Computing Minimax Decisions with Incomplete Observations

Thijs van Ommen

University of Amsterdam

ISIPTA — July 10, 2017
Monty Hall’s game show

Initial probability:

1/3  1/3  1/3

Illustration by Gracia Bovenberg-Murris
Monty Hall’s game show

Initial probability:
1/3 1/3 1/3

Illustration by Gracia Bovenberg-Murris
Monty Hall’s game show

Initial probability:
1/3
1/3
1/3

New probability:
? ? 0

Illustration by Gracia Bovenberg-Murris
Formalizing the problem

We will look at the part of the problem after the initial choice of door\(^1\)

Step 1 Outcome \(X\) is randomly drawn from \(\mathcal{X} = \{x_1, x_2, x_3\}\) (the three doors) according to the uniform distribution \(p\)

\(^1\)This is the setting of Van Ommen, Koolen, Feenstra and Grünewald (2016), IJAR
We will look at the part of the problem after the initial choice of door\(^1\)

**Step 1** Outcome \(X\) is randomly drawn from \(\mathcal{X} = \{x_1, x_2, x_3\}\) (the three doors) according to the uniform distribution \(p\)

**Step 2** The quizmaster, knowing \(X\), chooses a set \(Y \in \mathcal{Y} = \{\{x_1, x_2\}, \{x_2, x_3\}\}\) such that \(Y \ni X\)

- The structure of \(\mathcal{Y}\) reflects that the quizmaster will always open one door, but never the door the contestant picked
- The chosen set \(Y\) is called the *message*

---

\(^1\)This is the setting of Van Ommen, Koolen, Feenstra and Grünewald (2016), IJAR
We will look at the part of the problem after the initial choice of door$^1$

**Step 1** Outcome $X$ is randomly drawn from $\mathcal{X} = \{x_1, x_2, x_3\}$ (the three doors) according to the uniform distribution $p$

**Step 2** The quizmaster, knowing $X$, chooses a set $Y \in \mathcal{Y} = \{\{x_1, x_2\}, \{x_2, x_3\}\}$ such that $Y \ni X$
- The structure of $\mathcal{Y}$ reflects that the quizmaster will always open one door, but never the door the contestant picked
- The chosen set $Y$ is called the message

**Step 3** The contestant sees $Y$ but not $X$, and must make a decision based on this incomplete observation

---

$^1$This is the setting of Van Ommen, Koolen, Feenstra and Grünwald (2016), IJAR
We also want to know what probabilities to assign to the outcomes in a more general situation:

- For arbitrary (but finite) outcome spaces $\mathcal{X}$;
- For arbitrary marginal distribution $\rho$;
- For arbitrary families of allowed messages $\mathcal{Y}$. 
The quizmaster’s freedom of choice

- The quizmaster may use randomness when deciding which message $Y$ to give us.
- However, we don’t know what distribution $P(Y \mid X)$ he uses.

The conditional distribution $P(Y \mid X)$ together with the marginal distribution $p_X$ on $X$ gives a joint distribution $P(X, Y)$:

- Quizmaster uses a fair coin: 
  $$
  P(x_1, x_2, x_3) = \{x_1, x_2\} \frac{1}{3} \frac{1}{6} - \{x_2, x_3\} - \frac{1}{6} \frac{1}{3}
  $$
  $$
  p_{x_1} \frac{1}{3} \frac{1}{3} \frac{1}{3}
  $$

- Quizmaster always opens $x_3$: 
  $$
  P(x_1, x_2, x_3) = \{x_1, x_2\} \frac{1}{3} \frac{1}{3} - \{x_2, x_3\} - 0 \frac{1}{3}
  $$
  $$
  p_{x_1} \frac{1}{3} \frac{1}{3} \frac{1}{3}
  $$

Decision maker has aleatory uncertainty about $X$, and epistemic uncertainty about $Y$ given $X$. → the possible joint distributions form a credal set.
The quizmaster’s freedom of choice

- The quizmaster may use randomness when deciding which message \( Y \) to give us
- However, we don’t know what distribution \( P(Y \mid X) \) he uses
- The conditional distribution \( P(Y \mid X) \) together with the marginal distribution \( p \) on \( X \) gives a joint distribution \( P(X, Y) \):

