

Cube Lattices: a Framework for Multidimensional Data Mining

Alain CASALI*

Rosine CICHETTI†

Lotfi LAKHAL‡

Abstract

Constrained multidimensional patterns differ from the well-known frequent patterns from a conceptual and logical points of view because they are provided with a common structure and support various types of constraints. Classical data mining techniques are based on the power set lattice of binary attributes and, even extended, are not suitable when addressing the discovery of constrained multidimensional patterns. In this paper we propose a foundation for various multidimensional data mining problems by introducing a new algebraic structure called cube lattice which characterizes the search space to be explored. We take into consideration monotone and/or antimonotone constraints enforced when mining multidimensional patterns. In addition, we propose condensed representations of the constrained cube lattice which is a convex space. Finally, we place emphasis on advantages of the cube lattice when compared to the power set lattice of binary attributes used for multidimensional data mining.

Keywords: Levelwise algorithms, Multidimensional data-mining, Lattices, Inclusion-Exclusion identities.

1 Introduction

Discovering constrained multidimensional patterns is a very important issue for multidimensional data mining because it makes it possible to solve various problems such as mining multidimensional association rules, multidimensional constrained gradients, classification rules and correlation rules. It is also the fundamental step when computing full, iceberg or range cubes materialized for OLAP [Han].

Adapting to this new multidimensional context, approaches and algorithms successfully used when mining binary databases is possible but not really relevant. However such adaptations have been frequently proposed for the extraction of quantitative association rules [SA96] and for classification [Han]. We believe that a precise semantics is required for solving multidimensional data mining problems such

as computing datacubes, version spaces, quotient cubes (succinct summaries of datacubes) and template multidimensional associations. Such a semantics can be captured through an algebraic structure provided with a similar expression power than the power set lattice when it is used for binary database mining.

In this paper, we introduce and characterize the search space to be considered for multidimensional data mining problems. Such a search space only encompasses semantically valid solutions. By introducing an order relation between elements of this space, and proposing two construction operators, we define a new algebraic structure, called cube lattice. The second aspect of our contribution concerns condensed representations of constrained cube lattices. Condensed representations based on boundary sets (or borders) avoid enumerating all the solutions [Mit97, MT97]. According to our knowledge, it does not exist an algebraic approach for extracting various kinds of multidimensional patterns. Therefore we propose a comparison between our approach and extensions to the multidimensional context of approaches mining binary databases. We show in particular: (i) the relevance of our search and solution spaces when compared to the ones considered by the quoted extensions, and (ii) the preservation of levelwise algorithm complexity in our approach and its increase for the considered extensions.

Organization of the article:

In section 2, we detail the structure of the cube lattice. In section 3, we study its condensed representations for the various cases of constraint conjunctions. We propose a comparison between the cube lattice and the power set lattice in section 4. As a conclusion, we underline the advantages of our proposal and evoke further work.

2 Cube lattice framework

Throughout the paper, we make the following assumptions and use the introduced notations. Let r be a relation over the schema \mathcal{R} . Attributes of \mathcal{R} are divided in two sets (i) \mathcal{D} the set of dimensions, also called categorical or nominal attributes, which correspond to analysis criteria for OLAP [Han], classification or concept learning [Mit97] and (ii) \mathcal{M} the set of measures

*LIF - Université de la Méditerranée

†LIF - Université de la Méditerranée

‡LIF - Université de la Méditerranée

(for OLAP) or class attributes. Moreover, attributes of \mathcal{D} are totally ordered (the underlying order is denoted by \langle_a) and $\forall A \in \mathcal{D}$, $\text{Dim}(A)$ stands for the projection of r over A .

The multidimensional space of the categorical database relation r groups all the valid combinations built up by considering the value sets of attributes in \mathcal{D} , which are enriched with the symbolic value ALL . The latter is a generalization of all the possible values of any dimension. Thus $\forall A \in \mathcal{D}, \forall a \in \text{Dim}(A), \{a\} \subset ALL$.

DEFINITION 2.1. [Multidimensional Space] *The multidimensional space of r is noted and defined as follows: $\text{Space}(r) = \{\times_{A \in \mathcal{D}}(\text{Dim}(A) \cup ALL)\} \cup \{\langle \emptyset, \dots, \emptyset \rangle\}$ where \times symbolizes the Cartesian product, and $\langle \emptyset, \dots, \emptyset \rangle$ stands for the combination of empty values.*

Any combination belonging to the multidimensional space is a tuple and represents a multidimensional pattern.

