A Dual Approach for Learning Sparse Representations of Choquet Integrals

Margot Herin[1] and **Patrice Perny**[1] and **Nataliya Sokolovska**[2]
[1]**LIP6 - Sorbonne University** [2]**LCQB - Sorbonne University**
{margot.herin,patrice.perny}@lip6.fr, nataliya.sokolovska@sorbonne-universite.fr

Abstract. Learning simple and explainable preference models able to approximate human preferences is one of the key challenges of multicriteria decision support. In this paper we consider the Choquet integral as a general preference aggregation function and we study the problem of learning an instance, as simple as possible, of this general model to fit a database of preference examples. More precisely, we want to learn a sparse Möbius representation of the capacity defining the Choquet integral. To this end, we study an approach to sparse preference learning based on iterative re-weighted L_2 regularization and dualization. We show how to implement this approach and we share the results of numerical tests performed on synthetic preference data. We discuss the benefit of this approach compared to L_1 regularization.

1 Introduction

Multicriteria Decision Making consists in identifying an 'optimal' tradeoff among multiple alternatives evaluated with respect to various conflicting points of views. The notion of Pareto-optimality is often seen as a natural and objective prerequisite in the multicriteria setting. It restricts the investigation of possible choices to those solutions that cannot be improved on one criterion without being downgraded on another criterion. However it is well know that Pareto-optimality leaves multiple solutions incomparable. Obtaining a richer preference information is necessary to be able to discriminate between Pareto-optimal solutions and determine a tradeoff that best fits to the value system of the Decision Maker (DM).

In order to go beyond Pareto-optimality, families of aggregation functions have been put forward for their ability to model complex preference behaviors and to generate relevant tradeoffs through optimization [16]. Among them, the Choquet Integral (CI) is acknowledged as one of the most expressive decision models used for the aggregation of criteria. It includes various simpler models as special cases such as weighted means and ordered weighted averages [33, 31]. It makes it possible to model criteria interactions and to perform non-linear aggregations using weights assigned to every subset of criteria [12]. The function assigning a weight to every subset is named a *Choquet capacity*. It provides a high flexibility to model various aggregation logics and fits to various preferences systems. However, the definition of this capacity requires a number of weighting coefficients exponential in the number of criteria. Thus the learning of these weights from DM preference examples is computationally hard and prone to over-fitting. The risk of over-fitting is even more important that in practice preference data is limited and may be costly to obtain (preference queries must be asked to the DM or derived from a history of previous decisions).

For this reason, there is some interest in deriving sparse representations of CI parameters that best fit the available preference examples. Sparsity refers to the idea of minimizing the number of non-null parameters used to represent CI weights. More precisely we look for a tradeoff between minimizing the error on the preference examples and the number of non-null parameters i.e., the L_0 norm. However L_0-optimization is known to be NP-hard [23].

To obtain sparse representations of models and bypass the computational issue of L_0-optimization, a common approach consists in relaxing L_0 into L_1. Contrarily to L_0 regularization, the regularization term involving L_1 norm can be easily linearized without requiring the insertion of binary variables. In the case of linear regression, the LASSO is a standard method that relies on the L_1 norm penalization [30, 17]. In the case of Choquet regression, several attempts to derive sparse representation of CI [18, 19] have been made using the L_1 penalty [2, 1, 24, 18, 19]. The proposed methods consist in solving quadratic or linear programs that involve the exponential number of parameters defining the CI and thus are not scalable. Also, L_1-regularization is known to proceed to consistent parameter selection at the condition that some restrictive assumptions are verified [34]. In a nutshell, it may be biased because of statistical correlation between criteria (features), especially in high dimension.

An alternative approach consists of approximating L_0-regularized solutions using iterative reweighting schemes involving L_2-regularization. The idea is to solve a sequence of convex optimization problems using weighted L_2-norm penalizations that increasingly penalize small coefficients to push them to zero. In machine learning it has been applied in the context of ridge regression [11] and support vector machines (SVM) [22] where loss functions already include L_2 regularization. In both cases a sequence of weighted L_2-penalized versions of the models are used as surrogate to L_0-regularization. It was reported that the substitution of L_0-regularization by iterative re-weighted L_2 regularization provides very good results in practice. Both ridge regression and SVM problems are easy to solve with L_2 regularization (weighted or not), especially in the high dimension case (when the number of training examples is very small in front of the number of features). Indeed they consist in quadratic convex optimization problems that admit a very efficient dual formulation, the complexity of which only depends on the number of training examples. Then iterative re-weighted L_2-regularization benefits from this efficiency at each iteration. Set aside sparsity, the efficiency of the dual form of the SVM has been already leveraged for the learning of Choquet Integral for regression and classification in [26, 29].

In this paper, we propose to adapt this iterative scheme to the

learning of a sparse CI representation from preference examples and we test its practical efficiency compared to previous works based on L_1 regularization.

Related work. In Multicriteria decision analysis, the identification of the CI parameters from examples of preferences over alternatives has been performed by either minimizing the least squared error (total violation of preferences), either maximising the separation of the alternatives or minimizing the variance of the model under preference constraints [14, 15]. In machine learning, CI parameters are learned for binary classification in the context of logistic regression in [27]. CI Regression is also performed in the context of SVM and ridge regression [26, 20]. Also a neural architecture has been proposed to learn the capacity in hierarchical aggregation models based on Choquet integrals [8].

