15-859(B) Machine Learning Theory

Learning and Game Theory

Plan for Today & Next time

- 2-player zero-sum games
- 2-player general-sum games
  - Nash equilibria
  - Correlated equilibria
- Internal/swap regret and connection to correlated equilibria
- Many-player games with structure: congestion games / exact potential games
  - Best-response dynamics
  - Price of anarchy, Price of stability

2-Player Zero-Sum games

- Two players R and C. Zero-sum means that what's good for one is bad for the other.
- Game defined by matrix with a row for each of R's options and a column for each of C's options. Matrix tells who wins how much.
  - an entry (x,y) means: x = payoff to row player, y = payoff to column player. "Zero sum" means that y = -x.
- E.g., penalty shot:

goalie (columns), shooter (rows)

	Left	Right
Left	(0,0)	(1,-1)
Right	(1,-1)	(0,0)

Game Theory terminolgy

- Rows and columns are called pure strategies.
- Randomized algs called mixed strategies.
- "Zero sum" means that game is purely competitive. (x,y) satisfies x+y=0. (Game doesn't have to be fair).

goalie (columns), shooter (rows)

	Left	Right
Left	(0,0)	(1,-1)
Right	(1,-1)	(0,0)

Minimax-optimal strategies

- Minimax optimal strategy is a (randomized) strategy that has the best guarantee on its expected gain, over choices of the opponent. [maximizes the minimum]
- I.e., the thing to play if your opponent knows you well.

goalie (columns), shooter (rows)

	Left	Right
Left	(0,0)	(1,-1)
Right	(1,-1)	(0,0)

Minimax-optimal strategies

- Can solve for minimax-optimal strategies using Linear programming
- No-regret strategies will do nearly as well or better.
- I.e., the thing to play if your opponent knows you well.

goalie (columns), shooter (rows)

	Left	Right
Left	(0,0)	(1,-1)
Right	(1,-1)	(0,0)

Minimax Theorem (von Neumann 1928)

- Every 2-player zero-sum game has a unique value V.
- Minimax optimal strategy for R guarantees R's expected gain at least V.
- Minimax optimal strategy for C guarantees C's expected loss at most V.

Existence of no-regret strategies gives one way of proving the theorem.

Can use notion of minimax optimality to explain bluffing in poker

Simplified Poker (Kuhn 1950)

- Two players A and B.
- Deck of 3 cards: 1,2,3.
- Players ante $1.
- Each player gets one card.
- A goes first. Can bet $1 or pass.
  - If A bets, B can call or fold.
  - If A passes, B can bet $1 or pass.
    - If B bets, A can call or fold.
- High card wins (if no folding). Max pot $2.

- Two players A and B. 3 cards: 1,2,3.
- Players ante $1. Each player gets one card.
- A goes first. Can bet $1 or pass.
  - If A bets, B can call or fold.
  - If A passes, B can bet $1 or pass.
    - If B bets, A can call or fold.

Writing as a Matrix Game

- For a given card, A can decide to
  - Pass but fold if B bets. [PassFold]
  - Pass but call if B bets. [PassCall]
  - Bet. [Bet]
- Similar set of choices for B.

Can look at all strategies as a big matrix…

	[FP,FP,CB]	[FP,CP,CB]	[FB,FP,CB]	[FB,CP,CB]
[PF,PF,PC]	0	0	-1/6	-1/6
[PF,PF,B]	0	1/6	-1/3	-1/6
[PF,PC,PC]	-1/6	0	0	1/6
[PF,PC,B]	-1/6	-1/6	1/6	1/6
[B,PF,PC]	-1/6	0	0	1/6
[B,PF,B]	1/6	-1/3	0	-1/2
[B,PC,PC]	1/6	-1/6	-1/6	-1/2
[B,PC,B]	0	-1/2	1/3	-1/6
	0	-1/3	1/6	-1/6

And the minimax optimal strategies are…

- A:
  - If hold 1, then 5/6 PassFold and 1/6 Bet.
  - If hold 2, then ½ PassFold and ½ PassCall.
  - If hold 3, then ½ PassCall and ½ Bet.

