A Unified Approach to Learn Decision Models with Interactions*

Margot Herin[1], Patrice Perny[1], Nataliya Sokolovska[2]

[1] Sorbonne University, CNRS, LIP6, Paris, France
{margot.herin,patrice.perny}@lip6.fr
[2] Sorbonne University, CNRS, LCQB, Paris, France
nataliya.sokolovska@sorbonne-universite.fr

Keywords : *Preference Learning, Multicriteria Decision Making, Interacting Criteria, Choquet Capacities, Sparsity, Iterative Reweighted Least Square Algorithm, Lagrangian Duality.*

1 Introduction

One of the main challenges of preference modeling in the context of multicriteria (or multiattribute) decision making is to construct simple and explainable decision models while keeping sufficient flexibility to accurately model human preferences and decision behaviors. The presence of possible interactions among criteria is a source of complexity for preference modeling because it prevents representing preferences by simple linear models such as weighted arithmetic means. More sophisticated weighted evaluation models including non-linear terms measuring the joint benefit or penalty attached to some groups of criteria are needed. For instance, interactions may be represented by product terms as in the multilinear utility model [7], or by minimum operations as in the Choquet integral [11].

However, allowing the possibility of interactions in a decision model is a source of complexity in preference modeling and preference learning due to the combinatorial nature of these interactions. In an aggregation model involving n criteria, interactions may appear in any of the $2^n - n - 1$ subsets of criteria including more than one element. In order to preserve scalability in learning the interactions, a standard approach is to reduce the combinatorial aspect of the problem by allowing only a limited number of them. For example, many contributions only consider pairwise interactions of criteria. More generally, a common approach consists of limiting interactions to subsets of size k for some k significantly smaller than n. However, this prior restriction eliminates simple and natural preference systems that require larger interactions. For example, including a conjunctive term such as $\min\{x_1, \ldots, x_n\}$ in the aggregation function $F(x_1, \ldots, x_n)$ may be natural to promote balanced solutions. Such an interaction involves the entire set of criteria and cannot be simply approximated by interactions on smaller sets.

In this paper, we introduce another approach where no prior restriction on the possible interaction groups is made. The useful groups will emerge from preference data with the aim of constructing a model as simple as possible, that fits well the preference examples. In this perspective, we consider a large class of decision models defined by weighted sums of non-linear factors (interaction terms) wherein we look for a sparse instance that well fits preference data.

2 Evaluation Models with Interacting Criteria

We adopt the standard setting and notations for multiattribute or multicriteria decision making. Let $N = \{1, \ldots, n\}$ be the set of viewpoints to be considered in a decision problem. Every alternative x is described by a vector $(c_1(x), \ldots, c_n(x))$ of consequences where $c_i(x)$ represents the value of x with respect to the i^{th} viewpoint. Let X_i denote the set of possible consequences

*For an extended version of this work see: Herin Margot, Patrice Perny, and Nataliya Sokolovska. "Learning Preference Models with Sparse Interactions of Criteria". In Proc. of IJCAI 2023, pp. 3786–3794.

on the i^{th} viewpoint for all $i \in N$ and $X = X_1 \times \ldots \times X_n$ the set of all possible consequence vectors. One standard approach in preference modeling consists of representing the preference relation $\succsim$ of the decision maker (DM) over X by a decomposable function u on X of the form $u(x) = F(u_1(c_1(x)), \ldots, u_n(c_n(x)))$ where $u_i : X_i \to [0,1], i \in N$ are marginal utility functions representing the attractiveness of consequences $c_i(x)$ for the DM and $F : [0,1]^n \to [0,1]$ is an aggregation function non-decreasing in each argument. Function u is said to represent $\succsim$ when $x \succsim y$ if and only if $u(x) \geq u(y)$. Let us recall two standard examples of function u, widely used to represent preferences in multicriteria decision problems involving interacting criteria:

Example 1 *The multilinear utility model [7] defined by:*

$$ML_v(x) = \sum_{S \subseteq N} v(S) \prod_{i \in S} u_i(c_i(x)) \prod_{i \notin S} (1 - u_i(c_i(x))) \tag{1}$$

Function $v : 2^N \to [0,1]$ is called a *capacity*, assigns a weight to any subset of viewpoints. One can assume that v is normalized (i.e., $v(\emptyset) = 0$ and $v(N) = 1$) and *monotonic* w.r.t. set inclusion ($v(A) \leq v(B)$ for all subsets $A, B \subseteq N$) which guarantees the monotonicity of u w.r.t. weak Pareto dominance. Another well-known capacity-based decision model is the following:

Example 2 *The discrete Choquet integral [11] defined by:*

$$C_v(x) = \sum_{i=1}^{n} [v(X_{(i)}) - v(X_{(i+1)})] u_{(i)}(c_{(i)}(x)) \tag{2}$$

where (.) is any permutation of N such that $u_{(i)}(c_{(i)}(x)) \leq u_{(i+1)}(c_{(i+1)}(x))$ and $X_{(i)} = \{(i), \ldots, (n)\}, i \in N$ with $x_{(0)} = 0$ and $X_{(n+1)} = \emptyset$. For instance, if $n = 3$ and x is such that $u_2(c_2(x)) \leq u_1(c_1(x)) \leq u_3(c_3(x))$, then $C_v(x) = [v(1,2,3) - v(1,3)]u_2(c_2(x)) + [v(1,3) - v(3)]u_1(c_1(x)) + v(3)u_3(c_3(x))$.

The capacity is a preference parameter that must be elicited by questioning the DM or learned from preference examples. The other preference parameters used in these models are marginal utility functions $u_i, i \in N$. In the case of the multilinear model, marginal utilities can be elicited from comparisons of preference intensities under *weak difference independence*, an axiom usually assumed to justify the multilinear model in multicriteria/multiattribute decision making [9, Chap. 6]. For the Choquet integral, the utility functions can be obtained using standard sequences of tradeoff queries [22, 13], or constructed with the Macbeth method [10]. From now on, we assume that the marginal utilities have been elicitated beforehand. Then, any alternative x is described by the utility vector $\mathbf{x} = (x_1, \ldots, x_n) \in [0,1]^n$ where $x_i = u_i(c_i(x)), i \in N$, and we focus on the learning of capacity v from preference examples.

Related Work As far as the identification of the capacity used in a decision model is concerned, several approaches based on the least squares criterion or variance minimization of the model under preference constraints have been proposed in the field of multicriteria analysis for the Choquet integral [10]. In the field of machine learning different learning algorithms have been proposed, e.g., Choquistic regression [19], support vector machines (SVM) with Choquet kernel [18], and ridge regression for Choquet regression [14]. Moreover a neural network was recently proposed to learn a hierarchical Choquet model [4]. Some recent contributions using regression also exist for the multilinear model [16].

Very often, a prior complexity reduction is obtained by considering models with interaction terms involving at most k criteria (e.g., k-additivity assumption [8]), $k = 2$ being the most common choice [10, 4]. A less restrictive attempt to reduce models complexity is to derive a sparse capacity representation from preference data using the L_1-norm penalty [1, 16] where the regularization was applied either to the capacity, or to the interaction index [10]. Recently, after observing that Möbius representations often lead to more compact preference models, we have proposed an approach based on linear programming to learn a sparse Möbius transform of the capacity in the Choquet integral [12]. Following this line, we present a more efficient preference learning approach which, moreover, applies to a wider class of models:

A Möbius-based general model including interactions For any capacity v, its Möbius transform m_v is another set function defined as follows: $m_v(S) = \sum_{T \subseteq S} (-1)^{|S \setminus T|} v(T), S \subseteq N$. Note that capacity v is fully characterized by m_v since by construction $v(S) = \sum_{T \subseteq S} m_v(T), S \subseteq N$. Values $m_v(S), S \subseteq N$ are called Möbius masses. Both $ML_v(x)$ and $C_v(x)$ can be directly defined from Möbius masses as follows [15, 5]:

$$ML_v(x) = \sum_{S \subseteq N} m_v(S) \prod_{i \in S} x_i, \qquad C_v(x) = \sum_{S \subseteq N} m_v(S) \min_{i \in S} \{x_i\}$$

