

Logistic regression for criteria weight elicitation in PROMETHEE-based ranking methods

Elia Balugani¹ Francesco Lolli¹ Maria Angela Butturi¹
Alessio Ishizaka² Miguel Afonso Sellitto³

¹ University of Modena and Reggio Emilia,
Department of Sciences and Methods for Engineering,
Via Amendola 2, Padiglione Morselli, 42122 Reggio Emilia, Italy
{elia.balugani, francesco.lolli, mariaangela.butturi}@unimore.it

² NEOMA Business school,
Rue du Maréchal Juin 1, 76130 Mont-Saint-Aignan, France
alessio.ishizaka@neoma-bs.fr

³ Universidade do Vale do Rio dos Sinos,
São Leopoldo, RS 93022750, Brazil
sellitto@unisinós.br

Abstract. For a PROMETHEE II method used to rank concurrent alternatives both preference functions and weights are required, and if the weights are unknown, they can be elicited by leveraging present or past partial rankings. If the known partial ranking is incorrect, the eliciting methods are ineffective. In this paper a logistic regression method for weight elicitation is proposed to tackle this scenario. An experiment is carried out to compare the logistic regression method performance against a state-of-the-art linear weight elicitation method, proving the validity of the proposed methodology.

Keywords: MCDM · Elicitation · Criteria Weights · Outranking · PROMETHEE · Machine Learning · Logistic Regression

DOI: 10.1007/978-3-030-39512-4_74

Link: https://link.springer.com/chapter/10.1007/978-3-030-39512-4_74

1 Introduction

Multicriteria decision making (MCDM) methods compare multi-dimensional alternatives, presented to a decision maker (DM), and rank them according to his/her preferences. MCDM methods use preference functions to measure the DM's specific preferences, and to translate the alternatives into comparable units. The resulting preferences are then weighted to obtain a scalar value for each alternative. This scalar value can be directly used to rank the alternatives. Readers interested in the theoretical foundation of MCDM methods alongside an established MCDM method, can refer to [1], while those interested in a recent application in the Life-cycle assessment (LCA) context can refer to [2]. The availability of credible weights can severely impact the performance of most MCDM methods [3], thus a series of weight eliciting procedures have been designed [4]. This paper expands the methods proposed in [5] for weights eliciting in the PROMETHEE II context.

Important components required for the application of MCDM methods are often missing, both the preference functions and the weights might not be known. In case the weights are unknown, information on the ranking of the alternatives can be used to elicit them. In the literature specific weighting eliciting procedure has been designed for different MCDM methods, [6] propose an eliciting procedures for TOPSIS, [7] focus on a surrogate weighting procedure in PROMETHEE, and [8] revise Simo's procedure for the ELECTRE method. From a broader perspective, [9] propose a posterior analysis using the popular Simple Additive Weighting (SAW) method while [10] focus of multicriteria additive models. For the PROMETHEE method, if a partial ranking of present or past decisions is available, [5] propose various elicitation methods based on linear and convex constrained optimization.

If the preference functions are also unknown, Robust Ordinal Regression (ROR) methods bypass the elicitation problem by providing all the results obtainable using preference functions that are in line with a known partial ranking. Interested readers can refer to the first ROR publication [11], which is a re-design of the UTA method [1] in a robust framework. The PROMETHEE method re-designed in a ROR framework can be found in [12], while the ROR version of ELECTRE has been designed by [13].

Machine learning (ML) algorithms, like logistic regression, are often used in conjunction with MCDM methods. [14] outline the similarities and differences between ML algorithms and MCDM methods, while [15] bridge the gap between ML and ROR methods. Recent applications of K-means for AHP can be found in [16] and decision tree algorithms for Data Envelop Analysis (DEA) in [17].

In this paper a logistic regression algorithm is used in the PROMETHEE weight elicitation context of [5] rather than a linear model. The linear and logistic regression algorithms are experimentally compared in cases where the known partial ranking is incorrect.