  Has both bluffing and underbidding…
- B:
  - If hold 1, then 2/3 FoldPass and 1/3 FoldBet.
  - If hold 2, then 2/3 FoldPass and 1/3 CallPass.
  - If hold 3, then CallBet

  Minimax value of game is -1/18 to A.

Now, to General-Sum games…

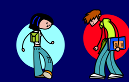
General-sum games

- In general-sum games, can get win-win and lose-lose situations.
- E.g., "what side of sidewalk to walk on?":

	Left	Right
Left	(1,1)	(-1,-1)
Right	(-1,-1)	(1,1)

you

person walking towards you



General-sum games

- In general-sum games, can get win-win and lose-lose situations.
- E.g., "which movie should we go to?":

	Peabody	Capt America
Peabody	(8,2)	(0,0)
Capt America	(0,0)	(2,8)

No longer a unique "value" to the game.

Nash Equilibrium

- A Nash Equilibrium is a stable pair of strategies (could be randomized).
- Stable means that neither player has incentive to deviate on their own.
- E.g., "what side of sidewalk to walk on":

	Left	Right
Left	(1,1)	(-1,-1)
Right	(-1,-1)	(1,1)

NE are: both left, both right, or both 50/50.

Uses

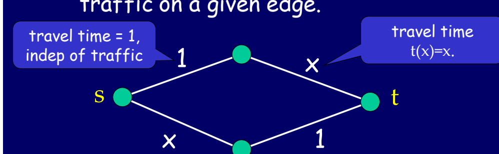
- Economists use games and equilibria as models of interaction.
- E.g., pollution / prisoner's dilemma:
  - (imagine pollution controls cost $4 but improve everyone's environment by $3)

	don't pollute	pollute
don't pollute	(2,2)	(-1,3)
pollute	(3,-1)	(0,0)

Need to add extra incentives to get good overall behavior.

NE can do strange things

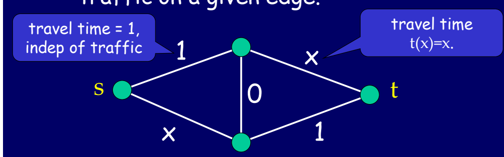
- Braess paradox:
  - Road network, traffic going from s to t.
  - travel time as function of fraction x of traffic on a given edge.



Fine. NE is 50/50. Travel time = 1.5

NE can do strange things

- Braess paradox:
  - Road network, traffic going from s to t.
  - travel time as function of fraction x of traffic on a given edge.



Add new superhighway. NE: everyone uses zig-zag path. Travel time = 2.

Existence of NE

- Nash (1950) proved: any general-sum game must have at least one such equilibrium.
  - Might require mixed strategies.
- This also yields minimax thm as a corollary.
  - Pick some NE and let V = value to row player in that equilibrium.
  - Since it's a NE, neither player can do better even knowing the (randomized) strategy their opponent is playing.
  - So, they're each playing minimax optimal.

Existence of NE in 2-player games

- Proof will be non-constructive.
- Unlike case of zero-sum games, we **do not know any** polynomial-time algorithm for finding Nash Equilibria in $n \times n$ general-sum games. [known to be "PPAD-hard"]
- Notation:
  - Assume an nxn matrix.
  - Use $(p_1,\ldots,p_n)$ to denote mixed strategy for row player, and $(q_1,\ldots,q_n)$ to denote mixed strategy for column player.

Proof

- We'll start with Brouwer's fixed point theorem.
  - Let S be a compact convex region in R^n and let $f:S \to S$ be a continuous function.
  - Then there must exist $x \in S$ such that $f(x)=x$.
  - x is called a "fixed point" of f.
- Simple case: S is the interval [0,1].
- We will care about:
  - S = {(p,q): p,q are legal probability distributions on 1,...,n}. I.e., $S = \text{simplex}_n \times \text{simplex}_n$

Proof (cont)

- S = {(p,q): p,q are mixed strategies}.
- Want to define f(p,q) = (p',q') such that:
  - f is continuous. This means that changing p or q a little bit shouldn't cause p' or q' to change a lot.
  - Any fixed point of f is a Nash Equilibrium.
- Then Brouwer will imply existence of NE.

