Master BDMA
Decision Modeling
Project

Elaboration of some decision models for the Nutri-Score label

1 What is the Nutri-Score?

The Nutri-Score is a nutrition label that converts the nutritional value of products into a simple code consisting of 5 letters, each with its own colour. Each product is awarded a score based on a scientific algorithm. This formula takes into account the nutrients to avoid (energy value and the amount of sugars, saturated fats and salt) and the positive ones (the amount of fibre, protein, fruit, vegetables and nuts). **You can therefore see at a glance which products are recommended and which should be avoided** [1].

In France, the Nutri-Score logo (see Figure 1) was elaborated by Santé publique France, a department of the Health Ministry, based on the scientific works of Professor Serge Hercberg (University Paris 13) and the experts of ANSES (Agence nationale de sécurité sanitaire de l'alimentation, de l'environnement et du travail), another department of this ministry.

Figure 1: Nutri-Score logo

A collection of scientific articles and documents related to the Nutri-Score is available at: `https://solidarites-sante gouv.fr/prevention-en-sante/preserver-sa-sante/nutrition/article/articles-scientifique`

Nutri-Score calculation method (March 2025)

This section is based on the ANSES scientific and technical support report on Nutri-Score. The Nutri-Score computes the nutritional score of food products based on:

- Nutrients or components to encourage: fiber, protein, fruits, vegetables, legumes and nuts;
- Nutrients or components to limit: energy (kcal), saturated fatty acids, sugars, and salt.

The score is calculated per 100 grams of product (or per 100 ml for beverages). More formally, the Nutri-Score is based on the nutritional scoring model defined by Rayner et al., which integrates:

- A "negative" component calculated from the contents of nutrients to be limited: energy (kJ/100g), simple sugars (g/100g), saturated fatty acids (g/100g), and sodium (mg/100g);
- A "positive" component that accounts for nutrients recommended for consumption: fiber (g/100g) and protein (g/100g);
- A second "positive" component calculated from the contents of a specific food category: fruits/vegetables/nuts (g/100g).

Each of the positive and negative components is associated with a point score depending on the nutritional composition of the product:

[1] `https://nutriscore.colruytgroup.com/colruytgroup/en/about-nutri-score`

- From 0 to 20 points for nutrients in the "negative" component N (see Table 1);
- From 0 to 7 points for elements in the "positive" component P (see Table 2).

Therefore, the negative component score can theoretically range from 0 to 55, while the positive component score can range from 0 to 17. In most cases, the nutritional score of a food is computed by subtracting the positive component score from the negative component score (see Equation (1)).

Points	Energy (KJ/100 g)	Saturated fatty acids (g/100 g)	Sugars (g/100 g)	Salt (g/100 g)
0	≤ 335	≤ 1	≤ 3.4	≤ 0.2
1	> 335	> 1	> 3.4	> 0.2
2	> 670	> 2	> 6.8	> 0.4
3	> 1005	> 3	> 10	> 0.6
4	> 1340	> 4	> 14	> 0.8
5	> 1675	> 5	> 17	> 1
6	> 2010	> 6	> 20	> 1.2
7	> 2345	> 7	> 24	> 1.4
8	> 2680	> 8	> 27	> 1.6
9	> 3015	> 9	> 31	> 1.8
10	> 3350	> 10	> 34	> 2
11			> 37	> 2.2
12			> 41	> 2.4
13			> 44	> 2.6
14			> 48	> 2.8
15			> 51	> 3
16				> 3.2
17				> 3.4
18				> 3.6
19				> 3.8
20				> 4

Table 1: Points attributed to a food w.r.t. the negative N component

Points	Proteins (g/100 g)	Fibers (g/100 g)	Fruits, vegetables & legumes (%)
0	≤ 2.4	≤ 3.0	≤ 40
1	> 2.4	> 3.0	> 40
2	> 4.8	> 4.1	> 60
3	> 7.2	> 5.2	-
4	> 9.6	> 6.3	-
5	> 12	> 7.4	> 80
6	> 14		
7	> 17		

Table 2: Points attributed to a food w.r.t. the positive P component

$$\text{Nutritional-Score} = \text{nutrients to limit (N)} - \text{nutrients to favor (P)} \tag{1}$$

However, if the negative component score is greater than or equal to 11 and the content of fruits/vegetables/nuts does not exceed 80%, then proteins are not taken into account in the nutritional score calculation.

