Capacitive MR-Sort model
Preference modeling and learning

Olivier Sobrie$^{1,2}$ - Vincent Mousseau$^1$ - Marc Pirlot$^2$

$^1$École Centrale de Paris - Laboratoire de Génie Industriel
$^2$University of Mons - Faculty of engineering

November 20, 2014
1 Introductory example

2 MR-Sort

3 Capacitive MR-Sort

4 Learning a Capacitive MR-Sort model

5 Experimentations

6 Comments and Conclusion
1 Introductory example

2 MR-Sort

3 Capacitive MR-Sort

4 Learning a Capacitive MR-Sort model

5 Experimentations

6 Comments and Conclusion
Introductory example

- Admission/Refusal of student
- Students are evaluated in 4 courses
- Admission condition: score above 10/20 in all the courses of one the minimal winning coalitions.

Minimal winning coalitions

- \{math, physics\}
- \{math, chemistry\}
- \{chemistry, history\}

Maximal loosing coalitions

- \{math, history\}
- \{physics, chemistry\}
- \{physics, history\}

<table>
<thead>
<tr>
<th></th>
<th>Math</th>
<th>Physics</th>
<th>Chemistry</th>
<th>History</th>
<th>A/R</th>
</tr>
</thead>
<tbody>
<tr>
<td>James</td>
<td>11</td>
<td>11</td>
<td>9</td>
<td>9</td>
<td>A</td>
</tr>
<tr>
<td>Marc</td>
<td>11</td>
<td>9</td>
<td></td>
<td>9</td>
<td>A</td>
</tr>
<tr>
<td>Robert</td>
<td>9</td>
<td>9</td>
<td>11</td>
<td>11</td>
<td>A</td>
</tr>
<tr>
<td>John</td>
<td>11</td>
<td>9</td>
<td>9</td>
<td>11</td>
<td>R</td>
</tr>
<tr>
<td>Paul</td>
<td>9</td>
<td>11</td>
<td>9</td>
<td>11</td>
<td>R</td>
</tr>
<tr>
<td>Pierre</td>
<td>9</td>
<td>11</td>
<td>11</td>
<td>9</td>
<td>R</td>
</tr>
</tbody>
</table>
1 Introductory example

2 MR-Sort

3 Capacitive MR-Sort

4 Learning a Capacitive MR-Sort model

5 Experimentations

6 Comments and Conclusion
MR-Sort I

Characteristics

- Allows to sort alternatives in ordered classes on basis of their performances on monotone criteria
- MCDA method based on outranking relations
- Simplified version of ELECTRE TRI

Parameters

- Profiles performances \((b_{h,j} \text{ for } h = 1, ..., p - 1; j = 1, ..., n)\)
- Criteria weights \((w_j \geq 0 \text{ for } n = 1, ..., n)\)
- Majority threshold \((\lambda)\)
MR-Sort II

Parameters

- Profiles performances ($b_{h,j}$ for $h = 1, \ldots, p-1; j = 1, \ldots, n$)
- Criteria weights ($w_j \geq 0$ for $n = 1, \ldots, n$)
- Majority threshold ($\lambda$)

Assignment rule

$$a \in C_h \iff \sum_{j: a_j \geq b_{h-1,j}} w_j \geq \lambda \text{ and } \sum_{j: a_j \geq b_{h,j}} w_j < \lambda$$
MR-Sort applied to the examples

- Profile fixed at 10/20 on each criterion
- Admission condition: score above 10/20 in all the courses of one of the minimal winning coalitions:
  - \{\text{math, physics}\}
  - \{\text{math, chemistry}\}
  - \{\text{chemistry, history}\}

- Maximal losing coalitions:
  - \{\text{math, history}\}
  - \{\text{physics, chemistry}\}
  - \{\text{physics, history}\}

- \[ w_{\text{math}} + w_{\text{physics}} \geq \lambda \]
- \[ w_{\text{math}} + w_{\text{chemistry}} \geq \lambda \]
- \[ w_{\text{chemistry}} + w_{\text{history}} \geq \lambda \]
- \[ w_{\text{math}} + w_{\text{history}} < \lambda \]
- \[ w_{\text{physics}} + w_{\text{chemistry}} < \lambda \]
- \[ w_{\text{physics}} + w_{\text{history}} < \lambda \]