In order to factorize and generalize the above equations, we now consider a general model F:

$$F(x) = \sum_{S \subseteq N} m_S \phi_S(x_S) \tag{3}$$

where m_S are Möbius masses and $\phi : \mathbb{R}^n \to \mathbb{R}^{2^n}$ maps x into a nonlinear feature space: $\phi(x) = (\phi_S(x_S))_{S \subseteq N}$. Note that $F(x) = \langle \mathbf{m}, \phi(\mathbf{x}) \rangle$ where $\mathbf{m} = (m_S)_{S \subseteq N}$ and $\phi(\mathbf{x})$ are indexed on the subsets $S \subseteq N$ taken in the lexicographic order. Our goal is to find a sparse instance of F (i.e., when the vector of Möbius masses includes many zeros) that well fits the available preference data. Instead of assuming that Möbius masses are null on sets larger that k (k-additivity assumption), we shall reveal the sparsity pattern from preference analysis to benefit from the full descriptive potential of F. The counterpart is that we have to handle an exponential number of Möbius masses which raises computational issues. Our approach tackles this learning problem using an optimization method based on iteratively re-weighted least squares and dualization as explained in the next section.

3 A Dual IRLS for Sparse Preference Learning

Our objective is to learn a sparse representation of $\mathbf{m}$ based on a training set of preferences statements $\{(x^i, y^i) \in \mathcal{X}^2 : x^i \succ y^i, i \in P\}$ and possibly of indifference statements $\{(x^i, y^i) \in \mathcal{X}^2 : x^i \sim y^i, i \in I\}$. A well-known workhorse for learning sparse models is the L_1-norm penalty. This is indeed a sparse-inducing penalty, in the sense that it promotes solutions with few non-null coefficients. A major application of this regularization is the LASSO linear regression [21]. Here we want to minimize both the error on the preference examples and the L_1 norm of the Möbius vector. Thus, the learning problem is formulated as follows:

$$\begin{aligned}
(\mathcal{P}) \ \min \quad & \textstyle\sum_{i \in P} \epsilon_i + \sum_{i \in I} (\epsilon_i^- + \epsilon_i^+) + \lambda \sum_{j=n+1}^{2^n} |m_j| \\
& \langle \mathbf{m}, \phi(\mathbf{x^i}) \rangle - \langle \mathbf{m}, \phi(\mathbf{y^i}) \rangle + \epsilon_i \geq \delta, \ i \in P & (4) \\
& \langle \mathbf{m}, \phi(\mathbf{x^i}) \rangle - \langle \mathbf{m}, \phi(\mathbf{y^i}) \rangle + \epsilon_i^+ - \epsilon_i^- = 0, \ i \in I & (5) \\
& \langle \mathbf{m}, \mathbf{1} \rangle = 1 & (6) \\
& \epsilon_i \geq 0, \ i \in P, \quad \epsilon_i^+, \epsilon_i^- \geq 0, \ i \in I & (7)
\end{aligned}$$

where variable m_j is the j^{th} component of vector $\mathbf{m}$. The hyper-parameter $\lambda > 0$ controls the level of regularization and δ is a strictly positive discrimination threshold used to separate preference from indifference situations. Note that the L_1-penalty is only applied to the terms involving at least two criteria so as to minimize interactions. Variable ϵ_i models the positive error made on the preference example $\mathbf{x^i} \succ \mathbf{y^i}$, while $\epsilon_i^+ - \epsilon_i^-$ models the signed error made on the indifference $\mathbf{x^i} \sim \mathbf{y^i}$. The constraint 6 guarantees $v(N) = 1$.