The lower the final nutritional score, the more favorable the product's nutritional profile. The assignment of a product to a class based on its Nutri-Score is defined by thresholds for the five categories (see Table 3).

Score range	Class	Color
[-17;0]	A	Dark green
[1;2]	B	Light green
[3,10]	C	Yellow
[11;18]	D	Light orange
[19;55]	E	Red or Dark orange

Table 3: 5-color nutritional classification

Examples

The computation of the Nuti-Score for two foods are given below :

- "Gerble-Sesame Cookie 230g" (see Figure 2 or `https://fr-en.openfoodfacts.org/product/31756800114 gerble-sesame-cookie-230g-8-2oz`)
- "Pain de mie grandes tranches Seigle & Graines 500g- La Boulangère" (see Figure 3 or `https://fr-en.openfoodfact org/product/3760049798609/pain-de-mie-grandes-tranches-seigle-graines-500g-la-boul`

2 The Nutri-Score viewed as a Mutlti-Criteria Decision Analysis problem

In this project, we will consider the calculation of the Nutri-Score of a food as a Multi-Criteria Decision Analysis problem where:

- The set of alternatives X corresponds to the foods analyzed.
- The seven criteria (the set N) to take into account are:
  1. en: **Energy** (KJ) (criterion to be minimized)
  2. su: **Sugars** (g) (criterion to be minimized)
  3. fa: **Saturated fatty acids** (g) (criterion to be minimized)
  4. sa: **Salt** (g) (criterion to be minimized)
  5. pr: **Proteins** (g) (criterion to be maximized)
  6. fi: **Fiber** (g) (criterion to be maximized)
  7. fr: **Fruit/vegetable** (%) (criterion to be maximized)

3 Green-score

The Green-score is an indicator that represents the environmental impact of food products. It classifies products from A (low) to E (high) according to their impact on the environment. Green-score aims to inform consumers on the environmental impact of the food they choose so as to guide them towards more responsible consumption. This indicator helps producers and consumers to decide and to be responsible.

The computation of the Green-score is based on the following principles:

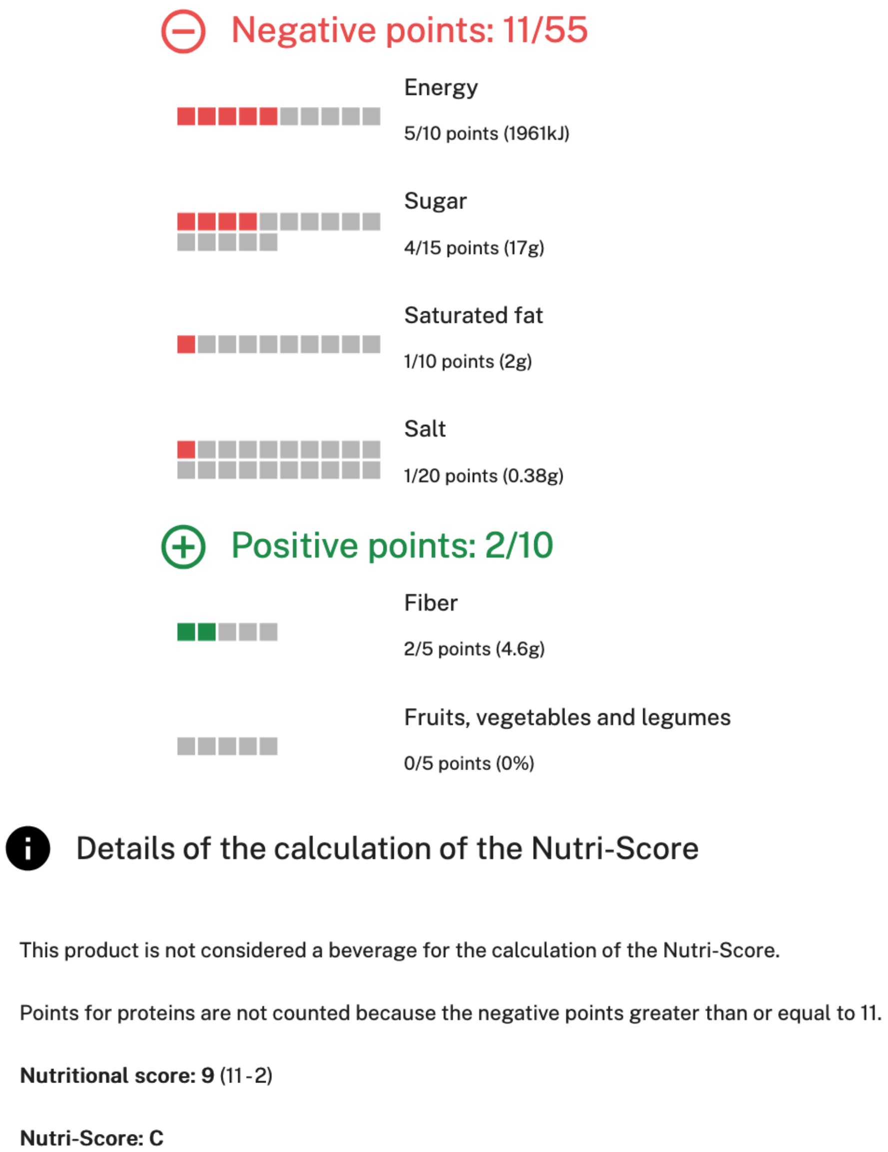

Figure 2: Computation of Nutri-Score of "Gerble-Sesame Cookie"

- Public data : quantitative data on product Life Cycle Assessment (LCA) from the Agribalyse database `https://doc.agribalyse.fr/documentation-en`, drawn up by experts and implemented in Agribalyse. Impacts on environment through production, transport, fabrication, and packaging are taken into account, giving a score out of 100.
- Data not included in LCA but which do take into account the positive or negative impact on the environment: data on the product label or given by the producer, as well as additional quality criteria : recyclability of packages, labels (Bio, quality etc.), where the ingredients come from, seasonality of food used (for recipes and ready meals). All these data will give a bonus/malus which will influence the score.
- The total mark out of 100 gives a score from A to E.

See `https://docs.score-environnemental.com/en` for more details about the Green-score.