Another approach to preference learning developed in AI consists in progressively specifying the capacity until a necessary winner can be identified. With this incremental approach, the aim is merely to find the best alternative in a given set rather than builiding a decision model representing the preferences of the DM. A first set of methods proceeds by successive reductions of the parameter space using preference queries adaptively selected for their information value (e.g., using the minimax regret criterion). This incremental approach was used for the identification of Choquet capacities in [6]. A second set of methods proposes another adaptive elicitation procedure based on a Bayesian approach used to iteratively revise a probability density on the parameter space, see e.g [7]. The advantage of the incremental approach is that it focuses the elicitation burden to the identification of an optimal tradeoff. On the other hand it does not provide a precise specification of the preference model and this undetermination may be a cause of indecision when facing new alternatives.

It must be emphasized that none of the above mentioned approaches addresses the question of learning sparse representations of the weighting parameters of a Choquet integral. However a prior model complexity reduction is often obtained by considering k-additive models i.e., by only authorizing interactions between coalitions of criteria of cardinality lower than k [13] for some k strictly lower than the number n of criteria. Such cardinal-based sparsity patterns do not allow a fine adaptation of the model complexity to the data. For instance, evidences for a loss of expressiveness with k-additive models ($k < n$) are provided through toy examples and experiments on synthetic data in [19]. Alternative concepts like k-maxitivity and k-interactivity have also been introduced to adapt the complexity of the model to preference data more efficiently [3, 4]. However one can easily find instances of CI admitting very sparse representations that do not satisfy such restrictions.

For these reasons we propose here a method that looks for a sparse representation of a Choquet integral where the sparsity pattern is derived from preference data and not arbitrarily chosen. In this perspective we investigate a regularization scheme based on L_2 rather than L_0 or L_1 for the sake of scalability.

The paper is organized as follows. In Section 2 we introduce preliminary background and notations related to the use of CI in preference modeling. In section 3 we introduce our dual approach to learn sparse representation of CI. Numerical results are presented and discussed in Section 4. Concluding remarks close the paper.

2 Background and Notations

We adopt the standard setting and notations for multicriteria decision making. Let $N = \{1, \ldots, n\}$ be the set of the n points of view to be considered in a decision problem. Let X be a set of alternatives of the form $x = (x_1, \ldots, x_n)$ where x_i represents the utility of x on criterion i for $i = 1, \ldots, n$ (utilities have been elicited beforehand [19] and x_i are commensurate).

Let us recall the definition of the Choquet Integral. Let v denote a capacity defined on 2^N, i.e., a set function such that $v(\emptyset) = 0$, $v(N) = 1$ and $v(A) \leq v(B)$ for all $A, B \subseteq N$ such that $A \subseteq B$. The CI defines the value of any consequence vector $x = (x_1, \ldots, x_n)$ as follows:

$$C_v(x) = \sum_{i=1}^{n} \big[v(X_{(i)}) - v(X_{(i+1)})\big] x_{(i)} \quad (1)$$

$$= \sum_{i=1}^{n} \big[x_{(i)} - x_{(i-1)}\big] v(X_{(i)}) \quad (2)$$

where (.) is any permutation of N such that $x_{(i)} \leq x_{(i+1)}$ and $X_{(i)} = \{(i), \ldots, (n)\}, i \in N$ with $x_{(0)} = 0$ and $X_{(n+1)} = \emptyset$.

Example 1. *If* $N = \{1, 2, 3\}$ *and* $x_1 \leq x_3 \leq x_2$ *then* $C_v((x_1, x_2, x_3)) = x_1 v(\{1, 2, 3\}) + [x_3 - x_1] v(\{2, 3\}) + [x_2 - x_3] v(\{2\})$ *by Equation 2.*

Then the preferences induced by CI on X are obviously defined as follows: for any solutions $x, y \in X$, x is at least as good as y (denoted $x \succsim y$) if and only if $C_v(x) \geq C_v(y)$. Similarly, x is indifferent to y (denoted $x \sim y$) if and only if $C_v(x) = C_v(y)$. Note that the monotonicity of v w.r.t. set inclusion implies the monotonicity of $\succsim$ (the preference induced by C_v) with respect to weak Pareto dominance. More formally, we have:

$$(\forall i \in N, x_i \geq y_i) \Rightarrow C_v(x) \geq C_v(y) \quad (3)$$

Interestingly CI also reads as a linear combination of the minimum of the consequences taken over every possible criteria coalition, as shown in [9]:

$$C_v(x) = \textstyle\sum_{A \subseteq N} m_v(A) \min_{i \in A}\{x_i\} \quad (4)$$

The weights $m_v(A)$ for all $A \subseteq N$ are called *Möbius masses* and defined by:

$$m_v(A) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} v(B) \quad (5)$$

Möbius masses completely characterize the capacity v that can be recovered, for any $A \subseteq N$ by:

$$v(A) = \sum_{B \subseteq A} m_v(B) \quad (6)$$

Note that Equation 4 allows to see CI as an inner product:

$$C_v(x) = \langle \mathbf{m}, \phi(\mathbf{x}) \rangle \quad (7)$$

where $\mathbf{m} = (m_v(A))_{A \subseteq N \setminus \emptyset}$ and $\phi : \mathbb{R}^n \to \mathbb{R}^{2^n - 1}$ maps x into a nonlinear feature space: $\phi(x) = (\min_{i \in A}\{x_i\})_{A \subseteq N \setminus \emptyset}$. Both $\mathbf{m}$ and $\phi(x)$ are constructed with $A \subseteq N \setminus \emptyset$, taken in the lexicographic order.

