Median-Based Aggregation Operators for Prototype Construction in Ordinal Scales

Josep Domingo-Ferrer,[1,*] Vicenç Torra[2,†]
[1]*Department of Computing Engineering and Maths, Universitat Rovira i Virgili, Av Països Catalans 26, 43007 Tarragona, Catalonia, Spain*
[2]*Institut d'Investigació en Intel·ligència Artificial, Campus Universitat Autònoma de Barcelona, 08193 Bellaterra, Catalonia, Spain*

This article studies aggregation operators in ordinal scales for their application to clustering (more specifically, to microaggregation for statistical disclosure risk). In particular, we consider these operators in the process of prototype construction. This study analyzes main aggregation operators for ordinal scales [plurality rule, medians, Sugeno integrals (SI), and ordinal weighted means (OWM), among others] and shows the difficulties for their application in this particular setting. Then, we propose two approaches to solve the drawbacks and we study their properties. Special emphasis is given to the study of monotonicity because the operator is proven nonsatisfactory for this property. Exhaustive empirical work shows that in most practical situations, this cannot be considered a problem. © 2003 Wiley Periodicals, Inc.

1. INTRODUCTION

Information fusion techniques and aggregation operators are commonly applied into several fields of human knowledge. Because different fields imply different requirements, a large number of aggregation operators exist today. In addition, differences in the way knowledge is represented forced the development of tools to deal with the different knowledge representation formalisms. In particular, methods exist to deal with different kinds of data. For example, there are methods to fuse numerical information (i.e., data in numerical scale[1]), categorical information (either ordinal[2] or nominal scales[3]), and information expressed by means of partitions (or, equivalently, equivalent relations[4]), dendrograms, and classification trees, preferences, orderings, images, etc.

This work is devoted to the case of categorical information. The development of operators of any kind for categorical information is always a difficult task because of the limited number of commonly established operators over these

*e-mail: jdomingo@etse.urv.es.
†Author to whom all correspondence should be addressed: e-mail: vtorra@iiia.csic.es.

INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS, VOL. 18, 633–655 (2003)
 Published online in Wiley InterScience (www.interscience.wiley.com). • DOI 10.1002/int.10107

scales. In the particular case of aggregation operators, this is even more noticeable because the corresponding operators over numerical scales are the means. These well-known operators are based on product and addition, two operations that do not apply to ordinal scales.

To overcome these difficulties, researchers have considered three main different approaches for the case of ordinal scales. We detail them considering operators over the scale $L = \{l_0, \cdots, l_R\}$, where $l_0 \leq_L l_1 \leq_L \cdots \leq_L l_R$. This classification is based on Ref. 5.

(1) *Explicit quantitative or fuzzy scales.* It is assumed that a translation function assigns values in a different numerical scale for all values in the original ordinal scale. The operators in the ordinal scale are defined from the operators in the underlying scale. Operators defined for fuzzy sets using the extensional principle belong to this class. In some cases, this explicit scale is not given but inferred from additional knowledge about the ordinal scale (e.g., one-to-many negation functions[6]). This is the case of the aggregation operator in Ref. 5.
(2) *Implicit numerical scale.* Operators assume an implicit numerical scale underlying the ordinal scale where values are defined. Usually, each category l_i is dealt as the corresponding integer i. This is the case of linguistic ordered weighted averaging (OWA)[7] and linguistic weighted OWA (WOWA).[8]
(3) *Operating directly on qualitative scales.* Operators stick to a purely ordinal scale and are based only on operators of this scale. This is the case of the median operator or the Sugeno integral (SI).[9] These operators only use the relation $\leq_L$ and *minimum* and *maximum* that rely on $\leq_L$. Other operators in this class [e.g., the weighted mean defined in Ref. 7] are based on t-norms and t-conorms, two operators that can be defined axiomatically over ordinal scales.

The motivation of our work is the application of aggregation operators to statistical disclosure risk. In particular, we consider the extension of existing microaggregation procedures for numerical scales to ordinal scales (see Ref. 4 for a state-of-the-art description of microaggregation procedures). Microaggregation techniques are applied to avoid disclosure of confidential data. To avoid the reidentification of the individual in a data file, the information of these individuals is distorted. Microaggregation consists of clustering the data in small clusters (>10 individuals) and replacing the original values by the prototype (an aggregated value) of the cluster. See Ref. 12 for a detailed analysis of the performance of microaggregation with respect to other distorting techniques for microdata protection.

Typically, in this setting not much information is available on the underlying semantics of categories in ordinal scales. In this study, we focus on the third class of aggregation operators. This is the only case in which no assumptions are made on the existence of an underlying structure beneath the ordinal scale.

The structure of this work is as follows. In Section 2 we review existing aggregation operators in ordinal scales. This section also reviews different usages of weights in aggregation operators. Then, in Section 3, we comment on the suitability of these operators for prototype building. Section 4 introduces new operators for solving the shortcomings of existing ones and analyzes their properties. In Section 5 we present our conclusions.

2. AGGREGATION OPERATORS IN ORDINAL SCALES

In this section, we review some of the existing aggregation operators in ordinal scales that operate directly on categorical values. We begin with the plurality rule. Then, we follow with the median and the SI. The SI generalizes the median and other aggregation operators in categorical scales. We finish, outlining the OWM.

2.1. Plurality Rule

The plurality rule (or plurality function) corresponds to the selection of the most frequent elements. In fact, the definition does not return a single element but returns the set of elements that appear more often. Assuming that values to be aggregated belong to the set L, the plurality rule can be formulated in the following terms (this definition is based on Ref. 3).

DEFINITION 1. *A mapping $P : L^N \rightarrow \wp(L)$ is a plurality function when $P(a_1, \cdots, a_N)$ is the set of all those y in L so that no z in L appears more often in $(a_1, \cdots, a_N)$ than y.*

This definition shows that the procedure can be applied to elements in ordinal scales as well as to elements in nominal scales. So, L is not required to be ordered.

The plurality rule can be extended to introduce weights to measure the reliability of or confidence in each value a_i. This is formulated making explicit the information sources $X = \{ x_1, \cdots, x_N\}$ (here, we assume that x_i supplies the value a_i) and defining the weights as either a function $\mathbf{w}$ from X into a given domain (e.g., [0, 1]) or as a weighting vector $\mathbf{w} = (w_1, \cdots, w_N)$. Both approaches are equivalent as $w_i = w(x_i)$. In the definition of the weighted plurality function, it is also considered a function f to relate each information source with the value it supplies: $f(x_i) = a_i$.

With all of this information, the weighted plurality rule selects the values that accumulate more weights. This is formalized in the following definition by means of a function acc that when applied to $a \in L$ returns the accumulation of the weights of all the sources x_j that supply the value a.

DEFINITION 2. *Let $\mathbf{w}$ be a weighting vector of dimension N; then, a mapping $WP_{\mathbf{w}}$: $L^N \rightarrow \wp(L)$ is a weighted plurality function when $P_{\mathbf{w}}(a_1, \cdots, a_N)$ is the set of all those y in L so that there are no z in L, $acc(z) > acc(y)$, where $acc(a) = \sum_{f(x_j)=a} w(x_j)$.*

In this definition, the range of the weights is restricted in such a way that addition is allowed. Therefore, real numbers and integer numbers are both appropriate for weighting vectors. Moreover, ordinal scales where addition-like operators are defined also are appropriate. This is the case of ordinal scales with t-conorms (see Ref. 13 for a detailed analysis of t-norms and t-conorms in ordinal scales). We would like to emphasize that there is no need to presume that the domain of the weights are equal to the domain of the data.

2.2. Median

The median procedure involves selecting the element that occupies the central position of a sequence of elements when they are ordered according to their value. This can be described formally for numerical data as follows.

