Degenerate Arrays: A Framework for Uncertain Data Tables

Margaret Miró-Julià

Departament de Ciències, Matemàtiques i Informàtica ,
Universitat de les Illes Balears, 07122 Palma de Mallorca, Spain
margaret.miro@uib.es

Abstract. Boolean algebra provides an important model for the description of knowledge using a binary language. This paper considers a multivalued language, based on arrays and co-arrays, that allows a multivalued description of the knowledge contained in a data table.

This multivalued algebra has some special elements which are arryas and co-arrays at the same time: the degenerate arrays. These degenerate arrays are singled out and their interpretation analyzed, which gives rise to the introduction of the array projections and co-array projections.

Finally, a relation between array projections, co-array projections and uncertain data tables is examined.

1 Introduction

It is well known that the set of all subsets of a given set C (the power set of C), constitutes a Boolean algebra $< \rho(C), \cup, \cap, \widehat{\ }, \emptyset, C >$. If a symbolic representation or a description of subsets is considered, there is a parallel Boolean algebra $< \mathcal{S}_c, +, \cdot, \hat{\ }, \vee_c, \wedge_c >$ defined on the set $\mathcal{S}_c$ of all possible symbols describing subsets of C.

The special elements of $\mathcal{S}_c$ are $\vee_c$ the symbol describing the empty set, $\vee_c \nrightarrow \emptyset_C$, and $\wedge_c$ the symbol describing set C, $\wedge_c \nrightarrow C$. Throughout this paper, the expression $c_i \nrightarrow C_i$ may be read as "c_i is the symbol describing set C_i".

All the concepts, operations and special elements introduced above make reference to only one set of values, that is, one attribute. A data table has more than one attribute. Let's consider g sets $G, \ldots, B$ and A, the elements of each of these sets are the 1-spec-sets (one specification). A g-spec-set, $[g_k, \ldots, b_j, a_i]$, is a chain ordered description of g specifications, one from set $G, \ldots,$ one from set B and one from set A. Each spec-set represents itself and all possible permutations.

The cross product $G \otimes \cdots \otimes B \otimes A$ is the set of all possible g-spec-sets formed by one element of $G, \ldots,$ one element of B and one element of A. The set of all possible g-spec-sets induced by sets $G, \ldots, B$ and A is called the universe and every subset of the universe is called a subuniverse.

It is important to mention that the cross product is not the cartesian product. A g-spec-set represents itself and all possible permutations whereas the elements of the cartesian product are different if the order in which they are written varies.

R. Moreno Díaz et al. (Eds.): EUROCAST 2005, LNCS 3643, pp. 21–26, 2005.

Recently, a non binary algebra that allows the treatment of multiple valued data tables with systematic algebraic techniques has been introduced [1]. The basic elements of this algebra are the arrays and co-arrays. An array is a description of those subuniverses (subsets of g-spec-sets) that can be written as a cross product. A co-array is a description of those subuniverses (subsets of g-spec-sets) whose complement (respect to the universe) can be written as a cross product.

Definition 1. *Given sets G, ..., B, A, let $G_i \subseteq G$, ..., $B_i \subseteq B$, $A_i \subseteq A$, an array $|t_i| = |g_i, \ldots, b_i, a_i|$ is the symbolic representation of the cross product $G_i \otimes \ldots \otimes B_i \otimes A_i$ where $g_i \leftarrow\!\!\!\shortmid\; G_i$, ..., $b_i \leftarrow\!\!\!\shortmid\; B_i$ and $a_i \leftarrow\!\!\!\shortmid\; A_i$.*

$$|t_i| = |g_i, \ldots, b_i, a_i| \leftarrow\!\!\!\shortmid\; G_i \otimes \cdots \otimes B_i \otimes A_i$$

Definition 2. *Given sets G, ..., B, A, let $G_p \subseteq G$, ..., $B_p \subseteq B$, $A_p \subseteq A$, the symbolic representation of the complement (in the universe) of the cross product of subsets $\hat{G}_p \otimes \ldots \otimes \hat{B}_p \otimes \hat{A}_p$ where $g_p \leftarrow\!\!\!\shortmid\; G_p$, ..., $b_p \leftarrow\!\!\!\shortmid\; B_p$ and $a_p \leftarrow\!\!\!\shortmid\; A_p$ is called a co-array.*

$$||t_p|| = ||g_p, \ldots, b_p, a_p|| \leftarrow\!\!\!\shortmid\; \sim (\hat{G}_p \otimes \cdots \otimes \hat{B}_p \otimes \hat{A}_p)$$

Arrays and co-arrays are symbolic representations of subuniverses, 2-dimensional (two attributes) arrays and co-arrays can be represented graphically as shown in Fig. 1.