The reformulation of CI using Möbius masses suggests that $C_v(x)$ might admit a compact representation when m_v is sparse (i.e., when the Möbius masses include many zeros or values that will not significantly impact the calculation). Let us give an example:

Example 2. *Let us consider the set $X = \{a, b, c, d, e, f\}$ of alternatives characterized by the following evaluation with respect to 4 criteria ($N = \{1, 2, 3, 4\}$):*

x	x_1	x_2	x_3	x_4
a	12	12	12	12
b	10	10	10	20
c	10	10	20	10
d	10	20	10	10
e	20	10	10	10
f	10	12	10	10

Assume that the DM value system is well described by a Choquet integral C_v with $v(\emptyset) = 0, v(\{1, 2, 3, 4\}) = 1$ and the following values:

A	$\{1\}$	$\{2\}$	$\{3\}$	$\{4\}$	$\{1, 2\}$	$\{1, 3\}$	$\{1, 4\}$
$v(A)$	0.04	0.08	0.12	0.16	0.12	0.16	0.20
A	$\{2, 3\}$	$\{2, 4\}$	$\{3, 4\}$	$\{1, 2, 3\}$	$\{1, 2, 4\}$	$\{1, 3, 4\}$	$\{2, 3, 4\}$
$v(A)$	0.20	0.24	0.28	0.24	0.28	0.32	0.36

If we compute the values C_v of the alternatives, we obtain the following preference order: $a \succ b \succ c \succ d \succ e \succ f$. We remark that this order cannot be represented by a weighted sum of criterion values (because a is the preferred alternative while belonging to the interior of the convex hull of b, c, d, e).

Although v is not sparse, it admits a sparse Möbius inverse which is everywhere zero except on singletons and N. More precisely we have $m_v(\{1\}) = 0.04, m_v(\{2\}) = 0.08, m_v(\{3\}) = 0.12, m_v(\{4\}) = 0.16$ and $m_v(N) = 0.6$. Hence, Equation 4 yields the following simplified expression of the Choquet integral:

$$\begin{aligned} C_v(x) \quad = \quad & 0.04x_1 + 0.08x_2 + 0.12x_3 + 0.16x_4 \\ & +0.60 \ \min\{x_1, x_2, x_3, x_4\} \end{aligned}$$

This formulation is easier to interpret. The main part of the total Möbius mass is put on $N = \{1, 2, 3, 4\}$ which supports solutions having a high minimum (i.e., no weakness). Moreover, this minimum is refined by a weighted sum assigning increasing importance to criterion 4, 3, 2 and 1. It is worth noting that the sparse representation of CI given above is not k-additive whatever $k < 4$. The main factor in the Möbius representation is of size 4 which suggests that k-additive approximations of v are unlikely to provide good approximations of overall utilities and preferences. This example illustrates that enforcing k-additivity for some $k < n$ might not be the best way to obtain relevant sparse representations of preferences.

Moreover, as suggested by this example, looking for a sparse Möbius representation m_v of v might be more natural than expecting sparsity of v. This can easily be explained by the fact that $\|v\|_0 \geq \|m_v\|_0$ as shown in [18]. Moreover, due to the monotonicity constraint, if $v(\{i\}) > 0$ for some i, then $v(A) > 0$ for any subset $A \supset \{i\}$ and thus v is unlikely to be sparse. This explains our focus on Möbius masses to obtain sparse representations of capacities. In the next section, we present an approach to learn a sparse Möbius representation of a capacity able to approximate preference data using the CI model without making any prior structural assumptions on the sparsity pattern.

3 A Dual Approach to Sparse CI Learning

We want to learn a sparse representation of $\mathbf{m}$ based on a training set of p preference examples $\mathcal{D} = (x^i, y^i)_{i=1}^p$ where in each example $x^i \succ y^i$, $i = 1 \dots p$. A natural formulation of this problem is the following optimization problem:

$$\begin{aligned} (\mathcal{P}) \quad \min \ & C \sum_{i=1}^{p} \epsilon_i + \|\mathbf{m}\|_0 \\ & \langle \mathbf{m}, \phi(\mathbf{x^i}) \rangle - \langle \mathbf{m}, \phi(\mathbf{y^i}) \rangle + \epsilon_i \geq \gamma, \ i = 1...p \\ & \epsilon_i \geq 0, \ i = 1...p \end{aligned}$$

where $C > 0$ is an hyper-parameter that controls the level of regularization and γ is a strictly positive discrimination threshold used to separate preference from indifference situations. The variable ϵ_i models the error made on the preference example $\mathbf{x^i} \succ \mathbf{y^i}$. Note that γ can arbitrarily be set to 1 without loss of generality. Taking another value of γ would only multiply the optimal solution $\mathbf{m}^*$ by γ and thus imply the same order over alternatives. In order to satisfy the boundary condition of a capacity ($v(N) = 1$) $\mathbf{m}^*$ is normalized to satisfy $\sum_{A \subseteq N} m_v(A) = 1$.

L_0-minimization being NP-hard one wants to avoid solving $\mathcal{P}$ directly. As proposed in the context of sparse support vector machine for binary classification in [22], we propose instead to solve the following iterative problem:

$$\mathbf{m}^{(k+1)} \leftarrow sol(\mathcal{P}_k) \tag{8}$$

$$\begin{aligned} \text{with} \quad (\mathcal{P}_k) \quad \min \ & C \sum \epsilon_i + \sum_{A \subseteq N} \frac{1}{m_A^{(k)_2}} m_A^2 \\ & \langle \mathbf{m}, \phi(\mathbf{x^i}) \rangle - \langle \mathbf{m}, \phi(\mathbf{y^i}) \rangle + \epsilon_i \geq 1, \ i = 1...p \quad (9) \\ & \epsilon_i \geq 0, \ i = 1...p \quad (10) \end{aligned}$$

where m_A is the variable representing the quantity $m_v(A)$ for all $A \subseteq N$. Remark that $\sum_{A \subseteq N} \frac{1}{m_A^{(k)_2}} m_A^2 = \mathbf{m}\mathbf{D}_k\mathbf{m}^T$ where $\mathbf{D}_k$ is a diagonal matrix whose diagonal values are the reciprocals of the squared values of the vector $\mathbf{m}^{(k)}$.

