Assortment Optimization

- Although we want choice models that capture a wide range of customer behavior, we also want the models to be easy to optimize over.

- The main pricing problem related to choice models is assortment optimization: given a set of items (with given prices) and a choice model, we want to offer a subset of items to maximize revenue.

  Formally: Given item set $N = \{1, 2, \ldots, n\}$, choice model $\Pi$ s.t. $\Pi_j(s) = P[\text{Item } j \text{ purchased from set } s \subseteq N]$.
  
  - Each item $j$ has price $P_j$
  
  - If set $S$ is offered, revenue $R(s) = \sum_{j \in S} \Pi_j(s) P_j$

  - Aim - Pick $S^* = \arg\max_{S \subseteq N} R(s)$
Note - # of subsets of \( N = 2^n - 1 \)
In general, finding \( S^* \) may be difficult.
- However, for the MNL model, it can be found efficiently using a simple algorithm!

**Assortment opt under the MNL model**

- Recall - For the MNL, \( \exists \ \theta_j \geq 0 \) for each item \( j \) s.t.
  \[
  \pi_i(s) = \frac{\theta_i}{\theta_0 + \sum_{i \in S} \theta_i} = \theta_i \left( 1 + \sum_{i \in S} \theta_i \right)^{-1} \theta(s)
  \]
  can assume \( \theta_0 = 1 \) by normalizing

  \[
  \Rightarrow \quad R(s) = \sum_{i \in S} \theta_i p_j \quad ; \quad R^* = \max_{S \subseteq N} R(s)
  \]

- Thm (Galego, Talluri, Van Ryzin) - Let \( p_1 \geq p_2 \geq \ldots \geq p_n \), and
  \[
  E_0 = \emptyset, \quad E_1 = \{1, 2\}, \quad E_2 = \{1, 2, 3\}, \quad \ldots, \quad E_n = N
  \]
  (nested-by-revenue sets)

  Then \( \exists k^* \in \{0, 1, \ldots, n\} \) s.t. \( E_{k^*} \in \arg \max_{S \subseteq N} R(s) \)

  (i.e., there is some nested-by-revenue set which has opt revenue!)
Pf - By definition, we know $\forall \text{SCN}$

$$R^* \geq \frac{\sum_{i \in S} v_{ij} p_j}{1 + \sum_{i \in S} \theta_i}$$

$\Rightarrow \ [R^* \geq \sum_{i \in S} v_{ij} (p_j - R^*)] \ \ \ \ \forall \text{SCN}$

- Now suppose an oracle told us $R^*$; then in order to find $S^*$, we can instead find:

$(S^* =) \quad S^* = \arg \max_{S \in \text{SCN}} \sum_{i \in S} v_{ij} (p_j - R^*)$ 'Virtual price'

- The solution to the above problem is to pick $S^* = \{j \mid p_j \geq R^*\}$

- Observe that $S^* = E_k$ (nested-by-revenue set) for some

- What if we do not know $R^*$?

  We can still search over the $\{E_k\}$ to find the best!

  $\Rightarrow \quad S^* = S^* = \max_{k \in \{0,1,\ldots,n\}} \left[R(E_k)\right]$
**Eg.** - (Assortment Opt under MNL)

5 items, prices \((P_1, P_2, \ldots, P_5) = (7, 6, 4, 3, 2)\)

MNL parameters \((\alpha_1, \alpha_2, \ldots, \alpha_5) = (3, 5, 6, 4, 5), \; \alpha_6 = 10\)

<table>
<thead>
<tr>
<th>Assortment (S)</th>
<th>{1,3}</th>
<th>{1,2}</th>
<th>{1,2,3}</th>
<th>{1,2,3,4}</th>
<th>{1,2,3,5}</th>
</tr>
</thead>
<tbody>
<tr>
<td>(R(S))</td>
<td>1.615</td>
<td>2.833</td>
<td>3.125</td>
<td>3.107</td>
<td>2.939</td>
</tr>
</tbody>
</table>

However, these problems are much harder to solve for general \(T_i; s\)!

**Eg.** - (2-class MNL)

2 classes \(\{a, b\}\), \(\alpha^a = 0.5\), \(\alpha^b = 0.5\)

\(\alpha^a = (5, 20, 1), \alpha^b = (10, 15, 10), \alpha^a = 1\)

Prices \(= (8, 4, 3)\)

Then: Opt for class \(a = \{1,3\}, R^* = 20/3\)

Opt for class \(b = \{1,2,3\}, R^* = 26/7\)

Opt for mixture \(= \{1,3\}, R^* = 4.48\)

Not nested-by-revenue!

**Thm.** - 2-class MNL assortment optimization is NP-complete.