Iranian Journal of Fuzzy Systems
Volume 20, Number 3, (2023), pp. 19-30
Original paper

Heuristics-based modelling of human decision process

M. Aggarwal[1] and A. F. Tehrani[2]

[1]*School of Artificial Intelligence and Data Science, IIT Jodhpur, Jodhpur, India*
[1]*Digital Humanities, IIT Jodhpur, Jodhpur, India*
[2]*Hof University of Applied Sciences, Hof, Germany*

magwaL8@gmail.com, ali.fallah.tehrani@hof-university.de

Abstract

Attitudinal Choquet integral (ACI) is a recent aggregation operator that considers in the aggregation process the criteria interaction and the DM's attitude, both of which are specific to the decision-maker. However, this capability comes at the cost of increased complexity that hinders its applicability in big data analytics. To address the same, in this paper, we explore some heuristics-based forms of the ACI operator, so as to somehow overcome its complexity. We devise new and efficient forms of $\mathcal{ACI}$, and test their validity in the real world datasets, against the backdrop of preference learning.

Keywords: Attitudinal Choquet integral, efficiency, complexity reduction, attitudinal character, multi criteria decision making.

1 Introduction

In human decision making, the aggregation operation is typically required to obtain a representative value [19, 25, 31]. For instance, different criteria evaluations for an alternative are aggregated to obtain a net score that is used to compare the alternatives. There are several aggregation operators that have been proposed in the literature, for example, weighted averaging and geometric mean are the conventional aggregation operators existing since long. Ordered weighted averaging (OWA) operator [17, 30], power averaging [30], and power geometric operators [29] are relatively recent operators that find applications in multi criteria decision making (MCDM).

The discrete Choquet integral [8] is an aggregation function that considers varying degree of interaction among the criteria. Its popularity is only increasing as indicated by many applications and extensions [4, 6, 18, 23, 27]. The criteria can interact positively, indicating synergic or desirable combination; or negatively indicating redundancy or undesirable combination. For instance, while making a choice of house, the criteria *size* and *location*, are complementary, and hence interact positively, *luxury* and *size* may interact negatively for a buyer who is looking for a luxurious but small house. It is shown in [13, 16] that most of the extant aggregation operators are the special cases of it.

It has been applied in policy capturing in strategic decision making [21], analysis of root dispersal with competition among wood species in forests [24], computation of the number of citations [28], clinical diagnosis [26], monitoring of the improvement of an overall industrial performance [7], selection of groups of genes with high classifying power in gene expression data analysis [11], evaluation of discomfort in sitting position when driving a car [15], feature selection [14], and MCDM [12, 13].

The human aggregation process is typically characterized by criteria interaction. However, besides the criteria interaction, the individualistic attitude of a DM adds to the complexity. In [33], Zimmermann empirically demonstrated the effect of criteria interaction and the attitude in experiments involving human subjects. The real subjects were first asked to evaluate different tiles against multiple criteria, and then also asked the overall aggregated evaluation. It was observed that each subject displayed a different compensatory effect in his aggregation process.

Corresponding Author: M. Aggarwal
Received: June 2022; Revised: October 2022; Accepted: November 2022.
https://doi.org/10.22111/IJFS.2023.7636

A tolerant DM stresses on meeting only *some* criteria. In contrast, a perfectionist DM would like *all* criteria to be met. This difference in attitudes also affect their aggregation processes. A tolerant DM has an OR-like or *disjunctive* aggregation, in which the aggregated value is towards the best of the values to be aggregated. Opposite to this, a perfectionist displays an AND-like or *conjunctive* aggregation, with the aggregated value towards the minimum of the arguments of aggregation.

To portray such attitudinal tendencies, Zimmermann [32] combined a T-norm and a T-conorm in a controllable proportion. The attitudinal tendencies of a DM are empirically validated in [10, 20, 22]. Yager [30] developed ordered weighted averaging (OWA) operator with the ability to adjust the DM's attitude through the weight vector. Among the recent operators in this regard are compensative weighted mean [1], generalized compensative weighted mean [2], and attitudinal Choquet integral ($\mathcal{ACI}$) [3].

$\mathcal{ACI}$ is perhaps the most comprehensive of these operators with the ability to represent a DM's attitude and the interaction among the criteria at the same time. It has most of the extant aggregation operators such as Choquet integral [8], weighted mean, compensative weighted mean as its special cases. It gives a range of aggregated values, lying between the min (AND) and max (OR) of the arguments of aggregation, for the same set of arguments, as per a DM's attitudinal character, and the degree of interaction among the criteria. Hence $\mathcal{ACI}$ operator is a very flexible aggregation function and thus theoretically promising to model well the human decision-making. This motivates us to test the validity of $\mathcal{ACI}$ in modelling human aggregation process on real data, which is the concern of the present paper.

1.1 Motivation

The process of modelling human aggregation process, from a machine learning perspective, at times, could be fairly complex. It is due to the fact that a broad learning can only be accomplished with (i) a suitable aggregation operator that can represent the nuances of the human decision-making, and (ii) a suitable machine learning method that leads to a good learning of the parameters of such an aggregation operator. Often, (i) and (ii) are counter-acting to each other. By this, we mean that if select a complex aggregation operator, the learning task also becomes quite complex. Hence, while complex aggregation operators are desirable, the complexity in learning alogorithms, or more specifically, the optimization tasks is undesirable.

For instance, $\mathcal{ACI}$ provides an interesting choice for (i), as it offers wide decision boundaries from linear to highly non-linear, and easily adjustable at the same time by two main parameters namely k-additivity and tolerance parameter. These properties make it quite superior to recognize interdependencies among criteria, but its obvious drawback is its computational complexity. Estimating optimal parameters for this operator requires an exponential number of constraints which must be considered during the optimization procedure. This makes the optimization problem no more tractable.