At each iteration, one solves a variant of $\mathcal{P}$ with a weighted L_2 penalization. The weights are defined as the reciprocals of the squared Möbius masses obtained at the previous iteration. Then from step to step, small Möbius masses are increasingly penalized and are finally discarded from the optimal solution. After convergence the optimal solution is sparse. For a proof of convergence of this type of iterative scheme the reader may consult [32, 22].

L_2-penalization is convex contrarily to L_0-penalization which simplifies the optimization. Also, as in kernel-based machine learning methods such as support vector machine [25], one can use Lagrangian duality to obtain a more compact mathematical programming formulation. Indeed, since $\mathcal{P}_k$ is a convex problem with linear constraints, strong duality holds and there is no duality gap. To compute the dual problem we write the Lagrangian function using the Lagrange multipliers α_i for the preferences constraints (Eq.(9)), and μ_i for the sign constraints (Eq.(10)), $i = 1, \dots, p$:

$$\begin{aligned} \mathcal{L}(\mathbf{m}, \epsilon, \alpha, \mu) = C \sum_{i=1}^{p} \epsilon_i + \mathbf{m}\mathbf{D}_k\mathbf{m}^T + \sum_{i=1}^{p} \alpha_i(-\mathbf{P}_i^T\mathbf{m} - \epsilon_i + 1) \\ - \sum_{i=1}^{p} \mu_i \epsilon_i \end{aligned} \tag{11}$$

where $\mathbf{P}_i = \phi(x^i) - \phi(y^i), \quad i = 1 \dots p$. The stationarity Karush–Kuhn–Tucker gives:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{m}} = \mathbf{m} - \sum_{i=1}^{N} \alpha_i \mathbf{D}_k^{-1} \mathbf{P}_i = 0 \tag{12}$$

$$\frac{\partial \mathcal{L}}{\partial \epsilon_i} = C - \alpha_i - \mu_i = 0 \tag{13}$$

Then one obtains the dual problem $\mathcal{D}_k$ by re-introducing these equations in Eq. (11):

$$(\mathcal{D}_k) \quad \max_{\alpha\in\mathbb{R}^p} -\frac{1}{2}\sum_{i=1}^{p}\sum_{j=1}^{p}\alpha_i\alpha_j\mathbf{P}_i\mathbf{D}_k^{-1}\mathbf{P}_j^T + \sum_{i=1}^{p}\alpha_i$$
$$0 \leq \alpha \leq C$$

Problem $\mathcal{D}_k$ is convex and quadratic with p variables and p constraints. This provides a much more scalable approach since the size of $\mathcal{D}_k$ only depends on the number of preference examples p. This allows to address multicriteria problems involving a significantly larger number of criteria n.

Remark 1. *This type of dualization is also used in kernel-based machine learning methods such as SVM. Here the kernel is embodied by the matrix $\mathbf{P}\mathbf{D}_k^{-1}\mathbf{P}^T$ where $P \in \mathbb{R}^{p\times 2^n}$ is the matrix the rows of which are equal to $\mathbf{P}_i$. Note that in this case exploiting kernelization is difficult since the weighting matrix $\mathbf{D}_k$ must be recalculated from the primal solution at each iteration. Then it prevents us frome taking advantage of the efficient calculus of the Choquet kernel proposed in [28]. This direction is left for further investigation.*

At every step k, the solution of the primal problem $\mathcal{P}_k$ is recovered using the Karush–Kuhn–Tucker condition Eq. (12):

$$\mathbf{m}^{(k+1)} = \sum_{i=1}^{p}\alpha_i^{(k+1)}\mathbf{D}_k^{-1}\mathbf{P}_i$$

Where $\alpha^{(k+1)} = (\alpha_1^{(k+1)}, \ldots, \alpha_p^{(k+1)})$ is the solution of $\mathcal{D}_k$.

We summarize the proposed method in the following algorithm:

Algorithm 1: Iterative weighted L_2-penalized method

inputs : $\mathbf{P}_i, i = 1\ldots p, C, \epsilon$
initialization: $k \leftarrow 0, \mathbf{m}^{(0)} \leftarrow (1, \ldots 1)$,
$\mathbf{D}_0 \leftarrow diag(\mathbf{m}^{(0)} \odot \mathbf{m}^{(0)})^{-1}$
repeat
  $\boldsymbol{\alpha^{(k+1)}} \leftarrow$ solution of $\boldsymbol{(\mathcal{D}_k)}$
  $\mathbf{m}^{(k+1)} \leftarrow \sum_{i=1}^{p}\boldsymbol{\alpha_i^{(k)}}\mathbf{D}_k^{-1}\mathbf{P}_i$
  $\mathbf{D}_{k+1} \leftarrow \boldsymbol{diag}(\mathbf{m}^{(k+1)} \odot \mathbf{m}^{(k+1)})^{-1}$
  $\boldsymbol{k \leftarrow k+1}$
until $\|\mathbf{m}^{(k+1)} - \mathbf{m}^{(k)}\| \leq \epsilon$;
output : $\mathbf{m}^{(k+1)}$

where $\odot$ denotes the elementary product and $diag(v)$ is the diagonal matrix whose diagonal is vector v.

