Learning preferences with multiple-criteria models

Olivier Sobrie

Université Paris-Saclay - CentraleSupélec
Université de Mons - Faculté polytechnique

June 21, 2016

Preferences

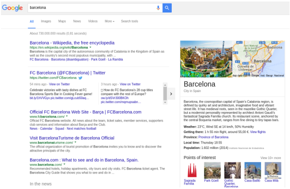
Preferences problems - some examples

Sorting of hotels

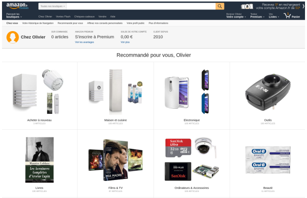
Choice of a pair of shoes

Preference learning - some examples

Google

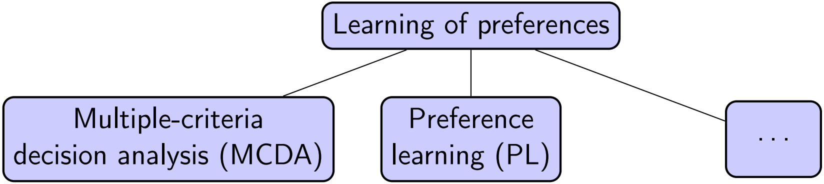


Amazon



Learning the preferences

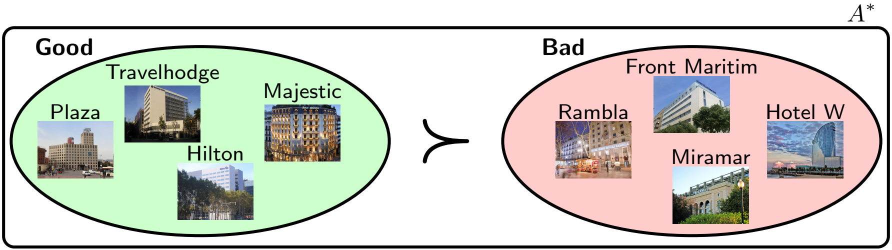
- **Hot topic** in last years
- **Several research communities** study the learning of preferences



- Examples of **sorting problems** (ordered classification) treated in MCDA and PL

Example of MCDA sorting problem I

- **Maria** (DM) has to choose for an accommodation for her next holidays in Barcelona
- She **sorts** a **small subset** of accommodations A^* in two **ordered** sets : "Bad" and "Good"



- She wants to obtain a **full sorting** of all the hotels in Barcelona
- She asks for the support of a **decision analyst**

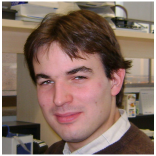
Example of MCDA sorting problem II

Decision maker (DM)

asks questions

provides preference information

Decision analyst (DA)

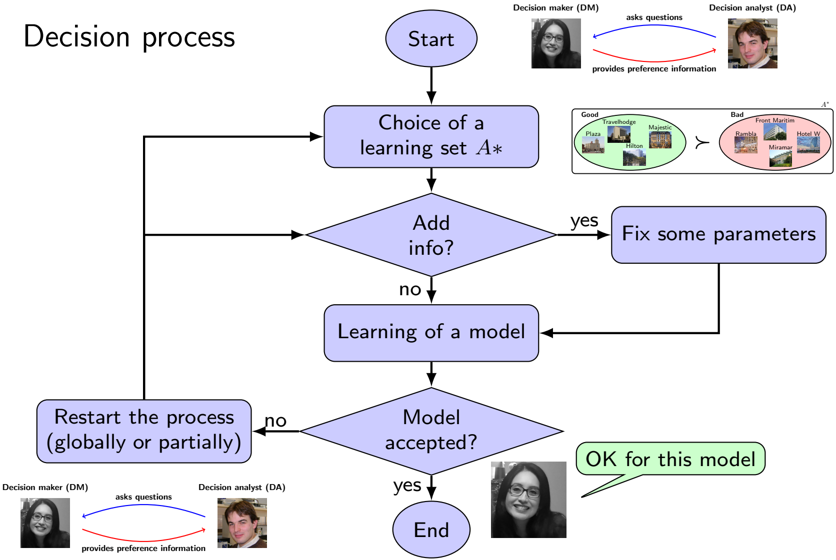


- The DA **helps** Maria identifying the criteria that amount for her

					...
distance to the beach	600m	300m	50m	200m	...
distance to the center	500m	100m	600m	300m	...
price	150€	130€	90€	80€	...
size	45m^2	35m^2	30m^2	25m^2	...
rating	★★★★★	★★★★	★★★	★★	...

Example of MCDA sorting problem III

Decision process



Example of PL sorting problem I

- From a **large** database, we would like to have a **model** predicting the health status of a patient before anesthesia

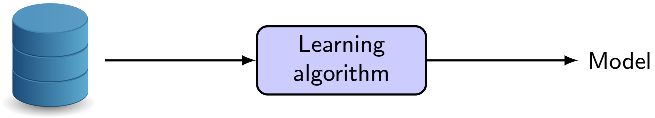
- **Database** built from different data sources
- Data generated by a **ground truth**
- The database contains ± 1000 **patients**
- Patients are **evaluated on attributes** and **assigned to a category** reflecting their health status

- Categories are **ordered** (ASA score)

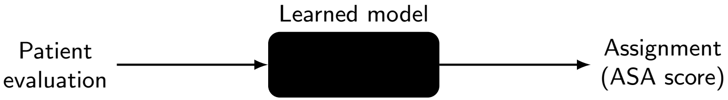
Healthy $\succ$ Mild systemic disease $\succ$ Sever systemic disease $\succ$ Incapaciting systemic disease $\succ$ Moribound

Example of PL sorting problem II

- The database is given as input to a **learning algorithm**



- The model learned is then used as a **blackbox** for predicting the assignments of other patients

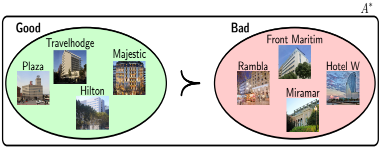


- The performance of the model and learning algorithm are assessed using indicators such as **classification accuracy**, **area under the curve**, etc.

MCDA versus PL

Multiple criteria decision analysis

- **Small** datasets



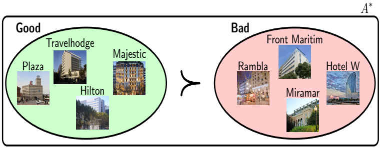
Preference learning

- **Large** datasets

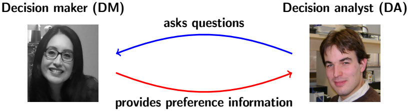
MCDA versus PL

Multiple criteria decision analysis

- **Small** datasets



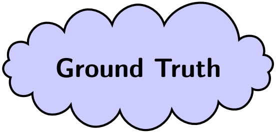
- **Strong** interactions



Preference learning

- **Large** datasets

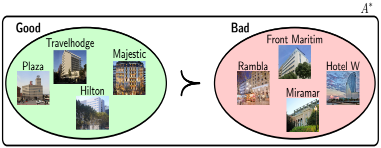
- **No/little** interactions



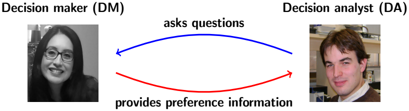
MCDA versus PL

Multiple criteria decision analysis

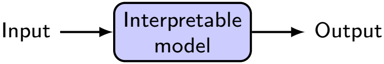
- **Small** datasets



- **Strong** interactions



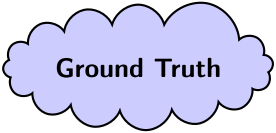
- **Interpretable** models



Preference learning

- **Large** datasets

- **No/little** interactions



- **Blackbox** models

Aim of this thesis

Make some links between MCDA and PL

Use MCDA models to deal with PL problems
(outranking models and additive value function models)

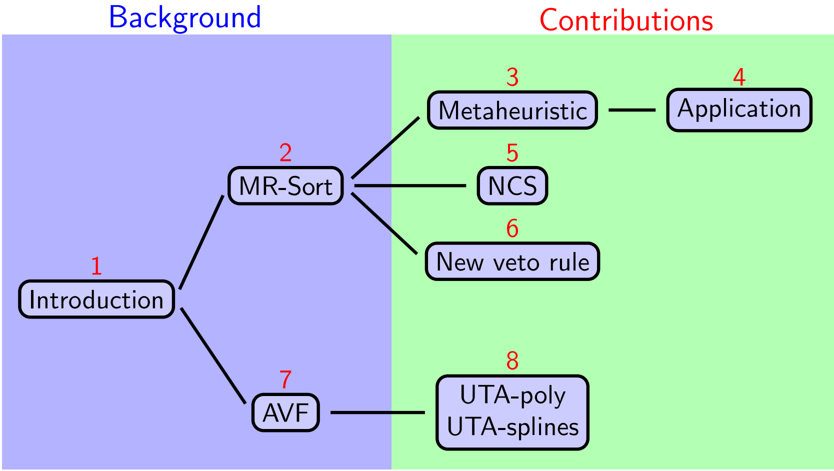
Validation of the learning algorithms as done in PL

Test the algorithms and models on a real application

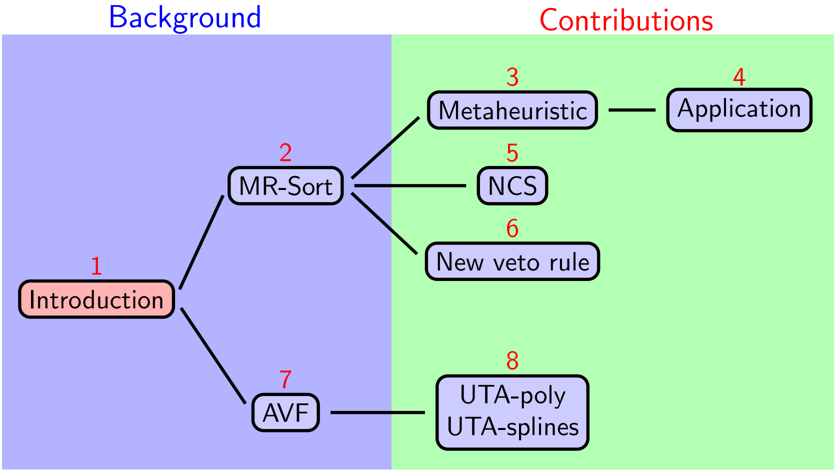
Study the expressivity of the MCDA models

