

Robust ordinal regression in preference learning and ranking

Salvatore Corrente · Salvatore Greco ·
Miłosz Kadziński · Roman Słowiński

Received: 7 January 2012 / Accepted: 19 April 2013 / Published online: 27 July 2013
© The Author(s) 2013. This article is published with open access at Springerlink.com

Abstract Multiple Criteria Decision Aiding (MCDA) offers a diversity of approaches designed for providing the decision maker (DM) with a recommendation concerning a set of alternatives (items, actions) evaluated from multiple points of view, called criteria. This paper aims at drawing attention of the Machine Learning (ML) community upon recent advances in a representative MCDA methodology, called Robust Ordinal Regression (ROR). ROR learns by examples in order to rank a set of alternatives, thus considering a similar problem as Preference Learning (ML-PL) does. However, ROR implements the interactive preference construction paradigm, which should be perceived as a mutual learning of the model and the DM. The paper clarifies the specific interpretation of the concept of preference learning adopted in ROR and MCDA, comparing it to the usual concept of preference learning considered within ML. This comparison concerns a structure of the considered problem, types of admitted preference information, a character of the employed preference models, ways of exploiting them, and techniques to arrive at a final ranking.

Keywords Robust ordinal regression · Ranking · Preference learning · Multiple criteria decision aiding · Comparison · Preference modeling · Preference construction

Editors: Eyke Hüllermeier and Johannes Fürnkranz.

S. Corrente · S. Greco

Department of Economics and Business, University of Catania, Corso Italia, 55, 95129 Catania, Italy

S. Greco

Operations & Systems Management, University of Portsmouth, Portsmouth PO1 3DE, UK

M. Kadziński (✉) · R. Słowiński

Institute of Computing Science, Poznań University of Technology, 60-965 Poznań, Poland

e-mail: miłosz.kadzinski@cs.put.poznan.pl

R. Słowiński

Systems Research Institute, Polish Academy of Sciences, 01-447 Warsaw, Poland

1 Introduction

Ranking problems In ranking problems one aims at ordering a finite set of alternatives (items, actions) from the best to the worst, using a relative comparison approach. On the one hand, ranking problems have been of major practical interest in such various fields as economy, management, engineering, education, and environment (Zopounidis and Parados 2010). For example, countries are ranked with respect to the quality of their information technology infrastructure, or established environmental policy goals, or level of introduced innovation. World-wide cities are rated on the basis of living conditions or human capital. When it comes to firms, they are ranked on the achieved level of business efficiency or reliability of being the best contractor for a particular project. Furthermore, magazines regularly publish rankings of MBA programs, schools, or universities.

On the other hand, a growing interest in ranking problems has recently emerged in fields related to information retrieval, Internet-related applications, or bio-informatics (see, e.g. Fürnkranz and Hüllermeier 2011; Liu 2011). Indeed, ranking is at the core of document retrieval, collaborative filtering, or computation advertising. In recommender systems, a ranked list of related items should be recommended to a user who has shown interest in some other items. In computational biology, one ranks candidate structures in protein structure prediction problem, whereas in proteomics there is a need for the identification of frequent top scoring peptides.

The popularity of ranking problems encouraged researchers in different fields to propose scientific methods supporting the users in solving these problems. Remark that one has to dissociate problem formulation from problem solving because differences between the various approaches concern rather problem solving than problem formulation. The central point of these approaches concerns accounting for human preferences.

Multiple criteria decision aiding As far as Multiple Criteria Decision Aiding (MCDA) (Figueira et al. 2005) is concerned, it offers a diversity of methods designed for structuring the decision problem and carrying forward its solution. An inherent feature of decision problems handled by MCDA is a multiple criteria evaluation of alternatives (criteria = attributes with preference ordered scales). As multiple criteria are usually in conflict, the only objective information that stems from the formulation of such a decision problem is a dominance relation (Pareto relation) in the set of alternatives. This relation leaves, in general, many alternatives incomparable, and thus one needs a method that would account for preferences of the decision maker (DM), making the alternatives more comparable. Thus, the DM is the main actor of the decision aiding process in whose name or for whom the decision aiding must be given. (S)he is usually assisted by an analyst who represents a facilitator of the process and needs to perform her/his role in the interaction with the DM.

MCDA is of essential help in preference elicitation, constructing a model of user's preferences, and its subsequent exploitation which leads to a recommendation, e.g., a ranking of the considered alternatives. Thus, it is often defined as an activity of using some models which are appropriate for answering questions asked by stakeholders in a decision process (Roy 2005). As noted by Stewart (2005), MCDA includes a comprehensive process involving a rich interplay between human judgment, data analysis, and computational processes.

Dealing with DM's preferences is at the core of MCDA. However, it is exceptional that facing a new decision problem, these preferences are well structured. Hence, the questions of an analyst and the use of dedicated methods should be oriented towards shaping these preferences. In fact, MCDA proceeds by progressively forming a conviction and communicating about the foundations of these convictions. The models, procedures, and provided

results constitute a communication and reflection tool. Indeed, they allow the participants of the decision process to carry forward their process of thinking, to discover what is important for them, and to learn about their values. Such elements of responses obtained by the DMs contribute to recommending and justifying a decision, that increase the consistency between the evolution of the process and objectives and value systems of the stakeholders. Thus, decision aiding should be perceived as a constructive learning process, as opposed to machine discovery of an optimum of a pre-existing preference system or a truth that exists outside the involved stakeholders.

The above says that MCDA requires active participation of the stakeholders, which is often organized in an interactive process. In this process, phases of preference elicitation are interleaved with phases of computation of a recommended decision. Preference elicitation can be either direct or indirect. In case of direct elicitation, the DM is expected to provide parameters related to a supposed form of the preference model. However, a majority of recently developed MCDA methods (see, e.g., Ehrgott et al. 2010; Siskos and Grigoroudis 2010), require the DM to provide preference information in an indirect way, in the form of decision examples. Such decision examples may either be provided by the DM on a set of real or hypothetical alternatives, or may come from observation of DM's past decisions. Methods based on the indirect preference information are considered more user-friendly than approaches based on explicitly provided parameter values, because they require less cognitive effort from the DM at the stage of preference elicitation. Moreover, since the former methods assess compatible instances of the preference model reproducing the provided decision examples, the DM can easily see what are the exact relations between the provided preference information and the final recommendation.

Robust ordinal regression One of the recent trends in MCDA concerning the development of preference models using examples of decisions is Robust Ordinal Regression (ROR) (Greco et al. 2010). In ROR, the DM provides some judgments concerning selected alternatives in the form of pairwise comparisons or rank-related requirements, expressed either holistically or with respect to particular criteria. This is the input data for the ordinal regression that finds the whole set of value functions being able to reconstruct the judgments given as preference information by the DM. Such value functions are called compatible with the preference information. The reconstruction of the DM's judgments by the ordinal regression in terms of compatible value functions is a preference learning step. The DM's preference model resulting from this step can be used on any set of alternatives. Within the framework of ROR, the model aims to reconstruct as faithfully as possible the preference information of the DM, while the DM learns from the consequences of applying all compatible instances of the model on the set of alternatives.

A typical application scenario of the ROR method is the following. Suppose a DM wants to rank countries with respect to their trend for innovation. Three criteria are used to evaluate the countries: innovation performance, direct innovation inputs, and innovation environment. The DM is sure of some pairwise comparisons between countries which do not dominate each other. Moreover, (s)he claims that some countries should be placed in a specified part of the ranking, e.g., at the bottom 5. This preference information is a starting point for the search of a preference model, which is a value function. If there is no instance of the preference model reproducing the provided holistic judgments, the DM could either decide to work with the inconsistency or revise some pieces of preference information impeding the incompatibility. Then, using ROR, i.e. solving a series of special optimization problems, one obtains two preference relation in the set of countries, called necessary and possible. While the first is true for all value functions compatible with the DM's preference information, the

other is true for at least one such compatible value function. Moreover, ROR provides extreme ranks of particular countries and a representative compatible value function. Thanks to the analysis of the both outcomes of ROR at the current state of interaction as well as the form of the displayed representative preference model, the DM gains insights on her/his preferences. This stimulates reaction of the DM who may add a new or revise the old preference information. Such an interactive process ends when the necessary preference relation, extreme ranks, and/or a representative ranking yield a recommendation which, according to the DM, is decisive and convincing. This application example is developed in Sect. 7.

Preference learning in machine learning When it comes to Machine Learning (ML), a “learning to rank” task is also involved with learning from examples, since it takes as an input a set of items for which preferences are known (Liu 2011). Precisely, the training data is composed of lists of items with some partial order specified between items in each list. Preference Learning in ML (PL-ML) consists in discovering a model that predicts preference for a new set of items (or the input set of items considered in a different context) so that the produced ranking is “similar” in some sense to the order provided as the training data. In PL-ML, learning is traditionally achieved by minimizing an empirical estimate of an assumed loss function on rankings (Doumpos and Zopounidis 2011).

Aim of the paper While there exist many meaningful connections and analogies between ROR-MCDA and PL-ML, there is also a series of noticeable differences concerning the process of learning a decision/prediction model from data. The aim of this paper is twofold. Firstly, we wish to draw attention of the ML community upon recent advances in ROR with respect to modeling DM’s preferences and interacting with her/him in the constructive learning process. Our goal is to introduce ROR to specialists of PL in ML, so that they are aware of an alternative technique coming from a different field (MCDA), but aiming at learning preferences with respect to a similar problem. In fact, we review all previous developments in ROR, combining them under a unified decision support framework that permits to learn preferences by taking advantage of the synergy between different types of admitted preference information and provided outcomes. Secondly, we compare philosophies of preference learning adapted in ROR and ML. In this way, we follow the preliminary comparisons between MCDA and ML made by Waegeman et al. (2009) and Doumpos and Zopounidis (2011). However, we focus the attention on ROR, which is closer to PL practiced in ML than many other MCDA methods, because it exploits preference information of similar type.

The contribution of the paper is methodological, and reference to PL-ML is at a philosophical level rather than at an experimental one. Note that an empirical comparison with respect to the output of ROR and PL-ML would not be meaningful, because there is no common context of their use and no objective truth is to be attained. Moreover, each method (ROR and PL-ML) is transforming the input preference information in a different way and introduces some instrumental bias in interactive steps, thus leading to different results. Furthermore, the concept of “learning” is implemented in different ways in PL-ML and in ROR. In ROR, learning concerns not only the preference model, but also the decision maker. Since the progress in learning of the DM is non-measurable, the experimental comparison of different methods is ill-founded. Consequently, instead of providing an empirical comparison, our aim is rather to well explain and illustrate the steps of ROR.

Organization of the paper The remainder of the paper is organized as follows. In the next section, we present some basic concepts of ROR and MCDA. We also compare different aspects of ranking problems and preference learning as considered in ROR and PL-ML. This

comparison is further continued throughout the paper with respect to the input preference information, exploitation of the preferences, and evaluation of the provided recommendation. In particular, in Sect. 3, we focus on different types of preference information admitted by the family of ROR methods. At the same time, we present the approach for learning of a set of compatible value functions. Section 4 describes the spectrum of procedures for robustness and sensitivity analysis that can be employed within the framework of ROR to support arriving at the final recommendation. In Sect. 5, we put emphasis on the interactivity of the process of preference information specification. Section 6 reveals how ROR deals with inconsistency in the preference information provided by the DM. A case study illustrating the presented methodology is presented in Sect. 7. The last section concludes the paper.

2 Basic concepts in MCDA and ROR and their comparison with preference learning in ML

2.1 Problem formulation

In the multiple criteria ranking problem, alternatives are to be ranked from the best to the worst. Precisely, the ranking of alternatives from set A results from the ordering of indifference classes of A which group alternatives deemed as indifferent (Roy 2005).

Comparison: formulation of a ranking problem A ranking problem considered in MCDA corresponds most closely to an object ranking in PL-ML (see, e.g., Fürnkranz and Hüllermeier 2011; Kamishima et al. 2011), that aims at finding a model returning a ranking order among analyzed items. On the other hand, an instance ranking in PL-ML is about ordering a set of instances according to their (unknown) preference degrees (see, e.g., Waegeman and De Baets 2011). This, in turn, is equivalent to a definition of a multiple criteria sorting problem (Zopounidis and Doumpos 2002).

Let us formulate the problem that is considered in Robust Ordinal Regression (its detailed explanation is provided subsequently along with the comparative reference to PL-ML):

Given:

- a finite set of alternatives A ,
- a finite set of reference alternatives $A^R \subseteq A$,
- a finite set of pairwise comparisons for some reference alternatives, expressed either holistically or with respect to a subset of criteria,
- a finite set of intensities of preference for some pairs of reference alternatives, expressed either holistically or with respect to a subset of criteria,
- a finite set of rank-related requirements for some reference alternatives, referring either to the range of allowed ranks or comprehensive values.

Find:

- a set of additive value functions \mathcal{U}_{AR} compatible with the preference information provided by the DM, i.e. value functions for which the pre-defined misranking error is equal to zero; each value function $U \in \mathcal{U}_{AR}$ assumes as input a set of alternatives A and returns a permutation (ranking) of this set,
- necessary and possible preference relations, for pairs or quadruples of alternatives in A ,
- extreme ranks and values for all alternatives in A ,
- representative value function returning a complete ranking of the set of alternatives A ,

- minimal sets of pieces of preference information underlying potential incompatibility of preference information.

Performance measure:

- margin of the misranking error comparing the provided pieces of preference information with the target ranking.

Technique:

- loop by interaction: analyze results provided in the previous iteration and supply preference information incrementally.

2.2 Data set

Generally, decision problems considered in MCDA involve a finite set $A = \{a_1, a_2, \dots, a_i, \dots, a_n\}$ of n alternatives, i.e., objects of a decision, which are possible to be implemented or have some interest within the decision aiding process.

Comparison: size of the data set Sets of alternatives in MCDA consist of modestly-sized collections of choices. In fact, multiple criteria ranking problems considered in Operations Research and Management Science (OR/MS) usually involve several dozens of alternatives (Wallenius et al. 2008). Consequently, MCDA methods, including ROR, are designed to deal with the relatively small data sets, and the development of the computationally efficient algorithms that scale up well with the number of alternatives is not at the core of these approaches. On the other hand, real world applications of PL-ML often involve massive data sets (Waegeman et al. 2009). Typical ML applications are related to the Internet, in particular to recommender systems and information retrieval, and, thus, the scalability of the algorithms is a fundamental issue. Nevertheless, it is worth noting the complementarity of MCDA and ML from the viewpoint of the data size: the previous is more appropriate for several dozens of alternatives and over hundreds, the latter is more suitable.

Let us additionally note that in case of large data sets, it is very rare that the whole ranking is analyzed by the user. In fact, the rank of at most several dozens of alternatives is of interest to the DM, while the rest of the alternatives is neglected and, in fact, can remain unordered. Thus, to overcome the curse of dimensionality, in case of over hundreds of alternatives it might be useful first to filter out some significant subsets of alternatives, consisting of these less relevant to the DM. For this purpose, one may take advantage of some simple classification methods, or elimination by dominance (with the benchmarks in form of some artificial alternatives whose attractiveness should be evaluated directly by the DM). This would lead to limiting down the data set to several dozens relevant alternatives, which can be directly handled by ROR.

2.3 Data description

In MCDA, an important step concerns selection or construction of attributes describing the alternatives, based on the set of their elementary features. The aim is to obtain a consistent family of m attributes $G = \{g_1, g_2, \dots, g_j, \dots, g_m\}$ which permits a meaningful evaluation and comparison of alternatives. Such attributes represent goals to be attained, objectives, impacts of considered alternatives, and points of view that need to be taken into account. In MCDA, one assumes that some values of attributes are more preferred than others, and thus, the corresponding value sets are transcoded into real-valued monotonic preference scales.

Such attributes are called criteria. Let us denote by $X_j = \{x_j \in \mathbb{R} : g_j(a_i) = x_j, a_i \in A\}$ the set of all different evaluations on criterion $g_j, j \in J = \{1, 2, \dots, m\}$. We will assume, without loss of generality, that the greater $g_j(a_i)$, the better alternative a_i on criterion g_j , for all $j \in J$.

Comparison: monotonicity of attributes/criteria In ROR (or, more generally, MCDA) one constructs criteria with explicit monotonic preference scales, whereas in ML the relationships between value sets of attributes and DM’s preferences (if any) are discovered from data for a direct use in classification or ranking. This means that in the majority of ML methods (e.g., approaches proposed in Chu and Ghahramani (2005a) and Herbrich et al. (1999), which solve the problem of ordinal regression), the monotonic preference scales converting attributes to criteria are neither used nor revealed explicitly.

Nevertheless, in the recent years, learning of predictive models that guarantee the monotonicity in the input variables has received increasing attention in ML (see, e.g., Feelders 2010; Tehrani et al. 2012a). In fact, the difficulty of ensuring such monotonicity increases with the flexibility or nonlinearity of a model. In PL-ML, it is obtained either by a modification of learning algorithms or a modification of the training data.

Apart from the monotonicity of criteria, the family G considered in MCDA is supposed to satisfy another two conditions: completeness (all relevant criteria are considered) and non-redundancy (no superfluous criteria are taken into account). The set of attributes considered in ML, in general, does not have to satisfy such strict requirements.