The above presented approach does not explicitly require monotonicity of the learned set function. We may expect the learned set function to be almost monotonic when preference data are monotonic w.r.t weak Pareto dominance. This observation has been empirically confirmed in similar learning contexts (see e.g., [26]). Nevertheless when monotonicity is strictly required (for normative reasons) we need to explicitly insert constraints enforcing the monotonicity of the capacity in the learning problem. This question is discussed below.

Enforcing monotonicity. Incorporating monotonicity constraints in $\mathcal{P}$ modifies problems $\mathcal{D}_k$ by adding a number of variables exponential in n. We then lose the efficiency of the dualization. Consequently we propose to use the primal problems $\mathcal{P}_k$ in which the monotonicity constraints are integrated. At each iteration we thus solve the following problem:

$$(\mathcal{P}_k^C) \quad \min \; C\sum\epsilon_i + \sum_{A\subseteq N}\frac{1}{m_A^{(k)2}}m_A^2$$
$$\langle\mathbf{m},\phi(\mathbf{x^i})\rangle - \langle\mathbf{m},\phi(\mathbf{y^i})\rangle + \epsilon_i \geq 1,\ i = 1\ldots p$$
$$\epsilon_i \geq 0,\ i = 1\ldots p$$
$$\sum_{B\subseteq A, i\ni B} m_B \geq 0,\ \forall i \in A, \forall A \subseteq N$$

The last constraints are the monotonicity constraints expressed in terms of the Möbius masses. They are equivalent to $v(A \setminus \{i\}) \leq v(A)$, for all $A \subseteq N$ and $i \in A$. The proposed method is summarized in the following algorithm:

Algorithm 2: Constrained iterative weighted L_2-penalized method

inputs : $\mathbf{P}_i, i = 1, \ldots, p, C, \epsilon$
initialization: $k \leftarrow 0, \mathbf{m}^{(0)} \leftarrow (1, \ldots 1)$,
$\mathbf{D}_0 \leftarrow diag(\mathbf{m}^{(0)} \odot \mathbf{m}^{(0)})^{-1}$
repeat
  $\mathbf{m}^{(k+1)} \leftarrow$ solution of $(\mathcal{P}_k^C)$
  $\mathbf{D}_{k+1} \leftarrow diag(\mathbf{m}^{(k+1)} \odot \mathbf{m}^{(k+1)})^{-1}$
  $k \leftarrow k+1$
until $\|\mathbf{m}^{(k+1)} - \mathbf{m}^{(k)}\| \leq \epsilon$;
output : $\mathbf{m}^{(k+1)}$

4 Numerical Tests

In this section we conduct experiments on synthetic data and investigate empirically the advantages of our iterative re-weighted L_2-penalized method over simple L_1 and L_2 regularizations.

The L_1-regularized solution is obtained by solving the following linear program:

$$(\mathcal{P}_{L_1}) \quad \min \; C\sum_{i=1}^{p}\epsilon_i + \sum_{A\subseteq N}(a_A + b_A)$$
$$\langle\mathbf{m},\phi(\mathbf{x^i})\rangle - \langle\mathbf{m},\phi(\mathbf{y^i})\rangle + \epsilon_i \geq \gamma,\ i = 1\ldots p$$
$$\epsilon_i \geq 0,\ i = 1\ldots p$$
$$m_A = a_A - b_A,\ A \subseteq N$$

The L_2-regularized solution is obtained by solving the quadratic problem :

$$(\mathcal{P}_{L_2}) \quad \min \; C\sum_{i=1}^{p}\epsilon_i + \|\mathbf{m}\|_2^2$$
$$\langle\mathbf{m},\phi(\mathbf{x^i})\rangle - \langle\mathbf{m},\phi(\mathbf{y^i})\rangle + \epsilon_i \geq \gamma,\ i = 1\ldots p$$
$$\epsilon_i \geq 0,\ i = 1\ldots p$$

Actually this problem also benefits from dualization, and we rather solve the dual problem:

$$(\mathcal{D}_{L_2}) \quad \max_{\alpha\in\mathbb{R}^p} -\frac{1}{2}\sum_{i=1}^{p}\sum_{j=1}^{p}\alpha_i\alpha_j\mathbf{P}_i\mathbf{P}_j^T + \sum_{i=1}^{p}\alpha_i$$
$$0 \leq \alpha \leq C$$

The primal solution is recovered with the KKT condition:

$$\mathbf{m}^{(k+1)} = \sum_{i=1}^{p}\alpha_i^{(k)}\mathbf{P}_i \tag{14}$$

Remark that it consists in the first iteration of the iterative weighted L_2-penalized approach.

The synthetic data used in the tests are generated as follows. First, random sparse Choquet models are obtained from randomly generated sparse vectors $\mathbf{m}$ verifying monotonicity. The generation of monotonic capacity is a difficult problem [10, 5], we used here a naive algorithm. In a first stage the number of non-null Möbius masses is chosen and their values are randomly drawn regardless monotonicity. Then $\mathbf{m}$ is constructed as the closest normalized monotonic vector (in the sense of the L_1 norm) using linear programming. Second, pairs (x^i, y^i) of alternatives are uniformly drawn within $[0,1]^n \times [0,1]^n$. For every pair, if $(\langle \mathbf{m}, \phi(\mathbf{x^i})\rangle + \sigma_{x_i}) - (\langle \mathbf{m}, \phi(\mathbf{y^i})\rangle + \sigma_{y_i}) > 0$, where $\sigma_{x_i}, \sigma_{y_i}$ are noise values uniformly taken within an interval $[-\sigma, \sigma]$, then the pair (x^i, y^i) is inserted in a preference examples data-set. Pairs with Pareto dominance are discarded. This process is repeated to generate training sets of size p which we vary in our experiments. We also generate test sets of 1000 preference examples. In the following, the generalizing performance of the models is evaluated as the percentage of preference inversion in a test set (test error). When not specified, the regularization parameter C is chosen by cross-validation with a number of folds equal to 3.