<table>
<thead>
<tr>
<th>( P )</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( {x_1, x_2} )</td>
<td>1/3</td>
<td>1/6</td>
<td>-</td>
</tr>
<tr>
<td>( {x_2, x_3} )</td>
<td>-</td>
<td>1/6</td>
<td>1/3</td>
</tr>
<tr>
<td>( p_x )</td>
<td>1/3</td>
<td>1/3</td>
<td>1/3</td>
</tr>
</tbody>
</table>

Quizmaster uses fair coin:

<table>
<thead>
<tr>
<th>( P )</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( {x_1, x_2} )</td>
<td>1/3</td>
<td>1/3</td>
<td>1/3</td>
</tr>
</tbody>
</table>

Quizmaster always opens \( x_3 \):

<table>
<thead>
<tr>
<th>( P )</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( {x_1, x_2} )</td>
<td>1/3</td>
<td>1/3</td>
<td>-</td>
</tr>
<tr>
<td>( {x_2, x_3} )</td>
<td>-</td>
<td>0</td>
<td>1/3</td>
</tr>
<tr>
<td>( p_x )</td>
<td>1/3</td>
<td>1/3</td>
<td>1/3</td>
</tr>
</tbody>
</table>

- Decision maker has aleatory uncertainty about \( X \), and epistemic uncertainty about \( Y \) given \( X \)
- → the possible joint distributions form a credal set
Minimax decision problem

- Worst-case approach: we want to give guarantees on our decisions that hold no matter what mechanism is used to choose the message
  - Corresponds to a two-player zero-sum game between the contestant and the quizmaster
Minimax decision problem

- Worst-case approach: we want to give guarantees on our decisions that hold no matter what mechanism is used to choose the message
  - Corresponds to a two-player zero-sum game between the contestant and the quizmaster

Different action spaces possible:
- Contestant’s action may be choosing a single outcome
  - Can put any loss function on this
  - We allow him to randomize, to ensure existence of Nash equilibrium

Interesting alternative: it may be a prediction $Q$ over the outcomes. Can then consider different loss functions (scoring rules); for example:
- Logarithmic loss: $L(x, Q) = -\log Q(x)$
- Brier loss: $L(x, Q) = \sum_{x' \in X} (Q(x') - 1)_{x'}^2$
Minimax decision problem

- Worst-case approach: we want to give guarantees on our decisions that hold no matter what mechanism is used to choose the message
  - Corresponds to a two-player zero-sum game between the contestant and the quizmaster

Different action spaces possible:

- Contestant’s action may be choosing a single outcome
  - Can put any loss function on this
  - We allow him to randomize, to ensure existence of Nash equilibrium

- Interesting alternative: it may be a prediction $Q$ over the outcomes
  - Can then consider different loss functions (scoring rules); for example:

  \[
  \begin{align*}
  \text{Logarithmic loss:} & \quad L(x, Q) = - \log Q(x) \\
  \text{Brier loss:} & \quad L(x, Q) = \sum_{x' \in \mathcal{X}} (Q(x') - 1_{x' = x})^2
  \end{align*}
  \]
If $L$ is logarithmic loss, the characterization of optimality takes a very nice form:

**Theorem (IJAR 2016 paper)**

For logarithmic loss, a joint distribution $P^*$ is optimal for the quizmaster if and only if there exists a vector $q \in [0, 1]^X$ such that

$$q_x = P^*(x \mid y) \quad \text{for all } x \in y \in \mathcal{Y} \text{ with } P^*(y) > 0, \text{ and}$$

$$\sum_{x \in y} q_x \leq 1 \quad \text{for all } y \in \mathcal{Y}$$

We call this condition on $P^*$ the RCAR condition.
If $L$ is logarithmic loss, the characterization of optimality takes a very nice form:

**Theorem (IJAR 2016 paper)**

For logarithmic loss, a joint distribution $P^*$ is optimal for the quizmaster if and only if there exists a vector $q \in [0, 1]^\mathcal{X}$ such that

$$q_x = P^*(x \mid y) \quad \text{for all } x \in y \in \mathcal{Y} \text{ with } P^*(y) > 0, \text{ and}$$

$$\sum_{x \in y} q_x \leq 1 \quad \text{for all } y \in \mathcal{Y}$$

We call this condition on $P^*$ the **RCAR condition**

Same condition applies if $\mathcal{Y}$ is a ‘graph game’ or a ‘matroid game’, for any loss function!
Previous theorem allows us to recognize whether a strategy is minimax optimal, but not to *find* such strategies.