Try #1

- What about f(p,q) = (p',q') where p' is best response to q, and q' is best response to p?
- Problem: not necessarily well-defined:
  - E.g., penalty shot: if p = (0.5,0.5) then q' could be anything.

	Left	Right
Left	(0,0)	(1,-1)
Right	(1,-1)	(0,0)

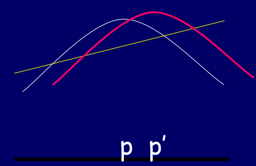
Try #1

- What about f(p,q) = (p',q') where p' is best response to q, and q' is best response to p?
- Problem: also not continuous:
  - E.g., if p = (0.51, 0.49) then q' = (1,0). If p = (0.49,0.51) then q' = (0,1).

	Left	Right
Left	(0,0)	(1,-1)
Right	(1,-1)	(0,0)

Instead we will use…

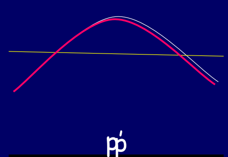
- f(p,q) = (p',q') such that:
  - q' maximizes [(expected gain wrt p) - $||q-q'||^2$]
  - p' maximizes [(expected gain wrt q) - $||p-p'||^2$]



Note: quadratic + linear = quadratic.

Instead we will use…

- f(p,q) = (p',q') such that:
  - q' maximizes [(expected gain wrt p) - $||q-q'||^2$]
  - p' maximizes [(expected gain wrt q) - $||p-p'||^2$]



Note: quadratic + linear = quadratic.

Instead we will use…

- f(p,q) = (p',q') such that:
  - q' maximizes [(expected gain wrt p) - $||q-q'||^2$]
  - p' maximizes [(expected gain wrt q) - $||p-p'||^2$]
- f is well-defined and continuous since quadratic has unique maximum and small change to p,q only moves this a little.
- Also fixed point = NE. (even if tiny incentive to move, will move little bit).
- So, that's it!

Internal/Swap Regret and Correlated Equilibria

What if all players minimize regret?

- In zero-sum games, empirical frequencies quickly approach minimax optimality.
- In general-sum games, does behavior quickly (or at all) approach a Nash equilibrium?
  - After all, a Nash Eq is exactly a set of distributions that are no-regret wrt each other. So *if the distributions stabilize*, they must converge to a Nash equil.
- Well, unfortunately, they might not stabilize.

A bad example for general-sum games

- Augmented Shapley game from [Zinkevich04]:
  - First 3 rows/cols are Shapley game (rock / paper / scissors but if both do same action then both lose).
  - 4th action "play foosball" has slight negative if other player is still doing r/p/s but positive if other player does 4th action too.

  RWM will cycle among first 3 and have no regret, but do worse than only Nash Equilibrium of both playing foosball.

- We didn't really expect this to work given how hard NE can be to find…

Another interesting bad example

- [Balcan-Constantin-Mehta12]:
  - Failure to converge even in Rank-1 games (games where R+C has rank 1).
  - Interesting because one can find equilibria efficiently in such games.

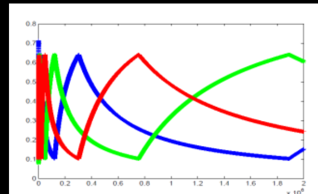


Figure 4. c_is of symmetric Shapley game with $a = 10, b = 1$

What *can* we say?

If algorithms minimize "internal" or "swap" regret, then empirical distribution of play approaches *correlated* equilibrium.

- Foster & Vohra, Hart & Mas-Colell,…
- Though doesn't imply play is stabilizing.

What are internal/swap regret and correlated equilibria?

More general forms of regret

1. "best expert" or "external" regret:
   - Given n strategies. Compete with best of them in hindsight.
2. "sleeping expert" or "regret with time-intervals":
   - Given n strategies, k properties. Let S_i be set of days satisfying property i (might overlap). Want to simultaneously achieve low regret over each S_i.
3. "internal" or "swap" regret: like (2), except that S_i = set of days in which we chose strategy i.

Internal/swap-regret

- E.g., each day we pick one stock to buy shares in.
  - Don't want to have regret of the form "every time I bought IBM, I should have bought Microsoft instead".
- Formally, swap regret is wrt optimal function f:{1,…,n}→{1,…,n} such that every time you played action j, it plays f(j).