Problem $\mathcal{P}$ can be solved by linear programming using standard linearizations of the L_1-norm. However, the obtained linear program still drags an exponential number of variables $(2^n(3-2n) + p + 2q)$ (where $p = |P|$, and $q = |I|$) and thus is hardly solvable for more than a dozen of criteria. For the sake of scalability, we propose to solve $\mathcal{P}$ by solving a sequence of sub-problems $\mathcal{P}_k$ that admit an efficient dual formulation. More precisely, we use an iteratively reweighted least square (IRLS) algorithm [6, 3] that consists in approximating the solution of a L_1-penalized problem with a sequence of least squares problems. Sparsity is recovered by increasingly penalizing non-significant coefficients with an L_2 regularization. The interest of

this method is that a least squares problem is easy to solve in general. In our case, we will show that the least square problem $\mathcal{P}_k$ admits a compact dual form whose size is no longer exponential in n, but linear in $p+q$th, the number of preference examples.

First, using the variational formulation of the L_1-norm [2], i.e., $|x| = \frac{1}{2}\min_{z\geq 0}\frac{x^2}{z}+z$, we establish Proposition 1 providing an IRLS algorithm that approximatively solves $\mathcal{P}$. The proof relies on [3] that gives conditions under which an optimization problem can be solved by alternating minimization (here on x and z) and insights on how it leads to IRLS sequences.

Proposition 1 *Let $\eta > 0$ be a smoothing parameter. Consider the sequence $\mathbf{m}^{(k)}$ initialized with $\mathbf{m}^{(0)} = \mathbf{1}$ such that:*

$$\mathbf{m}^{(k+1)} \in \sum_{i\in P}\epsilon_i + \sum_{i\in I}(\epsilon_i^- + \epsilon_i^+) + \sum_{j>n}\frac{\lambda m_j^2}{\sqrt{m_j^{(k)2}+\eta^2}} \text{ s.t. } (4),(5),(6),(7)$$

Then we have: $\lim_{k\to\infty} J(\mathbf{m}^{(k+1)}) - J^* \leq (2^n - n)\eta$ where J is the objective function of $\mathcal{P}$ and J^* its optimum. $\mathcal{P}_k$ refers to the problem solved at each iteration.

Proposition 1 ensures that solving problems $\mathcal{P}_k$ for a sufficient number of iterations and a sufficiently small η provides a near-optimal solution to $\mathcal{P}$. The special interest of the IRLS method in our case is revealed when considering the dual formulation $\mathcal{D}_k$ of each problem $\mathcal{P}_k$. Indeed, as in kernel-based machine learning methods such as support vector machines [17, 18], one can use Lagrangian duality to obtain a more compact mathematical programming formulation. More precisely, since $\mathcal{P}_k$ is a convex problem with linear constraints, strong duality holds, and solving $\mathcal{P}_k$ or $\mathcal{D}_k$ is equivalent. The efficiency of $\mathcal{D}_k$ is detailed below:

Proposition 2 *Problem $\mathcal{D}_k$ has $p+q+1$ variables and $2(p+q)$ constraints, and is defined by:*

$$(\mathcal{D}_k) \qquad \max_{\mathbf{\Gamma}=(\boldsymbol{\alpha},\boldsymbol{\beta},\mu)\in\mathbb{R}^{p+q+1}} -\frac{1}{4\lambda}\mathbf{\Gamma}^\intercal \boldsymbol{Q}^\intercal \boldsymbol{D_k}^{-1}\boldsymbol{Q}\mathbf{\Gamma} + \mathbf{\Gamma}^\intercal \boldsymbol{L} \quad s.t. \quad \mathbf{0}\leq\boldsymbol{\alpha}\leq\mathbf{1}, \quad -\mathbf{1}\leq\boldsymbol{\beta}\leq\mathbf{1}$$