Bring new techniques in MCDA and PL

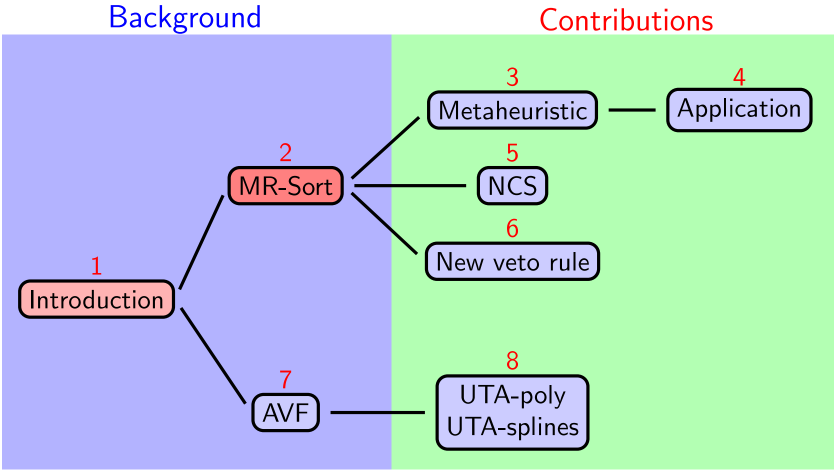
Outline of the presentation



Outline of the presentation

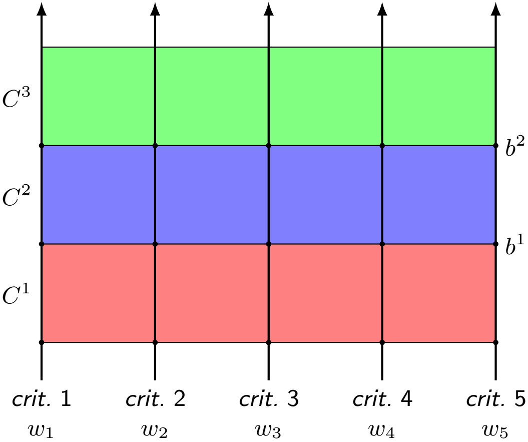


Outline of the presentation



Majority rule sorting model

- Sorting model (p **ordered categories**, i.e. $C^p \succ C^{p-1} \succ \ldots \succ C^1$)
- Axiomatized by Bouyssou and Marchant (2007a,b)



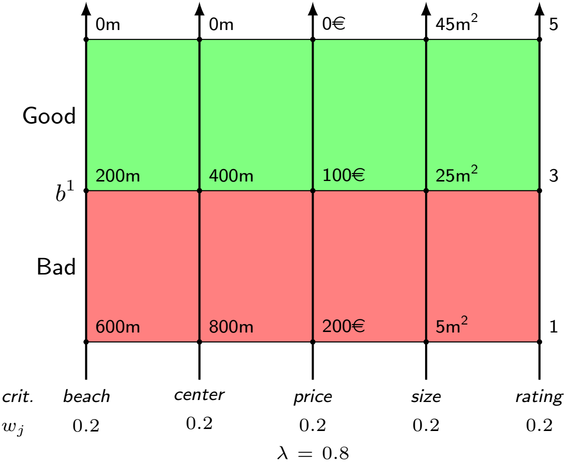
- n **weights** $(w_1, \ldots, w_n)$
- 1 **majority threshold** (λ)
- $p-1$ **profiles** $(b^1, \ldots, b^{p-1})$

Assignment rule

$$a \in C^h \iff \sum_{j: a_j \geq b_j^{h-1}} w_j \geq \lambda \text{ and } \sum_{j: a_j \geq b_j^h} w_j < \lambda$$

MR-Sort applied to Maria's decision problem

- **Sorting** accommodations in two categories : **Good** and **Bad**

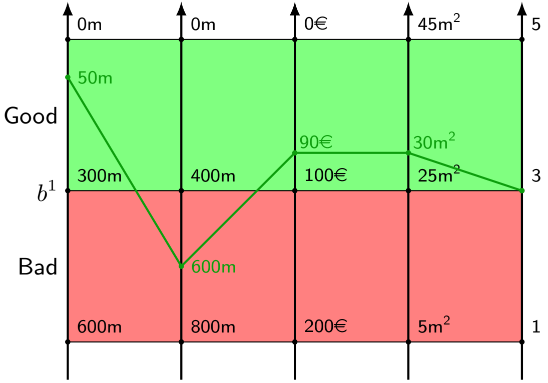


Assignment rule

$$\text{hotel} \in \text{Good} \Leftrightarrow \sum_{j : a_j \geq b_j^1} w_j \geq \lambda$$

MR-Sort applied to Maria's decision problem

- **Sorting** accommodations in two categories : **Good** and **Bad**



crit.	*beach*	*center*	*price*	*size*	*rating*
w_j	0.2	0.2	0.2	0.2	0.2

$\lambda = 0.8$

Assignment rule

$$\text{hotel} \in \text{Good} \Leftrightarrow \sum_{j:a_j \geq b_j^1} w_j \geq \lambda$$

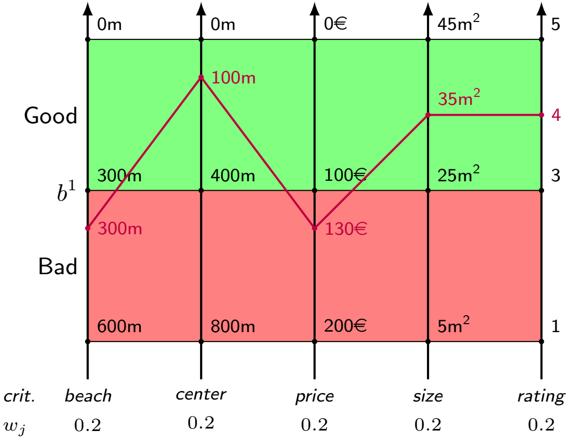
Hilton

$\in$ Good

$$\sum_{j:a_j \geq b_j^1} w_j = 0.8$$

MR-Sort applied to Maria's decision problem

- **Sorting** accommodations in two categories : **Good** and **Bad**



$\lambda = 0.8$

Assignment rule

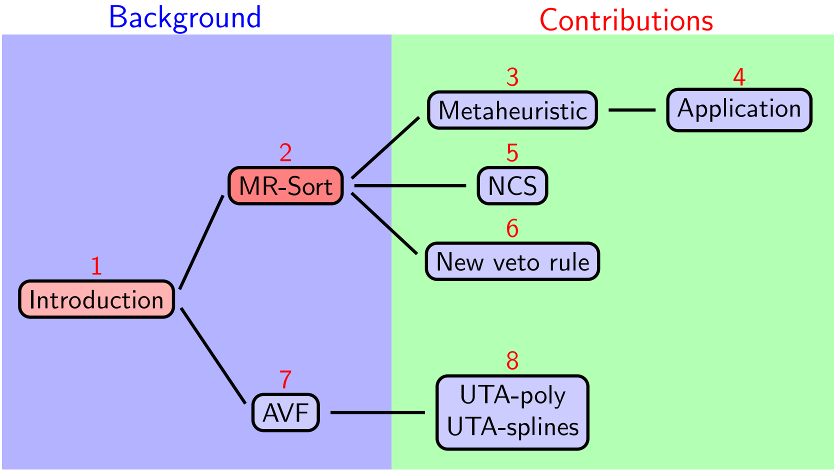
$$\text{hotel} \in \text{Good} \Leftrightarrow \sum_{j : a_j \geq b_j^1} w_j \geq \lambda$$

Plaza

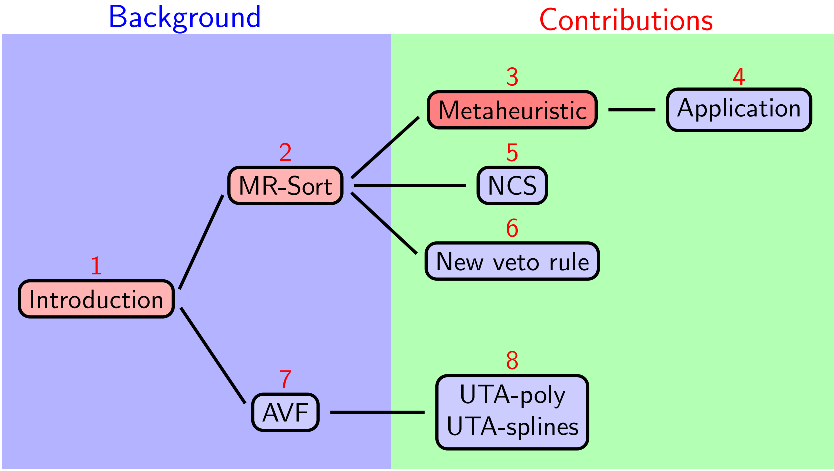
$\in$ Bad

$$\sum_{j : a_j \geq b_j^1} w_j = 0.6$$

Outline of the presentation

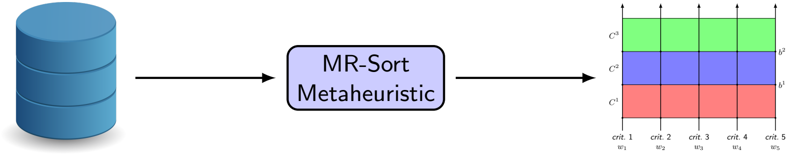


Outline of the presentation



Learning a MR-Sort model

Objective



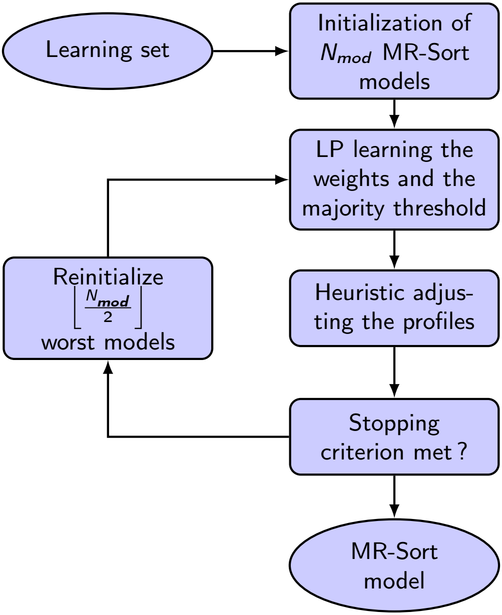
Previous research for learning a MR-Sort model