Correlated equilibrium

- Distribution over entries in matrix, such that if a trusted party chooses one at random and tells you your part, you have no incentive to deviate.
- E.g., Shapley game.

	R	P	S
R	-1,-1	-1,1	1,-1
P	1,-1	-1,-1	-1,1
S	-1,1	1,-1	-1,-1

In general-sum games, if all players have low swap-regret, then empirical distribution of play is apx correlated equilibrium.

Connection

- If all parties run a low swap regret algorithm, then empirical distribution of play is an apx correlated equilibrium.
  - Correlator chooses random time $t \in \{1,2,\ldots,T\}$. Tells each player to play the action j they played in time t (but does not reveal value of t).
  - Expected incentive to deviate: $\sum_j \Pr(j)(\text{Regret}|j)$ = swap-regret of algorithm
  - So, this suggests correlated equilibria may be natural things to see in multi-agent systems where individuals are optimizing for themselves

Correlated vs Coarse-correlated Eq

In both cases: a distribution over entries in the matrix. Think of a third party choosing from this distr and telling you your part as "advice".

"Correlated equilibrium"

- You have no incentive to deviate, even after seeing what the advice is.

"Coarse-Correlated equilibrium"

- If only choice is to see and follow, or not to see at all, would prefer the former.

Low external-regret $\Rightarrow$ apx coarse correlated equilib.

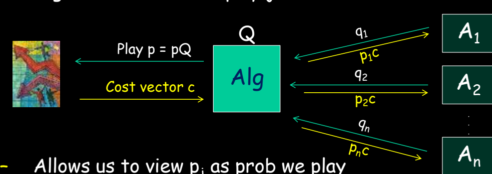
Internal/swap-regret, contd

Algorithms for achieving low regret of this form:

- Foster & Vohra, Hart & Mas-Colell, Fudenberg & Levine.
- Will present method of [BM05] showing how to convert any "best expert" algorithm into one achieving low swap regret.
- Unfortunately, #steps to achieve low swap regret is O(n log n) rather than O(log n).

Can convert any "best expert" algorithm A into one achieving low swap regret. Idea:

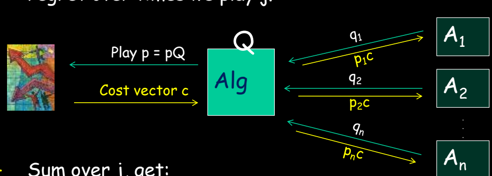
- Instantiate one copy A_j responsible for expected regret over times we play j.



- Allows us to view p_j as prob we play action j, or as prob we play alg A_j.
- Give A_j feedback of p_jc.
- A_j guarantees $\sum_t (p_j^t c^t)\cdot q_j^t \le \min_i \sum_t p_j^t c_i^t$ + [regret term]
- Write as: $\sum_t p_j^t (q_j^t \cdot c^t) \le \min_i \sum_t p_j^t c_i^t$ + [regret term]

Can convert any "best expert" algorithm A into one achieving low swap regret. Idea:

- Instantiate one copy A_j responsible for expected regret over times we play j.



- Sum over j, get:

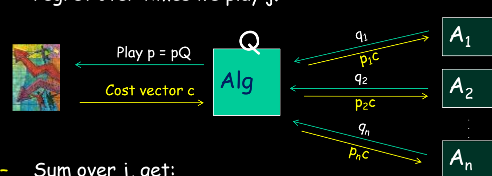
$$\sum_t p^t Q^t c^t \le \sum_j \min_i \sum_t p_j^t c_i^t + n[\text{regret term}]$$

Our total cost | For each j, can move our prob to its own i=f(j)

- Write as: $\sum_t p_j^t (q_j^t \cdot c^t) \le \min_i \sum_t p_j^t c_i^t$ + [regret term]

Can convert any "best expert" algorithm A into one achieving low swap regret. Idea:

- Instantiate one copy A_j responsible for expected regret over times we play j.



