Please cite this paper as: M. Gagolewski, S. James, G. Beliakov,
Supervised learning to aggregate data with the Sugeno integral, IEEE Transactions on Fuzzy Systems 27(4), 810+815, 2019, doi:10.1109/TFUZZ.2019.2895565

Supervised Learning to Aggregate Data with the Sugeno Integral

Marek Gagolewski, Simon James, and Gleb Beliakov

Abstract—**The problem of learning symmetric capacities (or fuzzy measures) from data is investigated toward applications in data analysis and prediction as well as decision making. Theoretical results regarding the solution minimizing the mean absolute error are exploited to develop an exact branch-refine-and-bound-type algorithm for fitting Sugeno integrals (weighted lattice polynomial functions, max-min operators) with respect to symmetric capacities. The proposed method turns out to be particularly suitable for acting on ordinal data. In addition to providing a model that can be used for the general data regression task, the results can be used, among others, to calibrate generalized h-indices to bibliometric data.**

Index Terms—**Weight learning, ordinal data fitting, fuzzy measures, Sugeno integral, lattice polynomials, h-index**

I. Introduction

The problem of fitting various models (like aggregation functions) to empirical data is of interest in many applications, including machine learning and decision making. In such a task, given a set of prototypical points, we are interested in finding a function that minimizes some measure of discrepancy between the outputs it generates and the desired outputs that are given a priori. Despite the fact that fitting particular classes of means, like OWA operators or other Choquet integrals, has already been covered in the literature, see, e.g., [1], [2], [3], [4], [5], the case of the Sugeno integral fitting has not been covered sufficiently for practical use. In [6], Yuan and Klir proposed an *approximate*, neural network-based algorithm for fitting a Sugeno integral to an empirical data set based on the least squared error criterion. In [7], Anderson, Keller, and Havens considered another *approximate* approach based on a genetic algorithm for fuzzy valued fuzzy measures. Approximate approaches of course do not give any guarantees regarding the true optimality of a solution. To address this issue, Prade, Rico, and Serrurier in [8] studied (in an algebraic framework) families of Sugeno integrals that are compatible with (*interpolate*) given data sets (provided that they exist). Yet, such an approach assumes that the desired outputs were generated by some (unknown) Sugeno integral and were not subject to any error. To address this problem, our recent contribution [9] gave some preliminary results on *exact* least absolute and squared error fitting of symmetric Sugeno integrals, but the algorithm therein proposed was based on a quite straightforward, and hence *very slow*, version of the branch-and-bound approach. In particular, it did not provide any results concerning the location of a minimizer, therefore it did not guarantee its convergence to a single solution within reasonable space and time limits.

In this contribution we would like to fill this gap and develop an exact algorithm that is finally usable in practice. Note that the Sugeno integral has many real-world use cases: for instance, it is often considered in bibliometrics (e.g., [10], [11]), as it generalizes the famous Hirsch's h-index. Moreover, due to the fact that this integral is solely defined based upon lattice operators $\vee$ $(\max)$ and $\wedge$ $(\min)$, as opposed to other models, it may naturally be applied in the context when data are on an *ordinal scale* (i.e., a bounded chain, here, mapped to some elements in the unit interval).

To recall, for a given fuzzy measure (normalized capacity or monotone measure) $\mu : 2^{\{1,2,\dots,n\}} \to [0,1]$, i.e., a set function such that $\mu(\emptyset) = 0$, $\mu(\{1,\dots,n\}) = 1$, $\mu(U) \le \mu(V)$ for $U \subseteq V$, the corresponding discrete Sugeno integral of $\mathbf{x} \in [0,1]^n$ is defined as:

$$\begin{aligned}\mathsf{S}_\mu(\mathbf{x}) &= \textstyle\bigvee_{j=1}^n x_{\sigma(j)} \wedge \mu\left(\{\sigma(1),\sigma(2),\dots,\sigma(j)\}\right)\\ &= \max\Big\{ \min\left\{x_{\sigma(1)}, \mu(\{\sigma(1)\})\right\},\dots,\\ &\qquad \dots, \min\left\{x_{\sigma(n)}, \mu(\{\sigma(1),\dots,\sigma(n)\})\right\}\Big\},\end{aligned}$$

where σ is a permutation of $\{1,2,\dots,n\}$ such that $x_{\sigma(1)} \ge \dots \ge x_{\sigma(n)}$.

To define an arbitrary capacity we need $O(2^n)$ coefficients, therefore, for n of considerable order of magnitude, we usually introduce some additional constraints on μ, like symmetry, k-additivity, k-maxitivity (e.g., [12], [13]), etc. To avoid over-complication, in the current paper we shall focus on perhaps the most user-friendly capacities (leaving the other ones for further research), i.e., symmetric μ such that if $|U| = |V|$, then $\mu(U) = \mu(V)$. Each such μ can be represented as a vector $\mathbf{h}$ of n weights with $0 \le h_1 \le h_2 \le \dots \le h_n = 1$ and $h_j = \mu(\{1,2,\dots,j\})$, yielding the Sugeno integral:

$$\mathsf{S}_{\mathbf{h}}(\mathbf{x}) = \textstyle\bigvee_{j=1}^n x_{\sigma(j)} \wedge h_j, \tag{1}$$

also referred to as the ordered weighted maximum (OWMax) operator, see [14]. Such aggregation functions obey nice properties, including nondecreasingness in each x_i as well as each h_i, internality, idempotence, symmetry, symmetric minitivity, maxitivity, and modularity (see [15]), and so forth.

Each model fitting task requires a training set. Thus, we assume we observe m input sequences, each with n elements in

Marek Gagolewski (corresponding author) is with the Faculty of Mathematics and Information Science, Warsaw University of Technology, ul. Koszykowa 75, 00-662 Warsaw, Poland as well as with the Systems Research Institute, Polish Academy of Sciences, ul. Newelska 6, 01-447 Warsaw, Poland; Email: M.Gagolewski@mini.pw.edu.pl.

Gleb Beliakov and Simon James are with the School of Information Technology, Deakin University, Geelong, Australia; Emails: gleb@deakin.edu.au, sjames@deakin.edu.au.