- MIP by Leroy et al. (2011) $\rightarrow$ inefficient for large datasets
- Learning the weights and majority threshold $\rightarrow$ easy (LP)
- Learning the profiles $\rightarrow$ difficult (MIP)

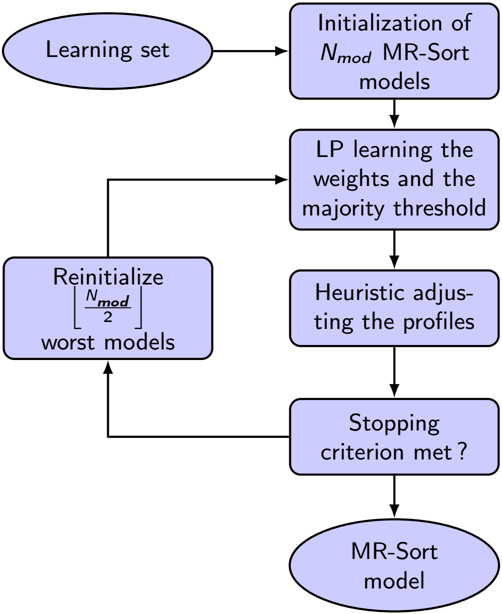
Strategy

Metaheuristic which takes advantage of the ease of learning the weights and leverages the difficulty for learning the profiles

Metaheuristic for learning a MR-Sort model

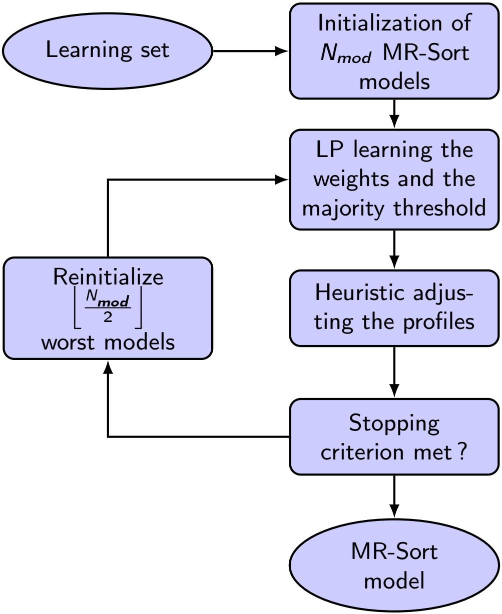


Metaheuristic for learning a MR-Sort model



Profiles initialized with a heuristic with some randomness

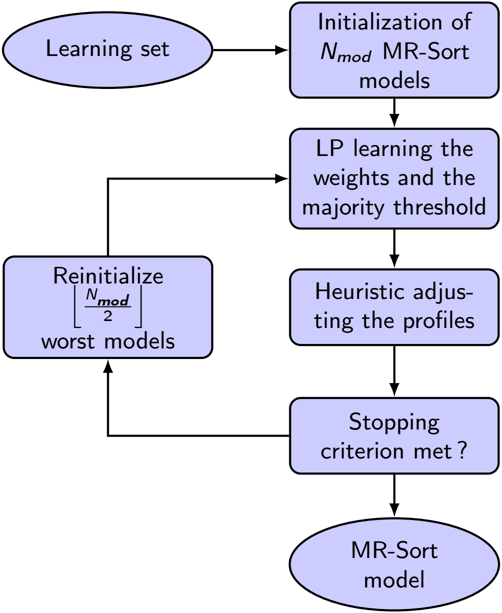
Metaheuristic for learning a MR-Sort model



Profiles initialized with a heuristic with some randomness

Fixed profiles
Maximization of the CA

Metaheuristic for learning a MR-Sort model

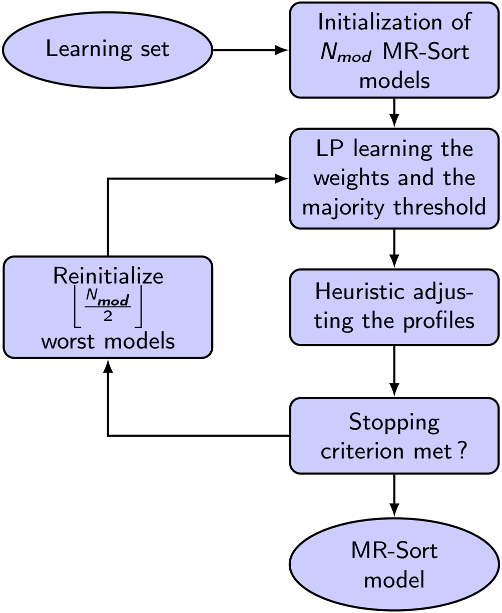


Profiles initialized with a heuristic with some randomness

Fixed profiles
Maximization of the CA

Fixed weights and majority threshold
Maximization of the CA

Metaheuristic for learning a MR-Sort model



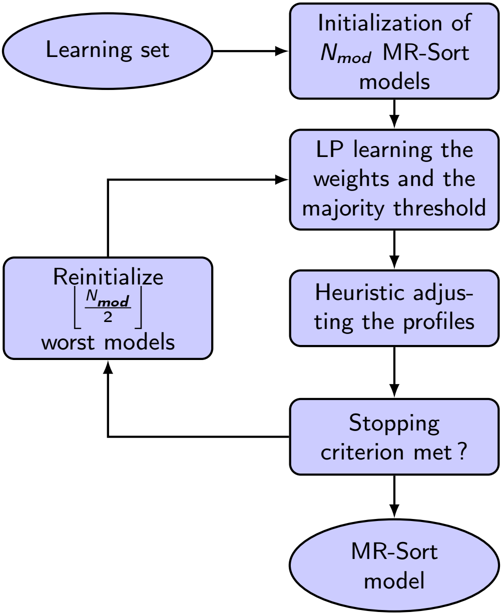
Profiles initialized with a heuristic with some randomness

Fixed profiles
Maximization of the CA

Fixed weights and majority threshold
Maximization of the CA

Once a model restores all the assignment examples or after N_{it} iterations

Metaheuristic for learning a MR-Sort model



Profiles initialized with a heuristic with some randomness

Fixed profiles
Maximization of the CA

Fixed weights and majority threshold
Maximization of the CA

Once a model restores all the assignment examples or after N_{it} iterations

The best model regarding CA or AUC is returned

Test with PL datasets I

- Datasets issued from the **PL field**
- Categories have been **binarized** by thresholding at the median
- Split in **learning** and **test** sets

Data set	#instances	#attributes	#categories
DBS	120	8	2
CPU	209	6	4
BCC	286	7	2
MPG	392	7	36
ESL	488	4	9
MMG	961	5	2
ERA	1000	4	4
LEV	1000	4	5
CEV	1728	6	4

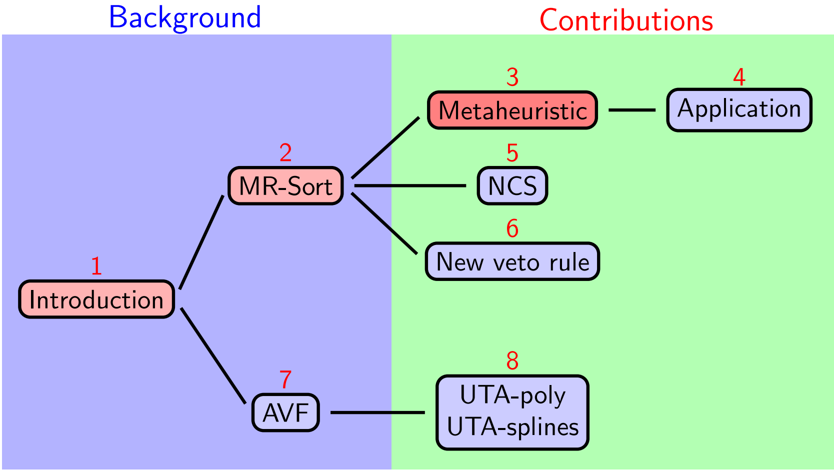
Tests with PL datasets II

Size	Data set	META	MIP	UTADIS	CR
20 %	DBS	18.97 ± 4.23	19.77 ± 4.81	20.08 ± 5.33	17.13 ± 4.24
	CPU	9.94 ± 3.23	9.00 ± 3.45	6.52 ± 3.62	8.11 ± 1.03
	BCC	28.24 ± 2.73	26.78 ± 2.76	29.15 ± 3.07	27.75 ± 3.35
	MPG	20.25 ± 3.56	20.80 ± 3.26	22.25 ± 3.18	7.09 ± 1.93
	ESL	10.42 ± 1.71	10.75 ± 1.58	8.89 ± 1.60	6.82 ± 1.29
	MMG	16.97 ± 0.87	17.16 ± 1.40	18.40 ± 1.84	17.25 ± 1.20
	ERA	21.36 ± 2.05	20.93 ± 1.74	23.68 ± 1.87	28.89 ± 2.73
	LEV	16.74 ± 1.87	16.08 ± 1.73	16.54 ± 1.60	14.99 ± 1.22
	CEV	9.37 ± 1.12	-	7.94 ± 0.59	4.48 ± 0.89
50 %	DBS	16.23 ± 4.69	16.27 ± 4.26	14.80 ± 4.21	15.72 ± 4.16
	CPU	6.75 ± 2.37	6.40 ± 2.39	2.30 ± 2.38	4.64 ± 2.81
	BCC	27.50 ± 3.17	-	28.54 ± 2.46	26.87 ± 2.82
	MPG	17.81 ± 2.37	-	20.90 ± 2.36	5.77 ± 2.51
	ESL	10.04 ± 1.86	10.18 ± 1.55	7.83 ± 1.63	6.01 ± 1.26
	MMG	17.32 ± 1.51	-	17.58 ± 1.52	16.67 ± 1.44
	ERA	20.56 ± 1.73	19.58 ± 1.37	23.42 ± 1.71	28.44 ± 3.06
	LEV	15.92 ± 1.22	14.22 ± 1.54	15.56 ± 1.32	13.72 ± 1.25
	CEV	9.36 ± 1.19	-	7.99 ± 0.91	3.76 ± 0.59
80 %	DBS	15.92 ± 6.98	14.80 ± 8.11	12.80 ± 5.01	14.16 ± 6.81
	CPU	6.40 ± 3.04	5.98 ± 3.15	1.52 ± 2.14	2.12 ± 3.01
	BCC	26.77 ± 5.47	-	29.13 ± 5.10	24.96 ± 4.85
	MPG	16.86 ± 3.69	-	20.80 ± 3.88	5.51 ± 1.60
	ESL	10.01 ± 2.97	10.08 ± 2.47	7.44 ± 2.35	5.42 ± 2.18
	MMG	16.98 ± 2.79	-	17.34 ± 2.65	15.84 ± 2.51
	ERA	20.31 ± 2.50	18.56 ± 2.60	23.56 ± 2.92	28.13 ± 2.80
	LEV	16.16 ± 2.22	13.59 ± 1.85	15.72 ± 2.22	13.14 ± 1.76
	CEV	9.66 ± 1.74	-	7.99 ± 1.32	2.73 ± 0.89