Hence, the foremost challenge in our objective is to reduce the otherwise high computational complexity of the optimization process. The problem is more pronounced in our case, as it is a constrained optimization with the presence of (possibly conflicting) constraints. In this regard, several approaches have been proposed in the literature. The prominent approaches in this context are :

- to embrace a subset of whole data and utilize it to train a model like cutting plane algorithm with underlying SVM settings, and
- to build an estimation of original model so that it can mimic partially the original model.

1.2 The proposed work

In this paper, we pursue the second approach to learn optimal parameters for $\mathcal{ACI}$. To overcome the difficulty of computational complexity, we present a heuristic, iterative algorithm. The algorithm builds step by step weights for the $\mathcal{ACI}$. More specifically, it starts with a linear $\mathcal{ACI}$, finds optimal weights and uses them to mimic monotonicity constraints. The algorithm proceeds iteratively, using learned weights from the previous step. The algorithm has a built-in check to check at each step, whether to go beyond or the optimum solution has reached from a performance point of view. Though the approach is heuristic but it still guarantees that in the end the learned weights obey monotonicity constraints.

Before delve into details, we discuss Attitudinal Choquet Integral formally in Section II. In Section III, we present our approach for complexity reduction of $\mathcal{ACI}$ operator. Section IV gives the experimental study, and the results of the same are discussed in Section V. Section VI concludes the paper.

2 Attitudinal Choquet integral

2.1 Choquet integral

Choquet Integral ($\mathcal{CI}$) is popular for its ability to consider interaction among the attributes. It is based on the concepts of game and capacity, defined on a set $N = \{1, \ldots, n\}$. A capacity $\mu(\cdot)$ on N is a set function, $\mu : 2^N \to \mathbb{R}$ satisfying $\mu(\emptyset) = 0$. For any two $A, B \subseteq N$, a capacity (or fuzzy measure) μ on N satisfying $\mu(A) \le \mu(B)$ whenever $A \subseteq B \subseteq N$. In particular, it follows that $\mu : 2^N \to [0, +\infty)$. A capacity μ is *normalized*, when $\mu(\emptyset) = 0$ and $\mu(N) = 1$.

Let $\mu(\cdot)$ be a capacity on N. Choquet integral $\mathcal{CI} : [0,1]^n \to [0,1]$ w.r.t. $\mu(\cdot)$ ranges on $[0,1]^n$ and is given as:

$$\mathcal{CI}_\mu(a_1, \ldots, a_n) = \sum_{i=1}^{n} a_{\sigma(i)}(\mu(B_i) - \mu(B_{i+1})), \tag{1}$$

where $\sigma(i)$ indicates a permutation on $\{1, \ldots, n\}$ such that $a_{\sigma(1)} \le \ldots \le a_{\sigma(n)}$; $B_i = \{\sigma(i), \ldots, \sigma(n)\}$; and $B_{n+1} = \emptyset$.

In general a Choquet integral for the values $(x_1, \ldots, x_n)$ can be defined as follows:

$$\mathcal{CI}_\mu(x_1, \ldots, x_n) \tag{2}$$

$$= \sum_{A \subseteq N} \left\{ \min\{x \mid x \in A\} + \sum_{T \subset N \setminus A} (-1)^{|T|} \min\{x \mid x \in A \cup T\} \right\} \cdot \mu(A). \tag{3}$$

This representation has an obvious advantage that it does not require sorted values.

2.2 Attitudinal Choquet integral

In this section we address an extension of the discrete Choquet integral, which empowers the original form of the CI. The new proposal is called Attitudinal Choquet Integral ($\mathcal{ACI}$). Formally, an *attitudinal Choquet integral* of dimension n given a fuzzy measure $\mu(\cdot)$ is a mapping $\mathcal{ACI} : [0,1]^n \to [0,1]$, given as

$$\mathcal{ACI}(x_1,\, x_2,\, \ldots,\, x_n) \;=\; \log_\lambda \left(\sum_{i=1}^{n} [\mu(B_{(i)}) - \mu(B_{(i+1)})] \lambda^{x_{\sigma(i)}} \right), \tag{4}$$

where $\sigma(\cdot)$ indicates a permutation on $\{1, \ldots, n\}$ such that $x_{\sigma(1)} \le \ldots \le x_{\sigma(n)}$; $B_{(i)} = \{\sigma(i), \ldots, \sigma(n)\}$; $B_{(n+1)} = \emptyset$ and moreover $\lambda \in (0, \infty], \lambda \neq 1$. In this setting $\mu(\cdot)$ indicates the degree of attribute interaction, λ tackles the compensation in the aggregation process. The parameter λ, specifies the DM's tolerance, i.e., the higher the values of λ, the more is OR-like of DM. On contrary the lower the value of λ the more is AND-like of DM.

The presentation in (4) has an obvious drawback, namely, from a computation point of view, it is required to sort values. There is a more efficient presentation of this operator, namely the presentation underlying Möbius transformation as follows:

$$\mathcal{ACI}(x_1,\, x_2,\, \ldots,\, x_n) \;=\; \log_\lambda \left(\sum_{T \subseteq N} \mathbf{m}(T) \lambda^{\min\left\{x_i | i \in T\right\}} \right), \tag{5}$$

where $\min\left\{x_i \mid i \in T\right\} := \min\limits_{\left\{i | i \in T\right\}} \{x_i\}$. For sake of simplicity we use the above formalism in this paper.

2.3 Properties of Attitudinal Choquet Integral

Proposition 1. *$\mathcal{ACI}_{\mu,\lambda}$ operator is an aggregation function.*

Proof. Let $\mathbf{x}, \mathbf{y} \in [0,1]^n$ are two pareto-dominant vectors, such that $\mathbf{x} \succeq_n \mathbf{y}$. We prove that for any value of λ,

$$\mathcal{ACI}_{\mu,\lambda}(\mathbf{x}) \ge \mathcal{ACI}_{\mu,\lambda}(\mathbf{y}).$$

First of all, note that $\mathbf{x} \succeq_n \mathbf{y}$ implies that $\min\{x_i \mid i \in T\} \ge \min\{y_i \mid i \in T\}$. To show above claim we take into consideration the following cases:

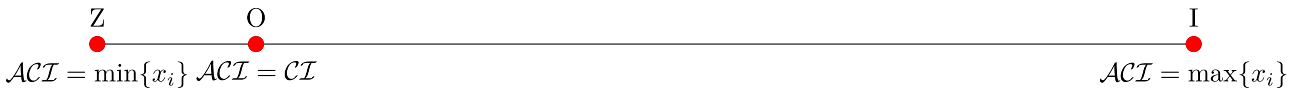

Figure 1: Aggregation Output of $\mathcal{ACI}$. Z, O, and I indicate λ tending to 0, 1, and ∞, respectively.

