Handling imprecise and missing evaluations in multi-criteria majority-rule sorting

Patrick Meyer[a], Alexandru-Liviu Olteanu[b,*]

[a] *IMT Atlantique, Lab-STICC, UMR CNRS 6285, F-29238 Brest, France*
[b] *Lab-STICC, CNRS, Université de Bretagne Sud, 17 Boulevard Flandres Dunkerque 1940, 56100 Lorient, France*

Abstract

In this paper we propose an extension of a multi-criteria majority-rule sorting model that allows the handling of problems where the decision alternatives contain imprecise or even missing evaluations. Due to the imprecise nature of the evaluations we offer the possibility of assigning an alternative to one or more neighboring categories, both as input for inferring the model parameters as well as the output of the classification. Our contribution also contains an algorithmic approach for extracting the parameters of this model during an elicitation process, which is validated across a wide range of generated datasets.

Keywords: multi-criteria decision aiding, majority-rule sorting, missing values, intervals of categories

1. Introduction

We consider in this article a decision situation in which a finite set of alternatives is evaluated on a finite set of criteria. When comparing two alternatives from a preferential point of view, a decision maker (DM) uses a majority-rule in accordance with the Multi-criteria Decision Aiding (MCDA) outranking paradigm [33]. This means that (s)he considers that an alternative a *outranks* an alternative b when a *weighted majority* of criteria validates the fact that a is performing at least as good as b and there is no criterion where b seriously outperforms a. The first condition is called concordance or majority rule, whereas the second condition is called discordance or veto rule. Various implementations of these conditions, and their combination, have been proposed in the literature (see for example [37]), together with additional concepts such as "reinforced preference", "counter-veto" (see [38]) or "dictator" (see [24]).

In this paper, we restrict our discourse to a specific type of decision problems, called sorting, which aims to assign the decision alternatives into a set of predefined ordered categories or classes, according to the preferences of the

*Corresponding author (alexandru.olteanu@univ-ubs.fr)

DM, represented by a majority rule model. The work in this article is based on a simplified version of the classical Electre Tri method [16; 30; 35], called MR-Sort [21], which is close to the version axiomatized in [3; 4], and which does not consider the previously mentioned discordance or veto principle. The MR-Sort procedure takes thus into account the preferences of the DM, which are represented through the following preferential parameters :

- weights, which give the relative importance of criteria;
- a majority threshold, which indicates the weight of a coalition of criteria in order to be considered sufficient;
- category limits, which are used to segment the evaluations scales into performance ranges appropriate for assignment into each category.

In this context, we focus on the problem raised by imprecise and incomplete evaluations in the performances of the alternatives on the criteria. We therefore propose an extension of the MR-Sort procedure, which handles both this imprecision and the lack of evaluations, and which possibly provides assignments into one or several neighboring categories. In order to determine the parameters of the model, the DM is asked to assign some alternatives, which may also contain missing or imprecise evaluations, to one or more contiguous categories. The proposed elicitation method then determines the preferential parameters of the model via an approximative approach.

The rest of the article is structured as follows. In Section 2 we start this paper by presenting an overview of sorting techniques in the outranking paradigm and related work on imprecise and missing evaluations in MCDA. Section 3 details the sorting model that we propose, whereas Section 4 shows how its parameters can be determined from assignment examples. In Section 5 we test this learning algorithm on artificially generated benchmarks . Finally, we draw some conclusions in Section 6 and present perspectives for future work.

2. State of the art

2.1. Majority-rule sorting

In this section, we introduce formally the MR-Sort assignment procedure, on which the work of this article is based. It is a simplified version of the classical Electre Tri method [16; 30; 35] and close to the version axiomatized in [3; 4].

Let us consider a finite set of alternatives A, a finite set of criteria indexes J and a set of category limits separating profiles $B = \{b_1, \dots b_{k-1}\}$. Each alternative and each category limit is a vector of evaluations with respect to all criteria. The evaluation with respect to criterion j can be viewed as a function $g_j : A \cup B \to \mathbb{R}$, where $g_j(a)$ denotes the evaluation of alternative $a \in A$ on criterion j and $g_j(b_h)$ denotes the evaluation of category limit $b_h, \forall h \in \{1, \dots, k-1\}$, on criterion j. The set of category limits are used to define a set of k categories $\{c_1, \dots, c_k\}$, ordered by their desirability, from c_1 being the worst category to c_k being the best one. Each category c_h is defined through its upper limit, b_h, and its lower limit, b_{h-1}, with the exception of the worst and best categories, which have only one limit. We assume, without loss of generality, that the performances are supposed to be such that a higher value denotes a better performance. Furthermore the performances of the category limits are non-decreasing, i.e. $\forall j \in J, 1 < h < k : g_j(b_{h-1}) \leq g_j(b_h)$.

MR-Sort uses two assignment rules for placing the alternatives into categories: the pessimistic and the optimistic assignment rules [37; 5]. The pessimistic rule assigns an alternative a to the highest possible category c_h such that a outranks the category's lower frontier b_{h-1}. The optimistic rule assigns a to the lowest possible category c_h such that the category's upper frontier b_h outranks a. The pessimistic rule is the most commonly used in practice.

An alternative a is said to outrank a frontier b_{h-1} if and only if there is a sufficient coalition of criteria supporting the assertion "a is at least as good as b_{h-1}". To measure this, we define for each criterion j a function $C_j : A \times B \rightarrow \{0, 1\}$ which assesses whether criterion j supports that statement or not:

$$\forall j \in J, a \in A, 1 \leq h \leq k : C_j(a, b_{h-1}) = \begin{cases} 1, \text{ if } g_j(a) \geq g_j(b_{h-1}), \\ 0, \text{ otherwise.} \end{cases} \quad (1)$$

To assess whether a coalition of criteria is in favor of the outranking or not, $\forall a \in A, 1 \leq h \leq k$, we first define the overall concordance as:

$$C(a, b_{h-1}) = \sum_{j \in J} w_j C_j(a, b_{h-1}), \quad (2)$$

where w_j is the weight of criterion j. The weights are defined so that they are positive ($w_j \geqslant 0, \forall j \in J$) and sum up to one ($\sum_{j \in J} w_j = 1$). This overall concordance is then compared to a majority threshold $\lambda \in [0.5, 1]$ extracted from the decision-maker's preferences along with the weights. As we do not consider any veto rule here, the outranking relation S is then defined as:

$$a \,\mathrm{S}\, b_{h-1} \iff C(a, b_{h-1}) \geqslant \lambda. \quad (3)$$

If $C(a, b_{h-1}) < \lambda$, the coalition of criteria is not sufficient, the alternative does not outrank the frontier b_{h-1} and will therefore be assigned in a category lower than c_h.

Alternative a is assigned to the highest category it outranks, hence this rule can be written as:

$$a \in c_h \iff a \,\mathrm{S}\, b_{h-1} \text{ and } a \not\mathrm{S}\, b_h. \quad (4)$$

In order for this assignment rule to be used for the limit categories, two dummy profiles b_0 and b_k need to be added to B, the first holding the lowest possible evaluations on all criteria, while the second holding the highest possible ones.

In Table 1 we provide a simple illustrative example for this assignment rule.

At the top we observe the parameters of the sorting model, involving 5 criteria and 3 classes. The first parameter (λ) is the majority threshold, followed by the vector of the criteria weights (w) and two category separating profiles (b_1

Table 1: Illustrative example for the MR-Sort assignment procedure.
$\lambda = \frac{3}{5}$ $\quad w = (\frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5})$ $\quad b_1 = (1, 1, 1, 1, 1)$ $\quad b_2 = (2, 2, 2, 2, 2)$

A	g_1	g_2	g_3	g_4	g_5	$a_i \, \mathrm{S} \, b_0$	$a_i \not\mathrm{S} \, b_1$	$a_i \, \mathrm{S} \, b_1$	$a_i \not\mathrm{S} \, b_2$	$a_i \, \mathrm{S} \, b_2$	$a_i \not\mathrm{S} \, b_3$	assignment
a_1	2	2	2	0	0	✓	✗	✓	✗	✓	✓	c_3
a_2	2	2	1	0	0	✓	✗	✓	✓	✗	✓	c_2
a_3	2	2	0	0	0	✓	✓	✗	✓	✓	✓	c_1

and b_2). Each criterion is defined on a scale from 0 to 2. Below the parameters we illustrate the assignment of three alternatives using the assignment rule from Equation (4). In the case of the first and last categories we have added two fictive limits in order to simplify the example. The lower category limit of the worst category (b_0) is considered to be always worse than all the alternatives in A on each criterion, while the upper profile of the best category (b_3) is considered to be always better than all alternatives in A on each criterion. The first alternative outranks both b_1 and b_2, as it is at least as good as these profiles on the first three criteria, hence it is assigned to the highest category. The second alternative is at least as good as b_1 on the first three criteria, but it is at least as good as b_2 on only the first two, therefore it is assigned to the second category. Finally, the last alternative is not at least as good as b_1, nor b_2, since it is at least as good as them on only two out of the 5 criteria, therefore it is placed in the lowest category.

The parameters of this model (and similar Electre Tri -like sorting procedures) may be both directly and indirectly elicited. However, in order to overcome the difficulties of eliciting these parameters directly from the decision-maker, several works have focused on the indirect approach. Mousseau and Słowiński [29] have proposed to find the entire model through the use of assignment examples. Mousseau et al. [28] only sought to find the criteria importance weights with the other parameters being supposedly known, while Ngo The and Mousseau [31] only looked for the category limits. Other more robust approaches compute for each alternative a range of possible categories to which they may be assigned when the parameters of the model are not completely determined [9; 10; 12]. Approaches that deal with inconsistent sets of assignment examples leading to non existing preference model solutions have also been explored in [27; 26].