where D_k is a square diagonal matrix of size 2^n whose diagonal contains the current weighting coefficients $1/\sqrt{m_j^{(k)2}+\eta^2}$ (and 0 for the singletons). Also, $\boldsymbol{Q}$ (respectively $\boldsymbol{L}$) is a data dependent matrix of size $2^n \times (p+q+1)$ (respectively $p+q+1$) such that $\boldsymbol{Q} = (\delta_{\mathbf{P}}, \delta_{\mathbf{I}}, \mathbf{1})$, and $\mathbf{L} = (\boldsymbol{\delta}, \mathbf{0}, 1)$ where $\delta_{\mathbf{P}} = (\phi(\mathbf{x^i}) - \phi(\mathbf{y^i}))_{\mathbf{i}\in\mathbf{P}}$ and $\delta_{\mathbf{I}} = (\phi(\mathbf{x^i}) - \phi(\mathbf{y^i}))_{\mathbf{i}\in\mathbf{I}}$ are matrices of size $2^n \times p$ and $2^n \times q$ respectively and where $\boldsymbol{\delta} = \delta(1,\ldots,1) \in \mathbb{R}^p$ and $\mathbf{0} = (0,\ldots,0) \in \mathbb{R}^q$.

Towards higher dimensions. For a high number of criteria n, the computation of $\boldsymbol{Q}^\intercal\mathbf{D_k}^{-1}\mathbf{Q}$ raises an issue since $\boldsymbol{Q}$ and $\boldsymbol{D_k}$ have 2^n columns. However, for $k=1$, $\boldsymbol{D_k}$ is the identity matrix and the matrix $\boldsymbol{Q}^\intercal\boldsymbol{D_k}^{-1}\mathbf{Q} = \boldsymbol{Q}^\intercal\mathbf{Q}$ can be computed in polynomial time. In kernel-based machine learning, this property is referred to as the 'kernel trick' [17] and refers to direct computations of inner products of the form $\langle\phi(\mathbf{x}),\phi(\mathbf{x}')\rangle$ that do not require the calculation of vectors $\phi(\mathbf{x})$ (of size 2^n here). A computation in $O(n^2)$ is provided for the Choquet integral ($\phi_S(x_S) = \min(x_S)$) in [20]. We also provide a computation in $O(n)$ of the multilinear kernel:

Proposition 3 (See also [17]) *When $\phi_S(x_S) = \prod_{i\in S} x_i$, $\langle\phi(\mathbf{x}),\phi(\mathbf{x}')\rangle$ can be computed in $O(n)$,i.e., $\langle\phi(\mathbf{x}),\phi(\mathbf{x}')\rangle = \sum_{S\subseteq N}\prod_{i\in S}x_i\prod_{i\in S}x'_i = \prod_{i=1}^n(x_ix'_i+1)-1$.*

Using these polynomial computations, we proceed to a kernelized computation of $\boldsymbol{Q}^\intercal\mathbf{Q}$ at the first iteration of the IRLS sequence. This provides a way to perform dimension reduction since non-significant coefficients obtained after this first iteration can be discarded before going on.

4 Numerical Tests

In this section we present the results of numerical tests performed on synthetic preference data. We test the ability of our algorithm (denoted D-IRLS for dual IRLS) to learn a Choquet