- $\lambda > 1$ implies that $\lambda^{\min\{x_i|i\in T\}} \geq \lambda^{\min\{y_i|i\in T\}}$ and hence $\sum_{T\subseteq N} \mathbf{m}(T)\lambda^{\min\{x_i|i\in T\}} \geq \sum_{T\subseteq N} \mathbf{m}(T)\lambda^{\min\{y_i|i\in T\}}$. log is an monotonic operator, therefore the following inequality is obtained:

$$\log_\lambda \left(\sum_{T\subseteq N} \mathbf{m}(T)\lambda^{\min\{x_i|i\in T\}} \right) \geq \log_\lambda \left(\sum_{T\subseteq N} \mathbf{m}(T)\lambda^{\min\{y_i|i\in T\}} \right),$$

  which in fact is the form of $\mathcal{ACI}$ operator.

- $0 < \lambda < 1$ implies that $\lambda^{\min\{x_i|i\in T\}} \leq \lambda^{\min\{y_i|i\in T\}}$, hence:

$$\log_\lambda \left(\sum_{T\subseteq N} \mathbf{m}(T)\lambda^{\min\{x_i|i\in T\}} \right) \leq \log_\lambda \left(\sum_{T\subseteq N} \mathbf{m}(T)\lambda^{\min\{y_i|i\in T\}} \right),$$

Now let us assume $\mathbf{x} = (x, \ldots, x) \in [0,1]^n$; therefore $\min\{x_i \mid i \in T\} = x$. All this together we obtain:

$$\log_\lambda \left(\sum_{T\subseteq N} \mathbf{m}(T)\lambda^{\min\{x_i|i\in T\}} \right) = \log_\lambda \left(\lambda^x \sum_{T\subseteq N} \mathbf{m}(T) \right) = \log_\lambda (\lambda^x) + \log_\lambda \left(\sum_{T\subseteq N} \mathbf{m}(T) \right) = x.$$

□

It is interesting to note that λ provides a lever to adjust the degree of compensation in the aggregation process. At the same time, the inherent nature of Choquet integral helps to represent the degree of interaction among the attributes. With each different value of $\lambda \in (0, +\infty)$ results in different aggregation operators with specific compensation degrees. The aggregation output is obtained as the minimum and the maximum of the arguments of aggregation at $\lambda \to 0^+$ and $\lambda \to +\infty$, respectively. The conventional $\mathcal{CI}$ can be obtained from $\mathcal{ACI}_{\mu,\lambda}$ by taking $\lambda \to 1$.

In Figure 1, the topology of different aggregation functions derived from different values of λ are shown. Let $\mathbf{x} = (x_1, \ldots, x_n)$ be a n-tuple of values to be aggregated. $\mathcal{ACI}_{\mu,\lambda}$ could thus be seen as generalizing $\mathcal{CI}$ operator [5].

2.4 Optimal parameters for $\mathcal{ACI}_{\mu,\lambda}$

The latent utility function $U(\cdot)$ is represented through attitudinal Choquet integral. Let the feature vectors of the objects $\mathbf{o} \in \mathcal{O}$ be denoted by:

$$f_{\mathbf{o}} = (f_{\mathbf{o}}(c_1), \ldots, f_{\mathbf{o}}(c_n)),$$

where $f_{\mathbf{o}}(c_i)$ represents evaluation of the object $\mathbf{o}$ for the criterion c_i. That is:

$$U(\mathbf{o}) = \mathcal{ACI}_{\mu,\lambda}(f_{\mathbf{o}}). \tag{6}$$

Assuming the set of pairs

$$\mathcal{P} = \Big\{ (\mathbf{a}_i, \mathbf{b}_i) \mid l_i^a > l_i^b, 1 \leq i \leq N \Big\},$$

is given, where $l_i^{\mathbf{a}}$ and $l_i^{\mathbf{b}}$ are the labels of objects $\mathbf{a}_i$ and $\mathbf{b}_i$ respectively, the question arises, how is possible to find a proper $\mathcal{ACI}_{\mu,\lambda}$, which is most representative for the set $\mathcal{P}$. To this end, we use the same approach in empirical risk minimization to find optimal parameters, basically the idea to maximize the margin between preferred objects to non-preferred objects. Formally, we apply the following algorithm:

$$\max_{M,\xi_1,\ldots,\xi_N} \left\{ M - \frac{\gamma}{|\mathcal{P}|} \sum_{(\mathbf{a}_i,\mathbf{b}_i)\in\mathcal{P}} \left(\xi_i^{\mathbf{a}} + \xi_i^{\mathbf{b}}\right) \right\}, \tag{7}$$

s.t.

$$\mathcal{ACI}_{\mu,\lambda}(f_{\mathbf{a}_i}) - \mathcal{ACI}_{\mu,\lambda}(f_{\mathbf{b}_i}) > M - \xi_i^{\mathbf{a}} - \xi_i^{\mathbf{b}} \qquad \forall (\mathbf{a}_i, \mathbf{b}_i) \in \mathcal{P} \tag{8}$$

$$\xi_i^{\mathbf{a}} \geq 0\,, \quad \xi_i^{\mathbf{b}} \geq 0 \qquad \forall i \in \{1,\ldots,N\}, \tag{9}$$

$$\lambda > 0 \qquad \lambda \neq 1, \tag{10}$$

$$\sum_{T\subseteq X} \mathbf{m}(T) = 1 \tag{11}$$

$$\sum_{B\subseteq A} \mathbf{m}(B) \geq 0 \qquad \forall A \subseteq X \tag{12}$$

$$\sum_{L\subseteq A} \mathbf{m}(L) \leq \sum_{K\subseteq B} \mathbf{m}(K) \qquad \forall A \subset B \subseteq X. \tag{13}$$

