorithm for computing an optimal policy in an unknown MDP.

$$\begin{aligned} Q(s,a) &\leftarrow (1-\alpha_t)Q(s,a) + \alpha_t[R(s,a) + \gamma V(s')] \\ V(s) &\leftarrow \max_{a \in A} Q(s,a) \end{aligned}$$

As is well known, with certain assumptions about the way in which actions are selected at each state over time and constraints on the learning rate schedule, α_t, Q-learning can be shown to converge to the optimal value function V^*.

The Q-learning algorithm can be extended to the multi-agent stochastic game setting by having each agent simply ignore the other agents and pretend that the environment is passive:

[1] We note that the term is used somewhat differently in the game theory literature.

$$\begin{aligned} Q_i(s,a_i) &\leftarrow (1-\alpha_t)Q_i(s,a_i) + \alpha_t[R_i(s,\vec{a}) + \gamma V_i(s')] \\ V_i(s) &\leftarrow \max_{a_i \in A_i} Q_i(s,a_i) \end{aligned}$$

Several authors have tested variations of the basic Q-learning algorithm for MAL. However, this approach ignores the multi-agent nature of the setting entirely. The Q-values are updated without regard for the actions selected by the other agents. While this can be justified when the opponents' distributions of actions are stationary, it can fail when an opponent may adapt its choice of actions based on the past history of the game.

A first step in addressing this problem is to define the Q-values as a function of all the agents' actions:

$$Q_i(s,\vec{a}) \leftarrow (1-\alpha)Q_i(s,\vec{a}) + \alpha[R_i(s,\vec{a}) + \gamma V_i(s')]$$

We are however left with the question of how to update V, given the more complex nature of the Q-values.

For (by definition, two-player) zero-sum SGs, the *minimax-Q* learning algorithm updates V with the minimax of the Q values:

$$V_1(s) \leftarrow \max_{P_1 \in \Pi(A_1)} \min_{a_2 \in A_2} \sum_{a_1 \in A_1} P_1(a_1) Q_1(s,(a_1,a_2)).$$

Later work proposed other update rules for the Q and V functions focusing on the special case of common-payoff (or 'team') games. A stage game is common-payoff if at each outcome all agents receive the same payoff. The payoff is in general different in different outcomes, and thus the agents' problem is that of coordination; indeed these are also called *games of pure coordination.*

The work on zero-sum and common-payoff games continues to be refined and extended; much of this work has concentrated on provably optimal tradeoffs between exploration and exploitation in unknown, zero-sum games. Other work attempted to extend the "Bellman heritage" to general-sum games (as opposed to zero-sum or common-payoff games), but the results here have been less conclusive.

2.3 Regret minimization approaches

Our third and final example of prior work in MAL is no-regret learning. It is an interesting example for two reasons. First, it has some unique properties that distinguish it from the work above. Second, both the AI and game theory communities appear to have converged on it independently. The basic idea goes back to early work on how to evaluate the success of learning rules in the mid-50s, and has since been extended and rediscovered numerous times over

the years under the names of universal consistency, no-regret learning, and the Bayes envelope. The following algorithm is a representative of this body of work. We start by defining the *regret*, $r_i^t(a_j, s_i)$ of agent i for playing the sequence of actions s_i instead of playing action a_j, given that the opponents played the sequence s_{-i}.

$$r_i^t(a_j, s_i|s_{-i}) = \sum_{k=1}^{t} R(a_j, s_{-i}^k) - R(s_i^k, s_{-i}^k)$$

The agent then selects each of its actions with probability proportional to $\max(r_i^t(a_j, s_i), 0)$ at each time step $t + 1$.

3 Some typical results

One sees at least three kinds of results in the literature regarding the learning algorithms presented above, and others like them. These are:

1. Convergence of the strategy profile to an (e.g., Nash) equilibrium of the stage game in self play (that is, when all agents adopt the learning procedure under consideration).

