ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230351

A Hybrid Approach to Preference Learning with Interaction Terms

Hugo Gilbert[b], Mohamed Ouaguenouni[a;*], Meltem Öztürk[b] and Olivier Spanjaard[a]

[a]Sorbonne Université, CNRS, LIP6, Paris, France
[b]LAMSADE, PSL, Université Paris-Dauphine, CNRS, Paris, France

Abstract. Preference learning is an essential component in numerous applications, such as recommendation systems, decision-making processes, and personalized services. We propose here a novel approach to preference learning that interleaves Gaussian Processes (GP) and Robust Ordinal Regression (ROR). A Gaussian process gives a probability distribution on the latent function values that generate users' preferences. Our method extends the traditional non-parametric Gaussian process framework by approximating the latent function by a very flexible parameterized function, that we call θ-additive function, where θ is the parameter set. The set θ reflects the degree of sophistication of the generalized additive model that can potentially represent the user's preferences. To learn what are the components of θ, we update a probability distribution on the space of all possible sets θ, depending on the ability of the parameterized function to approximate the latent function. We predict pairwise preferences by using the parameter set θ that maximizes the posterior distribution and by performing robust ordinal regression based on this parameter set. Experimental results on synthetic data demonstrate the effectiveness and robustness of our proposed methodology.

1 Introduction

Assuming a parametric decision model representing the preferences of a Decision Maker (DM) and a learning set of DM's preferences, the Robust Ordinal Regression (ROR) framework aims to infer robust preferences by working on the polyhedron of parameters values compatible with the expressed preferences. While the principles underlying ROR date back to Lahdelma *et al.* [22], this framework has gained increasing attention in multicriteria decision aiding [9, 10, 18] and artificial intelligence [3, 16], especially in preference elicitation problems with positive or negative interactions between criteria.

A question which is often eluded is how to select the decision model representing the DM's preferences. This model is usually assumed to be chosen *beforehand*. For instance, we may use a simple weighted average on the performances on the various criteria, or a more complex aggregator as a Choquet integral [17, 19, 21], which makes it possible to take into account synergies between criteria. Yet the choice of the decision model is critical on the predicted preferences and taking into account the DM's preferences for selecting the model itself (upstream of the model parameters elicitation) has been little studied until now.

A well-known and appealing preference learning framework in this matter is that of Gaussian processes for preference elicitation [6, 8, 25], which makes it possible to approximate any possible utility function defined on the set of alternatives. However, the fact that Gaussian processes do not rely on an analytical formulation of the utility function is both an advantage (for expressivity) and a disadvantage because the explanation of the preference predictions are made difficult by the absence of this analytical form.

In this work, given a reference set of items, we assume that each alternative corresponds to a subset of items. This can be viewed as a special case of multiattribute decision making where each attribute is binary and corresponds to the presence or not of an item in the considered subset. The utility $f(A)$ of a subset A of items is defined as the sum of values of specific combinations of items that are present in A (the singletons and the combinations of items that generate synergies). The form of f is close to a discrete Choquet integral [1, 5, 23] over binary vectors, expressed as the sum of Möbius masses but relaxing some of its constraints (e.g., monotonicity of the capacity). We denote by θ the set of combinations of items that are considered in the sum. However, instead of assuming that the set θ is known beforehand, we use a Bayesian approach to learn the specific form of the decision model from the learning set of preferences. Put another way, the decision model itself is learnt from the preference data. Once the decision model is learnt, we use ROR to make preference predictions by circumventing the possible parameter values based on the learning set of preferences.

Similarly to this work, Gilbert *et al.* [16] propose a preference learning method based on a generalized additive utility function that depends on θ. However, they do not adopt a Bayesian approach to learn the set θ but instead consider all "simplest" sets θ that fit the preferences in the learning set, which impacts the computational burden. Learning a single set θ makes it possible to significantly alleviate the calculations. Furthermore, the combination of a Bayesian approach to learn the decision model and a ROR approach to predict preferences yields other interesting advantages. On the one hand, learning a decision model instead of a more black-box approach yields the advantage of explainability of the recommendations, as the parameters of the model have a clear interpretation. On the other hand, using ROR (instead of setting precise parameters' values) allows for more reliability when inferring new preferences, as some preference instances may be rejected if the preference prediction for this pair of alternatives is likely to be inaccurate given the available preference data (i.e., no preferences prediction is made for this pair of alternatives). Hence, the proposed method falls within the framework of *learning with rejection* [11]. Before concluding this introduction, let us mention the works by Bigot *et al.* [4] and Domsh-

* Corresponding Author. Email: mohamed.ouaguenouni@lip6.fr.

lak and Joachims [13], proposing preference learning methods also based on generalized additive utility functions: the former is based on a PAC-learner approach while the latter is based on a Support Vector Machines (SVM) approach. These two methods differ nevertheless from ours, in particular because they do not fall within the framework of learning with rejection.

The paper is organised as follows. Section 2 describes how to perform robust ordinal regression for the generalized additive utility model we use. Section 3 focuses on the Bayesian approach to learn the "most probable" additive decomposition of the utility (i.e., to learn the decision model itself). Sections 4 and 5 are devoted to the presentation of the results of numerical experiments.

2 ROR on the θ-additive model

We consider a DM expressing her preferences on a set $\mathcal{A}$ of alternatives, where each alternative is described as a subset of elements in $\mathcal{F}=\{a_1,a_2,\ldots,a_n\}$. Put another way, for each $A\in\mathcal{A}$, we have $A\subseteq\mathcal{F}$. An alternative A can also be described by a binary vector of length n, in which the i^{th} component is 1 iff $a_i\in A$.

