Shall we play a game?

Game Theory and Computer Science

- Zero-sum games
- General-sum games
Plan for Today

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

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

- Proof of existence of Nash Equilibria
  - using Brouwer’s fixed-point theorem
Consider the following scenario...

- Shooter has a penalty shot. Can choose to shoot left or shoot right.

- Goalie can choose to dive left or dive right.

- If goalie guesses correctly, (s)he saves the day. If not, it’s a gooooooaaaaaaaall!

- Vice-versa for shooter.
## 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:

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

- **Shoot**
  - **Goalie**
    - **GOAALLL!!**
  - **No goal**
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.

\[
\begin{array}{c|cc}
 & \text{Left} & \text{Right} \\
\hline
\text{Left} & (0,0) & (1,-1) \\
\text{Right} & (1,-1) & (0,0) \\
\end{array}
\]
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.

• In class on Linear Programming, we saw how to solve for this using LP.
  - polynomial time in size of matrix if use poly-time LP alg.
Minimax-optimal strategies

- E.g., penalty shot:

<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>(0,0)</td>
</tr>
</tbody>
</table>

Minimax optimal strategy for both players is 50/50. Gives expected gain of \(\frac{1}{2}\) for shooter (-\(\frac{1}{2}\) for goalie). 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>$(\frac{1}{2},-\frac{1}{2})$</td>
<td>$(1,-1)$</td>
</tr>
<tr>
<td>Right</td>
<td>$(1,-1)$</td>
<td>$(0,0)$</td>
</tr>
</tbody>
</table>

Minimax optimal for shooter is $(2/3,1/3)$. Guarantees expected gain at least $2/3$. Minimax optimal for goalie is also $(2/3,1/3)$. Guarantees expected loss at most $2/3$. 
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 $V$.

**Counterintuitive:** Means it doesn’t hurt to publish your strategy if both players are optimal. (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 for a given problem.

- Think of rows as different algorithms, columns as different possible inputs.

- \( M(i,j) = \text{cost of algorithm } i \text{ on input } j. \)

- Algorithm design goal: good strategy for row player. Lower bound: good strategy for adversary.

One way to think of upper-bounds/lower-bounds: on value of this game
**Matrix games and Algorithms**

- Gives a useful way of thinking about guarantees on algorithms for a given problem.

- Think of rows as different algorithms, columns as different possible inputs.

- $M(i,j) = \text{cost of algorithm } i \text{ on input } j$.

- Algorithm design goal: good strategy for row player. Lower bound: good strategy for adversary.

*Of course matrix may be 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 algs?
  • Give randomized strategy for adversary that is bad for all $i$. Must also be bad for all distributions over $i$. 
E.g., hashing

- Rows are different hash functions.
- Cols are different sets of n items to hash.
- \( M(i,j) = \# \text{collisions incurred by alg } i \text{ on set } j \).
  
  [alg is trying to minimize]

- For any row, can reverse-engineer a bad column.

- Universal hashing is a randomized strategy for row player that has good behavior for every column.
  - For any set of inputs, if you randomly construct hash function in this way, you won't get many collisions in expectation.
Nice proof of minimax thm (sketch)

• Suppose for contradiction it was false.
• This means some game $G$ has $V_C > V_R$:
  - If Column player commits first, there exists a row that gets at least $V_C$.
  - But if Row player has to commit first, the Column player can make him get only $V_R$.
• Scale matrix so payoffs to row are in $[0,1]$. Say $V_R = V_C(1-\varepsilon)$. 

\[ V_C \]
\[ V_R \]
Proof sketch, contd

- Consider repeatedly playing game $G$ against some opponent. [think of you as row player]
- Use “picking a winner / expert advice” alg to do nearly as well as best fixed row in hindsight.
  - Alg gets $(1-\varepsilon/2)OPT - c\times \log(n)/\varepsilon > (1-\varepsilon)OPT$ [if play long enough]
  - $OPT \geq VC$ [Best against opponent’s empirical distribution]
  - $Alg \cdot VR$ [Each time, opponent knows your randomized strategy]
  - Contradicts assumption.
General-Sum Games

- Zero-sum games are good formalism for design/analysis of algorithms.
- General-sum games are good models for systems with many participants whose behavior affects each other’s interests
  - E.g., routing on the internet
  - E.g., online auctions
General-sum games

• In general-sum games, can get win-win and lose-lose situations.
• E.g., “what side of road to drive on?”:

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

person driving towards you

you
General-sum games

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

<table>
<thead>
<tr>
<th></th>
<th>Aeon Flux</th>
<th>Corpse-bride</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aeon Flux</td>
<td>(8,2)</td>
<td>(0,0)</td>
</tr>
<tr>
<td>Corpse-bride</td>
<td>(0,0)</td>
<td>(2,8)</td>
</tr>
</tbody>
</table>

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 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>Aeon Flux</th>
<th>Corpse-bride</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aeon Flux</td>
<td>(8,2)</td>
<td>(0,0)</td>
</tr>
<tr>
<td>Corpse-bride</td>
<td>(0,0)</td>
<td>(2,8)</td>
</tr>
</tbody>
</table>

NE are: both Gr, both CB, or (80/20,20/80)
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)

<table>
<thead>
<tr>
<th></th>
<th>don’t pollute</th>
<th>pollute</th>
</tr>
</thead>
<tbody>
<tr>
<td>don’t pollute</td>
<td>(2,2)</td>
<td>(-1,3)</td>
</tr>
<tr>
<td>pollute</td>
<td>(3,-1)</td>
<td>(0,0)</td>
</tr>
</tbody>
</table>

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 randomized strategies (called "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

- 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. [great open problem!]
- Notation:
  - Assume an $n \times n$ 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 $\mathbb{R}^n$ 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)

• $S = \{(p,q): p,q \text{ are mixed strategies}\}$.
• 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.
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., penalty shot: If $p = (0.51, 0.49)$ then $q' = (1,0)$. If $p = (0.49,0.51)$ then $q' = (0,1)$.
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:
  - E.g., if $p = (0.5,0.5)$ then $q'$ could be anything.

<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>
Instead we will use...

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

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 \([(\text{expected gain wrt } q) - ||p-p'||^2]\)

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!