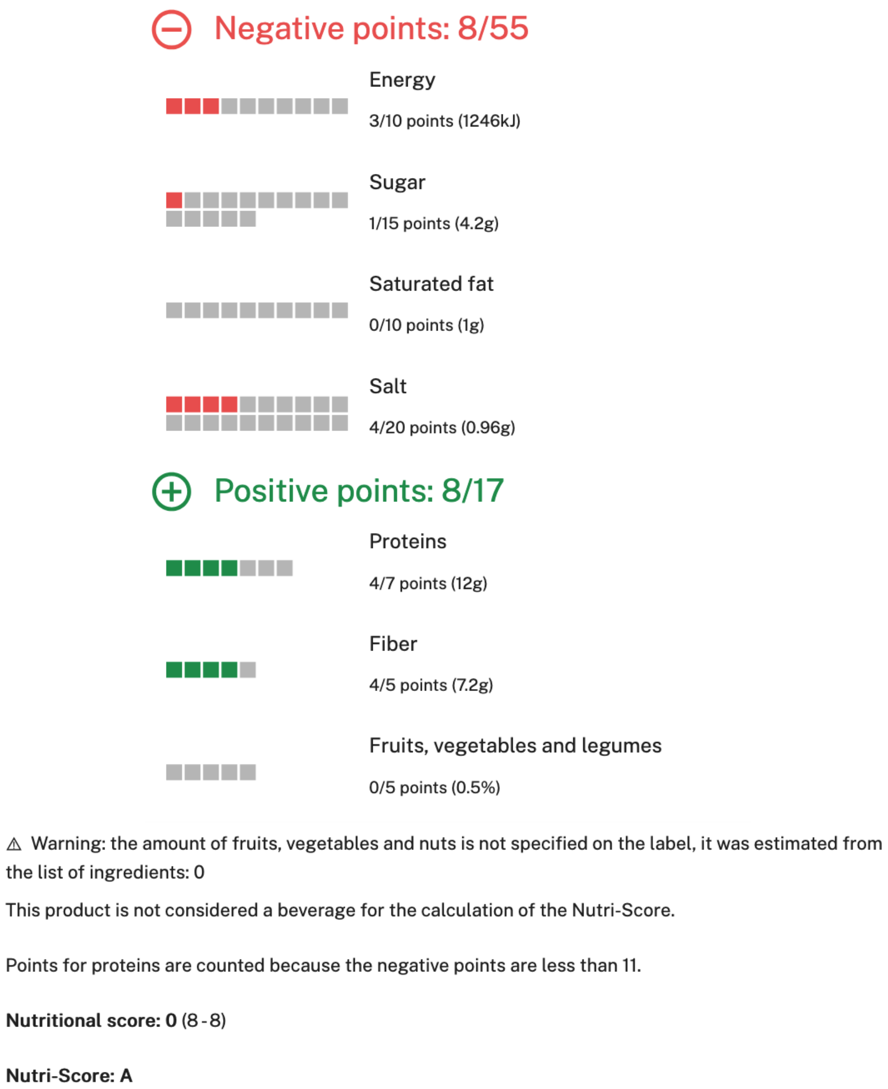

Figure 3: Computation of Nutri-Score of Pain de mie grandes tranches Seigle

Figure 4 shows the Green-Score of the two foods "Gerble-Sesame Cookie 230g" and "Pain de mie grandes tranches Seigle".

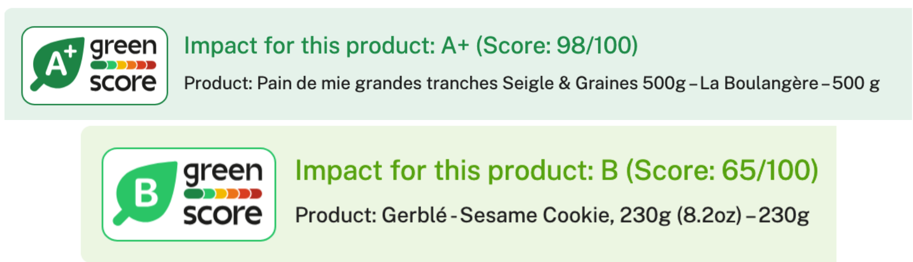

Figure 4: Green-Score label of "Gerble-Sesame Cookie 230g" and "Pain de mie grandes tranches Seigle"

4 Minimum requirements

4.1 Elaboration of your own database of foods as the set of alternatives

1. Each group have to elaborate his own set of alternatives, in an Excel file, from one or some categories of foods (e.g. breakfast cereals, cookies, aperitif, industrial ready-made dishes based on seafood, industrial ready-made dishes based on vegetables, industrial ready-made dishes based on meat,...). This database will contain at least 200 foods.
2. Each Nutri-Score label (A, B, C, D, E) must be represented by at least 15% of the total of products.
3. The Green-Score label (from A to E) and its global score, associated to each food, should be integrated in your database.
4. Each Green-Score label (A, B, C, D, E) must be represented by at least 10% of the total of products.
5. The set of alternatives to be evaluated will be the set of foods contained in this database.
6. The databases of any two different groups should not contain, in common, more than 30% of foods. Each group will implement a python function comparing two databases and allowing to satisfy this requirement.
7. The information about many foods is available by using the following this link: `https://fr-en.openfoodfacts.org/`. In this website, you can choose, for instance, the country of the products you want to evaluate. For instance, the information about foods of Spain are available in `https://es.openfoodfacts.org/`. The computation of the Nutri-Score label is given for all the foods.

4.2 Implementation and visualization of the Nutri-Score algorithm

Each group will implement:

- The current Nutri-Score algorithm in Python;
- A graphical user interface (in Python, HTML, or any other language) that allows a user to input the seven required components and display the computed Nutri-Score label and score.

4.3 Elaboration of a Nutri-Score model based on a simple sorting (ordered classification) model

1. Implement functions `PessimisticMajoritySorting` and `OptimisticMajoritySorting` based respectively on the pessimistic and optimistic assignment procedures of the ELECTRE TRI method applied to the seven Nutri-Score criteria and the Green-Score criterion. As with the Nutri-Score, you will consider five classes named A', B', C', D', and E', and define their semantics clearly.

   The implemented functions should return an Excel file containing the food products and the classes to which they are assigned.

   - The Excel file containing the food products will be one of the inputs to your functions.
   - Each group will define, with justification, a set of weights for testing.
   - Each group will propose a method to identify the six limiting profiles shown in Table 4. The constructed limiting profiles should be justified. They will be dedicated to an ELECTRE TRI application to the database developed by your group.