Here M stands for margin, $\xi_i^{\mathbf{a}}$'s and $\xi_i^{\mathbf{b}}$'s are slack variables related to soft margin, and γ is a trade-off parameter. Since typically we cannot meet conditions that the data is separable by $\mathcal{ACI}_{\mu,\gamma}$, we allow the ranker (in this case $\mathcal{ACI}_{\mu,\gamma}$) to make errors, although as much as possible small. The so called trade-off parameter γ controls the flexibility of the model; the higher the γ the higher the slacks punished. The constraints in (10) – (13) in essence assure that, the learnt ranker is a $\mathcal{ACI}_{\mu,\gamma}$. The optimization problem is accomplished by a non linear constrained optimization problem.

3 A Heuristics-based approach for learning parameters of ACI

As mentioned earlier, the computational complexity of ACI grows exponentially which restricted this operator just for small data-sets in terms of the number of attributes. Here we propose a heuristic level-wise approach to accomplish this barrier. In fact, in each level we use the prior information, namely, the weights learned in a previous level to build an upper level. The crux idea, indeed, is motivated by the structure of the fuzzy measure or let say its Möbius transformation. More concretely, we start with ACI under 1-additivity, in this case we require to ensure that all weights are positive and the sum of weights is equal to one, i.e.

$$\log_\lambda \big(\sum_{i=1}^{n} \mathbf{m}(\{x_i\}) \lambda^{x_i} \big), \tag{14}$$

$$\text{such that} \quad \mathbf{m}(\{x_i\}) \geq 0, \quad \sum_{i=1}^{n} \mathbf{m}(\{x_i\}) = 1. \tag{15}$$

Now assume that the optimal weights for 1-additive $\mathcal{ACI}$ (linear form) from setting in (7) have been already learned, in this setting the constraints in (11) – (13) are replaced by $\mathbf{m}(\{x_i\}) \geq 0, \quad \sum_{i=1}^{n} \mathbf{m}(\{x_i\}) = 1$.
To extend the above setting for 2-additive $\mathcal{ACI}$ we need to introduce 2-additive Möbius transformation in more details; in 2-additive case besides of $\mathbf{m}(\{x_i\}) \geq 0$, the following constraints must be satisfied, namely, for all j, k

$$\mathbf{m}(\{x_k, x_j\}) + \mathbf{m}(\{x_j\}) \geq 0, \tag{16}$$
$$\mathbf{m}(\{x_k, x_j\}) + \mathbf{m}(\{x_k\}) \geq 0. \tag{17}$$

By expanding the above inequality we obtain

$$\mathbf{m}(\{x_k, x_j\}) \geq -\mathbf{m}(\{x_j\}), \tag{18}$$
$$\mathbf{m}(\{x_k, x_j\}) \geq -\mathbf{m}(\{x_k\}). \tag{19}$$

This simply reveals that

$$\mathbf{m}(\{x_k, x_j\}) \geq -\min\big\{\mathbf{m}(\{x_j\}), \mathbf{m}(\{x_k\})\big\}. \tag{20}$$

In other words, the values of 1-additive Möbius transformation give exactly the constraints for building the 2-additive Möbius transformation. In this case, it is required to satisfy $\binom{n}{2}$ constraints which is much less than the number of constraints in the original form. By the following idea we go one level higher, namely, 3-additive $\mathcal{ACI}$. In this case besides of normalization constraint, for all i, j, k the following constrains must be fulfilled:

$$\mathbf{m}(\{x_i, x_j, x_k\}) + \sum_{T \subsetneqq \{x_j, x_k\}} \mathbf{m}(T \cup \{x_i\}) \geq 0, \tag{21}$$

$$\mathbf{m}(\{x_i, x_j, x_k\}) + \sum_{T \subsetneqq \{x_i, x_k\}} \mathbf{m}(T \cup \{x_j\}) \geq 0, \tag{22}$$

$$\mathbf{m}(\{x_i, x_j, x_k\}) + \sum_{T \subsetneqq \{x_j, x_i\}} \mathbf{m}(T \cup \{x_k\}) \geq 0. \tag{23}$$

Again the above constraints simply express that:

$$\mathbf{m}(\{x_i, x_j, x_k\}) \geq -\min\Bigg\{ \sum_{T \subsetneqq \{x_j, x_k\}} \mathbf{m}(\{T \cup \{x_i\}), \tag{24}$$

$$\sum_{T \subsetneqq \{x_i, x_k\}} \mathbf{m}(\{T \cup \{x_j\}), \tag{25}$$

$$\sum_{T \subsetneqq \{x_i, x_j\}} \mathbf{m}(\{T \cup \{x_k\})\Bigg\}. \tag{26}$$

Since all values with respect to the 2-additive Möbius transformation have been already given, in 3-additive case it is required to satisfy $\binom{n}{3}$ constraints. Please note that the other constraints have been already in lower levels satisfied. In fact, we are answering the question, given a 2-additive fulfilled, how we can build the 3-additive fulfilled. We follow this idea to build constraints for the level k-th, in this case, for any $S \subset \{x_1, \ldots, x_n\}$, where $\mid S \mid = k$ the following constraints must be taken into account:

$$\mathbf{m}(S) \geq -\min\Bigg\{ \sum_{T \subsetneqq K} \mathbf{m}\big(T \cup \Delta\big) \Bigg| K \subsetneqq S, \mid P \mid = \mid K \mid -1, \Delta = S \setminus K \Bigg\}.$$

In other words, for the level k (k-additive case), $\binom{n}{k}$ constrains must be considered. In addition, in each step weights must be normalized. Needless to say that the original representation of full ACI requires $n \cdot 2^n$ constrains, whereas our proposal needs just a quadratic complexity.