Manuscript received XX; revised YY.

the unit interval, $\mathbf{X} = [\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}] \in [0,1]^{n \times m}$, together with m corresponding desired output values $\mathbf{Y} = [y^{(1)}, \dots, y^{(m)}] \in [0,1]^{1 \times m}$. We employ the bracket notation so that an observation $y^{(k)}$ is distinguished from y raised to the power of k. We assume each of the input vectors has been ordered nonincreasingly in advance, so that for the k-th observation, $1 \geq x_1^{(k)} \geq x_2^{(k)} \geq \dots \geq x_n^{(k)} \geq 0$. The assumption that the elements are in the unit interval results in no loss in generality as all the presented results are valid for any $[a,b]$ with $a < b$.

Our goal is to develop an exact algorithm for finding a weight vector $\mathbf{h}$, $0 \leq h_1 \leq h_2 \leq \dots \leq h_n \leq 1$, that minimizes (globally) the mean absolute error (MAE, L_1):

$$E_1(\mathbf{h}) = \frac{1}{m} \sum_{k=1}^{m} \left| \mathsf{S}_{\mathbf{h}} \left(\mathbf{x}^{(k)} \right) - y^{(k)} \right|. \tag{2}$$

The reason for choosing such a metric lies in the fact that the Sugeno integral is often proposed for the case of ordinal inputs, and here the MAE approach is tantamount to minimizing the so-called *natural metric* defined over a chain. Also, we shall see (in Theorem 1) that contrary to other popular metrics, MAE allows a solution that is on the same scale as the training sample. What is more, MAE is known to be more robust in the presence of outliers than, e.g., the root mean squared error. Finally, note we assumed that h_n need not be equal to one – $\mathbf{h}$ being normalized shall easily follow as a particular case of our setting.

Let us emphasize here that already in [9] we observed that despite the objective function is Lipschitz continuous (more precisely, we can show that for any $\mathbf{h} \in [0,1]^n$, $\boldsymbol{\delta} \in \mathbb{R}^n$ such that $\mathbf{h} + \boldsymbol{\delta} \in [0,1]^n$ it holds $|E_1(\mathbf{h}' + \boldsymbol{\delta}) - E_1(\mathbf{h}')| \leq \bigvee_{j=1}^{n} |\delta_j|$), it is not necessarily convex. It can have multiple local minima (which are not global minima, compare Figure 1 in [9]), so generic mathematical programming solvers do not guarantee that the solution they find is globally optimal.

The paper is set out as follows. In the next section, we present some theoretical results concerning the location of the solution, error bounds, etc., that shall form the basis of the exact algorithm to fit a Sugeno integral to data presented in Section III. Section IV provides and analyses performance benchmarks of the proposed method in various data scenarios. Section V applies it in a bibliometrics exercise. Section VI finalizes the paper.

II. Auxiliary Results

A. Location of a Solution

The composition of the Sugeno integral is such that a single pair (x_i, h_i) generates the final output. Note that for any fixed nonincreasing $\mathbf{x}$ and nondecreasing $\mathbf{h}$, we have:

$$\begin{gathered} i = \min\{\lambda : x_\lambda \wedge h_\lambda = \textstyle\bigvee_{j=1}^{n} x_j \wedge h_j\} \text{ if and only if} \\ (x_i \wedge h_i > h_{i-1}) \text{ and } (h_i \geq x_{i+1} \text{ or } x_i \geq h_i = h_n), \\ i = \max\{\lambda : x_\lambda \wedge h_\lambda = \textstyle\bigvee_{j=1}^{n} x_j \wedge h_j\} \text{ if and only if} \\ (x_i \wedge h_i > x_{i+1}) \text{ and } (x_i \geq h_{i-1} \text{ or } h_i \geq x_i = x_1). \end{gathered}$$

The following result states that there exists $\mathbf{h}$ which is a minimizer of E_1 such that every h_i is one of the inputs or desired outputs. Note that it is a particularly useful property in the case when data are in fact on some discrete bounded chain mapped to the unit (or any other) interval.

Theorem 1. *Fix any $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)} \in [0,1]^n$ and $y^{(1)}, \dots, y^{(m)} \in [0,1]$ and let $\mathcal{I} = \{x_1^{(1)}, \dots, x_1^{(m)}, x_2^{(1)}, \dots, x_2^{(m)}, \dots, x_n^{(1)}, \dots, x_n^{(m)}, y^{(1)}, \dots, y^{(m)}\}$. Then there exists $\mathbf{h}^*$ with $h_1^*, \dots, h_n^* \in \mathcal{I}$ such that it is among the minimizers of E_1 given by* (2).

Proof. We need to show that there exist $h_1^*, \dots, h_n^* \in \mathcal{I}$ with:

$$\sum_{k=1}^{m} \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j^* - y^{(k)} \right| \leq \sum_{k=1}^{m} \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j - y^{(k)} \right|$$

for all $h_1, \dots, h_n \in [0,1]$. Assume otherwise, i.e., take $0 \leq h_1^\# \leq \dots \leq h_n^\# \leq 1$ with $h_j^\# \notin \mathcal{I}$ for some j such that:

$$\sum_{k=1}^{m} \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j^\# - y^{(k)} \right| - \sum_{k=1}^{m} \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j - y^{(k)} \right| < 0$$

for each nondecreasing $\mathbf{h} \in \mathcal{I}^n$.

First of all, we can safely assume that $h_1^\# \geq \min \mathcal{I}$ and $h_n^\# \leq \max \mathcal{I}$, because:

$$\begin{aligned} &\textstyle\sum_{k=1}^{m} \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j^\# - y^{(k)} \right| \\ &= \textstyle\sum_{k=1}^{m} \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge \left(h_j^\# \wedge \max \mathcal{I} \right) - y^{(k)} \right| \\ &\geq \textstyle\sum_{k=1}^{m} \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge \left(\left(\min \mathcal{I} \vee h_j^\# \right) \wedge \max \mathcal{I} \right) - y^{(k)} \right|. \end{aligned}$$