2.1 The θ-additive model

In this work, we assume that the DM's preferences are represented using a generalized additive utility model where each alternative A has a global utility $f(A)$, and alternative A is strictly preferred to alternative B, denoted by $A\succ B$, if and only if $f(A)>f(B)$. The simplest such model is the additive model [14], in which a parameter $w(a)\in\mathbb{R}$ is defined for each element $a\in\mathcal{F}$, such that for all $A\in\mathcal{A}$, $f(A)=\sum_{a\in A}w(a)$. However, this model does not make it possible to represent interactions (positive or negative) between the elements. To this end, one may rely on the k-additive model, which assumes the existence of a parameter $w(S)\in\mathbb{R}$ for each $S\in[\mathcal{F}]^k$, where $[\mathcal{F}]^k=\{S\subseteq\mathcal{F}:1\leq|S|\leq k\}$ and $f(A)=\sum_{S\in[\mathcal{F}]^k}I_A(S)w(S)$, with $I_A(S)=1$ if $S\subseteq A$ and 0 otherwise. The parameter $w(S)$ can be seen as the "weight" of subset S. While choosing k strictly greater than 1 makes it possible to account for synergies between subsets of at most k elements, the size of the model may become prohibitively expansive as k increases. Hence, to keep a compact representation, we rely on the θ-additive model [16]; given a set $\theta\subseteq 2^{\mathcal{F}}$ and a set function $w:\theta\to\mathbb{R}$ (also called weight function in the sequel), we assume that f is of the form $f(A)=\sum_{S\in\theta}I_A(S)w_S$, where w_S is a shorthand for $w(S)$. We use the notation $f_{\theta,w}(A)$ instead of $f(A)$ to make explicit that f is parameterised by weights w_S for $S\in\theta$. Note that the additive (resp. k-additive) model is the special case in which $\theta=\mathcal{F}$ (resp. $\theta=[\mathcal{F}]^k$).

Before launching the learning method, we assume that both the set θ and the set function w defined on θ are unknown. We detail in Section 3 how to select a relevant set θ from a collection of pairwise preferences expressed by the DM, which is the main contribution of the paper. Prior to that, we describe in the following how to use robust ordinal regression once the set θ is chosen.

2.2 Robust ordinal regression

We denote by R the set of strict pairwise preferences expressed by the DM, i.e., $R\subseteq\{(A,B)\in\mathcal{A}^2|A\succ B\}$, from which the set function w is partially specified in order to predict other pairwise preferences. Besides, we denote by $\mathcal{S}$ the set of alternatives that appears in R, i.e., $\mathcal{S}=\{A\in\mathcal{A}|\exists B\in\mathcal{A},(A,B)\in R\vee(B,A)\in R\}$.

Even when θ is learnt, the function $f_{\theta,w}$ is still unknown as several weight functions w on θ may be compatible with the preferences observed in R. We denote by W_θ^R the set of such weight functions:

$$W_\theta^R=\{w:\theta\to\mathbb{R}\,|\,\forall(A,B)\in R,\ f_{\theta,w}(A)>f_{\theta,w}(B)\}.$$

Note that a pair $\{w,w'\}$ of weight functions both in W_θ^R may yet lead to infer opposite preferences [2]. In order to infer further preferences not dependent on an arbitrary choice of the weight function in W_θ^R, we turn to a robust ordinal regression approach. More precisely, we use the *θ-ordinal dominance relation*[1].

Definition 1 (adapted from [16]) *Let $\mathcal{F}$ be a set of elements, $\mathcal{A}\subseteq 2^{\mathcal{F}}$ a set of subsets of $\mathcal{F}$ and R a set of pairs $(A,B)\in\mathcal{A}^2$ such that $(A,B)\in R$ iff $A\succ B$. The θ-ordinal dominance relation, denoted by $\succ_\theta^R$, is defined for every $A,B\in\mathcal{A}$ by:*

$$A\succ_\theta^R B\Leftrightarrow\forall w\in W_\theta^R,\ f_{\theta,w}(A)>f_{\theta,w}(B).$$

While the θ-ordinal dominance relation is independent from the choice of a specific $w\in W_\theta^R$, it may obviously be partial, and we define the rejection relation $\bowtie_\theta^R$ as:

$$A\bowtie_\theta^R B\Leftrightarrow\exists w,w'\in W_\theta^R,\\ (f_{\theta,w}(A)\geq f_{\theta,w}(B)\text{ and }f_{\theta,w'}(B)\geq f_{\theta,w'}(A)).$$

If $A\succ_\theta^R B$ then one can predict, based on R and for the θ-additive model, that A is strictly preferred to B. Otherwise, if $A\bowtie_\theta^R B$ then no prediction is made (i.e., the prediction is rejected).

3 Learning θ_{map}

Using R, this section explains how to learn the set θ used in the θ-additive model. We rely on a Bayesian approach to find a set θ_{map} (map stands for *maximum a posteriori*) maximizing the posterior distribution[2] $\mathbb{P}(\theta|R)$:

- We assume the existence of a latent function $f:\mathcal{S}\to\mathbb{R}$ which associates to each element $A\in\mathcal{S}$ a utility value $f(A)$. We start by approximating a posterior distribution $\mathbb{P}(f|R)$ on f given the available preferential data R, by using an appropriate likelihood function $\mathbb{P}(R|f)$ and a prior $\mathbb{P}(f)$ (see Section 3.1).
- After that, since the objective is to derive a generalized additive utility model, we model the conditional probability $\mathbb{P}(w|f,R)$ of a weight function $w:\theta_d\to\mathbb{R}$ given f, where d is the degree of R and $\theta_d=[\mathcal{F}]^d$ (see Section 3.2). The *degree* of R is the minimal integer k such that the preferences in R can be represented by a k-additive utility model. The better the values $f_{\theta_d,w}(A)$ approximates the values $f(A)$ for $A\in\mathcal{S}$, the higher is the probability $\mathbb{P}(w|f,R)$.
- Then, we sample vectors of weights from $\mathbb{P}(w|R)$ by using the formula $\mathbb{P}(w|R)=\int_f\mathbb{P}(w|f,R)\mathbb{P}(f|R)df$. We use the sampled vectors of weights to derive a posterior $\mathbb{P}(\theta|R)$ on the models θ by using a prior $\mathbb{P}(\theta)$ and a likelihood $\mathbb{P}(w|\theta)$ (see Section 3.3).
- Finally, in order to compute θ_{map}, we use a Markov Chain Monte Carlo (MCMC) approach to sample models θ from the posterior $\mathbb{P}(\theta|R)$ (see Section 3.4).

Each of these steps is detailed in the following subsections. At the end of this section, it should be clear that:

$$\mathbb{P}(\theta|R)=\mathbb{P}(\theta)\int_w\left[\int_f\mathbb{P}(w|f,R)\mathbb{P}(f|R)df\right]\mathbb{P}(w|\theta)dw.$$

[1] The idea of ordinal dominance based on a generalized additive utility function dates back to Fishburn and Lavalle [15].

[2] Following Chu and Ghahramani [8] and by abuse of notation, we use symbol $\mathbb{P}$ to denote both a probability mass function (for discrete variables) and a probability density function (for continuous variables), to avoid confusion with the latent (utility) function f.