Comparison: complexity of the considered alternatives/items As noted by Waegeman et al. (2009), ML is also concerned with more complex structures than alternatives described over a number of criteria. This includes, e.g., graphs in the prediction problems in bio-informatics or texts and images in the retrieval problems. Nevertheless, it is worth to note the recent effort of the MCDA community to apply decision aiding methods to geographical (spatial) data (Malczewski 2010).

2.4 Preference/prediction model

The most straightforward way of ranking the alternatives in MCDA consists in aggregating their individual performances on multiple criteria into a comprehensive (overall) performance. In particular, Multiple Attribute Utility Theory (MAUT) models the decision making situation with an overall value function U (Keeney and Raiffa 1976), and assigns a numerical score to each alternative. Such a score serves as an index used to decide the rank in a complete preorder. In ROR, in order to model the DM’s preference information, we use the additive value function:

$$U(a) = \sum_{j=1}^m u_j(g_j(a)) \tag{1}$$

where the marginal value functions $u_j, j \in J$, are monotone, non-decreasing and normalized so that the additive value (1) is bounded within the interval $[0, 1]$. Note that for simplicity of notation, one can write $u_j(a), j \in J$, instead of $u_j(g_j(a))$. Consequently, the basic set of constraints defining general additive value functions has the following form:

$$\left. \begin{aligned} u_j(x_j^k) - u_j(x_j^{(k-1)}) &\geq 0, \quad k = 2, \dots, n_j(A), \\ u_j(x_j^1) &= 0, \quad \sum_{j=1}^m u_j(x_j^{n_j(A)}) = 1, \end{aligned} \right\} E_{BASE}^{AR}$$

where $x_j^1, x_j^2, \dots, x_j^{n_j(A)}$ are the ordered values of X_j , $x_j^k < x_j^{k+1}$, $k = 1, 2, \dots, n_j(A) - 1$ ($n_j(A) = |X_j|$ and $n_j(A) \leq n$). General monotonic marginal value functions defined in this way do not involve any arbitrary or restrictive parametrization. On the contrary, the majority of existing methods employ marginal value functions which are linear or piecewise linear (Siskos et al. 2005). Note that piecewise linear functions require specification of the number of characteristic points which is not easy for most DMs.

Comparison: interpretability and regularization of the model The preference model in the form of an additive value function is appreciated by the MCDA community for both an easy interpretation of numerical scores of alternatives and a possibility of assessing relative importance of each evaluation on a particular criterion understood as its share in the comprehensive value. Indeed, the interpretability and descriptive character of preference models is essential in MCDA, since it encourages the participation of the DM in the decision process.

On the contrary, ML has mainly focused on the development of non-linear models, such as support vector machines (SVMs) (see, e.g., Herbrich et al. 2000; Joachims 2002) or neural networks. The higher predicting ability and possibility of capturing complex interdependencies by such models results, however, in less confidence in their employment by the users who need to interpret and understand the underlying process (Waegeman et al. 2009).

Note that in a regression learning problem in ML, the task of finding the utility function corresponds just to learning an appropriate mapping from data to real numbers. Thus, the utility function is used in a rather instrumental way, and the model is not deeply analyzed by the user in order to gain insights on the character of the alternatives.

The researchers in ML indicate also the need for the regularization, which takes into account the trade-offs between complexity and performance of the model, preventing overfitting on the data (Waegeman et al. 2009). In MCDA the focus is put on the explicative character of the employed models, rather than statistically predictive PL. Thus, MCDA models do not involve regularization, being, however, vulnerable to noise.

2.5 Input data

A preference elicitation process in MCDA consists in an interaction between the DM and the analyst, and leads the DM to express information on her/his preferences within the framework of the assumed preference model. Such information is materialized by a set of plausible values of the parameters related to the formulation of the model. At the end of the decision aiding process, the use of a preference model for the inferred parameters should lead to a result which is compatible with the DM's preferential system.

In case of an additive value function, some MCDA methods require the DM to provide constraints on the range of weights of linear marginal value functions, or on the range of variation of piecewise linear marginal value functions. The DM may have, however, difficulties to analyze the link between a specific value function and the resulting ranking. Thus, ROR implements the preference disaggregation analysis, which is a general methodological framework for the development of a decision model using examples of decision made by the DM. In fact, ROR admits and enhances the variety of indirect preference information concerning the set of reference alternatives $A^R = \{a^*, b^*, \dots\}$ (usually, $A^R \subseteq A$).

This information may have the form of pairwise comparisons of reference alternatives stating the truth or falsity of the weak preference relation (Greco et al. 2008). Such a comparison may be related to the holistic evaluation of alternatives on all considered criteria or on a subset of criteria considered in a hierarchical structure (Corrente et al. 2012). In the same spirit, the DM may provide holistic or partial comparisons of intensities of preference

between some pairs of reference alternatives (Figueira et al. 2009). Furthermore, (s)he is admitted to refer to the range of allowed ranks that a particular alternative should attain, or to constraints on the final value scores of the alternatives. In this way, the DM may rate a given alternative individually, at the same time collating it with all the remaining alternatives (Kadziński et al. 2013a). Finally, ROR accounts for the preference information regarding interactions between n -tuples (e.g., pairs or triples) of criteria (see, e.g., Greco et al. 2013).

ROR methods are intended to be used interactively, with an increasing subset of reference alternatives and a progressive statement of holistic judgments. The DM may assign gradual confidence levels to pieces of preference information provided in the subsequent iterations (Greco et al. 2008; Kadziński et al. 2013a).

The paradigm of learning from examples which is implemented in ML is a counterpart approach to preference modeling and decision aiding (Fürnkranz and Hüllermeier 2010). Traditional machine learning of preferences was motivated by the applications where decision examples come from observation of people's behavior rather than from direct questioning. For example, typical ML applications are related to the Internet, in particular to recommender systems and information retrieval. In the previous, the task is to recommend to the user a new item (like movie or book) that fits her/his preferences. The recommendation is computed on the base of the learning information describing the past behavior of the user. In the latter, the task is to sort (or rank) the documents retrieved by the search engine according to the user's preferences. Nevertheless, recently in PL-ML one also develops the procedures which require direct human assessment (Liu 2011). For example, in information retrieval, one uses pooling methods whose role is to collect documents that are more likely to be relevant. Considering the query, human annotators specify whether a document is relevant or not, or whether a document is more relevant than the other, or they provide the total order of the documents.

Comparison: inferring a faithful model Both ROR and PL-ML aim at inferring a (preference or prediction) model that permits to work out a final recommendation (e.g., a ranking of alternatives) being concordant with a value system of the DM; thus, both of them deal with the same decision problem.

Comparison: learning of a model from examples Both ROR and PL-ML try to build a DM's preference model from decision examples (exemplary judgments) provided by the DM—in ML decision examples form a training data set, while in ROR, preference information; thus, both of them comply with the paradigm of learning by example. In both approaches learning concerns the model, because one integrates into it the expressed/collected preferences.

Comparison: amount of the available human preferences The number of decision examples forming the training set or the preference information is quite different in ML and ROR: while the first is big enough for statistical learning, the second is usually limited to small sets of items, which excludes statistical analysis.

Comparison: preference elicitation for the utility functions Preference elicitation can be perceived as one of the main links between decision analysis and PL-ML. An extensive survey on the utility elicitation techniques is presented in Braziunas and Boutilier (2008). The authors classify the elicitation methods in different ways. For example, they distinguish the local and global queries. The previous involve querying only single attributes or small subsets of attributes, whereas the latter concern the comparison of the complete outcomes over all attributes. ROR takes advantage of both types of techniques.

As for the representation of uncertainty over user preferences, two main approaches have been proposed. On the one hand, in a Bayesian approach, uncertainty over utilities is quantified probabilistically (see, e.g., Braziunas and Boutilier 2005). On the other hand, in a feasible set uncertainty, the space of feasible utilities is defined by some constraints on the parameters of the user's utility function (this approach is also called Imprecisely Specified Multi-Attribute Utility Theory (ISMAUT) (see, e.g., White et al. 1983). ROR represents the latter approach.

2.6 Performance measure

Disaggregation analysis in MCDA aims at constructing a preference model that represents the actual decisions made by the DM as consistently as possible. To measure the performance of this process, ROR considers a margin of the misranking on a deterministic preference information. When the margin is not greater than zero, this means that for a given preference information, no compatible value function exists. This is considered as a learning failure. Subsequently, the DM could either decide to work with the inconsistency (agreeing that some of the provided pieces would not be reproduced by the inferred model) or remove/revise some pieces of preference information impeding the incompatibility. In fact, in ROR the application of the learned preference model on the considered set of alternatives makes greater sense when there is no inconsistency, and misranking error is reduced to zero (i.e. the margin of the misranking error is greater than zero).

On the other hand, in ML the predictive performance of a ranker is measured by a loss function. Indeed, any distance or correlation measure on rankings can be used for that purpose (Fürnkranz and Hüllermeier 2010). When it comes to the inconsistencies in training data, PL-ML treats it as noise and hard cases that are difficult to learn.

Comparison: minimization of a learning error Both ROR and ML try to represent preference information with a minimal error—ML considers a loss function, and ROR a misranking error. In this way, both of them measure the distance between the DM's preferences and the recommendation which can be obtained for the assumed model. However, the loss function considered in ML is a statistical measure of the performance of preference learning on an unknown probabilistic distribution of preference information, whereas a margin of the misranking error is a non-statistical measure.

Obviously, it is possible to optimize the error considered in ROR within a standard ML setting. Nevertheless, such an optimization is not conducted, because it would lead to select a single instance of the preference model. Instead, our aim is to work with all instances of the preference model for which the value of the margin of the misranking error is greater than the acceptable minimal threshold (i.e., zero in case the inconsistency is not tolerated). That is why, in ROR we analyze the mathematical constraints on the parameters of the constructed preference model to which the preference information of the DM was converted. Precisely, the preference relations in the whole set of alternatives result from solving some mathematical programming problems with the above constraints.

Comparison: dealing with inconsistencies ROR treats inconsistencies explicitly during construction of the preference model. In this case, the DM may either intentionally want to pursue the analysis with the incompatibility or (s)he may wish to identify its reasons with the use of some dedicated procedures, which indicate the minimal subsets of troublesome pieces of preference information. As emphasized in Sect. 6, analysis of such subsets is informative for the DM and it permits her/him to understand the conflicting aspects of her/his statements.

PL-ML methods process noise in the training data in a statistical way. In fact, the ML community perceives the noise-free applications as uncommon in practice and the way of tolerating the inconsistencies is at the core of ML methods.

Let us additionally note that the noise-free ROR is well adapted to handling preferences of a single DM. There are several studies (e.g. Pirlot et al. 2010) proving great flexibility of general value functions in representing preference information of the DM. Consequently, if preferences of the DM do not violate dominance and are not contradictory, it is very likely that they could be reproduced by the model used in ROR. Nevertheless, ROR has been also adapted to group decision (Greco et al. 2012; Kadziński et al. 2013b).

2.7 Ranking results

In case of MCDA methods using a preference model in form of a value function, traditionally, only one specific compatible function or a small subset of these functions has been used to determine the final ranking (see, e.g., Beuthe and Scannella 2001; Jacquet-Lagrèze and Siskos 1982). To avoid such arbitrary limitation of the instances underlying the provided recommendation and to prevent the DMs from an embarrassing selection of some compatible instances, which requires interpretation of their form, ROR (Greco et al. 2010) postulates taking into account the whole set of value functions compatible with the provided indirect preference information. Then, the recommendation for the set of alternatives is worked out with respect to all these compatible instances.

When considering the set of compatible value functions, the rankings which can be obtained for them can vary substantially. To examine how different can those rankings be, ROR conducts diverse sensitivity and robustness analysis. In particular, one considers two weak preference relations, necessary and possible (Greco et al. 2008). Whether for an ordered pair of alternatives there is necessary or possible preference depends on the truth of preference relation for all or at least one compatible value function, respectively. Obviously, one could reason in terms of the necessary and the possible, taking into account only a subset of criteria or a hierarchical structure of the set of criteria (Corrente et al. 2012). In this case, the presented results are appropriately adapted to reflect the specificity of a particular decision making situation.

However, looking at the final ranking, the DM is usually interested in the position which is taken by a given alternative. Thus, one determines the best and the worst attained ranks for each alternative (Kadziński et al. 2012a). In this way, one is able to assess its position in an overall ranking, and not only in terms of pairwise comparisons. Finally, to extend original ROR methods in their capacity of explaining the necessary, possible and extreme results, one can select a representative value function (Kadziński et al. 2012b). Such a function is expected to produce a robust recommendation with respect to the non-univocal preference model stemming from the input preference information. Precisely, the representative preference model is built on the outcomes of ROR. It emphasizes the advantage of some alternatives over the others when all compatible value functions acknowledge this advantage, and reduces the ambiguity in the statement of such an advantage, otherwise.

Comparison: considering the plurality of compatible models In traditional ML rank loss minimization leads to a choice of a single instance of the predictive model. This corresponds to the traditional UTA-like procedures in MCDA, which select a “mean”, “central”, “the most discriminant”, or “optimal” value function (see Beuthe and Scannella 2001; Despotis et al. 1990; Siskos et al. 2005).

Nevertheless, in PL-ML one can also indicate some approaches that account for the plurality of compatible instances. For example, Yaman et al. (2011) present a learning algorithm that explicitly maintains the version space, i.e., the attribute-orders compatible with all pairwise preferences seen so far. However, since enumerating all Lexicographic Preference Models (LPMs) consistent with a set of observations can be exponential in the number of attributes, they rather sample the set of consistent LPMs. Predictions are derived based on the votes of the consistent models; precisely, given two objects, the one supported by the majority of votes is preferred. When comparing with a greedy algorithm that produce the “best guess” lexicographic preference model, the voting algorithm proves to be better when the data is noise-free.

Furthermore, Viappiani and Boutilier (2010) present a Bayesian approach to adaptive utility selection. The system’s uncertainty is reflected in a distribution, or beliefs, over the space of possible utility functions. This distribution is conditioned by information acquired from the user, i.e. (s)he is asked questions about her preference and the answers to these queries result in updated beliefs. Thus, when an inference is conducted, the system takes into account multiple models, which are, however, weighted by their degree of compatibility.

Comparison: generalization to the new users/stakeholders The essence of the decision aiding process is to help a particular DM in making decision with respect to her/his preferences. In fact, ROR is precisely addressed to a given user and the generalization to other users is neither performed nor even desired. On the contrary, in some applications of PL-ML, the aim is to infer the preferential system of a new user on the basis of preferences of other users.

2.8 Interaction

ROR does not consider the DM’s preference model as a pre-existing entity that needs to be discovered. Instead, it assumes that the preference model has to be built in course of an interaction between the DM and the method that translates preference information provided by the DM into preference relations in the set of alternatives. ROR encourages the DM to reflect on some exemplary judgments concerning reference alternatives. Initially, these judgments are only partially defined in the DM’s mind. The preferences concerning the alternatives are not simply revealed nor they follow any algorithm coming from the DM’s memory. Instead the DM constructs her/his judgments on the spot when needed. This is concordant with both the constructivist approach postulated in MCDA (Roy 2010b) and the principle of posterior rationality postulated by March (1978). The provided preference information contributes to the definition of the preference model. In ROR, the preference model imposes the preference relations in the whole set of alternatives.

In this way, ROR emphasizes the discovery of DM’s intentions as an interpretation of actions rather than as a priori position. Analyzing the obtained preference relations, the DM can judge whether the suggested recommendation is convincing and decisive enough, and whether (s)he is satisfied with the correspondence between the output of the preference model and the preferences that (s)he has at the moment. If so, the interactive process stops. Otherwise, the DM should pursue the exchange of preference information. In particular, (s)he may enrich the preference information by providing additional exemplary judgments. Alternatively, if (s)he changed her/his mind or discovered that the expressed judgments were inconsistent with some previous judgments that (s)he considers more important, (s)he may backtrack to one of the previous iterations and continue from this point. In this way, the process of preference construction is either continued or restarted.

The use of the preference model shapes the DM's preferences and makes her/his convictions evolve. As noted by Roy (2010b), the co-constructed model serves as a tool for looking more thoroughly into the subject, by exploring, reasoning, interpreting, debating, testing scenarios, or even arguing. The DM is forced to confront her/his value system with the results of applying the inferred model on the set of alternatives. This confrontation leads the DM to gaining insights on her/his preferences, providing reactions in the subsequent iteration, as well as to better understanding of the employed method. In a way, ROR provokes the DM to make some partial decisions that lead to a final recommendation. The method presents its results so that to invite the DM to an interaction—indeed, comparing the necessary and possible relations, the DM is encouraged to supply preference information that is missing in the necessary relation. Let us emphasize that the knowledge produced during the constructive learning process does not aim to help the DM discover a good approximate decision that would objectively be one of the best given by her/his value system. In fact, the “true” numerical values of the preference model parameters do not exist, and thus it is not possible to refer directly to the estimation paradigm. Instead, the DM is provided with a set of results derived from applying some working assumptions and different reasoning modes, and it is the course of the interactive procedure that enhances the trust of the DM in the final recommendation.

