World of Knowledge

Online Learning and Game Theory

Chi-Jen Lu/ Institute of Information Science

In our daily life, we often face the difficult task of having to make decisions before knowing the resulting outcomes, and later paying the price for our decisions. What we do not like is to have regret, wishing that we had made different but better decisions. This would probably be a hopeless task if we only made such a decision once. However, sometimes we have to make such decisions repeatedly, in different but similar situations. In such cases, we may be able to learn from the past and make better decisions as time goes by. It is not hard to imagine many such scenarios. For example, we may want to predict the weather, to trade a stock, or to choose a route from home to work, not just once, but repeatedly. There are also many such examples in computer science applications, including network routing, scheduling, resource allocation, and online advertising.

From these scenarios, one can abstract the following problem, known as the online decision problem. Suppose there are T rounds to play and there is a space of actions to choose from. In each round, we have to choose an action first, and after that receive a corresponding loss (or reward), according to some loss (or reward) function of that round. From this feedback, we can then update the way we choose the action in the next round. We would like to have an online algorithm which can help us choose a good action in each round. The question is: what is the objective we want to achieve?

A natural objective is to minimize the total loss. However, a standard way of evaluating an online algorithm is by comparing its total loss with that of an offline algorithm, which is allowed to see all the loss functions before making decisions, but is required to play the same action in every round. The difference between these two losses is called regret. A major result in this area is that a small regret, about the square root of T, can in fact be achieved, which means that the average regret per round approaches zero as T grows to infinity. Algorithms achieving such a regret bound are called no-regret algorithms, and they turn out to have impacts far beyond the area of machine learning, with surprising applications even in settings which do not seem to involve online decisions. In fact, several fundament results in different research areas can be easily derived from the existence of no-regret algorithms, including the minimax theorem of von Neumann in game theory, the linear programming duality theorem in optimization, the powerful boosting algorithms in machine learning, and the hard-core lemma for derandomization in complexity theory. Moreover,

they have been used to design efficient approximation algorithms for hard computational problems, and they have also been used to model evolutionary dynamics in biology.

Motivated by their immense importance, we investigated the possibility of further improving and generalizing no-regret algorithms. We noted that previous works mostly focused on adversarial cases with arbitrary loss functions, but we believe that the world around us may not always be adversarial and loss functions may sometimes have patterns, which could be exploited for achieving a smaller regret. We observed that the world typically evolves in a somewhat smooth way. For example, the weather condition or stock price at one moment may have some correlation with the next and their difference is usually small, while abrupt changes only occur sporadically. To model this, we introduced a new measure on how much the loss functions deviate by summing the distances between consecutive loss functions. By taking this into account, we designed a new online algorithm which can achieve a regret bound about the square root of the deviation measure. This means that when the loss functions have a small deviation, our algorithm can actually achieve a much smaller regret than previously achievable ones. On the other hand, even when going back to the adversarial case in which loss functions have no pattern and their deviation is as large as T (the number of rounds), our algorithm still recovers the same square root of T regret as previous algorithms. Therefore, previous results can be seen as special cases of ours.

We also extended our work to the more general online convex optimization problem, in which the space of actions can be infinite, but the loss functions are convex. We designed algorithms for cases in which the loss functions are (i) linear, (ii) convex, or (iii) strongly convex; again, these algorithms achieve small regrets when the loss functions have small deviations. Interestingly, all our algorithms can be unified by a single meta-algorithm: by instantiating a parameter of the meta-algorithm appropriately, we obtain all of our different algorithms. This also allows us to analyze the regrets of our algorithms for the different types of loss functions in the same framework.

In addition to designing better online algorithms, we also investigated how they might be applied to other areas. One application we discovered is in the area of game theory. As you may know, game theory studies strategic situations when there are conflicts of interest among a system of selfish players. We are interested in the setting of repeated games, in which the games are played not just once, but repeatedly, so that players will be able to adjust their plays adaptively. Nash equilibrium is a widely-adopted solution concept to predict the outcome of such a system, as it corresponds to a steady state in which the system will remain once it is reached. However, this raises the issue of how such a state can even be reached. In fact, it is now widely believed that there

is no efficient algorithm for computing a Nash equilibrium for a general game. This means that equilibria may not be reached in a reasonable amount of time in general, and the outcomes we have observed may be far out of any equilibrium, which would render the study on equilibria meaningless. To address this issue, a new line of research is to consider natural algorithms in which players have incentives to play, and to study how the system evolves according to such dynamics. One could argue that a plausible incentive for a player is to maximize his average utility (reward) over time, and hence he has an incentive to play a no-regret algorithm. We showed that for a broad class of games called congestion games, if players play a certain type of no-regret algorithms called mirror-descent algorithms, then they indeed converge to Nash equilibria quickly. Our result is sufficiently general, as the congestion games include several important games, such as the routing game, and the mirror-descent algorithms include well-known algorithms, such as the multiplicative updates algorithm and the gradient-descent algorithm. Moreover, we showed that the equilibria they converge to are good in the sense that the corresponding social welfares are, in fact, close to the optimal, the best social welfare achievable in any possible (not necessarily equilibrium) state.

To illustrate our result in a more concrete manner, let us take the routing game as an example. In this game, there is an underlying network consisting of a set of nodes connected by some set of edges. Each edge is associated with some latency function, which increases with the amount of flow passing through. There is a set of players, each having some amount of flow to be routed from some source node to some destination node, hoping to minimize the latency he would experience. However, the players have conflict of interests, as each wants to use the edges with smaller latency functions, but would not like others to do the same, as that would increase his latency. Note that the best route of a player actually depends on how other players choose their routes. Therefore, after seeing the choices of other players, some player may wish to change his choice, but if that player does so, other players may in turn find their previous routes sub-optimal and wish to change too. It is not clear if such a cascade of changes will go on forever or if the system will eventually converge. Our result shows that if each player plays the mirror-descent algorithm, the system will indeed quickly converge to a Nash equilibrium, in which no player has an incentive to change.