- \[ w_{\text{math}} + w_{\text{physics}} + w_{\text{chemistry}} + w_{\text{history}} = 1 \]
MR-Sort applied to the examples

- Profile fixed at 10/20 on each criterion
- Admission condition: score above 10/20 in all the courses of one the minimal winning coalitions:
  - \{math, physics\}
  - \{math, chemistry\}
  - \{chemistry, history\}
- Maximal loosing coalitions:
  - \{math, history\}
  - \{physics, chemistry\}
  - \{physics, history\}
- \( w_{\text{math}} + w_{\text{physics}} + w_{\text{chemistry}} + w_{\text{history}} = 1 \)
- \( w_{\text{math}} + w_{\text{physics}} \geq \lambda \) and \( w_{\text{chemistry}} + w_{\text{history}} \geq \lambda \) \( \Rightarrow \lambda \leq \frac{1}{2} \)
MR-Sort applied to the examples

- Profile fixed at 10/20 on each criterion
- Admission condition: score above 10/20 in all the courses of one the minimal winning coalitions:
  - \{math, physics\}
  - \{math, chemistry\}
  - \{chemistry, history\}

\[\begin{align*}
\Rightarrow \quad 
w_{\text{math}} + w_{\text{physics}} & \geq \lambda \\
\Rightarrow \quad 
w_{\text{math}} + w_{\text{chemistry}} & \geq \lambda \\
\Rightarrow \quad w_{\text{chemistry}} + w_{\text{history}} & \geq \lambda
\end{align*}\]

- Maximal losing coalitions:
  - \{math, history\}
  - \{physics, chemistry\}
  - \{physics, history\}

\[\begin{align*}
\Rightarrow \quad 
w_{\text{math}} + w_{\text{history}} & < \lambda \\
\Rightarrow \quad w_{\text{physics}} + w_{\text{chemistry}} & < \lambda \\
\Rightarrow \quad w_{\text{physics}} + w_{\text{history}} & < \lambda
\end{align*}\]

\[w_{\text{math}} + w_{\text{physics}} + w_{\text{chemistry}} + w_{\text{history}} = 1\]

\[w_{\text{math}} + w_{\text{physics}} \geq \lambda \quad \text{and} \quad w_{\text{chemistry}} + w_{\text{history}} \geq \lambda \Rightarrow \lambda \leq \frac{1}{2}\]

\[w_{\text{math}} + w_{\text{history}} < \lambda \quad \text{and} \quad w_{\text{physics}} + w_{\text{chemistry}} < \lambda \Rightarrow \lambda > \frac{1}{2}\]
MR-Sort applied to the examples

- Profile fixed at 10/20 on each criterion
- Admission condition: score above 10/20 in all the courses of one of the minimal winning coalitions:
  - \{math, physics\}
  - \{math, chemistry\}
  - \{chemistry, history\}
- Maximal losing coalitions:
  - \{math, history\}
  - \{physics, chemistry\}
  - \{physics, history\}

\[
\begin{align*}
\text{\textit{w}}_{\text{math}} + \text{\textit{w}}_{\text{physics}} & \geq \lambda \\
\text{\textit{w}}_{\text{math}} + \text{\textit{w}}_{\text{chemistry}} & \geq \lambda \\
\text{\textit{w}}_{\text{chemistry}} + \text{\textit{w}}_{\text{history}} & \geq \lambda \\
\text{\textit{w}}_{\text{math}} + \text{\textit{w}}_{\text{chemistry}} & < \lambda \\
\text{\textit{w}}_{\text{physics}} + \text{\textit{w}}_{\text{chemistry}} & < \lambda \\
\text{\textit{w}}_{\text{physics}} + \text{\textit{w}}_{\text{history}} & < \lambda
\end{align*}
\]

\[
\text{\textit{w}}_{\text{math}} + \text{\textit{w}}_{\text{physics}} + \text{\textit{w}}_{\text{chemistry}} + \text{\textit{w}}_{\text{history}} = 1
\]

\[
\text{\textit{w}}_{\text{math}} + \text{\textit{w}}_{\text{physics}} \geq \lambda \quad \text{and} \quad \text{\textit{w}}_{\text{chemistry}} + \text{\textit{w}}_{\text{history}} \geq \lambda \quad \Rightarrow \lambda \leq \frac{1}{2}
\]

\[
\text{\textit{w}}_{\text{math}} + \text{\textit{w}}_{\text{history}} < \lambda \quad \text{and} \quad \text{\textit{w}}_{\text{physics}} + \text{\textit{w}}_{\text{chemistry}} < \lambda \quad \Rightarrow \lambda > \frac{1}{2}
\]

