Graduate AI

Lecture 21:
Game Theory IV

Teachers:
Zico Kolter
Ariel Procaccia (this time)
Reminder: The Minimax Theorem

- Theorem [von Neumann, 1928]:
  Every 2-player zero-sum game has a unique value $v$ such that:
  - Player 1 can guarantee value at least $v$
  - Player 2 can guarantee loss at most $v$

- We will prove the theorem via no-regret learning
How to reach your spaceship

- Each morning pick one of $n$ possible routes
- Then find out how long each route took
- Is there a strategy for picking routes that does almost as well as the best fixed route in hindsight?

53 minutes
47 minutes
...

15780 Spring 2017: Lecture 21
The model

- View as a matrix (maybe infinite columns)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Adversary</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Algorithm picks row, adversary column
- Alg pays cost of (row,column) and gets column as feedback
- Assume costs are in \([0,1]\)
THE MODEL

• Define average regret in $T$ time steps as (average per-day cost of alg) − (average per-day cost of best fixed row in hindsight)
• No-regret algorithm: regret $\to 0$ as $T \to \infty$
• Not competing with adaptive strategy, just the best fixed row
**Example**

- **Algorithm 1**: Alternate between U and D
- **Poll 1**: What is algorithm 1’s worst-case average regret?
  1. $\Theta(1/T)$
  2. $\Theta(1)$
  3. $\Theta(T)$
  4. $\infty$

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Adversary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
EXAMPLE

- **Algorithm 2**: Choose action that has lower cost so far
- **Poll 2**: What is algorithm 2’s worst-case average regret?
  1. $\Theta(1/T)$
  2. $\Theta(1/\sqrt{T})$
  3. $\Theta(1/\log T)$
  4. $\Theta(1)$

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Adversary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
What can we say more generally about deterministic algorithms?
**Using expert advice**

- Want to predict the stock market
- Solicit advice from $n$ experts
  - **Expert** = someone with an opinion

<table>
<thead>
<tr>
<th>Day</th>
<th>Expert 1</th>
<th>Expert 2</th>
<th>Expert 3</th>
<th>Charlie</th>
<th>Truth</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>−</td>
<td>−</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>2</td>
<td>+</td>
<td>−</td>
<td>+</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

- Can we do as well as best in hindsight?
**Simpler Question**

- One of the $n$ experts never makes a mistake
- We want to find out which one
- **Algorithm 3:** Take majority vote over experts that have been correct so far
- **Poll 3:** What is algorithm 3’s worst-case number of mistakes?
  1. $\Theta(1)$
  2. $\Theta(\log n)$
  3. $\Theta(n)$
  4. $\Theta(2^n)$
WHAT IF NO EXPERT IS PERFECT?

• Idea: Run algorithm 3 until all experts are crossed off, then repeat
• Makes at most $\log n$ mistakes per mistake of the best expert
• But this is wasteful: we keep forgetting what we’ve learned
Reprise:
Algorithms that forget their history are doomed to repeat it!
WEIGHTED MAJORITY

• **Intuition:** Making a mistake doesn’t disqualify an expert, just lowers its weight

• **Weighted Majority Algorithm:**
  - Start with all experts having weight 1
  - Predict based on weighted majority vote
  - Penalize mistakes by cutting weight in half
<table>
<thead>
<tr>
<th></th>
<th>Expert 1</th>
<th>Expert 2</th>
<th>Expert 3</th>
<th>Charlie</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Weights</strong></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td><strong>Prediction</strong></td>
<td>−</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td><strong>Weights</strong></td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td><strong>Prediction</strong></td>
<td>+</td>
<td>+</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td><strong>Weights</strong></td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

