Constraint-Based Mining of Fault-Tolerant Patterns from Boolean Data

Jérémy Besson[1,2], Ruggero G. Pensa[1], Céline Robardet[3], and Jean-François Boulicaut[1]

[1] INSA Lyon, LIRIS CNRS UMR 5205, F-69621 Villeurbanne cedex, France
[2] UMR INRA/INSERM 1235, F-69372 Lyon cedex 08, France
[3] INSA Lyon, PRISMA, F-69621 Villeurbanne cedex, France
{Firstname.Name}@insa-lyon.fr

Abstract. Thanks to an important research effort during the last few years, inductive queries on local patterns (e.g., set patterns) and their associated complete solvers have been proved extremely useful to support knowledge discovery. The more we use such queries on real-life data, e.g., biological data, the more we are convinced that inductive queries should return fault-tolerant patterns. This is obviously the case when considering formal concept discovery from noisy datasets. Therefore, we study various extensions of this kind of bi-set towards fault-tolerance. We compare three declarative specifications of fault-tolerant bi-sets by means of a constraint-based mining approach. Our framework enables a better understanding of the needed trade-off between extraction feasibility, completeness, relevance, and ease of interpretation of these fault-tolerant patterns. An original empirical evaluation on both synthetic and real-life medical data is given. It enables a comparison of the various proposals and it motivates further directions of research.

1 Introduction

According to the inductive database approach, mining queries can be expressed declaratively in terms of constraints on the desired patterns or models [16, 10, 6]. Thanks to an important research effort the last few years, inductive queries on local patterns (e.g., set or sequential patterns) and complete solvers which can evaluate them on large datasets (Boolean or sequence databases) have been proved extremely useful. Properties of constraints have been studied in depth (e.g., monotonicity, succinctness, convertibility) and sophisticated pruning strategies enable to compute complete answer sets for many constraints (i.e., Boolean combination of primitive constraints) of practical interest. However, the more we use these techniques on intrinsically dirty and noisy real-life data, e.g., biological or medical data, the more we are convinced that inductive queries should return fault-tolerant patterns. One interesting direction of research is to introduce softness w.r.t. constraint satisfaction [1, 5]. We consider in this paper another direction leading to crispy constraints in which fault-tolerance is declaratively specified.

F. Bonchi and J.-F. Boulicaut (Eds.): KDID 2005, LNCS 3933, pp. 55–71, 2006.

Table 1. A Boolean context $\mathbf{r}_1$

	g_1	g_2	g_3	g_4	g_5	g_6	g_7
t_1	1	0	1	0	1	0	0
t_2	1	1	1	1	0	1	0
t_3	0	1	1	1	1	1	1
t_4	0	0	0	1	1	1	0
t_5	1	0	0	0	0	1	0
t_6	1	1	1	1	1	0	0
t_7	1	1	1	1	1	0	0

Our starting point is the fundamental limitation of formal concept (i.e., connected closed sets) discovery from noisy data. Formal concept analysis has been developed for more than two decades [24] as a way to extract knowledge from Boolean datasets. Informally, formal concepts are maximal bi-sets/rectangles of true values[1]. For instance, Table 1 is a toy example dataset $\mathbf{r}_1$ and the bi-set $(\{t_6, t_7\}, \{g_1, g_2, g_3, g_4, g_5\})$ is a formal concept in $\mathbf{r}_1$.

Some algorithms are dedicated to the computation of complete collections of formal concepts [17]. Since, by construction, formal concepts are built on closed sets, the extensive research on (frequent) closed set computation (see [15] for a survey) has obviously opened new application domains for formal concept discovery. When considering very large and/or dense Boolean matrices, constraint-based mining of formal concepts has been studied [23, 4]: every formal concept which furthermore satisfies some other user-defined constraints is computed. For example, we can extract formal concepts with minimal size constraints for both set components. Given our previous example, if we want formal concepts with at least 3 elements in each set, the formal concept $(\{t_3, t_6, t_7\}, \{g_2, g_3, g_4, g_5\})$ satisfies the constraint whereas $(\{t_6, t_7\}, \{g_1, g_2, g_3, g_4, g_5\})$ does not.

A formal concept associates a maximal set of objects to a maximal set of properties which are all in relation. Such an association is often too strong in real-world data. Even though the extraction might remain tractable, the needed post-processing and interpretation phases turn out to be tedious or even impossible. Indeed, in noisy data, not only the number of formal concepts explodes but also many of them are not relevant enough. It has motivated new directions of research where interesting bi-sets are considered as dense rectangles of true values [2, 14, 13, 3, 19].

In this paper, we consider a constraint-based mining approach for relevant fault-tolerant formal concept mining. We decided to look for an adequate formalization for three of our recent proposals (i.e., CBS [2], FBS [19], and DRBS [3]) which have been motivated by a declarative specification for fault-tolerance. We do not provide the algorithms which have been recently published for solving inductive queries on such patterns [2, 19, 3]. The contribution of this paper is to propose a simple framework to support a better understanding of the needed

[1] We might say combinatorial rectangles since it is up to arbitrary permutations of rows and columns in the Boolean matrix.

trade-off between extraction feasibility, completeness, relevance, and ease of interpretation of these various pattern types. This formalization enables to predict part of the behavior of the associated solvers and some formal properties can be established. An original empirical evaluation on both synthetic and real-life medical data is given. It enables to compare the pros and cons of each proposal. An outcome of these experiments is that fault-tolerant bi-set mining is possible. Used in conjunction with other user-defined constraints, it should support the dissemination of relevant local set pattern discovery techniques for intrinsically noisy data.