In most cases, the approaches of inferring the parameters of majority-rule sorting models use mathematical programming techniques involving binary variables, such as in Leroy et al. [21]. As these approaches find the optimal solution, they may also require large amounts of computational resources and time, making their use limited when large sets of assignment examples are considered. Sobrie et al. [40] have suggested to use a technique based on a metaheuristic to learn the parameters of the sorting model, which has been adapted and extended by Olteanu and Meyer [32] in order to additionally take into account veto thresholds. More recently, population-based metaheuristics have also been proposed in order to learn MR-Sort models with coalitional veto [41].

In most cases, the assignments take the form of a single category in which the DM places each alternative. An approach that extends these assignments to include multiple categories, in order to account for the possible hesitation of the DM, has been explored by Dias and Mousseau [11].

2.2. Imprecision and incompleteness

Some of the earliest detailed studies on data or information imperfection may be found in [14; 25]. Tacnet [43] borrows from these sources and proposes a taxonomy of data or information imperfection. Based on the literature, he considers that this imperfection can take four main forms : inconsistency, imprecision, incompleteness and uncertainty. *Inconsistency* is a type of imperfection which originates in conflicting information, and usually produces incoherent conclusions. *Imprecision* is related to information which is not sufficient for an agent (in this context a decision maker or a model of his/her preferences) to answer the question at hand. Either the numerical values are poorly known due to a flaw in the observation, or because natural language is used to vaguely describe the problem. *Incompleteness* corresponds to an absence of information on the underlying problem. *Uncertainty* is related to the knowledge that the agent has about an information, and more specifically it's validity or truth.

In most of the scientific disciplines, incompleteness or imprecision on the data occur very regularly. Very often, they are approached in a very simple and convenient way, by, e.g., excluding incomplete cases or replacing (imputing) missing values with the mean. Similarly, imprecision is also handled simply by considering that an imprecise evaluation can be represented by an average or median value of the possible evaluations.

More evolved techniques however exist, and they can be found in seminal books on incomplete (or missing) data analyses, as, e.g., [22; 39] : maximum likelihood estimation and multiple imputation. Regarding imprecision, classically such information is expressed as either a set of possible values, an interval of values, or as fuzzy numbers in case of a linguistic variable [43].

In MCDA, Roy [34] analyzes the main sources of difficulties linked to imprecision and uncertainty. In summary, these various types of imperfection can either be found in the performances of the alternatives on the criteria, in the values of the preferential parameters (weights of the criteria, discrimination thresholds, utility functions, ...), or in the recommendation that is proposed to the decision maker. As stated in [36], many approaches have been proposed to tackle imperfection issues in MCDA. Without being exhaustive, they make use of probability theory as in Multi-Attribute Utility Theory (see, e.g., [15] for a detailed overview), possibility theory [13], multi-valued logic, discrimination thresholds mainly used in outranking methods [16], fuzzy or valued binary relations, fuzzy numbers, or rough sets [18].

When it comes more specifically to missing (or incomplete) data in the evaluation of the alternatives, to our knowledge, this topic has only been poorly explored. Bisdorff et al. [2] propose an outranking-based method which allows to take into account missing values for the choice problem. The validation of an outranking relation is extended to a three value logic by adding a second majority threshold. The first threshold is used in order to validate whether an alternative outranks another, while the second threshold is used in order to invalidate this statement. If the value of the coalition of criteria that support the outranking statement falls within these two thresholds then it cannot be asserted whether the first alternative outranks the second one (indetermination). These two majority thresholds are constrained so that they are symmetrical with respect to the 50% support level. Greco et al. [17] deal with the same issue, using however a rough sets-based approach for the sorting problem. In both cases, no assumption is made on the values of the missing data. Brans and Mareschal [6] handle the missing data issue by considering that in the pairwise comparison, in case of a missing value on a criterion, both alternatives have equal evaluations on that criterion. Corrente et al. [8] handle imprecise evaluations using n-point intervals consisting of nested interval evaluations and their associated increasing levels of plausibility. Finally, dominance-based rough set approaches [42] have also recently explored the topic of handling incomplete information in classification.

In this article we wish to tackle more specifically the problem of imprecise performances of alternatives on the criteria, represented by interval evaluations, as well as missing performances. This latter case can in this framework be seen as a special case of interval evaluations, where the lower and upper bounds of the intervals correspond to the lowest and highest possible values of the evaluation scale of the considered criterion.

3. Handling imprecise or missing evaluations

3.1. The proposed MR-Sort extension

In order to express the *imprecision* of the evaluations of the alternatives, we propose to define two new performance functions for each criterion : $g_j^{\min}, g_j^{\max} : A \to \mathbb{R}$. These functions correspond to the minimum, respectively maximum evaluations that an alternative may take on criterion j. This leads to the definition of two versions of alternative a: an optimistic version evaluated using $g^{\min}$ and an optimistic version evaluated using $g^{\max}$.

The initially defined g_j functions are still used to give the evaluations of the category limits, as we do not introduce any imprecision in their evaluations.

Like in the case of the MR-Sort method, we will be using outranking relations in order to assess whether an alternative is at least as good as a category separating profile b_{h-1}. However, as we are now considering two versions of the same alternative, i.e. a pessimistic version evaluated using $g^{\min}$ and an optimistic version evaluated using $g^{\max}$, we also define two corresponding outranking relations. These relations may ultimately be identical, however they may also differ in order to illustrate the potential change in the perspective of the DM when considering the worst and best possible versions of the decision alternative.

We define the pessimistic and optimistic local concordance indexes between an alternative a and a category profile $b_h, \forall h \in \{1, \ldots, k\}$ on any criterion $j \in J$, as:

$$C_j^{\text{pes}}(a, b_h) = \begin{cases} 1 & \text{, if } g_j^{\min}(a) \geq g_j(b_h), \\ 0 & \text{, otherwise.} \end{cases} \tag{5}$$

$$C_j^{\text{opt}}(a, b_h) = \begin{cases} 1 & \text{, if } g_j^{\max}(a) \geq g_j(b_h), \\ 0 & \text{, otherwise.} \end{cases} \tag{6}$$

We define the overall support of a outranking a category profile b_h in the pessimistic and optimistic cases respectively as:

$$C^{\text{pes}}(a, b_h) = \sum_{j \in J} w_j C_j^{\text{pes}}(a, b_h) \tag{7}$$

$$C^{\text{opt}}(a, b_h) = \sum_{j \in J} w_j C_j^{\text{opt}}(a, b_h) \tag{8}$$

We say that a outranks the category profile b_h in the pessimistic case, i.e. $a \, \mathrm{S}^{\text{pes}} \, b_h$, when $C^{\text{pes}}(a, b_h) \geqslant \lambda^{\text{pes}}$. We also say that a outranks the category profile b_h in the optimistic case, i.e. $a \, \mathrm{S}^{\text{opt}} \, b_h$, when $C^{\text{opt}}(a, b_h) \geqslant \lambda^{\text{opt}}$.

The λ^{pes} and λ^{opt} thresholds are used to determine when a coalition of criteria is sufficient in order to validate their corresponding outranking relation. We relax the constraint that $\lambda \geqslant 0.5$, which usually accompanies MR-Sort, making $\lambda^{\text{pes}}, \lambda^{\text{opt}} \in]0, 1]$. In this way, the two relations are now governed by a generalized qualified weak majority rule (a term proposed in [23]).

The assignment rule that we propose computes a lower bound and an upper bound for the assignment of a using its pessimistic and its optimistic versions respectively.:

- the lower bound $c_{h^{\text{pes}}}$ corresponds to the highest category whose lower profile is outranked by a in the pessimistic case;
- the upper bound $c_{h^{\text{opt}}}$ corresponds to the lowest category whose upper profile is not outranked by a in the optimistic case.

3.2. Additional considerations

Based on the definition of a pessimistic and optimistic versions of an alternative containing imprecise evaluations, as well as two associated outranking relations, we find the following potential situations when comparing it to a precisely defined category profile:

- $a \, \mathrm{S}^{\text{pes}} \, b_h$ and $a \, \mathrm{S}^{\text{opt}} \, b_h$
  In this situation a is considered to be globally at least as good as b_h, despite any imprecise evaluations it may have. In this case a can be assigned only to category c_{h+1} or above.

- $a \not\mathrm{S}^{\mathrm{pes}} b_h$ and $a \not\mathrm{S}^{\mathrm{opt}} b_h$
  In this case a is globally not at least as good as b_h, despite any imprecise evaluations it may have. In this case a can be assigned only to category c_h or below.

- ($a \not\mathrm{S}^{\mathrm{pes}} b_h$ and $a\, \mathrm{S}^{\mathrm{opt}}\, b_h$) or ($a\, \mathrm{S}^{\mathrm{pes}} b_h$ and $a \not\mathrm{S}^{\mathrm{opt}} b_h$)
  This situation corresponds to a state of indetermination, since one version of a (pessimistic or optimistic) is not considered at least as good as b_h while at the same time the other version (optimistic or pessimistic) is. In this case a can be assigned to both categories c_h and c_{h+1}, as well as to any other categories above or below them based on how a compares to the other profiles.