3.1 Posterior distribution $\mathbb{P}(f|R)$ on the valuation functions f given the pairwise preferences in R

The posterior distribution on the valuation functions $\mathbb{P}(f|R)$ is learnt by using a Gaussian process, similarly as in the work of Chu and Ghahramani [8]. In the following, assuming that $\mathcal{S}=\{A_1,\ldots,A_N\}$, the function $f:\mathcal{S}\to\mathbb{R}$ is viewed as a vector $[f(A_1),\ldots,f(A_N)]$.

It is standard to consider that each utility value $f(A)$ is affected by a Gaussian noise $\delta\sim\mathcal{N}(0,\sigma_f)$ in the eyes of the DM, which yields the following likelihood for a set R of pairwise preferences given f:

$$\mathbb{P}(R|f)=\prod_{(A,B)\in R}\Phi\left(\frac{f(A)-f(B)}{2\sigma_f}\right),$$

where Φ is the cumulative distribution function of the standard normal distribution $\mathcal{N}(0,1)$. The prior we choose to consider for our valuation functions f is a centered multivariate normal with a covariance matrix Σ_f:

$$\mathbb{P}(f)=\frac{1}{2\pi^{\frac{N}{2}}|\Sigma_f|^{\frac{1}{2}}}e^{-\frac{1}{2}f^T\Sigma_f^{-1}f}.$$

The covariance matrix Σ_f is obtained by applying the Mercer kernel on each pair of alternatives $A,B\in\mathcal{S}$, i.e, the ij-th element of Σ_f is given by $k(A_i,A_j)$, where

$$k(A,B)=\exp\left(-\frac{k}{2}(\overrightarrow{A}-\overrightarrow{B})^T(\overrightarrow{A}-\overrightarrow{B})\right),\qquad(1)$$

with $\overrightarrow{A}=[I_A(a_1),\ldots,I_A(a_n)]$ the vectorial representation of an alternative A. Combining the likelihood $\mathbb{P}(R|f)$ and the prior $\mathbb{P}(f)$ using Bayes rule, the maximum a posteriori f_{map} can be obtained by minimizing the following cost function:

$$\mathcal{C}(f)=-\sum_{(A,B)\in R}\ln\left(\Phi\left(\frac{f(A)-f(B)}{2\sigma_f}\right)\right)+\frac{1}{2}f^T\Sigma_f^{-1}f.$$

To approximate the posterior distribution $\mathbb{P}(f|R)$, we proceed using the Laplace approximation. This amounts to approximate the posterior on f with a normal $\mathcal{N}(\mu_{f|R},\Sigma_{f|R})$ where

$$\mu_{f|R}=f_{map}=\arg\min_f\mathcal{C}(f),$$

$$\sigma_{f|R}=(\Lambda|_{f_{map}}+\Sigma_f^{-1})^{-1},$$

where Λ is the Hessian matrix of shape (N,N) of the negative log-likelihood w.r.t to f, i.e., $\Lambda_{i,j}$ is the following second derivative:

$$\Lambda_{i,j}=\frac{\partial^2\left(-\sum_{(A,B)\in R}\ln\left(\Phi\left(\frac{f(A)-f(B)}{2\sigma_f}\right)\right)\right)}{\partial f(x_i)\partial f(x_j)}.$$

Since there are few hyperparameters (only σ_f and k), they are fitted on an approximation of the evidence using a grid-search approach as done by Chu and Ghahramani [8].

3.2 Posterior $\mathbb{P}(w|f,R)$ on the weight vectors w given the valuation function f and the preferences R

Let d be the degree[3] of the preferences in R, i.e., the minimal integer value such that $\sum_{S\in[\mathcal{F}]^d}w_SI_A(S)\geq\sum_{S\in[\mathcal{F}]^d}w_SI_B(S)$ for all $(A,B)\in R$. For the sake of model conciseness, we limit ourselves to sets $\theta\subseteq[\mathcal{F}]^d$ (i.e., θ only contains sets of size less than d). Denoting by $S_1,\ldots,S_m$ the elements of $[\mathcal{F}]^d$, we work in this section on weight vectors $w=[w_{S_1},\ldots,w_{S_m}]$. The set $\mathcal{S}=\{A_1,\ldots,A_N\}$ of alternatives can then be represented by a binary matrix D of shape (N,m) where D_{ij} is 1 if $S_j\subseteq A_i$ (the i-th alternative of $\mathcal{S}$), and 0 otherwise.

From the matrix D and the valuation vector f, we perform a Bayesian linear regression to obtain a distribution on the weight vectors w. To evaluate this distribution, we use a Gaussian prior on the weight vectors $\mathbb{P}(w)\sim\mathcal{N}(w|\mu_w,\Sigma_w)$:

$$\mathbb{P}(w)=\frac{1}{(2\pi)^{m/2}|\Sigma_w|^{1/2}}\exp\left(-\frac{1}{2}(w-\mu_w)^T\Sigma_w^{-1}(w-\mu_w)\right).$$

For a fixed set $\mathcal{S}$ and a fixed set θ ($\theta=\theta_d$), the distribution $\mathbb{P}(w|f,R)$ does not depend on R and can be abbreviated by $\mathbb{P}(w|f)$. To find the posterior distribution $\mathbb{P}(w|f)$, we apply Bayes theorem and combine the expressions for the prior and likelihood:

$$\mathbb{P}(w|f)\propto\exp\left(-\frac{1}{2\sigma_w^2}(f-Dw)^T(f-Dw)\right)\times\exp\left(-\frac{1}{2}(w-\mu_w)^T\Sigma_w^{-1}(w-\mu_w)\right).$$

We have then $\mathbb{P}(w|f)\sim\mathcal{N}(w|\mu_{w|f},\Sigma_{w|f})$, where:

$$\Sigma_{w|f}^{-1}=\frac{1}{\sigma_w^2}D^TD+\Sigma_w^{-1},$$

$$\mu_{w|f}=\Sigma_{w|f}\left(\frac{1}{\sigma_w^2}D^Tf+\Sigma_w^{-1}\mu_w\right).$$

