CIS 410/510 (Spring 2020): Multi-Agent Systems and Game Theory

Lecture 7: Behavioral Game Theory (P2)

Thanh H. Nguyen
University of Oregon
Some slides are by Milind Tambe
Reminder and Announcement

- Programming assignment 2
  - Deadline: April 25, 2020

- Written assignment 2
  - Will be posted today
  - Deadline: May 01, 2020

- Written assignment 1: Solution
  - Will be posted tomorrow (April 21, 2020)
Learning and Planning

Learn Attacker Behavior
Psychology
Machine Learning

Design Defense Strategy
Game Theory
Optimization

Real-world Deployment
Human Subject Experiments
Evaluation
Quantal Response (QR)

- Error in individual’s response
  - Still: more likely to select better choices than worse choices

- Probability distribution of different responses

- **Boundedly rational** attacker: attack a target $t_i$ with probability:

$$q(t_i, x) = \frac{e^{\lambda U_{att}(t_i, x)}}{\sum_j e^{\lambda U_{att}(t_j, x)}}$$

- $\lambda$: represents error level, governs rationality of attacker
Different values of $\lambda$

- $\lambda = 0$: attack targets uniformly at random
- $\lambda = \infty$: perfectly rational attacker
Learn Model Parameter: Maximum Likelihood Estimation (MLE)

- **Attack dataset**
  - $M$ games, $N$ targets.
  - Each game $G_m$:
    - Payoff matrix: $\{U_{\text{def}}^{\text{cov}}(t_i), U_{\text{def}}^{\text{uncov}}(t_i), U_{\text{att}}^{\text{cov}}(t_i), U_{\text{att}}^{\text{uncov}}(t_i)\}$
    - Defender strategy: $\tilde{x}_m = \{x_m(t_i)\}$
    - $N_m$ attacks: $N_m(t_i)$ -- number of attacks at target $t_i$

- **MLE**

$$\max_\lambda \sum_m \sum_i N_m(t_i) \log q(t_i, \tilde{x}_m) = \max_\lambda \sum_m \sum_i N_m(t_i) \log \frac{e^{\lambda U_{\text{att}}(t_i, \tilde{x}_m)}}{\sum_j e^{\lambda U_{\text{att}}(t_j, \tilde{x}_m)}}$$
Design Optimal Defense Strategy

- Example: $\lambda = 0.75$

<table>
<thead>
<tr>
<th></th>
<th>$t_1$</th>
<th>$t_2$</th>
<th>$t_3$</th>
<th>$t_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Covered</td>
<td>7, -8</td>
<td>6, -4</td>
<td>2, -5</td>
<td>10, -1</td>
</tr>
<tr>
<td>Uncovered</td>
<td>-3, 9</td>
<td>-5, 8</td>
<td>-4, 5</td>
<td>-10, 1</td>
</tr>
</tbody>
</table>

- Defender strategy: $\vec{x} = (0.3, 0.1, 0.2, 0.4)$
- Attack probabilities
  
  $$ q(t_1, \vec{x}) = \frac{e^{\lambda u_{att}(t_1, \vec{x})}}{\sum_j e^{\lambda u_{att}(t_j, \vec{x})}} = \frac{e^{0.75 \times [0.3 \times (-8) + (1-0.3) \times 9]}}{e^{0.75 \times [0.3 \times (-8) + (1-0.3) \times 9]} + e^{0.75 \times [0.1 \times (-4) + (1-0.1) \times 8]} + e^{0.75 \times [0.2 \times (-5) + (1-0.2) \times 5]} + e^{0.75 \times [0.4 \times (-1) + (1-0.4) \times 1]}} $$
Design Optimal Defense Strategy

Example: $\lambda = 0.75$

<table>
<thead>
<tr>
<th></th>
<th>$t_1$</th>
<th>$t_2$</th>
<th>$t_3$</th>
<th>$t_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Covered</td>
<td>7, -8</td>
<td>6, -4</td>
<td>2, -5</td>
<td>10, -1</td>
</tr>
<tr>
<td>Uncovered</td>
<td>-3, 9</td>
<td>-5, 8</td>
<td>-4, 5</td>
<td>-10, 1</td>
</tr>
</tbody>
</table>

Defender strategy: $\vec{x} = (0.3, 0.1, 0.2, 0.4)$

Attack probabilities: $(q(t_1, \vec{x}), q(t_2, \vec{x}), q(t_3, \vec{x}), q(t_4, \vec{x}))$

Defender’s utility:
- $U_{\text{def}}(\vec{x}) = q(t_1, \vec{x}) \times U_{\text{def}}(t_1, \vec{x}) + q(t_2, \vec{x}) \times U_{\text{def}}(t_2, \vec{x}) + q(t_3, \vec{x}) \times U_{\text{def}}(t_3, \vec{x}) + q(t_4, \vec{x}) \times U_{\text{def}}(t_4, \vec{x})$
BRQR: Optimize Defender Strategy