This paper is organized as follows: in Section 2 the proposed weight elicitation models is described. In Section 3 the experimental setting is outlined, and its results are analysed. Section 4 concludes the paper with a summary of the key findings and suggestions for future research.

2 Logistic regression model

Using the formalism presented in [5], if all the preference between alternatives are known they can be used to elicit unknown weights solving the optimization problem:

$$\max \sum_{\{i,k\}: \bar{a}_i P \bar{a}_k} \log \left(\frac{1}{1 + e^{\sum_{j=1}^n -((P_j)_{ik} - (P_j)_{ki}) \cdot (w)_j}} \right). \quad (1)$$

s.t.

$$\sum_{j=1}^n (w)_j = 1. \quad (2)$$

$$0 \leq (w)_j \leq 1 \quad \forall j \in \{1, \dots, n\}. \quad (3)$$

(1) maximizes the log-likelihood of identifying preferred alternatives by linearly separating the preference space. Each alternative i preferred over an alternative k is a success event drawn from a Bernoulli distribution. The distribution is characterized by a success probability p_{ik} parametrized, in the preference space, by net flow differences through the inverse canonical link function. If the linear predictor for the inverse canonical link function is defined without an intercept its parameters are maximum likelihood estimators for the PROMETHEE II weights. Since no failure event is considered, the Bernoulli distribution log-likelihood simplifies to (1).

This model is effective even if some of the known preferences are incorrect. In addition, the probabilistic interpretation of the weights allows the DM to identify the known preferences that are most likely to be faulty.

3 Experimental setting and results

3.1 Experimental setting

The experiment objective was to compare the two models in scenarios where some of the known preferences are incorrect. The Linear Model is expected to outperform the Logistic Regression Model if all the known preferences are correct, while erroneous information is expected to favour the proposed model over the original one. The Linear Model is also expected to be unable to find a feasible solution if wildly incorrect inputs are provided.

The dataset for alternatives and weights is used in [5], which contains 5-dimensional weights for 1000 DMs and their rankings of 100 different alternatives each. It is artificially expanded by permuting the rankings 1000 times, with each permutation being independent from the previous one, and it affects all the rankings leading to 1,000,000 permuted rankings.

The preference function for each weight j is linear in the entire interval $[\min(D_j)_{ik}, \max(D_j)_{ik}]$.

For each DM l and permutation h both the Linear Method and the Logistic Regression Method are applied thereby obtaining two sets of elicited weights. Each set s of elicited weights is compared with the DM weights thus obtaining a performance measure in the interval $[0,1]$:

$$performance_{slh} = \frac{\sum_{j=1}^n (we_{slh})_j \cdot (w_l)_j}{\sqrt{\sum_{j=1}^n (we_{slh})_j^2} \cdot \sqrt{\sum_{j=1}^n (w_l)_j^2}} \quad (4)$$

where $(w_l)_j$ is the DM's weight for criterion j , and $(we_{slh})_j$ is the elicited weight for the same criterion.

For each permutation h and set s , the DMs' performance measures are aggregated by estimating their expected value:

$$\hat{E}(\text{performance}_{sh}) = \sum_{l=1}^{1000} \text{performance}_{slh} . \quad (5)$$

Each permutation h is rated according to its distance from the unpermuted ranking:

$$\text{per}_h = \frac{\sum_{r=1}^{100} 1\{\text{pos}_{hr} > r\}}{\binom{100}{2}} \quad (6)$$

where pos_{hr} is the permuted value in position r and $\binom{100}{2}$ is a normalization constant to constraint per_h in the range $[0,1]$. The rating of the unpermuted ranking is 0, while the rating of the reversed ranking is 1. The permutations are generated to achieve ratings that are uniformly distributed between 0 and 1.

The two sets of $\hat{E}(\text{performance}_{sh})$ can be compared across different per_h to gauge how different permutation ratings affect the model's performance.

3.2 Results

Figure 1 depicts the obtained $\hat{E}(\text{performance}_{sh})$ for the two methods against the permutations per_h . The expected values for the performance measure of the Linear Model are plotted with circles, while the Logistic Regression Model values are plotted with crosses.

