Goal: Compute a Nash equilibrium

1st idea: Best-response Dynamics
- start from arbitrary profile
- each agent switches to some BR
- repeat until convergence

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>W</td>
<td>7, -4</td>
<td>0, -9</td>
<td>-7, -3</td>
<td>-7, -6</td>
</tr>
<tr>
<td>X</td>
<td>-1, -6</td>
<td>-7, -8</td>
<td>1, -2</td>
<td>4, 1</td>
</tr>
<tr>
<td>Y</td>
<td>6, -4</td>
<td>0, -9</td>
<td>4, -1</td>
<td>0, 4</td>
</tr>
<tr>
<td>Z</td>
<td>4, 5</td>
<td>-2, -3</td>
<td>6, 9</td>
<td>3, -2</td>
</tr>
</tbody>
</table>

Only finds PSNE
Learning from Experts

Simple version:
- N experts
- T rounds
- at each round t, expert i predicts $p_i^t \in \{U, D\}
- each round we see all predictions and decide on U/D as ours
- assume one perfect expert

Goal: minimize our mistakes

**Halving Algorithm:**

$S \leftarrow \{1, \ldots, N\}$

For $t$ in $1, \ldots, T$:

$U_t \leftarrow \{i \in S \mid p_i^t = U\}$

$D_t \leftarrow \{i \in S \mid p_i^t = D\}$

If $|U_t| > |D_t|$: predict $U$
Else: predict $D$

If $O_t$ is $U$: $O_t$ is actual outcome in round $t$
For

How many mistakes can we make? (as a function of $T+N$) 

$\log_2(N) \rightarrow$ does not depend on $T$

- every mistake cuts $S$ in half
- $|S| \geq 1$ always

Harder Version: all experts make mistakes.

Let $OPT = \text{smallest number of mistakes by any expert}$

Iterated Halving Algorithm:

$S \leftarrow 1, \ldots, N^3$

for $t$ in $1, \ldots, T$:

if $|S|$ is 0:

$S \leftarrow 1, \ldots, N^3$

else:

$S \leftarrow S \setminus U$

end For

Every round if $S$ is empty reset to all experts.
$U_t \leftarrow \{i \in S \mid P_i = U\}$
$D_t \leftarrow \{i \in S \mid P_i = D\}$

if $|U_t| > |D_t|$ : predict $U$
else : predict $D$

if $O^t$ is $U$:

$s \leftarrow S \setminus D$
else:
$s \leftarrow S \setminus U$

end for

How many mistakes?

$\cdot T$
$\cdot N$
$\cdot OPT$

Each mistake by best could force a reset. Between resets, the best expert is right, so we can only make $\log_2 (N)$ mistakes.

So total: $(OPT+1) \log_2 (N) + OPT$

Times we have to identify the best mistakes to identify
Weighted Majority Algorithm

for i in 1...N:
    $W_i \leftarrow 1$

for t in 1...T:
    $U_t \leftarrow \sum i \mid \rho_i^t \text{ is } U$
    $D_t \leftarrow \sum i \mid \rho_i^t \text{ is } D$
    $W_U^t \leftarrow \sum_{i \in U_t} W_i$
    $W_D^t \leftarrow \sum_{i \in D_t} W_i$

if $W_U^t > W_D^t$: predict $U$
else: predict $D$

if $o^t$ is $U$:
    for $i \in D_t$:
        $W_i \leftarrow W_i / 2$
else:
    for i \in U_t:
        \text{\textbf{W}_i} \leftarrow \text{\textbf{W}_i} / 2
end for

\text{Let } M \text{ be the total number of mistakes.}
\text{Let } W^t \text{ be the total weight at time } t.

If we made a mistake on round \( t \), then
\[ W^{t+1} \leq \frac{3}{4} W^t \]

At round \( T \), \[ W^T \leq N \left( \frac{3}{4} \right)^M \]

Best expert makes OPT mistakes
\[ \left( \frac{1}{2} \right)^{\text{OPT}} = W^T_{\text{best}} \]