Section 2 provides the needed definitions. Section 3 presents a discussion on some important properties that fault-tolerant bi-set mining should satisfy. Section 4 provides not only experimental results on synthetic data when various levels of noise are added but also experiments on a real-life medical dataset. Section 5 is a short conclusion.

2 Pattern Domains

We now define the different classes of patterns to be studied in this paper. Assume a set of objects $\mathcal{O} = \{t_1, \ldots, t_m\}$ and a set of Boolean properties $\mathcal{P} = \{g_1, \ldots, g_n\}$. The Boolean context to be mined is $\mathbf{r} \subseteq \mathcal{O} \times \mathcal{P}$, where $r_{ij} = 1$ if property g_j is satisfied by object t_i, 0 otherwise. Formally, a bi-set is an element (X, Y) where $X \subseteq \mathcal{O}$ and $Y \subseteq \mathcal{P}$. $\mathcal{L} = 2^{\mathcal{O}} \times 2^{\mathcal{P}}$ denotes the search space for bi-sets. We say that a bi-set (X, Y) is included in a bi-set (X', Y') (denoted $(X, Y) \subseteq (X', Y')$) iff $(X \subseteq X' \wedge Y \subseteq Y')$.

Definition 1. *Let us denote by $\mathcal{Z}_l(x, Y)$ the number of false values of a row x on the columns in Y: $\mathcal{Z}_l(x, Y) = \sharp\{y \in Y | (x, y) \notin \mathbf{r}\}$ where $\sharp$ denotes the cardinality of a set. Similarly, $\mathcal{Z}_c(y, X) = \sharp\{x \in X | (x, y) \notin \mathbf{r}\}$ denotes the number of false values of a column y on the rows in X.*

Let us now give an original definition of formal concepts (see, e.g., [24] for a classical one). Sub-constraint 2.1 expresses that a formal concept contains only true values. Sub-constraint 2.2 denotes that formal concept relevancy is enhanced by a maximality property.

Definition 2 (FC). *A bi-set $(X, Y) \in \mathcal{L}$ is a formal concept in $\mathbf{r}$ iff*
(2.1) $\forall x \in X,\ \mathcal{Z}_l(x, Y) = 0 \ \wedge\ \forall y \in Y,\ \mathcal{Z}_c(y, X) = 0$
(2.2) $\forall x \in \mathcal{O} \setminus X,\ \mathcal{Z}_l(x, Y) \geq 1 \ \wedge\ \forall y \in \mathcal{P} \setminus Y,\ \mathcal{Z}_c(y, X) \geq 1$.

Example 1. *Given $\mathbf{r}_1$, we have $\mathcal{Z}_l(t_6, \{g_4, g_5, g_6\}) = 1$ and $\mathcal{Z}_c(g_5, \mathcal{O}) = 2$. $(\{t_3, t_4, t_6, t_7\}, \{g_4, g_5\})$ and $(\{t_3, t_4\}, \{g_4, g_5, g_6\})$ are* FC *patterns (see Table 2).*

Let us now define the so-called DRBS, CBS and FBS fault tolerant patterns.

Definition 3 (DRBS [3]). *Given integer parameters δ and ϵ, a bi-set $(X, Y) \in \mathcal{L}$ is called a* DRBS *pattern (Dense and Relevant Bi-Set) in $\mathbf{r}$ iff*

Table 2. A row permutation on $\mathbf{r}_1$ to illustrate Example 1

	g_1	g_2	g_3	g_4	g_5	g_6	g_7
t_3	0	1	1	1	1	1	1
t_4	0	0	0	1	1	1	0
t_6	1	1	1	1	1	0	0
t_7	1	1	1	1	1	0	0
t_1	1	0	1	0	1	0	0
t_2	1	1	1	1	0	1	0
t_5	1	0	0	0	0	1	0

(3.1) $\forall x \in X,\ \mathcal{Z}_l(x,Y) \leq \delta \ \wedge\ \forall y \in Y,\ \mathcal{Z}_c(y,X) \leq \delta$
(3.2) $\forall e \in \mathcal{O} \setminus X,\ \forall x \in X,\ \mathcal{Z}_l(e,Y) \geq \mathcal{Z}_l(x,Y) + \epsilon$
$\wedge\ \forall e' \in \mathcal{P} \setminus Y,\ \forall y \in Y,\ \mathcal{Z}_c(e',X) \geq \mathcal{Z}_c(y,X) + \epsilon$
(3.3) It is maximal, i.e., $\nexists (X',Y') \in \mathcal{L}$ *s.t.* (X',Y') *is a* DRBS *pattern and* $(X,Y) \subseteq (X',Y')$.

DRBS patterns have at most δ false values per row and per column (Sub-constraint 3.1) and are such that each outside row (resp. column) has at least ϵ false values plus the maximal number of false values on the inside rows (resp. columns) according to Sub-constraint 3.2. The size of a DRBS pattern increases with δ such that when $\delta > 0$, it happens that several bi-sets are included in each other. Only maximal bi-sets are kept (Sub-constraint 3.3). Notice that δ and ϵ can take different values on rows and on columns.

Property 1. *When* $\delta = 0$ *and* $\epsilon = 1$, DRBS $\equiv$ FC.