The algorithm starts with 1-additive $\mathcal{ACI}$, learns weights and hereinafter builds new constraints for the 2-additive case. For 2-additive case the singletons $\mathbf{m}(\{c_i\})$ have been already identified, thus the algorithm requires to just find $\mathbf{m}(\{c_i, c_j\}), \forall i, j$. Now the weights for 2-additive $\mathcal{ACI}$ have been estimated. The algorithm again uses the learned weights and b build new constraints to identify $\mathbf{m}(\{c_i, c_j, c_k\}), \forall i, j, k$. The algorithm terminates after k-steps and delivers the k-additive $\mathcal{ACI}$ and in addition corresponding fuzzy measure.

4 Experimental setting

4.1 Data

We use the real datasets in our experiments, which are available in UCI and Weka repositories. Since monotonicity is a basic property in all the aggregation operators, all the chosen datasets are monotone in the sense that "the more the better". A brief discription of the datasets is as follows.

- **Employee Selection (ESL):** The dataset gives the details of different applicants who have applied for some industrial jobs. The values of the four input criteria were determined by expert psychologists based upon psychometric test results and interviews with the candidates. The output is an overall score on an ordinal scale between 1 and 9, corresponding to the degree of suitability of each candidate to this type of job.

- **Employee Rejection/Acceptance (ERA):** This dataset pertains to decision-making in an academic job. The input criteria are features of a candidate such as past experience, verbal skills, etc., and the output is the subjective judgment of a decision-maker, measured on an ordinal scale from 1 to 9, to which degree he or she tends to accept the applicant for the job.

Algorithm 1 Heuristic Algorithm for $\mathcal{ACI}$

Input: DATA, λ, ν, k

Output: k-additive Möbius Transformation, k-additive $\mathcal{ACI}$-operator

for $i = 1$ to k **do**

for all $S \subset \{x_1, \ldots, x_n\}$, where $|S| = i$ build the following constraints:

$$\mathbf{m}(S) \geq -\min\left\{ \sum_{T \subsetneq K} \mathbf{m}\big(T \cup \Delta\big) \middle| K \subsetneq S, |K| = |S| - 1, \Delta = S \setminus K \right\}.$$

Learn the i-additive ACI operator by supposing above constraints and considering learned weights from $i-1$-additive ACI as follows:

$$\max_{M, \xi_1, \ldots, \xi_N} \left\{ M - \frac{\gamma}{|\mathcal{P}|} \sum_{(\mathbf{a}_i, \mathbf{b}_i) \in \mathcal{P}} \left(\xi_i^{\mathbf{a}} + \xi_i^{\mathbf{b}} \right) \right\}$$

s.t.

$$\mathcal{ACI}_{\mu^i,\lambda}(f_{\mathbf{a}_i}) - \mathcal{ACI}_{\mu^i,\lambda}(f_{\mathbf{b}_i}) > M - \xi_i^{\mathbf{a}} - \xi_i^{\mathbf{b}}, \qquad \forall (\mathbf{a}_i, \mathbf{b}_i) \in \mathcal{P}$$

$$\xi_i^{\mathbf{a}} \geq 0\ , \quad \xi_i^{\mathbf{b}} \geq 0, \qquad \forall i \in \{1, \ldots, N\}$$

$$\lambda > 0, \lambda \neq 1$$

$$\mathbf{m}(S) \geq -\min\left\{ \sum_{T \subsetneq K} \mathbf{m}^*\big(T \cup \Delta\big) \middle| K \subsetneq S, \mid K \mid = \mid S \mid -1, \Delta = S \setminus K \right\}, \ \ \forall S \subseteq \{x_1, \ldots, x_n\}, |S| = i$$

$$\Bigg(\text{Note that } \ \mathcal{ACI}_{\mu^i,\lambda}(x_1, \ldots, x_n) := \log_\lambda \left(\sum_{T \subseteq N, |T| < i} \mathbf{m}^*(T) \lambda^{\min\left\{x_i | i \in T\right\}} + \sum_{T \subseteq N, |T| = i} \mathbf{m}(T) \lambda^{\min\left\{x_i | i \in T\right\}} \right),$$

where $\mathbf{m}^*(\cdot)$ has been computed in the previous steps.$\Big)$

$\mathbf{m}^* \leftarrow \mathbf{m}^* \cup \mathbf{m}^\diamond$, where $\mathbf{m}^\diamond$ is the solution of the above optimization setting. Note that $\mathbf{m}^\diamond$ considers only the elements which have a cardinality of i and the junction considers all elements which have a cardinality less than $i+1$.

$\mathbf{m}^* \leftarrow \frac{\mathbf{m}^*}{\sum_T m^*(T)}$.

end for

return Möbius transform, k-additive $\mathcal{ACI}$

- **Lecturers Evaluation (LEV):** Lecturer evaluations for MBA courses are given in this dataset. Students were asked to score their lecturers according to four criteria such as oral skills and contribution to their professional/general knowledge. The output was a total evaluation of each lecturer's performance, measured on an ordinal scale from zero to four.

- **Mammography (MMG):** The mammography for breast cancer constitutes this dataset. The severity of the breast cancer is given for different combinations of the criteria values such as mass shape, mass margin, density, and the patient's age.

- **CPU:** There were 10 criteria, out of which 2 criteria - vendor name and model name - were eliminated, being of no predictive value. The relative performance of a CPU is given based on the following criteria: machine cycle (nanoseconds), minimum main memory (kilobytes), maximum main memory (kilobytes), cache memory (kilobytes), minimum channels (units), and maximum channels (units).