In order to better illustrate the range of problems that our proposed extension is able to model, we present the following scenarios as described through the choice of the λ^{pes} and λ^{opt} thresholds:

- $0.5 \leqslant \lambda^{\mathrm{opt}} = \lambda^{\mathrm{pes}}$:
  This scenario is identical to the classical MR-Sort method when alternatives do not have imprecise evaluations. Multi-category assignments only occur when imprecision in alternatives' evaluations is considered;

- $\lambda^{\mathrm{opt}} < 0.5 \leqslant \lambda^{\mathrm{pes}}$:
  A strong coalition of criteria is needed for validating the pessimistic outranking relation while a weak one is needed to validate the optimistic outranking relation, leading to potentially many situations of indetermination (a particular case where $\lambda^{\mathrm{opt}} > 0.5$ and $\lambda^{\mathrm{pes}} = 1 - \lambda^{\mathrm{opt}}$ is very similar to the β-cut polarization described by Bisdorff in [1]);

- $\lambda^{\mathrm{pes}} \leqslant \lambda^{\mathrm{opt}}$:
  The DM is imprecision averse and requires a higher level of support for validating the outranking relations of the optimistic version of an alternative as opposed to its pessimistic version. It is possible, in this case, to have an alternative outrank in the pessimistic case a category profile, while at the same time not outrank in the optimistic case the same profile;

- $\lambda^{\mathrm{opt}} < \lambda^{\mathrm{pes}}$:
  The DM is less imprecision averse and accepts a lower level of support for validating the outranking relations of the optimistic version of an alternative as opposed to its pessimistic version. The range between λ^{opt} and λ^{pes} may be seen as levels of support for which the DM is hesitant to either validate or invalidate an assignment;

It is also worth noting that missing evaluations may be easily handled as they are a special case of handling imprecise evaluations. A missing evaluation of a on criterion j may be summarized by $g_j^{\mathrm{pes}}(a)$ taking the smallest possible value on the scale of criterion j and $g_j^{\mathrm{opt}}(a)$ taking the highest possible value on the scale of criterion j.

3.3. Illustrative example

We illustrate the proposed MR-Sort extension when considering precisely defined alternatives as well as several containing imprecision in their evaluations, as depicted in Table 2.

Table 2: Illustrative example for the extended MR-Sort assignment procedure.
$\lambda^{\text{opt}} = \lambda^{\text{pes}} = \frac{3}{5}$ $\quad w = (\frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5})$ $\quad b_1 = (1, 1, 1, 1, 1)$ $\quad b_2 = (2, 2, 2, 2, 2)$

							$a_i\, \text{S}^{\text{pes}}\, b_h$				$a_i\, \text{S}^{\text{opt}}\, b_h$					
A	g_1	g_2	g_3	g_4	g_5	h	0	1	2	3	0	1	2	3	h^{pes}	h^{opt}
a_1	2	2	2	0	0		✓	✓	✓	✗	✓	✓	✓	✗	3	3
a_2	2	2	1	0	0		✓	✓	✗	✗	✓	✓	✗	✗	2	2
a_3	2	2	0	0	0		✓	✗	✗	✗	✓	✗	✗	✗	1	1
a_4	2	2	$[0, 1]$	0	0		✓	✗	✗	✗	✓	✓	✗	✗	1	2
a_5	2	2	?	0	0		✓	✗	✗	✗	✓	✓	✓	✗	1	3
a_6	2	2	2	0	?		✓	✓	✓	✗	✓	✓	✓	✗	3	3

This example is similar to the example from Table 1, depicting a model with 5 equally weighted criteria and three categories defined using the category limits b_1 and b_2. The evaluations on each criterion can vary between 0 and 2, while the majority thresholds of the pessimistic and optimistic outranking relations are equal to $\frac{3}{5}$. This model is identical to the MR-Sort model from the previous example, as seen through the first three alternatives, which are classified in the same way. We have indicated here the indexes of the lower and upper bound category assignments through h^{pes} and h^{opt}. This extended MR-Sort model, however, is also able to properly handle imprecise or missing evaluations, as seen through the following three alternatives.

Alternative a_4 is identical to the first three alternatives with the exception that its evaluation on the third criterion is not precisely defined, potentially falling anywhere between 0 and 1. As in the pessimistic scenario a_4 could be identical to a_3, while in the optimistic scenario it could be identical to a_2, it appears natural that this alternative should be classified in either c_1 or c_2. The illustrated model achieves this exact result. The following alternative adds even more imprecision to its evaluation on the third criterion. As this evaluation may fall anywhere within our evaluation scale from 0 to 2, the pessimistic version of this alternative does not outrank any of the category profiles (except the fictitious bottom profile), while its pessimistic version outranks all of them (except the fictitious top profile). Hence, this alternative may be assigned to any of the three categories. Finally, alternative a_6 is similar to alternative a_1, except that the value for g_5 is missing. Still, this last example illustrates that even when imprecision within an alternative's evaluations is present, it's precise evaluations may still be sufficient in order to allow its assignment to a single category.

4. Inferring the parameters of the sorting model

In order to use the model presented in Section 3, the preferences of the DM need to be elicited. In particular, the criteria importance parameters, the majority thresholds, and the profiles separating the categories on each criterion have to be determined. These parameters can either be obtained directly from the DM (which in most practical situations is not realistic), or learned from assignment examples provided by the DM for each of the categories. When assignment examples are employed, algorithms are used to find a set of parameters such that the classification accuracy over the set of assignment examples is maximized.

4.1. The fitness measure

We denote a classical MR-Sort model instance, represented by its parameters, with $\Omega = \big(\lambda, w = (w_j, j \in J), B\big)$. Let $a \in A$ be an alternative which is used by the DM as an assignment example. Let $\text{assig}_\Omega(a)$ be the category to which a is assigned by Ω and $\text{assig}_{\text{DM}}(a)$ the category to which the DM assigns it.

We remind below the definition of the classification accuracy:

$$CA = \frac{1}{|A|}\sum_{a\in A} CA(a), \quad CA(a) = \begin{cases} 1 & \text{, if } \text{assig}_\Omega(a) = \text{assig}_{\text{DM}}(a) \\ 0 & \text{, if } \text{assig}_\Omega(a) \neq \text{assig}_{\text{DM}}(a). \end{cases} \tag{9}$$

The MR-Sort model for handling missing and imprecise evaluations allows the assignment of alternatives to more than one category. Furthermore, in the preference elicitation process, the DM may also wish to assign examples to more than one category, especially when the alternative contains missing evaluations. Therefore the classification accuracy, which simply gives the percentage of alternatives in the assignment examples which were correctly assigned to the single category expressed by the DM needs to be extended to this case.

Let $\Omega^* = \big(\lambda^{\text{pes}}, \lambda^{\text{opt}}, w = (w_j, j \in J), B\big)$ be an extended MR-Sort model. We expand the definition of $\text{assig}_{\Omega^*(a)}$ and $\text{assig}_{\text{DM}}(a)$ to sets of categories to which the DM, respectively the MR-Sort model, assigns an alternative a.

The classification accuracy defined in Equation (9) could still be used in this context if we replaced the equality and inequality operations with set operations. Nevertheless, this may be seen as too strict, as only missing one category from a multi-category assignment would immediately lower this measure to 0. We may therefore consider the following modification to this measure:

$$CA' = \frac{1}{|A|}\sum_{a\in A} CA'(a), \quad CA'(a) = \begin{cases} 1 & \text{, if } \text{assig}_{\Omega^*}(a) \cap \text{assig}_{\text{DM}}(a) \neq \emptyset \\ 0 & \text{, if } \text{assig}_{\Omega^*}(a) \cap \text{assig}_{\text{DM}}(a) = \emptyset. \end{cases} \tag{10}$$

This measure, on the other hand, may be considered not strict enough, as even a slight overlap in the two sets of assignments will maximize it.

We present below several properties that we consider our fitness measure should hold, so that it can effectively discriminate between a multi-category assignment of an alternative by the DM and the multi-category assignment provided by an inferred model.

Property 1 (Maximality)**.** *The fitness measure for an alternative a reaches its maximum value only when two corresponding sets of category assignments are identical, i.e.* $assig_{\Omega^*}(a) = assig_{DM}(a)$*.*

Property 2 (Minimality)**.** *The fitness measure for an alternative a reaches its minimum value only when two corresponding sets of category assignments are exclusive, i.e.* $assig_{\Omega^*}(a) \cap assig_{DM}(a) = \emptyset$*.*

Property 3 (Monotonicity w.r.t. model assignments)**.** *Let a and b be two alternatives with the same DM assignments (*$assig_{DM}(a) = assig_{DM}(b)$*) and with* $assig_{\Omega^*}(b) \subset assig_{\Omega^*}(a)$ *and* $c_l = assig_{\Omega^*}(a) \backslash assig_{\Omega^*}(b)$*. If* $c_l \in assig_{DM}(a)$ *than the fitness of a is strictly lower than that of b, while if* $c_l \notin assig_{DM}(a)$ *than the fitness of a is strictly higher than that of b.*

Property 4 (Monotonicity w.r.t. DM assignments)**.** *Let a and b be two alternatives with the same model assignments (*$assig_{\Omega^*}(a) = assig_{\Omega^*}(b)$*) and with* $assig_{DM}(b) \subset assig_{DM}(a)$ *and* $c_l = assig_{DM}(a) \backslash assig_{DM}(b)$*. If* $c_l \in assig_{DM}(a)$ *than the fitness of a is strictly lower than that of b, while if* $c_l \notin assig_{DM}(a)$ *than the fitness of a is strictly higher than that of b.*

In order to illustrate how different fitness measures satisfy these properties, we refer to Table 3.