Example. Table 1 presents the categorical database relation used all along the paper to illustrate the introduced concepts. In this relation, A, B, C are dimensions and M is a measure.

RowId	A	B	C	M
1	a_1	b_1	c_1	3
2	a_1	b_1	c_2	2
3	a_1	b_2	c_1	2
4	a_1	b_2	c_2	2
5	a_2	b_1	c_1	1
6	a_3	b_1	c_1	1

Table 1: Relation example r

The following tuples are elements of $\text{Space}(r)$: $t_1 = \langle a_1, b_1, ALL \rangle$, $t_2 = \langle a_1, b_1, c_1 \rangle$, $t_3 = \langle a_1, b_2, c_1 \rangle$, $t_4 = \langle a_1, ALL, c_1 \rangle$ and $t_5 = \langle ALL, b_1, ALL \rangle$.

2.1 Generalization/Specialization order

The multidimensional space of r is structured by the generalization/specialization order between tuples. This order is originally introduced by T. Mitchell [Mit97] in the context of machine learning. It has the very same semantics as the cube rollup/drilldown [Han].

DEFINITION 2.2. [Generalization/Specialization] *Let u, v be two tuples of the multidimensional space of r :*

$$u \geq_g v \Leftrightarrow \begin{cases} \forall A \in \mathcal{D}, v[A] \subseteq u[A] \\ \text{or } v = \langle \emptyset, \dots, \emptyset \rangle \end{cases}$$

If $u[A]$ and $v[A] \neq ALL$ then $u[A]$ and $v[A]$ correspond to singletons just encompassing a value of $\text{Dim}(A)$. If $u \geq_g v$, we say that u is more general than v in $\text{Space}(r)$.

Example. In the multidimensional space of our relation example (Cf. Table 1), we have: $t_5 \geq_g t_2$, i.e. t_5 is more general than t_2 and t_2 is more specific than t_5 . Moreover any tuple generalizes the tuple $\langle \emptyset, \dots, \emptyset \rangle$ (or \top) and specializes the tuple $\langle ALL, \dots, ALL \rangle$ (or \perp).

When applied to a set of tuples, the operators \min and \max yield the tuples which are the most general ones in the set or the most specific ones respectively.

DEFINITION 2.3. [min/max w.r.t. \geq_g] *Let $T \subseteq \text{Space}(r)$ be a set of tuples:*

- $\min_{\geq_g}(T) = \{t \in T \mid \nexists u \in T : u \geq_g t\}$.
- $\max_{\geq_g}(T) = \{t \in T \mid \nexists u \in T : t \geq_g u\}$.

2.2 Basis operators

The two basic operators provided for tuple construction are: Sum (denoted by $+$) and Product (noted \bullet). The Semi-Product operator (noted \odot) is a constrained product.

The Sum of two tuples yields the most specific tuple which generalizes the two operands. It is defined as follows.

DEFINITION 2.4. [Sum Operator] *Let u and v be two tuples in $\text{Space}(r)$.*

$$t = u + v \Leftrightarrow \forall A \in \mathcal{D}, t[A] = \begin{cases} u[A] \text{ if } u[A] = v[A] \\ ALL \text{ elsewhere.} \end{cases}$$

We say that t is the Sum of the tuples u and v .

Example. In $\text{Space}(r)$, we have $t_2 + t_3 = t_4$. This means that t_4 is built up from the tuples t_2 and t_3 .

The Product of two tuples yields the most general tuple which specializes the two operands. If it exists, for these two tuples, a dimension A having distinct and real values (i.e. existing in the original relation), then the only tuple specializing them is the tuple $\langle \emptyset, \dots, \emptyset \rangle$ (apart from it, the tuple sets which can be used to construct them are disjoint).

DEFINITION 2.5. [Product Operator] *Let u and v be two tuples in $\text{Space}(r)$. We define the tuple z as follows: $\forall A \in \mathcal{D}, z[A] = u[A] \cap v[A]$. Then,*

$$t = u \bullet v \Leftrightarrow \begin{cases} t = z \text{ if } \nexists A \in \mathcal{D} \mid z[A] = \{\emptyset\} \\ \langle \emptyset, \dots, \emptyset \rangle \text{ elsewhere.} \end{cases}$$

We say that t is the Product of the tuples u and v .