   Possible approaches include using information from Figure 5 on the definition of the Nutri-Score classes, k-nearest neighbors algorithms, or statistical methods related to quantile determination. Alternatively, a group member may act as the decision maker to set these boundary profiles.

π^6: Upper bound of limiting profiles
π^5: Limiting profile between A' and B'
π^4: Limiting profile between B' and C'
π^3: Limiting profile between C' and D'
π^2: Limiting profile between D' and E'
π^1: Lower bound of limiting profiles

Table 4: Limiting profiles to be determined

2. You will test your functions with at least the two majority thresholds $\lambda = 0.6$ and $\lambda = 0.7$.

   The two implemented ELECTRE TRI functions will allow you to obtain your own classification models for food products.

3. Determine all assignments obtained for products from your database.

4. Compare your results with the assignments given by the Nutri-Score (for example using a confusion matrix, class-wise proportions, dashboards, etc.).

4.4 Elaboration of a weighted sum model combining Nutri-Score and Green-Score

Elaborate a weighted sum model combining the criteria of the Nutri-score and the criterion Green-score. You will justify your choice about the weights and compare the results with one of your previous ELECTRE-TRI models.

4.5 Elaboration of Nutri-Score+Eco-score model based on a machine learning classification model

Is it possible to elaborate a combination of Nutri-Score and Eco-score by using a machine learning classification model ? Justify your answers. If yes, explain the obtained model and compare the obtained results with one of your previous results.

4.6 A comparison with another group

Apply one of your MCDA models to a database used by another group. Compare the obtained results with their results.

5 Other matters and deadlines

1. For this project, each group will be constituted by **two or three students**.

2. Each group will present (by using a power-point slides, for instance), on **December 8th 2025**, their preliminary results during 15 minutes.

3. Each group will write a report (document in .doc or .pdf) explaining and justifying their results, the parameters chosen, the interpretation of the obtained results, etc.

4. Which model you seem comfortable with (among all the models developed in this project)? Justify all your answers.

5. Any additional initiative for the development of the project is encouraged, will be appreciated, and will be taken into account in the final grade.

6. Send your report and source files by the **January 11th 2026, 23h59 (Paris hour)**.

A ELECTRE TRI methods

A.1 Elaboration of the outranking relation $\mathcal{S}_\lambda$

Let A be a set of alternatives evaluated on n real-valued criteria $g_i : A \to \mathbb{R}$, $i \in N = \{1, \ldots, n\}$. We denote by $g_i(a)$ the performance of the alternative a on criterion i. A nonnegative weight w_i is also assigned to each criterion i (w.l.o.g. we suppose $\sum_{i=1}^{n} w_i = 1$).

We associate with each criterion $i \in N$, a nonnegative preference threshold $p_i \geq 0$. If the value $g_i(a) - g_i(b)$ is positive but less than p_i, it is supposed that this difference is not significant, given the way g_i has been built. Hence, on this criterion, the two alternatives should be considered indifferent.

Using this information, we define on each criterion $i \in N$ the partial concordance index $c_i : A \times A \to [0, 1]$ as follows:

$$c_i(a,b) = \begin{cases} 1 \text{ if } g_i(b) - g_i(a) \leq p_i \\ 0 \text{ if } g_i(b) - g_i(a) > p_i \end{cases} \tag{2}$$

The valued relations c_i are aggregated to a single concordance index $c : A \times A \to \mathbb{R}$ by using the following Equation:

$$c(a,b) = \sum_{i=1}^{n} w_i c_i(a,b) \tag{3}$$

The binary relation on A called outranking relation is defined by:

$$a\ \mathcal{S}_\lambda\ b \text{ iff } c(a,b) \geq \lambda \tag{4}$$

where $\lambda \in [0, 1]$ is a cutting level (usually called a threshold and taken above $\frac{1}{2}$).

Interpretation: An alternative $a \in A$ outranks an alternative $b \in A$ if it can be considered at "least as good" as the latter (i.e., a is not worse than b), given the values (performances) of a and b at the n criteria. If a is not worse than b in every criterion, then it is obvious that $a\ \mathcal{S}_\lambda\ b$. However, if there are some criteria where a is worse than b, then a may outrank b or not, depending on the relative importance of those criteria and the differences in the evaluations (small differences might be ignored).

