Bandit algorithms, internal & swap regret, and correlated equilibria

Your guide:
Avrim Blum

[Readings: Ch. 4.4-4.6 of AGT book]
"No-regret" algorithms for repeated decisions:

- Algorithm has N options. World chooses cost vector. *Can view as matrix like this (maybe infinite # cols)*

- At each time step, algorithm picks row, life picks column.
  - Alg pays cost (or gets benefit) for action chosen.
  - Alg gets column as feedback (or just its own cost/benefit in the "bandit" model).
  - Goal: do nearly as well as best fixed row in hindsight.

Recap
Guarantee: $E[\text{cost}] \leq \text{OPT} + 2(\text{OPT} \cdot \log n)^{1/2}$

Since $\text{OPT} \leq T$, this is at most $\text{OPT} + 2(T\log n)^{1/2}$.

So, regret/time step $\leq 2(T\log n)^{1/2}/T \to 0$. 
[ACFS02]: applying RWM to bandit setting

- What if only get your own cost/benefit as feedback?

- Use of RWM as subroutine to get algorithm with cumulative regret $O(\sqrt{TN \log N})$.
  
  \[ \text{[average regret } O(\sqrt{\frac{N \log N}{T}}) \text{].} \]

- Will do a somewhat weaker version of their analysis (same algorithm but not as tight a bound).

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

$2$

$3.00$ a glass

View each possible price as a different row/expert
Multi-armed bandit problem

Exponential Weights for Exploration and Exploitation (exp³)

[Auer, Cesa-Bianchi, Freund, Schapire]

1. RWM believes gain is: \( p^\dagger \cdot \hat{g}^\dagger = p_i^\dagger (\frac{g_i^\dagger}{q_i^\dagger}) \equiv g_{RWM}^\dagger \)
2. \( \sum_t g_{RWM}^\dagger \geq \hat{\text{OPT}} (1-\epsilon) - O(\epsilon^{-1} \frac{nh}{\gamma} \log n) \)
3. Actual gain is: \( g_i^\dagger = g_{RWM}^\dagger \left( \frac{q_i^\dagger}{p_i^\dagger} \right) \geq g_{RWM}^\dagger (1-\gamma) \)
4. \( \hat{\text{OPT}} \geq \text{OPT} \). Because \( E[\hat{g}_j^\dagger] = (1- q_j^\dagger)0 + q_j^\dagger (\frac{g_j^\dagger}{q_j^\dagger}) = g_j^\dagger \),
   so \( E[\max_j [\sum_t \hat{g}_j^\dagger]] \geq \max_j [ E[\sum_t \hat{g}_j^\dagger] ] = \text{OPT} \).
Multi-armed bandit problem

Exponential Weights for Exploration and Exploitation ($\exp^3$)  
[Auer, Cesa-Bianchi, Freund, Schapire]

$\text{Exp3}$  
$q^\dagger = (1-\gamma)p^\dagger + \gamma \text{ unif}$  
$\hat{g}^\dagger = (0, \ldots, 0, g_i^\dagger / q_i^\dagger, 0, \ldots, 0)$

Conclusion ($\gamma = \epsilon$):  
$E[\text{Exp3}] \geq \text{OPT}(1-\epsilon)^2 - O(\epsilon^{-2} \text{ nh log}(n))$

Balancing would give $O((\text{OPT nh log n})^{2/3})$ in bound because of $\epsilon^{-2}$. But can reduce to $\epsilon^{-1}$ and $O((\text{OPT nh log n})^{1/2})$ more care in analysis.
Summary

Algorithms for online decision-making with strong guarantees on performance compared to best fixed choice.

• Application: play repeated game against adversary. Perform nearly as well as fixed strategy in hindsight.

Can apply even with very limited feedback.

• Application: which way to drive to work, with only feedback about your own paths; online pricing, even if only have buy/no buy feedback.
Internal/Swap Regret and Correlated Equilibria
What if all players minimize regret?

- In zero-sum games, empirical frequencies quickly approaches minimax optimal.
- 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, no.
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).
  - $4^{th}$ action “play foosball” has slight negative if other player is still doing r/p/s but positive if other player does $4^{th}$ 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...**
A bad example for general-sum games

- [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. $c_i$'s 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 $S_i = \text{set of days in which we chose strategy } i$. 
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 AT&T, I should have bought Microsoft instead”.

- Formally, swap regret is wrt optimal function $f: \{1, \ldots, n\} \rightarrow \{1, \ldots, n\}$ such that every time you played action $j$, it plays $f(j)$.

- So, competing with the best of these $n^n$ “rewiring” functions.
Formally

- Let \( c^t \) denote the cost vector (loss vector) at time \( t \).
- The algorithm’s total expected cost (loss) is:
  \[
  \sum_t p^t \cdot c^t = \sum_t \sum_j p_j c_j^t.
  \]
- For standard external regret, we are comparing this to the cost (loss) of the best action in hindsight:
  \[
  \min_i \sum_t c_i^t.
  \]
- For swap regret, we compare to the best rewiring of our probability mass:
  \[
  \min_f \sum_t \sum_j p_j^t c_{f(j)}^t = \sum_j \min_i \sum_t p_j^t c_i^t.
  \]
- In other words, our probability mass on action \( j \) gets rewired to action \( i = f(j) \).

Note: if you replace the \( \Sigma_j \min_i \) with \( \min_i \Sigma_j \) then you get back to external regret.
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.

<table>
<thead>
<tr>
<th></th>
<th>R</th>
<th>P</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>-1,-1</td>
<td>-1,1</td>
<td>1,-1</td>
</tr>
<tr>
<td>P</td>
<td>1,-1</td>
<td>-1,-1</td>
<td>-1,1</td>
</tr>
<tr>
<td>S</td>
<td>-1,1</td>
<td>1,-1</td>
<td>-1,-1</td>
</tr>
</tbody>
</table>
Correlated equilibrium

• Can solve for CEQ using linear programming.

• Solve for $D_{ij} \geq 0, \sum_{ij} D_{ij} = 1$, such that:

  For all $i, i'$,  \[
  \sum_j D_{ij} R_{ij} \geq \sum_j D_{ij} R_{ij'} 
  \]
  [Conceptually, divide LHS and RHS by $\sum_j D_{ij}$]

  For all $j, j'$,  \[
  \sum_i D_{ij} C_{ij} \geq \sum_i D_{ij} C_{ij'}
  \]
  [Conceptually, divide LHS and RHS by $\sum_i D_{ij}$]

(E.g., Google maps tells each person what route to take, and it’s a CEQ if nobody has any incentive to deviate)

• Can’t do for Nash since replacing $D_{ij}$ with $p_i q_j$ makes quadratic.
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 $t$).

- If each player had no swap regret, then no matter what action $j$ they are told to play, they will not have any incentive to deviate $\Rightarrow$ correlated equilibrium.

- Expected incentive to deviate: $\Sigma_j Pr(j)(\text{Regret}|j) = \text{swap-regret experienced}$. 
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 $\sum_t (p_j^t c^t) q_j^t \leq \min_i \sum_t p_j^t c_i^t + \text{[regret term]}$

- Write as: $\sum_t p_j^t (q_j^t c^t) \leq \min_i \sum_t p_j^t c_i^t + \text{[regret term]}$
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 $j$.

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

- Sum over $j$, get:

- Write as:

\[ \sum_t p_j^t (q_j^t \cdot c^t) \leq \min_i \sum_t p_j^t c_i^t + [\text{regret term}] \]
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 $i=f(j)$

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