Example. In $\text{Space}(r)$, we have $t_1 \bullet t_4 = t_2$. This means that t_1 and t_4 generalize t_2 and t_2 participates to the construction of t_1 and t_4 (directly or not). The tuples t_1 and t_3 have no common point apart from the tuple of empty values.

The Semi-Product operator is a constrained product operator useful for candidate generation in a level-wise approach [AMS⁺96, MT97].

DEFINITION 2.6. [Semi-Product Operator] Let u and v be two tuples in $\text{Space}(r)$, $X = \{A \in \mathcal{D} \mid u[A] \neq ALL\}$ and $Y = \{A \in \mathcal{D} \mid v[A] \neq ALL\}$.

$$t = u \odot v \Leftrightarrow \begin{cases} t = u \bullet v & \text{if } X \setminus \max_{<_d}(X) = Y \setminus \max_{<_d}(Y) \\ & \text{and } \max_{<_d}(X) <_d \max_{<_d}(Y) \\ <\emptyset, \dots, \emptyset> & \text{elsewhere.} \end{cases}$$

where $<_d$ is a total order over \mathcal{D} .

Example. The lexicographical order is chosen for ordering attributes of \mathcal{D} . In $\text{Space}(r)$, we have $t_1 \odot t_4 = t_2$ and $t_5 \odot t_2 = <\emptyset, \dots, \emptyset>$.

By providing the multidimensional space of r with the generalization order between tuples and using the above-defined operators Sum and Product, we define an algebraic structure which is called cube lattice. Such a structure provides a sound and foundation for multidimensional data mining issues.

THEOREM 2.1. Let r be a categorical database relation over $\mathcal{D} \cup \mathcal{M}$. The ordered set $\text{CL}(r) = \langle \text{Space}(r), \geq_g \rangle$ is a complete, graded, atomistic and coatomistic lattice, called cube lattice in which Meet (\wedge) and Join (\vee) elements are given by:

1. $\forall T \subseteq \text{CL}(r), \bigwedge T = +_{t \in T} t$
2. $\forall T \subseteq \text{CL}(r), \bigvee T = \bullet_{t \in T} t$

Example. Figure 1 exemplifies the cube lattice of the projection of our relation example (Cf. Table 1) over the attributes A and B . In this diagram, the edges represent the generalization or specialization links between tuples.

PROPOSITION 2.1. Let $\mathcal{L}(r)$ be the power set lattice of binary attributes, i.e. the lattice $\langle \mathcal{P}(\cup_{A \in \mathcal{D}} \text{Dim}(A)), \subseteq \rangle^1$. Then it exists an embedded order: $\Phi : \text{CL}(r) \rightarrow \mathcal{L}(r)$

$$t \mapsto \begin{cases} \cup_{A \in \mathcal{D}} \text{Dim}(A) & \text{if } t = <\emptyset, \dots, \emptyset> \\ \{t[A] \mid \forall A \in \mathcal{D}, t[A] \neq ALL\} & \text{elsewhere.} \end{cases}$$

¹ $\mathcal{P}(X)$ is the power set of X .

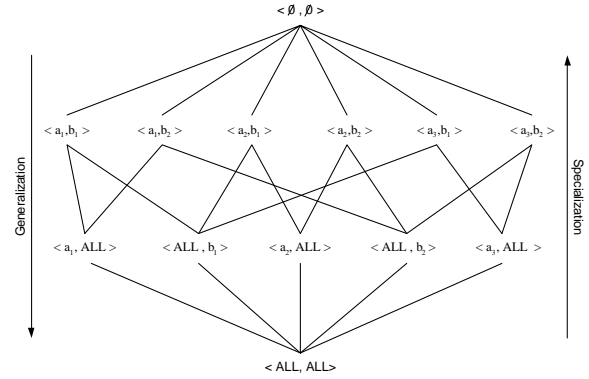


Figure 1: Hasse diagram of the cube lattice for the projection of r over AB

The rank of a tuple t is the length of the minimal path linking the tuple $<ALL, \dots, ALL>$ and the tuple t .

$$\text{Thus we have: } \text{rank}(t) = \begin{cases} |\Phi(t)| & \text{if } t \neq <\emptyset, \dots, \emptyset> \\ |\mathcal{D}| + 1 & \text{elsewhere.} \end{cases}$$