- **Car Evaluation (CEV):** The overall rating of different car models is given as unacceptable, acceptable, good, and very good based on the following criteria: buying price, maintenance price, number of doors, number of seats, size of luggage boot, and estimated safety.

data set	#instances	#criteria	#classes	source
Employee Selection (ESL)	488	4	4	WEKA
Employee Rejection \ Acceptance (ERA)	1000	4	4	WEKA
Lecturers Evaluation (LEV)	1000	4	3	WEKA
Mamographic (MMG)	830	5	2	UCI
CPU	209	6	2	UCI
Car Evaluation (CEV)	1728	6	4	UCI
Breat-Cancer(BCC)	286	7	2	UCI
DenBosch	120	8	2	[9]
Auto MPG	398	8	6	UCI

Table 1: Data sets and their properties

- **Breast Cancer (BCC):** This dataset originally pertains to the University Medical Center, Institute of Oncology, Ljubljana. A collection of criteria such as menopause gain, tumor-size, inv-nodes, node-caps, degree of malignancy, etc. is mapped to no-recurrence and recurrence cases.
- **DenBosch (DBS):** The houses in Den Bosch are classified as a low or high priced based on the following criteria: district, area, number of bedrooms, type of house, volume, storeys, type of garden, garage, and price.
- **Auto MPG:** The city-cycle fuel consumption (miles per gallon) is given for different combinations of the criteria: mpg, cylinders, displacement, horsepower, weight, acceleration, model year and origin.

We give an overview of these datasets in Table 1.

4.2 Experiment steps

We implement weighted averaging (WA), Choquet integral (CI), $\mathcal{ACI}$ and finally our heuristic methods on a set of nine datasets shown in Table 1 by performing the following steps:

1. Given a set of alternatives $\mathcal{A}$, two halves are created, namely $\mathcal{A}_{train}$ and $\mathcal{A}_{test}$, for training and testing, respectively.
2. The training information is provided in terms of the pairwise preference pairs of the form $\mathbf{a}_n \succ \mathbf{b}_n$ are identified through the standard method of random sampling from $\mathcal{A}_{train}$ The number of preferences are taken as $N = 1000$. For a healthy comparison, the same set of N random preferences is provided to different methods, in each iteration.
3. With different approaches, different learning models are induced on the given preference pairs.
4. The ranks of the given alternatives in $\mathcal{A}_{test}$ are predicted through WA, CI, and $\mathcal{ACI}$ methods.
5. The corresponding prediction accuracies are evaluated by comparing the predicted ranking for $\mathcal{A}_{test}$ through C-index.
6. For sanctity, the process is repeated 100 times for each dataset and for each approach.
7. The average of the 100 accuracy values, so obtained, is taken as the accuracy value. The respective standard deviation is also determined in the multiple accuracy values. The figures are shown in Table 2.

4.3 Simulation environment

The proposed algorithm launches a linear and monotone model; in this case since the objective function is linear the *linprog* solver from the MATLAB has been considered. In addition, the *linprog* solver can deal with equalities and inequalities constraints, which certainly is required in this case. The equality constraint considers the normalization (sum of all weights equal to one) and eventually inequality constraints are assigned to the preferences. They are constraints in (8) imposing that the pairwise preferences in the training data are fulfilled. In this case, the maximum number of function evaluations and the maximum number of iterations are set to 10^6 and $5 * 10^5$, respectively. Depending on the number of predictive variables one can apply *LargeScale* option.
The *LargeScale* option in our case is preponderate and consequently can overcome the large number of learning attributes better during the optimization process. As already mentioned, in the paper the $\mathcal{ACI}$ underlying its Möbius

transformation is taken into consideration, that is, the learned weights derived from the algorithm are immediately transformed for generating intermediate constraints. To this end, once the algorithm terminates the upper and lower bounds constraints are generated. After termination of this phase, the algorithm begins with the next layer, namely, uses the similar setting, however, this time enforces more constraints in terms of lower and upper bounds which derived from the previous phase.

The intermediate constraints, then are used to update the weights through the linprog setting. The process is done successively until the number of k exceeds. Once the algorithm fully terminated, it reveals the $\mathcal{ACI}$ and the corresponding Möbius transformation. In each trial, the values of ν and λ are chosen through a nest cross validation based on training data. In this regard, while the main concern of the computed results is to investigate the efficiency of the proposed heuristic approach in terms of the runtime, yet, another off-shoot of the study is the gain, i.e., the results in general confirm that the proposed approach is a suitable surrogate for the original $\mathcal{ACI}$ approach.

More concretely, for the datasets MMG, CEV, BCC and MPG the heuristic method offers a significant better results versus the original formalism. The reason perhaps lies in the nature of the datasets, that is, the datasets equipped with a large number of attributes have an exponential number of variables in the extended feature space. The exponential number of attributes in the lack of enough number of data-points leads to an overfitting effect, that is, the model fits on training data perfectly but has a poor performance on the test data, however, contrary to this defeat, the heuristic approach attempts to learn proper weights levelwise which at each level the solver deals with a limited number of variables. In this regard, this limitation, in fact, can mimic a regularization effect and hence the proposed approach arguably estimates proper weights in a more significant way.

4.4 Performance evaluation

The scoring function $U(\mathbf{a})$ helps to predict the rank of an alternative $\mathbf{a} \in \mathcal{A}_{test}$.

The predicted rankings are compared with the ground truth ranking for the given alternatives in $\mathcal{A}_{test}$.

The performance of a method is determined by computing the degree of match between the predicted and ground truth ranks.

Based on the ordinal ground-truth rankings, an ordered partition of $\mathcal{A}_{test}$ can be formed shown as $\underline{\mathcal{A}}_{test} = \{\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_k\}$, where $\mathcal{A}_i = \{\mathbf{a} \in \mathcal{A}_{test} \mid L(\mathbf{a}) = L_i\}, i = 1, \dots, p$.

C-index is a measure of such degree of match (or mismatch) between the true and the predicted rankings. It is shown as follows:

$$C(U, \mathbf{A}) = \frac{1}{\sum_{i<j} |\mathbf{A}_i| \cdot |\mathbf{A}_j|} \sum_{1 \le i < j \le k} \sum_{(\mathbf{a},\mathbf{b}) \in \mathbf{A}_i \times \mathbf{A}_j} S(U(\mathbf{a}), U(\mathbf{b})), \tag{27}$$

where $\mathbf{A}_i$ is the subset of objects $\mathbf{A}$ whose true class is l_i and

$$S(d, e) = \begin{cases} 1 & \text{if } d > e, \\ 0 & \text{otherwise.} \end{cases} \tag{28}$$

5 Results

Table 2 presents the comparison in terms of gain between the original $\mathcal{ACI}_{\mu,\lambda}$ and its surrogate which is assigned by *. To this end, we have considered 2-additive, 3-additive and finally full $\mathcal{ACI}$ heuristic versions.