The measure from Equation (9) satisfies Property 1 since it is equal to 1 only when the assignment of the model is identical to that of the DM (examples 1 and 3 from Table 3). However it does not satisfy Property 2, as it is 0 for overlapping assignments (examples 5 to 10). The measure from Equation (10) satisfies Property 2 since it is 0 only when the two assignments are completely disjoint (examples 2 and 4). It does not, however, satisfy Property 1 as it is 1 for partially overlapping assignments (examples 5 to 10). Both these measures do not satisfy Property 3 (examples 5 to 7), nor Property 4 (examples 6, 8 and 9).

In order to construct a measure satisfying all four properties, we borrow several concepts from statistical analysis, such as precision and recall [19], and adapt them to our problem:

$$Pr(a) = \frac{|\mathrm{assig}_{\Omega^*}(a) \cap \mathrm{assig}_{\mathrm{DM}}(a)|}{|\mathrm{assig}_{\Omega^*}(a)|}, \quad Re(a) = \frac{|\mathrm{assig}_{\Omega^*}(a) \cap \mathrm{assig}_{\mathrm{DM}}(a)|}{|\mathrm{assig}_{\mathrm{DM}}(a)|} \quad (11)$$

In this case, *Precision* corresponds to the percentage of correctly identified categories among all assignments proposed by Ω^*, while *Recall* corresponds to the percentage of correctly identified categories among all assignments proposed by the DM.

Table 3: Illustrative example of assignment scenarios and associated fitness measures.

#	Assignments						$CA(a)$	$CA'(a)$	$Pr(a)$	$Re(a)$	$F_1(a)$
1	$\text{assig}_{\text{DM}}(a)$	c_1	c_2	$\mathbf{c_3}$	c_4	c_5	1	1	1	1	1
	$\text{assig}_{\Omega^*}(a)$	c_1	c_2	$\mathbf{c_3}$	c_4	c_5					
2	$\text{assig}_{\text{DM}}(a)$	c_1	c_2	$\mathbf{c_3}$	c_4	c_5	0	0	0	0	0
	$\text{assig}_{\Omega^*}(a)$	c_1	$\mathbf{c_2}$	c_3	c_4	c_5					
3	$\text{assig}_{\text{DM}}(a)$	c_1	c_2	$\mathbf{c_3}$	$\mathbf{c_4}$	$\mathbf{c_5}$	1	1	1	1	1
	$\text{assig}_{\Omega^*}(a)$	c_1	c_2	$\mathbf{c_3}$	$\mathbf{c_4}$	$\mathbf{c_5}$					
4	$\text{assig}_{\text{DM}}(a)$	c_1	c_2	$\mathbf{c_3}$	$\mathbf{c_4}$	$\mathbf{c_5}$	0	0	0	0	0
	$\text{assig}_{\Omega^*}(a)$	c_1	$\mathbf{c_2}$	c_3	c_4	c_5					
5	$\text{assig}_{\text{DM}}(a)$	c_1	c_2	$\mathbf{c_3}$	$\mathbf{c_4}$	$\mathbf{c_5}$	0	1	0.5	0.33	0.4
	$\text{assig}_{\Omega^*}(a)$	c_1	$\mathbf{c_2}$	$\mathbf{c_3}$	c_4	c_5					
6	$\text{assig}_{\text{DM}}(a)$	c_1	c_2	$\mathbf{c_3}$	$\mathbf{c_4}$	$\mathbf{c_5}$	0	1	0.66	0.66	0.66
	$\text{assig}_{\Omega^*}(a)$	c_1	$\mathbf{c_2}$	$\mathbf{c_3}$	$\mathbf{c_4}$	c_5					
7	$\text{assig}_{\text{DM}}(a)$	c_1	c_2	$\mathbf{c_3}$	$\mathbf{c_4}$	$\mathbf{c_5}$	0	1	0.33	0.33	0.33
	$\text{assig}_{\Omega^*}(a)$	$\mathbf{c_1}$	$\mathbf{c_2}$	$\mathbf{c_3}$	c_4	c_5					
8	$\text{assig}_{\text{DM}}(a)$	c_1	c_2	c_3	$\mathbf{c_4}$	$\mathbf{c_5}$	0	1	0.33	0.5	0.4
	$\text{assig}_{\Omega^*}(a)$	c_1	$\mathbf{c_2}$	$\mathbf{c_3}$	$\mathbf{c_4}$	c_5					
9	$\text{assig}_{\text{DM}}(a)$	c_1	c_2	$\mathbf{c_3}$	$\mathbf{c_4}$	c_5	0	1	0.66	1	0.8
	$\text{assig}_{\Omega^*}(a)$	c_1	$\mathbf{c_2}$	$\mathbf{c_3}$	$\mathbf{c_4}$	c_5					
10	$\text{assig}_{\text{DM}}(a)$	c_1	c_2	$\mathbf{c_3}$	$\mathbf{c_4}$	$\mathbf{c_5}$	0	1	1	0.66	0.8
	$\text{assig}_{\Omega^*}(a)$	c_1	c_2	$\mathbf{c_3}$	$\mathbf{c_4}$	c_5					

Precision satisfies Property 2, as it is 0 only when the assignments are disjoint (examples 2 and 4). It also satisfies Property 3 since adding (removing) a category c_l to $\text{assig}_{\Omega^*}(a)$ has an effect of raising (lowering) *Precision* when $c_l \in \text{assig}_{\text{DM}}(a)$ as seen in examples 5 and 6, while when $c_l \notin \text{assig}_{\text{DM}}(a)$ the effect on *Precision* is reversed as seen in examples 5 and 7. However, Property 1 is not satisfied since the measure is 1 also for partially overlapping assignments (example 10), and neither is Property 4 as seen through examples 6 and 9.

Recall satisfies Property 2 (examples 2 and 4), while it also satisfies Property 4. The latter is due to the fact that adding (removing) a category c_l to $\text{assig}_{\text{DM}}(a)$ has an effect of raising (lowering) *Recall* when $c_l \in \text{assig}_{\Omega^*}(a)$ as seen in examples 6 and 8 while when $c_l \notin \text{assig}_{\Omega^*}(a)$ the effect on *Recall* is reversed as seen in examples 6 and 9. However, Property 1 is not satisfied since the measure is 1 also for partially overlapping assignments (example 9), and neither is Property 3 as seen through examples 5 and 7 (adding or removing a category in the model assignment which does not correspond to the DM assignment has no effect on the performance metric).

In order to overcome these issues, we propose to use the F-measure, which combines both precision and recall as follows:

$$F_\beta = \frac{1}{|A|}\sum_{a \in A} F_\beta(a), \quad F_\beta(a) = (1+\beta^2)\frac{Pr(a) \cdot Re(a)}{\beta^2 \cdot Pr(a) + Re(a)}, \quad \beta > 0 \tag{12}$$

The special case in which $\beta = 1$ corresponds to the harmonic average between *Precision* and *Recall*. The F_1 score is illustrated together with all the other presented measures in Table 3.

$$F_1 = 2\frac{|assig_{\Omega^*} \cap assig_{\text{DM}}|}{|assig_{\Omega^*}| + |assig_{DM}|} \tag{13}$$

We observe from these examples that the F_1 measure satisfies all four properties. Property 1 is satisfied since the measure is maximal only when $assig_{\text{DM}} = assig_{\Omega^*}$ (examples 1 and 3). Property 2 is satisfied due to the measure being minimal only when $assig_{\text{DM}} \cap assig_{\Omega^*} = \emptyset$ (examples 2 and 4). Property 3 is satisfied since, for the same DM assignment, F_1 increases (decreases) with the inclusion (exclusion) of categories in the model assignment which also belong to $assig_{\text{DM}}$ (examples 5 and 6), while the effect is reversed for categories not belonging to $assig_{\text{DM}}$ (examples 6 and 7). Finally, Property 4 is satisfied since, for the same model assignment, F_1 increases (decreases) with the inclusion (exclusion) of categories to the DM assignment which also belong to $assig_{\Omega^*}$ (examples 6 and 8), while the effect is reversed for categories not belonging to $assig_{\Omega^*}$ (examples 6 and 9).

The β parameter may be used in order to place more emphasis on finding more of the correct assignments while allowing at the same time more incorrect assignments ($\beta > 1$), or proposing fewer incorrect assignments while at the same time also fewer correct ones ($\beta < 1$).

We also wish to note that the proposed fitness measure becomes identical to the classification accuracy when both the assignments of the DM and those of the inferred model become single-category.

4.2. The inference approach

Due to the F_β measure containing fractions in its definition, mathematical programs containing linear constraints may not be employed in order to solve the problem of maximizing it over a set of assignment examples. As such programs may not scale well with the size of the set of assignment examples, we propose to extend the mixed linear program and metaheuristic approach proposed in [32]. The main structure of the algorithm is presented in Algorithm 1.

Algorithm 1 Proposed approach.

Input: Initial temperature T_0 and temperature decrease dT.

$\lambda^{\text{pes}}, \lambda^{\text{opt}}, w, b = \textbf{Initialize}()$;
$best\_f = \textsc{Fitness}(\lambda, w, b, v)$;
while not $\textsc{StoppingCondition}()$ **do**
  /* Linear program for weights and majority threshold */
  $\lambda^{\text{pes}'}, \lambda^{\text{opt}'}, w^{'} = \textbf{LP}()$;
  /* Metaheuristic for category profiles */
  $b^{'} = \textbf{MH}()$;
  if $best\_f < \textsc{Fitness}(\lambda^{\text{pes}'}, \lambda^{\text{opt}'}, w^{'}, b^{'})$ **then**
    $best\_f = \textsc{Fitness}(\lambda^{\text{pes}'}, \lambda^{\text{opt}'}, w^{'}, b^{'})$;
    $\lambda^{\text{pes}}, \lambda^{\text{opt}}, w, b = \lambda^{\text{pes}'}, \lambda^{\text{opt}'}, w^{'}, b^{'}$;
  $dT = \textsc{Update}()$;

Output: $\lambda^{\text{pes}}, \lambda^{\text{opt}}, w, b$.