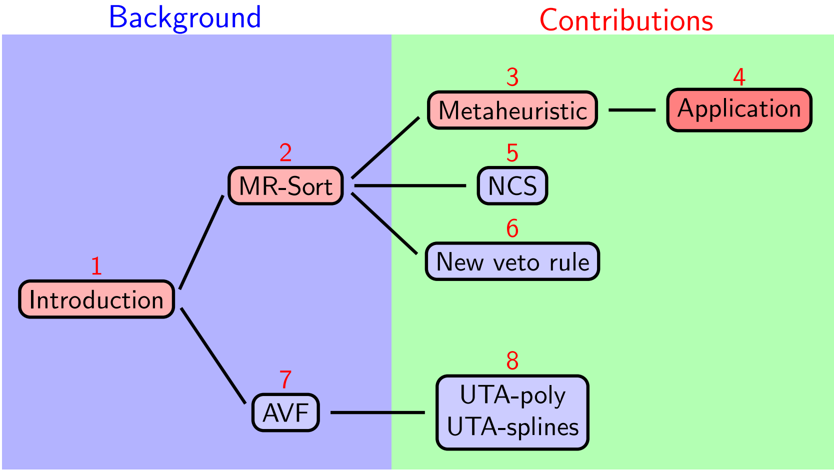
Contributions

- Sobrie, O., Mousseau, V., and Pirlot, M. (2012). Learning the parameters of a multiple criteria sorting method from large sets of assignment examples.
  In *DA2PL 2012 Workshop From Multiple Criteria Decision Aid to Preference Learning*, pages 21–31.
  Mons, Belgique
- Sobrie, O., Mousseau, V., and Pirlot, M. (2013). Learning a majority rule model from large sets of assignment examples.
  In Perny, P., Pirlot, M., and Tsoukiás, A., editors, *Algorithmic Decision Theory*, volume 8176 of *Lecture Notes in Artificial Intelligence*, pages 336–350, Brussels, Belgium. Springer

Outline of the presentation

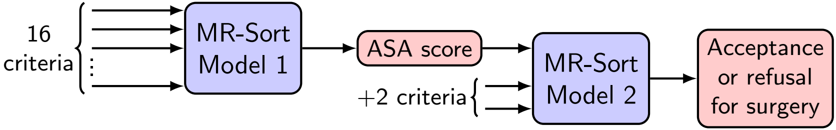


Outline of the presentation



Application

- Medical application : **prediction of the ASA score** and **acceptance or refusal for surgery** from a database containing **898 patients**

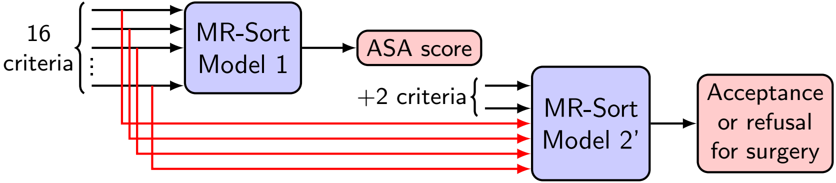


- Results have been compared to **other machine learning algorithms**

Learning algorithm	ASA score	A/R (3 criteria)
SVM	0.8752	0.9142
C4.5	0.9154	0.9012
KNN	0.8468	0.9085
MLP	0.8927	0.9292
RBF	0.8333	0.8981
Majority voting	0.9259	0.9407
MR-Sort	0.9615	0.9235

Application

- Medical application : **prediction of the ASA score** and **acceptance or refusal for surgery** from a database containing **898 patients**



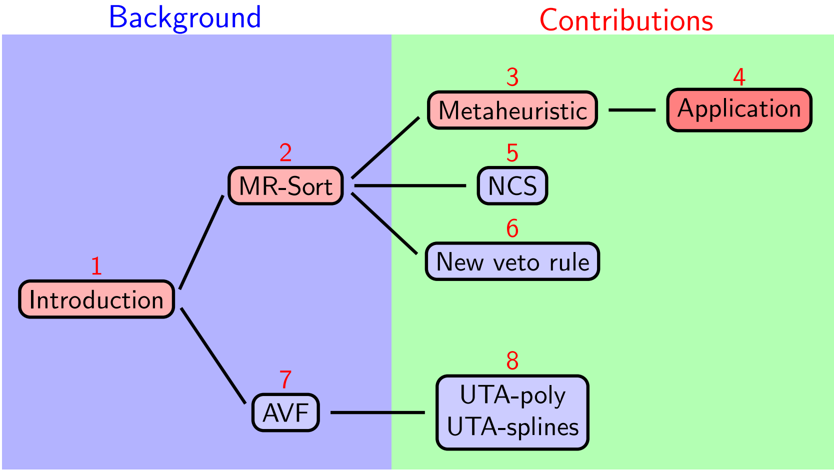
- Results have been compared to **other machine learning algorithms**

Learning algorithm	ASA score	A/R (3 criteria)	A/R (18 criteria)
SVM	0.8752	0.9142	
C4.5	0.9154	0.9012	
KNN	0.8468	0.9085	
MLP	0.8927	0.9292	
RBF	0.8333	0.8981	
Majority voting	0.9259	0.9407	
MR-Sort	0.9615	0.9235	0.9525

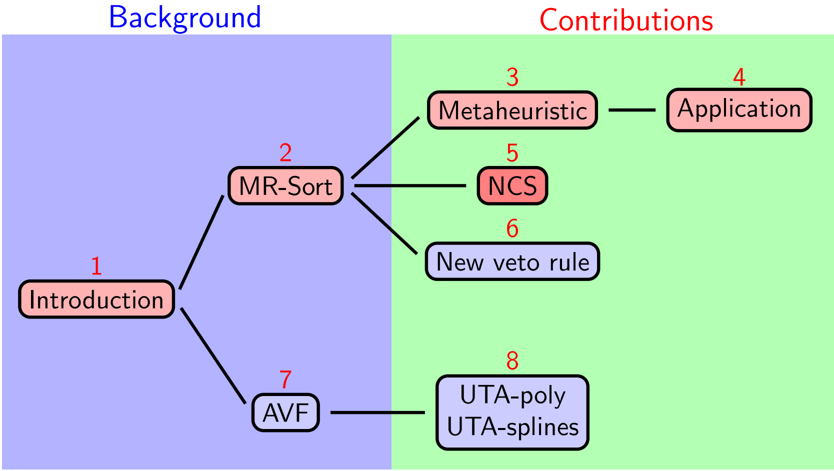
Contributions

- Sobrie, O., Lazouni, M. E. A., Mahmoudi, S., Mousseau, V., and Pirlot, M. (2016b). A new decision support model for preanesthetic evaluation.
  Computer Methods and Programs in Biomedicine.
  Accepted

Outline of the presentation



Outline of the presentation



NCS model - Learning and expressivity I

Improvement of the expressivity of MR-Sort

- MR-Sort is not able to take **criteria interactions** into account
- We added **capacities** in the outranking rule → **NCS model**

NCS model - Learning and expressivity I

Improvement of the expressivity of MR-Sort

- MR-Sort is not able to take **criteria interactions** into account
- We added **capacities** in the outranking rule → **NCS model**

Learning a NCS model

- **MIP** : only usable for small datasets
- **Metaheuristic** : modification of the MR-Sort metaheuristic
- Test with **PL datasets** → Performances are **not** much improved

NCS model - Learning and expressivity I

Improvement of the expressivity of MR-Sort

- MR-Sort is not able to take **criteria interactions** into account
- We added **capacities** in the outranking rule → **NCS model**

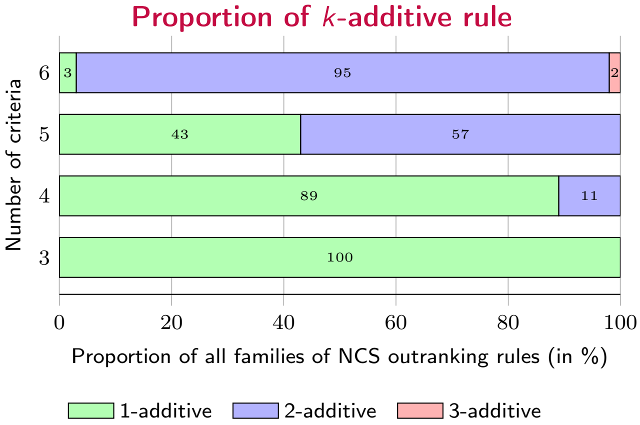
Learning a NCS model

- **MIP** : only usable for small datasets
- **Metaheuristic** : modification of the MR-Sort metaheuristic
- Test with **PL datasets** → Performances are **not** much improved

Study of the expressivity of the model

- **Proportion** of NCS outranking rule that cannot be represented by 1-additive weights and a threshold ?
- How can we **approximate** non 1-additive rules by a set of 1-additive weights and threshold ?