PROPOSITION 2.2. The height (level number) of the cube lattice is $|\mathcal{D}| + 1$. The element number by level i ($i \in 1..|\mathcal{D}|$) is:

$$\sum_{\substack{X \subseteq \mathcal{D} \\ |X|=i}} \left(\prod_{A \in X} |\text{Dim}(A)| \right) \leq \binom{|\mathcal{D}|}{i} \max_{A \in \mathcal{D}} (|\text{Dim}(A)|)^i.$$

The total number of elements in the cube lattice is:

$$\left(\prod_{A \in |\mathcal{D}|} (|\text{Dim}(A)| + 1) \right) + 1$$

3 Condensed representations of constrained cube lattices

The cube lattice defines a graded search space for various multidimensional data mining problems. In this section, we study the structure of the cube lattice in presence of constraint conjunctions. Provided with such a structure, we propose condensed representations (or borders) of the constrained cube lattice with a twofold objective: defining in a compact way the solution space and deciding whether a tuple t belongs to the solution space or not. Finally, following from principles of level-wise approaches, we give an algorithm for computing condensed representations of constrained cube lattices. We take into account monotone and antimonotone constraints frequently used in binary data mining [Han].

We recall the definitions of monotone and antimonotone constraints w.r.t \geq_g .

DEFINITION 3.1. [*Constraint*]

1. A constraint $Const$ is monotone if and only if:
 $\forall t, u \in CL(r) : [t \geq_g u \text{ and } Const(t)] \Rightarrow Const(u)$.
2. A constraint $Const$ is antimonotone if and only if:
 $\forall t, u \in CL(r) : [t \geq_g u \text{ and } Const(u)] \Rightarrow Const(t)$.

3.1 Structure of the constrained cube lattice

The cube lattice faced with monotone and/or antimonotone constraints does not necessarily remain a lattice. We show in this section that such a partially ordered set is provided with a mathematical structure which is a convex space [Vel93] and thus it is boundary representable.

Notations: We note cmc ($camc$ respectively) a conjunction of monotone constraints (antimonotone respectively) and chc an hybrid conjunction of constraints (monotone and antimonotone).

Remarks (extreme cases):

- We suppose that the tuple $\langle ALL, \dots, ALL \rangle$ always satisfies the conjunction of antimonotone constraints and the tuple $\langle \emptyset, \dots, \emptyset \rangle$ always verifies the conjunction of monotone constraints. Under these assumptions, the solution space encompasses at least one element (possibly the tuple of empty values).
- Moreover, we assume that the tuple $\langle ALL, \dots, ALL \rangle$ never verifies the conjunction of monotone constraints and the tuple $\langle \emptyset, \dots, \emptyset \rangle$ never satisfies the conjunction of antimonotone constraints, because without making these assumptions, the solution space is $CL(r)$.

DEFINITION 3.2. [*Convex Space*] Let $\langle P, \preceq \rangle$ be a partially ordered set, $C \subseteq P$ is a convex space if $\forall x, y, z \in P, x \preceq y \preceq z$ and $x, z \in C \Rightarrow y \in C$. Thus C is bounded by two sets: an upper bound or “upper set” defined by $\max_{\preceq}(C)$ and a lower bound or “lower set” defined by $\min_{\preceq}(C)$.

THEOREM 3.1. The constrained cube lattice $CL(r)_{const}$ is a convex space. Its upper set S_{const}^+ and lower set G_{const}^+ are:

1. if $const = cmc$, $G_{cmc}^+ = \min_{\geq_g}(\{t \in CL(r) \mid cmc(t)\})$ and $S_{cmc}^+ = \langle \emptyset, \dots, \emptyset \rangle$.
2. if $const = camc$, $G_{camc}^+ = \langle ALL, \dots, ALL \rangle$ and $S_{camc}^+ = \max_{\geq_g}(\{t \in CL(r) \mid camc(t)\})$.

3. if $const = chc$, $G_{chc}^+ = \min_{\geq_g}(\{t \in CL(r) \mid chc(t)\})$ and $S_{chc}^+ = \max_{\geq_g}(\{t \in CL(r) \mid chc(t)\})$.