DEFINITION 3. *A mapping $M : \mathbb{R}^N \rightarrow \mathbb{R}$ is a median of dimension N if*

$$M(a_1, \cdots, a_N) = \begin{cases} \dfrac{a_{\sigma(N/2)} + a_{\sigma(N/2+1)}}{2} & \text{when } N \text{ is even} \\ a_{\sigma((N+1)/2)} & \text{when } N \text{ is odd} \end{cases}$$

where $\{\sigma(1), \ldots, \sigma(N)\}$ is a permutation of $\{1, \ldots, N\}$ such that $a_{\sigma(i-1)} \geq a_{\sigma(i)}$ for all $i = \{2, \ldots, N\}$ [i.e., $a_{\sigma(i)}$ is the ith largest element in the collection $a_1, \ldots, a_N$].

When dealing with categorical data (i.e., M is a function $M : L^N \rightarrow L$), one of the following expressions will be used for the case of N being even:

$$a_{\sigma(\lfloor (N+1)/2 \rfloor)} \quad \text{or} \quad a_{\sigma(\lceil (N+1)/2 \rceil)}$$

They correspond, respectively, to $a_{\sigma(N/2)}$ and $a_{\sigma(N/2+1)}$.

This definition also can be extended to include weighting vectors. In this case, the central element is a relative position according to the weights. As in the case of the plurality rule, we formalize this definition considering the set of sources X, the function f that assigns the values to the sources, and the weighting vector $\mathbf{w}$.

DEFINITION 4. *Let $\mathbf{w}$ be a weighting vector of dimension N; then, a mapping $WM_{\mathbf{w}} : L^N \rightarrow L$ is a weighted median of dimension N if*

$$WM_{\mathbf{w}}(a_1, \cdots, a_N) = a \text{ iff } acc(a) > 0.5 \geq acc(b)$$

where acc is a function over the values in $\{a_1, \ldots, a_N\}$ defined as $acc(a) = \sum_{f(x_j) \leq a} w(x_j)$ and where b is the largest element in $\{a_1, \ldots, a_N\}$ that is smaller than a. This is, $\mathbf{b} = \max\{x | x \in \{a_1, \ldots, a_N\}, x < a\}$.

In this case, the most natural weighting vector is one defined by positive real numbers that add to one. This is, $\sum w_i = 1$ and $w_i \in (0, 1]$ (note that the definition requires $w_i \neq 0$). However, other possibilities also are possible. In particular, natural numbers can be considered. The weighted median for weights in $\mathbb{N}$ can be translated easily into the previous one through normalization, i.e., defining a new weighting vector $w'_i = w_i/\sum_j w_j$. Moreover, an ordinal scale O with multivalued logic operators can be used also. In this case, besides a t-conorm for addition, an involutive negation is also required (a function n from O to O). In such case, instead of selecting a value on the basis of the value 0.5, we would use the element $x \in O$ such that its negation is also x [i.e., $x = n(x)$].

2.2.1. *Order Statistics*

There exists a set of aggregation operators that are similar to the median. They are the so-called order statistics (OS). OSs permit the selection of the ith greatest value. To do so, the operator requires a preliminary ordering process as in the case of the median and then an integer value i in the range $[1, N]$ to select the ith element. Alternatively, a definition can be given when instead of an integer value, a real number α in the unit interval is given, i.e., selecting the element that occupies the $\alpha \cdot 100$ percentage of the domain. Because the operator only relies on the ordering, it can be applied to ordinal scales.

When the selection of an element is based on a real number (in the unit interval), weights can be included in the definition. This corresponds to the replacement of 0.5 by α in Definition 4. It is clear that the approach is similar to the case of the weights in the median. As previously established, weights correspond to the importance of the sources and can either be real or natural numbers. In the latter case, normalization is required. Ordinal scales also can be used. In this case, the parameter i should be a value in the same ordinal scale (instead of a real number in the unit interval).

2.3. SI

An alternative aggregation operator that also permits the inclusion of weights for the information sources is the SI[9] (see Ref. 14 for a detailed account of its properties). However, this integral does not consider weighting vectors but instead considers the so-called fuzzy measures. If $X = \{x_1, \cdots, x_N\}$ is the set of information sources, a fuzzy measure is a set function that given a subset A of X returns a measure of its importance.

Fuzzy measures satisfy three axioms: (i) the measure of the empty set is zero (when no source is considered, the importance is zero); (ii) the measure of the whole set is one (when all the sources are considered, the importance is maximal and settled to one); (iii) the larger the set of sources, the larger its importance. The first two conditions correspond to boundary conditions and the third condition corresponds to monotonicity. A formal definition of these conditions is given in the following definition.

DEFINITION 5. *A fuzzy measure μ on a set X is a set function $\mu : \wp(X) \rightarrow [0, 1]$, satisfying the following axioms:*

(*i*) $\mu(\emptyset) = 0$, $\mu(X) = 1$ (*boundary conditions*)
(*ii*) $A \subseteq B$ *implies* $\mu(A) \leq \mu(B)$ (*monotonicity*)

This definition is given in the interval [0, 1], but the same definition applies to any ordinal scale $L = \{l_0, \cdots, l_R\}$. In this latter case, the measure is a function from $\wp(X)$ into L and the boundary conditions are $\mu(\emptyset) = l_0$ and $\mu(X) = l_R$.

The SI[9] is defined as the integral of a function f (the one that establishes the value $f(x_i)$ for the information source x_i) with respect to a fuzzy measure. In a numerical scale, the definition is as follows.

DEFINITION 6. *Let μ be a fuzzy measure on X; then, the SI of a function $f : X \rightarrow [0, 1]$ with respect to μ is defined by*

$$(S)\int f d\mu = \max_{i=1,N} \min(f(x_{s(i)}), \mu(A_{s(i)})) \tag{1}$$

where $f(x_{s(i)})$ *indicates that the indices have been permuted so that* $0 \le f(x_{s(1)}) \le \ldots \le f(x_{s(N)}) \le 1$, $A_{s(i)} = \{x_{s(i)}, \ldots, x_{s(N)}\}$ *and* $f(x_{s(0)}) = 0$.

When the values belong to an ordinal scale, an analogous definition is applied. In this latter case it is important to emphasize that both the function f and the fuzzy measure μ are defined as mappings into the same ordinal scale L otherwise the minimum and the maximum operators are not meaningful.

The SI is a very general operator because it generalizes several other aggregation operators. In particular, it generalizes the weighted minimum and the weighted maximum (see Ref. 15 for a detailed description of these operators and of their properties). They are aggregation operators to be used to model logical conjunction and disjunction when the sources are weighted. We review the weighted maximum. The weighted minimum has a similar definition. Both operators use weighting vectors for expressing importance or reliability. Here, the weights map each source into a value in an ordinal scale. Note that as indicated previously, the scale for the values to be aggregated should be the same as the scale for the weights because the minimum combines the values of the weighting vector and the value a_i.

DEFINITION 7. *A vector* $\mathbf{v} = (v_1 \ldots v_N)$ *is a possibilistic weighting vector of dimension N if and only if* $v_i \in L$ *and* $\max_i v_i = l_R$.

DEFINITION 8. *Let* $\mathbf{u}$ *be a weighting vector of dimension N; then, a mapping* $WMax : L^N \to L$ *is a weighted maximum of dimension N if* $WMax_{\mathbf{u}}(a_1, \ldots, a_N) = \max_i \min(u_i, a_i)$.

2.4. Ordinal Weighted Mean

In this section, we give an overview of an ordinal weighted mean (OWM) without going into detail. See Ref. 10 for detailed definitions and properties and Ref. 16 for an extension of the approach to Choquet integrals (CIs).

The ordinal weighted mean (OWM) is a different approach to extend the weighted mean to ordinal scales. The general idea of the operator is to translate addition and product in the weighted mean by similar operations in the ordinal scale. Two operations of multivalued logics are selected for this purpose: t-norms and t-conorms.

T-conorms are addition-like operators that satisfy monotonicity, commutativity, and associativity and have as a neutral element the value 0 (l_0 in the ordinal scale $L = \{l_0, \cdots, l_R\}$). T-norms are product-like operators that satisfy the same properties except for the neutral element that in this case is 1 (l_R when defined in the ordinal scales L).