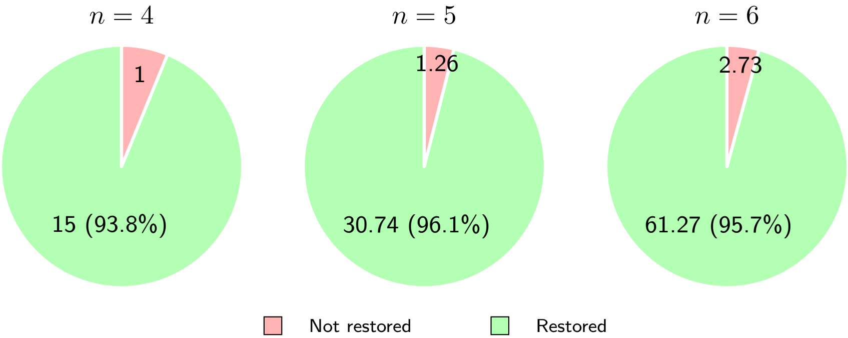
NCS model - Learning and expressivity II



NCS model - Learning and expressivity III

Approximation of a k-additive rule ($k > 1$) by a 1-additive rule

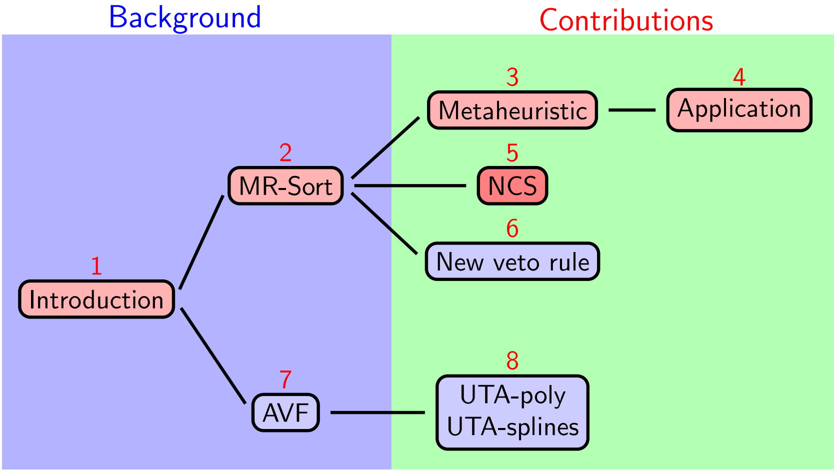
- Generation of **all possible inputs** (2^n) regarding a fixed profile
- **Assignment** of these inputs in two categories using a **k-additive rule**
- MIP inferring a **1-additive rule**



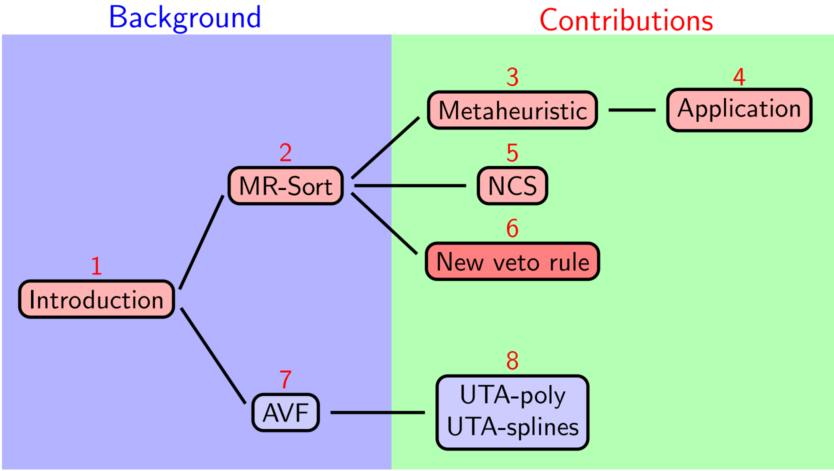
Contributions

- Sobrie, O., Mousseau, V., and Pirlot, M. (2015). Learning the parameters of a non compensatory sorting model.
  In Walsh, T., editor, *Algorithmic Decision Theory*, volume 9346 of *Lecture Notes in Artificial Intelligence*, pages 153–170, Lexington, KY, USA. Springer
- Ersek Uyanık, E., Sobrie, O., Mousseau, V., and Pirlot, M. (2016).
  Families of sufficient coalitions of criteria involved in ordered classification procedures.
  Submitted

Outline of the presentation

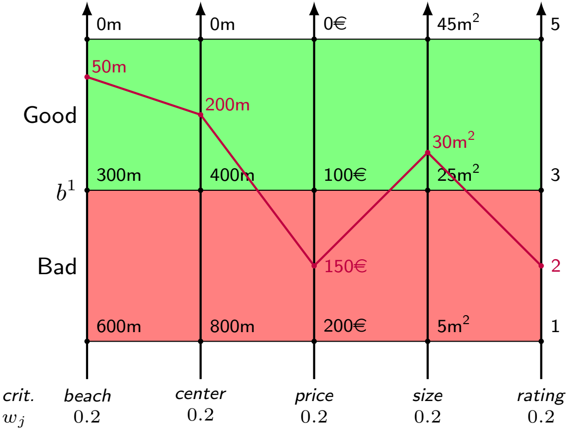


Outline of the presentation



MR-Sort without veto

- **Best** possible MR-Sort model (CA) regarding the learning set



$\lambda = 0.6$

Assignment rule

hotel ∈ Good

$$\Leftrightarrow$$

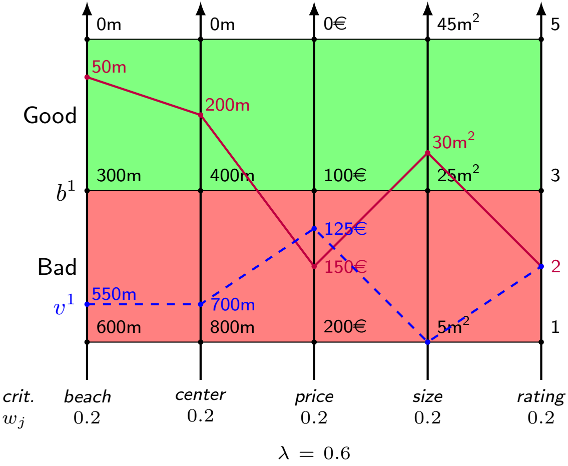
$$\sum_{j : a_j \geq b_j^1} w_j \geq \lambda$$

Rambla

∈ ~~Bad~~

Binary veto rule

- Veto if alternative worse than the **veto profile** on **any criterion**



Assignment rule

hotel $\in$ Good

$\Leftrightarrow$

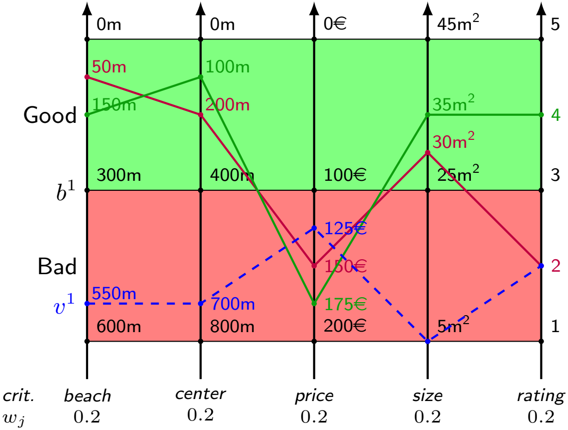
$$\sum_{j : a_j \geq b_j^1} w_j \geq \lambda \text{ and } \nexists j : a_j \leq v_j^1$$

Rambla

$\in$ **Bad**

Binary veto rule

- Veto if alternative worse than the **veto profile** on **any criterion**



$\lambda = 0.6$

Assignment rule

hotel $\in$ Good

$\Leftrightarrow$

$$\sum_{j : a_j \geq b^1_j} w_j \geq \lambda \text{ and } \nexists j : a_j \leq v^1_j$$

Rambla

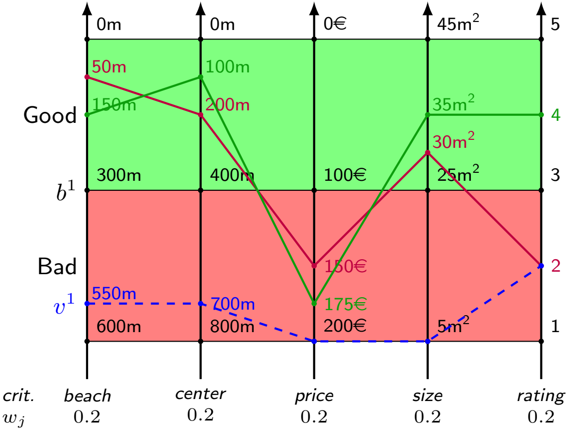
$\in$ Bad

Majestic

$\in$ ~~Good~~

Binary veto rule

- Veto if alternative worse than the **veto profile** on **any criterion**



$\lambda = 0.6$

Assignment rule

hotel $\in$ Good

$\Leftrightarrow$

$$\sum_{j : a_j \geq b_j^1} w_j \geq \lambda \text{ and } \nexists j : a_j \leq v_j^1$$

Rambla

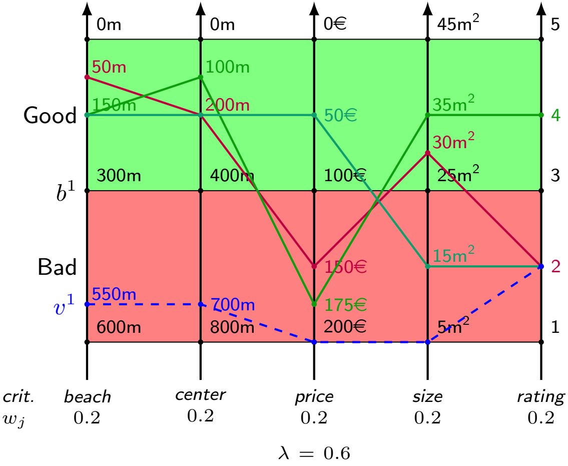
$\in$ **Bad**

Majestic

$\in$ **Good**

Binary veto rule

- Veto if alternative worse than the **veto profile** on **any criterion**



Assignment rule

hotel $\in$ Good

$\Leftrightarrow$

$$\sum_{j : a_j \geq b_j^1} w_j \geq \lambda \text{ and } \nexists j : a_j \leq v_j^1$$