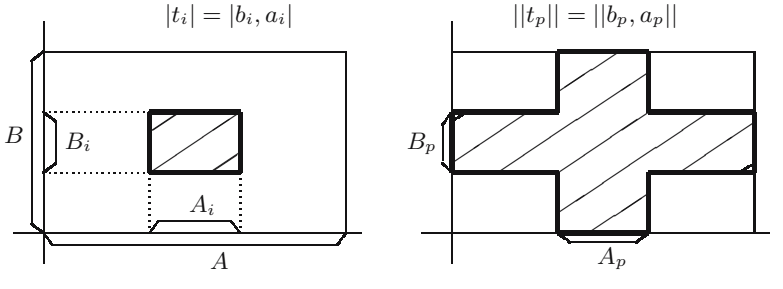

Fig. 1. Arrays and co-arrays in 2 dimensions

2 Degenerate Arrays

This array algebra has some special elements which are arrays and co-arrays at the same time.

Definition 3. *Given sets G, ..., B, A, if $|t_i| = \|t_i\|$ then $|t_i|$ is called a degenerate array or a degenerate co-array.*

The degenerate arrays were introduced in [2]. There are three types of degenerate arrays:

– The identity array $\bigwedge$, which describes the universe:

$$\bigwedge = |\wedge_g, \ldots, \wedge_b, \wedge_a| = ||\wedge_g, \ldots, \wedge_b, \wedge_a|| \looparrowright U = G \otimes \cdots \otimes B \otimes A$$

– The zero array $\bigvee$, describing the empty universe:

$$\bigvee = |\vee_g, \ldots, \vee_b, \vee_a| = ||\vee_g, \ldots, \vee_b, \vee_a|| \looparrowright \emptyset_U = \emptyset_G \otimes \cdots \otimes \emptyset_B \otimes \emptyset_A$$

– Arrays of the form:

$$|t_i| = |\wedge_g, \ldots, b_i, \wedge_a| = ||\vee_g, \ldots, b_i, \vee_a||$$

The 2-dimensional degenerate arrays can be represented graphically as can be seen in Fig. 2. The identity array describes the universe, whereas the zero array describes the empty universe. The third type of degenerate arrays describes subuniverses with only one distinguising attribute.

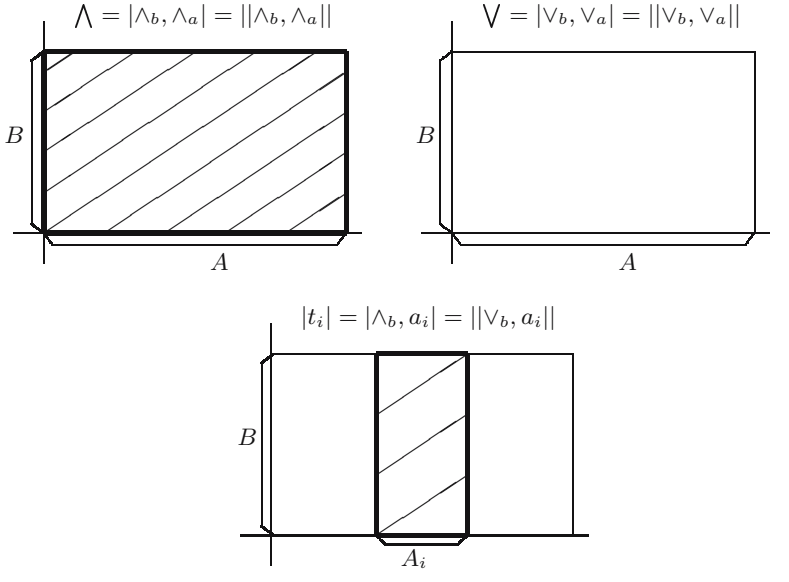

Fig. 2. Degenerate arrays in 2 dimensions

3 Array Projections and Co-array Projections

Even though the cross product is not the cartesian product it inherits some of its properties. It is well known that the cartesian product of any set by the empty set is the empty set. In array algebra this can be stated as follows: any array with a $\vee$ component is equal to $\bigvee$. The dual statement maintains that any co-array with a $\wedge$ component is equal to $\bigwedge$. These statements have been proved and used throughout the development of the array algebra, however they give rise to some interesting questions.

