An Intro to Game Theory

15-451  Avrim Blum  12/02/03

Plan for Today

- 2-Player Zero-Sum Games (matrix games)
  - Minimax optimal strategies

- General-Sum Games (bimatrix games)
  - notion of Nash Equilibrium

- Proof of existence of Nash Equilibria
  - using Brouwer’s fixed-point theorem

- do FCEs at end...

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.
- E.g., matching pennies / penalty shot / hide-a-coin:

```
<table>
<thead>
<tr>
<th></th>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>(-1,1)</td>
<td>(1,-1)</td>
</tr>
<tr>
<td>Right</td>
<td>(1,1)</td>
<td>(-1,1)</td>
</tr>
</tbody>
</table>
```

Minimax-optimal strategies

- Minimax optimal strategy is a (randomized) strategy that has the best worst-case expected gain. [maximizes the minimum]
- I.e., it’s the thing to play if your opponent knows you well.
- Same as our notion of a randomized strategy with a good worst-case bound.

An algorithmic example

Sorting three items (A,B,C):

- Compare two of them. Then compare 3rd to larger of 1st two. If we’re lucky it’s larger, else need one more comparison.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>adversary</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A,B) first</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>(A,C) first</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>(B,C) first</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

Minimax-optimal strategies

Sorting three items (A,B,C): Compare two of them. Then compare 3rd to larger of 1st two. Minimax optimal cost is 2*(2/3).

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>adversary</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A,B) first</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>(A,C) first</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>(B,C) first</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
Minimax-optimal strategies

- E.g., matching pennies / penalty shot / hide-a-coin

<table>
<thead>
<tr>
<th></th>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>(-1,1)</td>
<td>(1,-1)</td>
</tr>
<tr>
<td>Right</td>
<td>(1,-1)</td>
<td>(-1,1)</td>
</tr>
</tbody>
</table>

Minimax optimal for both players is 50/50. Gives expected gain of 0. Any other is worse.

Minimax optimal strategies

- E.g., penalty shot with goalie who’s weaker on the left.

<table>
<thead>
<tr>
<th></th>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>(0,0)</td>
<td>(1,-1)</td>
</tr>
<tr>
<td>Right</td>
<td>(1,-1)</td>
<td>(-1,1)</td>
</tr>
</tbody>
</table>

Minimax optimal for both players is (2/3,1/3). Gives expected gain 1/3. Any other is worse.

Minimax Theorem (von Neumann 1928)

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

Counterintuitive: against an optimal opponent, it doesn’t hurt to reveal your randomized strategy. (Borel had proved for symmetric 5x5 but thought was false for larger games)

Matrix games and Algorithms

- Gives a useful way of thinking about guarantees on algorithms.
- Think of rows as different algorithms, columns as different possible inputs.
- $M(i,j) =$ cost of algorithm $i$ on input $j$.

Of course matrix is HUGE. But helpful conceptually.

Matrix games and Algs

- What is a deterministic alg with a good worst-case guarantee?
  - A row that does well against all columns
- What is a lower bound for deterministic algorithms?
  - Showing that for each row $i$ there exists a column $j$ such that $M(i,j)$ is bad
- How to give lower bound for randomized alg?
  - Give randomized strategy for adversary that is bad for all $i$.

E.g., hashing

- Rows are different hash functions.
- Cols are different sets of items to hash.
- $M(i,j) =$ #collisions incurred by alg $i$ on set $j$.
  [alg is trying to minimize]
- For any row, can reverse-engineer a bad column.
- Universal hashing is a randomized strategy for row player.
**One more example**

1-card poker in a 3-card deck (J,Q,K):

- [PF,PF,PC] [FP,CP,CB] [FB,FP,CB] [FB,CP,CB]
- [PF,PF,B] [PF,PC,PC] [PF,PC,B] [B,PF,PC]
- [B,PF,B] [B,PC,PC] [B,PC,B]

---

**Minimax-optimal strategy**

- Minimax optimal for 1st player is:
  - If hold J, then 5/6 PassFold and 1/6 Bet.
  - If hold Q, then 1/3 PassFold and 2/3 PassCall.
  - If hold K, then 1/3 PassCall and 2/3 Bet.
- Note the bluffing and underbidding...
  (Minimax for 2nd player has this too)
- Minimax value of game is -1/18 for 1st player and 1/18 for 2nd.
  (Remember can solve for minimax with LP)