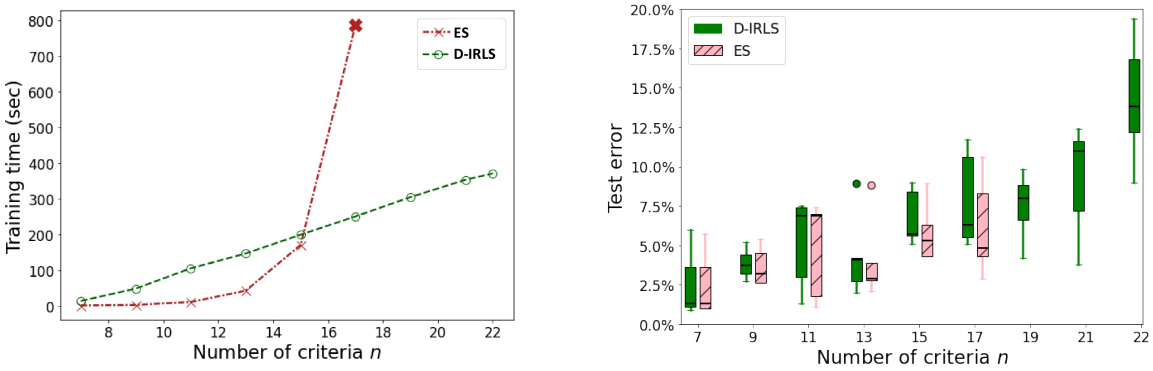

FIG. 1: Mean training time (left) and test errors (right) for D-IRLS and ES with ML_v and C_v.

integral for a growing number of criteria. We compare it to an exact solving of $\mathcal{P}$ with linear programming (denoted ES). Preference data are generated through randomly drawn sparse Möbius vectors $\mathbf{m}$ (verifying monotonicity constraints) and utilities vectors x, y are uniformly drawn within $[0,1]^n$. The overall values $u(x)$ and $u(y)$ are computed and perturbated with a Gaussian noise ($\sigma = 0.03$) before being classified as preference or indifference training examples. We set the size of the training sets to $|P| + |I| = 500$ and of the test sets to $|P| = 1000$. The regularization parameter λ is set to $\lambda = 1$. All tests are conducted on a 2.8 GHz Intel Core i7 processor with 16GB RAM and we used the mathematical programming Gurobi solver (version 9.1.2). For the D-IRLS method, the smoothing parameter is set to $\eta = 10^{-50}$ and the algorithm terminates when $\|\mathbf{m}^{(k+1)} - \mathbf{m}^{(k)}\|_2 \leq 10^{-3}$. Also, coefficients with absolute values smaller than 10^{-5} are discarded at each iteration.

Training time and generalizing performance. In the first experiment, we generate 10 training/test sets and evaluate the average training time of both algorithms as well as the generalizing performances of the learned models (average preference inversion on a test set). In order to evaluate the scalability of our method we vary the number of criteria from $n = 7$ to $n = 22$. Figure 1 shows the results for the learning of the Choquet integral. We observe that ES does not provide any solution after $n = 17$. However D-IRLS allows more than 4 millions of coefficients ($n = 22$) to be learned in less than 400 seconds. In contrast, we observe that the generalizing performances of the learned decision models obtained with D-IRLS and ES are comparable. Since the number of training preference examples is constant, the test error globally increases with the number of criteria for both methods.

Comparison with k-additive models. We compare D-IRLS to ES with k-additivity constraints for $k = 2$ (2-add) and $k = 3$ (3-add), still under the same experimental setting. The advantage of using sparse models with possible large interactions instead of k-additive models is clear: for $n = 16$, D-IRLS has an average test error of 6 %, against 17% and 23% for 2-add and 3-add respectively. In addition, for this number of criteria, 2-add and 3-add are two times slower than D-IRLS (on average about 345 sec. for 2-add and 3-add and 187 sec. for D-IRLS).

5 Conclusion

We have addressed the problem of preference learning with interacting criteria by considering a large class of capacity-based decision models including the multilinear utility and the Choquet integral, known for their expressiveness. We proposed a unified approach to learn the models of this class based on the search of sparse Möbius representations of capacities, leading to simple models with sparse interaction patterns. This approach applies to instances possibly involving more than 20 criteria and allows the most significant interaction factors to be identified within millions of possibilities. This represents a significant improvement compared to

previous approaches limited to a dozen of criteria. Moreover, the sparsity pattern is revealed from preference examples instead of resulting from a prior cardinality-based simplification of interactions, which greatly enhances the descriptive possibilities. A natural continuation of this work would be to extend the approach to learn interaction functions ϕ_S from preference data (model selection problem).

References

[1] Titilope A. Adeyeba, Derek T. Anderson, and Timothy C. Havens. Insights and characterization of l1-norm based sparsity learning of a lexicographically encoded capacity vector for the Choquet integral. In *FUZZ-IEEE*, pages 1 – 7, 2015.