Thus, let us take the smallest u such that $h_u^\# \notin \mathcal{I}$. Set $\alpha = \max\{\alpha \in \mathcal{I} : \alpha < h_u^\#\}$ and $\beta = \min\{\beta \in \mathcal{I} : \beta > h_u^\#\}$. Also, let $v = \max\{v \leq n : h_v^\# < \beta\}$. We shall show that there exists a nondecreasing $\mathbf{h}$ with $h_i = h_i^\#$ for $i \notin [u, v]$ and $h_i \in \{\alpha, \beta\}$ otherwise, which yields $E_1(\mathbf{h}) \leq E_1(\mathbf{h}^\#)$. This is sufficient for contradicting our assumption as we can modify the elements in $\mathbf{h}$ using the very same update scheme as many times as needed to get a nondecreasingly ordered $\mathbf{h}$ with elements only in $\mathcal{I}$. Next, we have $E_1(\mathbf{h}^\#) - E_1(\mathbf{h}) =$

$$\begin{aligned} &= \sum_{k : x_u^{(k)} \leq \alpha} \left(\left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j^\# - y^{(k)} \right| - \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j - y^{(k)} \right| \right) \\ &+ \sum_{\substack{k : x_u^{(k)} \geq \beta \\ \text{and } x_{u+1}^{(k)} \leq \alpha}} \left(\left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j^\# - y^{(k)} \right| - \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j - y^{(k)} \right| \right) \\ &+ \sum_{\substack{k : x_{u+1}^{(k)} \geq \beta \\ \text{and } x_{u+2}^{(k)} \leq \alpha}} \left(\left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j^\# - y^{(k)} \right| - \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j - y^{(k)} \right| \right) \\ &+ \dots \\ &+ \sum_{\substack{k : x_v^{(k)} \geq \beta \\ \text{and } x_{v+1}^{(k)} \leq \alpha}} \left(\left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j^\# - y^{(k)} \right| - \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j - y^{(k)} \right| \right) \\ &+ \sum_{k : x_{v+1}^{(k)} \geq \beta} \left(\left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j^\# - y^{(k)} \right| - \left| \bigvee_{j=1}^{n} x_j^{(k)} \wedge h_j - y^{(k)} \right| \right) \end{aligned}$$

$$= \sum_{i=u}^{v} \sum_{k: x_i^{(k)} \ge \beta \text{ and } x_{i+1}^{(k)} \le \alpha} \left(\left| h_i^{\#} - y^{(k)} \right| - \left| h_i - y^{(k)} \right| \right),$$

denoted (*). Now let us set for $i = u, u+1, \ldots, v$:

$$c_i = \left| \{ k : y^{(k)} \le \alpha \text{ and } x_i^{(k)} \ge \beta \text{ and } x_{i+1}^{(k)} \le \alpha \} \right|,$$
$$d_i = \left| \{ k : y^{(k)} \ge \beta \text{ and } x_i^{(k)} \ge \beta \text{ and } x_{i+1}^{(k)} \le \alpha \} \right|.$$

Continuing the above, we reduce (*) to:

$$\sum_{i=u}^{v} \sum_{\substack{k: y^{(k)} \le \alpha \text{ and} \\ x_i^{(k)} \ge \beta \text{ and } x_{i+1}^{(k)} \le \alpha}} (h_i^{\#} - y^{(k)} - h_i + y^{(k)})$$
$$+ \sum_{i=u}^{v} \sum_{\substack{k: y^{(k)} \ge \beta \text{ and} \\ x_i^{(k)} \ge \beta \text{ and } x_{i+1}^{(k)} \le \alpha}} (y^{(k)} - h_i^{\#} - y^{(k)} + h_i)$$
$$= \sum_{i=u}^{v} (c_i - d_i)(h_i^{\#} - h_i).$$

We need to show that the assumption $(*) < 0$ – which is now equivalent to $\sum_{j=u}^{i} (c_j - d_j)(h_j^{\#} - \alpha) + \sum_{j=i+1}^{v} (d_j - c_j)(\beta - h_j^{\#}) < 0$ for all $i = u-1, \ldots, v$ – leads to a contradiction. Observe that we may safely assume that $\sum_{j=u}^{i} (c_j - d_j)(h_j^{\#} - \alpha) < 0$ holds for all $i = u, \ldots, v$, as otherwise we could set $h_u, \ldots, h_i := \alpha$ and either get a better error rate or an identical one. In the latter case, we can start our reasoning over from $u \mapsto i+1$. By the optimality of $\mathbf{h}^{\#}$, we also have $\sum_{j=u}^{i} (c_j - d_j)(h_j^{\#} - h_j^{\#}) \le 0$ for all i. Hence, $\sum_{j=u}^{i} c_j \le \sum_{j=u}^{i} d_j$ but then setting $h_i := \beta$ yields $E_1(\mathbf{h}^{\#}) \ge E_1(\mathbf{h})$, contrary to our assumption – which completes our proof. □

B. Search Space Boundary Refinement

Next, we observe that if we have an initial guess regarding the location (lower and upper bounds) of a global minimizer of E_1, we may try refining the search space boundaries by solving a series of simple unidimensional optimization tasks, see Figure 1. Note that it is always possible to set $\boldsymbol{\zeta}^- = (0, 0, \ldots, 0, 0 \text{ or } 1)$ and $\boldsymbol{\zeta}^+ = (1, 1, \ldots, 1, 1)$ below.

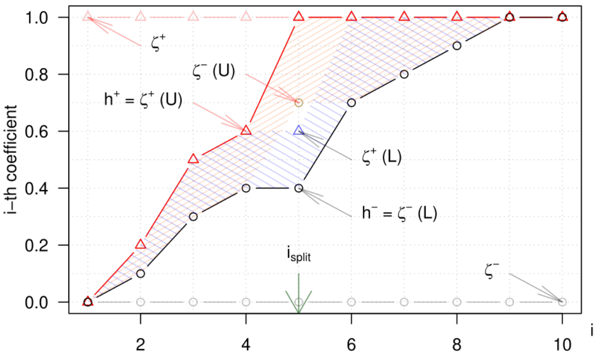

Figure 1. Example bounds $\mathbf{h}^-, \mathbf{h}^+$ – refined versions of initial guesses $\boldsymbol{\zeta}^- = (0, 0, \ldots, 0)$ and $\boldsymbol{\zeta}^+ = (1, 1, \ldots, 1)$; see Theorem 2. The region $\{\mathbf{h} \text{ nondecreasing} : \mathbf{h}^- \le \mathbf{h} \le \mathbf{h}^+\}$ is then split into two subregions, denoted L and U, by the proposed algorithm.