Rambla

$\in$ **Bad**

Majestic

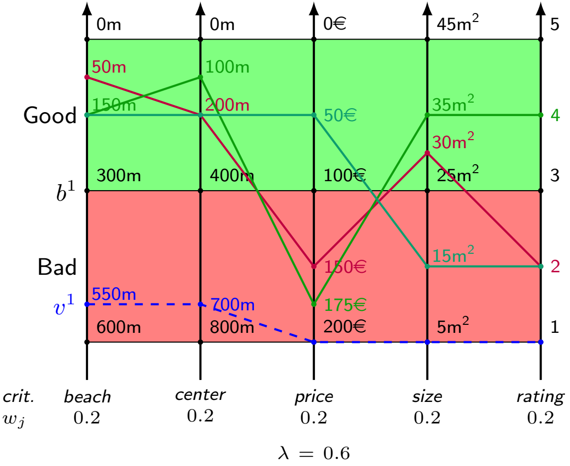
$\in$ **Good**

Travelodge

$\in$ ~~Good~~

Binary veto rule

- Veto if alternative worse than the **veto profile** on **any criterion**



Assignment rule

hotel $\in$ Good

$\Leftrightarrow$

$$\sum_{j : a_j \geq b_j^1} w_j \geq \lambda \text{ and } \nexists j : a_j \leq v_j^1$$

Rambla

$\in$ ~~Bad~~

Majestic

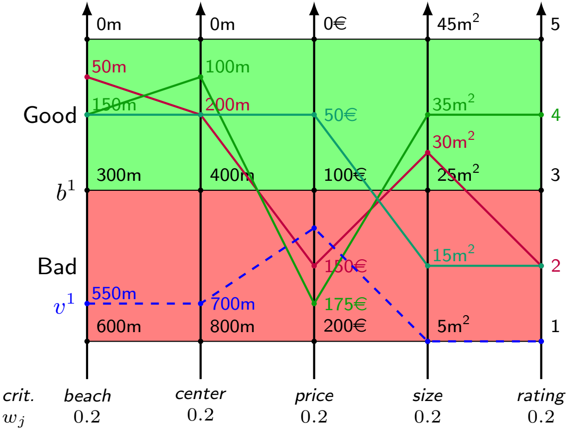
$\in$ Good

Travelodge

$\in$ Good

Binary veto rule

- Veto if alternative worse than the **veto profile** on **any criterion**



$\lambda = 0.6$

Assignment rule

hotel $\in$ Good

$\Leftrightarrow$

$$\sum_{j : a_j \geq b_j^1} w_j \geq \lambda \text{ and } \nexists j : a_j \leq v_j^1$$

Rambla

$\in$ Bad

Majestic

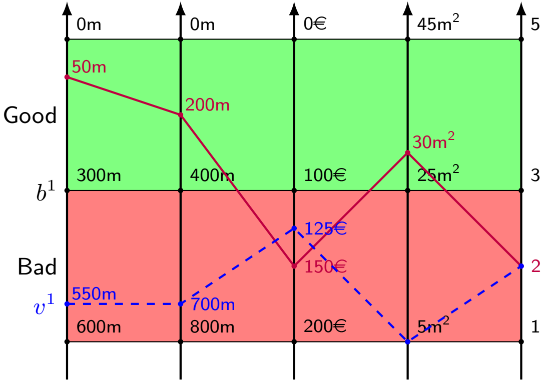
$\in$ ~~Good~~

Travelodge

$\in$ Good

Coalitional veto rule

- Veto if alternative worse than the **veto profile** on a **subset of criteria**



crit.	beach	center	price	size	rating
w_j	0.2	0.2	0.2	0.2	0.2
z_j	0.2	0.2	0.2	0.2	0.2

$\lambda = 0.6$ $\Lambda = 0.4$

Assignment rule

hotel $\in$ Good

$$\Leftrightarrow$$

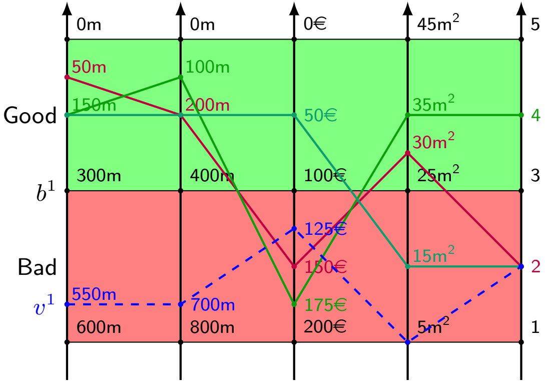
$$\sum_{j:a_j \geq b_j^1} w_j \geq \lambda \text{ and } \sum_{j:a_j \leq v_j^1} z_j < \Lambda$$

Rambla

$\in$ **Bad**

Coalitional veto rule

- Veto if alternative worse than the **veto profile** on a **subset of criteria**



crit.	beach	center	price	size	rating
w_j	0.2	0.2	0.2	0.2	0.2
z_j	0.2	0.2	0.2	0.2	0.2

$\lambda = 0.6$ $\quad$ $\Lambda = 0.4$

Assignment rule

hotel $\in$ Good

$$\Leftrightarrow$$

$$\sum_{j:a_j \geq b_j^1} w_j \geq \lambda \text{ and } \sum_{j:a_j \leq v_j^1} z_j < \Lambda$$

Rambla

$\in$ **Bad**

Majestic

$\in$ **Good**

Travelodge

$\in$ **Good**

Learning a MR-Sort model with coalitional veto

Problem size

- Number of parameters to learn **doubled** compared to a classical MR-Sort model without veto

Mixed integer program

- Adapted for **small problems**
- Tested on a **small example**

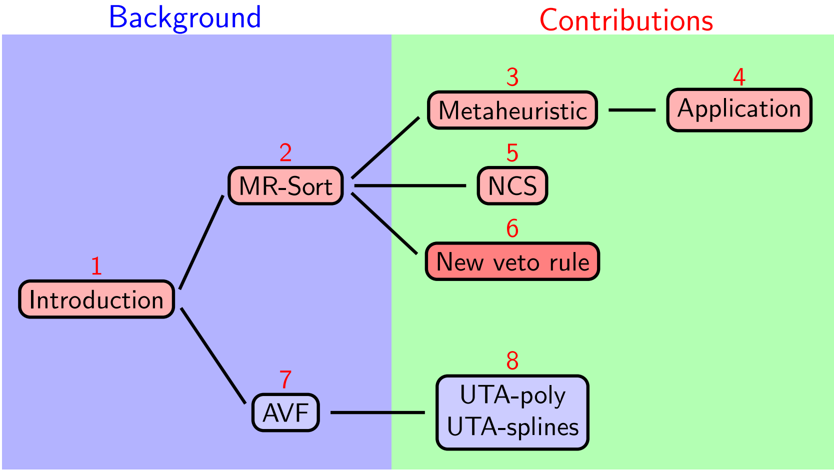
Adaptation of the MR-Sort metaheuristic

- **Outline** of an approach for integrating the veto in the metaheuristic

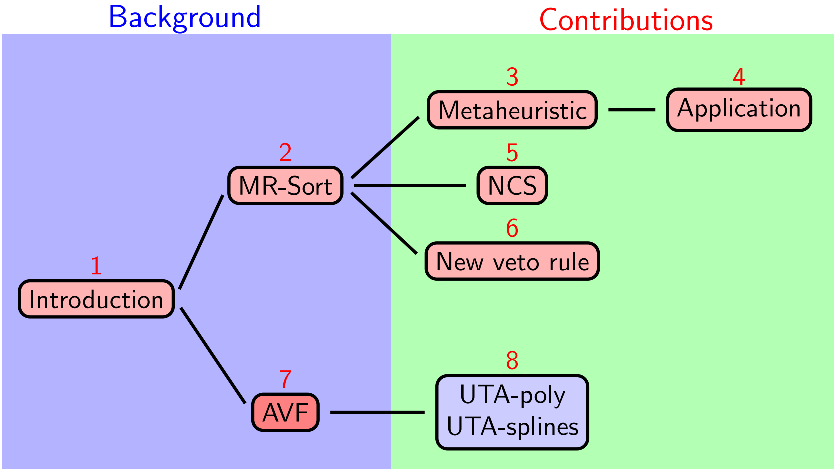
Contributions

- Sobrie, O., Mousseau, V., and Pirlot, M. (2014). New veto rules for sorting models.
  In *20th Conference of the International Federation of Operational Research Societies*, Barcelona, Spain

Outline of the presentation

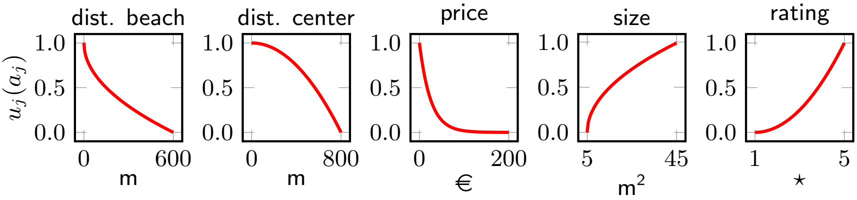


Outline of the presentation



Additive value function model I

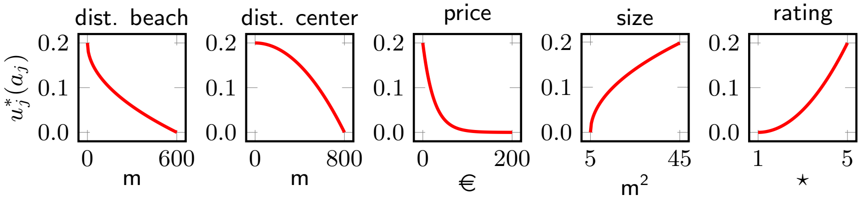
- A **marginal value function** is associated to each criterion



- Marginal value functions are **monotone**
- A **weight** w_j is associated to each criterion j
- A **score** $U(a)$ can be computed for an alt. a