From $\mathcal{S}_\lambda$ we derive the following three binary relations:

☞ "Strictly better than" relation:

$$a\ \mathcal{P}_\lambda\ b \text{ iff } [a\ \mathcal{S}_\lambda\ b \text{ and not}(b\ \mathcal{S}_\lambda\ a)] \tag{5}$$

☞ "Indifferent to" relation:

$$a\ \mathcal{P}_\lambda\ b \text{ iff } [a\ \mathcal{S}_\lambda\ b \text{ and } (b\ \mathcal{S}_\lambda\ a)] \tag{6}$$

☞ "Incomparable to" relation:

$$a\ \mathcal{P}_\lambda\ b \text{ iff } [\text{not}(a\ \mathcal{S}_\lambda\ b) \text{ and not}(b\ \mathcal{S}_\lambda\ a)] \tag{7}$$

A.2 ELECTRE TRI (also called ELECTRE TRI B)

Let us consider r ordered categories $C^1, C^2, \ldots, C^r$, C^1 is the worst one and C^r is the best one. The category C^k is modeled by using limiting profiles. The lower limiting profile of C^k is π^k. The upper limiting profile of C^k is π^{k+1}. We suppose that the limiting profiles are such that π^{k+1} strictly dominates π^k [2]. The profile π^1 (respectively π^{r+1}) is taken low (respectively

[2] An alternative a *dominates* an alternative b, we note $a\ \Delta\ b$ iff [for all $i \in N$, $g_i(a) - g_i(b) \geq 0$]. a strictly dominates b if [$a\ \Delta\ b$ and not($b\ \Delta\ a$)]

high). It will be convenient to suppose that $\pi^k \in A$, for each $k = 2, 3, \ldots, r$, while $\pi^1, \pi^{r+1} \notin A$. With this convention we have

$$\text{For all } a \in A, a \mathcal{P}_\lambda \pi^1 \text{ and } \pi^{r+1} \mathcal{P}_\lambda a. \tag{8}$$

ELECTRE TRI ([9], chap. 6) renamed ELECTRE TRI-B by Almeida-Dias et al. [4] is a MultiCriteria Decision Aid method using limiting profiles. It has two versions called "pessimistic" and "optimistic" in [9]. In [8] the name "pseudo-conjunctive" is used for the "pessimistic" version and "pseudo-disjunctive" for the "optimistic" version. These two versions are defined as follows:

Définition 1 (Pessimistic version: ETRI-B-pc)**.** *Decrease* k *from* $r+1$ *until the first value* k *such that* $a \mathcal{S}_\lambda \pi^k$. *Assign alternative* a *to* C^k.

ETRI-B-pc assigns an alternative a to the unique category C^k such that a is at least as good as to the lower limiting profile of this category and is not at least as good as its upper limiting profile (the relation "at least as good as" being $\mathcal{S}_\lambda$).

Définition 2 (Optimistic version: ETRI-B-pd)**.** *Increase* k *from* 1 *until the first value* k *such that* $\pi^k \mathcal{P}_\lambda a$. *Assign alternative* a *to* C^{k-1}.

ETRI-B-pd assigns an alternative a to the category C^k such that the upper limiting profile of this category is better than a and the lower limiting profile of this category is not better than a (the relation "better than" being $\mathcal{P}_\lambda$).

Remarque 1. *Roy and Bouyssou ([9],chap.6,pp.393-395) have shown that if* $a \in A$ *is assigned to the category* C^k *by the pessimistic version and to the category* C^l *by the Optimistic version, then* $k \leq l$.

A.3 Majority Rule sorting procedure (MR-Sort)

MR-Sort is a simplified version of the ELECTRE TRI sorting model directly inspired by the work of Bouyssou and Marchant [1, 2] who provide an axiomatic characterization of non-compensatory sorting methods. The general principle of MR-Sort (without veto) is to assign alternatives by comparing their performances to those of profiles delimiting the categories. An alternative is assigned to a category "above" a profile if and only if it is at least as good as the profile on a (weighted) majority of criteria.