Theorem 2. *Fix any $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)} \in [0,1]^n$ and $y^{(1)}, \ldots, y^{(m)} \in [0,1]$ and take any nondecreasing $\boldsymbol{\zeta}^- \le \boldsymbol{\zeta}^+$. Then the nondecreasing $\mathbf{h}^-, \mathbf{h}^+$ iteratively defined as follows so that $\boldsymbol{\zeta}^- \le \mathbf{h}^- \le \mathbf{h}^+ \le \boldsymbol{\zeta}^+$, enjoy the property that $\min_{\mathbf{h}: \boldsymbol{\zeta}^- \le \mathbf{h} \le \boldsymbol{\zeta}^+} E_1(\mathbf{h}) = \min_{\mathbf{h}: \mathbf{h}^- \le \mathbf{h} \le \mathbf{h}^+} E_1(\mathbf{h})$.*

$$h_1^+ = \mathrm{L}\arg\min_{\widehat{h} \in [\zeta_1^-, \zeta_1^+]} E_1(\widehat{h}, \zeta_2^+, \zeta_3^+, \ldots, \zeta_n^+),$$
$$h_2^+ = \mathrm{L}\arg\min_{\widehat{h} \in [\zeta_2^- \vee h_1^+, \zeta_2^+]} E_1(h_1^+, \widehat{h}, \zeta_3^+, \ldots, \zeta_n^+),$$
$$\vdots$$
$$h_n^+ = \mathrm{L}\arg\min_{\widehat{h} \in [\zeta_n^- \vee h_{n-1}^+, \zeta_n^+]} E_1(h_1^+, \ldots, h_{n-1}^+, \widehat{h}),$$
$$h_n^- = \mathrm{U}\arg\min_{\widehat{h} \in [\zeta_n^-, h_n^+]} E_1(\zeta_1^-, \ldots, \zeta_{n-2}^-, \zeta_{n-1}^-, \widehat{h}),$$
$$h_{n-1}^- = \mathrm{U}\arg\min_{\widehat{h} \in [\zeta_{n-1}^-, h_n^- \wedge h_{n-1}^+]} E_1(\zeta_1^-, \ldots, \zeta_{n-2}^-, \widehat{h}, h_n^-),$$
$$\vdots$$
$$h_1^- = \mathrm{U}\arg\min_{\widehat{h} \in [\zeta_1^-, h_2^- \wedge h_1^+]} E_1(\widehat{h}, h_2^-, \ldots, h_{n-1}^-, h_n^-),$$

where $\mathrm{L}\arg\min_x f(x) = \min\{x : x = \min_x f(x)\}$ and $\mathrm{U}\arg\min_x f(x) = \max\{x : x = \min_x f(x)\}$ denote the lower and upper $\arg\min$, respectively.

Moreover, $\mathbf{h}^-, \mathbf{h}^+ \in \mathcal{I}^n$.

Proof. Let $e := \min_{\mathbf{h}: \boldsymbol{\zeta}^- \le \mathbf{h} \le \boldsymbol{\zeta}^+} E_1(\mathbf{h})$. Firstly, consider any $\mathbf{h}^*$ with $\boldsymbol{\zeta}^+ \ge \mathbf{h}^* \not\le \mathbf{h}^+$ such that $E_1(\mathbf{h}^*) = e$. Take the smallest u such that $h_u^* > h_u^+$. Note that $h_{u-1}^* \le h_{u-1}^+ \le h_u^+ < h_u^* \le h_{u+1}^* \le \zeta_{u+1}^+$. Then:

$$E_1(\mathbf{h}^*) - E_1(h_1^*, \ldots, h_{u-1}^*, h_u^+, h_{u+1}^*, \ldots, h_n^*)$$
$$= \sum_{\substack{k: x_{u+1}^{(k)} \le h_u^+ \\ \text{and } x_u^{(k)} \ge h_{u-1}^+}} \left(\left| x_u^{(k)} \wedge h_u^* - y^{(k)} \right| - \left| x_u^{(k)} \wedge h_u^+ - y^{(k)} \right| \right)$$
$$+ \sum_{\substack{k: h_u^+ < x_{u+1}^{(k)} \le h_u^* \text{ and} \\ x_u^{(k)} \ge h_{u-1}^+}} \left(\left| x_u^{(k)} \wedge h_u^* - y^{(k)} \right| \right.$$
$$\left. - \left| \cancelto{x_{u+1}^{(k)} = x_{u+1}^{(k)} \wedge \zeta_{u+1}^+}{x_{u+1}^{(k)} \wedge h_{u+1}^*} - y^{(k)} \right| \right)$$
$$+ \sum_{\substack{k: \quad x_{u+1}^{(k)} \le h_u^+ \text{ and} \\ h_{u-1}^+ > x_u^{(k)} \ge h_{u-1}^*}} \left(\left| \cancelto{x_u^{(k)}}{x_u^{(k)} \wedge h_u^*} - y^{(k)} \right| \right.$$
$$\left. - \left| \cancelto{x_u^{(k)}}{x_u^{(k)} \wedge h_u^+} - y^{(k)} \right| \right)$$
$$+ \sum_{\cancelto{k \in \emptyset}{\substack{k: h_u^+ < x_{u+1}^{(k)} \le h_u^* \text{ and} \\ h_{u-1}^+ > x_u^{(k)} \ge h_{u-1}^*}}} \left(\left| \cancelto{x_u^{(k)}}{x_u^{(k)} \wedge h_u^*} - y^{(k)} \right| \right.$$
$$\left. - \left| \cancelto{x_{u+1}^{(k)}}{x_{u+1}^{(k)} \wedge h_{u+1}^*} - y^{(k)} \right| \right)$$