- Impossible to represent all the coalitions with a MR-Sort model
1 Introductory example

2 MR-Sort

3 Capacitive MR-Sort

4 Learning a Capacitive MR-Sort model

5 Experimentations

6 Comments and Conclusion
Capacitive MR-Sort

Characteristic

- Take criteria interactions into account
- Improvement of the expressivity of the model
- Non Compensatory Sorting Model [Bouyssou and Marchant, 2007]

Capacity

- $F = \{1, \ldots, n\}$ : set of criteria
- A capacity is a function $\mu : 2^F \rightarrow [0, 1]$ such that:
  - $\mu(B) \geq \mu(A)$, for all $A \subseteq B \subseteq F$ (monotonicity);
  - $\mu(\emptyset) = 0$ and $\mu(F) = 1$ (normalization).

New assignment rule

$$a \in C_h \iff \mu(\{j \in F : a_j \geq b_{h-1,j}\}) \geq \lambda \quad \text{and} \quad \mu(\{j \in F : a_j \geq b_{h,j}\}) < \lambda$$
1 Introductory example

2 MR-Sort

3 Capacitive MR-Sort

4 Learning a Capacitive MR-Sort model

5 Experimentations

6 Comments and Conclusion
Learning a Capacitive MR-Sort model

Mixed Integer Programming

- Objective: Finding a model compatible with as much example as possible
- MIP to learn an MR-Sort model in [Leroy et al., 2011]
- Limitation to 2-additive capacities
- For Capacitive MR-Sort, more constraints and binary variable are required

Table: Max number of constraints

<table>
<thead>
<tr>
<th></th>
<th>MIP MR-Sort</th>
<th>MIP Capacitive MR-Sort</th>
</tr>
</thead>
<tbody>
<tr>
<td># binary variables</td>
<td>(n(2m + 1))</td>
<td>(n(2m + 1 + 2m(m + 1)))</td>
</tr>
<tr>
<td># constraints</td>
<td>(2n(5m + 1) + n(p - 3) + 1)</td>
<td>(2n(5m + 1) + n(p - 3) + 1 + 2m(n^2 + 1) + n^2)</td>
</tr>
</tbody>
</table>

- Too much variables and constraints to be used with large datasets
Learning a Capacitive MR-Sort model - MIP II

Application to the introductory example

- Admission condition: score above 10/20 in all the courses of one of these coalitions:
  - \{math, physics\}
  - \{math, chemistry\}
  - \{chemistry, history\}

- MIP is able to find a model matching all the rules

\[
\begin{array}{c|c}
J & m(J) \\
\hline
\{\text{math}\} & 0 \\
\{\text{physics}\} & 0 \\
\{\text{chemistry}\} & 0 \\
\{\text{history}\} & 0 \\
\hline
\lambda = 0.3
\end{array}
\]

\[
\begin{array}{c|c}
J & m(J) \\
\hline
\{\text{math, physics}\} & 0.3 \\
\{\text{math, chemistry}\} & 0.3 \\
\{\text{math, history}\} & 0 \\
\{\text{physics, chemistry}\} & 0 \\
\{\text{physics, history}\} & 0 \\
\{\text{chemistry, history}\} & 0.4 \\
\hline
\end{array}
\]
Learning a Capacitive MR-Sort model - Meta I

Metaheuristic to learn a Capacitive MR-Sort model

- Objective: Finding a model compatible with as much example as possible
- Being able to handle large datasets

Recall: Metaheuristic to learn parameters of a MR-Sort model


Recall: Metaheuristic to learn a MR-Sort model

- Principle (genetic algorithm):
  - Initialize a population of MR-Sort models
  - Evolve the population by iteratively
    - Optimizing weights (profiles fixed) with a LP
    - Improving profiles (weights fixed) with a heuristic
    - Selecting the best models and reinitializing the others
  - ... to get a “good” MR-Sort model in the population

- Stopping criteria:
  - If one of the models restores all examples
  - Or after $N$ iterations
Recall: Metaheuristic to learn a MR-Sort model