Example 2. *If* $\delta = \epsilon = 1$, $(X,Y) = (\{t_1,t_2,t_3,t_4,t_6,t_7\},\{g_3,g_4,g_5\})$ *is a* DRBS *pattern in* $\mathbf{r}_1$. *Columns* g_1, g_2, g_6 *and* g_7 *contain at least two false values on* X, *and* t_5 *contains three false values on* Y *(see Table 3).*

Table 3. A row permutation on $\mathbf{r}_1$ to illustrate Example 2

	g_1	g_2	g_3	g_4	g_5	g_6	g_7
t_1	1	0	1	0	1	0	0
t_2	1	1	1	1	0	1	0
t_3	0	1	1	1	1	1	1
t_4	0	0	0	1	1	1	0
t_6	1	1	1	1	1	0	0
t_7	1	1	1	1	1	0	0
t_5	1	0	0	0	0	1	0

The whole collection of DRBS can be computed (in rather small datasets) by using the correct and complete algorithm DR-Miner described in [3]. It is a generic algorithm for bi-set constraint-based mining which is an adaptation of Dual-Miner [9]. It is based on an enumeration strategy of bi-sets which enables efficient anti-monotonic and monotonic pruning (Sub-constraint 3.1 in conjunction with other user-defined constraints which have monotonicity properties), and partial pruning for Sub-constraint 3.2. Sub-constraint 3.3 is checked in a post-processing phase.

We now consider a preliminary approach for specifying symmetrical fault-tolerant formal concepts. Indeed, DRBS class has been designed afterwards.

Definition 4 (CBS [2]). *Given an integer parameter δ, a bi-set $(X, Y) \in \mathcal{L}$ is called a* CBS *pattern (Consistent Bi-Set) iff*

(4.1) $\forall x \in X,\ \mathcal{Z}_l(x, Y) \leq \delta \ \wedge\ \forall y \in Y,\ \mathcal{Z}_c(y, X) \leq \delta$

(4.2) No row (resp. column) outside (X, Y) is identical to a row (resp. column) inside (X, Y)

(4.3) It is maximal, i.e., $\nexists (X', Y') \in \mathcal{L}$ s.t. (X', Y') is a CBS *pattern and $(X, Y) \subseteq (X', Y')$.*

Notice that again, parameter δ can be chosen with different values on rows and on columns.

Example 3. *If $\delta = 1$, $(X, Y) = (\{t_1, t_2, t_3, t_6, t_7\}, \{g_1, g_3, g_5\})$ is a* CBS *pattern in $\mathbf{r}_1$. Columns g_6 and g_7 contain more than one false value on X, t_4 and t_5 contain more than one false value on Y. g_2 and g_4 contain only one false value, but as they are identical on X, either we add both or they are both excluded. As there are two false values on t_1, we do not add them (see Table 4).*

Table 4. A row and column permutation on $\mathbf{r}_1$ to illustrate Example 3

	g_1	g_3	g_5	g_2	g_4	g_6	g_7
t_1	1	1	1	0	0	0	0
t_2	1	1	0	1	1	1	0
t_3	0	1	1	1	1	1	1
t_6	1	1	1	1	1	0	0
t_7	1	1	1	1	1	0	0
t_4	0	0	1	0	1	1	0
t_5	1	0	0	0	0	1	0

Property 2. *When $\delta = 0$,* CBS $\equiv$ FC*. Furthermore, when $\epsilon = 1$, each* DRBS *pattern is included in one of the* CBS *patterns.*

In [2], the authors propose an algorithm for computing CBS patterns by merging formal concepts which have been extracted beforehand. The obtained bi-sets are then processed to keep only the maximal ones having less than δ false values per row and per column. This principle is however incomplete: every bi-set which satisfies the above constraints can not be extracted by this principle. In other terms, some CBS patterns can not be obtained as a merge between two formal concepts. CBS patterns might be extracted by a straightforward adaptation of the DR-Miner generic algorithm but the price to pay for completeness would be too expensive.

Let us finally consider another extension of formal concepts which is not symmetrical. It has been designed thanks to some previous work on one of the few approximate condensed representations of frequent sets, the so-called δ-free sets [7, 8]. δ-free sets are well-specified sets whose counted frequencies enable to infer the frequency of many sets (sets included in their so-called δ-closures) without further counting but with a bounded error. When $\delta = 0$, the 0-closure on a 0-free set X is the classical closure and it provides a closed set. The idea is to consider bi-sets built on δ-free sets with the intuition that it will provide strong associations between sets of rows and sets of columns. It has been introduced for the first time in [19] as a potentially interesting local pattern type for bi-cluster characterization.

Providing details on δ-freeness and δ-closures is beyond the objective of this paper (see [7, 8] for details). We just give here an intuitive definition of these notions. A set $Y \subseteq \mathcal{P}$ is δ-free for a positive integer δ if its absolute frequency in $\mathbf{r}$ differs from the frequency of all its strict subsets by at least $\delta + 1$. For instance, in $\mathbf{r}_1$, $\{g_2\}$ is a 1-free set. The δ-closure of a set $Y \subseteq \mathcal{P}$ is the superset Z of Y such that every added property ($\in Z \setminus Y$) is almost always true for the objects which satisfy the properties from Y: at most δ false values are enabled. For instance, the 1-closure of $\{g_2\}$ is $\{g_1, g_2, g_3, g_4, g_5\}$. It is possible to consider bi-sets which can be built on δ-free sets and their δ-closures on one hand, on the sets of objects which support the δ-free set on the properties on the other hand.