1. Arrays with a $\vee$ component are equal to $\bigvee$, in other words, just because there is a missing piece of information, we have no information at all. Furthermore,

$$|g_i, \ldots, b_i, \vee_a| = |g_i, \ldots, \vee_b, a_i| = \cdots = |\vee_g, \ldots, b_i, a_i|$$

2. Co-arrays with a $\wedge$ component are equal to $\bigwedge$, that is, just because all values of some attributes appear, we have complete information. Furthermore,

$$||g_i, \ldots, b_i, \wedge_a|| = ||g_i, \ldots, \wedge_b, a_i|| = \cdots = ||\wedge_g, \ldots, b_i, a_i||$$

The first question was studied in [3] and the array projections were introduced.

Definition 4. *Given an array $|t_i| = |g_i, \ldots, b_i, a_i|$, a first order array projection, $|P^1|$, is an array with one $\vee$ component and $(g-1)$ non-zero components, a second order array projection, $|P^2|$, is an array with two $\vee$ components and $(g-2)$ non-zero components, a nth order array projection ($n < g$), $|P^n|$, is an array with n $\vee$ components and $(g-n)$ non-zero components.*

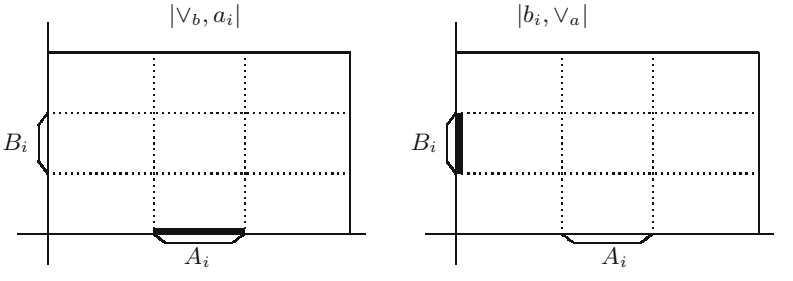

Fig. 3. 2-dimensional array projections

The array projections are descriptions of a reality with missing values for some of the attributes. An nth order array projection, $|P^n|$, is a description with no attribute values for n of the g attributes. The array projections are descriptions of incomplete data tables, some attributes do not take any of the attribute values.

Given a 2-dimensional array $|t_i| = |b_i, a_i|$, the first order array projections describe the following:

$$|P^1_a| = |b_i, \vee_a| \nrightarrow B_i \otimes \emptyset_A$$
$$|P^1_b| = |\vee_b, a_i| \nrightarrow \emptyset_B \otimes A_i$$

If two dimensions (attributes) are considered, an array $|b_i, a_i|$ can be graphically represented by a rectangle $b_i \times a_i$. When one of the sides becomes zero, then the rectangle has zero area. In this sense $|b_i, \vee_a| = |\vee_b, a_i|$. But even though one of the sides is zero, the other is not. The array $|b_i, \vee_a|$ becomes a line of size b_i, whereas the array $|\vee_b, a_i|$ becomes a line of size a_i. Therefore there is a difference. These lines are the array projections.

These array projections are shown on Fig. 3.

Let's address the second question, by studying co-arrays with a $\wedge$ component.

Definition 5. *Given a co-array* $||t_p|| = ||g_p, \ldots, b_p, a_p||$*, a first order co-array projection,* $||P^1||$*, is a co-array with one* $\wedge$ *component and* $(g-1)$ *non-identity components, a second order co-array projection,* $||P^2||$*, is a co-array with two* $\wedge$ *components and* $(g-2)$ *non-identity components, a nth order co-array projection* $(n < g)$*,* $||P^n||$*, is a co-array with* n $\wedge$ *components and* $(g-n)$ *non-identity components.*

The co-array projections are descriptions of a reality with non distinguishing values. An nth order co-array projection, $||P^n||$, is a description with all attribute values for n of the g attributes. The co-array projections are descriptions of ambiguous data tables, where some of the attributes take all possible attribute values.