- **Optimization**
  - \[
  \max_{\vec{x}} \sum_i q(t_i, \vec{x}) \times U_{def}(t_i, \vec{x})
  \]
  - s.t. \( x_{t_i} \in [0,1], \forall i \)
  - \( \sum_i x_{t_i} = K \)

- **Non-convex**
  - Randomly set the starting point
  - **Hill climbing** to find local optimal
  - Reset the starting point and iterates

Defender utility
Resource constraint
The Online Game: LAX Airport Security

- Human subjects plays as an attacker
- Human subjects are given $8 as the starting budget
- For each point they gain, $0.1 real money is paid
The Online Game: Wildlife Protection

Game 2
Caught!
Total: $1.3 = $1.4 - $0.1

Reward if successful: $9
Penalty if caught by rangers: $-1
Money earned if successful: $1

Percentage of success: 0%
Percentage of failure: 100%

End Game
Evaluate the Strategies

- Average expected utility of the defender
  - Based on human subject responses

\[ \tilde{U}_{\text{def}} = \frac{1}{N_{\text{subject}}} \sum_{k=1}^{N_{\text{subject}}} U_{\text{def}}(s_k, \bar{x}) \]

- $N_{\text{subject}}$: total number of human subjects
- $s_k$: subject’s choice of target to attack
Limitations of Expected Utility

- In real-world domains with various features influencing attacker’s decision making, is expected utility an appropriate measurement?

- Is attacker behavior actually governed by expected utility?
Subjective Utility

- Linear combination of feature values: $f_1(t_i), f_2(t_i), \ldots, f_M(t_i)$
  - $f_m(t_i)$: value of $m^{th}$ feature at target $i$
  - $M$: total number of feature

- Attacker’s subjective utility at target $i$

\[
SU_{\text{att}}(t_i, \bar{x}) = w_1 f_1(t_i) + w_2 f_2(t_i) + \cdots + w_M f_M(t_i)
\]

- $w_m$: weight of $m^{th}$ feature for attacker
SUQR

- Predict probability attacker attacks each target
- A combination of subjective utility and Quantal Response

Learn weights of subjective utility: MLE

We can simply set $\lambda=1$: why?

$q(t_i, x) = \frac{e^{\lambda SU_{att}(t_i, x)}}{\sum_j e^{\lambda SU_{att}(t_j, x)}}$

Attacker’s SU at $t_i$
Limitations of Expected Utility

- In real-world domains with various features influencing attacker’s decision making, is expected utility an appropriate measurement?

- Is attacker behavior actually governed by expected utility?
  - Perceived utility (value) of each outcome
  - Perceived probabilities
Risk Aversion: Example

- A person is given the choice between 2 scenarios:
  - Guaranteed scenario: the person receives $50
  - Uncertain scenario: a coin is flipped to decide the person receive $100 or not.

- Which choice would that person make?
Risk Aversion

- **Risk averse**: would accept the guaranteed payment of (less than) $50 rather than take the gamble

- **Risk neutral**: indifferent between the bet and the guaranteed $50 payment

- **Risk seeking**: would accept the bet even when the guaranteed payment is more than $50
Risk Aversion

Risk averse individual: $E[U(x)] < U(\bar{x})$

Risk neutral individual: $E[U(x)] = U(\bar{x})$

Risk loving individual: $E[U(x)] > U(\bar{x})$

Prospect Theory

- A behavioral model (in psychology)
  - How people decide between alternatives that involve risks and uncertainty

- Components
  - **Value function**: values of outcomes
  - **Weight function**: perceived probabilities
Prospect Theory: Value Function

- **Risk aversion**: convexity
  - Risk averse regarding gain
  - Risk seeking regarding loss

- **Loss aversion**
  - Losses are felt more strong than gains

- **Endowment effect**
  - We values things we own more highly
  - Reference point: differentiate gains and loss

Source: https://www.economicshelp.org/blog/glossary/prospect-theory/
Prospect Theory: Weight Function

- **Underweight** of high probability
- **Overweight** of small probability

Weight function:

$$\pi(p) = \frac{p^\gamma}{(p^\gamma + (1-p)^\gamma)^\frac{1}{\gamma}}$$

Prospect Theory vs Expected Utility

- **Expected utility**
  \[ U_{\text{att}}(t_i, \tilde{x}) = U_{\text{att}}^{\text{uncov}} \times (1 - x_{t_i}) + U_{\text{att}}^{\text{cov}} \times x_{t_i} \]

- **Prospect theory utility**
  \[ U_{\text{att}}^{\text{PT}}(t_i, \tilde{x}) = V(U_{\text{att}}^{\text{uncov}}) \times \pi(1 - x_{t_i}) + V(U_{\text{att}}^{\text{cov}}) \times \pi(x_{t_i}) \]
Other Models in Machine Learning

- Decision tree
- Logistic regression
  - Multiple attacks
- Neural networks


Prospect Theory: http://www.albacharia.ma/xmlui/bitstream/handle/123456789/31987/kahnmtversky.pdf?sequence=1