$$\begin{aligned}
&= \sum_{\substack{k: x_{u+1}^{(k)} \le h_u^+ \\ \text{and } x_u^{(k)} \ge h_{u-1}^+}} \left(\left| x_u^{(k)} \wedge h_u^* - y^{(k)} \right| - \left| x_u^{(k)} \wedge h_u^+ - y^{(k)} \right| \right) \\
&+ \sum_{\substack{k: h_u^+ < x_{u+1}^{(k)} \le h_u^* \\ \text{and } x_u^{(k)} \ge h_{u-1}^+}} \left(\left| x_u^{(k)} \wedge h_u^* - y^{(k)} \right| - \left| x_{u+1}^{(k)} \wedge \zeta_{u+1}^+ - y^{(k)} \right| \right) \\
&= \sum_{\substack{k: x_{u+1}^{(k)} \le h_u^* \text{ and} \\ x_u^{(k)} \ge h_{u-1}^+}} \left| x_u^{(k)} \wedge h_u^* - y^{(k)} \right| + \sum_{k: x_{u+1}^{(k)} \wedge \zeta_{u+1}^+ > h_u^*} \left| x_{u+1}^{(k)} - y^{(k)} \right| \\
&- \sum_{\substack{k: x_{u+1}^{(k)} \le h_u^+ \text{ and} \\ x_u^{(k)} \ge h_{u-1}^+}} \left| x_u^{(k)} \wedge h_u^+ - y^{(k)} \right| - \sum_{k: x_{u+1}^{(k)} \wedge \zeta_{u+1}^+ > h_u^+} \left| x_{u+1}^{(k)} - y^{(k)} \right| \\
&= \quad E_1(h_1^+, \dots, h_{u-1}^+, h_u^*, \zeta_{u+1}^+, \dots, \zeta_n^+) \\
&- \quad E_1(h_1^+, \dots, h_{u-1}^+, h_u^+, \zeta_{u+1}^+, \dots, \zeta_n^+) \qquad \ge 0.
\end{aligned}$$

We see that we can always set h_u^* in $\mathbf{h}^*$ to be equal to h_u^+ and that this results in the error measure not being increased. After transforming the remaining elements in $\mathbf{h}^*$ in a similar manner (taking the next smallest u with $h_u^* > h_u^+$), we get that there exists $\mathbf{h}^* \le \mathbf{h}^+$ such that $E_1(\mathbf{h}^*) = e$.

Secondly, now let $\mathbf{h}^* \le \mathbf{h}^+$ with $\boldsymbol{\zeta}^- \le \mathbf{h}^* \not\ge \mathbf{h}^-$ such that $E_1(\mathbf{h}^*) = e$. Take the greatest v such that $h_v^* < h_v^-$. Note that $\zeta_{v-1}^- \le h_{v-1}^* \le h_v^* < h_v^- \le h_{v+1}^- \le h_{v+1}^*$. We have:

$$\begin{aligned}
& E_1(\mathbf{h}^*) - E_1(h_1^*, \dots, h_{v-1}^*, h_v^-, h_{v+1}^*, \dots, h_n^*) \\
&= \sum_{\substack{k: x_{v+1}^{(k)} \le h_v^* \\ \text{and } x_v^{(k)} \ge h_{v-1}^*}} \left(\left| x_v^{(k)} \wedge h_v^* - y^{(k)} \right| - \left| x_v^{(k)} \wedge h_v^- - y^{(k)} \right| \right) \\
&+ \sum_{\substack{k: h_v^* < x_{v+1}^{(k)} \le h_v^- \\ \text{and } x_v^{(k)} \ge h_{v-1}^*}} \Big(\big| \cancelto{x_{v+1}^{(k)} \wedge h_{v+1}^-}{x_{v+1}^{(k)} \wedge h_{v+1}^*} - y^{(k)} \big| \\
&\qquad - \big| x_v^{(k)} \wedge h_v^- - y^{(k)} \big| \Big) \\
&= \sum_{k: x_{v+1}^{(k)} \le h_v^*} \left| x_v^{(k)} \wedge h_v^* - y^{(k)} \right| \\
&+ \sum_{k: x_{v+1}^{(k)} > h_v^*} \left| \bigvee_{j=v+1}^n x_j^{(k)} \wedge h_j^- - y^{(k)} \right| \\
&- \sum_{k: x_{v+1}^{(k)} \le h_v^-} \left| x_v^{(k)} \wedge h_v^- - y^{(k)} \right| \\
&- \sum_{k: x_{v+1}^{(k)} > h_v^+} \left| \bigvee_{j=v+1}^n x_j^{(k)} \wedge h_j^- - y^{(k)} \right| \\
&= \quad E_1(\zeta_1^-, \dots, \zeta_{v-1}^-, h_v^*, h_{v+1}^-, \dots, h_n^-) \\
&- \quad E_1(\zeta_1^-, \dots, \zeta_{v-1}^-, h_v^-, h_{v+1}^-, \dots, h_n^-) \qquad \ge 0.
\end{aligned}$$

Similarly as above, we can always set h_v^* in $\mathbf{h}^*$ to be equal to h_v^-. Applying this iteratively, it results that there exists $\mathbf{h}^*$ with $E_1(\mathbf{h}^*) = e$ such that $\mathbf{h}^* \ge \mathbf{h}^-$.

Lastly, we can easily imply $\mathbf{h}^-, \mathbf{h}^+ \in \mathcal{I}^n$ from the fact that $\tilde{E}(\widehat{h}) = E_1(h_1, \dots, h_{i-1}, \widehat{h}, h_{i+1}, \dots, h_n)$ for any possible fixed $h_1, \dots, h_{i-1}, h_{i+1}, \dots, h_n$ is a continuous function and that it is differentiable at each $\widehat{h} \notin \mathcal{I}$, where its derivative is equal to some constant (possibly 0), QED. □

III. The BRNB Algorithm

We are now in position to introduce a method to determine a capacity such that the corresponding Sugeno integral minimizes the mean absolute error between the transformed inputs and the outputs, i.e.:

$$\text{minimize } \sum_{k=1}^m \frac{1}{m} \left| \mathsf{S}_{\mathbf{h}}\left(\mathbf{x}^{(k)}\right) - y^{(k)} \right| \text{ with respect to } \mathbf{h} \quad (3)$$

such that $0 \le h_1 \le h_2 \le \dots \le h_n \le 1$. Note that a global minimizer is not necessarily unique (compare Figure 1 in [9]). Based on Theorem 1, we know we may restrict the search space to coefficients vectors $\mathbf{h}$ with $h_i \in \mathcal{I}$ for all i, which is particularly appealing from the computational side: there are countably many candidate points to consider. Moreover, if the inputs are on a finite chain (ordinal data), then $\mathbf{h} \in \mathcal{I}^n$ implies $\mathsf{S}_{\mathbf{h}}(\mathbf{x}) \in \mathcal{I}$ for every $\mathbf{x} \in \mathcal{I}^n$.