Definition 5 (FBS). *A bi-set* $(X, Y) \in \mathcal{L}$ *is a* FBS *pattern (Free-set based Bi-Set) iff* Y *can be decomposed into* $Y = K \cup C$ *such that* K *is a* δ*-free set in* $\mathbf{r}$*,* C *is its associated* δ*-closure and* $X = \{t \in \mathcal{O} \mid \forall k \in K,\ (t, k) \in \mathbf{r}\}$*. By construction,* $\forall y \in Y,\ \mathcal{Z}_c(y, X) \leq \delta$ *and* $\forall y \in K, \mathcal{Z}_c(y, X) = 0$*.*

Property 3. *When* $\delta = 0$*,* FBS $\equiv$ FC*.*

Example 4. *If* $\delta = 1$*,* $\{g_2\}$ *is a* δ*-free set and* $(\{t_2, t_3, t_6, t_7\}, \{g_1, g_2, g_3, g_4, g_5\})$ *is a* FBS *pattern in* $\mathbf{r}_1$*. Another one is* $(\{t_3, t_4\}, \{g_2, g_3, g_4, g_5, g_6, g_7\})$*. Notice that we get at most one false value per column but we have three false values on* t_4 *(see Table 5).*

Table 5. A row permutation on $\mathbf{r}_1$ to illustrate Example 4

	g_1	g_2	g_3	g_4	g_5	g_6	g_7
t_2	1	1	1	1	0	1	0
t_3	0	1	1	1	1	1	1
t_6	1	1	1	1	1	0	0
t_7	1	1	1	1	1	0	0
t_1	1	0	1	0	1	0	0
t_4	0	0	0	1	1	1	0
t_5	1	0	0	0	0	1	0

The extraction of FBS can be extremely efficient thanks to δ-freeness anti-monotonicity. The implementation described in [8] can be straightforwardly extended to output FBS patterns. Notice that FBS patterns are bi-sets with a bounded number of exception per column but every bi-set with a bounded number of exception per column is not necessarily a FBS pattern. An example of a bi-set with at most 1 false value per column which is not a FBS pattern in $\mathbf{r}_1$ is $(\{t_1, t_2, t_3, t_4, t_6, t_7\}, \{g_3, g_4, g_5\})$.

3 Discussion

This section discusses the desired properties for formal concept extensions towards fault-tolerant patterns. It enables to consider the pros and the cons of the available proposals and to better understand related open problems.

• **Fault tolerance:** Can we control the number of false values inside the bi-sets?
• **Relevancy:** Are they consistent w.r.t. the outside rows and columns? At least two views on consistency exist. We might say that a bi-set B is weakly consistent if it is maximal and if we have no row (resp. column) outside B identical to one row (resp. column) inside B. B is called strongly consistent if we have no row (resp. column) outside B with at most the same number of false values than one row (resp. column) of B.
• **Ease of interpretation:** For each bi-set (X, Y), does it exist a function which associates X and Y or even better a Galois connection? If a function exists which associates to each set X (resp. Y) at most a unique set Y (resp. X), the interpretation of each bi-set is much easier. Furthermore, it is interesting that such functions are monotonically decreasing, i.e., when the size of X (resp. Y) increases, the size of its associated set Y (resp. X) decreases. Such a property is meaningful: the more we have rows inside a bi-set, the less there are columns that can be associated to describe them (or vice versa). One of the appreciated properties of formal concepts is clearly the existence of such functions. If $f_1(X, \mathbf{r}) = \{g \in \mathcal{P} \mid \forall t \in X, (t, g) \in \mathbf{r}\}$ and $f_2(Y, \mathbf{r}) = \{t \in \mathcal{O} \mid \forall g \in Y, (t, g) \in \mathbf{r}\}$, (f_1, f_2) is a Galois connection between $\mathcal{O}$ and $\mathcal{P}$: f_1 and f_2 are decreasing functions w.r.t. set inclusion.
• **Completeness and efficiency**: Can we compute the whole collection of specified bi-sets, i.e., can we ensure a completeness w.r.t. the specified constraints? Is it tractable in practice?

The formal concepts satisfy these properties except the first one. Indeed, we have an explicit Galois connection which enables to compute the complete collection in many datasets of interest. These bi-sets are maximal and consistent but they are not fault-tolerant.

In a FBS pattern, the number of false values are only bounded on columns. The definition of this pattern is not symmetrical. They are not strongly consistent because we can have rows outside the bi-set with the same number of false values than a row inside (one of this false value must be on the δ-free set supporting set). On the columns, the property is satisfied. These bi-sets are however weakly consistent. There is no function from column to row sets (e.g., using $\delta = 1$ in $\mathbf{r}_1$, $(\{t_2, t_6, t_7\}, \{g_1, g_2, g_3, g_4, g_5\})$ and $(\{t_1, t_6, t_7\}, \{g_1, g_2, g_3, g_4, g_5\})$ are two FBS with the same set of columns, see Table 6 left). However, we have

Table 6. Illustration of the lack of function for FBS (left) and CBS (right)

	g_1	g_2	g_3	g_4	g_5	g_6	g_7
t_1	1	0	1	0	1	0	0
t_6	1	1	1	1	1	0	0
t_7	1	1	1	1	1	0	0
t_2	1	1	1	1	0	1	0
t_3	0	1	1	1	1	1	1
t_4	0	0	0	1	1	1	0
t_5	1	0	0	0	0	1	0

	g_2	g_3	g_4	g_1
t_1	0	1	0	1
t_2	1	1	1	1
t_3	1	1	1	0
t_4	0	0	1	0

a function between $2^{\mathcal{O}}$ to $2^{\mathcal{P}}$. In many datasets, including huge and dense ones, complete collections of FBS can be extracted efficiently. Further research is needed for a better characterization of more relevant FBS patterns which might remain easy to extract from huge databases, e.g., what is the impact of different δ-thresholds for the δ-free-set part and the δ-closure computation? how can we avoid an unfortunate distribution of the false values among the same rows?