OWM assumes that weights are natural numbers. Then, the multiplication of a weight by a value corresponds to multiple additions of the corresponding value.

Here, addition is achieved through the t-conorm. Because the ordinal scale usually is not enough to accumulate all the values to be aggregated, a new scale is introduced that extends the original scale. This new scale is the product of the subset of natural numbers $\{1, \cdots, N\}$ (where N is the number of values to be aggregated) and the original scale. Once the accumulated value is obtained in this new scale, division by the accumulation of the weights leads to the final aggregated value.

Extensions of this operator exist that consider scales other than natural numbers for the weights. Also, the same approach was applied to extend the CI[2] to ordinal scales. This is the so-called ordinal CI (OCI). The CI is the natural extension of the weighted mean to the case of considering numerical fuzzy measures. In some way, SIs are the ordinal counterpart of CIs.

2.5. Considering Weights in Aggregation Operators

Aggregation operators use parameters for expressing additional knowledge about the values, the sources, and its current application. The following are some of the common uses of the parameters:

Expressing importances of individual information sources. This is the typical case of weighting vectors in WMs and similar aggregation operators (weighted maximum, weighted minimum, plurality rule, and median). We associate to each source a weight in a given scale. The larger the weight, the more important is the source in determining the aggregated value.

Expressing importances of values. This is the approach considered in the OWA operator (operator defined by Yager in Ref. 18; see also Ref. 19 about including other types of weights). Weights do not measure the importance of a source but of the values. For instance, it is possible to give more importance to small values than to larger ones. This would be the case if a robot fuses estimated distance to a nearby object; it is more important to consider small values than larger values to avoid collisions. OWA operators and related operators (e.g., the CI that generalizes OWA operators) can be used for this purpose.

Expressing importances of sets of information sources. This is the case of the SI and other similar operators (e.g., the CI[17] and the fuzzy t-conorm integral[20]). These operators do not only allow expression of the importance of a particular information source, but also allow the importance of a set of sources. Fuzzy measures can be used to represent this information. In the numerical case, it can be proven that fuzzy measures can be used to represent both the importances of the individuals and the importance of the values.

2.6. Summary of Aggregation Operators in Ordinal Scales

Table I gives an overview of the main characteristics of the aggregation operators reviewed so far. The first row shows whether the function can be used for an arbitrary number of values to be aggregated and the parameters required, if any.

Table I. Characteristics of ordinal aggregation operators.

	P	*WP*	*M*	WM	OS	WOS	SI	WMax	OWM	OCI
1	√	w	√	w	if $i \in I$	w, if $i \in I$	μ	π	w $(O, \oplus, \otimes, n)$	μ $(O, \oplus, \otimes, n)$
2	0	N	0	N	1	$N + 1$	2^N	N	N	2^N
3		$\mathbb{R}, \mathbb{N}$ $(O, \oplus)$		$\mathbb{R}, \mathbb{N}$ $(O, \oplus, n)$		$\mathbb{R}, \mathbb{N}$ $(O, \oplus, n)$	L	L	$\mathbb{N}$ $(O, \oplus)$	$\mathbb{N}$ $(O, \oplus)$
4	*X*	√	*X*	√	*X*	√	√	√	√	√
5	*X*	*X*	*X*	*X*	*X*	*X*	√	*X*	*X*	√
6	*X*	*X*	*X*	*X*	*X*	*X*	*X*	*X*	√	√

The symbol √ means that the characteristic is always fulfilled; *X* means that it is never possible; other values correspond to particular characteristics. Here, *I* stands for the unit interval; *O* corresponds to an arbitrary ordinal scale; $(O, \oplus)$ corresponds to an ordinal scale with a t-conorm; $(O, \oplus, n)$ corresponds to an arbitrary ordinal scale with a t-conorm and a negation; and $(0, \oplus, \otimes, n)$ corresponds to an ordinal scale with a t-conorm, a t-norm, and a negation.

In fact, all functions can be applied to an arbitrary number of parameters easily. In the case of the order statistics, it is appropriate that the parameter used is a real number in the unit interval in order that the selection of the ith element does not change the meaning when additional elements are considered. With a real number, the parameter corresponds to the selection of the element that occupies the i percent.

The second row is the number of parameters required when the number of values to be aggregated is N.

The third row is the range of the weights (if any). In this row, O corresponds to an arbitrary ordinal scale and L is used when the scale should be the same as the one for the values to be aggregated. The $\oplus$, $\otimes$, and n stand for t-conorm, t-norm, and negation functions, respectively, over O. The terms $\mathbb{R}$ and $\mathbb{N}$ stand for real and natural numbers, respectively.

The fourth row shows whether the aggregation procedure allows the weighting of the sources. The fifth row shows the weighting of the values. Positive marks are given for the SI and the CI to both kinds of weights because fuzzy measures can be defined to express this information. However, for measures in ordinal scales, it is difficult to model at the same time the weighting of the sources and the weighting of the elements. This is not the case in the numerical setting when the measure can be built from two weighting vectors, one modeling each alternative (as for the WOWA in Ref. 8).

The last row shows the possibility of obtaining a value that is not present in the original set of values to be aggregated.

3. AGGREGATION PROCEDURES FOR PROTOTYPE CONSTRUCTION

In this section, we review the difficulties of using the aggregation procedures reviewed so far when applied to building prototypes within clustering methods.

Although our point of view is biased to clustering methods for microdata protection, the analysis is applicable to most clustering problems.

Clustering methods are applied to multidimensional data to build a set of clusters in which similar elements are put together and dissimilar elements are left into different classes. One of the open problems in clustering is how to deal with categorical data. In fact, several difficulties arise in this case: computation of similarities between categories, combination of similarities when each individual is represented in terms of different variables evaluated in different scales, and prototype calculation for each cluster. In this study we are interested in the latest problem: the computation of the cluster prototype.

The computation of the prototype usually is achieved in numerical scales using some kind of aggregation procedure, usually an arithmetic mean, although some other aggregation operators are conceivable, in particular, the weighted mean (e.g., to give different importance to different individuals in the cluster as in Ref. 21) or the OWA (e.g., to give more importance to central elements than to elements with large or small values as in Ref. 22).

In the case of categorical data, the methods described in Section 2 are applicable. Now, we consider in detail the applicability of each method for prototype calculation.

3.1. Plurality Rule

The application of the majority rule is straightforward. However, some inconveniences can be distinguished. The first one is that the majority rule returns a set of the most frequent values. Therefore, when the prototype is a single value, a selection procedure has to be considered to select one of the values. Another drawback is that the function does not allow for compensation. We understand here for compensation the fact that when the data to be fused contain two values a_i and a_j, the output can be a value in between, say a_k, regardless, $a_k \in \{a_1, \cdots, a_N\}$ or not. In other words, an aggregation function $\mathbb{C}$ is not compensative if for all a_i, $a_j \in \{a_1, \cdots, a_N\}$, the aggregated value is always one of the original ones: $\mathbb{C}(a_1, \cdots, a_N) \in \{a_1, \cdots, a_N\}$ for all $a_i \in L$.

Therefore, when large and small values but not medium ones are fused, the final value will be either a large or a small one. Note that in the numerical case, the mean $x = \sum_i x_i/N$ minimizes $\sum_i (x_i - \bar{x})^{17}$, and the selection of a large value (or a small one) instead of $\bar{x}$ would give a larger difference.

An additional difficulty of this lack of compensation is that when the number of values to aggregate is small, small variations on the elements can provoke large modifications of the output; e.g., the aggregation of the values l_0, l_0, l_3, and l_4 is l_0 and the aggregation of the values l_4, l_0, l_3, and l_4 is l_4. Thus, a small modification of the inputs (a single value) results in a large variation of the output (from l_0 to l_4).

Weighted median has an additional difficulty: it is not always possible to have available the required weighting vector. This is so because in prototype selection, it would require a weight for each individual. Although this is possible in some applications (see, e.g., Ref. 21), this is not the general case. Usually, all the elements are equally representative for all the application domain.