In [9] we considered a quite naïve approach – a straightforward version of the time and space costly branch-and-bound-type algorithm. Based on the theoretical results derived above, we can now formulate an exact branch-*refine*-and-bound-type algorithm that explores the possible solutions $\mathbf{h}$ consisting of elements in $\mathcal{I}$ much more intelligently – see Figure 2 for the pseudocode. Candidate regions bounded by nondecreasing $\boldsymbol{\zeta}^-$ and $\boldsymbol{\zeta}^+$ are split, refined (Theorem 2), and possibly rejected from further inspection if they fail to guarantee the inclusion of a potential solution. For this we need some lower bound e such that, given $\boldsymbol{\zeta}^-, \boldsymbol{\zeta}^+$, it holds $E_1(\mathbf{h}) \ge e$ for all nondecreasing $\mathbf{h}$ with $\boldsymbol{\zeta}^- \le \mathbf{h} \le \boldsymbol{\zeta}^+$. As the Sugeno integral $\mathsf{S}_{\mathbf{h}}$ is a nondecreasing function of $\mathbf{h}$ (for each fixed $\mathbf{x}$), we may consider:

$$e(\boldsymbol{\zeta}^-, \boldsymbol{\zeta}^+) = \frac{1}{m} \sum_{k=1}^n \begin{cases} y^{(k)} - \mathsf{S}_{\boldsymbol{\zeta}^+}(\mathbf{x}^{(k)}) & \text{if } y^{(k)} > \mathsf{S}_{\boldsymbol{\zeta}^+}(\mathbf{x}^{(k)}), \\ \mathsf{S}_{\boldsymbol{\zeta}^-}(\mathbf{x}^{(k)}) - y^{(k)} & \text{if } y^{(k)} < \mathsf{S}_{\boldsymbol{\zeta}^-}(\mathbf{x}^{(k)}), \\ 0 & \text{otherwise.} \end{cases}$$

The use of a FIFO queue gives a breadth-first order of the subregions' inspection (a LIFO stack turned out to give slower run-times). Note that if the refinement and bound steps were not applied, a globally optimal solution would still be found in a finite number of steps, but this would require many more iterations of the algorithm. Hence, in the next section we shall verify whether these two components of the proposed routine are beneficial in practice.

IV. Performance Benchmarks

In order to verify the performance of the proposed method, for a given n and m, we shall consider $M = 1{,}000$ random replications of the following experiment. In each scenario, we generate u and v from the uniform distribution $U[0.25, 5]$ and then sample the elements in $\mathbf{h}$ independently from the beta distribution $B(u, v)$, fixing $h_n = 1$. The resulting vector is then sorted nondecreasingly. Note that a beta distribution captures a wide range of possible shapes of the coefficient vectors. Next, each input vector is determined by independently

Algorithm BRNB. Inputs: $\mathbf{X} = [\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}] \in [0,1]^{n\times m}$ and $\mathbf{Y} = [y^{(1)}, \dots, y^{(m)}] \in [0,1]^{1\times m}$;

```
1. Compute $\mathcal{I}$, see Theorem 1;
2. Let $\boldsymbol{\zeta}^- = (0,0,\dots,0)$ and $\boldsymbol{\zeta}^+ = (1,1,\dots,1)$;
3. Set $\mathrm{cur}_{\mathbf{h}} = \emptyset$, $\mathrm{cur}_e = \infty$;
4. $q$ = new FIFO Queue;
5. $q$.push(($\boldsymbol{\zeta}^-, \boldsymbol{\zeta}^+$));
6. While $q$ is not empty:
   6.1. ($\boldsymbol{\zeta}^-, \boldsymbol{\zeta}^+$) = $q$.pop();
   6.2. Refine $\boldsymbol{\zeta}^-, \boldsymbol{\zeta}^+$ by applying Theorem 2 to determine $\mathbf{h}^-$ and $\mathbf{h}^+$ with $\boldsymbol{\zeta}^- \le \mathbf{h}^- \le \mathbf{h}^+ \le \boldsymbol{\zeta}^+$;
   6.3. If $e(\mathbf{h}^-, \mathbf{h}^+) \ge \mathrm{cur}_e$:        (lower bound is not promising)
     6.3.1. return;        (reject the current subregion)
   6.4. If $E_1(\mathbf{h}^-) < \mathrm{cur}_e$:        ($\mathbf{h}^-$ is a new solution candidate)
     6.4.1. $\mathrm{cur}_e = E_1(\mathbf{h}^-)$;
     6.4.2. $\mathrm{cur}_h = \mathbf{h}^-$;
   6.5. If $\mathbf{h}^- = \mathbf{h}^+$:        (a singleton)
     6.5.1. return;
   6.6. If $E_1(\mathbf{h}^+) < \mathrm{cur}_e$:        ($\mathbf{h}^+$ is a new solution candidate)
     6.6.1. $\mathrm{cur}_e = E_1(\mathbf{h}^+)$;
     6.6.2. $\mathrm{cur}_h = \mathbf{h}^+$;
   6.7. Pick any $i$ such that $h_i^- < h_i^+$;        (e.g., the middle one)
   6.8. Determine $\mathcal{I}' = \{\iota'_1, \dots, \iota'_{|\mathcal{I}'|}\} = \{\iota \in \mathcal{I} \cap [h_i^-, h_i^+]\}$ and assume $\iota'_1 < \dots < \iota'_{|\mathcal{I}'|}$;
   6.9. Pick any $j$ such that $1 \le j < |\mathcal{I}'|$;
   6.a. $\boldsymbol{\zeta}^{+(L)} = (h_1^+ \wedge \iota'_j, \dots, h_i^+ \wedge \iota'_j, h_{i+1}^+, \dots, h_n^+)$;
   6.b. $\boldsymbol{\zeta}^{-(U)} = (h_1^-, \dots, h_{i-1}^-, h_i^- \vee \iota'_{j+1}, \dots, h_n^- \vee \iota'_{j+1})$;
   6.c. $q$.push(($\mathbf{h}^-, \boldsymbol{\zeta}^{+(L)}$));        (assert: $\boldsymbol{\zeta}^{+(L)} \not\geq \mathbf{h}^+$)
   6.d. $q$.push(($\boldsymbol{\zeta}^{-(U)}, \mathbf{h}^+$));        (assert: $\boldsymbol{\zeta}^{-(U)} \not\leq \mathbf{h}^-$)
7. Return $\mathrm{cur}_{\mathbf{h}}$ as result;
```