Comparison: preference construction in ROR vs. preference discovery in typical ML In fact, ROR can be qualified as *preference construction* method, whereas the typical methods of ML can be named as *preference discovery*. The main differences between the two approaches are the following:

- preference construction is subjective while preference discovery is objective: within preference construction, the same results can be obtained (i.e., accepted, rejected, doubted, etc.) in a different way by different DMs, while within preference discovery, there is no space for taking into account the reactions of the DM;
- preference construction is interactive while preference discovery is automatic: the DM actively participates in the process of preference construction, while in the preference discovery the DM is asked to give only some preference information that are transformed in a final result by the adopted methodology without any further intervention of the DM;
- preference construction provides recommendations while preference discovery gives predictions: the results of a preference construction give to the DM some arguments for making a decision, while the results of preference discovery give a prevision of what will be some decisions;
- preference construction is DM-oriented while preference discovery is model-oriented: preference construction aims the DM learns something about her/his preferences, while preference discovery assumes the model would learn something about the preferences of the DM.

In the view of above remarks, let us emphasize that preference learning in MCDA does not only mean statistical learning of preference patterns, i.e. discovery of statistically validated preference patterns. MCDA proposes a constructivist perspective of preference learning in which the DM takes part actively.

Comparison: recommendation vs. elicitation The main objective of the majority of PL-ML methods is to exploit current preferences and to assess a preference model applicable on the set of alternatives in a way that guarantees the satisfying concordance between the discovered/predicted results and the observed preferences. Thus, PL-ML is focused rather

on working out a recommendation, being less sensitive to gaining explanation of the results. On the other hand, the role of preference elicitation within ROR is to acquire enough knowledge and arguments for explanation of the decision. This enables the DM to establish the preferences that previously had not pre-existed in her/his mind, to accept the recommendation, and to appropriately use it (and possibly share with the others). ROR (or, generally, “preference construction”) is a mutual learning of the model and the DM to assess and understand preferences on the considered set of alternatives.

Nevertheless, this issue has been also considered in PL-ML especially with respect to the setting of recommender systems and product search. In fact, there are several works that advocate for eliciting user preferences, suggesting an incremental and interactive user system. Although the context of their use and the feedback presented to the user is significantly different than in ROR, they can be classified as “preference construction” as well.

For example, Pu and Chen (2008) formulate a set of interaction design guidelines which help users state complete and sound preferences with recommended examples. They also describe strategies to help users resolve conflicting preferences and perform trade-off decisions. These techniques allow gaining a better understanding of the available options and the recommended products through explanation interfaces.

Moreover, there are also noticeable advances in the active learning, which deals with the algorithms that are able to interactively query the user to obtain the desired outputs for new items (Settles 2012). In particular, in Viappiani and Boutilier (2010) one presents Bayesian approaches to utility elicitation that involve an interactive preference elicitation in order to learn to make optimal recommendation to the user. The system is equipped with an active learning component which is employed to find relevant preference queries that are asked to the user. Similar task has been considered, e.g., in Radlinski and Joachims (2007) and Tian and Lease (2011). These studies report that active exploration strategy substantially outperforms passive observation and random exploration, and quickly leads to a presentation of the improved ranking to a user.

Comparison: user interface vs. truly interactive process Obviously, an interaction with the user in the PL-ML methods that have not considered this aspect can be modeled with the use of an interface accounting for user preferences. However, in our understanding a truly interactive process requires exploitation of the learned models, delivering to the user consequences of such exploitation, and encouraging her/him to the further involvement. Note that the feedback that can be provided by the majority of PL-ML methods is formed by the results predicted by the single optimized model. The user can then react by reinforcing or neglecting some parts of the outcome. Nevertheless, such a feedback is rather poor, because it does not guide the user through the process. Moreover, since PL-ML setting admits noise, there is no guarantee that the provided preferences will be integrated into the model in the subsequent iteration. On the other hand, noise-free ROR reveals the possible, necessary, extreme, and representative consequences of the preference information provided at the current stage of interaction. In this way, it leads the DM to have a better understanding of her/his preferences and invites her/him to a deeper exploitation of the preference model.

Comparison: analyst vs. user interface Let us also note that real-world MCDA problems involve an analyst, who interacts with the DM in order to guide the process. Once the analyst enters into the interaction with the DM, (s)he becomes a co-constructor of the knowledge produced; thus, (s)he cannot be perceived as being outside the decision aiding process. It is difficult to imagine that a software interface can play the same role.

Comparison: validation of the results In MCDA it is assumed that the analyst cooperates with the DM and the quality of the model is validated by an interactive process during which (s)he judges the correspondence between the output of the preference model and one's preferential system. On the other hand, the generalizing ability of the model is the core issue in the validation stage of all statistical learning models. In PL-ML the validation stage involves, e.g., additional testing sets or the use of resampling methods.

2.9 Features of preference learning in robust ordinal regression

Belton et al. (2008) consider some important features of a preference learning process within MCDA. Let us recall them, pointing out how they are extant in ROR:

- *flexibility* of the interactive procedure, which is related to the capacity of incorporating any preference information coming from the DM: ROR admits the DM to provide a wide spectrum of indirect preference information; note that flexibility can be decomposed in generality, reversibility and zooming capacity;
- *generality* of the preference model, which is related to the universality and the plurality of the decision model: for the features of plurality of instances and universality of value functions in ROR, there is a large credit for generality;
- *universality* of the preference model, which considers the non-specificity of the form of the value function, in the sense that the less specific the form, the greater the chance that the model learns in a sequence of iterations: the additive value function with monotone marginal value functions considered within ROR constitutes a very general preference model and consequently it reaches a very good level of universality, which is far more universal than the model admitting only linear marginal value functions;
- *plurality of instances of value functions* which regards consideration of only one, several, or even all compatible instances of the considered preference model: ROR takes into account the whole set of additive value functions compatible with the preference information provided by the DM, which is evidently more plural than considering only one value function as in the traditional approaches;
- *reversibility* which is understood as the possibility for the DM to return to a previous iteration of interaction with the method: ROR permits in any moment to retract, to modify or to remove the already expressed pieces of preference information;
- *zooming* which regards the possibility to represent preferences in a limited zone of the evaluation spaces of considered criteria: ROR enables to add preference information relative to alternatives from a particular region of the evaluation space of considered criteria, which results in a more precise representation of preferences in this local region.

3 Preference information

In this section, we present a variety of preference information admitted by the family of ROR methods designed for dealing with multiple criteria ranking problems. The wide spectrum of the accounted types of preference information guarantees the flexibility of the interactive procedure. We discuss the usefulness of each accounted type of preference information, and we present mathematical models which are able to reproduce preferences of the DM, i.e. translate the exemplary decisions of the DM into parameters of the value function. The constraints related to every new piece of preference information can reduce the feasible polyhedron of all compatible value functions. Let us denote the holistic set of constraints obtained in this way by E^{A^R} and the corresponding set of value functions compatible with

the provided preference information by \mathcal{U}_{A^R} . Although the considered set of additive value functions composed of marginal monotone value functions already ensures a large credit for generality, plurality, and universality, we extend the basic model by accounting for interactions between criteria.

Comparison: variety of preference information admitted by the method ROR admits the variety of indirect preference information concerning the set of reference alternatives. This means that the DM can use pairwise comparisons or rank-related requirements or magnitudes of preference if (s)he feels comfortable with this kind of information and (s)he is able to provide it for the problem at hand. However, (s)he is not obliged to specify preference information of each type. On the contrary, ML methods consider rather preference information of a given type and the user has to express her/his preferences in the precise form required by the employed approach (e.g., solely pairwise comparisons or only the magnitudes of preference).

3.1 Pairwise comparisons

Comparing alternatives in a pairwise fashion, which is admitted by numerous decision-theoretic methods, is consistent with intuitive reasoning of DMs, and requires from the DM relatively small cognitive effort (Fürnkranz and Hüllermeier 2010). In the UTA^{GMS} method (Greco et al. 2008), which initiated the stream of further developments in ROR, the ranking of reference alternatives does not need to be complete as in the original UTA method (Jacquet-Lagrèze and Siskos 1982). Instead, the DM may provide pairwise comparisons just for those reference alternatives (s)he really wants to compare. Precisely, the DM is expected to provide a partial preorder \succsim on A^R such that, for $a^*, b^* \in A^R$, $a^* \succsim b^*$ means a^* is at least as good as b^* .

Obviously, one may also refer to the relations of strict preference $>$ or indifference \sim , which are defined as, respectively, the asymmetric and symmetric part of \succsim . The transition from a reference preorder to a value function is done in the following way: for $a^*, b^* \in A^R$,

$$\left. \begin{array}{l} U(a^*) \geq U(b^*) + \varepsilon, \quad \text{if } a^* > b^*, \\ U(a^*) = U(b^*), \quad \text{if } a^* \sim b^*, \end{array} \right\} E_{GMS}^{A^R}$$

where ε is a (generally small) positive value.

Comparison: use of pairwise comparisons in PL-ML Learning by pairwise comparisons paradigm is a ML counterpart of MCDA. Thus, the rankers in ML require exemplary pairwise preferences of the form $a^* > b^*$ suggesting that a^* should be ranked higher than b^* . Then, the algorithm needs to care about preserving the relative order between the compared pairs of items. Such decomposition of the original problem into a set of presumably simpler subproblems is not only advantageous for human decision making, but it is also useful from a ML point of view (Fürnkranz and Hüllermeier 2010). It is the case, because the resulting learning problem (i.e. operating on every two items under investigation and minimizing the number of misranked pairs) can typically be solved in a more accurate and efficient way. In PL-ML, one has proposed several algorithms, whose major differences are in the loss function. Let us mention a few of them.

Fürnkranz and Hüllermeier (2010) discuss various standard loss functions on ranking that can be minimized in expectation. In particular, they consider the expected Spearman rank correlation between the true and the predicted ranking, or the number of pairwise inversions,

i.e., Kendall’s tau, which is traditionally considered in the UTA-like methods. Furthermore, RankBoost (Freund et al. 2003) formalizes learning to rank as a problem of binary classification on instance pairs. By adopting the boosting approach, it trains one weak ranker at each iteration. After each round, the item pairs are re-weighted in order to relatively increase the weight of wrongly ranked pairs. In fact, RankBoost minimizes the exponential loss. When it comes to the model of Ranking SVM (Joachims 2002), it uses representation of the examples as points in space, which carry the rank information. These labeled points are used to find a boundary that specifies the order of the considered points. Precisely, Ranking SVM employs the loss function in the form of a hinge loss defined on item pairs. Finally, instead of explicitly defining the loss function, LambdaRank (Burges et al. 2006) directly defines the gradient. The authors note that it is easier to specify rules determining how to change the rank order of documents than to construct a smooth loss function. Some improvements of the “learning to rank” algorithms that enable better ranking performance consist in emphasizing likely top-ranked items and balancing the distribution of items pairs across queries (Liu 2011).

3.2 Intensities of preference

In some decision making situations the DMs are willing to provide more information than a partial preorder on a set of reference alternatives, such as “ a^* is preferred to b^* at least as much as c^* is preferred to d^* ”. The information related to the intensity of preference is accounted by the GRIP method (Figueira et al. 2009). It may refer to the comprehensive comparison of pairs of reference alternatives on all criteria or to a particular criterion only. Precisely, in the previous case, the DM may provide a partial preorder \succsim^* on $A^R \times A^R$, whose meaning is: for $a^*, b^*, c^*, d^* \in A^R$,

$$(a^*, b^*) \succsim^* (c^*, d^*) \Leftrightarrow \begin{matrix} a^* \text{ is preferred to } b^* \text{ at least as much as} \\ c^* \text{ is preferred to } d^*. \end{matrix}$$

When referring to a particular criterion $g_j, j \in J$, rather than to all criteria jointly, the meaning of the expected partial preorder \succsim_j^* on $A^R \times A^R$ is the following: for $a^*, b^*, c^*, d^* \in A^R$,

$$(a^*, b^*) \succsim_j^* (c^*, d^*) \Leftrightarrow \begin{matrix} a^* \text{ is preferred to } b^* \text{ at least as much as} \\ c^* \text{ is preferred to } d^* \text{ on criterion } g_j. \end{matrix}$$

In both cases, the DM is allowed to refer to the strict preference and indifference relations rather than to weak preference only. The transition from the partial preorder expressing intensity of preference to a value function is the following: for $a^*, b^*, c^*, d^* \in A^R$,

$$\left. \begin{matrix} U(a^*) - U(b^*) \geq U(c^*) - U(d^*) + \varepsilon, & \text{if } (a^*, b^*) \succ^* (c^*, d^*), \\ U(a^*) - U(b^*) = U(c^*) - U(d^*), & \text{if } (a^*, b^*) \sim^* (c^*, d^*), \\ u_j(a^*) - u_j(b^*) \geq u_j(c^*) - u_j(d^*) + \varepsilon, \\ \text{if } (a^*, b^*) \succ_j^* (c^*, d^*) \text{ for } g_j \in G, \\ u_j(a^*) - u_j(b^*) = u_j(c^*) - u_j(d^*), \\ \text{if } (a^*, b^*) \sim_j^* (c^*, d^*) \text{ for } g_j \in G, \end{matrix} \right\} E_{GRIP}^R$$

where \succ^* and \sim^* are defined, respectively, as the asymmetric and symmetric part of \succsim^* .

Comparison: use of intensities/magnitudes of preference in PL-ML When using the preference information in the form of pairwise comparisons, one loses the granularity in the relevance judgments (any two items with different relevance/attractiveness degrees can construct an item pair). Thus, several algorithms have been proposed in PL-ML to tackle the problem of considering the magnitude of preference. In particular, Qin et al. (2007) suggested the use of leveraged multiple hyperplanes to preserve the magnitude of rating differences on the basis of the Ranking SVM algorithm and demonstrated the importance of preference magnitude. Further, Cortes et al. (2007) analyzed the stability bounds of magnitude preserving loss functions for generalization error. They proposed two magnitude-preserving ranking algorithms, MPRank and SVRank, with reports of the improvement on mis-ordering loss. Finally, in the MPBoost method (Zhu et al. 2009) one applies the preference magnitude into the exponential loss function of boosting to improve the accuracy of ranking.

Note that introducing a quaternary relation \succsim^* , ROR considers the intensity of preference in an ordinal way. On the contrary, in PL-ML it is supposed that there exists a cardinal rating such that the magnitude of preference is a monotonic function of difference between the cardinal ratings. Moreover, in ROR the intensity of preference can be referred also to a single criterion, while this option has not been yet considered in PL-ML.

3.3 Rank-related requirements

When looking at the final ranking, the DM is mainly interested in the position which is attained by a given alternative and, possibly, in its comprehensive score. Therefore, in the RUTA method (Kadziński et al. 2013a), the kind of preference information that may be supplied by the DM have been extended by information referring to the desired rank of reference alternatives, i.e. final positions and/or scores of these alternatives. In fact, when employing preference disaggregation methods in the context of sorting problems in MCDA (instance ranking in PL-ML), the DM is allowed to refer to the desired final assignment (label) of the reference alternatives. In this perspective, it was even more justified to adapt similar idea to multiple criteria ranking problems, and to allow DMs expressing their preferences in terms of the desired ranks of reference alternatives.

In fact, people are used to refer to the desired ranks of the alternatives in their judgments. In many real-world decision situations (e.g., evaluation of candidates for some position) they use statements such as a^* should be among the 5 % of best/worst alternatives, or b^* should be ranked in the second ten of alternatives, or c^* is predisposed to secure the place between 4 and 10. These statements refer to the range of allowed ranks that a particular alternative should attain. When using such expressions, people do not confront “one vs one” as in pairwise comparisons or “pair vs pair” as in statements concerning intensity of preference, but rather rate a given alternative individually, at the same time somehow collating it with all the remaining alternatives jointly.

Moreover, specification of the desired ranks of the alternatives addresses one of the commonly encountered disadvantages of using traditional UTA-like methods. On the one hand, very often all reference alternatives or their significant subsets are grouped together in the upper, or middle, or lower part of the ranking. As a result, the positions of reference alternatives in the final ranking are very close to each other. On the other hand, some UTA-like procedures (Beuthe and Scannella 2001) allow discrimination of the comprehensive values of reference alternatives. This usually results in the uniform distribution of their positions in the final ranking, e.g., one alternative is placed at the very top, the other just in the middle, and the third one at the very bottom. If the rankings obtained in the two above described scenarios are inconsistent with the DM’s expectations, accounting for the rank-related requirements may be used as a tool to prevent such situations. It is the case, since the ranks

specified by the DM may be interpreted in terms of the desired “parts” of the final ranking in which reference alternatives should be placed.

Finally, there is a correspondence between the input in form of rank-related requirements and the output presented as the extreme ranks (see Sect. 4.2). Consequently, the presentation of the extreme ranks may constitute a good support for generating reactions from the part of the DM, who may incrementally supply rank-related requirements.