- One thing that makes this hard: combinatorial search due to distinction $P^*(y) > 0$ vs. $P^*(y) = 0$
Previous theorem allows us to recognize whether a strategy is minimax optimal, but not to find such strategies

- One thing that makes this hard: combinatorial search due to distinction $P^*(y) > 0$ vs. $P^*(y) = 0$
- And another: may require solving system of polynomial equations
Partition matroid: partition $\mathcal{X}$ into $S_1, \ldots, S_k$; $\mathcal{Y}$ consists of all subsets of $\mathcal{X}$ that take one element from each $S_i$

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${x_1, x_3}$</td>
<td>*</td>
<td>−</td>
<td>*</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>${x_1, x_4}$</td>
<td>*</td>
<td>−</td>
<td>−</td>
<td>*</td>
<td>−</td>
</tr>
<tr>
<td>${x_1, x_5}$</td>
<td>*</td>
<td>−</td>
<td>−</td>
<td>−</td>
<td>*</td>
</tr>
<tr>
<td>${x_2, x_3}$</td>
<td>−</td>
<td>*</td>
<td>*</td>
<td>−</td>
<td>−</td>
</tr>
<tr>
<td>${x_2, x_4}$</td>
<td>−</td>
<td>*</td>
<td>−</td>
<td>*</td>
<td>−</td>
</tr>
<tr>
<td>${x_2, x_5}$</td>
<td>−</td>
<td>*</td>
<td>−</td>
<td>−</td>
<td>*</td>
</tr>
</tbody>
</table>
Well-behaved case: Partition matroids

**Partition matroid**: partition $\mathcal{X}$ into $S_1, \ldots, S_k$; $\mathcal{Y}$ consists of *all* subsets of $\mathcal{X}$ that take one element from each $S_i$

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${x_1, x_3}$</td>
<td>*</td>
<td>—</td>
<td>*</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>${x_1, x_4}$</td>
<td>*</td>
<td>—</td>
<td>—</td>
<td>*</td>
<td>—</td>
</tr>
<tr>
<td>${x_1, x_5}$</td>
<td>*</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>*</td>
</tr>
<tr>
<td>${x_2, x_3}$</td>
<td>—</td>
<td>*</td>
<td>*</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>${x_2, x_4}$</td>
<td>—</td>
<td>*</td>
<td>—</td>
<td>*</td>
<td>—</td>
</tr>
<tr>
<td>${x_2, x_5}$</td>
<td>—</td>
<td>*</td>
<td>—</td>
<td>—</td>
<td>*</td>
</tr>
</tbody>
</table>

**Example:**

- messages (rows) are products
- $S_1 = \{x_1, x_2\}$ are brands, $S_2 = \{x_3, x_4, x_5\}$ are colours; customers buy products based on preference for either a brand or a colour
- shopkeeper observes customer buying a product and wants to know underlying preference
For partition matroid, RCAR solution can be computed directly:

- \( q_x = \sum_{x' \in S_i} p_{x'} \), where \( S_i \) is the set containing \( x \)
- Possible choice for \( P(y) \) (may not be unique):

\[
P(y) = \prod_{x \in y} \frac{p_x}{q_x}.
\]
For partition matroid, RCAR solution can be computed directly:

- \( q_x = \sum_{x' \in S_i} p_{x'} \), where \( S_i \) is the set containing \( x \)
- Possible choice for \( P(y) \) (may not be unique):

\[
P(y) = \prod_{x \in y} \frac{p_x}{q_x}.
\]

Interpretation: this \( P \) makes the message \( Y \) independent of the index \( l \) of the true set \( S_i \) — tells the decision maker nothing extra!
RCAR solutions play a central role in this decision problem with incomplete observations, but are often hard to compute. . . but are very easy to compute if $\mathcal{Y}$ is a partition matroid!

Efficient algorithms for graph games and general matroid games also exist (Chapter 8 of Van Ommen, 2015).

Thank you!
Optimal strategy may depend on the loss function

This strategy \( P \) is optimal for logarithmic loss (it satisfies the RCAR condition), but not for Brier loss.
If the set of available messages $\mathcal{Y}$ forms a graph (meaning that each message contains exactly two outcomes), then the RCAR condition characterizes optimality regardless of the loss function;

If $\mathcal{Y}$ forms a matroid (satisfies the matroid basis exchange property), then the same is true;

For any other $\mathcal{Y}$, this is not the case: there exists some marginal $p$ such that the optimal strategies for log loss and Brier loss are different.