- Principle (genetic algorithm):
  - Initialize a population of MR-Sort models
  - Evolve the population by iteratively
    - Optimizing weights (profiles fixed) with a LP
    - Improving profiles (weights fixed) with a heuristic
    - Selecting the best models and reinitializing the others
  - ... to get a “good” MR-Sort model in the population

- Stopping criteria:
  - If one of the models restores all examples
  - Or after $N$ iterations

Metaheuristic to learn a Capacitive MR-Sort model

- Adaptation of the LP to learn capacities and adaptation of the heuristic
Learning a Capacitive MR-Sort model

Linear Program to learn the capacities and the majority threshold

- Fixed profiles
- Expression of the capacities with the Möbius transform
  \[ \mu(A) = \sum_{B \subseteq A} m(B) \], for all \( A \subseteq F \), with \( m(B) \) defined as:
  \[ m(B) = \sum_{C \subseteq B} (-1)^{|B| - |C|} \mu(C) \]
- Limitation to 2-additive capacities in view of limiting the number of variables and constraints
  \[ \mu(A) = \sum_{i \in A} m\{i\} + \sum_{\{i,j\} \in A} m\{i,j\} \]
- Minimization of a slack that tends to be equal to 0 when all examples are correctly assigned
Learning a Capacitive MR-Sort model - Meta III

Linear Program to learn the capacities and the majority threshold

\[
\begin{aligned}
\min_{\sum_{a \in A}} & \quad (x'_a + y'_a) \\
\text{s.t.} & \quad \sum_{j: a_j \geq b_{h-1}, j} \left( m_j + \sum_{k: a_k \geq b_{h-1}, k} m_{j,k} \right) - x_a + x'_a = \lambda \quad \forall a \in A_h, \forall h \in P \setminus \{1\} \\
& \quad \sum_{j: a_j \geq b_h, j} \left( m_j + \sum_{k: a_k \geq b_h, k} m_{j,k} \right) + y_a - y'_a = \lambda - \varepsilon \quad \forall a \in A_h, \forall h \in P \setminus \{p - 1\} \\
& \quad \sum_{j=1}^{n} m_j + \sum_{j=1}^{n} \sum_{k=1}^{j} m_{j,k} = 1 \\
& \quad m_j + \sum_{k \in J} m_{j,k} \geq 0 \quad \forall j \in F, \forall J \subseteq F \setminus \{j\} \\
& \quad \lambda \in [0.5; 1] \\
& \quad m_j \in [0, 1] \quad \forall j \in F \\
& \quad m_{j,k} \in [-1, 1] \quad \forall j \in F, \forall k \in F, k < j \\
& \quad x_a, y_a, x'_a, y'_a \in \mathbb{R}_0^+ \quad a \in A.
\end{aligned}
\]
Heuristic to adjust the profiles

- Fixed Möbius indices and majority threshold
- Principle of the heuristic: moving the profile in view of increasing the number of alternatives correctly assigned
- Multiple iterations over each profile and each criterion
- Same principles as in [Sobrie et al., 2013], adapted for capacities instead of weights
1 Introductory example
2 MR-Sort
3 Capacitive MR-Sort
4 Learning a Capacitive MR-Sort model
5 Experimentations
6 Comments and Conclusion
## Experimentations I

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#instances</th>
<th>#attributes</th>
<th>#categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBS</td>
<td>120</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>CPU</td>
<td>209</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>BCC</td>
<td>286</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>MPG</td>
<td>392</td>
<td>7</td>
<td>36</td>
</tr>
<tr>
<td>ESL</td>
<td>488</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>MMG</td>
<td>961</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>ERA</td>
<td>1000</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>LEV</td>
<td>1000</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>CEV</td>
<td>1728</td>
<td>6</td>
<td>4</td>
</tr>
</tbody>
</table>

- Instances split in two parts: learning and generalization sets
- Binarization of the categories