Table 2 illustrates a significant improvement over the original version in many of the datasets considered. This can be explained by the fact that the heuristic version requires less complexity during *optimization process*, which obviously reduces the complexity of efforts and therefore quickly leading to a precise solution. This is especially interesting from an optimization point of view.

Another advantage of this setting is in its efficient run-time. We have simultaneously measured run-time of each of the methods. Table 3[1] shows the average run-time of the given methods in milliseconds. For datasets with large number of attributes, the proposed heuristic approach is superior in terms of the runtime. However, for datasets with not so large number of attributes this difference is not noticeable.

Note that, the original setting of $\mathcal{ACI}$ for k-additivity has almost the same runtime like the full-$\mathcal{ACI}$.

[1]Experiments were carried out on an Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz and 16 GB RAM under Windows 10.

datasets \ methods	**WA**	**CI**	$\mathcal{ACI}_{\mu,\lambda}$	$\mathcal{ACI}^*_{\mu,\lambda}$-2-add	$\mathcal{ACI}^*_{\mu,\lambda}$-3-add	$\mathcal{ACI}^*_{\mu,\lambda}$ - full
ESL	.054 ± .022	.048 ± .026	.054 ± .022	.073 ± .037	.059 ± .033	.057 ± .028
ERA	.304 ± .054	.306 ± .051	.290 ± .056	.337 ± .053	.304 ± .055	.309 ± .051
LEV	.165 ± .038	.169 ± .039	.162 ± .036	.149 ± .028	.145 ± .024	.145 ± .024
MMG	.165 ± .139	.156 ± .027	.155 ± .022	.082 ± .068	.082 ±.068	.081 ± .067
CPU	.049 ± .040	.012 ± .014	.010 ± .014	.012 ± .006	.028 ± .023	.025 ± .021
CEV	.132 ± .014	.063 ± .019	.061 ± .022	.162 ± .015	.061 ±.003	.051 ± .010
BCC	.242 ± .107	.362 ± .079	.331 ± .084	.147 ± .090	.171 ±.152	.147 ± .151
DBS	.031 ± .062	.056 ± .065	.027 ± .032	.072 ± .040	.075 ± .034	.077 ± .035
MPG	.039 ± .032	.070 ± .015	.070±.015	.021 ± .018	.025 ± .019	.029 ± .024
Avg. rank						

Table 2: Error in terms of the average C-Index ± standard deviation.

datasets \ methods	$\mathcal{ACI}_{\mu,\lambda}$	$\mathcal{ACI}^*_{\mu,\lambda}$-2-add	$\mathcal{ACI}^*_{\mu,\lambda}$-3-add	$\mathcal{ACI}^*_{\mu,\lambda}$-full
ESL	5.33	5.02	5.32	5.61
ERA	5.64	4.53	4.91	5.28
LEV	4.38	6.19	6.59	6.96
MMG	7.44	6.89	7.24	7.94
CPU	18.71	8.65	9.15	10.61
CEV	20.60	8.41	8.90	10.32
BCC	33.74	11.08	11.63	14.07
DBS	169.57	23.57	24.52	30.62
MPG	45.20	12.50	13.11	15.66

Table 3: Runtime reported in miliseconds

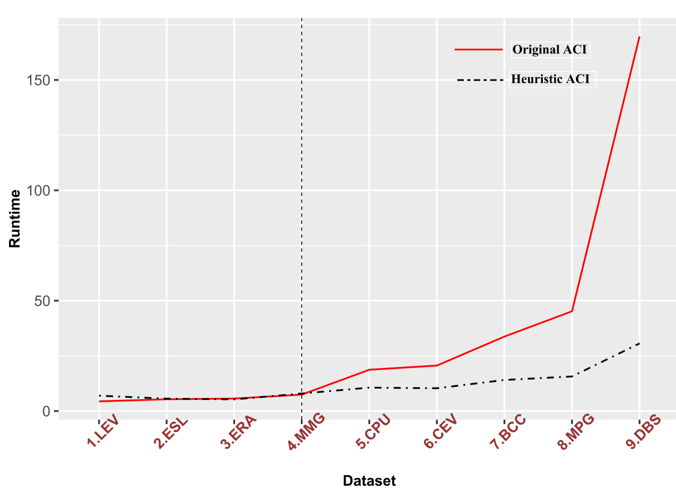

Figure 2: Runtime comparison in term of milliseconds.

Conclusion

The proposed version heuristic form of $\mathcal{ACI}$ offers a unique provision to obtain optimal parameters quickly while preserving the basic advantages of the $\mathcal{ACI}$ operator. The proposed approach is developed so as to reduce the complexity in each step, and to have an overall more favorable optimization settings, which together could lead to finding a better solution efficiently overcoming the complexity of the $\mathcal{ACI}$ in its original form. It is also notable that in the original $\mathcal{ACI}$ setting, the solver is burdened with an exponential number of constraints, which naturally affects the quality of the solution obtained. The proposed form imparts simplicity to the optimization process that can be carried out conveniently resulting in an overall better performance both in terms of the results as well as the time required. It is also interesting to note that despite a large number of attributes (as the case in a few of the chosen datasets), our approach displays almost linear characteristics. Lastly but perhaps more importantly, the weights derived from the proposed approach obey monotonicity property that makes it very much desirable in most of human decision making applications.

As a future work, it would be useful to develop the measures such as Shapley index, interaction index and specifically fuzzy measure for the proposed form of $\mathcal{ACI}$. Besides, it would be worth to investigate the application of the proposed form of $\mathcal{ACI}$ and such measures in a human decision making problem.

Acknowledgement

The authors wish to express their appreciation for several excellent suggestions for improvements in this paper made by the referees.