Figure 2. Pseudocode of the proposed branch-refine-and-bound algorithm. The reference Cython (for use in the Python 3 environment) implementation of the presented algorithm is available at https://github.com/gagolews/SugenoIntegralFitting.

generating and then ordering n random variates $\sim B(u,v)$ with u and v sampled from $U[0.25, 5]$. In each scenario, all m vectors either a) share the same (u,v) or (with probability 50%) b) have their (u,v) sampled individually (m times). What is more, the m reference outputs are generated in one of the 3 following manners (each chosen with probability $1/3$). For each $\mathbf{x}^{(k)}$, the corresponding $y^{(k)}$ is set to: a) $0 \vee \left(1 \wedge \left(\mathsf{S}_{\mathbf{h}}\left(\mathbf{x}^{(k)}\right) + \varepsilon^{(k)}\right)\right)$, where $\varepsilon^{(k)}$ is sampled from the normal distribution with expected value 0 and standard deviation 0.1, b) $0 \vee \left(1 \wedge \left(\mathsf{S}_{\mathbf{h}}\left(\mathbf{x}^{(k)}\right) + \varepsilon^{(k)}\right)\right)$, where $\varepsilon^{(k)}$ is sampled from the Cauchy distribution with location 0 and scale 0.1, c) $\mathsf{S}_{\mathbf{h}}\left(\mathbf{x}^{(k)}\right)$ with probability 0.5 and a random variate $\sim U[0,1]$ otherwise (i.e., half of the outputs on average were not contaminated while the remaining were pure random noise). Lastly, there is an optional post-processing step. In the *continuous* case, data are kept as-is. In the *ordinal* case, all elements in $\mathbf{h}, \mathbf{X}, \mathbf{Y}$ are rounded to two decimal digits, i.e., the cardinality of $\mathcal{I}$ is at most 101.

Table I
The number of iterations of different versions of the BRNB algorithm for $n = 5$ and $m = 10$; the refinement step (based on Theorem 2) significantly decreases its run-time

	data	Q.25	Q.50	Q.75	Q.95	Max
full	ordinal	1	3	7	17	135
no bound	ordinal	1	3	11	29	441
no refine	ordinal	561	1,285	3,397	10,207	48,185
full	cont.	1	3	9	23	169
no bound	cont.	1	5	15	49	575
no refine	cont.	1,079	2,678	8,278	26,270	139,983

Table II
The number of iterations of BRNB for different n and m

n	m	data	Q.25	Q.50	Q.75	Q.95	Max
5	10	ordinal	1	3	7	17	135
5	100	ordinal	1	1	7	17	235
5	1,000	ordinal	1	1	3	11	105
5	10,000	ordinal	1	1	1	9	29
10	10	ordinal	3	11	25	119	16,741
10	100	ordinal	5	15	53	429	22,439
10	1,000	ordinal	1	7	15	71	2,257
10	10,000	ordinal	1	3	9	23	199
25	10	ordinal	13	35	35	1,895	>100k
50	10	ordinal	23	62	384	11,436	>100k
100	10	ordinal	37	85	930	30,969	>100k
5	10	cont.	1	3	9	23	169
5	100	cont.	1	1	17	59	4,317
5	1,000	cont.	1	1	9	143	28,963
5	10,000	cont.	1	1	3	271	94,017
10	10	cont.	5	17	53	343	228,079
25	10	cont.	27	157	1,140	52,710	>100k
50	10	cont.	95	1,028	28,670	>100k	>100k
100	10	cont.	424	7,241	>100k	>100k	>100k

First let us inspect the effects of including the refinement (based on Theorem 2) and bound (based on $e(\boldsymbol{\zeta}^-, \boldsymbol{\zeta}^+)$) steps on the number of iterations of the BRNB algorithm (i.e., the loop in Step 6). As all the empirical distributions of the iteration counts are highly right-skewed, we shall be reporting basic sample quantiles only. In Table I we present the case of $n = 5$ and $m = 10$, which is representative to all the settings we inspected. We observe that the bound step's (6.3) impact is somehow limited by the quality of the lower error estimate e we have chosen. However, it is the refinement step (6.2) that drastically improves the speed of the algorithm – without it, BRNB would reduce to a classical branch-and-bound type algorithm.

Table II gives the number of iterations needed to find a global minimum of the error function for different n and m. Obviously, the number of iterations increases as n increases. However, in the ordinal case (recall that here $|\mathcal{I}| \le 101$) we also observe the tendency that for each fixed n we tested, the larger the m, the smaller the number of iterations. This does not happen in the continuous case, where the algorithm does not scale well due to the fact that $|\mathcal{I}| = O(nm)$.

V. Application in Bibliometrics

Many modifications of the bibliometric Hirsch's h-index have been proposed in the literature. As noted, e.g., in [10], [11], [16], most of them can be expressed as some Sugeno

Table III
LEARNING $\mathbf{h}$ WITH NOISY REFERENCE OUTPUTS; DBLP DATASET, $n_0 = 7$

scenario	i	1	2	3	4	5	6	7
$h_i = i$	mean h_i	1.0	2.0	3.0	4.0	5.0	6.0	7.0
$\sigma = 1$	st.dev. h_i	0.0	0.0	0.0	0.0	0.0	0.0	0.0
$h_i = i$	mean h_i	1.0	2.0	3.0	4.0	5.0	6.0	7.0
$\sigma = 5$	st.dev. h_i	0.0	0.0	0.0	0.0	0.0	0.2	0.6
$h_i = i^2$	mean h_i	1.0	4.0	9.0	16.0	25.0	36.0	40.4
$\sigma = 1$	st.dev. h_i	0.0	0.0	0.0	0.0	0.0	0.3	0.7
$h_i = i^2$	mean h_i	1.0	4.0	9.0	16.0	25.0	35.8	39.0
$\sigma = 5$	st.dev. h_i	0.0	0.0	0.0	0.3	0.7	1.4	1.6

integrals (acting on elements in the set of nonnegative integers) with coefficients vectors $\mathbf{h}$ like $h_i = i$ (the classical h-index), $h_i = i^\beta$ (for some $\beta > 0$), $h_i = \alpha i$ (for some $\alpha > 0$), etc. In a typical regression scenario (with an additive error term), we assume that the observed $(\mathbf{x}^{(i)}, y^{(i)})$ are linked by equation $y^{(i)} = \mathsf{S}_\mathbf{h}(\mathbf{x}^{(i)}) + \varepsilon^{(i)}$ where all $\varepsilon^{(i)}$ are independent and identically distributed with expectation 0 and $\mathbf{h}$ is fixed but unknown.