Source: [Tehrani et al., 2012]
## Experimentations II

### Average Classification Accuracy

<table>
<thead>
<tr>
<th>Dataset</th>
<th>META MR-Sort</th>
<th>META Capa-MR-Sort</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBS</td>
<td>0.8400 ± 0.0456</td>
<td>0.8306 ± 0.0466</td>
</tr>
<tr>
<td>CPU</td>
<td>0.9270 ± 0.0294</td>
<td>0.9203 ± 0.0315</td>
</tr>
<tr>
<td>BCC</td>
<td>0.7271 ± 0.0379</td>
<td>0.7262 ± 0.0377</td>
</tr>
<tr>
<td>MPG</td>
<td>0.8174 ± 0.0290</td>
<td>0.8167 ± 0.0468</td>
</tr>
<tr>
<td>ESL</td>
<td>0.8992 ± 0.0195</td>
<td>0.9018 ± 0.0172</td>
</tr>
<tr>
<td>MMG</td>
<td>0.8303 ± 0.0154</td>
<td>0.8318 ± 0.0121</td>
</tr>
<tr>
<td>ERA</td>
<td>0.6905 ± 0.0192</td>
<td>0.6927 ± 0.0165</td>
</tr>
<tr>
<td>LEV</td>
<td>0.8454 ± 0.0221</td>
<td>0.8445 ± 0.0223</td>
</tr>
<tr>
<td>CEV</td>
<td>0.9217 ± 0.0067</td>
<td>0.9187 ± 0.0153</td>
</tr>
</tbody>
</table>

- 50% of the dataset used as learning set
- Results are not convincing, overfitting?
## Experimentations II

### Average Classification Accuracy

<table>
<thead>
<tr>
<th>Dataset</th>
<th>META MR-Sort</th>
<th>META Capa-MR-Sort</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBS</td>
<td>0.9318 ± 0.0036</td>
<td>0.9247 ± 0.0099</td>
</tr>
<tr>
<td>CPU</td>
<td>0.9761 ± 0.0000</td>
<td>0.9694 ± 0.0072</td>
</tr>
<tr>
<td>BCC</td>
<td>0.7737 ± 0.0013</td>
<td>0.7700 ± 0.0077</td>
</tr>
<tr>
<td>MPG</td>
<td>0.8418 ± 0.0000</td>
<td>0.8418 ± 0.0000</td>
</tr>
<tr>
<td>ESL</td>
<td>0.9180 ± 0.0000</td>
<td>0.9180 ± 0.0000</td>
</tr>
<tr>
<td>MMG</td>
<td>0.8491 ± 0.0011</td>
<td>0.8508 ± 0.0005</td>
</tr>
<tr>
<td>ERA</td>
<td>0.7142 ± 0.0028</td>
<td>0.7158 ± 0.0004</td>
</tr>
<tr>
<td>LEV</td>
<td>0.8650 ± 0.0000</td>
<td>0.8650 ± 0.0000</td>
</tr>
<tr>
<td>CEV</td>
<td>0.9225 ± 0.0000</td>
<td>0.9225 ± 0.0000</td>
</tr>
</tbody>
</table>

- Full dataset used as learning set
- Results are not convincing
1 Introductory example

2 MR-Sort

3 Capacitive MR-Sort

4 Learning a Capacitive MR-Sort model

5 Experimentations

6 Comments and Conclusion
What to conclude after the experiments?

- Expressivity of the model is not so much improved?
- Algorithm not well adapted?
- To what extent MR-Sort approximates non-additive learning sets?
Comments II

To what extent MR-Sort approximates non-additive learning sets?

- Generation of $2^n$ binary vectors of performances
- Generation of Capacitive MR-Sort model non-additive and assignment
- Learning of a MR-Sort model from assignment
- Test with all the non-additive models
### Comments III

To what extent MR-Sort approximates non-additive learning sets?

<table>
<thead>
<tr>
<th>$n$</th>
<th>$D(n)$</th>
<th>% non-additive</th>
<th>0/1 loss</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>min.</td>
</tr>
<tr>
<td>4</td>
<td>168</td>
<td>11 %</td>
<td>1/16</td>
</tr>
<tr>
<td>5</td>
<td>7 581</td>
<td>57 %</td>
<td>1/32</td>
</tr>
<tr>
<td>6</td>
<td>7 828 354</td>
<td>97 %</td>
<td>1/64</td>
</tr>
</tbody>
</table>

- Few alternatives are incorrectly assigned
Conclusion

- For problems involving small number of criteria (< 7), we don’t win so much in expressivity with Capacitive MR-Sort
- Metaheuristic can be improved to better deal with interactions
- Tests with datasets in which there exist interactions between criteria
Thank you for your attention!
References I