The hyperparameters μ_w and Σ_w are fixed so that $\mathbb{P}(w)\sim\mathcal{N}(0,1)$, and the hyperparameter σ_w is fit by maximizing the marginal likelihood defined by:

$$\mathbb{P}(f|\sigma_w)=\int_w\mathbb{P}(f|w,\sigma)\mathbb{P}(w)dw.$$

The maximization is performed as follows. Since the likelihood $\mathbb{P}(f|w,\sigma)\sim\mathcal{N}(f|Dw,\sigma^2I_n)$ and the prior $\mathbb{P}(w)\sim\mathcal{N}(w|\mu_w,\Sigma_w)$ are both Gaussian, the marginal likelihood is also Gaussian:

$$\mathbb{P}(f|\sigma_w)=\mathcal{N}(f|D\mu_w,D\Sigma_wD^T+\sigma_w^2I_n).$$

We estimate σ_w by maximizing $\log\mathbb{P}(f|\sigma_w)$, using a gradient descent approach:

$$\log\mathbb{P}(f|\sigma_w)=-\frac{n}{2}\log(2\pi)-\frac{1}{2}\log|D\Sigma_wD^T+\sigma_w^2I_n|-\frac{1}{2}(f-D\mu_w)^T(D\Sigma_wD^T+\sigma_w^2I_n)^{-1}(f-D\mu_w).$$

3.3 The posterior distribution on the parameters sets $\mathbb{P}(\theta|R)$

A preliminary step to compute θ_{map} with an MCMC approach is to learn the posterior distribution $\mathbb{P}(\theta|R)$ on the parameters sets θ. We recall that we restrict our attention to sets $\theta\subseteq[\mathcal{F}]^d$, where d is the degree of R. As a result, all models explored through our MCMC approach will be encompassed within the set θ_d.

The core concept of our method to estimate $\mathbb{P}(\theta|R)$ can be summarized as follows: we sample random weight vectors utilizing the distribution $\mathbb{P}(w|R)$ and assess the proportion of these vectors that conform to the θ model. We consider that a weight vector w *conform* to a θ model if the values of all parameters in $[\mathcal{F}]^d\setminus\theta$ fall within the range $[-\epsilon,\epsilon]$.

We now present more formally the method using the classic Bayesian framework. We have:

$$\mathbb{P}(\theta|R)=\int_w\mathbb{P}(\theta|w)\mathbb{P}(w|R)dw.$$

[3] The computation of the degree of the preferences in R is explained in Section 3.4.

But since $\mathbb{P}(\theta|w) \propto \mathbb{P}(w|\theta)\mathbb{P}(\theta)$ we have:

$$\mathbb{P}(\theta|R) = \mathbb{P}(\theta) \int_w \mathbb{P}(w|\theta)\mathbb{P}(w|R)dw$$
$$= \mathbb{P}(\theta) \int_w \left[\int_f \mathbb{P}(w|f)\mathbb{P}(f|R)df \right] \mathbb{P}(w|\theta)dw.$$

In this expression, $\mathbb{P}(w|\theta)$ captures the "compatibility" of the weight vectors with the model θ. Although various functions could be employed to represent this compatibility, we will opt for the most straightforward choice:

$$\mathbb{P}(w|\theta) = \begin{cases} 1 & \text{if } -\epsilon \leq w_S \leq \epsilon \quad \forall S \in [\mathcal{F}]^d \setminus \theta, \\ 0 & \text{otherwise.} \end{cases}$$

Using this expression for $\mathbb{P}(w|\theta)$ and expressions $\mathbb{P}(w|f, R)$ and $\mathbb{P}(f|R)$ previously given, $\mathbb{P}(\theta|R)$ is approximated by using a Monte Carlo sampling method, provided that a prior $\mathbb{P}(\theta)$ is defined.

Various priors on θ could be adopted. As our aim is to promote models with few parameters and the simplest possible interactions, we consider a prior on each parameter and assume that the probability of a given set θ is the product of the probabilities of its parameters. To favour interactions involving few elements, the prior on a parameter is defined as a function of its cardinality:

$$\mathbb{P}(\theta) \propto \prod_{S \in \theta} \frac{1}{k_{|S|}} \mathbb{P}(|S|)$$

where $k_{|S|} = \binom{n}{|S|}$ is the number of sets of $|S|$ elements. For $\mathbb{P}(|S|)$, we opt for the Half-Cauchy distribution with parameter γ, whose probability density function is $\mathbb{P}(x;\gamma) = 2/\left(\pi\gamma\left(1 + x^2/\gamma^2\right)\right)$. Although the Half-Cauchy distribution is continuous rather than discrete, we have chosen it because of its ability to concentrate the probability mass to small cardinalities, hence favoring small sets S.

3.4 Computing θ_{map} from $\mathbb{P}(\theta|R)$

To determine the parameter set θ_{map} (that maximizes the posterior distribution $\mathbb{P}(\theta|R)$), we employ an MCMC approach based on the Metropolis-Hastings algorithm [7, 24].

To implement it, we take advantage of the fact that all the models we sample are included in $[\mathcal{F}]^d = \{S_1, \ldots, S_m\}$, and map each θ to $\overrightarrow{\theta} = [\mathbb{1}_\theta(S_1), \ldots, \mathbb{1}_\theta(S_m)]$, where $\mathbb{1}_\theta(S) = 1$ if $S \in \theta$, 0 otherwise. We then define a proposal distribution $Q(\overrightarrow{\theta}'|\overrightarrow{\theta})$ that associates to each vector $\overrightarrow{\theta}$ a probability distribution on the next model $\overrightarrow{\theta}'$ whose posterior probability will be evaluated:

$$Q(\overrightarrow{\theta}'|\overrightarrow{\theta}) = \begin{cases} \frac{1}{|\theta_d|} & \text{if } d_H(\overrightarrow{\theta}', \overrightarrow{\theta}) = 1, \\ 0 & \text{otherwise,} \end{cases}$$

where d_H is the Hamming distance on binary vectors.

Note that it may happen that θ_{map} is not fully compatible with R in the sense that $W^R_{\theta_{map}} = \emptyset$. In order to perform ROR with θ_{map}, we propose a post-treatment of R in order to obtain a subset R' of R (as large as possible) such that $W^{R'}_{\theta_{map}} \neq \emptyset$. We proceed in a greedy manner to build R'. First, we test in polynomial time whether the polyhedron W^R_θ is empty by solving the following linear program:

$$(\mathcal{P}_\theta) \begin{cases} \min_{e_{A,B}, v_S} \sum_{(A,B) \in R} e_{A,B} & \\ \sum_{S \in \theta} (I_A(S) - I_B(S)) v_S \geq 1 - e_{A,B}, & \forall (A,B) \in R \\ e_{A,B} \geq 0, & \forall (A,B) \in R \\ v_S \in \mathbb{R}, & \forall S \in \theta \end{cases}$$

If the optimal objective value is 0, we conclude that $W^R_\theta \neq \emptyset$.