Additive value function model I

- A marginal value function is associated to each criterion

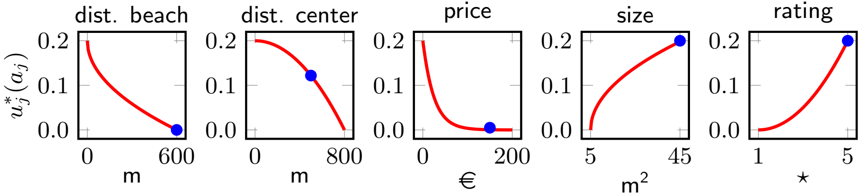


- Marginal value functions are monotone
- A weight w_j is associated to each criterion j
- A score $U(a)$ can be computed for an alt. a

$$u_j^*(a_j) = w_j u_j(a_j)$$

Additive value function model I

- A **marginal value function** is associated to each criterion



- Marginal value functions are **monotone**
- A **weight** w_j is associated to each criterion j
- A **score** $U(a)$ can be computed for an alt. a

$$u_j^*(a_j) = w_j u_j(a_j)$$

$$U(a) = \sum_{j=1}^{5} u_j^*(a_j)$$

Miramar

0.51

Plaza

0.53

Hilton

0.43

Hotel W

0.41

Additive value function model II

Ranking

Plaza

0.53

≻

Miramar

0.51

≻

Hilton

0.43

≻

Hotel W

0.41

Sorting

Good 0.5 **Bad**

Plaza

0.53

Miramar

0.51

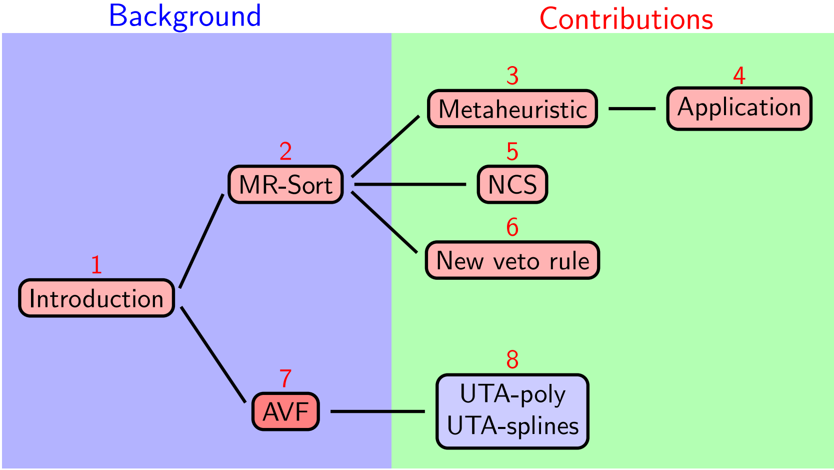
Hilton

0.43

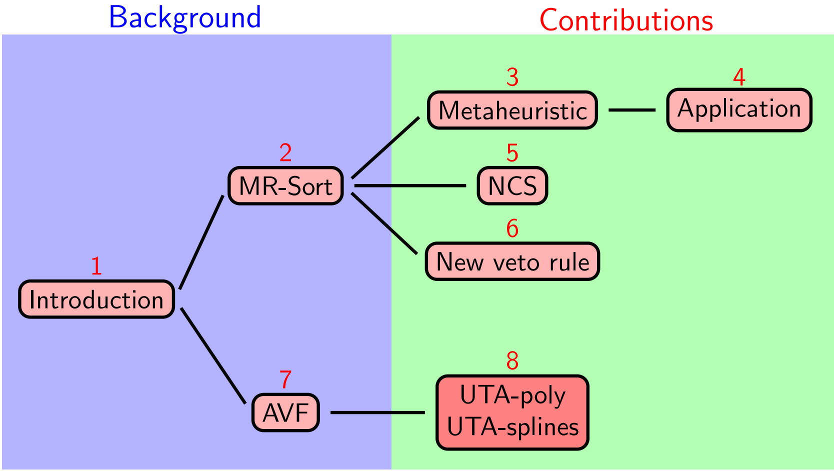
Hotel W

0.41

Outline of the presentation



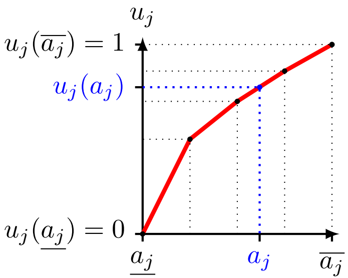
Outline of the presentation



Learning an AVF model

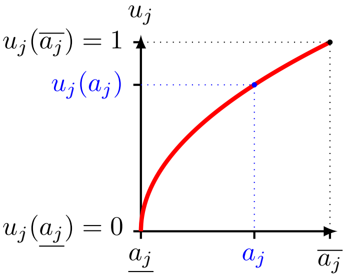
Existing methods

- **UTA** : LP for learning the parameters of an AVF-ranking model
- **UTADIS** : LP for learning the parameters of an AVF-sorting model
- Other methods : **UTA***, **ACUTA**, ...
- **Monotonicity** of the marginals is ensured
- Marginals are modeled with **piecewise linear functions**



UTA-poly and UTA-splines

- Marginals are modeled by **polynomials** or **splines** (continuity of the marginals up to the second derivative)
- Use of **semi-definite programming**
- Monotonicity guaranteed if **first derivative nonnegative**
- Hilbert's theorems



Theorem (Hilbert)

A polynomial $F : \mathbb{R}^n \to \mathbb{R}$ is nonnegative if it is possible to decompose it as a sum of squares (SOS) :

$$F(z) = \sum_s f_s^2(z) \qquad \text{with } z \in \mathbb{R}^n.$$

Theorem (Hilbert)

A non-negative polynomial in one variable is always a SOS.

UTA-poly - Example I

	x	y
a^1	10	7
a^2	6	8
a^3	7	5

$$a^1 \succ a^2 \succ a^3$$

- We define $u_1^*(x)$ and $u_2^*(y)$ as **third degree polynomials** :

$$u_1^*(x) = p_{x,0} + p_{x,1} \cdot x + p_{x,2} \cdot x^2 + p_{x,3} \cdot x^3,$$
$$u_2^*(y) = p_{y,0} + p_{y,1} \cdot y + p_{y,2} \cdot y^2 + p_{y,3} \cdot y^3.$$

- **Scores** of a^1, a^2 and a^3 are given by :

$$U(a^1) = p_{x,0} + 10p_{x,1} + 100p_{x,2} + 1000p_{x,3} + p_{y,0} + 7p_{y,1} + 49p_{y,2} + 343p_{y,3},$$
$$U(a^2) = p_{x,0} + 6p_{x,1} + 36p_{x,2} + 324p_{x,3} + p_{y,0} + 8p_{y,1} + 64p_{y,2} + 512p_{y,3},$$
$$U(a^3) = p_{x,0} + 7p_{x,1} + 49p_{x,2} + 343p_{x,3} + p_{y,0} + 5p_{y,1} + 25p_{y,2} + 125p_{y,3}.$$

UTA-poly - Example II

- **Scores** of a^1, a^2 and a^3 are given by :

$$U(a^1) = p_{x,0} + 10p_{x,1} + 100p_{x,2} + 1000p_{x,3} + p_{y,0} + 7p_{y,1} + 49p_{y,2} + 343p_{y,3},$$
$$U(a^2) = p_{x,0} + 6p_{x,1} + 36p_{x,2} + 324p_{x,3} + p_{y,0} + 8p_{y,1} + 64p_{y,2} + 512p_{y,3},$$
$$U(a^3) = p_{x,0} + 7p_{x,1} + 49p_{x,2} + 343p_{x,3} + p_{y,0} + 5p_{y,1} + 25p_{y,2} + 125p_{y,3}.$$

- We have $a^1 \succ a^2$ **and** $a^2 \succ a^3$, which implies :

$$\begin{cases} U(a^1) - U(a^2) + \sigma^+(a^1) - \sigma^-(a^1) - \sigma^+(a^2) + \sigma^-(a^2) & > & 0, \\ U(a^2) - U(a^3) + \sigma^+(a^2) - \sigma^-(a^2) - \sigma^+(a^1) + \sigma^-(a^1) & > & 0. \end{cases}$$

- By replacing $U(a^1)$, $U(a^2)$ and $U(a^3)$, we have :

$$\begin{cases} 4p_{x,1} + 64p_{x,2} + 776p_{x,3} - p_{y,1} - 15p_{y,2} - 231p_{y,3} + \sigma^+(a^1) - \sigma^-(a^1) & & \\ \qquad -\sigma^+(a^2) + \sigma^-(a^2) & > & 0, \\ -p_{x,1} - 13p_{x,2} - 19p_{x,3} + 3p_{y,1} + 39p_{y,2} + 387p_{y,3} + \sigma^+(a^2) - \sigma^-(a^2) & & \\ \qquad -\sigma^+(a^3) + \sigma^-(a^3) & > & 0. \end{cases}$$

UTA-poly - Example III

- We impose the derivative of u_1^* and u_2^* to be **SOS** :

$$
\begin{aligned}
u_1^{*'} &= \overline{x}^\mathsf{T} Q \overline{x} \\
&= \begin{pmatrix} 1 \\ x \end{pmatrix}^\mathsf{T} \begin{pmatrix} q_{0,0} & q_{0,1} \\ q_{1,0} & q_{1,1} \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix} \\
&= q_{0,0} + \left(q_{0,1} + q_{0,1}\right) x + q_{1,1} x^2, \\
u_2^{*'} &= \overline{y}^\mathsf{T} R \overline{y} \\
&= r_{0,0} + \left(r_{0,1} + r_{1,0}\right) y + r_{1,1} y^2.
\end{aligned}
$$

- Q and R have to be **semi-definite positive**, in conjunction with :

$$
\begin{cases} p_{x,1} &= q_{0,0}, \\ 2p_{x,2} &= q_{0,1} + q_{1,0}, \\ 3p_{x,3} &= q_{1,1}, \end{cases} \quad \text{and} \quad \begin{cases} p_{y,1} &= r_{0,0}, \\ 2p_{y,2} &= r_{0,1} + r_{1,0}, \\ 3p_{y,3} &= r_{1,1}. \end{cases}
$$

UTA-poly - Example IV

- We add **normalization** constraints :

$$\begin{cases} p_{x,0} = 0, \\ p_{y,0} = 0, \\ 10p_{x,1} + 100p_{x,2} + 1000p_{x,3} + 10p_{y,1} + 100p_{y,2} + 1000p_{y,3} = 1. \end{cases}$$

UTA-poly - Example V

$$\min \sigma^+(a^1) + \sigma^-(a^1) + \sigma^+(a^2) + \sigma^-(a^2) + \sigma^+(a^3) + \sigma^-(a^3).$$