2. Successful learning of an opponent's strategy (or opponents' strategies).

3. Obtaining payoffs that exceed a specified threshold.

Each of these types comes in many flavors; here are some examples. The first type is perhaps the most common in the literature, in both game theory and AI. For example, while fictitious play does not in general converge to a Nash equilibrium of the stage game, the distribution of its play can be shown to converge to an equilibrium in zero-sum games, 2x2 games with generic payoffs, or games that can be solved by iterated elimination of strictly dominated strategies. Similarly in AI, minimax-Q learning is proven to converge in the limit to the correct Q-values for any zero-sum game, guaranteeing convergence to a Nash equilibrium in self-play. This result makes the standard assumptions of infinite exploration and the conditions on learning rates used in proofs of convergence for single-agent Q-learning.

Rational learning exemplifies results of the second type. The convergence shown is to correct beliefs about the opponent's repeated game strategy; thus it follows that, since each agent adopts a best response to their beliefs about the other agent, in the limit the agents will converge to a Nash equilibrium of the repeated game. This is an impressive result, but it is limited by two factors; the convergence depends on a very strong assumption of absolute continuity, and the beliefs converged to are only correct with respect to the aspects of history that are observable given the strategies of the agents. This is an involved topic, and the reader is referred to the literature for more details.

The literature on no-regret learning provides an example of the third type of result, and has perhaps been the most explicit about criteria for evaluating learning rules. For example, one pair of criteria that have been suggested is as follows. The first criterion is that the learning rule be 'safe', which is defined as the requirement that the learning rule guarantee at least the minimax payoff of the game. (The minimax payoff is the maximum expected value a player can guarantee against any possible opponent.) The second criterion is that the rule should be 'consistent'. In order to be 'consistent', the learning rule must guarantee that it does at least as well as the best response to the empirical distribution of play when playing against an opponent whose play is governed by independent draws from a fixed distribution. 'Universal consistency' is then defined as the requirement that a learning rule do at least as well as the best response to the empirical distribution regardless of the actual strategy the opponent is employing (this implies both safety and consistency). The requirement of 'universal consistency' is in fact equivalent to requiring that an algorithm exhibit *no-regret*, generally defined as follows, against all opponents.

$$\forall \epsilon > 0, (lim_{t \to \inf}[\frac{1}{t} \max_{a_j \in A_i} r_i^t(a_j, s_i | s_{-i})] < \epsilon)$$

In both game theory and artificial intelligence, a large number of algorithms have been show to satisfy universal consistency or no-regret requirements.

4 Recommended reading

Requisite background in game theory can be obtained from the many introductory texts, and most compactly from [Leyton-Brown and Shoham, 2008]. Game theoretic work on multiagent learning is covered in [Fudenberg and Levine, 1998] and [Young, 2004]. An expanded discussion of the problems addressed under the header of MAL cab be found in [Shoham et al., 2007], and the responses to it in [Vohra and (eds.), 2007]. Discussion of MAL algorithms, both traditional and more novel ones, can be found in the above references, as well as in [Greenwald and (Eds.), 2007].

References

[Fudenberg and Levine, 1998] Fudenberg, D. and Levine, D. (1998). *The Theory of Learning in Games*. MIT Press.

[Greenwald and (Eds.), 2007] Greenwald, A. and (Eds.), M. L. L. (2007). Special issue on learning and computational game theory. *Machine Learning*, 67(1-2).

[Leyton-Brown and Shoham, 2008] Leyton-Brown, K. and Shoham, Y. (2008). *Essentials of Game Theory*. Morgan and Claypool Publishers.

[Shoham et al., 2007] Shoham, Y., Powers, W. R., and Grenager, T. (2007). If multiagent learning is the answer, what is the question? *Artificial Intelligence*, 171(1):365–377. Special issue on Foundations of Multiagent Learning.

[Vohra and (eds.), 2007] Vohra, R. and (eds.), M. P. W. (2007). Special issue on foundations of multiagent learning. *Artificial Intelligence*, 171(1).

[Young, 2004] Young, H. P. (2004). *Strategic Learning and its Limits*. Oxford University Press.