CBS are symmetrical on rows and columns. Indeed, the number of exceptions is bounded on rows and on columns. CBS are weakly consistent but not strongly consistent (see Example 3). There are neither a function from $2^{\mathcal{O}}$ to $2^{\mathcal{P}}$ nor from $2^{\mathcal{P}}$ to $2^{\mathcal{O}}$ (e.g., $(\{t_1, t_2, t_3, t_4\}, \{g_1, g_3, g_4\})$ and $(\{t_1, t_2, t_3, t_4\}, \{g_2, g_3, g_4\})$ are two CBS with $\delta = 2$ having the same set of rows in Table 6 right). According to the implementation in [2], extracting these patterns can be untractable even in rather small datasets and this extraction strategy is not complete w.r.t. the specified constraints.

By definition, a DRBS has a bounded number of exceptions per row and per column and they are strongly consistent. Two new properties can be considered.

Property 4 (Existence of functions ϕ and ψ on DRBS ($\epsilon > 0$)). *For $\epsilon > 0$,* DRBS *patterns are embedded by two functions ϕ (resp. ψ) which associate to X (resp. Y) a unique set Y (resp. X).*

Property 5 (Monotonicity of ϕ and ψ on DRBS (δ fixed)). *Let $\mathcal{L}_{\delta,\epsilon}$ the collection of* DRBS *patterns and $\mathcal{L}'_{\tau\tau'}$ the subset of $\mathcal{L}_{\delta,\epsilon}$ s.t. $(X, Y) \in \mathcal{L}'_{\tau\tau'}$ iff (X, Y) contains at least a row (resp. column) with τ (resp. τ') false values in Y (resp. X), and such that no row (resp. column) contains more. Then, ϕ and ψ are decreasing functions on $\mathcal{L}'_{\tau\tau'}$.*

Unfortunately, the functions loose this property on the whole DRBS collection. Furthermore, we did not identified yet an intensional definition of these functions. As a result, it leads to a quite expensive computation of the complete collection. Looking for such functions is clearly one of the main challenges for further work.

4 Related Work

There are only few papers which propose definitions of set patterns with exceptions. To the best of our knowledge, most of the related work has concerned mono-dimensional patterns and/or the use of heuristic techniques. In [25], the frequent set mining task is extended towards fault-tolerance: given a threshold ϵ, an itemset P holds in a transaction X iff $\sharp(X \cap P) \geq (1-\epsilon)\sharp P$, where $\sharp X$ denotes the size of X. A level-wise algorithm is proposed but their fault-tolerant property is not anti-monotonic while this is crucially needed to achieve tractability. Therefore, [25] provides a greedy algorithm leading to an incomplete computation. [22] revisits this work and it looks for an anti-monotonic constraint such that a level-wise algorithm can provide every set whose density of 1 values is greater than δ in at least σ situations. Anti-monotonicity is obtained by enforcing that every subset of extracted sets satisfies the constraint as well. The extension of such dense sets to dense bi-sets is difficult: the connection which associates objects to properties and vice-versa is not decreasing while this is an appreciated property of formal concepts.

Instead of using a relative density definition, [18] considers an absolute threshold to define fault-tolerant frequent patterns: given a threshold δ, a set of columns P, such that $\sharp P > \delta$, holds in a row X iff $\sharp(X \cap P) \geq \sharp P - \delta$. To ensure that the support is significant for each column, they use a minimum support threshold per column beside the classical minimum support. Thus, each row of an extracted pattern contains less than δ false values and each column contains more true values than the given minimum support for each column. This definition is not symmetrical and the more the support increases, the less the patterns are relevant.

In [14], the authors are interested in geometrical tiles (i.e., dense bi-sets which involve contiguous elements given predefined orders on both dimensions). To extract them, they propose a local optimization algorithm which is not deterministic and thus can not guarantee the global quality of the extracted patterns. The hypothesis on built-in orders can not be accepted on many Boolean datasets.

Co-clustering (or bi-clustering) can be also applied to extract fault-tolerant bi-clusters [11, 20] from boolean data. It provides linked partitions on both dimensions and tend to compute rectangles with mainly true (resp. false) values. Heuristic techniques (i.e., local optimization) enable to compute one bi-partition, i.e., a quite restrictive collection of dense bi-sets. In fact, bi-clustering provides a global structure over the data while fault-tolerant formal concepts are typical local patterns. In other terms, these bi-sets are relevant but they constitute a quite restrictive collection of dense bi-sets which lack from formal properties.

5 Empirical Evaluation

In this section we investigate on the added-value of fault-tolerant pattern mining by considering experiments on both synthetic and "real world" data. For each experiment, we compare the formal concept mining algorithm output with the fault-tolerant approaches. The goal is not to assess the supremacy of a single class over the other ones, but to present an overview of the principal pros and cons of each approach in practical applications. First, we process artificially noised datasets to extract formal concepts and the three types of fault-tolerant bi-sets.

Then, we mine a "real world" medical dataset to get various collections of bi-sets for different parameters. Besides evaluating the performances and the size of the collections, we analyze the relevancy of the extracted bi-sets.

5.1 Experiments on Artificially Noised Data

Let us first discuss the evaluation method. We call $\mathbf{r}_2$ a reference data set, i.e., a dataset which is supposed to be noise free and contains built-in patterns. Then, we derive various datasets from it by adding some quantity of uniform random noise (i.e., for a X% noise level, each value is randomly changed with a probability of X%). Our goal is to compare the collection of formal concepts holding in the reference dataset with several collections of fault-tolerant formal concepts extracted from the noised matrices.