**Alg** | **Truth**
---|---
+ | +
- | +
**Weighted Majority: Analysis**

- $M = \#$mistakes we’ve made so far
- $m = \#$mistakes of best expert so far
- $W = \text{total weight (starts at } n\text{)}$
- For each mistake, $W$ drops by at least 25%  
  $\Rightarrow$ after $M$ mistakes: $W \leq n(3/4)^M$
- Weight of best expert is $(1/2)^m$

\[
(\frac{1}{2})^m \leq n \left(\frac{3}{4}\right)^M \Rightarrow \left(\frac{4}{3}\right)^M \leq n 2^m \Rightarrow M \leq 2.5(m + \lg n)
\]
Randomized Weighted Majority

• Randomized Weighted Majority Algorithm:
  o Start with all experts having weight 1
  o Predict proportionally to weights: the total weight of + is $w_+$ and the total weight of $-$ is $w_-$, predict + with probability $\frac{w_+}{w_++w_-}$ and $-$ with probability $\frac{w_-}{w_++w_-}$
  o Penalize mistakes by removing $\epsilon$ fraction of weight
Randomized Weighted Majority

Idea: smooth out the worst case

The worst-case is ~50-50: now we have a 50% chance of getting it right

What about 90-10? We’re very likely to agree with the majority
**Analysis**

- At time $t$ we have a fraction $F_t$ of weight on experts that made a mistake
- Prob. $F_t$ of making a mistake, remove $\epsilon F_t$ fraction of total weight
- $W_{final} = n \prod_t (1 - \epsilon F_t)$
- $\ln W_{final} = \ln n + \sum_t \ln (1 - \epsilon F_t)$
  $\leq \ln n - \epsilon \sum_t F_t = \ln n - \epsilon M$

$\ln(1 - x) \leq -x$
(next slide)
\[ f(x) = \ln(1 - x) \]
\[ f(x) = -x \]
**Analysis**

- Weight of best expert is $W_{\text{best}} = (1 - \epsilon)^m$
- $\ln n - \epsilon M \geq \ln W_{\text{final}} \geq \ln W_{\text{best}} = m \ln (1 - \epsilon)$
- By setting $\epsilon = \sqrt{\ln n/m}$ and solving, we get $M \leq m + 2\sqrt{m \ln n}$
- Since $m \leq T$, $M \leq m + 2\sqrt{T \ln n}$
- Average regret is $\left(2\sqrt{T \ln n}\right)/T \to 0$
MORE GENERALLY

• Each expert is an action with cost in $[0,1]$
• Run Randomized Weighted Majority
  ○ Choose expert $i$ with probability $w_i/W$
  ○ Update weights: $w_i \leftarrow w_i(1 - c_i\varepsilon)$
• Same analysis applies:
  ○ Our expected cost: $\sum_j c_jw_j/W$
  ○ Fraction of weight removed: $\varepsilon \sum_j c_jw_j/W$
  ○ So, fraction removed $= \varepsilon \cdot (\text{our cost})$
Proof of the Minimax THM

• Suppose for contradiction that zero-sum game $G$ has $V_C > V_R$ such that:
  - If column player commits first, there is a row that guarantees row player at least $V_C$
  - If row player commits first, there is a column that guarantees row player at most $V_R$

• Scale matrix so that payoffs to row player are in $[-1,0]$, and let $V_C = V_R + \delta$
Proof of the Minimax Thm

• Row player plays RWM, and column player responds optimally to current mixed strategy
• After $T$ steps
  • $\text{ALG} \geq \text{best row in hindsight} - 2\sqrt{T \log n}$
  • $\text{ALG} \leq T \cdot V_R$
  • Best row in hindsight $\geq T \cdot V_C$
• It follows that $T \cdot V_R \geq T \cdot V_C - 2\sqrt{T \log n}$
• $\delta T \leq 2\sqrt{T \log n} - \text{contradiction for large enough } T$
SUMMARY

• Terminology:
  o Regret
  o No-regret learning

• Algorithms:
  o Randomized weighted majority

• Big ideas:
  o It is possible to achieve no-regret learning guarantees!
  o Connections between game theory and learning theory