---

**General-sum games**

- In general-sum (bimatrix) games, can have win-win and lose-lose situations.
- E.g., "what side of road to drive on?":

<table>
<thead>
<tr>
<th></th>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>(1,1)</td>
<td>(-1,-1)</td>
</tr>
<tr>
<td>Right</td>
<td>(-1,1)</td>
<td>(1,1)</td>
</tr>
</tbody>
</table>

---

**General-sum games**

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

<table>
<thead>
<tr>
<th></th>
<th>loveactually</th>
<th>MatRev</th>
</tr>
</thead>
<tbody>
<tr>
<td>loveactually</td>
<td>(0,0)</td>
<td>(2,8)</td>
</tr>
<tr>
<td>MatRev</td>
<td>(8,2)</td>
<td>(0,0)</td>
</tr>
</tbody>
</table>

No longer a unique "value" to the game.

---

**General-sum games**

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

<table>
<thead>
<tr>
<th></th>
<th>pollute</th>
<th>don't pollute</th>
</tr>
</thead>
<tbody>
<tr>
<td>pollute</td>
<td>(-1,-1)</td>
<td>(2,2)</td>
</tr>
<tr>
<td>don't pollute</td>
<td>(-2,2)</td>
<td>(1,1)</td>
</tr>
</tbody>
</table>

Need to add extra incentives to get desired behavior.

---

**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 road to drive on”:

<table>
<thead>
<tr>
<th></th>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>(1,1)</td>
<td>(-1,-1)</td>
</tr>
<tr>
<td>Right</td>
<td>(-1,1)</td>
<td>(1,1)</td>
</tr>
</tbody>
</table>

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

- A Nash Equilibrium is a stable pair of strategies (could be randomized).
- Stable means that neither player has incentive to deviate.
- E.g., "which movie to go to":

<table>
<thead>
<tr>
<th></th>
<th>MatRev</th>
<th>loveactually</th>
</tr>
</thead>
<tbody>
<tr>
<td>MatRev</td>
<td>(8,2)</td>
<td>(0,0)</td>
</tr>
<tr>
<td>loveactually</td>
<td>(0,0)</td>
<td>(2,8)</td>
</tr>
</tbody>
</table>

NE are: both MR, both la, or (80/20,20/80)

Existence of NE

- Nash (1950) proved: any general-sum game must have at least one such equilibrium.
  - Might require randomized strategies (called "mixed strategies")
  - This also yields minimax thm as a corollary.
  - Pick some NE and let $V$ be 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.

Proof

- We’ll start with Brouwer’s fixed point theorem.
  - Let $S$ be a compact convex region in $\mathbb{R}^n$ 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 $1,\ldots,n\}$. 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.

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 continuous:
  - E.g., matching pennies. If $p = (0.51, 0.49)$ then $q' = (1,0)$. If $p = (0.49,0.51)$ then $q = (0,1)$.

<table>
<thead>
<tr>
<th></th>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>(-1,-1)</td>
<td>(1,-1)</td>
</tr>
<tr>
<td>Right</td>
<td>(1,-1)</td>
<td>(-1,-1)</td>
</tr>
</tbody>
</table>
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 necessarily well-defined:
  - Eg., if \( p = (0.5,0.5) \) then \( q' \) could be anything.

<table>
<thead>
<tr>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-1,1)</td>
<td>(1,-1)</td>
</tr>
</tbody>
</table>

Instead we will use...
- \( f(p,q) = (p',q') \) such that:
  - \( q' \) maximizes \( [[\text{expected gain wrt } p] - |q-q'|^2] \)
  - \( p' \) maximizes \( [[\text{expected gain wrt } q] - |p-p'|^2] \)

\[ p \quad p' \]

Note: quadratic + linear = quadratic.

Instead we will use...
- \( f(p,q) = (p',q') \) such that:
  - \( q' \) maximizes \( [[\text{expected gain wrt } p] - |q-q'|^2] \)
  - \( p' \) maximizes \( [[\text{expected gain wrt } q] - |p-p'|^2] \)

\[ p \quad p' \]

Note: quadratic + linear = quadratic.

Instead we will use...
- \( f(p,q) = (p',q') \) such that:
  - \( q' \) maximizes \( [[\text{expected gain wrt } p] - |q-q'|^2] \)
  - \( p' \) maximizes \( [[\text{expected gain wrt } q] - |p-p'|^2] \)
- \( f \) 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!