Special difficulties arise when weights are not numerical but defined in ordinal scales. This is so because not all the clusters have the same number of elements; therefore, normalization is required in each cluster [otherwise, with a few elements we can get that all elements a_i have $acc(a_i)$ equal to 1, and, therefore, selection is not possible]. Also, selection of the appropriate t-conorm is not an easy task, specially for non-experienced users.

In addition, in relation to weights, no weights for the values are considered in the function.

3.2. Median

The application of the median operator for prototype selection is straightforward. However, it presents some of the drawbacks of the plurality rule: the median always returns one of the values to be aggregated (e.g., the median of l_0, l_{N-1}, l_N is l_{N-1} while a straight average of the indices gives $l_{(2M-1)/3}$); it does not allow consideration of weights for the values; and the same comments about the weighting vector given for the plurality rule apply to this case. Order statistics have similar properties although in this case, the weight allows the selection of values other than the central one.

3.3. SI

The main difficulty for the application of the SI in the setting of prototype selection is the definition of the corresponding fuzzy measure. According to the definition of the integral, the fuzzy measure has to be defined into L (the same domain used for a_i). Several difficulties apply in this case: defining measures for all possible clusters requires a huge number of fuzzy measures (only parameterized families of fuzzy measures can be used, and parameterization is difficult in ordinal scales); when several variables are used in the clustering process, fuzzy measures have to be defined for each variable (the set L usually changes for each variable and the fuzzy measures has to be defined on the same scale as that of the variable), and this increases the complexity of this definition; for each variable and each set of N sources, 2^N values are required.

Another drawback of the SI is that it does not allow for compensation. In fact, this statement has to be tinged because the final value can be different from the original ones. This is possible because the final value can be one of the ones used by the fuzzy measure. This can cause some sort of compensation.

Some of these difficulties also apply to weighted maximum.

3.4. OWM

The main difficulty for using the OWM mean is the requirement of a t-norm and t-conorm for the domains of the variables. This means having one pair (t-norm and t-conorm) for each of the variables.

In conclusion, we can say that the two most relevant difficulties for applying the aforementioned aggregation operators for prototype construction is that most operators do not allow for compensation and that most of them do not allow for weighting the sources.

Detailed analysis of the methods shows that the most relevant operation for the problem of prototype selection is the median. That is, in fact, the operator usually considered as the ordinal counterpart of the weighted mean. The SI and OWM are specially difficult to apply in the prototype construction setting because of, respectively, the need of fuzzy measures and definitions of t-norms and t-conorms.

In the next section we introduce WOW operators for including compensation and weighting for the sources to categorical aggregation operators. Then, we particularize the approach to the case of the median.

4. WEIGHTING OF VALUES AND COMPENSATION

Our approach to include the weights for the values is based on the weighted OWA (WOWA) operator defined in Ref. 6. This operator is a generalization of both the weighted mean and the OWA operator (defined by Yager[18]) allowing users to have in a single operator the parameters of both operators. In fact, both weighted mean and OWA have parameters of the same form (weighting vectors: positive weights that add to one). However, in spite of having the same form, the parameters have different meaning. Let us recall both operators.

DEFINITION 9. *A vector* $\mathbf{v} = (v_1 \ldots v_N)$ *is a weighting vector of dimension N if and only if* $v_i \in [0, 1]$ *and* $\sum_i v_i = 1$.

DEFINITION 10. *Let* $\mathbf{p}$ *be a weighting vector of dimension N; then, a mapping weighted mean:* $\mathbb{R}^N \rightarrow \mathbb{R}$ *is a weighted mean of dimension N if weighted* $mean_{\mathbf{p}}(a_1, \ldots, a_N) = \sum_i p_i a_i$.

DEFINITION 11. *Let* $\mathbf{w}$ *be a weighting vector of dimension N; then, a mapping OWA* : $\mathbb{R}^N \rightarrow \mathbb{R}$ *is an OWA operator of dimension N if*

$$\mathrm{OWA}_{\mathbf{w}}(a_1, \ldots, a_N) = \sum_{i=1}^{N} w_i a_{\sigma(i)}$$

where $\{\sigma(1), \ldots, \sigma(N)\}$ *is a permutation of* $\{1, \ldots, N\}$ *that such* $a_{\sigma(i-1)} \geq a_{\sigma(i)}$ *for all* $i = \{2, \ldots, N\}$ *(i.e.,* $a_{\sigma(i)}$ *is the ith largest element in the collection* $a_1, \ldots, a_N$*).*

Similarities and differences between both operators can be underlined as follows:

- The weighted mean is a linear combination of weights and values where the weights are linked to the values we aggregate. Usually, this is understood as the importance or reliability of the information sources. The larger the weight, the more influence the corresponding value has on the final output. The smaller the weight, the lesser influence the corresponding value has on the final output.

- The OWA operator also is a linear combination of weights and values. However, in this operator, weights are not linked to the values themselves but on their relative position. Note that any permutation π of the values to be aggregated leads to the same result: $\mathrm{OWA}_{\mathbf{p}}(a_1, \ldots, a_N) = \mathrm{OWA}_{\mathbf{p}}(a_{\pi(1)}, \ldots, a_{\pi(N)})$.

The WOWA operator that generalizes both operators is defined as follows.

DEFINITION 12. *Let* $\mathbf{p}$ *and* $\mathbf{w}$ *be two weighting vectors of dimension* N; *then, a mapping WOWA* : $\mathbb{R}^N \rightarrow \mathbb{R}$ *is a WOWA operator of dimension N if*

$$\mathrm{WOWA}_{\mathbf{p},\mathbf{w}}(a_1, \ldots, a_N) = \sum_i \boldsymbol{\omega}_i a_{\sigma(i)}$$

where σ *is defined as in the case of the OWA* (*i.e.*, $a_{\sigma(i)}$ *is the ith largest element in the collection* $a_1, \ldots, a_N$), *and the weight* $\boldsymbol{\omega}_i$ *is defined as*

$$\boldsymbol{\omega}_i = w^*\left(\sum_{j\leq i} p_{\sigma(j)}\right) - w^*\left(\sum_{j<i} p_{\sigma(j)}\right)$$

with w^* *being a monotonic increasing function that interpolates the points* $(i/N, \sum_{j\leq i} w_j)$ *together with the point* (0, 0). *The function* w^* *is required to be a straight line when the points can be interpolated in this way.*

In this definition, the weighting vector $\mathbf{p}$ corresponds to the weighting vector of the weighted mean and $\mathbf{w}$ corresponds to the weighting vector of the OWA operator. Then, $\boldsymbol{\omega}$ is a new weighting vector that considers the interactions between $\mathbf{p}$ and $\mathbf{w}$.

The function w^* built previously from the vector $\mathbf{w}$ can be understood as a fuzzy quantifier (a nondecreasing fuzzy quantifier) and the weights $\mathbf{p}$ can be seen as a probability distribution. A nondecreasing fuzzy quantifier is a monotonic function Q [i.e., $Q(a) \geq Q(b)$ for all $a > b$] such that $Q(0) = 0$ and $Q(1) = 1$.

In the definitions given previously, weighting vectors are presented in conjunction with the definition of the operator. However, these vectors and their transformation can be established without the corresponding operator and used in other families of operators. This is defined using the nondecreasing fuzzy quantifier Q (Q can be interpolated from $\mathbf{w}$ when required as shown previously).

DEFINITION 13. *Let* $(a_i, p_i)_{i=1,N}$ *be a pair defined by a value and the importance of* a_i *expressed in a given domain* $D \subset \mathbb{R}^+$, *and let Q be a fuzzy nondecreasing fuzzy quantifier. Then, the WOW weighting vector* $\boldsymbol{\omega} = (\boldsymbol{\omega}_1, \cdots, \boldsymbol{\omega}_N)$ *for* $(\mathbf{a}, \mathbf{p})$ *and Q is defined as follows:*

$$\boldsymbol{\omega}_i = Q\left(\frac{\sum_{j\leq i} p_{\sigma(i)}}{\sum_{j\in L} p_{\sigma(i)}}\right) - Q\left(\frac{\sum_{j<i} p_{\sigma(j)}}{\sum_{j\in L} p_{\sigma(i)}}\right)$$

where σ *is a permutation as aforementioned such that* $a_{\sigma(i-1)} \geq a_{\sigma(i)}$.