The upper bound S_{const}^+ represents the most specific tuples satisfying the constraint conjunction and the lower bound G_{const}^+ is the set of the most general tuples satisfying the constraint conjunction. Thus S_{const}^+ and G_{const}^+ are condensed representations of the constrained cube lattice with conjunction of monotone and/or antimonotone constraints.

COROLLARY 3.1. 1. Given S_{camc}^+ and cmc , the condensed representation of $CL(r)_{cch}$ is:

$$G_{chc}^+ = \min_{\geq_g}(\{t \in CL(r) \mid \exists t' \in S_{camc}^+ : t \geq_g t' \text{ and } cmc(t)\}) \text{ and } S_{chc}^+ = \{t \in S_{camc}^+ \mid \exists t' \in G_{chc}^+ : t' \geq_g t\}.$$

2. Given G_{cmc}^+ and $camc$, the condensed representation of $CL(r)_{cch}$ is:

$$S_{chc}^+ = \max_{\geq_g}(\{t \in CL(r) \mid \exists t' \in G_{cmc}^+ : t' \geq_g t \text{ and } camc(t)\}) \text{ and } G_{chc}^+ = \{t \in G_{camc}^+ \mid \exists t' \in S_{chc}^+ : t \geq_g t'\}.$$

The algorithm GLA, a generalized levelwise construction of the borders S^+, G^+ representing the cube lattice with conjunctions of monotone and/or antimonotone constraints is given in [CCL02].

4 Cube lattices versus power set lattices in multidimensional data mining

We propose in this section a comparative analysis between binary data mining approaches used in a multidimensional context and ours by studying search spaces to be traversed, solution spaces, and the behavior of levelwise algorithms. By considering the power set lattice $\mathcal{L}(r)$ and the cube lattice $CL(r)$ as search spaces for the discovery of constrained multidimensional patterns, our comparison focuses on three points: lattice and level sizes, the correctness of obtained solutions faced with constraint conjunctions, and the levelwise algorithm complexity.

• **lattice and level sizes:**

Let us examine the sizes of the compared lattices and their largest level. The following contributes to a more precise analysis of the computational complexity of multidimensional data mining algorithms.

$|\mathcal{L}(r)| = 2^{\sum_{A \in \mathcal{D}} |\text{Dim}(A)|}$, whereas $|CL(r)| = \prod_{A \in \mathcal{D}} (|\text{Dim}(A)| + 1) + 1$ (Cf. proposition 2.2). An upper bound for the cube lattice cardinality is $\mathcal{O}((\max_{A \in \mathcal{D}} (|\text{Dim}(A)| + 1))^{|\mathcal{D}|})$. Let us consider for instance a relation having 5 attributes,

each of which having 10 possible values, we have $|\mathcal{L}(r)| = 2^{50}$, whereas $|\text{CL}(r)| = 11^5 + 1$.

We set $n = \sum_{A \in \mathcal{D}} |\text{Dim}(A)|$. The size of the largest level in $\mathcal{L}(r)$ is bounded by $\binom{n}{n/2}$, which is asymptotic to $\frac{2^n}{\sqrt{n}} \sqrt{\frac{2}{\pi}}$ whereas the maximal size of levels in the cube lattice is bounded by $\binom{|\mathcal{D}|}{|\mathcal{D}|/2} \max_{A \in \mathcal{D}} (|\text{Dim}(A)|)^{|\mathcal{D}|}$ which is asymptotic to $\frac{2^{|\mathcal{D}|}}{\sqrt{|\mathcal{D}|}} \sqrt{\frac{2}{\pi}} * \max_{A \in \mathcal{D}} (|\text{Dim}(A)|)^{|\mathcal{D}|}$.

Thus the size of the largest level in $\mathcal{L}(r)$ is exponential in the value number of dimensional attributes of the relation (i.e. $\sum_{A \in \mathcal{D}} |\text{Dim}(A)|$). On the other hand, the size of $\text{CL}(r)$ is exponential in the number of attributes (i.e. $|\mathcal{D}|$).