We first illustrate the ability of the iterative re-weighted L_2-penalization method to recover a sparse Choquet model. We generate $p = 200$ preferences examples on alternatives of dimension $n = 8$ with an arbitrarily sparse Choquet model. On Figure 1 are represented from top to bottom the ground truth model, the estimated model with L_1 and L_2 regularization and with the iterative re-weighted L_2-penalization. Vertical dotted lines highlight non-null ground truth Möbius masses indices. We can see that the solution obtained with L_2 regularization is clearly over-fitted while the iterative version provides a more sparse model than the model obtained with L_1 regularization. The estimated model with iterative re-weighted L_2-penalization is almost equal to the ground truth model.

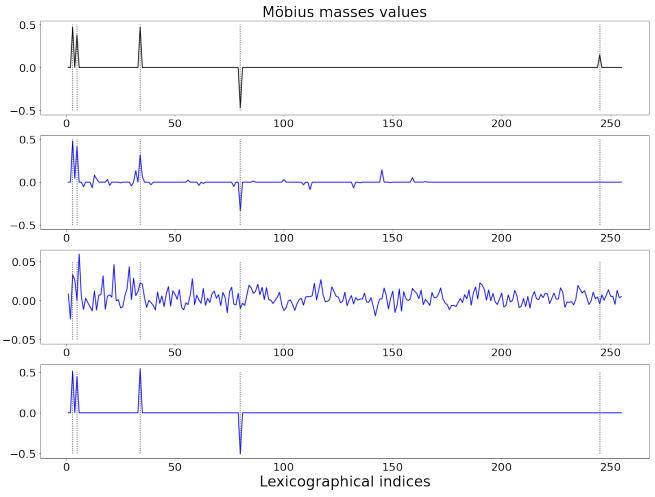

Figure 1. From top to bottom, ground truth and estimated models with L_1 and L_2 regularizations and the iterative weighted L_2-penalization ($n = 8, p = 200$).

For the same instance than Figure 1, we represent on Figure 2 the evolution of the learned Möbius coefficients as the regularization hyper-parameter C decreases. The trace of ground truth non-null coefficients are marked with stars. The Möbius masses vector is normalized for each value of C. We can see that by decreasing C the learned representation focuses on the ground truth non-null coefficients.

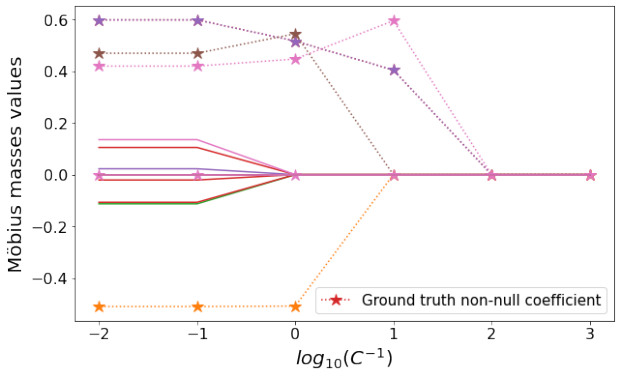

Figure 2. Regularization path: estimated Möbius masses w.r.t C for our approach.

Then we conduct a larger experiment with synthetic data associated to an increasing number of criteria for n ranging from $n = 4$ to $n = 20$. For each value of n, synthetic data of size $p = 500$ are generated 10 times and models are trained on the data and compared in terms of training time, test error and L_0 norm. Note that when $p > 2^n$ we solve the L_2-penalized problem and the iterative re-weighted L_2-penalized problem in the primal. In the tables of results the iterative re-weighted L_2-penalized method is denoted L_2^*.

In Table 1 we can see that L_2 regularization outperforms all the methods in terms of training time. For $n = 20$, this method achieves the estimation of $2^n = 1048576$ coefficients in 18.21 seconds. Again, this is possible because of the dualization that provides a problem whose number of variables and constraints equals to p and not to 2^n. Our approach benefits from this at each iteration and thus is significantly more efficient than the L_1-penalization method for n greater than 16. It reaches a result in 450.9 seconds in average for $n = 20$ while the L_1 method does not reach an optimal solution within 3 hours. Beyond 20 criteria the computation of the vectors $\phi(x^i)$ goes beyond the capacity of our computer. The tests are processed with the Gurobi 9.1 solver on a computer with 16GB of RAM and a Intel(R) Core(TM) i7-1165G7 @ 2.80GHz processor.

	Methods		
n	L_1	L_2	L_2^*
4	0.34 ± 0.03	$\mathbf{0.31 \pm 0.03}$	4.31 ± 1.40
8	5.06 ± 0.75	$\mathbf{4.85 \pm 0.67}$	112.41 ± 26.60
12	75.10 ± 3.39	$\mathbf{15.40 \pm 1.67}$	364.27 ± 69.99
16	1034.98 ± 96.34	$\mathbf{12.31 \pm 1.55}$	285.09 ± 34.05
20	$> 3h$	$\mathbf{18.21 \pm 1.22}$	450.90 ± 62.54

Table 1. Average training time for 10 simulations.

In Table 2 are presented the averaged L_0 norm of the different solutions. As expected, L_2 regularization provides dense models while L_1 regularization provide sparse models. However, our approach reaches better results in terms of compacity. In Table 3 we can see that this strong coefficient selection is consistent since our approach

reaches better or very similar generalizing performance results than the two other methods.