This definition permits us to include the weighting of the sources to aggregation operators for categorical data. The following definition exploits this fact to define a WOW $\mathbb{C}$ operator from an operator $\mathbb{C}$.

Definition 14. *Let* $\mathbf{X} = \{x_1, \ldots, x_N\}$ *be a set of information sources; let* a_i *be the value supplied by the source* x_i*; let* $\mathbb{C}$ *be an aggregation operator with parameter* $\mathbf{p} : X \to D$*; and let* Q *be a nondecreasing fuzzy quantifier* Q*. Then, the WOW* $\mathbb{C}$ *operator is defined as follows*:

$$\text{WOW } \mathbb{C}_{\mathbf{p}Q}(a_1, \cdots, a_N) = \mathbb{C}_{\boldsymbol{\omega}}(a_1, \cdots, a_N)$$

where $\boldsymbol{\omega}$ *is the WOW weighting vector of* $(a_i, p_i)_{i=1,N}$ *and* Q *following Definition 13.*

The second aspect to be introduced in the aggregation process is compensation. This is achieved, following Ref. 23, making *data values convex.* Recall that compensation corresponds to the case that values a_k such that $\min_i a_i < a_k < \max_i a_i$ can be selected, although they are not in the set $\{a_1, \cdots, a_N\}$. Our approach to allow compensation is to redefine the function acc in Definition 4 so that $acc(a_k) \neq 0$. In this way, a_k can be selected by the aggregation function.

Definition 15. *Let* $\mathbf{p} : X \to D \subset \mathbb{R}$ *be a weighting vector; then, a mapping* $CWM_{\mathbf{p}} : L^N \to L$ *is a convex weighted median of dimension N if*

$$CWM_{\mathbf{p}}(a_1, \cdots, a_N) = a \text{ iff } acc'''(a) > 0.5 \geq acc'''(b)$$

where $acc'''(a) = \sum b \leq a\ acc''(b)$, $acc''(a) = acc'(a)/\sum_{b \in L} acc'(b)$, $acc'(a) = \min(\max b \leq a\ acc(b), \max_{b \geq a} acc(b))$, *and* $acc(a) = \sum_{f(x_j)=a} p(x_j)$ *and where b is the element next to b in L. This is,* $b = \max\{x | x \in L, x < a\}$.

Now, we show the application of these two procedures (the one for weighting the values and the one for allowing compensation) to the median and to the plurality rule. This application leads to the convex WOW median (CWOW Me) and the CWOW *plurality rule.*

Definition 16. *Let* $\mathbf{p} : X \to D \subset \mathbb{R}$ *be a weighting vector and let Q be a nondecreasing fuzzy quantifier; then, a mapping* $CWOWM_{\mathbf{p}} : L^N \to L$ *is a convex WOW median of dimension N if*

$$\text{CWOWM}_{\mathbf{w}}(a_1, \cdots, a_N) = a \text{ iff } acc^{iv}(a) > 0.5 \geq acc^{iv}(b)$$

where $acc^{iv}(a) = \sum_{b \leq a} acc'''(b)$, acc''' *is the WOW weighting vector of* (L, acc'') *and* Q, $acc''(a) = acc'(a)/\sum_{b \in L} acc'(b)$, $acc'(a) = \min(\max_{b \leq a} acc(b), \max_{b \geq a} acc(\mathbf{b}))$, *and* $acc(a) = \sum_{f(x_j)=a} p(x_j)$ *and where b is the element next to b in L. This is* $b = \max\{x | x \in L, x < a\}$.

Definition 17. *Let* $\mathbf{w}$ *be a weighting vector and Q a nondecreasing fuzzy quantifier; then, a mapping* $WP_{\mathbf{w}} : L^N \to \wp(L)$ *is a CWOW plurality rule when* $P_{\mathbf{w}}(a_1, \cdots, a_N)$ *is the set of all those y in L so that no z in L,* $acc''(z) > acc''(y)$

Table II. Information sources and values to be aggregated.

Sources	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
Values	l_4	l_1	l_0	l_2	l_1	l_0	l_4	l_5	l_1

where $acc''(a)$ *is the WOW weighting vector of* (L, acc') *and* Q, $acc'(a) = \min(\max_{b\le a} acc(b), \max_{b\ge a} acc(b))$, *and* $acc(a) = \sum_{f(x_j)=a} w(x_j)$.

4.1. CWOWM

In this section, we study the CWOWM procedure defined in Definition 16. We begin giving an example that shows the suitability of the approach for obtaining, with appropriate parameterizations, values between the minimum and the maximum of the value to be aggregated. Then, we analyze the properties of the operator, focusing in the monotonicity condition.

Example 1. Let $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9\}$ be a set of information sources; let $f(x_i) = a_i$ be defined as in Table II (here $L = \{l_0, l_1, l_2, l_3, l_4, l_5, l_6\}$); and let $p(x_i) = 1$ for all x_i; then, the CWOWM for $Q(x) = x^\alpha$ for $\alpha \in \{1/8, 1/4, 1/2, 1, 2, 4, 8\}$ is given in Table III. This table includes the computed functions acc, acc', and acc'' that are common for all CWOWM operators and then the function acc''' for each considered α. The last row of the column describes the aggregated values CWOWM for each of the values and also the median for the original data (first row, acc row) and for the convex weighted median (CWM; second and third row, denoted by acc' and acc'' rows).

This example shows that the CWOWM permits the compensation inconvenience faced by the original median operator to be overcome. Note that it is possible to obtain l_3 as the output when $\alpha = 2$ while l_3 was not one of the values to be aggregated. It also can be observed that the operator, by means of the α parameter, permits values between the minimum and the maximum of the a_i to be obtained. In our case, the function moves from l_0 to l_5. Moreover, the function cannot result in values larger than the maximum of the a_i or smaller than the

Table III. The CWOWM for $\alpha \in \{1/8, 1/4, 1/2, 1, 2, 4, 8\}$.

	l_0	l_1	l_2	l_3	l_4	l_5	l_6	CWOWM
acc	2	3	1	0	2	1	0	l_4
acc'	2	3	2	2	2	1	0	l_2
acc''	2/12	3/12	2/12	2/12	2/12	1/12	0	l_2
acc''' ($\alpha = 1/8$)	0.7993	0.0970	0.0385	0.0298	0.0245	0.0108	0.0	l_0
acc''' ($\alpha = 1/4$)	0.6389	0.1644	0.0705	0.0566	0.0478	0.0215	0.0	l_0
acc''' ($\alpha = 1/2$)	0.4082	0.2372	0.1182	0.1022	0.0914	0.0425	0.0	l_1
acc''' ($\alpha = 1$)	0.1666	0.25	0.1666	0.1666	0.1666	0.0833	0.0	l_2
acc''' ($\alpha = 2$)	0.0277	0.1458	0.1666	0.2222	0.2777	0.1597	0.0	l_3
acc''' ($\alpha = 4$)	0.0007	0.0293	0.0856	0.2006	0.3896	0.2939	0.0	l_4
acc''' ($\alpha = 8$)	0.0000	0.0009	0.0124	0.0867	0.3984	0.5014	0.0	l_5

Table IV. Example of nonmonotonicity for the CWOWM.

	l_0	l_1	l_2	l_3	l_4	l_5	l_6	l_7	l_8	l_9	l_{10}	CWOWM
acc	6	1	1	1	1	1	1	1	1	1	1	l_3
acc′	6	1	1	1	1	1	1	1	1	1	1	l_3
acc″	6/16	1/16	1/16	1/16	1/16	1/16	1/16	1/16	1/16	1/16	1/16	l_3
acc	5	1	2	1	1	1	1	1	1	1	1	l_3
acc′	5	2	2	1	1	1	1	1	1	1	1	l_2
acc″	5/17	2/17	2/17	1/17	1/17	1/17	1/17	1/17	1/17	1/17	1/17	l_2

minimum of the a_i. This fact also implies that the operator satisfies unanimity (if all sources agree on a value l_i, the outcome is the value l_i)

PROPOSITION 1. *CWOWM is an aggregation operator satisfying*

(1) $\min(a_1, \cdots, a_N) \leq CWOWM(a_1, \cdots, a_N) \leq \max(a_1, \cdots, a_N)$
(2) *Unanimity:* $CWOWM(l, l, \cdots, l) = l$ *for all* $l \in L$

Nevertheless, this operator presents a drawback. The following proposition establishes this negative property.