We also use the linear program $\mathcal{P}_\theta$ to compute the degree of the set R of pairwise preferences: the degree corresponds to the first integer k in $1, 2, \ldots$ such that $W^R_{\theta_k} \neq \emptyset$, i.e., the optimal value of $\mathcal{P}_{\theta_k}$ is 0.

For a pair of alternatives $(A, B) \notin R$, the predictive distribution $\mathbb{P}(f(A) > f(B)|R)$ can then be obtained through the estimation of the posterior distribution $\mathbb{P}(f|R)$ by integrating on the f-space, as explained in detail in paragraph "Gaussian Process" of Section 4.2.

Using these two concepts, if the polyhedron W^R_θ is empty we proceed as follows:

1. We sort the preferences based on their likelihood which is estimated using the predictive distribution $\mathbb{P}(f(A) > f(B)|R)$.
2. We add the preferences from the most likely to the less likely; we test at each step if W^R_θ is empty, and if it is the case we remove the last added preference.

4 Numerical Setup

The purpose of the numerical tests is twofold: first, we try to determine whether the model θ_{map} we compute using our Bayesian approach yields good results for the ROR; second, we evaluate the prediction performances of our approach with two baseline models (Support Vector Classifier and Gaussian Process) using widely-accepted metrics (Precision, Recall and F-score).

In this section, we succinctly explain how our data are generated. We then present the baseline learning methods to which our approach will be compared, and the evaluation metrics we will use.

4.1 Synthetic Data Generation

Following [16], we go through three steps to generate a synthethic dataset of preferences: 1) sampling a random θ-additive function, 2) associating it to a ranking function r, 3) using the function to collect a set R of preferences.

Sampling a random θ-additive function. This phase is conditioned by two parameters α and p. We start by sampling a model θ. To do so, we initialise it to $\theta = \{\{a_1\}, \ldots, \{a_n\}\}$ and then we add a proportion α of all subsets $S \subseteq \mathcal{F}$ to it, sampled as follows:

1. We initialise S with a random item a_i.
2. We augment the set S by appending an element that is randomly sampled within $\mathcal{F} \setminus S$.
3. We exit the process with probability p, otherwise go back to 2.

After that, each coefficient w_S for $S \in \theta$ is sampled w.r.t. a normal distribution $\mathcal{N}(0, 100)$. We obtain then a θ-additive function $f_{\theta,w}$.

Associating the θ-additive function to a ranking function. We define an ordinal classification function $r : \mathcal{A} \to \{0, 1, \ldots t-1\}$, where t is the number of classes:

- We compute $M = \max_{A \in \mathcal{A}} f_{\theta,w}(A)$ and $m = \min_{A \in \mathcal{A}} f_{\theta,w}(A)$.
- We divide the interval $[m, M]$ into t subintervals.

The class of each alternative A is the interval in which its utility $f_{\theta,w}(A)$ lies in (the higher the rank, the better the class).

Collecting the preferences. The preferences are collected by sampling a set $\mathcal{A}_{train}$ of alternatives, computing $f_{\theta,w}(A)$ and thus $r(A)$ for each $A \in \mathcal{A}_{train}$, and deducing the preferences as follows:

$$R = \{(A, B) \in \mathcal{A}^2_{train} | r(A) > r(B)\}$$

It should be noted that even though elements from different classes can be compared, we assume that two elements belonging to the same class remain incomparable (i.e., R is a strict weak ordering). For instance, in hotel classification, knowing that two hotels have two stars does not allow us to conclude that one is better than the other, or that they are indifferent. It is sometimes said that incomparability reflects a degree of *ignorance* [20].

4.2 Baseline Models

In this section, we discuss the baseline models against which our proposed approach is compared.

We consider two widely-used baseline models for preference learning. The first model is the Support Vector Classifier, a popular supervised learning algorithm introduced by Vapnik [12] and widely used in the context of preference learning (for example in [13]). The second one is the classic Gaussian process approach for preference learning, by Chu and Ghahramani [8].

Support Vector Classifier (SVC). To apply SVC to preference learning, we first transform the preferences $(A, B) \in R$ into vectors by mapping each alternative into a $|\theta|$-dimensional space:

$$v_\theta^{(A,B)} = \overrightarrow{A}_\theta - \overrightarrow{B}_\theta$$

where the vector $\overrightarrow{A}_\theta$ is defined as $\overrightarrow{A}_\theta = [I_A(X_1), \ldots, I_A(X_p)]$, assuming that $\theta = \{X_1, \ldots, X_p\}$. We associate then to each preference (A, B) two vectors, the vector $v_\theta^{(A,B)}$ with the label 1 and the vector $v_\theta^{(B,A)}$ with the label 0. The algorithm proceeds by fitting a hyperplane in a $|\theta|$-dimensional space for a given θ that separates the different classes of data points. The inference is then made for each pair of alternative (S, T) by predicting the label of the vector $v_\theta^{(S,T)}$ and the label of the vector $v_\theta^{(T,S)}$. We predict that $S \succ T$ (resp. $T \succ S$) if the predicted label of $v_\theta^{(S,T)}$ is 1 and the predicted label of $v_\theta^{(T,S)}$ is 0 (resp. the predicted label of $v_\theta^{(S,T)}$ is 0 and the predicted label of $v_\theta^{(T,S)}$ is 1) otherwise no prediction is made.

Gaussian Process (GP). This method first approximates the distribution $\mathbb{P}(f|R)$ as detailed by Chu and Ghahramani [8] (and explained in Subsection 3.1). Then, the prediction is made by supposing that for a pair of alternative $A, B \in \mathcal{A}$ the latent variable $f_t = [f(A), f(B)]$ will follow a zero-mean gaussian distribution and will be correlated to the N latent variables of the training samples $f = [f(A_1), \ldots, f(A_N)]$ following the covariance function defined using the Mercer kernel (see Equation 1):

$$\begin{bmatrix} f^T \\ f_t^T \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \Sigma_t & k_t \\ k_t^T & \Sigma_t \end{bmatrix} \right),$$

where

$$k_t = \begin{bmatrix} k(A, A_1) & \ldots & k(A, A_n) \\ k(B, A_1) & \ldots & k(B, A_n) \end{bmatrix}$$

and

$$\Sigma_t = \begin{bmatrix} k(A, A) & k(A, B) \\ k(B, A) & k(B, B) \end{bmatrix}.$$

The predictive distribution is then obtained by integrating over the f-space:

$$\mathbb{P}(f_t|R) = \int_f \mathbb{P}(f_t|f)\mathbb{P}(f|R)df,$$

and can be simplified into a gaussian $\mathcal{N}(\mu^*, \Sigma^*)$ where

$$\mu^* = k_t^T \Sigma^{-1} f_{map},$$
$$\Sigma^* = \Sigma_t - k_t^T (\Sigma_t + \Lambda_{map}^{-1})^{-1} k_t.$$

A prediction is made by computing

$$\mathbb{P}(f(A) > f(B)) = \int_{f_t} \mathbb{P}(A \succ B|f_t)\mathbb{P}(f_t|R)df_t.$$

We predict $A \succ B$ if this probability is greater than a threshold (generally 0.65).

