Advanced Mixed Integer Programming Formulations for Non-Convex Optimization Problems in Statistical Learning

Juan Pablo Vielma

Massachusetts Institute of Technology

### Feature Comparison:

<table>
<thead>
<tr>
<th>Feature</th>
<th>SX530</th>
<th>RX100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zoom</td>
<td>50x</td>
<td>3.6x</td>
</tr>
<tr>
<td>Prize</td>
<td>$249.99</td>
<td>$399.99</td>
</tr>
<tr>
<td>Weight</td>
<td>15.68 ounces</td>
<td>7.5 ounces</td>
</tr>
<tr>
<td>Prefer</td>
<td>✔</td>
<td>☐</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Feature</th>
<th>TG-4</th>
<th>Galaxy 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Waterproof</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Prize</td>
<td>$249.99</td>
<td>$399.99</td>
</tr>
<tr>
<td>Weight</td>
<td>7.36 lb</td>
<td>7.5 lb</td>
</tr>
<tr>
<td>Prefer</td>
<td>☐</td>
<td>✔</td>
</tr>
</tbody>
</table>

We recommend:

1. SX530
2. TG-4

---

We recommend: Galaxy 2

---

(Custom) Product Recommendations via CBCA
Towards Optimal Product Recommendation

• Find enough information about preferences to recommend

• How do I pick the next (1st) question to obtain the largest reduction of uncertainty or “variance” on preferences
### Choice-based Conjoint Analysis

#### Product Profile

<table>
<thead>
<tr>
<th>Feature</th>
<th>Chewbacca</th>
<th>BB-8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wookiee</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Droid</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Blaster</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td><strong>I would buy toy</strong></td>
<td>✔️</td>
<td>☐</td>
</tr>
</tbody>
</table>

#### MIP Formulations for Non-convex Optimization

\[
\begin{pmatrix}
0 \\
1 \\
0
\end{pmatrix} \times \begin{pmatrix}
 x^1 \\
x^2
\end{pmatrix} = x^2
\]
MNL Preference Model

- Utilities for 2 products, n features (e.g. n = 12)

\[ U_1 = \beta \cdot x^1 + \epsilon_1 = \sum_{i=1}^{n} \beta_i x_i^1 + \epsilon_1 \]

\[ U_2 = \beta \cdot x^2 + \epsilon_2 = \sum_{i=1}^{n} \beta_i x_i^2 + \epsilon_2 \]

- Utility maximizing customer: \( x^1 \geq x^2 \iff U_1 \geq U_2 \)

- Noise can result in response error:

\[
L(\beta | x^1 \geq x^2) = \mathbb{P}(x^1 \geq x^2 | \beta) = \frac{e^{\beta \cdot x^1}}{e^{\beta \cdot x^1} + e^{\beta \cdot x^2}}
\]

MIP Formulations for Non-convex Optimization
Next Question To Reduce “Variance”: Bayesian

Prior Distribution of $\beta$

<table>
<thead>
<tr>
<th>Feature</th>
<th>SX530</th>
<th>RX100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zoom</td>
<td>50x</td>
<td>3.6x</td>
</tr>
<tr>
<td>Prize</td>
<td>$249.99</td>
<td>$399.99</td>
</tr>
<tr>
<td>Weight</td>
<td>15.68 ounces</td>
<td>7.5 ounces</td>
</tr>
<tr>
<td>Prefer</td>
<td>✔</td>
<td>☐</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Feature</th>
<th>TG-4</th>
<th>Galaxy 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Waterproof</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Prize</td>
<td>$249.99</td>
<td>$399.99</td>
</tr>
<tr>
<td>Viewfinder</td>
<td>Electronic</td>
<td>Optical</td>
</tr>
<tr>
<td>Prefer</td>
<td>✔</td>
<td>☐</td>
</tr>
</tbody>
</table>

• **Black-box objective:** Question Selection = Enumeration 😞

• **Question selection by Mixed Integer Programming (MIP)**
Avoiding Enumeration with MIP
Traveling Salesman Problem (TSP): Visit Cities Fast
How about 49 cities?