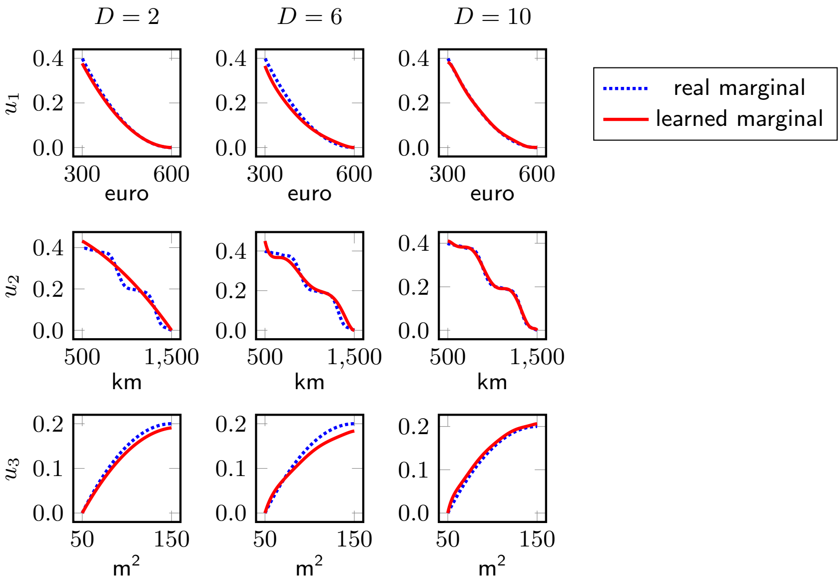
such that :

$$\left\{\begin{array}{rcl}
4p_{x,1} + 64p_{x,2} + 776p_{x,3} - p_{y,1} - 15p_{y,2} - 231p_{y,3} & & \\
+\sigma^+(a^1) - \sigma^-(a^1) - \sigma^+(a^2) + \sigma^-(a^2) & > & 0, \\
-p_{x,1} - 13p_{x,2} - 19p_{x,3} + 3p_{y,1} + 39p_{y,2} + 387p_{y,3} & & \\
+\sigma^+(a^2) - \sigma^-(a^2) - \sigma^+(a^3) + \sigma^-(a^3) & > & 0, \\
p_{x,0} & = & 0, \\
p_{y,0} & = & 0, \\
10p_{x,1} + 100p_{x,2} + 1000p_{x,3} + 10p_{y,1} + 100p_{y,2} + 1000p_{y,3} & = & 1, \\
p_{x,1} & = & q_{0,0}, \\
2p_{x,2} & = & q_{0,1} + q_{1,0}, \\
3p_{x,3} & = & q_{1,1}, \\
p_{y,1} & = & r_{0,0}, \\
2p_{y,2} & = & r_{0,1} + r_{1,0}, \\
3p_{y,3} & = & r_{1,1},
\end{array}\right.$$

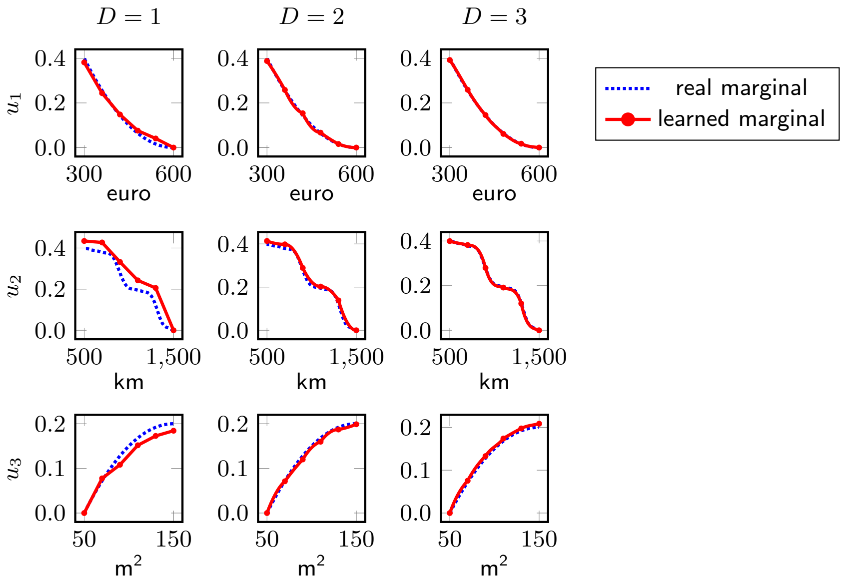
with :

$$\left\{\begin{array}{rcl}
Q, R & & PSD, \\
\sigma^+(a^1), \sigma^-(a^1), \sigma^+(a^2), \sigma^-(a^2), \sigma^+(a^3), \sigma^-(a^3) & \geq & 0.
\end{array}\right.$$

Example of marginals learning with UTA-poly



Example of marginals learning with UTA-splines



Experiments with UTA-poly and UTA-splines

Artificial datasets

- Artificial datasets built on the basis of **various type of additive value functions** (exponentials, polynomials, etc.)
- **UTA-poly** and **UTA-splines** models learned
- UTA(DIS)-poly and UTA(DIS)-splines **computing time** of the **same order of magnitude** as UTA(DIS)
- **Model retrieval**

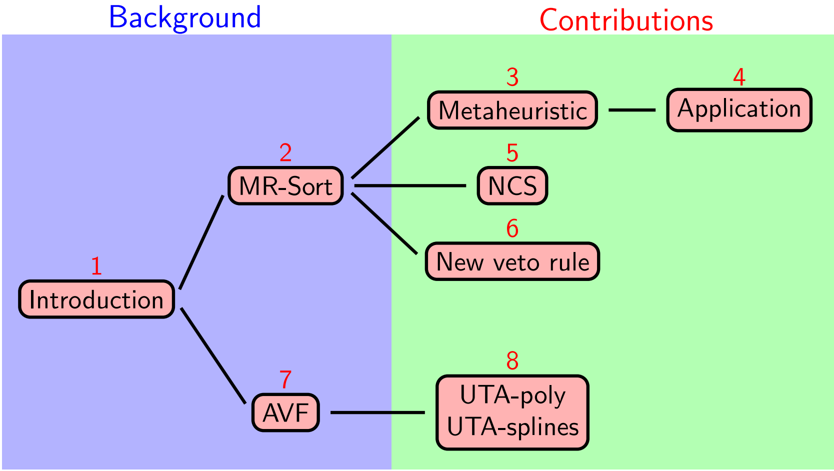
Real datasets

- Datasets issued from the **preference learning field**
- Results at least **as good as with UTADIS**
- **Overfitting** if too much degrees of freedom let to the semi-definite program

Contributions

- Sobrie, O., Gillis, N., Mousseau, V., and Pirlot, M. (2016a). UTA-poly and UTA-splines: additive value functions with polynomial marginals. Submitted

Outline of the presentation



Conclusion and further research I

Use MCDA models to deal with PL problems
(outranking models and additive value function models)

- **MR-Sort** and **NCS** outranking methods
- **Algorithms** for learning MR-Sort and NCS models from **large datasets**
- Methods for learning **AVF models**

Conclusion and further research II

Validation of the learning algorithms as done in PL

- Tests with **PL datasets**
- **Statistical** tests (learning and test sets)

Conclusion and further research III

Test the algorithms and models on a real application

- Test of MR-Sort with the **ASA dataset**
- Results **comparable** to other machine learning algorithms
- MR-Sort **easier to explain** than other algorithms

Conclusion and further research IV

Study the expressivity of the MCDA models

- Expressivity of **MR-Sort** and **NCS** has been studied
- Proportion of rule that can be represented by a set of k-additive weights for models involving a number of criteria **smaller than 7**

- Extension of the expressivity with **coalitional veto**

Conclusion and further research V

Bring new techniques in MCDA and PL

- UTA-poly and UTA-splines
- Semi-definite programming

Further research

- Use of **relaxation techniques** for learning the models

- Improvement of the **interpretability** of MR-Sort (weights and cut thresholds)

- Study of rules that can be represented by k-additive weights for models involving **7 criteria**
- **Analysis of complexity** of the MR-Sort model (e.g. VC dimension)

- **Algorithm** for learning a MR-Sort model using coalitional veto

- Extend semi-definite programming to **other MCDA methods** (MACBETH, GAI network)
- Improvement of **UTA(DIS)-poly/splines objective function**

Thank you for your attention !

References I

Bouyssou, D. and Marchant, T. (2007a). An axiomatic approach to noncompensatory sorting methods in MCDM, I : The case of two categories. *European Journal of Operational Research*, 178(1) :217–245.

Bouyssou, D. and Marchant, T. (2007b). An axiomatic approach to noncompensatory sorting methods in MCDM, II : More than two categories. *European Journal of Operational Research*, 178(1) :246–276.

Ersek Uyanık, E., Sobrie, O., Mousseau, V., and Pirlot, M. (2016). Families of sufficient coalitions of criteria involved in ordered classification procedures. Submitted.

Leroy, A., Mousseau, V., and Pirlot, M. (2011). Learning the parameters of a multiple criteria sorting method. In Brafman, R., Roberts, F., and Tsoukiàs, A., editors, *Algorithmic Decision Theory*, volume 6992 of *Lecture Notes in Artificial Intelligence*, pages 219–233. Springer.

References II

Sobrie, O., Gillis, N., Mousseau, V., and Pirlot, M. (2016a). UTA-poly and UTA-splines : additive value functions with polynomial marginals. Submitted.

Sobrie, O., Lazouni, M. E. A., Mahmoudi, S., Mousseau, V., and Pirlot, M. (2016b). A new decision support model for preanesthetic evaluation. *Computer Methods and Programs in Biomedicine*. Accepted.

Sobrie, O., Mousseau, V., and Pirlot, M. (2012). Learning the parameters of a multiple criteria sorting method from large sets of assignment examples. In *DA2PL 2012 Workshop From Multiple Criteria Decision Aid to Preference Learning*, pages 21–31. Mons, Belgique.

Sobrie, O., Mousseau, V., and Pirlot, M. (2013). Learning a majority rule model from large sets of assignment examples. In Perny, P., Pirlot, M., and Tsoukiás, A., editors, *Algorithmic Decision Theory*, volume 8176 of *Lecture Notes in Artificial Intelligence*, pages 336–350, Brussels, Belgium. Springer.

References III

Sobrie, O., Mousseau, V., and Pirlot, M. (2014). New veto rules for sorting models. In *20th Conference of the International Federation of Operational Research Societies*, Barcelona, Spain.

Sobrie, O., Mousseau, V., and Pirlot, M. (2015). Learning the parameters of a non compensatory sorting model. In Walsh, T., editor, *Algorithmic Decision Theory*, volume 9346 of *Lecture Notes in Artificial Intelligence*, pages 153–170, Lexington, KY, USA. Springer.