Let us denote the range of desired ranks for a particular reference alternative $a^* \in A$ provided by the DM with $[P_{DM}^*(a^*), P_{*,DM}(a^*)]$. The constraints referring to the desired values of some reference alternatives may be expressed as follows:

$$U(a^*) \in [U_{*,DM}(a^*), U_{DM}^*(a^*)],$$

where $U_{*,DM}(a^*) \leq U_{DM}^*(a^*)$ are precise values from the range $[0, 1]$ that are provided by the DM. Formally, these requirements are translated into the following constraints: for $a^* \in A^R$

$$\left. \begin{aligned} & U(a^*) - U(b) + M \cdot v_{a^*,b}^> \geq \varepsilon, \\ & \text{for all } b \in A \setminus \{a^*\} \\ & \sum_{b \in A \setminus \{a^*\}} v_{a^*,b}^> \leq P_{*,DM}(a^*) - 1 \\ & U(b) - U(a^*) + M \cdot v_{a^*,b}^< \geq \varepsilon, \\ & \text{for all } b \in A \setminus \{a^*\} \\ & \sum_{b \in A \setminus \{a^*\}} v_{a^*,b}^< \leq n - P_{DM}^*(a^*) \\ & v_{a^*,b}^> + v_{a^*,b}^< \leq 1, \\ & \text{for all } b \in A \setminus \{a^*\} \end{aligned} \right\} \text{if } a^* \Rightarrow [P_{DM}^*(a^*), P_{*,DM}(a^*)],$$

$$\left. \begin{aligned} & U(a^*) \geq U_{*,DM}(a^*) \\ & U(a^*) \leq U_{DM}^*(a^*) \end{aligned} \right\} \text{if } U(a^*) \in [U_{*,DM}(a^*), U_{DM}^*(a^*)],$$

$$\left. \begin{aligned} & \left. \begin{aligned} & U(a^*) - U(b) + M \cdot v_{a^*,b}^> \geq \varepsilon, \\ & \text{for all } b \in A \setminus \{a^*\} \\ & \sum_{b \in A \setminus \{a^*\}} v_{a^*,b}^> \leq P_{*,DM}(a^*) - 1 \\ & U(b) - U(a^*) + M \cdot v_{a^*,b}^< \geq \varepsilon, \\ & \text{for all } b \in A \setminus \{a^*\} \\ & \sum_{b \in A \setminus \{a^*\}} v_{a^*,b}^< \leq n - P_{DM}^*(a^*) \\ & v_{a^*,b}^> + v_{a^*,b}^< \leq 1, \\ & \text{for all } b \in A \setminus \{a^*\} \end{aligned} \right\} \text{if } a^* \Rightarrow [P_{DM}^*(a^*), P_{*,DM}(a^*)], \\ & \left. \begin{aligned} & U(a^*) \geq U_{*,DM}(a^*) \\ & U(a^*) \leq U_{DM}^*(a^*) \end{aligned} \right\} \text{if } U(a^*) \in [U_{*,DM}(a^*), U_{DM}^*(a^*)], \end{aligned} \right\} E_{RUTA}^A$$

where M is an auxiliary variable equal to a big positive value, $v_{a^*,b}^>$ and $v_{a^*,b}^<$ are binary variables associated with comparison of a^* to alternative b . Using these binary variables, the above set of constraints guarantees that there are at most $P_{*,DM}(a^*) - 1$ alternatives which are ranked better than a^* , and at most $n - P_{DM}^*(a^*)$ alternatives which are ranked worse than a^* .

Comparison: use of rank-related requirements in PL-ML Rank-related requirements concerning positions of the alternative have not been yet considered as an input information in PL-ML. Nevertheless, some algorithms (see, e.g., Rudin 2009; Usunier et al. 2009) do consider the positions of items in the final ranking. Precisely, since top positions are important for users, the focus is put on the top-ranked items by punishing the errors occurring in the head of the ranking. Moreover, the direct feedback in form of exemplary utility degrees is admitted by several PL-ML methods.

3.4 Hierarchy of criteria

Complex real-world decision problems, such as choosing a new product pricing strategy, deciding where to locate manufacturing plants, or forecasting the future of a country, involve

factors of different nature. These factors may be political, economic, cultural, environmental, technological, or managerial. Obviously, it is difficult for the DMs to consider so different points of view simultaneously when assessing the quality of the alternatives.

By applying a hierarchical structure, such complex decision problems can be decomposed into a hierarchy of more easily comprehended sub-problems, each of which can be analyzed independently (Saaty 2005). It is the matter of fact that a hierarchy is an efficient way to organize complex systems, being efficient both structurally, for representing a system, and functionally, for controlling and passing information down the system. Once the hierarchy is built, the DMs may first judge the alternatives and then receive feedback about them, with respect to their impact on a particular element in the hierarchy.

In fact, practical applications are often explicitly imposing a hierarchical structure of criteria. For example, in economic ranking, alternatives may be evaluated on indicators which aggregate evaluations on several sub-indicators, and these sub-indicators may aggregate another set of sub-indicators, etc. In this case, the marginal value functions may refer to all levels of the hierarchy, representing values of particular scores of the alternatives on indicators, sub-indicators, sub-sub-indicators, etc. In order to treat this case, we extend the previously introduced notation:

- l is the number of levels in the hierarchy of criteria,
- \mathcal{G} is the set of all criteria at all considered levels,
- $\mathcal{I}_{\mathcal{G}}$ is the set of indices of particular criteria representing position of criteria in the hierarchy,
- m is the number of the first level criteria, G_1, \dots, G_m ,
- $G_{\mathbf{r}} \in \mathcal{G}$, with $\mathbf{r} = (i_1, \dots, i_h) \in \mathcal{I}_{\mathcal{G}}$, denotes a subcriterion of the first level criterion G_{i_1} at level h ; the first level criteria are denoted by $G_{i_1}, i_1 = 1, \dots, m$,
- $n(\mathbf{r})$ is the number of subcriteria of $G_{\mathbf{r}}$ in the subsequent level, i.e. the direct subcriteria of $G_{\mathbf{r}}$ are $G_{(\mathbf{r},1)}, \dots, G_{(\mathbf{r},n(\mathbf{r}))}$,
- $g_{\mathbf{t}}: A \rightarrow \mathcal{R}$, with $\mathbf{t} = (i_1, \dots, i_l) \in \mathcal{I}_{\mathcal{G}}$, denotes an elementary subcriterion of the first level criterion G_{i_1} , i.e a criterion at level l of the hierarchy tree of G_{i_1} ,
- EL is the set of indices of all elementary subcriteria:

$$EL = \{ \mathbf{t} = (i_1, \dots, i_l) \in \mathcal{I}_{\mathcal{G}} \} \quad \text{where} \quad \begin{cases} i_1 = 1, \dots, m, \\ i_2 = 1, \dots, n(i_1), \\ \dots \\ i_l = 1, \dots, n(i_1, \dots, i_{l-1}) \end{cases}$$

- $E(G_{\mathbf{r}})$ is the set of indices of elementary subcriteria descending from $G_{\mathbf{r}}$, i.e.

$$E(G_{\mathbf{r}}) = \{ (\mathbf{r}, i_{h+1}, \dots, i_l) \in \mathcal{I}_{\mathcal{G}} \} \quad \text{where} \quad \begin{cases} i_{h+1} = 1, \dots, n(\mathbf{r}), \\ \dots \\ i_l = 1, \dots, n(\mathbf{r}, i_{h+1}, \dots, i_{l-1}) \end{cases}$$

thus, $E(G_{\mathbf{r}}) \subseteq EL$.

In case of hierarchy of criteria, the DM may provide a partial preorder $\succsim_{\mathbf{r}}$ on A^R or a partial preorder $\succsim_{\mathbf{r}}^*$ on $A^R \times A^R$, which should be interpreted analogously as pairwise comparisons or statements regarding the intensity of preference, however, limited only to a criterion/subcriterion $G_{\mathbf{r}}$. In this context, the value function of an alternative $a \in A$ with respect to criterion/subcriterion $G_{\mathbf{r}}$ is:

$$U_{\mathbf{r}}(a) = \sum_{\mathbf{t} \in E(G_{\mathbf{r}})} u_{\mathbf{t}}(g_{\mathbf{t}}(a)).$$

The transition from the partial preorders to a value function is the following: for $a^*, b^*, c^*, d^* \in A^R$,

$$\left. \begin{aligned} U_r(a^*) &\geq U_r(b^*) + \varepsilon, && \text{if } a^* \succ_r b^*, \text{ for } G_r \in \mathcal{G}, \\ U_r(a^*) &= U_r(b^*), && \text{if } a^* \sim_r b^*, \text{ for } G_r \in \mathcal{G}, \\ U_r(a^*) - U_r(b^*) &\geq U_r(c^*) - U_r(d^*) + \varepsilon, \\ &\text{if } (a^*, b^*) \succ_r^* (c^*, d^*), \text{ for } G_r \in \mathcal{G}, \\ U_r(a^*) - U_r(b^*) &= U_r(c^*) - U_r(d^*), \\ &\text{if } (a^*, b^*) \sim_r^* (c^*, d^*), \text{ for } G_r \in \mathcal{G}. \end{aligned} \right\} E_{HIER}^{AR}$$

Note that limiting the preference information only to a criterion/subcriterion G_r allows more precise representation of preferences in this region which is of great importance for the zooming capacity of a preference construction process.

Comparison: use of a hierarchical attribute structure in PL-ML PL-ML has not been concerned with the organization of the attributes into a hierarchical structure. Nevertheless, one has developed approaches that decompose the complex problem into a set of simpler sub-problem, and then aggregate the partial results into the final ranking. For example, in Qin et al. (2007) one trains a set of Ranking SVM models; each one for the item pairs with two categories of judgments. Subsequently, rank aggregation is used to merge the ranking results given by each model to produce the final ranking result. It seems straightforward to account for the hierarchical structure of attributes within such setting.

3.5 Interaction between criteria

Even if the additive model is among the most popular ones, some critics have been addressed to this model because it has to obey an often unrealistic hypothesis about preferential independence among criteria. In consequence, it is not able to represent *interactions* among criteria. For example, consider evaluation of cars using such criteria as maximum speed, acceleration and price. In this case, there may exist a negative interaction (*negative synergy*) between maximum speed and acceleration because a car with a high maximum speed also has a good acceleration, so, even if each of these two criteria is very important for a DM who likes sport cars, their joint impact on reinforcement of preference of a more speedy and better accelerating car over a less speedy and worse accelerating car will be smaller than a simple addition of the two impacts corresponding to each of the two criteria considered separately in validation of this preference relation. In the same decision problem, there may exist a positive interaction (*positive synergy*) between maximum speed and price because a car with a high maximum speed usually also has a high price, and thus a car with a high maximum speed and relatively low price is very much appreciated. Thus, the comprehensive impact of these two criteria on the strength of preference of a more speedy and cheaper car over a less speedy and more expensive car is greater than the impact of the two criteria considered separately in validation of this preference relation.

To handle the interactions among criteria, one can consider *non-additive integrals*, such as Choquet integral and Sugeno integral (see, e.g., Grabisch 1996). However, the non-additive integrals suffer from limitations within MCDA (see Roy 2009); in particular, they need that the evaluations on all criteria are expressed on the same scale. This means that in order to apply a non-additive integral it is necessary, for example, to estimate if the maximum speed of 200 km/h is as valuable as the price of 35,000 euro.

Thus, in the UTA^{GMS}-INT method (Greco et al. 2013), one has proposed to consider a value function, composed not only of the sum of marginal non-decreasing value functions $u_j(a)$ ($j = 1, \dots, m$), but also of sums of functions

$$\text{syn}_{j_1, j_2}^+ : [x_{j_1}^1, x_{j_1}^{n_{j_1}(A)}] \times [x_{j_2}^1, x_{j_2}^{n_{j_2}(A)}] \rightarrow [0, 1],$$

$$\text{and } \text{syn}_{j_1, j_2}^- : [x_{j_1}^1, x_{j_1}^{n_{j_1}(A)}] \times [x_{j_2}^1, x_{j_2}^{n_{j_2}(A)}], \quad (j_1, j_2) \in J \times J, j_1 > j_2.$$

Functions $\text{syn}_{j_1, j_2}^+(x_{j_1}, x_{j_2})$ and $\text{syn}_{j_1, j_2}^-(x_{j_1}, x_{j_2})$, are non-decreasing in both their two arguments, for all pairs of (possibly) interacting criteria $(j_1, j_2) \in J \times J$, such that $j_1 > j_2$. They correspond to positive or negative interactions, respectively, and add to or subtract from the additive component of the value function. This is why one can call them *bonus* or *penalty functions* with respect to the main additive component. Obviously, a pair of interacting criteria can either be in positive or negative synergy, which means that $\text{syn}_{j_1, j_2}^+(\cdot, \cdot)$ and $\text{syn}_{j_1, j_2}^-(\cdot, \cdot)$ are mutually exclusive:

$$\text{syn}_{j_1, j_2}^+(x_{j_1}, x_{j_2}) \times \text{syn}_{j_1, j_2}^-(x_{j_1}, x_{j_2}) = 0,$$

$$\text{for all } (j_1, j_2) \in J \times J, j_1 > j_2, \text{ and } (x_{j_1}, x_{j_2}) \in X_{j_1} \times X_{j_2}.$$

Under these conditions, for all $a \in A$, the value function is defined as:

$$U^{int}(a) = \sum_{j=1}^m u_j(a) + \sum_{(j_1, j_2) \in J \times J, j_1 > j_2} \text{syn}_{j_1, j_2}^+(g_{j_1}(a), g_{j_2}(a)) - \sum_{(j_1, j_2) \in J \times J, j_1 > j_2} \text{syn}_{j_1, j_2}^-(g_{j_1}(a), g_{j_2}(a)).$$

U^{int} should satisfy usual normalization and monotonicity conditions of value functions. Moreover, to ensure both non-negativity of $U^{int}(a)$, for all $a \in A$, and the monotonicity for the components concerning positive and negative interactions, it is necessary to impose some additional constraints:

$$\left. \begin{aligned} &U^{int}(a) \geq 0, \quad \text{for all } a \in A, \\ &\text{syn}_{j_1, j_2}^+(g_{j_1}(a), g_{j_2}(a)) \geq \text{syn}_{j_1, j_2}^+(g_{j_1}(b), g_{j_2}(b)), \\ &\text{syn}_{j_1, j_2}^-(g_{j_1}(a), g_{j_2}(a)) \geq \text{syn}_{j_1, j_2}^-(g_{j_1}(b), g_{j_2}(b)), \quad (j_1, j_2) \in J \times J, \\ &\quad j_1 > j_2, \text{ if } g_{j_1}(a) \geq g_{j_1}(b) \text{ and } g_{j_2}(a) \geq g_{j_2}(b), \text{ for all } a, b \in A, \\ &\text{syn}_{j_1, j_2}^+(x_{j_1^*}, x_{j_2^*}) = 0, \quad \text{syn}_{j_1, j_2}^-(x_{j_1^*}, x_{j_2^*}) = 0, \\ &\quad (j_1, j_2) \in J \times J, j_1 > j_2, \\ &u_{j_1}(g_{j_1}(a)) + u_{j_2}(g_{j_2}(a)) - \text{syn}_{j_1, j_2}^-(g_{j_1}(a), g_{j_2}(a)) \\ &\quad \geq u_{j_1}(g_{j_1}(b)) + u_{j_2}(g_{j_2}(b)) - \text{syn}_{j_1, j_2}^-(g_{j_1}(b), g_{j_2}(b)), \\ &\quad (j_1, j_2) \in J \times J, j_1 > j_2, \text{ for all } a, b \in A, \text{syn}_{j_1, j_2}^+(x_{j_1}^*, x_{j_2}^*) \leq \delta_{j_1, j_2}^+, \\ &\text{syn}_{j_1, j_2}^-(x_{j_1}^*, x_{j_2}^*) \leq \delta_{j_1, j_2}^-, \\ &\delta_{j_1, j_2}^- + \delta_{j_1, j_2}^+ \leq 1, \\ &\delta_{j_1, j_2}^-, \delta_{j_1, j_2}^+ \in \{0, 1\} \quad \text{for } (j_1, j_2) \in J \times J, j_1 > j_2. \end{aligned} \right\} E_{INT}^A$$

Obviously, using U^{int} it is possible to incorporate preference information of the DM in the same way as with the use of a traditional additive value function.

Although U^{int} takes into account all possible positive and negative interactions between pairs of criteria, in practical decision situations a limited number of interacting criteria would be preferable. One can easily identify a minimal set of pairs of criteria for which there is either positive or negative interaction using a procedure discussed in (Greco et al. 2013). Such a set is presented to the DM for validation. If the DM accepts the proposed solution as relevant set of interacting pairs of criteria, it is fixed for computing recommendation on the whole set A of alternatives. Alternatively, (s)he may either deny or impose interaction between a specific pair of criteria, posing requirements that need to be taken into account by the method in the next proposal.

Comparison: interactions between the inputs in PL-ML In order to properly capture the dependencies between the inputs and the output, in ML one employs complex non-linear models such as neural networks or kernel machines. While being sufficient for this purpose, their comprehensibility to the user is very limited, the monotonicity is difficult to assure, some non-desired restrictions on the model space are imposed, and, moreover, some of these models fail to account for the negative interactions. To address these problems, one has advocated in PL-ML for the use of the Choquet integral (Tehrani et al. 2012a) (for an application of ROR to Choquet integral see Angilella et al. 2010). The presented experimental results suggest that the combination of monotonicity and flexibility offered by this operator facilitates strong performance in practical applications.