4.3 Evaluation Metrics

An important aspect of the ROR approach is that it may not always provide a prediction for a given pair of alternatives. As a result, the evaluation of the ROR approach requires a special set of metrics to assess and compare the quality of the trade-off between the number of predictions and their accuracy. In this section, we outline the specific metrics that will be used to evaluate the ROR approach and compare it to other methods. To define our metrics we consider the 9 cases that can occur in the confusion matrix defined below.

Confusion Matrix. For a given pair of alternatives $(A, B) \in \mathcal{A}^2$ each model could either infer that A is *better* than B ($A \succ B$), or that A is *worse* than B ($B \succ A$) or it could return that the relation between A and B is *unknown*. Then, as outlined in the section 4.1, by comparing $r(A)$ and $r(B)$, we can have that A is indeed better than B if $r(A) > r(B)$ or that A is worse than B if $r(B) > r(A)$ or that the relation between them is unknown if they belong to the same class. Our metrics will then be based on the following confusion matrix where the rows symbolizes the predicted output and the columns the real outputs.

Predicted/Real	(B)etter	(W)orst	(U)nknown
(B)etter	BB	BW	BU
(W)orst	WB	WW	WU
(U)nknown	UB	UW	UU

Table 1. Confusion Matrix.

We now present the main metrics that we use in the experiments.

Precision. The precision is defined as the ratio between the number of correct predictions among all the predictions that *were* made.

$$P = \frac{BB + WW}{BB + WW + BW + WB + BU + WU}.$$

Recall. The recall is defined as the ratio between the number of correct predictions among all the predictions that *could be* made.

$$R = \frac{BB + WW}{BB + WW + BW + WB + UB + UW}.$$

The precision metric thus penalizes the models that make unreliable predictions, while the recall metric penalizes the models that avoid making predictions.

F-Score. F-score is a metric that combines precision and recall to provide a balanced evaluation of a model's performance. It is obtained by computing the harmonic mean of precision and recall:

$$F = 2\frac{P \times R}{P + R}.$$

As the F-score captures both precision and recall, it is an ideal metric for evaluating the robustness and accuracy of the studied models. Hence, we strongly rely on it when presenting our figures.

Prediction Correctness. This metric is similar to precision, except that it does not take into account predictions that cannot be evaluated because we do not have preference information to verify if they are correct or incorrect.

$$PC = \frac{BB + WW}{BB + WW + BW + WB}.$$

Prediction Rate. This metric does not take into account the correctness of the predictions, it simply evaluates the probability that the model produces predictions:

$$PR = 1 - \frac{UB + UW + UU}{M},$$

where M represents all the cases of Table 1 ($BB + WW + BW + WB + BU + WU + UB + UW + UU$).

5 Numerical Results

The numerical tests, the results of which we present here, were carried out on Google Colab (2 virtual CPU at 2.2GHz, 13GB RAM) using synthetic data generated as detailed above. The code is available online at https://github.com/ouaguenouni/PMTK-GaussianProcess.

5.1 Experiment 1

This first experiment aims to compare the results obtained with our approach using θ_{map} and the results obtained by robust ordinal regression with θ_d, where we recall that d is the degree of the preferences in R. Since the preferences are generated using a θ-additive model, we also use the model θ that was used to produce the preference data in R, denoted by θ_{true}.

Experimental Setting. The following parameters were used for generating the preference data and implementing the MCMC part of the learning process:

- For generating the preference data, we set $|\mathcal{F}| = 8$, $\alpha = p = 0.1$, $t = 12$ (number of classes, see Section 4.1) and $\gamma = 0.5$ (scale parameter, see paragraph "Prior on θ" of Section 3.3).
- In the MCMC part of the learning process, 1000 utility functions f were sampled and, for each of them, 1000 weight vectors. These vectors were used to sample 1000 models θ with the posterior distribution $\mathbb{P}(\theta|R)$ and with a burn-in period of 200. The value of ϵ was set to 1e-3 using trial-and-error iterations, by looking for a value ensuring that when $|w_S| \leq \epsilon$, the parameter w_S can be safely assumed as not significant.

The tests were conducted on 400 datasets R. The size of R varies from 10 to 600 pairwise comparisons. Remark that $|\mathcal{F}| = 8$ implies a total of 32640 possible pairwise comparisons between subsets of $\mathcal{F}$; 600 comparisons is less then 2% of this total number. After each training, the models were evaluated by considering the predictions on the comparisons of 100 pairs of alternatives.

Results and discussions. The results are summarized in Table 2. To facilitate our analysis, we divided our dataset of 600 preference sets into three groups based on their size. The first group contains datasets involving between 10 and 200 preference, the second group contains datasets involving between 200 and 400 preferences, and the third group contains datasets with more than 400 preferences.

In this experiment, we focus on two key metrics to evaluate the performances of our models: Prediction Rate and Prediction Correctness. We see in Table 2 that the predictions made using θ_{map} are more frequent but less accurate than those made using θ_d. Thus θ_{map} offers an interesting tradeoff between the prediction rate and the prediction correctness. This is due to the fact that the polyhedron of compatible utilities yielded by the use of θ_{map} is contained in the one yielded by the use of θ_d (with a few exceptions that happen when θ_{map} is not compatible with R and the set R is post-treated to remove some preferences). Therefore, the predictions that are made using the robust ordinal regression with θ_d are most of the time also made using θ_{map}. However, as the dataset size increases, the loss in accuracy attributable to θ_{map} progressively decreases while the gain in prediction rate increases resulting in an even more favourable tradeoff between prediction rate and accuracy for θ_{map} compared to θ_d for datasets above 200 preferences. It is worth noting that although θ_{true} is the model used to generate the data, it achieves the worst compromise and especially for small datasets (less than 200 preferences) where the prediction rate is only about 6%. This has to do with the fact that the polyhedron of compatible utilities is much larger for θ_{true} than for θ_{map} and θ_d, which badly impacts the prediction rate.