To measure the relevancy of each extracted collection ($\mathcal{C}_e$) w.r.t the reference one ($\mathcal{C}_r$), we test the presence of a subset of the reference collection in each of them. Since both sets of objects and properties of each formal concept can be changed when noise is introduced, we identify those having the largest area in common with the reference. Our measure, called σ, takes into account the common area and is defined as follows:

$$\sigma(\mathcal{C}_r, \mathcal{C}_e) = \frac{\rho(\mathcal{C}_r, \mathcal{C}_e) + \rho(\mathcal{C}_e, \mathcal{C}_r)}{2}$$

where ρ is computed as follows:

$$\rho(\mathcal{C}_1, \mathcal{C}_2) = \frac{1}{\sharp\mathcal{C}_1} \sum_{(X_i,Y_i)\in\mathcal{C}_1} \max_{(X_j,Y_j)\in\mathcal{C}_2} \frac{\sharp(X_i \cap X_j) + \sharp(Y_i \cap Y_j)}{\sharp(X_i \cup X_j) + \sharp(Y_i \cup Y_j)}$$

Here, $\mathcal{C}_r$ is the collection of formal concepts computed on the reference dataset, $\mathcal{C}_e$ is a collection of patterns in a noised dataset. When $\rho(\mathcal{C}_r, \mathcal{C}_e) = 1$, all the bi-sets belonging to $\mathcal{C}_r$ have identical instances in the collection $\mathcal{C}_e$. Analogously, when $\rho(\mathcal{C}_e, \mathcal{C}_r) = 1$, all the bi-sets belonging to $\mathcal{C}_e$ have identical instances in the collection $\mathcal{C}_r$. Indeed, when $\sigma = 1$, the two collections are identical. High values of σ, mean not only that we can find all the formal concepts of the reference collection in the noised matrix, but also that the noised collection does not contain many bi-sets that are too different from the reference ones.

In this experiment, $\mathbf{r}_2$ concerns 30 objects (rows) and 15 properties (columns) and it contains 3 formal concepts of the same size which are pair-wise disjoint. In other terms, the formal concepts in $\mathbf{r}_2$ are $(\{t_1, \ldots, t_{10}\}, \{g_1, \ldots, g_5\})$, $(\{t_{11}, \ldots, t_{20}\}, \{g_6, \ldots, g_{10}\})$, and $(\{t_{21}, \ldots, t_{30}\}, \{g_{11}, \ldots, g_{15}\})$. Then, we generated 40 different datasets by adding to $\mathbf{r}_2$ increasing quantities of noise (from 1% to 40% of the matrix). A robust technique should be able to capture the three formal concepts even in presence of noise. Therefore, for each dataset, we have extracted a collection of formal concepts and different collections of fault-tolerant patterns with different parameters. For FBS collection, we considered δ values between 1 and 6. Then we extracted two groups of CBS collections given parameter δ (resp. δ') for the maximum number of false values per row

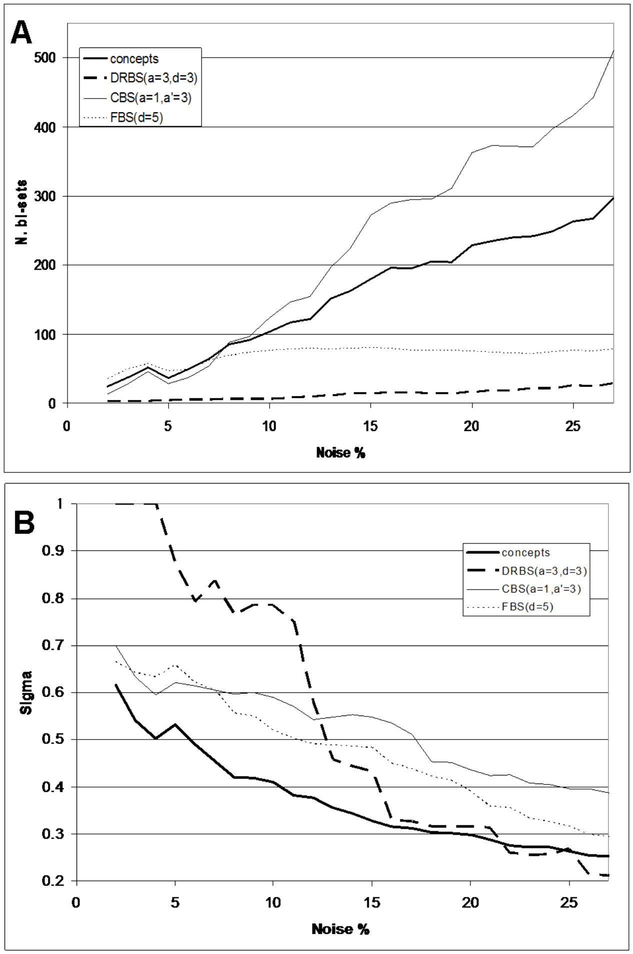

Fig. 1. Size of different collections of bi-sets and related values of σ w.r.t. noise level for all types of bi-sets

(resp. per column): one with $\delta = 1$ and $\delta' = 1 \ldots 3$ and the second with $\delta' = 1$ and $\delta = 1 \ldots 3$. Finally we extracted DRBS collections for each combination of $\delta = 1 \ldots 3$ and $\epsilon = 1 \ldots 3$.