PROPOSITION 2. *The CWOWM does not satisfy monotonicity, i.e., it does not hold*

$$\text{CWOWM}(a_1, \cdots, a_N) \leq \text{CWOWM}(a'_1, \cdots, a'_N)$$

for some $a_i \leq a'_i$, *where* $i \in \{1, \cdots, N\}$.

Nonmonotonicity is a consequence of the fact of making the function *acc* convex. Augmenting the values of *acc* for all the elements below the previous median value can violate monotonicity. This is illustrated in the following example.

Example 2. Let us consider 16 information sources $X = \{x_1, x_2, \cdots, x_{16}\}$ giving information over a set L of 11 ordered categories $L = \{l_0, l_1, \cdots, l_{10}\}$. The information supplied by the sources is as follows: 6 of the sources supply the value l_0 and the other 10 sources supply the values $l_1, l_2, \cdots, l_{10}$, i.e.,

$$a = (l_0, l_0, l_0, l_0, l_0, l_0, l_1, l_2, \cdots, l_9, l_{10})$$

To aggregate this values, the CWOWM is used. The corresponding *acc* function is given in the first row of Table IV. The application of the simple median to these values is given in the last column of the first row. The second and the third row of this table gives the *acc′* and *acc″* functions, i.e., the convex function and the normalized function (the one that adds to one). The last column of these rows shows the value of the CWOWM, l_3.

Let us now consider that one of the sources that supplied the category l_0 (say x_1) changes the value by l_2. The corresponding **a′** vector is now

$$\mathbf{a}' = (l_2, l_0, l_0, l_0, l_0, l_0, l_1, l_2, \cdots, l_9, l_{10})$$

Note that this vector is monotonic increasing in relation to the previous vector **a** because $a'_i \geq a_i$ for all $i \in \{1, \cdots, N\}$.

The corresponding *acc* function is given in the fourth row of Table IV. The last column of this row gives the median of the values. The median is a monotonic function and it can be seen that in this case, the final value is not modified by the change of l_0 by l_2. In the last two columns of this table, functions acc' and acc'' are displayed. The last column in the rows give the result for the CWOWM function l_2.

The example shows that monotonicity is not satisfied because changing the value $a_1 = l_0$ by $a'_1 = l_2$ (and keeping all the others $a'_i = a_i$), the outcome of the function is l_2 instead of l_3 and thus violates the equation

$$\mathrm{CWOWM}(a_1, \cdots, a_N) \leq \mathrm{CWOWM}(a'_1, \cdots, a'_N)$$

The violation of the monotonicity condition is caused by several factors (see Table IV): (i) the replacement of the value l_0 by l_2 also causes changes in the frequency of l_1 (see acc' in Table IV), and thus increments the total number of values to be considered by the median from 16 to 17 (see denominators in row acc''); (ii) the values l_0 and l_2 and also the additional value l_1 are smaller than l_3 and thus l_1 forces the decrement of the final outcome. Both factors are caused by the process of making acc' a convex function (in fact, incrementing the number of values l_i smaller than l_3). Note that for the original median function, the final aggregated value is not modified (the function is indeed monotonic).

Nevertheless, although these examples do not satisfy the monotonicity condition, it is clear that variations on the result are small (one label is changed by a contiguous one) and can be accepted from the point of view that we are using ordinal scales with no established semantics. In fact, the violation of the monotonicity condition is found when for a category l_i the acc''' function $[acc'''(l_i)]$ is near the cutting point 0.5. Note that $acc'''(l_2) = 0.5$ and $acc'''(l_3) = 0.5625$ for the **a** vector, and that $acc'''(l_2) = 0.5294118$ for **a**$'$. On the light of ordinal scales as scales with some uncertainty (e.g., imprecision or fuzzy terms), we can understand nonmonotonicity results as errors in the limits of the meaning of the category.

In general, it has to be said that it is possible to find examples of nonmonotonicity in which replacing a value l_a by a larger value l_b results in a large change of the outcome. However, this requires a set L with a large number of categories and a large set of sources, a situation that is not common when dealing with ordinal scales (especially, the case of having a large set of categories). This is illustrated in the following example.

Example 3. Let us consider a set X consisting of 1006 information sources, each supplying a value in the ordinal scale $L = \{l_0, l_1, l_2, \cdots, l_{1000}\}$. Let x_i supply the value l_i for $i = 1, \cdots, 1000$ and let $x_{1001}, \cdots, x_{1006}$ supply the value l_0. Then, the CWOWM of these values is l_{498}.

Let us consider now that x_{1001} replaces the value l_0 by the value l_{250}; then, the CWOWM is l_{373}.

As $l_{250} > l_0$, but $l_{373} < l_{498}$, the monotonicity condition is not satisfied. In this case, the number of categories between the original value and the new one is large but the number of categories in the set L is also very large.

To have a better understanding of the situations in which the CWOWM violates monotonicity (this understanding is required to apply the aggregation operator properly), we have studied in detail different situations and analyzed them to know whether the operator satisfies monotonicity or not.

We have considered two different scenarios and randomly generated several instantiations. In each instantiation, two monotonic vectors $\mathbf{a}^1 = (a_1^1, \cdots, a_N^1)$ and $\mathbf{a}^2 = (a_1^2, \cdots, a_N^2)$ (i.e., $a_i^1 \leq a_i^2$) were generated and the CWOWM was applied to them. Monotonicity was then checked.

In both scenarios, we consider an ordinal scale consisting on l categories ($l = R + 1$ using the notation $L = \{l_0, l_1, \cdots, l_R\}$ used so far) and N information sources and that the difference between vectors $\mathbf{a}^1$ and $\mathbf{a}^2$ is that K information sources have changed their value in $\mathbf{a}^1$ by a larger one in $\mathbf{a}^2$ (i.e., $|\{a_i | a_i^1 \neq a_i^2\}| \approx K$). For each scenario, m random instantiations have been considered.

The following are the two scenarios that we studied:

(1) All the sources changing a value in $\mathbf{a}^1$ to another one in $\mathbf{a}^2$ had the same value in $\mathbf{a}^1$ and change to the same value in $\mathbf{a}^2$. This is, for all i such that $a_i^1 \neq a_i^2$, $a_i^1 = \alpha$, and $a_i^2 = \beta$. In this case, if for a given parameterization K, the number of categories $a_i^1 = \alpha$ is K' with $K' < K$, only K' sources will change their value.
(2) Sources that change their values can have different values both in $\mathbf{a}^1$ and $\mathbf{a}^2$.

According to this, for each of the scenarios, an example is defined according to four parameters (l, N, K, and m). For evaluating the aggregation function, we have considered the following parameters for the two scenarios:

- The number of categories: $l = 2, 3, 4, 5, 6, 7, 8, 9, 10, 20, 30, 50, 100$
- The number of sources: $N = 5, 10, 15, 20, 25, 30, 40, 50, 75, 100, 200, 500, 1000$
- The number of changed values: $K = 2, 3, 4, 5, 10, 100$

Experiments were run either 1000 or 10,000 times ($m = 1000$ or $m = 10{,}000$). The results of the experiments are displayed in Tables V–XII. Tables show the number of cases that violate monotonicity. This is, each cell of the table indicates how many times the monotonicity condition was violated when m experiments were executed.