Let us consider a custom subset of the DBLP computer science bibliography database (DBLP-Citation-network V10 [17], see https://aminer.org/citation, dated October 27, 2017) that consists of 3,079,007 papers and 25,166,994 citation relationships. In each scenario we selected the citation sequences of scholars with paper count within some range n_0. Table III gives typical reconstruction errors (aggregates over 100 different replications) of the true $\mathbf{h}$ assuming that $\varepsilon^{(i)}$ is normally distributed $N(0, \sigma)$ and then rounded to the nearest integer (all negative $y^{(i)}$ are truncated to 0). In all the cases inspected, we observe that the larger the i, the worse the quality of h_i, which can be naturally explained by the fact that citation data tend to be power-law-like distributed. Note that in most cases, BRNB converged in just a few iterations, for example for $n_0 \in [10, 20]$ ($m = 35{,}481$), $h_i = i$, and $\sigma = 5$, the median number of iterations was equal to 15 while the maximum was 275.

VI. CONCLUSIONS AND FUTURE WORK

The Sugeno integral has achieved considerable attention in the fuzzy sets community due to its flexibility and ability to handle inputs expressed over an ordinal scale. Nonetheless, up to now methods for learning the Sugeno integral parameters have not yet been well-developed, and hence its potential in data analysis, machine learning, and decision making (including bibliometrics) has, thus far, been unfulfilled.

We have proposed an exact algorithm for learning the Sugeno integral parameters that makes it tractable for many real-world datasets. Note that the same routine can be used to find a minimum of the objective function with respect to a capacity constrained between any given nondecreasing $\boldsymbol{\zeta}^-$ and $\boldsymbol{\zeta}^+$ (e.g., with $\zeta_n^- = \zeta_n^+ = 1$ yielding a fuzzy measure).

In the future, we can look to make a number of refinements and extensions including: improving the lower bound of the error estimate, $e(\boldsymbol{\zeta}^-, \boldsymbol{\zeta}^+)$; generalizing the approach to more general (not necessarily symmetric) capacities, for instance k-additive or k-maxitive ones for some k (e.g., [12], [13]; such monotone measures are much easier to handle computationally than the full-fledged ones, and, at the same time, leave some room for modeling the possible interactions between the input data dimensions); considering other optimization objectives such as least squares and maximum residual fitting (or other ones generated by, e.g., penalty functions; however, note that even in the case of the least squares criterion, a result similar to Theorem 1 does not hold, and hence such error measures might be not suitable for ordinal data).

ACKNOWLEDGEMENTS

Marek Gagolewski wishes to acknowledge the support by the Czech Science Foundation through the project No.18-06915S and the SEBE Researcher in Residence Program, Deakin University, 2017. Simon James acknowledges the support of the 2018 Deakin University Academic Study Program and the host institution, Warsaw University of Technology.

REFERENCES

[1] G. Beliakov, “How to build aggregation operators from data,” *International Journal of Intelligent Systems*, vol. 18, pp. 903–923, 2003.

[2] G. Beliakov and S. James, “Using linear programming for weights identification of generalized Bonferroni means in R,” *Lecture Notes in Computer Science*, vol. 7647, pp. 35–44, 2012.

[3] G. Beliakov, S. James, and G. Li, “Learning Choquet integral-based metrics in semi-supervised classification,” *IEEE Transactions on Fuzzy Systems*, vol. 19, pp. 562–574, 2011.

[4] V. Torra, “Learning weights for the quasi-weighted means,” *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 5, pp. 653–666, 2002.

[5] G. Beliakov, H. Bustince, and T. Calvo, *A Practical Guide to Averaging Functions*. Springer, 2016.

[6] B. Yuan and G. J. Klir, “Constructing fuzzy measures: A new method and its application to cluster analysis,” in *Proc. NAFIPS’96*, 1996, pp. 567–571.

[7] D. Anderson, J. Keller, and T. Havens, “Learning fuzzy-valued fuzzy measures for the fuzzy-valued Sugeno fuzzy integral,” *Lecture Notes in Artificial Intelligence*, vol. 6178, pp. 502–511, 2010.

[8] H. Prade, A. Rico, and M. Serrurier, “Elicitation of Sugeno integrals: A version space learning perspective,” *Lecture Notes in Computer Science*, vol. 5722, pp. 392–401, 2009.

[9] M. Gagolewski and S. James, “Fitting symmetric fuzzy measures for discrete Sugeno integration,” in *Advances in Intelligent Systems and Computing*. Springer, 2018, vol. 642, pp. 104–116.

[10] R. Mesiar and M. Gagolewski, “H-index and other Sugeno integrals: Some defects and their compensation,” *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 6, pp. 1668–1672, 2016.

[11] V. Torra and Y. Narukawa, “The h-index and the number of citations: Two fuzzy integrals,” *IEEE Transactions on Fuzzy Systems*, vol. 16, no. 3, pp. 795–797, 2008.

[12] R. Mesiar, “k-order additivity and maxitivity,” *Atti del Seminario Matematico e Fisico dell’Universita di Modena*, vol. 23, no. 51, pp. 179–189, 2003.

[13] M. Grabisch, “k-order additive discrete fuzzy measures and their representation,” *Fuzzy Sets and Systems*, vol. 92, pp. 167–189, 1997.

[14] D. Dubois, H. Prade, and C. Testemale, “Weighted fuzzy pattern matching,” *Fuzzy Sets and Systems*, vol. 28, pp. 313–331, 1988.

[15] M. Gagolewski, “On the relationship between symmetric maxitive, minitive, and modular aggregation operators,” *Information Sciences*, vol. 221, pp. 170–180, 2013.

[16] M. Gagolewski and R. Mesiar, “Monotone measures and universal integrals in a uniform framework for the scientific impact assessment problem,” *Information Sciences*, vol. 263, pp. 166–174, 2014.

[17] J. Tang *et al.*, “ArnetMiner: Extraction and mining of academic social networks,” in *Proceedings of the Fourteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD’2008)*, 2008, pp. 990–998.