In Fig. 1, we only report the best results w.r.t. σ for each class of patterns. Fig. 1A provides the number of extracted patterns in each collection. Fault-tolerant bi-set collections contain almost always less patterns than the collection of formal concepts. The only exception is the CBS class when $\delta = 1$. The DRBS class performs better than the other ones. The size of its collections is almost constant, even for rather high levels of noise. The discriminant parameter is ϵ. In Fig. 1B, the values of the σ measure for DRBS collections obviously decrease when the noise ratio increases. In general, every class of fault-tolerant bi-set

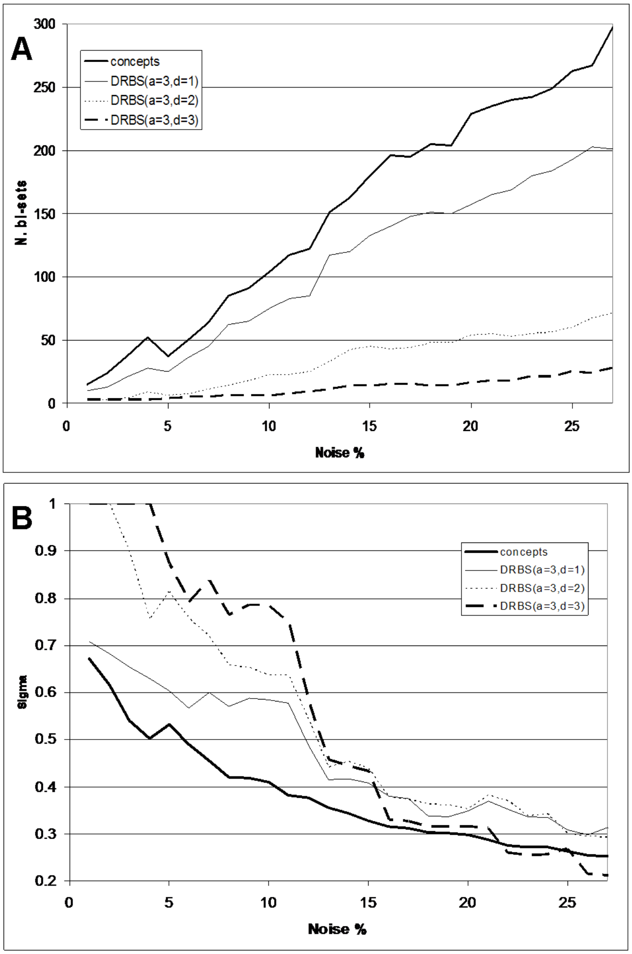

Fig. 2. Size of different collections of bi-sets and related values of σ w.r.t. noise level for different instances of DRBS collections

performs better than the formal concept one. In terms of relevancy, the DRBS pattern class gives the best results as well. Notice that the results for FBS and CBS classes are not significantly different when their parameters change. The parameter that has the greatest impact on σ value for the DRBS patterns is ϵ. For reasonable levels of noise ($< 15\%$), it makes sense to use DRBS. For higher levels, CBS and FBS perform slightly better.

In Fig. 2, we report the experiments on the extraction of DRBS collections with $\delta = 3$ and $\epsilon = 1 \ldots 3$. Fig. 2A shows the number of extracted patterns. The size of the collections is drastically reduced when ϵ grows. Fig. 2B provides the σ measure for these collections. Using a higher ϵ value improves the quality of the results because less patterns are produced. When the noise level is smaller than 5%, the collection of DRBS, with $\epsilon = 2..3$, is the same as the three formal concepts holding in $\mathbf{r}_2$. This experiment confirms that fault-tolerant bi-sets are more robust to noise than formal concepts, and that the provided collection for the crucially needed expert-driven interpretation is considerably reduced.

5.2 Experiments on a Medical Dataset

It is important to get a qualitative feedback about fault-tolerant pattern relevancy in a real dataset. For this purpose, we have considered the real world medical dataset meningitis [12]. These data have been gathered from children hospitalized for acute meningitis over a period of 4 years. The pre-processed Boolean dataset is composed of 329 patients described by 60 Boolean properties encoding clinical signs (e.g., consciousness troubles), cytochemical analysis of the cerebrospinal fluid (e.g., C.S.F proteins) and blood analysis (e.g., sedimentation rate). The majority of the cases are viral infections, whereas about one quarter of the cases are caused by bacteria. It is interesting to look at the bacterial cases since they need treatment with suitable antibiotics, while for the viral cases a simple medical supervision is sufficient. A certain number of attribute-variable pairs have been identified as being characteristic of the bacterial form of meningitis [12, 21]. In other terms, the quality of the fault-tolerant patterns can be evaluated w.r.t. available medical knowledge. Our idea is that by looking for rather large fault-tolerant bi-sets, the algorithms will provide some new associations between attribute-value pairs (Boolean properties) and objects. If the whole sets of objects and properties within bi-sets are compatible (e.g., all the objects are of bacterial type, and all the properties are compatible with bacterial meningitis), then we can argue that we got new relevant information.

A straightforward approach to avoid some irrelevant patterns and to reduce the pattern collection size is to use size constraints on bi-set components. For this experiment, we enforce a minimal size of 10 for sets of objects and 5 for sets of properties. Using D-Miner [4], we computed the collection of such large enough formal concepts and we got more than 300 000 formal concepts in a

Table 7. Size and extraction time for FBS and DRBS in meningitis

Formal Concepts						
size	354 366					
time	5s					
FBS						
δ	1	2	3	4	5	6
size	141 983	67 898	39 536	25 851	18 035	13 382
time	19s	10s	6s	4s	3s	2s
DRBS (δ=1)						
ϵ	1	2	3	4	5	6
size	-	75 378	22 882	8 810	4 164	2 021
time	-	1507s	857s	424s	233s	140s

relatively short time (see Table 7). It is obviously hard to exploit such a quantity of patterns. For instance, we were not able to post-process this collection to produce CBS patterns according to [2].