- Number of tours $= \frac{48!}{2} \approx 10^{60}$
- Fastest supercomputer $\approx 10^{17}$ flops
- Assuming one floating point operation per tour:
  $> 10^{35}$ years $\approx 10^{25}$ times the age of the universe!
- How long does it take on an iphone?
  - Less than a second!
  - 4 iterations of cutting plane method!
  - Dantzig, Fulkerson and Johnson 1954 did it by hand!
  - For more info see tutorial in ConcordeTSP app
- Cutting planes are the key for effectively solving (even NP-hard) MIP problems in practice.
50+ Years of MIP = Significant Solver Speedups

• Algorithmic Improvements (Machine Independent):
  – Commercial, but free for academic use
• (Reasonably) effective free / open source solvers:
  – GLPK, CBC and SCIP (free only for non-commercial)
• Easy to use, fast and versatile modeling languages
  – Julia based JuMP modelling language
• Linear MIP solvers very mature and effective:
  – Convex nonlinear MIP getting there (quadratic nearly there)
Question Selection with MIP
Bayesian Update and Geometric Updates

Prior distribution

\[ \beta \sim N(\mu, \Sigma) \]

\[ \phi(\beta; \mu, \Sigma) \]

Answer likelihood

\[ x^1 \succeq x^2 \]

\[ L(\beta \mid x^1 \succeq x^2) \]

Posterior distribution

\[ f(\beta \mid x^1 \succeq x^2) = \frac{\phi(\beta; \mu, \Sigma) L(\beta \mid x^1 \succeq x^2)}{\int_{\mathbb{R}} \phi(\beta; \mu, \Sigma) L(\beta \mid x^1 \succeq x^2) d\beta} \]

Multidimensional Integration?

non-convex on \( x^1, x^2 \in \{0, 1\}^n \)
D-Efficiency and Posterior Covariance Matrix

"Variance" = D-Efficiency:

- $f(x^1, x^2) := E_{\beta,x^1} \succeq x^2 \left( \text{det}(\Sigma_i)^{1/p} \right)$
- Non-convex function
- Even evaluating expected D-Efficiency for a question requires multidimensional integration

$\beta \sim N(\mu, \Sigma)$

$\text{cov}(\beta) = \Sigma_1$

$\text{cov}(\beta) = \Sigma_2$
Standard Question Selection Criteria

\[(\beta - \mu)' \cdot \Sigma^{-1} \cdot (\beta - \mu) \leq r\]

- **Choice balance:**
  - Minimize distance to center

\[\mu \cdot (x^1 - x^2)\]

- **Postchoice symmetry:**
  - Maximize variance of question

\[(x^1 - x^2)' \cdot \Sigma \cdot (x^1 - x^2)\]
D-efficency: Balance Question Trade-off

- **D-efficiency** = Non-convex function $f(d, v)$ of
distance: $d := \mu \cdot (x^1 - x^2)$

variance: $v := (x^1 - x^2)' \cdot \sum \cdot (x^1 - x^2)$

Can evaluate $f(d, v)$ with 1-dim integral 😃
Optimization Model

\[
\begin{align*}
\text{min} & \quad f(d, v) \quad \times \\
\text{s.t.} & \quad \mu \cdot (x^1 - x^2) = d \quad \checkmark \\
& \quad (x^1 - x^2)' \cdot \sum \cdot (x^1 - x^2) = v \quad \xmark \\
& \quad A^1 x^1 + A^2 x^2 \leq b \quad \checkmark \\
& \quad x^1 \neq x^2 \quad \xmark \\
& \quad x^1, x^2 \in \{0, 1\}^n
\end{align*}
\]
Technique 1: Binary Quadratic \( x^1, x^2 \in \{0, 1\}^n \)

\[
(x^1 - x^2)' \cdot \sum (x^1 - x^2) = \nu
\]

\[
X^l_{i,j} = x^l_i \cdot x^l_j \quad (l \in \{1, 2\}, \quad i, j \in \{1, \ldots, n\}) :
\]

\[
X^l_{i,j} \leq x^l_i, \quad X^l_{i,j} \leq x^l_j, \quad X^l_{i,j} \geq x^l_i + x^l_j - 1, \quad X^l_{i,j} \geq 0
\]

\[
W_{i,j} = x^1_i \cdot x^2_j :
\]

\[
W_{i,j} \leq x^1_i, \quad W_{i,j} \leq x^2_j, \quad W_{i,j} \geq x^1_i + x^2_j - 1, \quad W_{i,j} \geq 0
\]

\[
\sum_{i,j=1}^{n} (X^1_{i,j} + X^2_{i,j} - W_{i,j} - W_{j,i}) \sum_{i,j} = \nu
\]