- **Solution correctness:**

The power set lattice $\mathcal{L}(r)$ encompasses solutions semantically erroneous whereas the cube lattice is exactly the valid search space. More precisely, the embedded order Φ (Cf. proposition 2.1) shows that for any tuple in the cube lattice it exists an equivalent combination in the power set lattice which is semantically valid whereas the converse equivalence does not hold because Φ is not bijective. $\forall t \in \text{CL}(r), \nexists \alpha_i, \alpha_j \in \Phi(t)$ and $\alpha_i, \alpha_j \in \text{Dim}(A_k)$ according to definition 2.1. On the other hand $\forall A_k \in \mathcal{D}, \forall \alpha_i, \alpha_j \in \text{Dim}(A_k)$ with $i \neq j, (\alpha_i, \alpha_j) \in \mathcal{L}(r)$, nevertheless such combinations are proved to be erroneous because multidimensional patterns cannot encompass two values of a single attribute.

- **Levelwise algorithm complexity:**

The generation of erroneous patterns obviously alter performances of underlying algorithms. We study such an alteration through the comparison of size of borders relevant for monotone ($\text{Freq}() \leq \text{threshold}$) and antimonotone ($\text{Freq}() \geq \text{threshold}$) constraints. Let us consider the most general solutions satisfying cmc for $\mathcal{L}(r)$ and $\text{CL}(r)$. We have $|\mathbb{G}_{\text{cmc}}^+(\mathcal{L}(r)_{\text{cmc}})| \geq |\mathbb{G}_{\text{cmc}}^+(\text{CL}(r)_{\text{cmc}})| + \sum_{A \in \mathcal{D}} \frac{|\text{Dim}(A)|^2 - |\text{Dim}(A)|}{2}$. For antimonotone constraints, the negative border for $\mathcal{L}(r)$ also encompasses erroneous patterns (couples of values of a very same attribute), its size is greater than the size of $\mathbb{G}_{\text{camc}}^-$ for $\text{CL}(r)$. In fact, the number of additional elements in the border $\mathbb{G}_{\text{camc}}^-$ for $\mathcal{L}(r)$ is exactly the maximal number previously given (the very same couples are to be considered), i.e. $|\mathbb{G}_{\text{camc}}^-(\mathcal{L}(r)_{\text{camc}})| \geq |\mathbb{G}_{\text{camc}}^-(\text{CL}(r)_{\text{camc}})| + \sum_{A \in \mathcal{D}} \frac{|\text{Dim}(A)|^2 - |\text{Dim}(A)|}{2}$. The larger the attribute value sets are, the worse are the conse-

quences of the negative border size. This is the reason behind the inefficiency, in a multidimensional context, of levelwise algorithms over $\mathcal{L}(r)$.

Conclusion

In this paper, we introduce a formal framework for solving various problems of multidimensional data mining. We propose a novel algebraic structure, the cube lattice as a graded search space. We also derived condensed representations of the cube lattice faced with monotone and/or antimonotone constraints. Such a result is based on the particular structure of constrained cube lattice which is a convex space.

A work in progress addresses the definition of a closure operator on the cube lattice. Our aim is to show that the set of closed tuples provided with the generalization order is a coatomic lattice which seems to be isomorphic to the Galois (concept) lattice of the binary relation representing the original database relation.

References

- [AMS⁺96] Rakesh Agrawal, Heikki Mannila, Ramakrishnan Srikant, Hannu Toivonen, and A. Inkeri Verkamo. Fast Discovery of Association Rules. In *Advances in Knowledge Discovery and Data Mining*, pages 307–328, 1996.
- [CCL02] Alain Casali, Rosine Cicchetti, and Lotfi Lakhil. Treillis Relationnel: Une Structure Algébrique pour le Data Mining Multidimensionnel. In *Proceedings of the 18th French Conference on Advanced Databases, BDA*, pages 445–467, 2002.
- [Han] Jiawei Han. Data mining, data warehousing, and knowledge discovery in databases (dbminer). In <http://db.cs.sfu.ca/sections/publication/kdd/kdd.html>.
- [Mit97] Tom M. Mitchell. *Machine learning*. MacGraw-Hill Series in Computer Science, 1997.
- [MT97] Heikki Mannila and Hannu Toivonen. Levelwise Search and Borders of Theories in Knowledge Discovery. In *Data Mining and Knowledge Discovery*, volume 1(3), pages 241–258, 1997.
- [SA96] Ramakrishnan Srikant and Rakesh Agrawal. Mining quantitative association rules in large relational tables. In *Proceedings of the International Conference on Management of Data, SIGMOD*, pages 1–12, 1996.
- [Vel93] M. Van de Vel. *Theory of Convex Structures*. North-Holland, Amsterdam, 1993.