While modeling interactions between attributes in ROR inherits the aforementioned advantages, it compares positively to the Choquet integral for two main reasons:

- it does not require that all criteria are expressed on the same scale, which is a serious burden for the use of the Choquet integral in real-world decision problems,
- it offers greater flexibility than the Choquet integral, being able to represent preferences of the DM in some simple scenarios in which preference independence is not satisfied, when the Choquet integral fails to reproduce such preferences (see Greco et al. 2013).

3.6 Margin of the misranking error

In order to verify that the set of value functions \mathcal{U}_{AR} compatible with preference information provided by the DM is not empty, we consider the following mathematical programming problem:

$$\text{Maximize: } \varepsilon, \quad \text{subject to } E^{AR}, \quad (2)$$

where E^{AR} is composed of E_{BASE}^{AR} (monotonicity and normalization constraints), E_{GMS}^{AR} (the set of constraints encoding the pairwise comparisons as in the UTA^{GMS} method), E_{GRIP}^{AR} (intensities of preference in GRIP), E_{RUTA}^{AR} (rank-related requirements in RUTA), E_{HIER}^{AR} (judgments in Hierarchical ROR), and E_{INT}^{AR} (interactions between criteria in UTA^{GMS}-INT).

Let us denote by ε^* the maximal value of ε obtained from the solution of the above MILP problem, i.e., $\varepsilon^* = \max \varepsilon$, subject to E^{AR} . It corresponds to a margin of the misranking error. We conclude that \mathcal{U}_{AR} is not empty, if E^{AR} is feasible and $\varepsilon^* > 0$. In such a case, there exists ε greater than 0 for which the set of constraints is feasible, which means that all pieces of preference information could be reproduced by at least one value function. On the contrary, when E^{AR} is infeasible or the margin of the misranking error is not greater than zero, some pieces of DM's preference on the set of reference alternatives A^R are conflicting (thus, a misranking error is greater than zero).

Comparison: interpretation of the notion of “ordinal regression” in ROR-MCDA and ML The original meaning of the regression analysis by Linear Programming (LP) has been given by Charnes et al. (1955), who applied goal programming to estimate salaries of new employees in an enterprise; their model explains both a numerical variable (typical salary) and an ordinal variable (hierarchy of the enterprise) by numerical variables, and aims at minimizing the sum of absolute spreads between typical and estimated salaries. The original meaning of the ordinal regression has been given by Srinivasan (1976), who applied the ideas of Charnes et al. (1955) in a model called ORDREG; this model used goal programming to explain in terms of numerical variables (weights of multiple attributes) a set of pairwise comparisons of some stimuli.

In statistics, the ordinal regression based on LP has been used to find a numerical representation (encoding) of ordinal variables while minimizing an error function. This idea has been rigorously applied in the UTA method (Jacquet-Lagrèze and Siskos 1982), where a LP model is used to find an optimal additive encoding in ordinal regression. In UTA, a total preorder is explained by a sum of monotone functions involving both qualitative and quantitative variables having the meaning of (also monotonic) criteria. In the same spirit, a partial pre-order is explained in the UTA^{GMS} method (Greco et al. 2008). Let us remind that this approach initiated the stream of further developments in ROR, providing a name for the whole spectrum of methods.

In the ML community, ordinal regression (also known as ordinal classification) is considered as a supervised learning task that consists in determining the implied ordinal rating of items on a fixed, discrete rating scale. A closely related problem, which is also considered in ML as an ordinal regression, focuses more on the relative order between pairs of items rather than on the accurate assignment of an item to one of the ordered categories. The latter interpretation is analogous to the one considered in ROR.

Let us briefly recall a few PL-ML methods designed for dealing with the ordinal regression problems. Comparing them to ROR, they take into account neither the monotonicity nor the variability of the input information. Moreover, most of them deal with a single model instance (Herbrich et al. 2000; Joachims 2002), and even when taking into account multiple instances (Chu and Ghahramani 2005a), they consider a probability distribution on these instances, which is not the case in ROR.

In particular, Herbrich et al. (2000) applied the principle of Structural Risk Minimization to ordinal regression leading to a new distribution-independent learning algorithm based on a loss function between pairs of ranks. A similar kernel approach for representing ranking functions of the generalized form within the context of SVM formulations, was presented in Joachims (2002). Further, Waegeman et al. (2009) extended this method to relational models.

A different approach consists in applying Gaussian process for ordinal regression (Chu and Ghahramani 2005a). Its fundamental assumption is that there is an unobservable latent function value $f(a^*)$ associated with each training sample a^* , and that the function values $\{f(a^*)\}$ preserve the preference relations observed in the data sets (Chu and Ghahramani 2005b). One imposes a Gaussian process prior on these latent function values, and employs an appropriate likelihood function to learn from the pairwise preference between samples. Then, all the parameters are estimated by using a Bayesian approach.

4 Recommendation

Any value function belonging to the set of compatible value functions \mathcal{U}_{AR} reproduces all pieces of preference information given by the DM. Considering all compatible instances of

the preference model involves the trade-off between acting with prudence and arriving at a complete recommendation in a fast way, which is, however, vulnerable to some risks. In fact, selection of a single model instance is usually attained by solving a kind of an optimization problem. Such approach fails to investigate whether there are other optimal (or slightly sub-optimal) models and it sticks to a rather arbitrary rule for the selection (e.g., a specific formulation of the loss function). In this way, the user is not provided with the possible results in case some other optimal model or selection rule was considered. Moreover, in a constructive learning perspective, where the aim is not to predict, but rather to construct the preferences from scratch, the user has the interest in investigating what are the consequences of her/his partial preferences. Thus, in this case providing immediately a complete order of alternatives is not desirable.

Obviously, the final ranking may vary substantially depending on which solution is selected. ROR applies all compatible functions to work out a recommendation for the set of alternatives A , and examines the influence of input variability or imprecision on variability of the proposed recommendation. In this section, we discuss the wide spectrum of procedures for robustness and sensitivity analysis that could be employed within the framework of ROR.

Comparison: credibility of the compatible models ROR does not consider a probability distribution on the set of all compatible value functions, assigning to all of them the same credibility. On the other hand, as already mentioned in Sect. 3.6, some approaches that consider multiple instance of the model in PL-ML, such as the Gaussian processes for ordinal regression (Chu and Ghahramani 2005a), provide a full probability distribution conditioned on the observed data.

4.1 Necessary and possible preference relations

When comparing a pair of alternatives $(a, b) \in A \times A$ in terms of the recommendation that is provided by any compatible value function, it is reasonable to verify whether a is ranked at least as good as b for all or at least one compatible value function. Answering these questions, UTA^{GMS} (Greco et al. 2008) produces two preference relations in the set of alternatives A :

- necessary weak preference relation \succsim^N holds for a pair of alternatives $(a, b) \in A \times A$, in case $U(a) \geq U(b)$ for all compatible value functions,
- possible weak preference relation \succsim^P holds for a pair of alternatives $(a, b) \in A \times A$, in case $U(a) \geq U(b)$ for at least one compatible value function.

Thus defined, the necessary relations specify the most certain recommendation worked out on the basis of all compatible value functions, while the possible relations identify a recommendation provided by at least one compatible value function. Consequently, the necessary outcomes can be considered as robust with respect to the preference information, as they guarantee that a definite relation is the same whichever compatible model would be used. To verify the truth of the necessary and possible weak preference relations the following programs need to be solved:

$$\begin{array}{l} \text{Maximize: } \varepsilon \\ \text{s.t. } U(b) - U(a) \geq \varepsilon \\ \quad E^{A^R} \end{array} \left. \vphantom{\begin{array}{l} \text{Maximize: } \varepsilon \\ \text{s.t. } U(b) - U(a) \geq \varepsilon \\ \quad E^{A^R} \end{array}} \right\} E^N(a, b),$$

and

$$\begin{aligned} & \text{Maximize: } \varepsilon \\ & \left. \begin{aligned} & U(a) - U(b) \geq 0 \\ & E^{A^R} \end{aligned} \right\} E^P(a, b). \end{aligned}$$

We conclude that $a \succsim^N b$, if $E^N(a, b)$ is not feasible or $\varepsilon_* = \max \varepsilon$, s.t. $E^N(a, b)$, is not greater than 0, and that $a \succsim^P b$, if $E^P(a, b)$ is feasible and $\varepsilon^* = \max \varepsilon$, s.t. $E^P(a, b)$, is greater than 0.

Let us remark that preference relations \succsim^N and \succsim^P are meaningful only if there exists at least one compatible value function. Observe also that in this case, for any $a, b \in A^R$, $a \succ b \Rightarrow a \succsim^N b$ and $a > b \Rightarrow \text{not}(b \succsim^P a)$. In fact, if $a \succ b$, then for any compatible value function $U(a) \geq U(b)$, and, therefore, $a \succsim^N b$. Moreover, if $a > b$, then for any compatible value function $U(a) > U(b)$, and, consequently, there is no compatible value function such that $U(b) \geq U(a)$, which means $\text{not}(b \succsim^P a)$.

The necessary weak preference relation \succsim^N is a partial preorder (i.e., it is reflexive ($a \succsim^N a$, since for all $a \in A$, $U(a) = U(a)$), and transitive (for all $a, b, c \in A$, if $a \succsim^N b$ and $b \succsim^N c$, then $a \succsim^N c$). Possible weak preference relation \succsim^P is a strongly complete binary relation (i.e. for all $a, b \in A$, $a \succsim^P b$ or $b \succsim^P a$), and negatively transitive (i.e. $\forall a, b, c \in A$, if $\text{not}(a \succsim^P b)$ and $\text{not}(b \succsim^P c)$, then $\text{not}(a \succsim^P c)$).

In the same spirit, in GRIP (Figueira et al. 2009) one may consider the necessary and possible weak preference relations connected to the comprehensive (on all criteria) or partial (on a particular criterion) intensity of preference. For example, $(a, b) \succsim^{*N}(c, d)$, if $U(a) - U(b) \geq U(c) - U(d)$ for all compatible value functions. On the other hand, $(a, b) \succsim_j^{*P}(c, d)$, if $u_j(a) - u_j(b) \geq u_j(c) - u_j(d)$ for at least one compatible value function and $j \in J$.

In the case of hierarchy of criteria (Corrente et al. 2012), for each criterion/subcriterion $G_r \in \mathcal{G}$ one could also introduce the necessary and possible preference relations related to the pairwise comparisons (\succsim_r^N and \succsim_r^P) or comparison of intensities of preference between pairs of alternatives (\succsim_r^{*N} and \succsim_r^{*P}). For example, the necessary weak preference relation \succsim_r^N holds for a pair of alternatives $(a, b) \in A \times A$, in case $U_r(a) \geq U_r(b)$ for all compatible value functions. Furthermore, the possible weak preference relation \succsim_r^{*P} holds for two pairs of alternatives $(a, b), (c, d) \in A \times A$, in case $U_r(a) - U_r(b) \geq U_r(c) - U_r(d)$ for at least one compatible value function. Obviously, when verifying the truth or falsity of these relations, one should refer to U_r defined in Sect. 3.4, rather than to U being the sum of all marginal values.

4.2 Extreme ranking analysis

An interesting approach to examine how different can be rankings provided by all compatible value functions is to determine the highest and the lowest ranks, and the score that an alternative can attain. Such an analysis of extreme results (Kadziński et al. 2012a) provides information about its relative performance in comparison to all the remaining alternatives simultaneously rather than in terms of separately conducted pairwise comparisons. In order to identify the range of ranks that a particular alternative $a \in A$ could attain (we denote it by $[P^*(a), P_*(a)]$), we propose some mixed-integer programming models:

$$\begin{aligned} \text{Minimize: } f_{\max}^{\text{rank}} &= \sum_{b \in A \setminus \{a\}} v_b \\ \text{s.t. } U(a) - U(b) + Mv_b &\geq \varepsilon, \quad \text{for all } b \in A \setminus \{a\} \end{aligned} \left. \vphantom{\begin{aligned} \text{Minimize: } f_{\max}^{\text{rank}} &= \sum_{b \in A \setminus \{a\}} v_b \\ \text{s.t. } U(a) - U(b) + Mv_b &\geq \varepsilon, \quad \text{for all } b \in A \setminus \{a\} \end{aligned}} \right\} E_{\max}^{AR}$$

and

$$\begin{aligned} \text{Minimize: } f_{\min}^{\text{rank}} &= \sum_{b \in A \setminus \{a\}} v_b \\ \text{s.t. } U(b) - U(a) + Mv_b &\geq \varepsilon, \quad \text{for all } b \in A \setminus \{a\} \end{aligned} \left. \vphantom{\begin{aligned} \text{Minimize: } f_{\min}^{\text{rank}} &= \sum_{b \in A \setminus \{a\}} v_b \\ \text{s.t. } U(b) - U(a) + Mv_b &\geq \varepsilon, \quad \text{for all } b \in A \setminus \{a\} \end{aligned}} \right\} E_{\min}^{AR}$$

where M is an auxiliary variable equal to a big positive value, and v_b is a binary variable associated with comparison of a to alternative b . Note that in both above problems there are $n - 1$ such variables, each corresponding to $b \in A \setminus \{a\}$. We conclude that:

$$P^*(a) = f_{\max}^{\text{rank}} + 1 \quad \text{and} \quad P_*(a) = n - f_{\min}^{\text{rank}}.$$

Obviously, one may also analyze the ranges of comprehensive values $[U_*(a), U^*(a)]$ that a particular alternative a could attain. Identification of the bounds of such a range requires minimization and maximization of $U(a)$, subject to E^{AR} .

Let us remark that the ranges of allowed ranks and comprehensive values for reference alternatives constitute the subsets of the ranges desired by the DM, i.e.: for all $a^* \in A^R$,

$$\begin{aligned} [P_{DM}^*(a^*) \leq P^*(a^*) \text{ and } P_*(a^*) \leq P_{*,DM}(a^*)] \\ \text{and } [U_*(a^*) \geq U_{*,DM}(a^*) \text{ and } U_{DM}^*(a^*) \geq U^*(a^*)]. \end{aligned}$$

In Kadziński and Tervonen (2013) the analysis of the possible and necessary preference relations and the ranges of ranks the alternatives may obtain has been enriched with exposition of probabilities of the possible relations and distribution of the ranks. Moreover, one has proved that we do not need to assess whether an alternative can obtain ranks between the extreme ones; assuming no shared ranks, an alternative can obtain them all.

Comparison: risk of erroneous predictions The preference relations and extreme ranks resulting from ROR minimize the risk of a false declaration that:

- a is preferred to b when there is no compatible value function for which a is preferred to b , i.e. $\neg(a \succsim^P b)$;
- a is not preferred to b when for all compatible value functions a is preferred to b , i.e. $a \succsim^N b$;
- a should be ranked outside the calculated interval of positions, i.e. $[P^*(a), P_*(a)]$.

Let us note that such a risk is not considered by default in the majority of ML methods (it is not exhibited to the DM as it is done in ROR), which apply a single preference model minimizing the loss function on the set of alternatives.

Nevertheless, within a ML setting, it is possible to equip the prediction with information about its uncertainty. For example, a confidence interval in regression is used to indicate the reliability of an estimate, i.e. how much an estimator can deviate from a “true” value.

Moreover, there exist approaches which admit a refusal of a prediction in case of uncertainty. For example, Herbei and Wegkamp (2006) consider classifiers that render three possible outputs: 0, 1 and R . The option R expresses doubt and is used to distinguish observations that are hard to classify in an automatic way. The possibility of taking no decision (“I do not know”) is of great importance in practice, for instance, in case of medical diagnoses. This option has been subsequently considered, e.g., in the context of SVMs (see Bartlett and Wegkamp 2008; Grandvalet et al. 2008).

Comparison: interpretation of the concept of “robustness” in ROR-MCDA and ML Although the term “robust” is widely used in the MCDA context, it does not have a unique definition and clear interpretation. As noted by Vincke (1999), a decision is robust if it keeps open for the future as many good plans as possible, whereas a solution is robust if it is good for all or most plausible sets of values for the data in the model. Furthermore, conclusion is said to be robust if it is valid for all or most acceptable sets of values for the parameters of the model. Note that these explanations are valid for both the necessary preference relations and the range of extreme ranks.

According to another related interpretation, robustness concerns the capacity for withstanding “vague approximations” or “zone of ignorance” in order to prevent undesirable impacts (Roy 2010a). In ROR, the vague approximation and zone of ignorance regard the considered set of value functions. The concepts of the necessary, possible, and extreme results are appropriate for avoiding undesirable impacts related to mis-ranking of some alternatives in case of neglecting some compatible value function.

The above interpretation of robustness is similar to an understanding of this concept in different domains. In statistics, robustness regards the search for methods that are not unduly affected by outliers or other small departures from model assumptions (Hampel et al. 1986). Indeed in statistics, and consequently in PL based on the statistical methods, an estimation relies heavily on the assumptions which are often not met in practice, such as normal distribution of data errors.