- Sum over j, get:

$$\sum_t p^t Q^t c^t \le \sum_j \min_i \sum_t p_j^t c_i^t + n[\text{regret term}]$$

Our total cost | For each j, can move our prob to its own i=f(j)

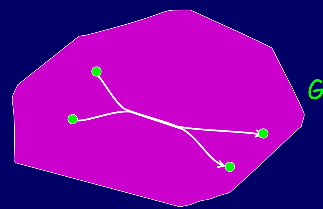
- Get swap-regret at most n times orig external regret.

Congestion games

- Many multi-agent interactions have structure. One nice class: Congestion Games
- Always have a pure-strategy equilibrium.
- Have a potential function s.t. whenever a player switches, potential drops by exactly that player's improvement.
  - So, best-response dynamics always gives an equilibrium.
  - Can view as positive statement about very simple learning rule.
- Let's start with an example.

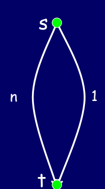
Fair cost-sharing

Fair cost-sharing: n players in weighted directed graph G. Player i wants to get from s_i to t_i, and they share cost of edges they use with others.



Good equilibria, Bad equilibria

Fair cost-sharing: n players in weighted directed graph G. Player i wants to get from s_i to t_i, and they share cost of edges they use with others.



Good equilibrium: all use edge of cost 1. (cost 1/n per player)

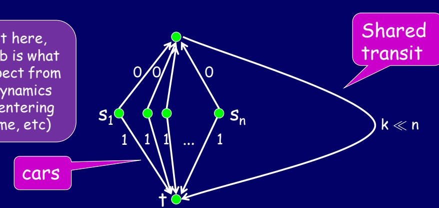
Bad equilibrium: all use edge of cost n. (cost 1 per player)

Cost(bad equilib) = n·Cost(good equilib)

Good equilibria, Bad equilibria

Fair cost-sharing: n players in weighted directed graph G. Player i wants to get from s_i to t_i, and they share cost of edges they use with others.

Note that here, bad equilb is what you'd expect from natural dynamics (players entering one at time, etc)



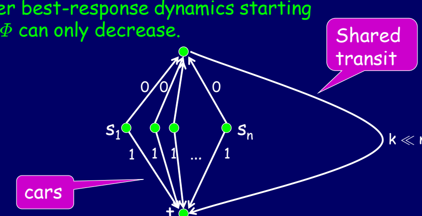
Price of Anarchy and Price of Stability

- Price of Anarchy: ratio of worst equilibrium to social optimum. (worst-case over games in class)
  - We saw for cost-sharing PoA = $\Omega(n)$. Also O(n).
- Price of Stability: ratio of best equilibrium to social optimum. (worst-case over games in class)
  - For cost-sharing, PoS = $\Theta(\log n)$.
- Exact Potential function: Function Φ s.t. if player moves, potential changes by exactly as much as cost of player who moved.
  - Guarantees that best-response dynamics will reach Nash equilibrium

Potential functions and PoS

For cost-sharing, PoS = O(log n):

- Given state S, let n_e = # players on edge e. Cost(S) = $\sum_{e:n_e\geq 1} c_e$
- Define potential $\Phi(S) = \sum_e \sum_{i=1}^{n_e} c_e/i$
- So, cost(S) $\leq \Phi(S) \leq \log(n) \times$ cost(S).
- Now consider best-response dynamics starting from OPT. Φ can only decrease.



Congestion games more generally

Game defined by n players and m resources.

- Each player i choses a set of resources (e.g., a path) from collection S_i of allowable sets of resources (e.g., paths from s_i to t_i).
- Cost of a resource j is a function $f_j(n_j)$ of the number n_j of players using it.
- Cost incurred by player i is the sum, over all resources being used, of the cost of the resource.
- Generic potential function: $\sum_j \sum_{i=1}^{n_j} f_j(i)$
- Best-response dynamics may take a long time to reach equil, but if gap between Φ and cost is small, can get to apx-equilib (additive appropximation) fast.

Current/recent research directions

(esp in relation to machine learning)

- Are there natural dynamics that can manage to reach good equilibria on their own?
- Can one say anything interesting about "combining expert advice" types of problems where the quality of an expert depends on what the other players are doing? (In particular, in comparison to best equilibrium)