	Methods		
n	L_1	L_2	L_2^*
4	10.32 ± 2.28	15.00 ± 0.00	$\mathbf{5.20 \pm 1.5}$
8	52.50 ± 6.25	255.00 ± 0.00	$\mathbf{10.82 \pm 3.74}$
12	85.80 ± 8.00	4092.70 ± 1.95	$\mathbf{17.81 \pm 3.87}$
16	147.52 ± 5.32	64101.00 ± 122.33	$\mathbf{19.51 \pm 2.65}$
20	$- - - - -$	998481.72 ± 554.54	$\mathbf{43.20 \pm 5.82}$

Table 2. Average L_0 norm for 10 simulations.

	Methods		
n	L_1	L_2	L_2^*
4	$\mathbf{1.95 \pm 0.60}\%$	$2.56 \pm 0.79\%$	$2.05 \pm 0.95\%$
8	$4.97 \pm 0.61\%$	$6.17 \pm 0.67\%$	$\mathbf{3.86 \pm 0.74}\%$
12	$6.22 \pm 1.52\%$	$9.70 \pm 1.30\%$	$\mathbf{5.55 \pm 1.74}\%$
16	$9.45 \pm 1.23\%$	$9.95 \pm 0.72\%$	$\mathbf{8.92 \pm 0.81}\%$
20	$- - - - -$	$\mathbf{16.33 \pm 1.72}\%$	$16.91 \pm 1.90\%$

Table 3. Average Test Error for 10 simulations.

In the last experiment we compare the iterative re-weighted L_2-penalized method with its constrained version. In order to investigate the sole effect of the monotonicity constraints we solve all the problems in their primal form for both methods. We generate synthetic data associated to an increasing number of criteria from $n = 4$ to $n = 12$. For each value of n, synthetic data of size $p = 200$ is generated 10 times. We compare both approaches in terms of training time, test error and monotonicity degree of the models learned on the data. The monotonicity degree of a model is defined as the percentage of verified monotonicity constraints (there is one constraint for each $A \subseteq N$ and each $i \in A$). Results are presented in Table 4,Table 5 and Table 6.

n	Constrained L_2^*	Unconstrained L_2^*
4	1.2 ± 0.5	1.4 ± 0.5
6	15.5 ± 8.4	10.5 ± 2.5
8	34.7 ± 8.8	40.7 ± 10.2
10	487.4 ± 103.0	166.1 ± 29.6
12	$> 3h$	784.2 ± 269.7

Table 4. Average training time (sec.) for the constrained and unconstrained iterative weighted L_2-penalized method (10 simulations, $p = 200$).

In average the unconstrained approach satisifies at least 70% of the monotonicity constraints. Also the generalizing performances (test error) of both methods are similar. As expected, the constrained approach (enforcing monotonicity) becomes computationally heavier than the unconstrained approach for n greater or equal to 10. For $n = 12$ it does not provide a solution within 3 hours.

5 Conclusion

The iterative re-weighted L_2-penalized method presented in the paper appears to be quite efficient for learning capacities admitting sparse representations in terms of Möbius masses. This enables to model human preferences in the presence of possibly interacting criteria by

n	Constrained L_2^*	Unconstrained L_2^*
4	$100 \pm 0.0\%$	$82.5 \pm 5.8\%$
6	$100 \pm 0.0\%$	$87.3 \pm 8.9\%$
8	$100 \pm 0.0\%$	$69.3 \pm 14.8\%$
10	$100 \pm 0.0\%$	$74.2 \pm 9.9\%$
12	$- - - - -$	$76.2 \pm 21.0\%$

Table 5. Average monotonicity degree for the constrained and unconstrained iterative weighted L_2-penalized method (10 simulations, $p = 200$).

n	Constrained L_2^*	Unconstrained L_2^*
4	$4.0 \pm 0.1\%$	$3.9 \pm 0.6\%$
6	$7.6 \pm 1.5\%$	$8.0 \pm 2.0\%$
8	$6.8 \pm 2.1\%$	$7.8 \pm 0.7\%$
10	$10.8 \pm 1.8\%$	$11.9 \pm 2.3\%$
12	$- - - - -$	$17.3 \pm 4.6\%$

Table 6. Average test error for the constrained and unconstrained iterative weighted L_2-penalized method (10 simulations, $p = 200$).

simple instances of the Choquet integral with very good empirical performance in generalization. In particular, this approach significantly improves the approach based on L_1 regularization in terms of computation time and scalability. Our tests show that we can efficiently learn sparse models for problems involving up to 20 criteria (which represent more than 1 million of possible interactions that have to be investigated in order to identify the most significant). Let us remark that the proposed approach is not restricted to the Choquet integral and could be applied to learn other aggregation functions based on a capacity. In particular, it can directly be applied to learn sparse capacities in the multilinear utility model [21].

To go further, the main challenges are: 1) further improving scalability and 2) gaining efficiency in the management of monotonicity constraints. A natural idea that might be worth exploring to improve computational efficiency would be to exploit a kernalisation but the challenge here lies in the fact that the weighting matrix must be recalculated from the primal solution at each iteration which may neutralize the benefit of kernalization. Furthermore, when the learned model is required to be monotonic with respect to Pareto dominance, the learned capacity must be monotonic with respect to set inclusion. Although this property is not necessary for general multiple regressions it becomes highly desirable for models designed to aggregate criterion values or utilities. The approach we proposed to guarantee the monotonicity of the learned models, based on primal formulations of the learning problem, is computationally less efficient than the approach based on dualization. In order to improve scalability under monotonicity constraints, an alternative approach might be to incrementally establish monotonicity along the iterative learning process using constraint generation. This option is left for further research.

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*, pp. 1 – 7, (2015).