Within the theory of decision under uncertainty, robustness is also related to an assumption about probabilities of various outcomes, particularly if rare but extreme-valued events are highly influential. In this context several approaches have been proposed. Some of them do not consider any probability distribution on the states of the world, proposing criteria such as Wald maxmin criterion or Savage minimax of regrets. In other models a family of probability distributions is considered suggesting a cautious decision such as maxmin of expected utility (Gilboa and Schmeidler 1989) or a partial order representing the preferences that hold for all the considered probability distributions (Bewley 2002). Note that the latter corresponds to the necessary preference relation in ROR. Finally, let us mention the concept of “fairness” that is used to describe decisions which are fair with respect to uncertainty. It is related, e.g., to the Lorenz dominance which refines Pareto dominance and favors well-balanced alternatives allowing to establish a preliminary preference relation (Ogryczak and Ruszczyński 1999).

In PL-ML, the concept of robustness has been considered with respect to the sensitivity to noise in learning. In this perspective, a model is robust if it avoids that few noisy data can lead to a large number of mis-rankings (Carvalho et al. 2008) or it minimizes the probability of switching neighboring pairs in a search result when ranking score turbulence happens (Li et al. 2009). In any case, it is acknowledged that the results of research on robustness within PL-ML are still very preliminary (Liu 2011).

4.3 Representative value function

The necessary, possible and extreme results may be difficult to understand by some DMs. To address these potential problems, we propose to select a representative value function (Kadziński et al. 2012b). The motto underlying our proposal is “one for all, all for one”. The representative value function represents all compatible value functions, which also do contribute to its definition. Precisely, this function makes use of the necessary and possible preference relations and extreme ranks. Consideration of these outcomes leads us to formulation of a few targets to be possibly attained by a representative value function. They concern enhancement of differences between comprehensive values of two alternatives. In particular, the DM may wish to emphasize the advantage of some alternatives over the others, which is acknowledged by all compatible value functions, or reduce the ambiguity in the statement of such an advantage, if in the context of all rankings determined by the set of compatible value functions, the result of the comparison of a pair of alternatives is not univocal. In this way, the introduced concept does not contradict the rationale of ROR, because we do not lose the advantage of knowing all compatible instances of the preference model.

Within an interactive procedure for selection of the representative value function, the DM may either wish that the targets are attained one after another, according to a given priority order, or that a compromise between the targets is attained according to some aggregation formula. We propose the following policy with respect to selection of the representative value function: for each pair of alternatives $(a, b) \in A \times A$, the desired difference between their values $U(a)$ and $U(b)$ is conditioned by the target corresponding to one of five relations that is imposed for the pair. In particular, for pairs (a, b) such that:

- $a \succ^N b$ (i.e., $a \succ^N b$ and $\neg(b \succ^N a)$), or $a \succ^P b$ (i.e., $a \succ^P b$ and $\neg(b \succ^P a)$), or $P_*(a) < P^*(b)$, the difference between $U(a)$ and $U(b)$ should be maximized to emphasize the advantage of a over b in the rankings provided by all compatible value functions;
- $a \succ^N b$ (i.e., $\neg(a \succ^N b)$ and $\neg(b \succ^N a)$), or $P^*(a) < P^*(b)$ and $P_*(a) > P_*(b)$, the difference between $U(a)$ and $U(b)$ should be minimized to reduce the ambiguity in designating a better alternative among a and b , when using all compatible value functions.

The optimizations are performed on an incrementally changing set of constraints, which accounts for results from previous optimizations. In case the DM wants to maximize or minimize the difference between values of alternatives $a, b \in A$ related by one of the five relations, the optimization is straightforward. On the other hand, if the DM wishes to obtain a compromise solution with respect to maximization of the difference between $U(a)$ and $U(b)$ for pairs $(a, b) \in A \times A$, such that $P_*(a) < P^*(a)$, or $a \succ^P b$, or $a \succ^N b$, and minimization of the difference between $U(c)$ and $U(d)$ for pairs $(c, d) \in A \times A$, such that $a \succ^N b$ or $P^*(a) < P^*(b)$ and $P_*(a) > P_*(b)$, we add the following constraint to the linear programming constraints considered at the current stage of interaction:

$$U(a) - U(b) \geq U(c) - U(d) + v.$$

Then, we maximize v . The comprehensive values assigned to the alternatives by the representative value function can be used to obtain a complete ranking. The suggested score and position reflect a reasonable compromise between all states of this alternative.

Comparison: use of linear programming in ROR and ML The preference relations in the whole set of alternatives, extreme ranks, as well as a representative value function result from solving some Linear Programming problems. This technique has been also applied in several PL-ML methods. This includes:

- a linear programming-based ranking method proposed in Ataman et al. (2006), which is designed to train a scoring function that ranks all positive points higher than all negative points (from data that is assumed to have binary output);
- a 1-norm SVM (Mangasarian 1998) which is a popular approach for classification; it is well known to be effective in reducing the input space features;
- a method for modeling a utility function with the Choquet integral (Tehrani et al. 2012b); the authors solve an optimization problem whose formulation is in the spirit of ROR mathematical models.

5 Credibility of preference information and recommendation

Robust ordinal regression methods enhance the DM to provide incrementally the preference information by possibly small pieces. This allows both avoiding the necessity of dealing with a large set of reference alternatives already at the initial stages of the interaction as well as controlling the impact of each piece of information (s)he supplied on the result. Such a control is desirable for a truly interactive process.

In particular, the nature of the necessary and possible relations enhances interactive specification of pairwise comparisons. The suggested way of proceeding is to state the truth of the preference relation for a pair of alternatives for which the possible relation was satisfied, but not the necessary one. When it comes to the analysis of extreme results, the DM may wish to narrow down the allowed ranges of ranks or values obtained at the current stage by specifying some new rank-related requirements. Finally, presentation of the complete ranking determined by the representative value function is a good support for generating reactions from the part of the DM. Namely, (s)he could wish to enrich the necessary ranking or to contradict some possible relations, so that these statements are reflected by the representative value function in the next iteration.

Let us denote the preference information provided by the DM in a particular iteration t , $t = 1, \dots, s$, by PI_t^{DM} and the corresponding set of constraints by E_t^{AR} . Let $PI_1^{DM} \subseteq PI_2^{DM} \subseteq \dots \subseteq PI_s^{DM}$, be embedded sets of pieces of preference information. In particular, they may represent the pairwise comparisons ($\succsim_1^{DM} \subseteq \succsim_2^{DM} \subseteq \dots \subseteq \succsim_s^{DM}$) and/or the desired ranges of possible ranks or scores ($[l_{ij}^R]_1^{DM} \subseteq [l_{ij}^R]_2^{DM} \subseteq \dots \subseteq [l_{ij}^R]_s^{DM}$) for some reference alternatives. Clearly, PI_t^{DM} contains more credible pieces of preference information than PI_{t-1}^{DM} , $t = 2, \dots, s$. Any new piece of preference information makes the information more precise, and puts additional constraints to E_t^{AR} , which possibly reduces the set of compatible value functions U_t^{AR} , $t = 1, \dots, s$. Thus, the sets of compatible value functions are embedded in the inverse order of the related sets of pieces of preference information, i.e. $U_1^{AR} \supseteq U_2^{AR} \supseteq \dots \supseteq U_s^{AR}$. We suppose that $U_s^{AR} \neq \emptyset$.

For each iteration t , we can compute the corresponding results in the same way as presented in Sect. 4, but referring to the set of constraints E_t^{AR} rather than E^{AR} . An important property of these outcomes is stated by Proposition 1.

Proposition 1 For $t = 1, \dots, s$:

- \succsim_{t-1}^N and \succsim_t^P are nested relations: $\succsim_{t-1}^N \subseteq \succsim_t^N$ and $\succsim_{t-1}^P \supseteq \succsim_t^P$ (Greco et al. 2008);
- $\succsim_{r,t}^N$ and $\succsim_{r,t}^P$ for each criterion/subcriterion $G_r \in \mathcal{G}$ are nested relations: $\succsim_{r,t-1}^N \subseteq \succsim_{r,t}^N$ and $\succsim_{r,t-1}^P \supseteq \succsim_{r,t}^P$ (Corrente et al. 2012);
- $[P_t^*(a), P_{*,t}(a)]$ and $[U_{*,t}(a), U_t^*(a)]$ are nested intervals: $[P_t^*(a), P_{*,t}(a)] \subseteq [P_{t-1}^*(a), P_{*,t-1}(a)]$ and $[U_{*,t}(a), U_t^*(a)] \subseteq [U_{*,t-1}(a), U_{t-1}^*(a)]$ (Kadziński et al. 2013a).

As a consequence, it is easier for the DMs to associate pieces of their preference information with the result and, therefore, to control the impact of each piece of information (s)he provides on the result.

Obviously, we admit that the DM may remove or modify previously provided pieces of preference information. This is likely to happen, e.g., when the DM changed her/his point of view or in case of inconsistent judgments (see Sect. 6).

6 Dealing with the inconsistency in ROR

In case of incompatibility in ROR, the set of value functions consistent with the provided preference information is empty. This may occur if the preference information of the DM does not match the underlying preference model, or the DM has violated the dominance in her/his statements, or the provided statements are contradictory. Dealing with the inconsistency, the DM may want either to pursue the analysis with such an incompatibility or to identify its reasons in order to remove it.

If the DM wants to pursue the analysis with the incompatibility, (s)he has to accept that some of her/his pairwise comparisons or rank-related requirements will not be reproduced by any value function. From a formal viewpoint, if the polyhedron generated by the set of constraints is empty, then the necessary and possible preference relations as well as extreme ranks are meaningless. The acceptance of inconsistency means that the DM does not change the preference information, and rather uses a set of constraints E_{ext}^{AR} differing from the original one E^{AR} by an additional constraint on the acceptable margin of the misranking error:

$$\varepsilon \geq \varepsilon^{ext},$$

where $\varepsilon^{ext} < \varepsilon^*$, such that $\varepsilon^* = \max \varepsilon$, subject to E^{AR} , so that the resulting new constraints E_{ext}^{AR} are feasible.

Obviously, the provided results would not fully restore the provided pairwise comparisons or rank-related requirements. For instance, there may exist at least one pair $a, b \in A^R$, such that $a \succ b$, but it is false that for all the compatible value functions $U(a) \geq U(b)$, or there may exist at least one $a \in A^R$, such that $a \Rightarrow [P_{DM}^*(a), P_{*,DM}(a)]$, but a value function satisfying E_{ext}^{AR} ranks a better than $P_{DM}^*(a)$ or worse than $P_{*,DM}(a)$.

If the DM does not want to pursue the analysis with the incompatibility, it is necessary to identify the troublesome pieces of preference information responsible for this incompatibility, so as to remove or revise some of them. There may exist several sets of preference information pieces which, once removed, make the set of compatible value functions non-empty. Identifying the troublesome pieces amounts at finding a minimal subset of constraints that, once removed, leads to a set of constraints generating a non-empty polyhedron of compatible value functions.

For this reason, let us associate with each piece of preference information (e.g., the desired range of ranks or pairwise comparisons of reference alternatives) a new binary variable w_C . Using these binary variables, we rewrite a constraint or a set of constraints corresponding to a particular preference statement so that in case $w_C = 1$ it is satisfied whatever the value function is, which is equivalent to its elimination. For example, the pairwise comparison $a^* > b^*$, for $a^*, b^* \in A^R$, is translated into the following constraint:

$$M \cdot w_{a^*,b^*} + U(a^*) \geq U(b^*) + \varepsilon, \tag{3}$$

whereas a rank-related requirement $a^* \Rightarrow [P_{DM}^*(a^*), P_{*,DM}(a^*)]$ is translated into the following set of constraints:

$$\left. \begin{aligned} &M \cdot w_{P(a^*)} + U(a^*) - U(b) + M \cdot v_{a^*,b}^{\geq} \geq \varepsilon, \\ &\quad \text{for all } b \in A \setminus \{a^*\} \\ &\sum_{b \in A \setminus \{a^*\}} v_{a^*,b}^{\geq} \leq P_{*,DM}(a^*) - 1 \\ &M \cdot w_{P(a^*)} + U(b) - U(a^*) + M \cdot v_{a^*,b}^{\leq} \geq \varepsilon, \\ &\quad \text{for all } b \in A \setminus \{a^*\} \\ &\sum_{b \in A \setminus \{a^*\}} v_{a^*,b}^{\leq} \leq n - P_{DM}^*(a^*) \\ &v_{a^*,b}^{\geq} + v_{a^*,b}^{\leq} \leq 1, \quad \text{for all } b \in A \setminus \{a^*\} \end{aligned} \right\} \quad (4)$$

where M is an arbitrarily big positive value.

Then, identifying a minimal subset of troublesome pieces of preference information can be performed by minimizing the sum of all w_C , subject to the rewritten set of constraints E^{AR} . The optimal solution of such problem indicates one of the subsets of smallest cardinality being the cause of incompatibility. Searching for the smallest subset of constraints is consistent with the idea according to which the DMs will first consider the “less complex” ways to solve inconsistency. Other subsets can be obtained following the general scheme for dealing with incompatibility presented in Mousseau et al. (2003). In general, in the subsequent steps we forbid finding again the same solutions which have been already identified in the previously conducted optimizations, which permits to discover new minimal subsets of incompatible constraints. All these subsets of pieces of preference information are to be presented to the DM as alternative ways of removing incompatibility.

Revealing such different possibilities is informative for the DM. Knowing the various ways of solving inconsistency permits her/him to understand the conflicting aspects of her/his statements, to question previously expressed judgments, and to make the elicitation process more flexible. Thus, analyzing and confronting the alternative solutions for removing inconsistency provides opportunity for the DM to learn about her/his preferences as the interactive process evolves.

7 Illustrative case study

In this section, we report results of an illustrative case study concerning innovation. We reconsider data set published by the Economist Intelligence Unit (EIU) in 2007 (EIU 2007). The study aims to measure application of knowledge in a novel way for economic benefit, which is important for both governments and firms. For clarity, we focus on 28 European countries. They are evaluated on two main criteria: innovation performance (g_1) and innovation enablers (g_2). The previous one is based on the international patent data which is the single best available proxy measure for innovation outputs. The natural logarithms of patents per million population are converted by EIU into an index on a 1–10 scale. The other criterion is composed of two other sub-criteria: direct innovation inputs (g_{21}) and innovation environment (g_{22}). They combine several other factors such as: quality of the local research infrastructure, education and technical skills of the workforce (for g_{21}) or political and macroeconomic stability, tax regime, and flexibility of labour market (for g_{22}). The performance matrix is provided in Table 1. To address the problem, we will take advantage of different types of preference information. We will also present a variety of results discussed in this paper.

Table 1 The evaluation matrix, extreme ranks and representative comprehensive values for the problem of evaluating innovation of European countries

Country	g_1	g_{21}	g_{22}	P_1^*	$P_{*,1}$	P_2^*	$P_{*,2}$	U_1^R	U_2^R
SWI	9.71	9.88	8.50	1	5	1	4	0.917	0.857
SWE	9.45	9.94	8.24	2	6	2	5	0.917	0.929
FIN	9.43	10.0	8.48	1	4	1	2	1.000	1.000
GER	9.38	9.56	7.95	4	7	4	6	0.833	0.786
DEN	9.29	9.94	8.61	1	5	2	5	1.000	0.929
NET	9.12	9.63	8.37	4	7	4	6	0.833	0.786
AUT	8.91	8.69	7.31	11	14	11	14	0.583	0.571
FRA	8.90	9.44	7.52	7	9	7	9	0.750	0.714
BEL	8.80	9.06	7.69	7	10	7	9	0.667	0.714
NOR	8.73	8.44	7.38	11	14	11	14	0.583	0.571
UK	8.72	8.81	8.52	2	9	7	9	0.833	0.714
IRE	8.46	8.44	8.42	9	10	10	10	0.667	0.642
SLO	7.68	7.50	6.26	13	20	13	19	0.417	0.428
SPA	7.47	7.94	7.44	11	15	11	15	0.583	0.571
HUN	7.26	6.94	6.76	14	16	14	16	0.500	0.499
EST	6.75	6.94	7.54	11	16	11	16	0.500	0.571
CRO	6.73	6.00	5.54	17	22	17	21	0.250	0.357
POR	6.58	6.75	6.75	16	18	16	18	0.417	0.428
SVK	6.26	6.50	6.70	17	19	17	19	0.333	0.357
RUS	6.07	6.06	4.59	19	23	19	23	0.250	0.285
LAT	5.89	5.63	6.59	18	22	20	22	0.250	0.285
BUL	5.79	5.63	5.51	21	23	21	23	0.167	0.214
POL	5.65	6.25	6.57	18	23	18	23	0.250	0.285
LIT	5.52	5.63	6.43	24	24	24	24	0.083	0.143
UKR	5.37	5.25	3.82	25	28	25	27	0.000	0.071
ROM	5.25	4.69	5.85	25	28	25	28	0.000	0.000
TUR	5.21	4.38	5.93	25	28	26	28	0.000	0.000
SER	4.52	4.94	4.59	25	28	25	28	0.000	0.000

7.1 First iteration

7.1.1 Preference information

In order to rank all alternatives, the DM has to provide preference information concerning some reference alternatives. This information could take the form of pairwise comparisons, or intensities of preference, or desired ranks, or constraints referring to the desired scores of these alternatives. Let us assume, that the DM is familiar with innovation level of some countries, and having analyzed their evaluation profiles, (s)he is able to provide two pairwise comparisons: $FRA > IRE$ and $FIN > SWE$, as well as an imprecise judgment about the desired ranks of two other countries: “AUT should not be ranked in top 10” and “LIT should be ranked among bottom 5 alternatives”.