The algorithm starts by initializing the model parameters followed by several iterations composed of a linear program and a metaheuristic. The fitness corresponds to the average F_β measure across all provided assignment examples.

4.2.1. The initialization step

The initialization step fixes the majority thresholds to 0.5 and sets the weights for all criteria to be equal, which is a standard practice when no additional preferential information is considered. The category profiles are constructed using a greedy heuristic which considers each criterion independently from the rest and places the value of a profile b_h so that it separates as much as possible the values of the alternatives that are classified in categories above the profile and the values of the alternatives that are classified in categories below:

$$g_j(b_h) = \arg\max_{\mathbb{R}} \sum_{a \in A} h_{\text{init}}(a,h), \forall j \in J, \forall h \in \{1,\dots,k-1\} \text{ ,where} \tag{14}$$

$$h_{\text{init}}(a,h) = \begin{cases} 1 & \text{, if } \forall l \in \text{assig}_{\text{DM}}(a), l > h \text{ and } g_j^{\min}(a) \geqslant g_j(b_h) \\ & \text{or } \forall l \in \text{assig}_{\text{DM}}(a), l \leqslant h \text{ and } g_j^{\max}(a) < g_j(b_h) \\ 0 & \text{, otherwise.} \end{cases} \tag{15}$$

4.2.2. The linear programming step

By fixing the category profiles, a linear program may be employed in order to find the optimal values of the majority thresholds and the criteria weights, for the given category profiles. The program is presented in Fig. 1.

Since we try to fix the lower and upper limits of the category assignments of each alternative a, we only require to impose constraints on the relations between a and profiles $b_h^{min} - 1$, b_h^{min}, $b_h^{max} - 1$ and b_h^{max}. $C_j^{\text{pes}}(a, b_{h^{\min}-1})$,

Figure 1: Linear program for the first step.

Parameters:

$A, A^{'}, A^{''}, J$

$C_j^{\text{pes}}(a, b_{h^{\min}-1}), C_j^{\text{opt}}(a, b_{h^{\min}-1}), C_j^{\text{pes}}(a, b_{h^{\max}}), C_j^{\text{opt}}(a, b_{h^{\max}})$ $\quad \forall a \in A,\ \forall j \in J$

$C_j^{\text{pes}}(a, b_{h^{\min}}), C_j^{\text{opt}}(a, b_{h^{\min}}$ $\quad \forall a \in A^{'},\ \forall j \in J$

$C_j^{\text{pes}}(a, b_{h^{\max}-1}), C_j^{\text{opt}}(a, b_{h^{\max}-1})$ $\quad \forall a \in A^{''},\ \forall j \in J$

$\gamma \in [0,1]$

Variables:

$\lambda^{\text{pes}}, \lambda^{\text{opt}} \in]0,1]$

$w_j \in [0,1]$ $\quad \forall j \in J$

$x(a), y(a) \in [0,1]$ $\quad \forall a \in A$

$z(a), z_1(a), z_2(a) \in [0,1]$ $\quad \forall a \in A^{'}$

$t(a), t_1(a), t_2(a) \in [0,1]$ $\quad \forall a \in A^{''}$

Objective:

$$\min \sum_{a \in A} \big(x(a) + y(a)\big) + \sum_{a \in A^{'}} z(a) + \sum_{a \in A^{''}} t(a)$$

Constraints:

$$
\begin{array}{lll}
s.t. & \sum_{j \in J} w_j = 1 & \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{pes}}(a, b_{h^{\min}-1})\big) + (1 - x(a)) \geqslant \lambda^{\text{pes}} & \forall a \in A \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{opt}}(a, b_{h^{\min}-1})\big) + (1 - x(a)) \geqslant \lambda^{\text{opt}} & \forall a \in A \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{pes}}(a, b_{h^{\max}})\big) - (1 - y(a)) \leqslant \lambda^{\text{pes}} - \gamma & \forall a \in A \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{opt}}(a, b_{h^{\max}})\big) - (1 - y(a)) \leqslant \lambda^{\text{opt}} - \gamma & \forall a \in A \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{pes}}(a, b_{h^{\min}})\big) + (1 - z_1(a)) \geqslant \lambda^{\text{pes}} & \forall a \in A^{'} \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{opt}}(a, b_{h^{\min}})\big) - (1 - z_1(a)) \leqslant \lambda^{\text{opt}} - \gamma & \forall a \in A^{'} \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{opt}}(a, b_{h^{\min}})\big) + (1 - z_2(a)) \geqslant \lambda^{\text{opt}} & \forall a \in A^{'} \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{pes}}(a, b_{h^{\min}})\big) - (1 - z_2(a)) \leqslant \lambda^{\text{pes}} - \gamma & \forall a \in A^{'} \\
& z(a) \geqslant z_1(a) & \forall a \in A^{'} \\
& z(a) \geqslant z_2(a) & \forall a \in A^{'} \\
& z(a) \leqslant z_1(a) + z_2(a) & \forall a \in A^{'} \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{pes}}(a, b_{h^{\max}-1})\big) + (1 - t_1(a)) \geqslant \lambda^{\text{pes}} & \forall a \in A^{''} \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{opt}}(a, b_{h^{\max}-1})\big) - (1 - t_1(a)) \leqslant \lambda^{\text{opt}} - \gamma & \forall a \in A^{''} \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{opt}}(a, b_{h^{\max}-1})\big) + (1 - t_2(a)) \geqslant \lambda^{\text{opt}} & \forall a \in A^{''} \\
& \sum_{j \in J} \big(w_j \cdot C_j^{\text{pes}}(a, b_{h^{\max}-1})\big) - (1 - t_2(a)) \leqslant \lambda^{\text{pes}} - \gamma & \forall a \in A^{''} \\
& t(a) \geqslant t_1(a) & \forall a \in A^{''} \\
& t(a) \geqslant t_2(a) & \forall a \in A^{''} \\
& t(a) \leqslant t_1(a) + t_2(a) & \forall a \in A^{''}
\end{array}
$$

$C_j^{\text{opt}}(a, b_{h^{\min}-1})$, $C_j^{\text{pes}}(a, b_{h^{\min}})$, $C_j^{\text{opt}}(a, b_{h^{\min}})$, $C_j^{\text{pes}}(a, b_{h^{\max}-1})$, $C_j^{\text{opt}}(a, b_{h^{\max}-1})$, $C_j^{\text{pes}}(a, b_{h^{\max}})$ and $C_j^{\text{opt}}(a, b_{h^{\max}})$ are 0-1 flags corresponding to the result of the pair-wise comparison between the pessimistic and optimistic versions of an alternative and the profiles of it's lower and upper-bound category assignments. We have one flag per criterion, with 0 meaning that the alternative is strictly worse than the profile on that criterion, and 1 meaning that it is at least as good as the profile. These flags can be computed before executing the linear program since both the alternatives and the category profiles evaluations are fixed at this stage.

The linear program seeks to enforce that $c_{h^{\min}}$ is the lowest category in which a is assigned (by making a outrank profile $b_{h^{\min}-1}$ in both optimistic and pessimistic cases and not outrank $b_{h^{\min}}$ either in the optimistic or the pessimistic case) and that $c_{h^{\max}}$ is the highest one (by making a not outrank profile $b_{h^{\max}}$ in both optimistic and pessimistic cases and outrank profile $b_{h^{\max}-1}$ either in the optimistic or the pessimistic case). The x and y variables are used to ensure the limits with respect to the lower profile of the bottom category and respectively the upper profile of the top category. These variables will be 1 when their corresponding constraint is fulfilled and between 0 and 1 otherwise. Variables z_1 and z_2 are used to determine whether an alternative outranks the upper profile of its bottom category in the pessimistic case and at the same time does not outrank it in the optimistic case, or vice-versa. The z variable computes the logical OR operation between these statements. Variables t_1, t_2 and t do exactly the same for the lower profile of the top category.

Notice that some variables and constraints are limited to sets A, $A^{'}$ and $A^{''}$. Set A contains all of the alternatives, set $A^{'}$ contains those that have been assigned to more than one category while $A^{''}$ contains alternatives assigned to more than two categories. This is done in order to properly apply the constraints corresponding to intermediate category profiles.

4.2.3. The metaheuristic step

This step of the approach is used to find the category profiles while the majority thresholds and the criteria weights are fixed. The algorithm corresponds to a slight adaptation of the simulated annealing algorithm [20], as seen in Algorithm 2.

The metaheuristic performs changes to the category profiles across several iterations. Each iteration is linked to a temperature parameter which decreases over time. In the beginning, at high temperatures, the algorithm may perform more frequently changes to the profiles which would lead to a decrease of the model fitness, while towards the end, as the temperature decreases, such changes get less frequent. Every iteration gives the opportunity for each profile to have each of its values on the set of criteria changed. The heuristic function is used to determine the amount of increase or decrease in the fitness of the model given a new value on a criterion for a category profile. This function would normally be computed as the difference between the F-measures of each of the two models, however, we propose a simplification. Having already computed the initial cat-

Algorithm 2 Simulated annealing.