These experiments were programmed in CLisp (running on RedHat 6.2 for a PC) and for scenario 1 with $K = 2$ and $m = 10{,}000$ it took 3 hours to compute all examples for all considered pairs of N and l (this is shown in Table VI). Instead, the computation of the Table XII (scenario 2, $K = 100$ and $m = 10{,}000$) took about 6 hours.

From the tables, it can be observed that for a small number of categories the number of monotonicity violations is small (<3%). This number is even smaller for the second scenario. The experiments also show that for the second scenario when the value K increases, the percentage of violations decreases (especially for the experiments with a small number of categories—see, e.g., Tables XI and XII

Table V. Experiments for scenario 1 with $K = 1$ (number of changed values from $\mathbf{a}^1$ to $\mathbf{a}^2$).

l\N	5	10	15	20	25	30	40	50	75	100	200	500	1000
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0
4	23	17	5	3	2	1	1	0	0	0	0	0	0
5	19	43	28	25	29	21	12	13	3	0	1	0	0
6	34	69	53	34	37	22	26	18	15	13	15	7	5
7	37	104	93	80	66	52	48	34	34	16	4	0	0
8	39	108	105	103	79	74	45	38	26	17	15	10	1
9	41	168	136	131	105	100	75	66	52	46	14	3	0
10	27	146	133	132	98	141	77	70	49	38	28	22	16
20	20	211	214	185	192	216	183	183	127	131	69	28	20
30	4	143	212	191	196	235	236	234	188	179	104	59	28
50	2644	3592	3617	3789	3750	3714	3766	3728	3667	3677	3632	1933	1115
100	0	27	220	697	1550	2695	5230	6649	6934	6846	6886	3168	111

Rows correspond to different number of labels (parameter l) and columns correspond to different number of information sources (parameter N). Ten thousand tests have been performed for each experiment.

and compare with Table X). For the first scenario, conclusions are not so clear, but it seems that the larger the values of K, the number of violations decreases for a small number of information sources and increases for a larger number of sources. For example, for $K = 1$ (see Table V) and $N = 10$ and $N = 15$, the cells for $l = 10$ are about 140 and for $K = 5$ (see Table VII), the same cells are about 100, for $K = 10$ (see Table VIII) they are about 85, for $K = 100$ (see Table IX) they are also about 85. Instead, the corresponding cells ($l = 10$) for $N = 40$ and $N = 50$ are, respectively, for $K = 1$ about 75, for $K = 5$ about 148, for $K = 10$, 127, and 164, for $K = 100$, 140, and 178. Thus, the number of violations tends to decrease for a small number of sources.

Table VI. Experiments for scenario 1 with $K = 2$.

l\N	5	10	15	20	25	30	40	50	75	100	200	500	1000
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0
4	15	26	5	2	2	3	0	0	0	0	0	0	0
5	17	38	29	28	30	27	26	19	7	3	0	0	0
6	31	54	58	67	55	37	33	28	16	11	28	7	15
7	23	72	91	70	79	81	58	62	35	27	6	0	0
8	41	62	84	101	92	106	68	76	54	43	37	20	7
9	21	90	110	106	133	109	112	94	84	65	31	2	0
10	27	94	111	118	146	137	99	104	82	73	39	19	16
20	14	112	109	146	206	218	213	224	185	182	111	68	35
30	6	74	97	113	181	215	246	269	223	257	177	92	51
50	734	1010	2922	3625	3729	3705	3730	3770	3744	3710	3809	3688	2318
100	0	0	3	3	9	12	50	123	1949	6228	6905	6647	616

Rows correspond to number of labels and columns correspond to number of information sources. Ten thousand tests have been performed for each experiment.

Table VII. Experiments for scenario 1 with $K = 5$.

l\N	5	10	15	20	25	30	40	50	75	100	200	500	1000
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0
4	14	31	4	11	0	1	1	0	0	0	0	0	0
5	13	44	40	42	40	39	41	21	21	6	0	0	0
6	25	52	72	65	70	76	72	70	49	42	36	28	23
7	31	65	74	93	84	83	97	101	78	50	17	0	0
8	30	90	79	114	125	101	115	105	86	75	56	48	28
9	23	79	103	99	125	126	155	148	97	96	44	9	0
10	27	87	104	115	146	124	148	148	115	116	80	45	35
20	14	106	116	135	137	155	179	221	237	253	175	98	81
30	5	64	121	114	140	183	188	212	234	298	273	179	100
50	697	243	123	103	323	735	2033	3114	3692	3773	3835	3731	3709
100	0	0	1	6	7	13	13	13	15	31	202	6951	6866

Rows correspond to number of labels and columns correspond to number of information sources. Ten thousand tests have been performed for each experiment.

The worst cases are found for large number of categories (≥30). In this case, a small variation of the input data implies a large modification of the convex function (this is the case of Example 3). For example, Table V shows that for 50 categories and 5 information sources, there is 26.44% of the cases that do not satisfy monotonicity, for 100 categories and 75 sources we have almost 70% of the cases.

From the point of view of aggregation for prototype selection, the experiments show that the proposed aggregation method is a valid alternative because the usual number of categories usually is <15. For example, in the experiments in Ref. 12, the average number of categories is 13, only 30% of the variables have >15

Table VIII. Experiments for scenario 1 with $K = 10$.

l\N	5	10	15	20	25	30	40	50	75	100	200	500	1000
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0
4	14	27	14	5	2	1	1	0	0	0	0	0	0
5	25	42	24	54	46	52	47	42	17	11	1	0	0
6	30	63	71	78	85	73	106	102	93	83	57	38	31
7	29	65	87	89	88	116	107	114	95	73	18	1	0
8	28	72	84	88	119	97	119	147	156	151	82	71	54
9	24	80	82	122	112	131	148	141	158	182	67	11	1
10	31	83	85	97	111	155	127	164	172	175	126	79	66
20	13	82	99	136	150	158	183	191	248	259	295	180	90
30	6	74	96	116	133	160	195	225	261	315	414	315	171
50	744	256	117	68	52	40	31	49	310	1744	3790	3941	3779
100	0	0	0	5	5	13	18	18	25	28	47	458	6902

Rows correspond to number of labels and columns correspond to number of information sources. Ten thousand tests have been performed for each experiment.

Table IX. Experiments for scenario 1 with $K = 100$.

I\N	5	10	15	20	25	30	40	50	75	100	200	500	1000
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0
4	26	24	11	4	3	1	0	0	0	0	0	0	0
5	18	40	19	46	43	56	45	42	18	13	0	0	0
6	31	67	80	76	78	81	84	93	129	152	222	230	265
7	26	58	95	93	97	107	95	121	123	108	102	71	10
8	22	65	68	90	102	106	125	150	172	165	257	375	341
9	30	70	95	105	150	134	144	164	160	191	164	173	103
10	25	84	99	123	118	147	140	178	181	199	256	385	474
20	13	113	118	123	141	150	179	196	271	304	308	374	466
30	6	67	100	105	146	178	196	239	251	330	407	453	437
50	684	257	98	68	39	26	40	52	73	71	118	202	322
100	0	3	0	5	6	7	11	19	26	28	42	33	16

Rows correspond to number of labels and columns correspond to number of information sources. Ten thousand tests have been performed for each experiment.

categories and the variable with a larger number has 25 categories. In addition, in the particular case of clustering for microaggregation,[11,12] the number of values to be aggregated usually is <10. In a general clustering problem, this number will be quite larger but the number of categories will be about the same.

An additional element to be taken into account is that in our experiments the values are generated randomly and thus a given vector a can have very dissimilar values. However, when applying aggregation to clustering, the values would be similar. In fact, they have to be so because they are put together in the same cluster because they are similar. The effects of nonmonotonicity would be smaller in this latter case. Recall that nonmonotonicity is caused by the introduction of new

Table X. Experiments for scenario 2 with $K = 2$.

