Topics in Machine Learning Theory

The Adversarial Multi-armed Bandit Problem, Internal Regret, and Correlated Equilibria

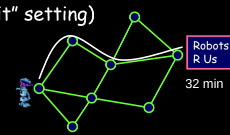
Avrim Blum

10/8/14

Plan for today

Online game playing / combining expert advice but:

- What if we only get feedback for the action we chose? (called the "multi-armed bandit" setting)



- What about stronger forms of regret-minimization (internal regret)?
- Connection to notion of "correlated equilibria"
- But first, a quick discussion of [0,1] vs {0,1} costs for RWM algorithm

[0,1] costs vs {0,1} costs.

We analyzed Randomized Wtd Majority for case that all costs in {0,1} (and slightly hand-waved extension to [0,1])

Here is an alternative simple way to extend to [0,1].

- Given cost vector c, view c_i as bias of coin. Flip to create vector $c' \in \{0,1\}^n$, s.t. $E[c'_i] = c_i$. Feed c' to alg A.

world $\xrightarrow{c}$ \$ $\xrightarrow{c'}$ A

- For any sequence of vectors $c' \in \{0,1\}^n$, we have:
  - $E_A[cost'(A)] \leq \min_i cost'(i) + [\text{regret term}]$ (Cost' = cost on c' vectors)
- So, $E_\$[E_A[cost'(A)]] \leq E_\$[\min_i cost'(i)] + [\text{regret term}]$
- LHS is $E_A[cost(A)]$. (since $E_\$[E_A[cost'(A)]] = E_\$[c' \cdot \vec{p}] = c \cdot \vec{p}$)
- RHS $\leq \min_i E_\$[cost'(i)] + [r.t.] = \min_i[cost(i)] + [r.t.]$

In other words, costs between 0 and 1 just make the problem easier…

Experts → Bandit setting

- In the bandit setting, only get feedback for the action we choose. Still want to compete with best action in hindsight.
- [ACFS02] give algorithm with cumulative regret $O((TN \log N)^{1/2})$. [average regret $O(((N \log N)/T)^{1/2})$.]
- Will do a somewhat weaker version of their analysis (same algorithm but not as tight a bound).
- Talk about it in the context of online pricing…

Online pricing

- Say you are selling lemonade (or a cool new software tool, or bottles of water at the world cup).
- For t=1,2,…T
  - Seller sets price p^t
  - Buyer arrives with valuation v^t
  - If $v^t \geq p^t$, buyer purchases and pays p^t, else doesn't.
  - Repeat.

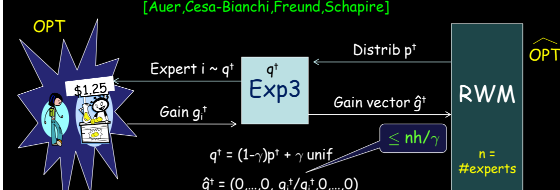
View each possible price as a different row/expert

- Assume all valuations $\leq$ h.
- Goal: do nearly as well as best fixed price in hindsight.
- If v^t revealed, run RWM. $E[gain] \geq OPT(1-\epsilon) - O(\epsilon^{-1} h \log n)$.

Multi-armed bandit problem

Exponential Weights for Exploration and Exploitation (exp[3])

[Auer,Cesa-Bianchi,Freund,Schapire]

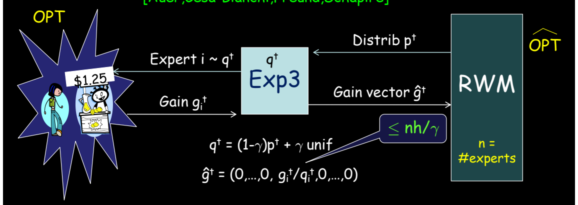


1. RWM believes gain is: $p^t \cdot \hat{g}^t = p_i^t(g_i^t/q_i^t) \equiv g^t_{RWM}$
2. $\sum_t g^t_{RWM} \geq \widehat{OPT}(1-\epsilon) - O(\epsilon^{-1}\, nh/\gamma \log n)$
3. Actual gain is: $g_i^t = g^t_{RWM}(q_i^t/p_i^t) \geq g^t_{RWM}(1-\gamma)$
4. $E[\widehat{OPT}] \geq OPT$. Because $E[\hat{g}_j^t] = (1- q_j^t)0 + q_j^t(g_j^t/q_j^t) = g_j^t$, so $E[\max_j[\sum_t \hat{g}_j^t]] \geq \max_j [E[\sum_t \hat{g}_j^t]] = OPT$.

Multi-armed bandit problem

Exponential Weights for Exploration and Exploitation (exp^3)

[Auer,Cesa-Bianchi,Freund,Schapire]



Conclusion ($\gamma = \epsilon$):

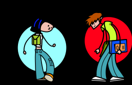
$E[Exp3] \geq OPT(1-\epsilon)^2 - O(\epsilon^{-2}\, nh \log(n))$

Balancing would give $O((OPT\ nh \log n)^{2/3})$ in bound because of ϵ^{-2}. But can reduce to ϵ^{-1} and $O((OPT\ nh \log n)^{1/2})$ with better analysis.

General-sum games

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

you \ person walking towards you	Left	Right
Left	(1,1)	(-1,-1)
Right	(-1,-1)	(1,1)



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)

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)

Existence of NE

- Nash (1950) proved: any general-sum game must have at least one such equilibrium.
  - Might require mixed strategies.
  - Proof is non-constructive.
  - Finding Nash equilibria in general appears to be hard (is PPAD-hard).

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.

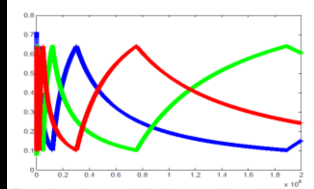


Figure 4. c_is of symmetric Shapley game with $a = 10, b = 1$

Internal/Swap Regret and Correlated Equilibria

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 $f:\{1,…,n\}\rightarrow\{1,…,n\}$ 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.

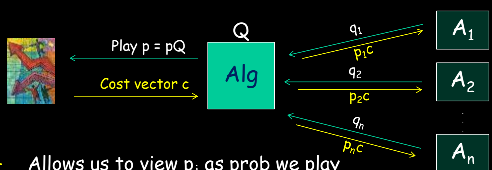
Internal/swap-regret, contd

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:

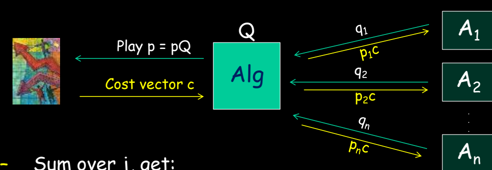
- Instantiate one copy A_j responsible for expected regret over times we play j.



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

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

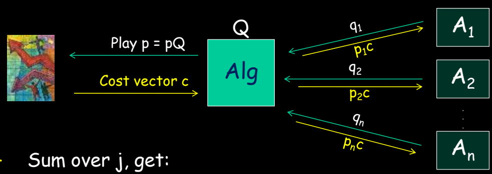
Our total cost

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

- Write as: $\sum_t p_j^t (q_j^t \cdot c^t) \leq \min_i \sum_t p_j^t c_i^t$ + [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.