Input: Initial model parameters λ^{pes}, λ^{opt}, w, b and T_0, dT and α.

$T = T_0$;
while $T > 0$ **do**
  for all $j \in J$ **do**
    for all $h \in \{1, \ldots, k-1\}$ **do**
      pick α random values X in the interval $[g_j(b_{h-1}), g_j(b_{h+1})]$
      $x = \underset{x \in X}{\arg\max}(Heuristic(x))$
      if $Heuristic(x) > 0$ or $random < e^{\frac{-1}{T}}$ **then**
        $g_j(b_h) = x$
  $T = T - dT$;

Output: b.

egory assignments of all alternatives $a \in A$ as $\text{assig}_{\Omega^*}(a) = \{c_{h^{\min}}, \ldots, c_{h^{\max}}\}$, we infer the new sets of assignments $\{c_{h'^{\min}}, \ldots, c_{h'^{\max}}\}$ as follows:

$$h'^{\min} = \begin{cases} h+1, \text{ if } \big(g_j(b_h) > g_j^{\max}(a) \geqslant x \text{ and } C^{\text{pes}}(a,b_h) \geqslant \lambda^{\text{pes}} \text{ and } C^{\text{opt}}(a,b_h) < \lambda^{\text{opt}} \leqslant C^{\text{opt}}(a,b_h) + w_j\big) \\ \qquad \text{or } \big(g_j(b_h) > g_j^{\min}(a) \geqslant x \text{ and } C^{\text{opt}}(a,b_h) \geqslant \lambda^{\text{opt}} \text{ and } C^{\text{pes}}(a,b_h) < \lambda^{\text{pes}} \leqslant C^{\text{pes}}(a,b_h) + w_j\big); \\ h \quad , \text{ if } \big(x > g_j^{\max}(a) \geqslant g_j(b_h) \text{ and } C^{\text{pes}}(a,b_h) \geqslant \lambda^{\text{pes}} \text{ and } C^{\text{opt}}(a,b_h) - w_j < \lambda^{\text{opt}} \leqslant C^{\text{opt}}(a,b_h)\big) \\ \qquad \text{or } \big(x > g_j^{\min}(a) \geqslant g_j(b_h) \text{ and } C^{\text{opt}}(a,b_h) \geqslant \lambda^{\text{opt}} \text{ and } C^{\text{pes}}(a,b_h) - w_j < \lambda^{\text{pes}} \leqslant C^{\text{pes}}(a,b_h)\big); \\ h^{\min}, \text{ otherwise.} \end{cases} \tag{16}$$

$$h'^{\max} = \begin{cases} h+1, \text{ if } \big(g_j(b_h) > g_j^{\max}(a) \geqslant x \text{ and } C^{\text{pes}}(a,b_h) < \lambda^{\text{pes}} \text{ and } C^{\text{opt}}(a,b_h) < \lambda^{\text{opt}} \leqslant C^{\text{opt}}(a,b_h) + w_j\big) \\ \qquad \text{or } \big(g_j(b_h) > g_j^{\min}(a) \geqslant x \text{ and } C^{\text{opt}}(a,b_h) < \lambda^{\text{opt}} \text{ and } C^{\text{pes}}(a,b_h) < \lambda^{\text{pes}} \leqslant C^{\text{pes}}(a,b_h) + w_j\big); \\ \qquad \text{or } \big(x > g_j^{\min}(a) \geqslant g_j(b_h) \text{ and } C^{\text{opt}}(a,b_h) < \lambda^{\text{opt}} \text{ and } C^{\text{pes}}(a,b_h) - w_j < \lambda^{\text{pes}} \leqslant C^{\text{pes}}(a,b_h)\big); \\ h^{\max}, \text{ otherwise.} \end{cases} \tag{17}$$

The first line of the first term in Equation (16) considers the case where a decrease in the evaluation of b_h on criterion j, from $g_j(b_h)$ to x, leads to a change in the optimistic outranking relation between a and b_h from false to true. If at the same time the pessimistic outranking relation is also true, then a becomes at least as good as profile b_h both pessimistically and optimistically, therefore category c_{h+1} becomes the lowest category to which a is assigned. The second line of the first term considers the opposite case, where the change in evaluation of the profile leads to a outranking the profile in the pessimistic case when originally it was not, while also outranking the profile in the optimistic case.

The second term from Equation (16) considers the case where an increase in the evaluation of b_h on criterion j invalidates either the optimistic or the pessimistic outranking relation between a and the profile, when previously both the optimistic and pessimistic relations were in effect. This means that while profile b_h was initially bounding a from below, it no longer does so, and therefore the lower limit of categories to which a is assigned becomes category c_h.

In all other cases, the index of the lowest category to which a is assigned does not change from the previous value of $h^{\min}$.

Equation (17) illustrates similar situations where a change in the evaluation of b_h on one criterion leads to a change in the upper limit of the categories to which a is assigned by the model.

Computing the difference between the fitness of the original extended MR-Sort model and the new one using the previously computed ranges of categories to which all alternatives in A are assigned, gives us the value of the heuristic function used in Algorithm 2. When this value is positive, changing the evaluation of profile b_h on criterion j improves the fitness of the model, therefore the metaheuristic would more likely choose to perform this operation. Conversely, when the heuristic function is negative, the fitness of the model would decrease as a result of this operation, therefore the metaheuristic would less likely choose to perform this operation.

In the description of the heuristic we have made abstraction of the bottom (resp. top) categories which do not have a lower (resp. top) profile.

Finally, we have included an adaptive parameter dT, which increases or decreases based on each iteration improving or not the overall fitness of the best found model. The increase and decrease ratios are extracted empirically, as well as the initial temperature parameter.

5. Empirical validation

In order for the extended MR-Sort model and the proposed inference approach to be successfully used in practice, we address the following issues:

- The effect of adding missing evaluations on the range of categories to which alternatives are assigned;
- The capacity of the proposed inference approach to restore the provided assignment examples;
- The relation between the original model and the inferred one;
- The relation between problem size and required computational resources.

We will consider the most extreme case of imprecision with respect to the evaluations of the alternatives, the case of missing evaluations.

5.1. Design of the experiments

In order to address the previously listed issues, the experimental design presented in Fig. 2 was set up.

We begin by generating an initial extended MR-Sort model, denoted with Ω_o. We start by fixing the majority thresholds λ^{pes} and λ^{opt}, which are randomly generated within the $]0,1[$ interval. The criteria weights $w_j, \forall j \in J$ are randomly generated within a $]0,1[$ interval scale using the approach of [7]. In order to construct the set of category profiles, $b_h, \forall h \in \{1, ..., k-1\}$, for each criterion

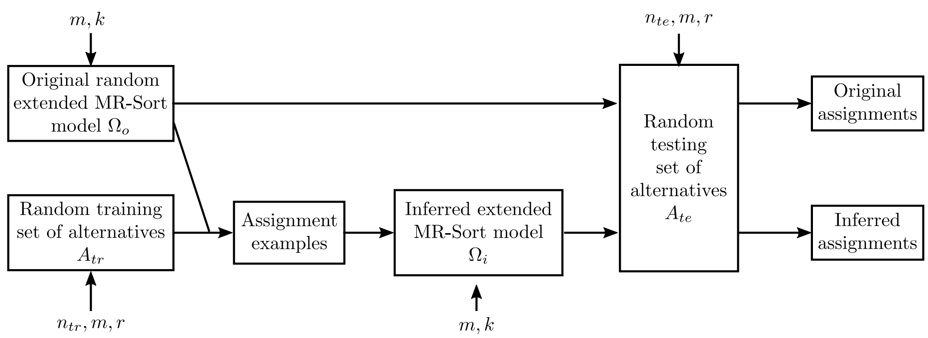


Figure 2: Design of experiments.

we generate $k-1$ values within $]0,1[$ and assign them to each category profile in ascending order.

The second step consists in generating a training set of alternatives (A_{tr}) of size n_{tr}, with the evaluations on each of the m criteria generated randomly. A given ratio r is used in order to randomly remove a part of these evaluations. Using the previously generated extended MR-Sort model, the assignments of the alternatives in A_{tr} are generated.

The inferred extended MR-Sort model Ω_i is constructed using the provided assignment examples by applying the proposed elicitation approach.

The following step involves the generation of a larger set of alternatives, A_{te}, constructed in the same way as A_{tr} and using the same ratio r of included missing evaluations. This set of alternatives is used in conjunction with both the original model and the inferred one in order to construct two sets of assignments.

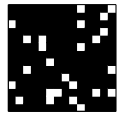
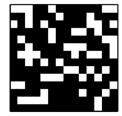
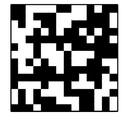
Throughout the experiments, we have used the F_1 fitness measure, placing equal emphasis on the precision and recall of a given set of category assignments.

5.2. Effect of including missing evaluations

In order to test the effect of including missing data within the alternatives' evaluations, we generated 100 problem instances for each value of $m = \{5, 10\}$, with $n_{te} = 10,000$ and $k = 5$. For each problem instance, we removed evaluations in proportion r, with $r = \{0.00, 0.15, 0.30, 0.45\}$. In Table 4 we present the distribution of the extended MR-Sort model assignments with respect to the number of categories of these assignments and the proportion r of removed evaluations.

We observe that, for 5 criteria, when all the alternatives have precise evaluations, the generated problem instances assign around 60% of the alternatives to a single category. Nevertheless, due to allowing the majority thresholds of the pessimistic and optimistic outranking relations to differ one from another, we also find 15% of the alternatives assigned to two categories, 15% to three categories, 5% to four categories and around one percent to all five. Similar

Table 4: Percentage of alternatives assigned to one or more classes (average result with standard deviation between parentheses).