[2] Francis Bach, Rodolphe Jenatton, Julien Mairal, Guillaume Obozinski, et al. Optimization with sparsity-inducing penalties. *Foundations and Trends® in Machine Learning*, 4(1):1–106, 2012.

[3] Amir Beck. On the convergence of alternating minimization for convex programming with applications to iteratively reweighted least squares and decomposition schemes. *SIAM Journal on Optimization*, 25(1):185–209, 2015.

[4] Roman Bresson, Johanne Cohen, Eyke Hüllermeier, Christophe Labreuche, and Michèle Sebag. Neural represent. and learning of hierarchical 2-additive Choquet integrals. In *IJCAI*, pages 1984–1991, 2020.

[5] Alain Chateauneuf and Jean-Yves Jaffray. Some characterizations of lower probabilities and other monotone capacities through the use of möbius inversion. *Math. Social Sciences*, 17(3):263–283, 1989.

[6] Ingrid Daubechies, Ronald DeVore, Massimo Fornasier, and C Sinan Güntürk. Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 63(1):1–38, 2010.

[7] James S Dyer and Rakesh K Sarin. Measurable multiattribute value functions. *Operations research*, 27(4):810–822, 1979.

[8] Michel Grabisch. K-order additive discrete fuzzy measures and their representation. *Fuzzy sets and systems*, 92(2):167–189, 1997.

[9] Michel Grabisch. *Set functions, games and capacities in decision making*. Springer, 2016.

[10] Michel Grabisch and Christophe Labreuche. A decade of application of the Choquet and Sugeno integrals in multi-criteria decision aid. *Annals of Operations Research*, 175(1):247–286, 2010.

[11] Michel Grabisch, Jean-Luc Marichal, Radko Mesiar, and Endre Pap. *Aggregation functions*, volume 127. Cambridge University Press, 2009.

[12] Margot Herin, Patrice Perny, and Nataliya Sokolovska. Learning sparse representations of preferences within Choquet expected utility theory. In *UAI*, pages 800–810. PMLR, 2022.

[13] Margot Herin, Patrice Perny, and Nataliya Sokolovska. Learning utilities and sparse representations of capacities for multicriteria decision making with the bipolar Choquet integral. In *Multidisciplinary Workshop on Advances in Preference Handling, IJCAI*, 2022.

[14] Siva K Kakula, Anthony J Pinar, Timothy C Havens, and Derek T Anderson. Choquet integral ridge regression. In *2020 IEEE International Conference on Fuzzy Systems*, pages 1–8. IEEE, 2020.

[15] Guillermo Owen. Multilinear extensions and the banzhaf value. *Naval research logistics quarterly*, 22(4):741–750, 1975.

[16] Guilherme Dean Pelegrina, Leonardo Tomazeli Duarte, Michel Grabisch, and João Marcos Travassos Romano. The multilinear model in multicriteria decision making: The case of 2-additive capacities and contributions to parameter identification. *European J. of Operational Research*, 282(3):945–956, 2020.

[17] John Shawe-Taylor and Nello Cristianini. *Kernel methods for pattern analysis*. Cambridge University Press, 2004.

[18] Ali Fallah Tehrani. The Choquet kernel on the use of reg. problem. *Inform. Sciences*, 556:256–272, 2021.

[19] Ali Fallah Tehrani and Eyke Hülermeier. Ordinal Choquistic reg. In *EUSFLAT*, pages 842–849, 2013.

[20] Ali Fallah Tehrani, Marc Strickert, and Eyke Hüllermeier. The Choquet kernel for monotone data. In *ESANN*, pages 337–342, 2014.

[21] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (methodological)*, 58(1):267 – 88, 1996.

[22] Peter Wakker and Daniel Deneffe. Eliciting von neumann-morgenstern utilities when probabilities are distorted or unknown. *Management Science*, 42(8):1131–1150, 1996.