According to Figure 1, the Linear Model outperforms the Logistic Regression Model in the non-permuted scenario, while the Logistic Regression Model is superior for sizable values of per_h . When the permutation becomes severe ($\text{per}_h \geq 0.5345$), the Linear Model is again the preferred option for weight eliciting, up to complete rank reversion.

Both the Linear Model and the Logistic Regression Model were always able to find feasible solutions.

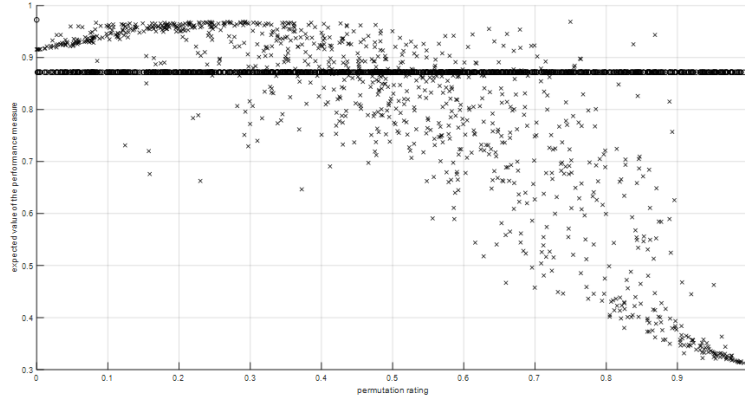


Fig. 1. Expected value of the performance measure for the two models and permutation ratings.

4 Conclusions

Unexpectedly, the Logistic Regression Model does not always outperform the Linear Model when some of the known preferences are incorrect. The advantage of the Logistic Regression Model is limited to those instances where the permuted ranking is closer to the non-permuted ranking than to the reverse ranking, with cut-point $per_h = 0.5345$. In nearly all the analysed cases, the Linear Model achieves, without incurring infeasibility issues, a constant expected value of the performance measure $per_h = 0.8716$, where chance alone would yield $per_h = 0.7370$. Leveraging the high dimensionality of the preference space, the Linear Model finds a single feasible solution and retains it for nearly all the permuted rankings, except for the reverse ranking that carries its own below-chance solution.

These results provide guidelines on when one method is preferable over the other, and prove that, when the correct method is selected, the elicited weights are close to the real ones above chance.

Further research will use the Logistic Regression Model to identify faulty known preferences, leveraging the probabilistic interpretation of the weights described in Section 2. Other machine-learning algorithms (e.g. Support Vector Machines, Neural Networks) will be specialized into weight-eliciting models. These models are expected to account for not only for incorrect known preferences but also for incorrect preference functions, discarding the hypothesis of the linear separability assumption in the preference space.

References

1. Jacquet-Lagrange, E., Siskos, J.: Assessing a set of additive utility functions for multicriteria decision-making, the UTA method. *Eur. J. Oper. Res.* 10, 151–164 (1982). [https://doi.org/10.1016/0377-2217\(82\)90155-2](https://doi.org/10.1016/0377-2217(82)90155-2).
2. Lolli, F., Ishizaka, A., Gamberini, R., Rimini, B., Balugani, E., Prandini, L.: Requalifying public buildings and utilities using a group decision support system. *J. Clean. Prod.* 164, 1081–1092 (2017). <https://doi.org/10.1016/j.jclepro.2017.07.031>.
3. Mareschal, B.: Weight stability intervals in multicriteria decision aid. *Eur. J. Oper. Res.* 33, 54–64 (1988). [https://doi.org/10.1016/0377-2217\(88\)90254-8](https://doi.org/10.1016/0377-2217(88)90254-8).
4. Riabacke, M., Danielson, M., Ekenberg, L.: State-of-the-Art Prescriptive Criteria Weight Elicitation. *Adv. Decis. Sci.* 2012, 1–24 (2012). <https://doi.org/10.1155/2012/276584>.
5. Lolli, F., Balugani, E., Ishizaka, A., Gamberini, R., Butturi, M.A., Marinello, S., Rimini, B.: On the elicitation of criteria weights in PROMETHEE-based ranking methods for a mobile application. *Expert Syst. Appl.* 120, 217–227 (2019). <https://doi.org/10.1016/j.eswa.2018.11.030>.
6. Alemi-Ardakani, M., Milani, A.S., Yannacopoulos, S., Shokouhi, G.: On the effect of subjective, objective and combinative weighting in multiple criteria decision making: A case study on impact optimization of composites. *Expert Syst. Appl.* 46, 426–438 (2016). <https://doi.org/10.1016/j.eswa.2015.11.003>.
7. de Almeida Filho, A.T., Clemente, T.R.N., Morais, D.C., de Almeida, A.T.: Preference modeling experiments with surrogate weighting procedures for the PROMETHEE method. *Eur. J. Oper. Res.* 264, 453–461 (2018). <https://doi.org/10.1016/j.ejor.2017.08.006>.