Missing data		 0%	 15%	 30%	 45%
	1 categ.	63.25 (31.61)	52.85 (21.23)	42.06 (16.70)	31.07 (15.39)
	2 categ.	15.95 (12.85)	16.10 (8.52)	14.39 (5.59)	11.45 (4.24)
$m = 5$	3 categ.	15.03 (14.52)	13.93 (8.91)	12.25 (4.88)	9.56 (2.69)
	4 categ.	4.90 (6.11)	7.33 (4.72)	8.08 (3.80)	7.70 (3.32)
	5 categ.	0.86 (1.45)	9.79 (7.42)	23.21 (12.08)	40.20 (14.24)
	1 categ.	60.61 (28.22)	49.85 (20.84)	38.21 (20.65)	26.27 (18.50)
	2 categ.	24.64 (15.00)	22.39 (10.98)	16.99 (8.36)	11.24 (6.72)
$m = 10$	3 categ.	12.27 (12.36)	15.22 (7.59)	15.80 (6.29)	14.01 (5.47)
	4 categ.	2.39 (3.54)	6.48 (4.47)	9.86 (6.23)	10.42 (5.00)
	5 categ.	0.08 (0.12)	0.61 (4.94)	19.13 (11.03)	38.05 (16.42)

values may be found for 10 criteria, with a slight increase in the percentage of alternatives assigned to two categories.

When including missing evaluations in our data, we observe that more and more alternatives are assigned to more than one category. When 15% of the evaluations within the generated problem instances are removed (illustrated by the white dots in the images from Table 4), around 10% of the alternatives that were initially assigned to a single category are now assigned to more than one. Reaching 45% missing values lowers the percentage of single category assignments to around 30%. In addition it appears that the largest shift occurs with respect to alternatives assigned to all categories, a trend which decreases in strength when more criteria are involved. Nevertheless, considering the standard deviations, the differences between problem instances with 5 or 10 criteria are not significant.

5.3. Restoring the assignment examples

We continue by exploring the capacity of the proposed inference approach to construct extended MR-Sort models that restore the assignment examples provided as input. For this, we again use $m = \{5, 10\}$ and $k = \{5\}$, but also consider multiple sizes of the learning set, $n_{tr} = \{100, 300, 500\}$. For each combination of these parameters 20 problem instances were randomly generated.

We execute the proposed approach 50 times over each problem instance. The parameters corresponding the temperature step of the simulated annealing algorithm are extracted using a prior series of executions of the algorithm, giving us an initial value for dT of 0.2, which decreases or increases by a factor of 20% following the improvement or non-improvement of the overall best solution during each iteration. We present below a summary of our findings, while a detailed presentation of the results, containing the number of iterations and time needed for 05% and 95% of the algorithm executions to reach a solution fitness of 90% and 95% respectively, may be found in the Appendix, in Tables A.5 andA.6.

Throughout all of the experiments, we do not find a significant link between the number of needed iterations / execution time and the percentage of missing evaluations. This may be due to a high variability between the actual distributions of category assignments (alternatives assigned to one category, two, etc.). Nevertheless, we find that as the size of the learning set increases, fewer iterations of the algorithm are needed. We attribute this to the fact that one iteration makes use of more information as we increase the size of the assignment examples, therefore fewer iterations are needed in total. Despite this decreasing trend in the number of iterations, the execution time continues to increase with the learning set size, as each iteration needs to consider more alternatives and therefore takes longer to process.

When considering problem instances containing alternatives characterized by fewer or more criteria, we find that the number of needed iterations is inversely proportional to the size of the criteria set. This may also be due to the increased amount of information present within a set of assignment examples. Because of this, the execution time for problems with 10 criteria may be lower than for problems with 5 criteria.

5.4. Restoring the original generated model

We have also tested the ability of the proposed approach to find back the initially generated extended MR-Sort models. For every execution of the inference approach, we took the inferred preference model Ω_i as well as the original model Ω_o and used the test set of alternatives A_{te} of size $n_{te} = 10,000$ in order to compute the F_1 fitness measure. This measure allows us to accurately compare the two models by measuring the differences in their assignments of the alternatives in A_{te}. The results are presented in Fig 3.

We observe that by increasing the size of the learning set we are able to more accurately restore the original extended MR-Sort model, in some cases almost reaching the maximum accuracy. Even with 100 assignment examples, we are able to reach an overall high fitness of around 90%, while as we increase the number of criteria, we notice that the effect of increasing the number of assignment examples becomes slightly more significant. Adding missing evaluations to each dataset generally has an effect of raising the overall values of the fitness measure, which is understandable due to the fact that more and more alternatives are assigned to multiple categories, and therefore raising the probability of the final model matching some of them.

6. Conclusion and perspectives

The contributions of this paper consist in the proposal of a multi-criteria sorting model which can handle both imprecise and missing evaluations, an algorithmic solution for inferring its parameters from assignment examples given by a DM, as well as a detailed analysis of both of them.

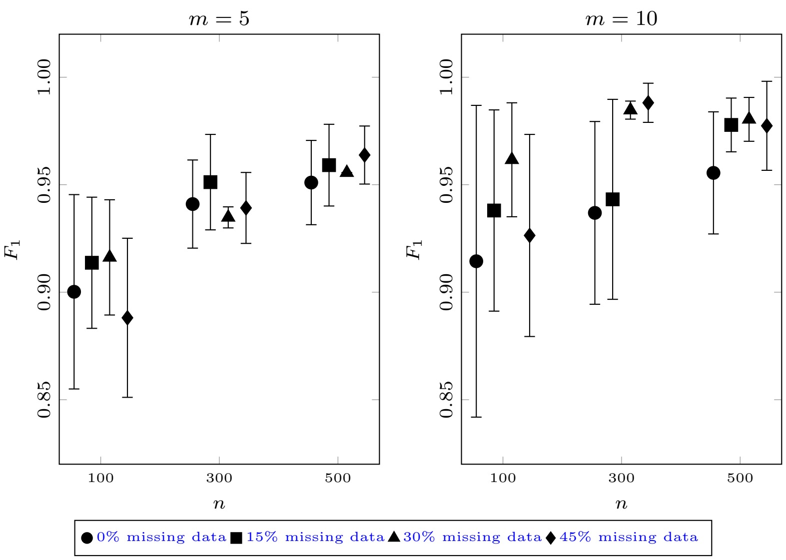


Figure 3: Average F_1 measure of the inferred model (at 95% fitness) over the test set of alternatives as a factor of the size of the learning set.

The imprecision in the alternatives evaluations is modeled in this case through intervals where all values within them are considered equally likely to occur if the imprecision source were to be removed. The proposed MR-Sort extension allows the alternatives to be assigned into one or several neighboring categories, both as input during a preference elicitation process as well as output of the classification. We furthermore consider more general assignment rules which may lead to alternatives being assigned into more than one category despite having precise evaluations on all criteria. Adding imprecision in the alternatives' evaluations has the effect of increasing the number of categories to which they are assigned, however there is no significant difference when considering problem instances characterized by fewer or more criteria.

The proposed algorithmic inference approach composed of two steps (a linear program and a simulated annealing algorithm) builds on previous work and is validated through experiments over a large set of generated problem instances. The approach scales particularly well when increasing the number of criteria. When increasing the number of assignment examples fewer iterations of the algorithm are needed, and the computation time does not increase drastically. The inferred models may also be seen to resemble well the original models, reaching fitness values close to 100% when 500 assignment examples are considered.

We further wish to consider the practical application of this approach to real-life scenarios, by extending this work in order to provide more readable model parameters as well as the integration of confidence degrees with respect to both the classification output and the assignments of the DM. We would also like to address the topic of integrating large performance differences inside the model, such as vetoes and dictators, which may increase its expressiveness. These issues will be explored in the future.

Appendix A. Experiments results

Table A.5: Minimum number of iterations and time needed to reach a given fitness value for problem instances containing 5 criteria.

Learning set size	Fitness	Missing data	90% algorithm executions iterations	90% algorithm executions time (s)	95% algorithm executions iterations	95% algorithm executions time (s)
100	90%	0%	167	563.99	246	878.56
		15%	726	2028.71	833	2145.43
		30%	787	2145.02	1256	3246.32
		45%	727	1536.67	921	1995.83
	95%	0%	373	1157.42	428	1513.82
		15%	677	1925.11	1108	3143.83
		30%	1466	3264.23	1466	3264.23
		45%	1356	3070.59	1356	3070.59
300	90%	0%	115	1189.96	150	1223.30
		15%	321	2886.65	357	3115.58
		30%	112	885.79	451	3554.13
		45%	219	1410.42	385	2741.08
	95%	0%	150	1463.26	171	1628.57
		15%	313	2839.55	313	2839.55
		30%	237	1639.50	237	1639.50
		45%	509	3202.62	509	3202.62
500	90%	0%	108	1741.22	135	2526.64
		15%	159	2502.02	202	3137.77
		30%	212	2726.21	220	3153.58
		45%	115	1267.79	123	1328.02
	95%	0%	126	1926.71	158	2683.49
		15%	214	3606.37	214	3606.37
		30%	235	3074.14	235	3074.14
		45%	216	2467.25	216	2467.25

References

[1] R. Bisdorff. Logical foundation of fuzzy preferential systems with application to the electre decision aid methods. *Computers & Operations Research*, 27(7):673 – 687, 2000.