References

[1] M. Aggarwal, *Compensative weighted averaging aggregation operators*, Applied Soft Computing, **28** (2015), 368-378.

[2] M. Aggarwal, *Generalized compensative weighted averaging aggregation operators*, Computers and Industrial Engineering, **87** (2015), 81-90.

[3] M. Aggarwal, *Attitudinal choquet integrals and applications in decision making*, International Journal of Intelligent Systems, **33**(4) (2018), 879-898.

[4] M. Aggarwal, *Logit choice models for interactive attributes*, Information Sciences, **507** (2020), 298-312.

[5] M. Aggarwal, A. Fallah Tehrani, *Modelling human decision behaviour with preference learning*, INFORMS Journal on Computing, **31**(2) (2019), 318-334.

[6] M. Aggarwal, M. Hanmandlu, K. K. Biswas, *Choquet integral vs. topsis: An intuitionistic fuzzy approach*, In 2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), (2013), 1-8.

[7] L. Berrah, G. Mauris, J. Montmain, *Monitoring the improvement of an overall industrial performance based on a choquet integral aggregation*, Omega, **36**(3) (2008), 340-351.

[8] G. Choquet, *Annales de l institut Fourier*, Theory of Capacities, **5** (1953), 131-295.

[9] H. Daniels, B. Kamp, *Applications of MLP networks to bond rating and house pricing*, Neural Computation and Applications, **8** (1999), 226-234.

[10] H. Dyckhoff, W. Pedrycz, *Generalized means as a model of compensation connectives*, Fuzzy Sets and Systems, **14** (1984), 143-154.

[11] V. Fragnelli, S. Moretti, *A game theoretical approach to the classification problem in gene expression data analysis*, Computers and Mathematics with Applications, **55**(5) (2008), 950-959.

[12] M. Grabisch, *Fuzzy integral in multicriteria decision making*, Fuzzy Sets and Systems, **69**(3) (1995), 279-298.

[13] M. Grabisch, *The application of fuzzy integrals in multicriteria decision making*, European Journal of Operational Research, **89** (1996), 445-456.

[14] M. Grabisch, *Fuzzy measures and integrals - theory and applications, chapter fuzzy integral for classification and feature extraction*, Physica Verlag, (2000), 415-434.

[15] M. Grabisch, J. Duchene, F. Lino, P. Perny, *Subjective evaluation of discomfort in sitting position*, Fuzzy Optimization and Decision Making, **1**(3) (2002), 287-312.

[16] M. Grabisch, J. L. Marichal, R. Mesiar, E. Pap, *Aggregation functions: Means*, Information Sciences, **181** (2011), 1-22.

[17] L. Jin, R. Mesiar, R. R. Yager, *Melting probability measure with owa operator to generate fuzzy measure: The crescent method*, IEEE Transactions on Fuzzy Systems, **27**(6) (2019), 1309-1316.

[18] L. Jin, R. Mesiar, R. R. Yager, *Derived fuzzy measures and derived choquet integrals with some properties*, IEEE Transactions on Fuzzy Systems, **29**(5) (2021), 1320-1324.

[19] L. Jin, R. Mesiar, R. R. Yager, *The properties of crescent preference vectors and their utility in decision making with risk and preferences*, Fuzzy Sets and Systems, **409** (2021), 114-117.

[20] R. Krishnapuram, J. Lee, *Fuzzy-connective-based hierarchical aggregation networks for decision making*, Fuzzy Sets and Systems, **46**(1) (1992), 11-27.

[21] D. Liginlal, T. T. Ow, *On policy capturing with fuzzy measures*, European Journal of Operational Research, **167** (2005), 461-474.

[22] M. K. Luhandjula, *Compensatory operators in fuzzy linear programming with multiple objectives*, Fuzzy Sets and Systems, **8**(3) (1982), 245-252.

[23] R. Mesiar, A. Mesiarová-Zemánková, K. Ahmad, *Discrete choquet integral and some of its symmetric extensions*, Fuzzy Sets and Systems, **184** (2011), 148-155.

[24] W. Nather, K. Walder, *Applying fuzzy measures for considering interaction effects in root dispersal models*, Fuzzy Sets and Systems, **158** (2007), 572-582.

[25] C. Rao, M. Gao, J. Wen, M. Goh, *Multi-attribute group decision making method with dual comprehensive clouds under information environment of dual uncertain z-numbers*, Information Sciences, **602** (2022), 106-127.

[26] K. Saito, Y. Watanabe, H. Hashimoto, A. Uchiyama, *An application of fuzzy integral model for the clinical diagnosis*, Journal of Biomedical Fuzzy Systems Association, **1**(1) (2007), 17-24.

[27] A. F. Tehrani, W. Cheng, K. Dembczyński, E. Hüllermeier, *Learning monotone nonlinear models using the Choquet integral*, Machine Learning, **89**(1-2) (2012), 183-211.

[28] V. Torra, Y. Narukawa, *The h-index and the number of citations: Two fuzzy integrals*, IEEE Transactions on Fuzzy Systems, **16**(3) (2008), 795-797.

[29] Z. Xu, R. R. Yager, *Power-geometric operators and their use in group decision making*, IEEE Transactions on Fuzzy Systems, **18**(1) (2010), 94-105.

[30] R. R. Yager, *On ordered weighted averaging aggregation operators in multi-criteria decision making*, IEEE Transactions on Systems, Man and Cybernetics, **18** (1988), 183-190.

[31] M. Zhang, H. Guo, M. Sun, S. Liu, J. Forrest, *A novel flexible grey multivariable model and its application in forecasting energy consumption in China*, Energy, **239** (2022). DOI: 10.1016/j.energy.2021.122441.

[32] H. J. Zimmermann, P. Zysno, *Latent connectives in human decision making*, Fuzzy Sets and Systems, **4** (1980), 37-51.

[33] H. J. Zimmermann, P. Zysno, *Decisions and evaluations by hierarchical aggregation of information*, Fuzzy Sets and Systems, **10**(1-3) (1983), 243-260.