8. Figueira, J.R., Roy, B.: Determining the weights of criteria in the ELECTRE type methods with a revised Simos' procedure. *Eur. J. Oper. Res.* 139, 317–326 (2002). [https://doi.org/10.1016/S0377-2217\(01\)00370-8](https://doi.org/10.1016/S0377-2217(01)00370-8).
9. Kaliszewski, I., Podkopaev, D.: Simple additive weighting—A metamodel for multiple criteria decision analysis methods. *Expert Syst. Appl.* 54, 155–161 (2016). <https://doi.org/10.1016/j.eswa.2016.01.042>.
10. de Almeida, A.T., de Almeida, J.A., Costa, A.P.C.S., de Almeida-Filho, A.T.: A new method for elicitation of criteria weights in additive models: Flexible and interactive tradeoff. *Eur. J. Oper. Res.* 250, 179–191 (2016). <https://doi.org/10.1016/j.ejor.2015.08.058>.
11. Greco, S., Mousseau, V., Słowiński, R.: Ordinal regression revisited: Multiple criteria ranking using a set of additive value functions. *Eur. J. Oper. Res.* 191, 416–436 (2008). <https://doi.org/10.1016/j.ejor.2007.08.013>.
12. Kadziński, M., Greco, S., Słowiński, R.: Extreme ranking analysis in robust ordinal regression. *Omega*. 40, 488–501 (2012). <https://doi.org/10.1016/j.omega.2011.09.003>.
13. Greco, S., Kadziński, M., Mousseau, V., Słowiński, R.: ELECTREGKMS: Robust ordinal regression for outranking methods. *Eur. J. Oper. Res.* 214, 118–135 (2011). <https://doi.org/10.1016/j.ejor.2011.03.045>.
14. Doumpos, M., Zopounidis, C.: Preference disaggregation and statistical learning for multicriteria decision support: A review. *Eur. J. Oper. Res.* 209, 203–214 (2011). <https://doi.org/10.1016/j.ejor.2010.05.029>.
15. Corrente, S., Greco, S., Kadziński, M., Słowiński, R.: Robust ordinal regression in preference learning and ranking. *Mach. Learn.* 93, 381–422 (2013). <https://doi.org/10.1007/s10994-013-5365-4>.
16. Ishizaka, A., Lolli, F., Gamberini, R., Rimini, B., Balugani, E.: AHP-K-GDSS: A new sorting method based on AHP for group decisions. In: Bruzzone A.G. Solis A., M.M.D.F.F.L.F.F.C. (ed.) *Proceedings of the Int. Conference on Modeling and Applied Simulation*. pp. 1–5. CAL-TEK S.r.l., Barcellona (2017).
17. Ishizaka, A., Lolli, F., Balugani, E., Cavallieri, R., Gamberini, R.: DEASort: Assigning items with data envelopment analysis in ABC classes. *Int. J. Prod. Econ.* 199, 7–15 (2018). <https://doi.org/10.1016/j.ijpe.2018.02.007>.