I\N	5	10	15	20	25	30	40	50	75	100	200	500	1000
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0
4	30	6	3	0	0	1	0	0	0	0	0	0	0
5	20	13	8	9	7	13	8	8	2	4	0	0	0
6	26	33	29	18	21	8	10	10	6	7	4	5	2
7	28	76	59	36	33	33	32	22	16	21	4	0	0
8	36	103	73	68	47	40	30	15	19	16	12	11	8
9	48	153	100	87	94	70	62	43	41	35	19	0	0
10	58	158	122	114	93	80	74	59	61	31	20	26	11
20	27	303	314	282	318	262	252	239	170	147	92	42	31
30	11	209	346	331	328	364	329	341	314	250	161	75	47
50	3392	4476	5033	5024	4949	4697	4514	4271	3864	3562	3152	2308	1410
100	1	15	131	367	783	1277	2318	2883	3591	6257	5910	4874	458

Rows correspond to number of labels and columns correspond to number of information sources. Ten thousand tests have been performed for each experiment.

Table XI. Experiments for scenario 2 with $K = 10$.

$l\backslash N$	5	10	15	20	25	30	40	50	75	100	200	500	1000
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0
4	2	0	0	0	0	0	0	0	0	0	0	0	0
5	1	0	0	0	0	0	0	0	0	0	0	0	0
6	1	1	0	0	0	0	0	0	0	0	0	0	0
7	3	2	0	0	1	0	0	0	0	0	0	0	0
8	1	3	0	1	1	0	1	2	0	1	0	0	0
9	10	8	5	1	1	1	2	0	1	1	2	0	0
10	21	14	10	5	2	2	3	3	3	1	2	2	2
20	53	284	148	169	128	120	62	69	48	50	42	23	21
30	41	405	437	353	337	332	269	242	208	163	123	66	41
50	2131	2121	3429	4569	5242	5809	6140	6009	5380	4970	4116	3359	2624
100	4	0	0	3	7	24	76	151	393	970	3301	8344	6666

Rows correspond to number of labels and columns correspond to number of information sources. Ten thousand tests have been performed for each experiment.

elements in the convex function acc'; therefore, when values are similar, the number of added elements will be small.

5. CONCLUSIONS

In this study we have reviewed existing aggregation operators in ordinal scales for their application to prototype construction. We have analyzed their drawbacks and we have proposed two general procedures to solve them. Then, we have applied these procedures to the median to define the CWOWM. We have analyzed some of the properties of this operator. We have seen that it satisfies unanimity and that the value belongs to the interval defined by the minimum and maximum of the values. We have shown that the procedure does not satisfy monotonicity. We have shown with an example that the modification of a single label does not modify in a substantial way the outcome (a label is changed by the contiguous one). Only for an example with a large domain L, the outcome of the

Table XII. Experiments for scenario 2 with $K = 100$.

$l\backslash N$	5	10	15	20	25	30	40	50	75	100	200	500	1000
2	0	0	0	0	0	0	0	0	0	0	0	0	0
⋮	⋮											⋮	⋮
10	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	4	1	0	1	0	0	0	0	0	0	0	1
50	416	286	91	39	20	10	13	83	954	2860	4747	3406	3093
100	5	0	0	0	0	0	0	0	0	0	0	0	21

Rows correspond to number of labels and columns correspond to number of information sources. Ten thousand tests have been performed for each experiment.

CWOWM is modified substantially. Experiments have confirmed that violations of monotonicity are not relevant for a small number of categories and of sources. This is the typical case in clustering and more specifically in microaggregation. Experiments show that monotonicity is not satisfied for a large proportion of scenarios when the number of categories is large. However, this is not a common situation.

As in usual applications, the number of categories in L is not large and the nonmonotonicity can be understood from the point of view of the uncertainty attached to categories (e.g., imprecision); we consider appropriate the use of CWOWM for prototype selection, particularly because it allows compensation for a property that the other operators lack, and also because it allows a parametric definition (through the quantifier) that allows the user to customize the application or to apply learning procedures. In particular, and as shown in Ref. 3, parameterization is a relevant aspect in microaggregation to find the best trade-off between information loss and the risk of releasing unprotected data.

Acknowledgments

We acknowledge partial support of the European Community under the contract "CASC" IST-2000-25069 and the Spanish Ministry of Science and Technology under the project "STREAMOBILE" (TIC2001-0633-C03-01/02).

References

1. Grabisch M, Nguyen HT, Walker EA. Fundamentals of uncertainty calculi with applications to fuzzy inference. Dordreht, The Netherlands: Kluwer Academic Publishers; 1995.
2. Herrera F, Herrera-Viedma E. Aggregation operators for linguistic weighted information. IEEE Trans Syst Man Cybern 1997;27:646–656.
3. Roberts FS. On the indicator function of the plurality function. Math Soc Sci 1991;22: 163–174.
4. Fishburn PC, Rubistein A. Aggregation of equivalence relations. J Classif 1986;3:61–65.
5. Torra V. Aggregation of linguistic labels when semantics is based on antonyms. Int J Intel Syste 2001;16:513–524.
6. Torra V. Negation functions based semantics for ordered linguistic labels. Int J Intell Syst 1996;11:975–988.
7. Herrera F, Herrera-Viedma E, Verdegay JL. A sequential selection process in group decision making with a linguistic assessment approach. Inf Sci 1995;85:223–239.
8. Torra V. The weighted OWA operator. Int J Intell Syst 1997;12:153–166.
9. Sugeno M. Theory of fuzzy integrals and its applications. PhD dissertation, Tokyo Institute of Technology, Tokyo, Japan, 1974.
10. Godo L, Torra V. On aggregation operators for ordinal qualitative information. IEEE Trans Fuzzy Syst 2000;8:143–154.
11. Domingo J, Torra V. Aggregation techniques for statistical confidentiality. In: Mesiar R, Calvo T, Mayor G, editors. Aggregation operators: New trends and applications. Berlin, Germany; Physica-Verlag, Springer; 2002. pp 260–271.
12. Domingo-Ferrer J, Torra V. A quantitative comparison of disclosure control methods for microdata. In: Doyle P, Lane JI, Theeuwes JJM, Zayatz LM, editors. Confidentiality, disclosure, and data access: Theory and practical applications for statistical agencies. Amsterdam, The Netherlands; Elsevier; 2001. pp 111–133.

13. Mayor G, Torrens J. On a class of operators for expert systems. Int J Intel Syst 1993;8: 771–778.
14. Marichal, J-L. Aggregation operators for multicriteria decision aid. PhD dissertation, Faculte des Sciences, Universite de Liege, Liege, Belgium, 1999.
15. Dubois D, Prade H. Weighted minimum and maximum operations. Inf Sci 1986;39:205–210.
16. Godo L, Torra V. Extending Choquet integrals for aggregation of ordinal values. Int J Uncert Fuzziness Knowl Bases. 2001;9(supplement):17–31.
17. Choquet G. Theory of capacities. Ann Inst Fourier 1953;5:131–296.
18. Yager RR. On ordered weighted averaging aggregation operators in multi-criteria decision making. IEEE Trans Syst Man Cybern 1988;18:183–190.
19. Yager RR. Quantifier guided aggregation using OWA operators. Int J Intell Syst 1996;11: 49–73.
20. Murofushi T, Sugeno M. Fuzzy t-conorm integral with respect to fuzzy measures: Generalization of Sugeno integral and Choquet integral. Fuzzy Sets Syst 1991;42:157–171.
21. Keller A, Klawonn F. Fuzzy clustering with weighting of data variables. Int J Uncertain Fuzziness Knowl Based Syst 2000;8:6.
22. Yager RR. On the control of hierarchical clustering. In: Proc IPMU, Madrid, Spain; 2000. pp 1335–1341.
23. Bardossy A, Duckstein L, Bogardi I. Combination of fuzzy numbers representing expert opinions. Fuzzy Sets Syst 1993;57:173–181.