[2] Derek T. Anderson, Stanton R. Price, and Timothy C. Havens, 'Regularization-based learning of the Choquet integral', in *FUZZ-IEEE*, pp. 2519 – 2526, (2014).

[3] Gleb Beliakov and Jian-Zhang Wu, 'Learning fuzzy measures from data: simplifications and optimisation strategies', *Information Sciences*, **494**, 100–113, (2019).

[4] Gleb Beliakov and Jian-Zhang Wu, 'Learning k-maxitive fuzzy measures from data by mixed integer programming', *Fuzzy Sets and Systems*, **412**, 41–52, (2021).

[5] Gleb Beliakov and Jian-Zhang Wu, 'Random generation of capacities and its application in comprehensive decision aiding', *Information Sciences*, **577**, 424–435, (2021).

[6] Nawal Benabbou, Patrice Perny, and Paolo Viappiani, 'Incremental elicitation of Choquet capacities for multicriteria choice, ranking and sorting problems', *Artificial Intelligence*, **246**, 152–180, (2017).

[7] Nadjet Bourdache, Patrice Perny, and Olivier Spanjaard, 'Incremental elicitation of rank-dependent aggregation functions based on Bayesian linear regression', in *IJCAI*, pp. 2023–2029, (2019).

[8] Roman Bresson, Johanne Cohen, Eyke Hüllermeier, Christophe Labreuche, and Michèle Sebag, 'Neural representation and learning of hierarchical 2-additive Choquet integrals', in *IJCAI*, pp. 1984–1991, (2020).

[9] Alain Chateauneuf and Jean-Yves Jaffray, 'Some characterizations of lower probabilities and other monotone capacities through the use of möbius inversion', *Mathematical social sciences*, **17**(3), 263–283, (1989).

[10] Elías F Combarro, Irene Díaz, and P Miranda, 'On random generation of fuzzy measures', *Fuzzy Sets and Systems*, **228**, 64–77, (2013).

[11] Florian Frommlet and Grégory Nuel, 'An adaptive ridge procedure for l 0 regularization', *PloS one*, **11**(2), e0148620, (2016).

[12] Michel Grabisch, 'The application of fuzzy integrals in multicriteria decision making', *European journal of operational research*, **89**(3), 445–456, (1996).

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

[14] Michel Grabisch, Ivan Kojadinovic, and Patrick Meyer, 'A review of methods for capacity identification in Choquet integral based multi-attribute utility theory: Applications of the Kappalab R package', *European journal of operational research*, **186**(2), 766–785, (2008).

[15] 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).

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

[17] Trevor Hastie, Robert Tibshirani, and Martin Wainwright, *Statistical learning with sparsity: the Lasso and Generalizations*, CRC Press, 2015.

[18] Margot Herin, Patrice Perny, and Nataliya Sokolovska, 'Learning sparse representations of preferences within choquet expected utility theory', in *The 38th Conference on Uncertainty in Artificial Intelligence*, (2022).

[19] Margot Herin, Patrice Perny, and Nataliya Sokolovska, 'Learning utilities and sparse representations of capacities for multicriteria decision making with the bipolar choquet integral', in *13th Multidisciplinary Workshop on Advances in Preference Handling*, (2022).

[20] 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 (FUZZ-IEEE)*, pp. 1–8. IEEE, (2020).

[21] Ralph L Keeney, Howard Raiffa, and Richard F Meyer, *Decisions with multiple objectives: preferences and value trade-offs*, Cambridge university press, 1993.

[22] Zhenqiu Liu, David Elashoff, and Steven Piantadosi, 'Sparse support vector machines with l0 approximation for ultra-high dimensional omics data', *Artificial intelligence in medicine*, **96**, 134–141, (2019).

[23] Thi Thanh Nguyen, Charles Soussen, Jérôme Idier, and El-Hadi Djermoune, 'Np-hardness of l_0 minimization problems: revision and extension to the non-negative setting', in *13th International Conference on Sampling Theory and Applications, SampTA 2019*, (2019).

[24] Anthony J. Pinar, Derek T. Anderson, Timothy C. Havens, Alina Zare, and Titilope Adeyeba, 'Measures of the Shapley index for learning lower complexity fuzzy integrals', *Granul. Comput.*, **2**, 303 – 319, (2017).

[25] John Shawe-Taylor, Nello Cristianini, et al., *Kernel methods for pattern analysis*, Cambridge university press, 2004.

[26] Ali Fallah Tehrani, 'The choquet kernel on the use of regression problem', *Information Sciences*, **556**, 256–272, (2021).

[27] Ali Fallah Tehrani and Eyke Hülermeier, 'Ordinal Choquistic regression', in *EUSFLAT*, (2013).

[28] Ali Fallah Tehrani, Christophe Labreuche, and Eyke Hüllermeier, 'Choquistic utilitaristic regression', in *DA2PL*, pp. 35 – 42, (2014).

[29] Ali Fallah Tehrani, Marc Strickert, and Eyke Hüllermeier, 'The choquet kernel for monotone data.', in *Esann*, (2014).

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

[31] Vicenç Torra, 'The weighted owa operator', *International Journal of Intelligent Systems*, **12**(2), 153–166, (1997).

[32] D Wipf and S Nagarajan, 'Iterative reweighted 1 and 2 methods for finding sparse solutions. uc san francisco', Technical report, Technical Report, (2008).

[33] Ronald R Yager, 'On ordered weighted averaging aggregation operators in multicriteria decision making', *IEEE Transactions on systems, Man, and Cybernetics*, **18**(1), 183–190, (1988).

[34] Peng Zhao and Bin Yu, 'On model selection consistency of lasso', *The Journal of Machine Learning Research*, **7**, 2541–2563, (2006).