	$\|R\| \in [10, 200]$			$\|R\| \in [200, 400]$			$\|R\| \in [400, 600]$		
	θ_{map}	θ_d	θ_{true}	θ_{map}	θ_d	θ_{true}	θ_{map}	θ_d	θ_{true}
PR	0.22	0.21	0.06	0.70	0.69	0.34	0.75	0.70	0.41
PC	0.96	0.97	1.00	0.98	0.99	1.00	0.99	0.99	1.00

Table 2. Average Prediction Rate and Prediction Correctness for ROR using several definitions of θ (with $|\mathcal{F}| = 8$, thus 32640 possible comparisons).

5.2 Experiment 2

The second experiment aims to compare the results obtained with our approach (ROR) using θ_{map} with the results obtained with the baseline models presented in Section 4.2. More precisely, we use well-identified metrics to evaluate the tradeoff between the quantity and the quality of predictions, and we study the evolution of this tradeoff w.r.t. the size of the data.

Experimental Setting. The second experiment was conducted with the same settings as the first one: we set $|\mathcal{F}| = 8$, $\alpha = p = 0.1$, $t = 12$ (number of classes, see Section 4.1) and $\gamma = 0.5$ (scale parameter, see paragraph "Prior on θ" of Section 3.3). The size of R is also between 10 and 600. This time we chose to keep the model θ_{map} for the SVC and the ROR approach, while the GP approach is model-independent.

	Prediction Rate	Precision	Recall	F-score
ROR	0.75	0.816	0.912	0.85
SVC	1.0	0.679	0.982	0.799
GP	1.0	0.676	0.978	0.796

Table 3. Average performances for different metrics measured on preferences set R with $|R| \in [250, 600]$ and $|\mathcal{F}| = 8$.

Results and discussion. The obtained results depend on the size $|R|$ of the dataset:

- Figures 1 and 2 illustrate the evolution of the metrics when the size $|R|$ varies between 10 and 250 (the curves show the mean and 95% confidence interval). From Figure 1 we can see that our approach performs relatively poorly on very small datasets (remember that 250 comparisons is less than 0.7% of all possible comparisons) in terms of the tradeoff between the quantity and the quality of predictions. However, as the dataset size increases, the quality of the tradeoff improves significantly. Figures 2-a and 2-b allow to better understand the reasons for this evolution. They indeed show that the correctness of the predictions made by ROR consistently outperforms that of the other approaches. Interestingly, as the data set size increases, our approach (ROR) makes more predictions, which leads to an improvement in the quality of the tradeoff, reflected by the F-score.
- Generally speaking, as the size of R increases, there is a corresponding increase in all the metrics for the ROR model. Nevertheless, our tests demonstrate that once we reach a certain percentage threshold for the size of R (that seems to be around 0.6% to 0.7%, meaning 200-250 comparisons for $|\mathcal{F}| = 8$), the results begin to exhibit minimal variations. For this reason, we opted to present the results in an aggregated manner, for $|R|$ ranging from 250 to 600 (equivalent to 0.7% to 1.8% of the total number of comparisons). As Table 3 shows, our approach significantly outperforms the other models in terms of the F-score. Although our ROR approach, being robust, generates fewer predictions than SVC or Gaussian Process models, it achieves a prediction rate of 75% with less than 2% of comparisons in the dataset. Notably, it is considerably more accurate than the baseline methods. These values demonstrate that for problems similar to those of our experiment, ROR offers a superior tradeoff between doing predictions and being correct.

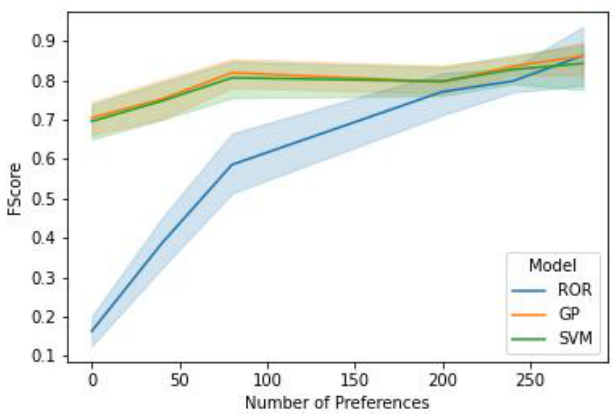
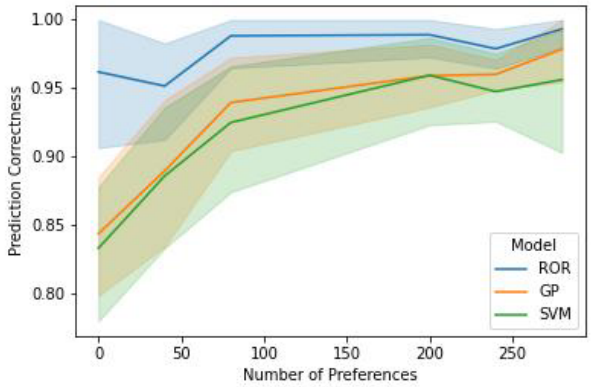


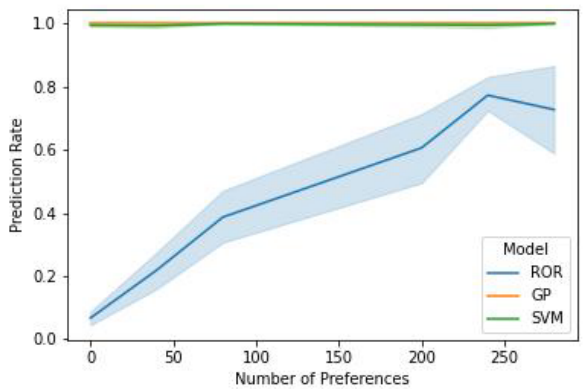
Figure 1. F-score obtained with ROR (blue), GP (orange), SVM (green), with $|\mathcal{F}| = 8$, as a function of the size of the training set R. The shaded area represents 95%-confidence intervals.