[2] R. Bisdorff, P. Meyer, and M. Roubens. Rubis: a bipolar-valued outranking method for the choice problem. *4OR*, 6(2):143–165, 2008.

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

Table A.6: Minimum number of iterations and time needed to reach a given fitness value for problem instances containing 10 criteria.

Learning set size	Fitness	Missing data	90% algorithm executions iterations	90% algorithm executions time (s)	95% algorithm executions iterations	95% algorithm executions time (s)
100	90%	0%	74	516.79	350	2469.11
		15%	2	11.13	21	122.12
		30%	100	536.33	159	789.31
		45%	26	118.77	82	372.01
	95%	0%	355	2269.65	434	2650.97
		15%	54	281.93	63	365.00
		30%	1	3.16	2	7.57
		45%	81	362.06	236	1117.31
300	90%	0%	80	1686.32	110	1998.00
		15%	38	683.01	93	1685.28
		30%	154	2564.95	154	2564.95
		45%	1	10.66	1	10.66
	95%	0%	117	2382.42	117	2589.20
		15%	154	2696.21	154	2696.21
		30%	3	40.16	3	40.16
		45%	2	19.24	2	19.24
500	90%	0%	81	2739.81	98	3145.44
		15%	53	1647.10	88	2783.04
		30%	96	2871.62	96	2871.62
		45%	6	129.24	6	129.24
	95%	0%	24	880.23	88	3025.18
		15%	1	28.37	1	28.37
		30%	96	2871.62	96	2871.62
		45%	6	129.24	6	129.24

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

[5] D. Bouyssou, T. Marchant, M. Pirlot, A. Tsoukiàs, and Ph. Vincke. Evaluation and decision models with multiple criteria: Stepping stones for the analyst. In *International Series in Operations Research and Management Science*, volume 86. Springer, Boston, 1st edition, 2006.

[6] J.P. Brans and B. Mareschal. PROMETHEE methods. In S. Greco, editor, *Multiple Criteria Decision Analysis: State of the Art Surveys*, volume 78 of *International Series in Operations Research & Management Science*, pages 163–186. Springer New York, 2005.

[7] J. Butler, J. Jia, and J. Dyer. Simulation techniques for the sensitivity analysis of multi-criteria decision models. *European Journal of Operational Research*, 103:531–546, 1997.

[8] S. Corrente, S. Greco, and R. Słowiński. Handling imprecise evaluations in multiple criteria decision aiding and robust ordinal regression by n-point

intervals. *Fuzzy Optimization and Decision Making*, 16(2):127–157, Jun 2017.

[9] L.C. Dias and J.N. Clímaco. On computing ELECTRE's credibility indices under partial information. *Journal of Multi-Criteria Decision Analysis*, 8(2):74–92, 1999.

[10] L.C. Dias and J.N. Clímaco. ELECTRE TRI for groups with imprecise information on parameter values. *Group Decision and Negotiation*, 9(5):355–377, September 2000.

[11] L.C. Dias and V. Mousseau. IRIS: A DSS for multiple criteria sorting problems. *Journal of Multi-Criteria Decision Analysis*, 12:285–298, 2003.

[12] L.C. Dias, V. Mousseau, J. Figueira, and J.N. Clímaco. An aggregation/disaggregation approach to obtain robust conclusions with ELECTRE TRI. *European Journal of Operational Research*, 138(2):332–348, April 2002.

[13] D. Dubois and H. Prade. *Théorie des possibilités: application à la représentation des connaissances en informatique*. Masson, Paris, 1988.

[14] D. Dubois and H. Prade. *Formal Representations of Uncertainty*, pages 85–156. ISTE, Wiley, 2010.

[15] J.S. Dyer. MAUT — multiattribute utility theory. In S. Greco, editor, *Multiple Criteria Decision Analysis: State of the Art Surveys*, volume 78 of *International Series in Operations Research & Management Science*, pages 265–292. Springer New York, 2005.

[16] J. Figueira, V. Mousseau, and B. Roy. ELECTRE methods. In S. Greco, editor, *Multiple Criteria Decision Analysis: State of the Art Surveys*, volume 78 of *International Series in Operations Research & Management Science*, pages 133–153. Springer New York, 2005.

[17] S. Greco, B. Matarazzo, and R. Słowiński. Dealing with missing data in rough set analysis of multi-attribute and multi-criteria decision problems. In S.H. Zanakis, G. Doukidis, and C. Zopounidis, editors, *Decision Making: Recent Developments and Worldwide Applications*, pages 295–316. Kluwer, Dordrecht, 2000.

[18] S. Greco, B. Matarazzo, and R. Słowiński. Decision rule approach. In S. Greco, editor, *Multiple Criteria Decision Analysis: State of the Art Surveys*, volume 78 of *International Series in Operations Research & Management Science*, pages 507–555. Springer New York, 2005.

[19] A. Kent, M. M. Berry, F. U. Luehrs, and J. W. Perry. Machine literature searching viii. operational criteria for designing information retrieval systems. *American Documentation*, 6(2):93–101.

[20] S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[21] A. Leroy, V. Mousseau, and M. Pirlot. Learning the parameters of a multiple criteria sorting method. In R. Brafman, F. Roberts, and A. Tsoukiàs, editors, *Algorithmic Decision Theory*, volume 6992, pages 219–233. Springer, 2011.

[22] R.J. Little and D.B. Rubin. *Statistical analysis with missing data (2nd ed.)*. Hoboken, NJ: Wiley, 2002.

[23] T. Marchant. An axiomatic characterization of different majority concepts. *European Journal of Operational Research*, 179(1):160–173, May 2007.

[24] P. Meyer and A-L. Olteanu. Integrating large positive and negative performance differences in multicriteria majority-rule sorting models. *Computers & Operations Research*, 81:216–230, 2017.

[25] A. Motro and Ph. Smets. *Uncertainty Management in Information Systems: from Needs to Solutions*. Kluwer, Boston, 1997.

[26] V. Mousseau, L.C. Dias, and J. Figueira. Dealing with inconsistent judgments in multiple criteria sorting models. *4OR*, 4(3):145–158, 2006.

[27] V. Mousseau, L.C. Dias, J. Figueira, C. Gomes, and J.N. Clímaco. Resolving inconsistencies among constraints on the parameters of an MCDA model. *European Journal of Operational Research*, 147(1):72–93, 2003.

[28] V. Mousseau, J. Figueira, and J.P. Naux. Using assignment examples to infer weights for ELECTRE TRI method: Some experimental results. *European Journal of Operational Research*, 130(2):263–275, April 2001.

[29] V. Mousseau and R. Słowiński. Inferring an ELECTRE TRI model from assignment examples. *Journal of Global Optimization*, 12(2):157–174, 1998.

[30] V. Mousseau, R. Słowiński, and P. Zielniewicz. A user-oriented implementation of the ELECTRE TRI method integrating preference elicitation support. *Computers & Operations Research*, 27(7-8):757–777, 2000.

[31] A. Ngo The and V. Mousseau. Using assignment examples to infer category limits for the ELECTRE TRI method. *JMCDA*, 11(1):29–43, November 2002.

[32] A-L. Olteanu and P Meyer. Inferring the parameters of a majority rule sorting model with vetoes on large datasets. In *DA2PL 2014: From multiple criteria Decision Aid to Preference Learning*, pages 87 – 94. Ecole Centrale Paris and Université de Mons, 2014.

[33] B. Roy. Classement et choix en présence de points de vue multiples: La méthode ELECTRE. *Revue Francaise d'Informatique et de Recherche Opérationnelle*, 8:57–75, 1968.

[34] B. Roy. Main sources of inaccurate determination, uncertainty and imprecision in decision models. *Mathematical and Computer Modeling*, 12(10–11):1245 – 1254, 1989.

[35] B. Roy. The outranking approach and the foundations of ELECTRE methods. *Theory and Decision*, 31:49–73, 1991.

[36] B. Roy. Paradigms and challenges. In S. Greco, editor, *Multiple Criteria Decision Analysis: State of the Art Surveys*, volume 78 of *International Series in Operations Research & Management Science*, pages 3–24. Springer New York, 2005.

[37] B. Roy and D. Bouyssou. *Aide Multicritère à la Décision : Méthodes et Cas*. Economica, Paris, 1993.

[38] B. Roy and R. Słowiński. Handling effects of reinforced preference and counter-veto in credibility of outranking. *European Journal of Operational Research*, 188(1):185–190, 2008.

[39] J.L. Schafer. *Analysis of incomplete multivariate data*. Boca Raton, FL: Chapman & Hall, 1997.

[40] O. Sobrie, V. Mousseau, and M. Pirlot. Learning a majority rule model from large sets of assignment examples. In *Algorithmic Decision Theory*, pages 336–350. Springer Berlin Heidelberg, 2013.

[41] O. Sobrie, V. Mousseau, and M. Pirlot. A population-based algorithm for learning a majority rule sorting model with coalitional veto. In H. Trautmann, G. Rudolph, K. Klamroth, O. Schütze, M. Wiecek, Y. Jin, and C. Grimme, editors, *Evolutionary Multi-Criterion Optimization*, pages 575–589, Cham, 2017. Springer International Publishing.

[42] J. Stefanowski and A. Tsoukiàs. Incomplete information tables and rough classification. *Computational Intelligence*, 17(3):545–566, 2001.

[43] J-M. Tacnet. *Prise en compte de l'incertitude dans l'expertise des risques naturels en montagne par analyse multicritères et fusion d'information*. Phd. thesis, Ecole Nationale Supérieure des Mines de Saint-Etienne, November 2009.