MIP Formulations for Non-convex Optimization
Technique 1: Binary Quadratic \( x^1, x^2 \in \{0, 1\}^n \)

\[
x^1 \neq x^2 \iff \|x^1 - x^2\|_2^2 \geq 1
\]

\[
X^l_{i,j} = x^l_i \cdot x^l_j \quad (l \in \{1, 2\}, \quad i, j \in \{1, \ldots, n\}) : \\
X^l_{i,j} \leq x^l_i, \quad X^l_{i,j} \leq x^l_j, \quad X^l_{i,j} \geq x^l_i + x^l_j - 1, \quad X^l_{i,j} \geq 0
\]

\[
W_{i,j} = x^1_i \cdot x^2_j : \\
W_{i,j} \leq x^1_i, \quad W_{i,j} \leq x^2_j, \quad W_{i,j} \geq x^1_i + x^2_j - 1, \quad W_{i,j} \geq 0
\]

\[
\sum_{i,j=1}^n (X^1_{i,j} + X^2_{i,j} - W_{i,j} - W_{j,i}) \geq 1
\]
Technique 2: Piecewise Linear Functions

- D-efficiency = Non-convex function $f(d, v)$ of distance: $d := \mu \cdot (x^1 - x^2)$

- variance: $v := (x^1 - x^2)' \cdot \sum \cdot (x^1 - x^2)$

Can evaluate $f(d, v)$ with 1-dim integral 😊

Piecewise Linear Interpolation

MIP formulation
2. Modeling Piecewise Linear Functions

An appropriate way of modeling a piecewise linear function \( f : D \to \mathbb{R}^n \) is to model its epigraph given by \( \text{epi}(f) = \{ (x, z) \in D \times \mathbb{R} : f(x) \leq z \} \). For example, the epigraph of the function in Figure 2(a) is depicted in Figure 2(b).

For simplicity, we assume that the function domain \( D \) is bounded and \( f \) is only used in a constraint of the form \( f(x) \leq 0 \) or as an objective function that is being minimized. We then need a model of \( \text{epi}(f) \) since \( f(x) \leq 0 \) can be modeled as \( (x, z) \in \text{epi}(f), z \geq 0 \) and the minimization of \( f \) can be achieved by minimizing \( z \) subject to \( (x, z) \in \text{epi}(f) \). For continuous functions we can also work with its graph, but modeling the epigraph will allow us to extend most of the results to some discontinuous functions and will simplify the analysis of formulation properties.
An appropriate way of modeling a piecewise linear function $f: \mathbb{D} \rightarrow \mathbb{R}^n$ is to model its epigraph given by $\text{epi}(f) = \{(x, z) \in \mathbb{D} \times \mathbb{R} : f(x) \leq z\}$. For example, the epigraph of the function in Figure 2(a) is depicted in Figure 2(b).

For simplicity, we assume that the function domain $\mathbb{D}$ is bounded and $f$ is only used in a constraint of the form $f(x) \leq 0$ or as an objective function that is being minimized. We then need a model of $\text{epi}(f)$ since $f(x) \leq 0$ can be modeled as $(x, z) \in \text{epi}(f), z \leq 0$ and the minimization of $f$ can be achieved by minimizing $z$ subject to $(x, z) \in \text{epi}(f)$. For continuous functions we can also work with its graph, but modeling the epigraph will allow us to extend most of the results to some discontinuous functions and will simplify the analysis of formulation properties.
Computational Performance

• Advanced formulations provide an computational advantage
• Advantage is significantly more important for free solvers
• State of the art commercial solvers can be significantly better that free solvers
• Still, free is free!
Summary and Main Messages

• Always choose Chewbacca!

• MIP can solve very challenging problems in practice

• Commercial solvers best, but free solvers reasonable
  – Easily accessible and integrated into complex systems through the JuMP modeling language github.com/JuliaOpt/JuMP.jl

• Formulations = speed-ups and are (relatively) easy to learn

• CBC application: http://ssrn.com/abstract=2798984