The condition for an alternative $a \in A$ to be assigned to a category C^k is expressed as follows:

$$\sum_{i:g_i(a) \geq g_i(\pi^{k-1})} w_i \geq \lambda \text{ and } \sum_{i:g_i(a) \geq g_i(\pi^k)} w_i < \lambda \tag{9}$$

The MR-Sort assignment rule described above involves $r \times n + 1$ parameters, i.e., n weights, $(r-1) \times n$ profiles evaluations and 1 majority threshold.

As demonstrated in [6], the problem of learning the parameters of a MR-Sort model on the basis of assignment examples can be formulated as a mixed integer linear program (MILP) but only instances of modest size can be solved in reasonable computing times. The MILP proposed in [6] contains $m \times (2n + 1)$ binary variables, with n, the number of criteria, and m, the number of alternatives. A problem involving 1000 alternatives, 10 criteria and 5 categories requires 21000 binary variables. For a similar program in [3], it is mentioned that problems with less than 400 binary variables can be solved within 90 minutes.

In [5] a genetic algorithm was proposed to learn the parameters of an ELECTRE TRI model. This algorithm could be transposed for learning the parameters of a MR-Sort model. However, it is well known in [7] that genetic algorithms which take the structure of the problem into account to perform crossovers and mutations give better results. It is not the case of the genetic algorithm proposed in [5] since the authors? definitions of crossover and mutation operators are standard.

Learning only the weights and the majority threshold of an MR-Sort model on the basis of assignment examples can be done using an ordinary linear program (without binary or integer variables). On the contrary, learning profiles evaluations is not possible by linear programming without binary variables. Taking these observations into account, [10] proposes an algorithm that takes advantage of the ease of learning the weights and the majority threshold by a linear program and adjusts the profiles by means of a dedicated heuristic. This algorithm uses the following components:

1. a heuristic for initializing the profiles;

2. a linear program learning the weights and the majority threshold, given the profiles;

3. a dedicated heuristic adjusting the profiles, given weights and a majority threshold.

References

[1] D. Bouyssou and T. Marchant. An axiomatic approach to noncompensatory scoring methods in MCDM, I: The case of two categories. *Eur. J. of Operational Research*, 178:217–245, 2007.

[2] D. Bouyssou and Th. Marchant. An axiomatic approach to noncompensatory scoring methods in MCDM, II: More than two categories. *Eur. J. of Operational Research*, 178:246–276, 2007.

[3] O. Cailloux, P. Meyer, and V. Mousseau. Eliciting electre tri category limits for a group of decision makers. *European Journal of Operational Research*, 223(1):133–140, 2012.

[4] J. Almeida Dias, J. Rui Figueira, and B. Roy. Electre tri-c: A multiple criteria sorting method based on characteristic reference actions. *European Journal of Operational Research*, 204(3):565–580, 2010.

[5] M. Doumpos, Y. Marinakis, M. Marinaki, and C. Zopounidis. An evolutionary approach to construction of outranking models for multicriteria classification: The case of the ELECTRE TRI method. *European Journal of Operational Research*, 199(2):496–505, 2009.

[6] A. Leroy, V. Mousseau, and M. Pirlot. Learning the parameters of a multiple criteria sorting method. In Ronen I. Brafman, Fred S. Roberts, and Alexis Tsoukiàs, editors, *Algorithmic Decision Theory - Second International Conference, ADT 2011, Piscataway, NJ, USA, October 26-28, 2011. Proceedings*, volume 6992 of *Lecture Notes in Computer Science*, pages 219–233. Springer, 2011.

[7] M. Pirlot. General local search methods. *European Journal of Operational Research*, 92(3):493–511, 1996.

[8] B. Roy. Présentation et interprétation de la méthode ELECTRE TRI pour affecter des zones dans des catégories de risque (p. 25). *Document du LAMSADE*, (124), 2002.

[9] B. Roy and D. Bouyssou. *Aide multicritère à la décision : Méthodes et Cas*. Economica, 1993.

[10] O. Sobrie, V. Mousseau, and M. Pirlot. Learning a majority rule model from large sets of assignment examples. In Patrice Perny, Marc Pirlot, and Alexis Tsoukiàs, editors, *Algorithmic Decision Theory - Third International Conference, ADT 2013, Bruxelles, Belgium, November 12-14, 2013, Proceedings*, volume 8176 of *Lecture Notes in Computer Science*, pages 336–350. Springer, 2013.