6 Conclusion

In this work, we have proposed a hybrid approach to preference learning, combining a GP method to learn the decision model representing the preferences of the decision maker and a robust ordinal regression approach to infer preferences using the learnt model. This combination provides two benefits. First, instead of using an a priori chosen decision model, we learn the specific form of decision model fitting the preferences expressed by the decision maker. Second, we profit from the reliability of the robust ordinal regression methodology when inferring preferences. The results of our numerical tests have shown the effectiveness and robustness of this hybrid approach.



(a) Prediction correctness



(b) Prediction rate

Figure 2. Evolution of the compromise quantity/quality of predictions with ROR (blue), GP (orange), SVM (green), for $|\mathcal{F}| = 8$, as a function of the size of the training set R. The shaded area represents 95%-confidence intervals.

Several directions are conceivable for future work. A first direction would be to consider how to ask efficiently new preferential information to the decision maker to improve our preference learning abilities, i.e., turning our approach into an active learning one. A second direction is related to the reliability of the preferential information provided by the decision maker. Indeed, the robust ordinal regression part of the method does not allow for errors from the decision-maker. An interesting challenge would be to adapt the robust ordinal regression phase in order to soften some constraints to allow for errors.

Acknowledgements

We acknowledge the support of the French Agence Nationale de la Recherche (ANR), under grant ANR20-CE23-0018 (THEMIS).

References

[1] Silvia Angilella, Salvatore Corrente, and Salvatore Greco, 'Stochastic multiobjective acceptability analysis for the Choquet integral prefer-

ence model and the scale construction problem', *European Journal of Operational Research*, **240**(1), 172–182, (January 2015).
[2] Edwin M Bartee, 'Problem solving with ordinal measurement', *Management Science*, **17**(10), B–622, (1971).
[3] 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).
[4] Damien Bigot, Hélène Fargier, Jérôme Mengin, and Bruno Zanuttini, 'Using and learning GAI-decompositions for representing ordinal rankings', in *ECAI'2012 Workshop on Preference Learning, PL 2012, Montpellier, France*, pp. 5–10, (2012).
[5] Roman Bresson, Johanne Cohen, Eyke Hüllermeier, Christophe Labreuche, and Michèle Sebag, 'Learning 2-additive hierarchical Choquet integrals with non-monotonic utilities', in *Workshop from Decision Aiding to Preference Learning, DA2PL 2020, Trento, Italy*, (2020).
[6] Eric Brochu, Nando de Freitas, and Abhijeet Ghosh, 'Active preference learning with discrete choice data', in *Proceedings of the 21st Conference on Neural Information Processing Systems, NeurIPS 2007, Vancouver, British Columbia, Canada*, pp. 409–416, (2007).
[7] Siddhartha Chib, 'Markov chain Monte Carlo methods: computation and inference', *Handbook of econometrics*, **5**, 3569–3649, (2001).
[8] Wei Chu and Zoubin Ghahramani, 'Preference learning with Gaussian processes', in *Proceedings of the 22nd International Conference on Machine Learning, ICML 2005, Bonn, Germany*, pp. 137–144, (2005).
[9] Salvatore Corrente, Salvatore Greco, Miłosz Kadziński, and Roman Słowiński, 'Robust ordinal regression in preference learning and ranking', *Machine Learning*, **93**(2), 381–422, (2013).
[10] Salvatore Corrente, Salvatore Greco, and Roman Slowinski, 'Handling imprecise evaluations in multiple criteria decision aiding and robust ordinal regression by n-point intervals', *Fuzzy Optimisation and Decision Making*, **16**(2), 127–157, (2017).
[11] Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri, 'Learning with rejection', in *Proceedings of the 27th International Conference on Algorithmic Learning Theory, ALT 2016, Bari, Italy*, pp. 67–82, (2016).
[12] Corinna Cortes and Vladimir Vapnik, 'Support-vector networks', *Machine Learning*, **20**(3), 273–297, (September 1995).
[13] Carmel Domshlak and Thorsten Joachims, 'Unstructuring user preferences: efficient non-parametric utility revelation', in *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence, UAI 2005, Edinburgh, Scotland*, pp. 169–177, (2005).
[14] Peter C. Fishburn, *Utility theory for decision making*, Wiley, New York, 1970.
[15] Peter C. Fishburn and Irving H. Lavalle, 'Binary interactions and subset choice', *European Journal of Operational Research*, **92**, 182–192, (1996).
[16] Hugo Gilbert, Mohamed Ouaguenouni, Meltem Öztürk, and Olivier Spanjaard, 'Cautious learning of multiattribute preferences', in *IJCAI'2022 Multidisciplinary Workshop on Advances in Preference Handling, M-PREF 2022, Vienna, Austria*, (2022).
[17] 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).
[18] Salvatore Greco, Vincent Mousseau, and Roman Slowinski, 'Robust ordinal regression for value functions handling interacting criteria', *European Journal of Operational Research*, **239**(3), 711–730, (2014).
[19] Margot Herin, Patrice Perny, and Nataliya Sokolovska, 'Learning sparse representations of preferences within Choquet expected utility theory', in *Proceedings of the 38th Conference on Uncertainty in Artificial Intelligence, UAI 2022, Eindhoven, The Netherlands*, pp. 800–810, (2022).
[20] Eyke Hüllermeier and Klaus Brinker, 'Learning valued preference structures for solving classification problems', *Fuzzy Sets and Systems*, **159**(18), 2337–2352, (2008).
[21] Christophe Labreuche and Michel Grabisch, 'Using multiple reference levels in multi-criteria decision aid: The generalized-additive independence model and the Choquet integral approaches', *European Journal of Operational Research*, **267**(2), 598–611, (2018).
[22] Risto Lahdelma, Joonas Hokkanen, and Pekka Salminen, 'SMAA-stochastic multiobjective acceptability analysis', *European Journal of Operational Research*, **106**(1), 137–143, (1998).
[23] Jean-Luc Marichal and Marc Roubens, 'Determination of weights of interacting criteria from a reference set', *European Journal of Operational Research*, **124**(3), 641–650, (2000).
[24] Christian Robert, George Casella, Christian P Robert, and George Casella, 'Metropolis–Hastings algorithms', *Introducing Monte Carlo Methods with R*, 167–197, (2010).
[25] Luisa M. Zintgraf, Diederik M. Roijers, Sjoerd Linders, Catholijn M. Jonker, and Ann Nowé, 'Ordered preference elicitation strategies for supporting multi-objective decision making', in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden*, pp. 1477–1485, (2018).