7.1.2 Necessary and possible preference relations

Let us first discuss the necessary and possible preference relations. The Hasse diagram of the necessary relation obtained for the preference information provided in the first iteration is presented in Fig. 1 (to the left). The matrix of the necessary relation is rather rich. The valid necessary relation \succsim^N for a pair of alternatives $(a, b) \in A \times A$, means that a and b are compared in the same way for all compatible value functions. The graph confirms that the inferred compatible instances of the preference model reproduce the relations which stem from the provided pairwise comparisons, i.e.: $FRA \succ^N IRE$ and $FIN \succ^N SWE$.

The necessary preference relation is transitive. Let us remind that the arrows that can be obtained by transitivity are not represented in the Hasse diagram (see, e.g., (SWI, FRA), (DEN, SER)). Moreover, if the necessary relation held for a given pair of alternatives, then possible relation holds as well. On the other hand, if there was no arrow representing the necessary relation between two countries $a, b \in A$ (e.g., (NET, UK), (EST, SLO)), then these alternatives are incomparable in terms of the necessary relation. This means that for at least one compatible value function a is preferred to b , whereas for some other compatible value function the preference relation is reversed. When analyzing the necessary relation, alternatives SWI, FIN, and DEN should be perceived as the best ones, SWE, NET, UK, and GER should be viewed as relatively good countries, whereas alternatives UKR, ROM, TUR, and SER need to be considered as the worst ones.

7.1.3 Extreme ranking analysis

The results of extreme ranking analysis in the first iteration are presented in Table 1 (columns P_1^* and $P_{*,1}$). One can see that the range of allowed ranks for reference alternatives constitutes the subset of the ranks desired by the DM, i.e.:

$$\begin{aligned} [P_1^*(AUT), P_{*,1}(AUT)] &= [11, 14] \subseteq [11, 28] = [P_{1,DM}^*(AUT), P_{*,1,DM}(AUT)], \\ [P_1^*(LIT), P_{*,1}(LIT)] &= [24, 24] \subseteq [24, 28] = [P_{1,DM}^*(LIT), P_{*,1,DM}(LIT)]. \end{aligned}$$

In fact, for this particular problem, the actual range of attained ranks is a proper subset of the range specified by the DM. Furthermore, since alternatives FRA and FIN were required to be preferred to IRE and SWE, respectively, their best and worst ranks are strictly better (e.g., $P_1^*(FRA) = 7 < P_1^*(IRE) = 9$ and $P_{*,1}(FRA) = 9 < P_{*,1}(IRE) = 10$). The average difference between the worst $P_*(a)$ and the best $P^*(a)$ rank that could be attained by the considered countries is equal to 3.4. Countries SWI, FIN, and DEN are potential top alternatives. Furthermore, SWE, UK, GER, and NET possibly take place in top 5, but UK is more sensitive to the choice of a compatible value function because its rank may drop to 9. Another 8 countries (e.g., AUT, SLO, HUN) are always ranked in the second ten. Finally, UKR, ROM, TUR, and SER are the least ranked alternatives.

7.1.4 Representative value function

We will suppose that the DM wants to select a function that emphasizes the evident advantage of some alternatives over the others acknowledged by all compatible value functions (i.e., maximizes the difference between comprehensive values of alternatives $a, b \in A$, such that $a \succ^N b$), and reduces the ambiguity in the statement of such an advantage, otherwise (i.e., minimizes the difference between comprehensive values of alternatives $c, d \in A$, such that $a \succ^N b$). The comprehensive values of the alternatives obtained for the representative

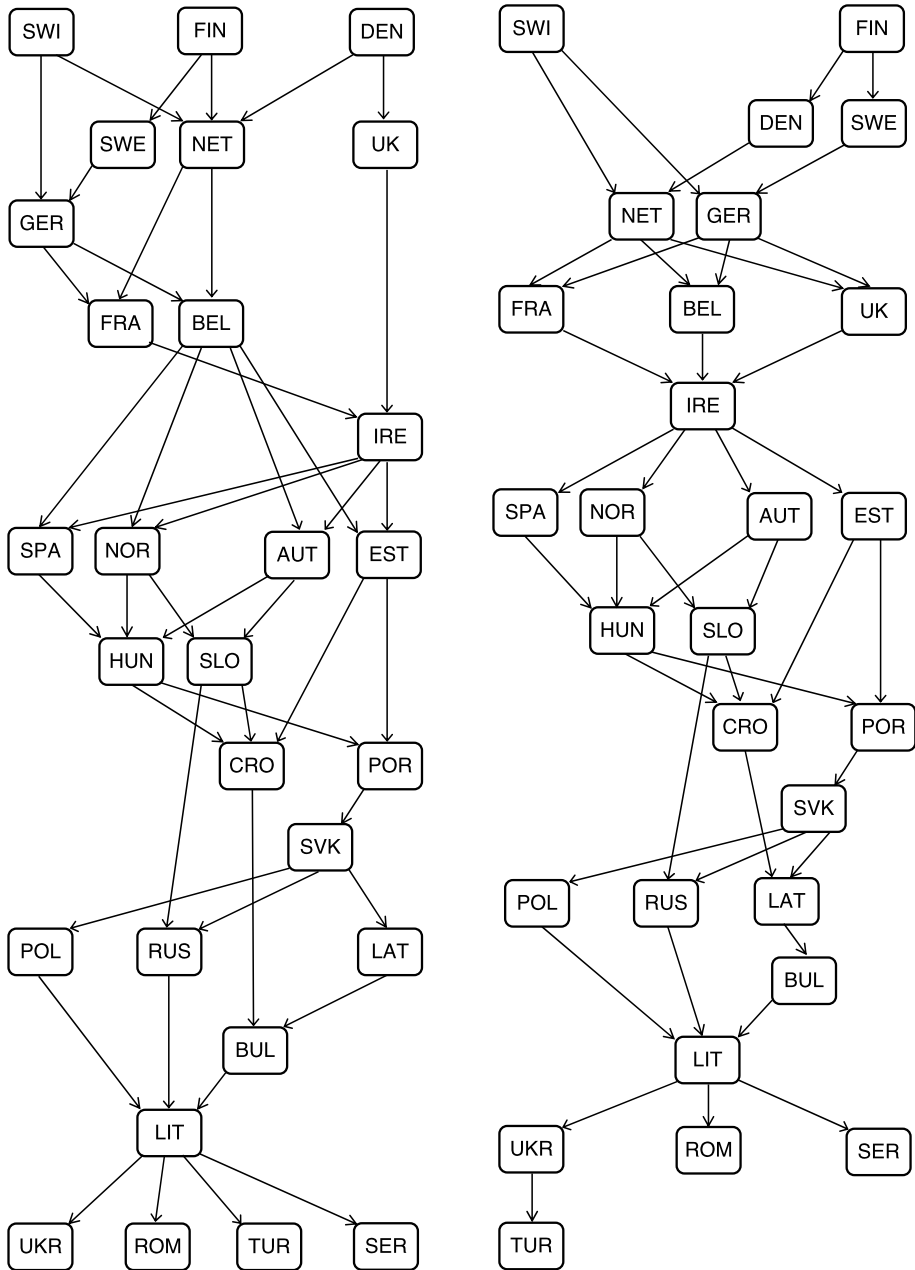


Fig. 1 Hasse diagram of the necessary relation for the problem of evaluating innovation of European countries—first iteration (to the *left*) and second iteration (to the *right*)

value function are presented in Table 1 (column U_1^R). Alternatives FIN and DEN are ranked first with score 1.0. Then, SWI and SWE are placed third with score 0.917. They are followed by GER, NET, and UK which share the same score 0.833. On the other hand, LIT is

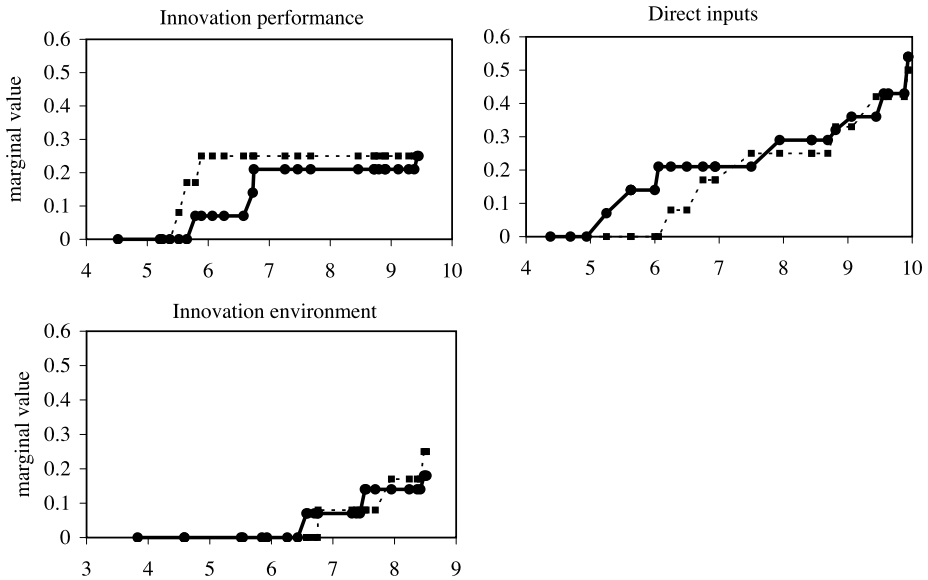


Fig. 2 Representative marginal value functions for the problem of evaluating innovation of European countries—first iteration (*dashed line*) and second iteration (*continuous line*)

ranked 24-th, whereas UKR, ROM, TUR, and SER are ranked at the four bottom place with score 0.0. This value function is the most discriminant with respect to the comprehensive values of alternatives which are related by the necessary preference. Note that the minimal difference between values of alternatives $U(a)$ and $U(b)$ for (a, b) , such that $a \succ^N b$, is equal to 0.083. On the other hand, one can notice that there are a few groups of alternatives which share the same comprehensive value, e.g., {CRO, RUS, LAT, POL}. This is intended, because these alternatives are possibly indifferent, so we wished to minimize the difference between their comprehensive values.

Representative comprehensive values constitute a synthetic representation of the result of the robust ordinal regression. It is the case, because the corresponding ranking “flattens” the graph of the necessary relation to a complete order according to some reasonable assumptions and requirements. As a result, we can focus on a single value function which represents all compatible models. Moreover, the DM may analyze corresponding marginal value functions of the evaluation criteria, which is less abstract than analysis of the whole set of such compatible functions, and may help to justify the decision to the counterparts. These functions are presented in Fig. 2 (dashed line). The constructed functions are usually not strictly monotonic. The characteristic points marked in the figure correspond to the performances of the considered alternatives. In the first iteration, direct inputs (g_{21}) has the greatest share in the comprehensive values and the greatest variation of marginal values.

7.2 Second iteration

Robust ordinal regression methods are intended to be used interactively, that is, the DM can provide progressively new pieces of preference information or change already provided ones. Let us imagine that considering the graph of the necessary relation and the extreme ranks obtained in the first iteration, (s)he provided two additional pairwise comparisons:

CRO \succ LAT and BEL \succ IRE (note that these preferences are missing in the graph of the necessary relation after the first iteration). Moreover, (s)he supplied another three comparisons referring only to the two sub-criteria grouped as innovation enablers (g_2): GER \succ_{g_2} UK, UKR \succ_{g_2} TUR, FIN \succ_{g_2} DEN. Finally, analyzing the complete preorder determined by the representative value function, the DM referred to the comprehensive intensity of preference, requiring that (DEN, UK) \succ (SVK, LIT) (in this way, (s)he reinforces the intensity imposed by the representative value function at the current stage of interaction).

The possible and necessary relations converge with the growth of preference information (see Fig. 1 (to the right)). In particular, the necessary partial preorder is enriched (e.g., GER \succ^N UK), and the possible relation is impoverished (e.g. $\neg(\text{LAT} \succ^P \text{CRO})$). When it comes to the outcomes of extreme ranking analysis, the range of possible ranks in the second iteration is narrower for 13 out of 28 countries. For example, UK and LAT are ranked only 7-th and 20-th in the best case, respectively, whereas in the first iteration they could attain 2-nd and 18-th positions. Furthermore, there are only two countries that could be ranked at the very top (SWI and FIN) and only three alternatives that could be placed at the very bottom (ROM, TUR, and SER). The representative comprehensive values obtained at this stage are provided in Table 1 (column U_2^R) and the corresponding marginal value functions are given in Fig. 2 (continuous line). With respect to the first iteration, one could observe slightly greater variation of the comprehensive values and even greater share of criterion g_{21} in the overall score of the alternatives. In general, incremental specification of preference information allowed obtaining more precise recommendation. It is the case, since new pieces of preference information constrained the set of compatible value functions. Obviously, the interactive process can be pursued until the obtained results are decisive enough for the DM.

Comparison: conducting experimental comparison of different methods In MCDA computational experiments based on comparison of recommendations given by different methods is not advised for:

- different axioms characterizing the methods,
- instrumental bias of the DM, which is unavoidable in a dialogue “question of the method—response of the DM”,
- unrealistic assumption that the DM’s preferences pre-exist, are stable, have an objective reality, and are insensitive to pieces of results communicated to the DM in the interactive process.

On the other hand, PL-ML methods can be compared in terms of the quality of the discovered and predicted preferences, since these preferences have an autonomous reality.

7.3 Computational cost

Let us denote the number of the provided pairwise comparisons of reference alternatives (related either to the holistic evaluation or to a subset of criteria) by PC , the number of statements concerning intensities of preference by INT , the number of rank related requirements by RR , and the number of value related requirements by VR . The number of constraints corresponding to these pieces of preference information is equal to $PI = PC + INT + 5 \cdot RR + 2 \cdot VR$.

In order to check the truth or falsity of the necessary and possible preference relations one needs to solve $2 \cdot n \cdot (n - 1)$ LP problems (to verify the truth or falsity of a given relation, a mathematical problem needs to be solved for each pair of alternatives $(a, b) \in A \times A$, $a \neq b$). However, each of these LP problems is relatively small—the range of dimensions

is $n \cdot m + PI$ constraints and $n \cdot m + n \cdot RR$ variables. On the other hand, to determine the extreme ranks, one needs to solve $2 \cdot n$ MILP problems. Their range of dimension is $n \cdot m + PI + n$ constraints and $n \cdot m + n \cdot RR + n$ variables (with $n \cdot RR + n$ binary variables). Finally, depending on the target chosen by the DM within the procedure for selection of a representative value function, a few LP problems are solved to determine representative comprehensive and marginal values. Additional computational effort in terms of new variables and new LP problems to solve needs to be taken into account in case of considering interactions between criteria and hierarchy of criteria, respectively.

Obviously, solving all these problems is not a burden for contemporary solvers in case the size of the problem is typical for MCDA (i.e., up to hundreds alternatives, and not thousands). For greater sets of alternatives, the mathematical programs would be too big and too many, and thus ROR cannot be used to solve problems in information retrieval, natural language processing, or bio-informatics. Moreover, in case of the big data problems, ROR is losing its advantage, because one cannot present to the DM the ranking of all items. Obviously, then one can employ a representative value function that determines a complete order of the alternatives, but to calculate it is a great burden. Nevertheless, for greater data sets one can take advantage of the two-stage approach mentioned in Sect. 2.2.

8 Conclusions

In this paper, we have reviewed a non-statistical methodology of preference learning designed for multiple criteria ranking. The methodology, called Robust Ordinal Regression, is based on learning of a set of value functions from decision examples given by the DM. Preference information may be composed of pairwise comparison of some alternatives, intensities of preference, rank-related requirements, or statements concerning interaction between criteria. These judgments are represented by a compatible form of a set of value functions, each one defining a complete ranking on the set of alternatives. A value function preference model is of particular interest in MCDA because of an easy interpretation of numerical scores of alternatives and straightforward translation of pieces of preference information to the final result. Moreover, our methodology admits the most general form of an additive model which does not involve any arbitrary parametrization. The marginal value functions composing the additive model are general monotone functions. The result of application of the set of compatible value function on the set of alternatives is presented in the form of the necessary and possible preference relations, extreme ranks, and a representative value function. These outcomes provide a clear justification of recommended rankings at different levels of certainty, and stimulate the DM to interact with the method by incrementally enriching the preference information and observing its necessary and possible consequences on the recommended rankings. These features reveal a specific aspect of learning adopted in ROR. As shown in the paper they contrast with ML which is oriented towards preference discovery without interaction with the DM, but also share many features with the recently proposed PL-ML algorithms that can be used for preference construction.