Given a 2-dimensional co-array $||t_p|| = ||b_p, a_p||$, the first order co-array projections describe the following:

$$||P^1_a|| = ||b_i, \wedge_a|| \looparrowright\sim (\hat{B}_i \otimes \emptyset_A)$$

$$||P^1_b|| = ||\wedge_b, a_i|| \looparrowright\sim (\emptyset_B \otimes \hat{A}_i$$

If two dimensions (attributes) are considered, a co-array $||b_p, a_p||$ can be graphically represented, as shown in Fig. 1. Its first order co-array projections are: $||P^1_a|| = ||b_p, \wedge_a||$ and $||P^1_b|| = ||\wedge_b, a_p||$. Even though one of the sides is the identity, the other is not. Therefore, the co-array projections are not completely the identity, there is a line missing, as is shown in Fig. 4. This line corresponds to a first order array projection.

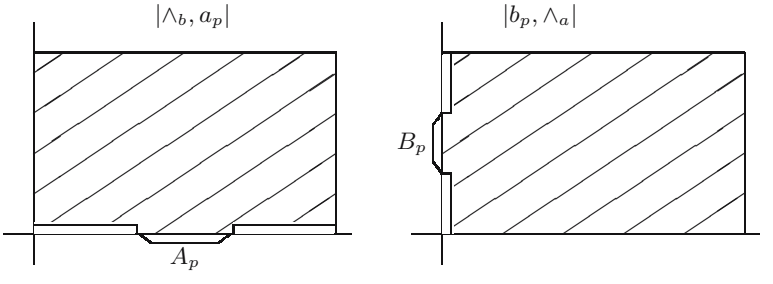

Fig. 4. 2-dimensional co-array projections

The number of nth order co-array projections can be easily found by counting the number of ways n $\wedge$ components can be placed in a co-array $||t_p|| = ||g_p, \ldots, b_p, a_p||$. Depending on the location of the $\wedge$ components, there are $\frac{g!}{n!(g-n)!}$ nth order co-array projections.

4 Conclusion

The multivalued algebra does not handle raw data, it handles declarative descriptions of tha data. The knowledge contained in a data table can be obtained using arrays and co-arrays.

Array projactions and co-array projections allow us to describe uncertain data tables, that is, those data tables that are incomplete (missing attribute values) and those that are ambiguous (non-distinguishing attributes).

On occasions data tables are incomplete, that is, several entries are empty. Data tables with no attribute values can be described by array projections. the order of the array projection is the number of missing attribute values.

Data tables can also be ambiguous, that is, some attributes are non-distinguishing (all attribute values apply). These data tables can be described by co-array projections. The order of the co-array projection is the number of non distinguishing attributes.

The array projections and co-array projections presented in this paper can be seen as a valid strategy for handling uncertain data tables.

Future work will deal with inductive learning [4], and the inclusion of the array projection in the learning process. Furthermore, the fact that $\bigvee$ and $\bigwedge$ are degenerate arrays originates the need to further investigate the third type of degenerate arrays and to try to forsee the relationship between the three types of degenerate arrays, their projections and uncertainty.

Acknowledgements

This work has been partially supported by the Dirección General de Investigación del Ministerio de Educación, Ciencia y Tecnología through the TIN2004-07926 project.

References

1. Miró-Julià M., Fiol-Roig G.: An Algebra for the Treatment of Multivalued Information Systems. Lecture Notes in Computer Science, **2652** (2003) 556–563.
2. Miró-Julià M.: A Contribution to Multivalued Systems. PhD thesis, Universitat de les Illes Balears, 2000.
3. Miró-Julià M.: The Zero Array: A Twilight Zone. Lecture Notes in Computer Science, **2809** (2003) 92–103.
4. Fiol G.: Inductive Learning from Incompletely Specified Examples. Frontiers in Artificial Intelligence and Applications, **100** (2003) 286–295.