We would like to end with a conclusion suggesting a consideration of some specific aspects of ROR in PL-ML:

- consideration of a plurality of instances of the considered preference model compatible with preference information within an acceptable probabilistic error;
- exploiting the concepts of the necessary and possible preference relation and of the extreme ranking analysis to ensure a proper trade-off between the completeness and prudence of the recommendation;

- accounting for types of preference information that are considered in ROR; for example, in ROR the DM can refer to rank-related requirements or intensities of preference on a single criterion, which is not the case in PL-ML methods; moreover, it seems to be challenging for ML-PL to account simultaneously for different types of preference information;
- considering some specific concepts introduced in ROR-MCDA, such as hierarchy of criteria, which seem not explored in PL-ML, while being useful for decomposing the complex problems, or interactions between different criteria;
- adopting the idea of preference construction, understood as a mutual learning of the DM and the model; this is related to the development of an interface that would surrogate an analyst and supply the user with the consequences of applying her/his preference in a way that would invite her/him to the interaction.

Acknowledgements The authors thank the editors of special issue on Preference Learning and Ranking, Johannes Fürnkranz and Eyke Hüllermeier, as well as four anonymous reviewers whose comments helped to improve previous versions of the paper. The third and the fourth authors wish to acknowledge financial support from the Polish National Science Centre, grant No. N N519 441939.

Open Access This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

References

- Angilella, S., Greco, S., & Matarazzo, B. (2010). Non-additive robust ordinal regression: a multiple criteria decision model based on the Choquet integral. *European Journal of Operational Research*, 201(1), 277–288.
- Ataman, K., Street, W. N., & Zhang, Y. (2006). Learning to rank by maximizing AUC with linear programming. In *IEEE international joint conference on neural networks, IJCNN 2006* (pp. 123–129).
- Bartlett, P. L., & Wegkamp, M. H. (2008). Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, 9, 1823–1840.
- Belton, V., Branke, J., Eskelinen, P., Greco, S., Molina, J., Ruiz, F., & Słowiński, R. (2008). Interactive multi-objective optimization from a learning perspective. In J. Branke, K. Deb, R. Słowiński, & K. Miettinen (Eds.), *Multiobjective optimization* (pp. 405–433). Berlin: Springer.
- Beuthe, M., & Scannella, G. (2001). Comparative analysis of UTA multicriteria methods. *European Journal of Operational Research*, 130(2), 246–262.
- Bewley, T. F. (2002). Knightian decision theory: part 1. *Decisions in Economics and Finance*, 25, 79–110.
- Braziunas, D., & Boutilier, C. (2005). Local utility elicitation in GAI models. In *Proceedings of the twenty-first conference on uncertainty in artificial intelligence, UAI-05* (pp. 42–49).
- Braziunas, D., & Boutilier, C. (2008). Elicitation of factored utilities. *Artificial Intelligence Magazine*, 29(4), 79–92.
- Burges, Ch. J. C., Ragno, R., & Viet Le, Q. (2006). Learning to rank with nonsmooth cost functions. In *Advances in neural information processing systems: Vol. 19. NIPS 2006* (pp. 193–200).
- Carvalho, V. R., Elsas, J. L., Cohen, W. W., & Carbonell, J. G. (2008). A meta-learning approach for robust rank learning. In *SIGIR (Special Interest Group on Information Retrieval) 2008 workshop on learning to rank for information retrieval, LR4IR 2008* (pp. 2352–2360).
- Charnes, A., Cooper, W. W., & Ferguson, R. (1955). Optimal estimation of executive compensation by linear programming. *Management Science*, 1, 138–151.
- Chu, W., & Ghahramani, Z. (2005a). Gaussian processes for ordinal regression. *Journal of Machine Learning Research*, 6, 1019–1041.
- Chu, W., & Ghahramani, Z. (2005b). Preference learning with Gaussian processes. In *Proceedings of the 22nd international conference on machine learning, ICML '05* (pp. 137–144). New York: ACM.
- Corrente, S., Greco, S., & Słowiński, R. (2012). Multiple criteria hierarchy process in robust ordinal regression. *Decision Support Systems*, 53(3), 660–674.
- Cortes, C., Mohri, M., & Rastogi, A. (2007). Magnitude-preserving ranking algorithms. In *Proceedings of the 24th international conference on machine learning, ICML '07* (pp. 169–176). New York: ACM.

- Despotis, D. K., Yanacopoulos, D., & Zopounidis, C. (1990). A review of the UTA multicriteria method and some improvements. *Foundations of Computing and Decision Sciences*, 15(2), 63–76.
- Doumpos, M., & Zopounidis, C. (2011). Preference disaggregation and statistical learning for multicriteria decision support: a review. *European Journal of Operational Research*, 209(3), 203–214.
- Ehrgott, M., Figueira, J., & Greco, S. (Eds.) (2010). *Trends in multiple criteria decision analysis*. Berlin: Springer.
- EIU (2007). *Innovation: transforming the way business creates*. London: Economist Intelligence Unit.
- Feelders, A. (2010). Monotone relabeling in ordinal classification. In *Proceedings of the 2010 IEEE international conference on data mining, ICDM '10* (pp. 803–808). Washington: IEEE Computer Society.
- Figueira, J., Greco, S., & Ehrgott, M. (Eds.) (2005). *Multiple criteria decision analysis: state of the art surveys*. Berlin: Springer.
- Figueira, J., Greco, S., & Słowiński, R. (2009). Building a set of additive value functions representing a reference preorder and intensities of preference: GRIP method. *European Journal of Operational Research*, 195(2), 460–486.
- Freund, Y., Iyer, R., Schapire, R. E., & Singer, Y. (2003). An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4, 933–969.
- Fürnkranz, J., & Hüllermeier, E. (2010). Preference learning and ranking by pairwise comparison. In J. Fürnkranz & E. Hüllermeier (Eds.), *Preference learning* (pp. 65–82). Berlin: Springer.
- Fürnkranz, J., & Hüllermeier, E. (2011). Preference learning: an introduction. In J. Fürnkranz & E. Hüllermeier (Eds.), *Preference learning* (pp. 1–17). Berlin: Springer.
- Gilboa, I., & Schmeidler, D. (1989). Maxmin expected utility with non-unique prior. *Journal of Mathematical Economics*, 18(2), 141–153.
- Grabisch, M. (1996). The application of fuzzy integrals in multicriteria decision making. *European Journal of Operational Research*, 89(3), 445–456.
- Grandvalet, Y., Rakotomamonjy, A., Keshet, J., & Canu, S. (2008). Support vector machines with a reject option. In *Advances in neural information processing systems: Vol. 21. NIPS 2008* (pp. 537–544).
- Greco, S., Mousseau, V., & Słowiński, R. (2008). Ordinal regression revisited: multiple criteria ranking using a set of additive value functions. *European Journal of Operational Research*, 191(2), 415–435.
- Greco, S., Słowiński, R., Figueira, J., & Mousseau, V. (2010). Robust ordinal regression. In M. Ehrgott, J. Figueira, & S. Greco (Eds.), *Trends in multiple criteria decision analysis* (pp. 273–320). Berlin: Springer.
- Greco, S., Kadziński, M., Mousseau, V., & Słowiński, R. (2012). Robust ordinal regression for multiple criteria group decision problems: UTA^{GMS}-GROUP and UTADIS^{GMS}-GROUP. *Decision Support Systems*, 52(3), 549–561.
- Greco, S., Mousseau, V., & Słowiński, R. (2013). *UTA^{GMS}-INT: robust ordinal regression of value functions handling interacting criteria*. Submitted manuscript.
- Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., & Stahel, W. A. (1986). *Robust statistics. The approach based on influence functions*. New York: Wiley.
- Herbei, R., & Wegkamp, M. H. (2006). Classification with reject option. *Canadian Journal of Statistics*, 4, 709–721.
- Herbrich, R., Graepel, T., & Obermayer, K. (1999). *Regression models for ordinal data: a machine learning approach* (Technical report TR-99/03). TU Berlin.
- Herbrich, R., Graepel, T., & Obermayer, K. (2000). Large margin rank boundaries for ordinal regression. In P. J. Bartlett, B. Schölkopf, D. Schuurmans, & A. J. Smola (Eds.), *Advances in large margin classifiers* (pp. 115–132). Cambridge: MIT Press.
- Jacquet-Lagrèze, E., & Siskos, Y. (1982). Assessing a set of additive utility functions for multicriteria decision making: the UTA method. *European Journal of Operational Research*, 10, 151–164.
- Joachims, T. (2002). Optimizing search engines using clickthrough data. In *Proceedings of the 8th ACM SIGKDD (Special Interest Group on Knowledge Discovery and Data Mining) international conference on knowledge discovery and data mining, KDD '02* (pp. 133–142). New York: ACM.
- Kadziński, M., & Tervonen, T. (2013). Robust multi-criteria ranking with additive value models and holistic pair-wise preference statements. *European Journal of Operational Research*, 228(1), 169–180.
- Kadziński, M., Greco, S., & Słowiński, R. (2012a). Extreme ranking analysis in robust ordinal regression. *Omega*, 40(4), 488–501.
- Kadziński, M., Greco, S., & Słowiński, R. (2012b). Selection of a representative value function in robust multiple criteria ranking and choice. *European Journal of Operational Research*, 217(3), 541–553.
- Kadziński, M., Greco, S., & Słowiński, R. (2013a). RUTA: a framework for assessing and selecting additive value functions on the basis of rank related requirements. *Omega*, 41(4), 735–751.
- Kadziński, M., Greco, S., & Słowiński, R. (2013b). Selection of a representative value function for robust ordinal regression in group decision making. *Group Decision and Negotiation*, 22(3), 429–462.

- Kamishima, T., Kazawa, H., & Akaho, S. (2011). A survey and empirical comparison of object ranking methods. In J. Fürnkranz & E. Hüllermeier (Eds.), *Preference learning* (pp. 181–201). Berlin: Springer.
- Keeney, R. L., & Raiffa, H. (1976). *Decisions with multiple objectives: preferences and value tradeoffs*. New York: Wiley.
- Li, X., Li, F., Ji, S., Zheng, Z., Chang, Y., & Dong, A. (2009). Incorporating robustness into web ranking evaluation. In *Proceedings of the 18th ACM conference on information and knowledge management, CIKM '09* (pp. 2007–2010). New York: ACM.
- Liu, T.-Y. (2011). *Learning to rank for information retrieval*. Berlin: Springer.
- Malczewski, J. (2010). Multiple criteria decision analysis and geographic information systems. In M. Ehrgott, J. Figueira, & S. Greco (Eds.), *International series in operations research & management science: Vol. 142. Trends in multiple criteria decision analysis* (pp. 369–395). Berlin: Springer.
- Mangasarian, O. L. (1998). Generalized support vector machines. In *Advances in large margin classifiers* (pp. 135–146). Cambridge: MIT Press.
- March, J. G. (1978). Bounded rationality, ambiguity and the engineering of choice. *Bell Journal of Economics*, 9, 587–608.
- Mousseau, V., Dias, L. C., Figueira, J., Gomes, C., & Clímaco, J. N. (2003). Resolving inconsistencies among constraints on the parameters of an MCDA model. *European Journal of Operational Research*, 147(1), 72–93.
- Ogryczak, W., & Ruszczyński, A. (1999). From stochastic dominance to mean-risk models: semideviations as risk measures. *European Journal of Operational Research*, 116(1), 33–50.
- Pirlot, M., Schmitz, H., & Meyer, P. (2010). An empirical comparison of the expressiveness of the additive value function and the Choquet integral models for representing rankings. In *Proceedings of the 25th Mini-EURO conference on uncertainty and robustness in planning and decision making, URPM 2010*, Coimbra, Portugal.
- Pu, P., & Chen, L. (2008). User-involved preference elicitation for product search and recommender systems. *Artificial Intelligence Magazine*, 29(4), 93–103.
- Qin, T., Zhang, X.-D., Wang, D.-S., Liu, T.-Y., Lai, W., & Li, H. (2007). Ranking with multiple hyperplanes. In *Proceedings of the 30th international ACM SIGIR (Special Interest Group on Information Retrieval) conference on research and development in information retrieval, SIGIR '07* (pp. 279–286). New York: ACM.
- Radlinski, F., & Joachims, T. (2007). Active exploration for learning rankings from clickthrough data. In *Proceedings of the 13th ACM SIGKDD (Special Interest Group on Knowledge Discovery and Data Mining) international conference on knowledge discovery and data mining, KDD '07* (pp. 570–579). New York: ACM.
- Roy, B. (2005). Paradigms and challenges. In J. Figueira, S. Greco, & M. Ehrgott (Eds.), *Multiple criteria decision analysis: state of the art surveys* (pp. 3–26). Boston: Springer.
- Roy, B. (2009). À propos de la signification des dépendances entre critères: quelle place et quels modes de prise en compte pour l'aide à la décision? *RAIRO. Recherche Opérationnelle*, 43(3), 255–275.
- Roy, B. (2010a). Robustness in operational research and decision aiding: a multi-faceted issue. *European Journal of Operational Research*, 200(3), 629–638.
- Roy, B. (2010b). Two conceptions of decision aiding. *International Journal of Multicriteria Decision Making*, 1(1), 74–79.
- Rudin, C. (2009). The p-norm push: a simple convex ranking algorithm that concentrates at the top of the list. *Journal of Machine Learning Research*, 10, 2233–2271.
- Saaty, T. L. (2005). The analytic hierarchy and analytic network processes for the measurement of intangible criteria and for decision-making. In J. Figueira, S. Greco, & M. Ehrgott (Eds.), *Multiple criteria decision analysis: state of the art surveys* (pp. 345–408). Boston: Springer.
- Settles, B. (2012). *Active learning*. San Francisco: Morgan & Claypool.
- Siskos, Y., & Grigoroudis, E. (2010). New trends in aggregation-disaggregation approaches. In C. Zopounidis, P. M. Pardalos, P. M. Pardalos, & D. Hearn (Eds.), *Applied optimization: Vol. 103. Handbook of multicriteria analysis* (pp. 189–214). Berlin: Springer.
- Siskos, Y., Grigoroudis, E., & Matsatsinis, N. F. (2005). UTA methods. In J. Figueira, S. Greco, & M. Ehrgott (Eds.), *Multiple criteria decision analysis: state of the art surveys* (pp. 297–344). Boston: Springer.
- Srinivasan, V. (1976). Linear programming computational procedures for ordinal regression. *Journal of the ACM*, 23(3), 475–487.
- Stewart, T. (2005). Dealing with uncertainties in MCDA. In J. Figueira, S. Greco, & M. Ehrgott (Eds.), *Multiple criteria decision analysis: state of the art surveys* (pp. 445–470). Boston: Springer.
- Tehrani, A. F., Cheng, W., Dembczyński, K., & Hüllermeier, E. (2012a). Learning monotone nonlinear models using the Choquet integral. *Machine Learning*, 89, 183–211.
- Tehrani, A. F., Cheng, W., & Hüllermeier, E. (2012b). Preference learning using the Choquet integral: the case of multipartite ranking. *IEEE Transactions on Fuzzy Systems*, 20(6), 1102–1113.

- Tian, A., & Lease, M. (2011). Active learning to maximize accuracy vs. effort in interactive information retrieval. In *Proceedings of the 34th international ACM SIGIR (Special Interest Group on Information Retrieval) conference on research and development in information retrieval, SIGIR '11* (pp. 145–154). New York: ACM.
- Usunier, N., Buffoni, D., & Gallinari, P. (2009). Ranking with ordered weighted pairwise classification. In *Proceedings of the 26th international conference on machine learning, ICML '09* (pp. 1057–1064). New York: ACM.
- Viappiani, P., & Boutilier, C. (2010). Optimal Bayesian recommendation sets and myopically optimal choice query sets. In *Advances in neural information processing systems, NIPS 2010* (pp. 2352–2360).
- Vincke, Ph. (1999). Robust solutions and methods in decision-aid. *Journal of Multi-Criteria Decision Analysis*, 8(3), 181–187.
- Waegeman, W., & De Baets, B. (2011). A survey on ROC-based ordinal regression. In J. Fürnkranz & E. Hüllermeier (Eds.), *Preference learning* (pp. 127–154). Berlin: Springer.
- Waegeman, W., De Baets, B., & Boullart, L. (2009). Kernel-based learning methods for preference aggregation. *4OR*, 7(2), 169–189.
- Wallenius, J., Dyer, J. S., Fishburn, P. C., Steuer, R. E., Zionts, S., & Deb, K. (2008). Multiple criteria decision making, multiattribute utility theory: recent accomplishments and what lies ahead. *Management Science*, 54(7), 1336–1349.
- White, C., Dozono, S., & Scherer, W. (1983). An interactive procedure for aiding multiattribute alternative selection. *Omega*, 11(2), 212–214.
- Yaman, F., Walsh, T. J., Littman, M. L., & desJardins, M. (2011). Learning lexicographic preference models. In J. Fürnkranz & E. Hüllermeier (Eds.), *Preference learning* (pp. 251–272). Berlin: Springer.
- Zhu, Ch., Chen, W., Zhu, Z., Wang, G., Wang, D., & Chen, Z. (2009). A general magnitude-preserving boosting algorithm for search ranking. In *Proceedings of the 18th ACM conference on information and knowledge management, CIKM '09* (pp. 817–826). New York: ACM.
- Zopounidis, C., & Doumpos, M. (2002). Multicriteria classification and sorting methods: a literature review. *European Journal of Operational Research*, 138, 229–246.
- Zopounidis, C. & Paradas, P. M. (Eds.) (2010). *Handbook of multicriteria analysis*. Berlin: Springer.