Then, we tried to extract different collections of FBS and DRBS. For FBS, with $\delta = 1$ (at most one exception per column), we got a 60% reduction on the size of the computed bi-sets. Using values of δ between 2 and 6, this size is reduced at each step by a coefficient between 0.5 and 0.3. Finally, we used DR-Miner to extract different collections of DRBS. The δ parameter was set to 1 (at most one exception per row and per column) and we used the parameter ϵ to further reduce the size of the computed collection. Setting $\epsilon = 1$ leads to an untractable extraction but, with $\epsilon = 2$, the resulting collection is 80% smaller than the related formal concept collection. Moreover, with $\delta = 1$ and $\epsilon = 2$ the size of the DRBS collection is much smaller than the computed FBS collection for the same constraint (i.e., $\delta = 1$). On the other hand, computational times are sensibly higher.

We now consider relevancy issues. We have been looking for bi-sets containing the property "presence of bacteria detected in C.S.F. bacteriological analysis" with at least one exception. This property is typically true in the bacterial type of meningitis [12, 21]. By looking for bi-sets satisfying such a constraint, we expect to obtain associations between bacterial meningitis objects and properties characterizing this class of meningitis. We analyzed the collection of FBS when $\delta = 1$. We got 763 FBS that satisfy the chosen constraint. Among these, 124 FBS contain only one viral meningitis object. We got no FBS containing more than one viral object. Properties belonging to these FBS are either characteristic features of the bacterial cases or non discriminant (but compatible) features such as the age and sex of the patient. When $\delta = 2$, the number of FBS satisfying the constraint is 925. Among them, 260 contain at least one viral case of meningitis, and about 25 FBS contain more than one viral case. For $\delta = 5$ the obtained bi-sets are no longer relevant, i.e., the exceptions include contradictory Boolean properties (e.g., presence and absence of bacteria). We performed the same analysis on DRBS for $\epsilon = 2$. We found 24 rather large DRBS. Among them, 2 contain also one viral object. Only one DRBS seems irrelevant: it contains 3 viral and 8 bacterial cases. Looking at its Boolean properties, we noticed that they were not known as discriminant w.r.t. the bacterial meningitis. If we analyze the collection obtained for $\epsilon = 3$, there is only one DRBS satisfying the constraint. It is a rather large bi-set involving 11 Boolean properties and 14 objects. All the 14 objects belong to the bacterial class and the 11 properties are compatible with the bacterial condition of meningitis. It appears that using DRBS instead of FBS leads to a smaller number of relevant bi-sets for our analysis task (24 against 763). Notice however that DRBS are larger than FBS (for an identical number of exceptions): it means that the information provided by several FBS patterns might be incorporated in only one DRBS pattern. Moreover we got no DRBS pattern whose set of properties is included in the set of properties of another one. This is not the case for FBS.

To summarize this experiment, let us first note that using size constraints to reduce the size of the collection is not always sufficient. meningitis is a rather small dataset which leads to the extraction of several hundreds of thousands of formal concepts (about 700 000 if no constraint is given). By extracting fault-tolerant bi-sets, we reduce the size of the collection to be interpreted and this is crucial for the targeted exploratory knowledge discovery processes. In particular, for DRBS, the ϵ parameter is more stringent than the δ parameter. Then, the relevancy of the extracted patterns can be improved if a reasonable number of exceptions is allowed. For instance, extracting FBS with a low δ (1 or 2) leads to relevant associations while a high δ (e.g., 5) introduces too many irrelevant bi-sets. From this point of view, the DRBS class leads to the most interesting results and their quality can be improved by tuning the ϵ parameter. On the other hand, FBS are easier to compute, even in rather hard contexts, while computing DRBS remains untractable in many cases.

5.3 Post-experiment Discussion

Both experiments have shown the advantages of using a fault-tolerant bi-set mining technique in noisy data. Let us emphasize that adding minimal size constraints on both dimensions to fault-tolerance constraints is useful: it ensures that the number of false values is quite small w.r.t. the bi-set size. It enables to speed up the mining process as well because such constraints can be exploited for efficient search space pruning.

Using CBS might be a good choice when a relatively small collection of formal concepts is already available. When data are dense or significantly correlated, such as in meningitis, CBS mining fails even in relatively small matrices. In this case, we can use either FBS or DRBS. Experiments have shown that the second class gives more relevant results and that DRBS pattern collection sizes tend to be significantly smaller. Trigging the ϵ parameter enables to further reduce the collection size while preserving relevancy. The problem is however that this task turns out to be untractable for large matrices. On the other hand, FBS can be rather easily extracted but their semantics is not symmetrical and it affects their relevancy. A post-processing step might be used to eliminate all the bi-sets which do not satisfy the maximum error constraint on rows.

6 Conclusion

We have discussed a fundamental limitation of formal concept mining to capture strong associations between sets of objects and sets of properties in possibly large and noisy Boolean datasets. Relevancy issues are crucial to avoid too many irrelevant patterns during the targeted data mining processes. It is challenging to alleviate the expensive interpretation phases while still promoting unexpectedness of the discovered (local) patterns. The lack of consensual extensions of formal concepts towards fault-tolerance has given rise to several ad-hoc proposals. Considering three recent proposals, we have formalized fault-tolerant bi-dimensional pattern mining within a constraint-based approach. It has been useful for a better understanding of the needed trade-off between extraction feasibility, completeness, relevancy, and ease of interpretation. An empirical evaluation on both synthetic and real-life medical data has been given. It shows that fault-tolerant formal concept mining is possible and this should have an impact on the dissemination of local set pattern discovery techniques in intrinsically noisy Boolean data. DRBS pattern class appears as a well-designed class but the price to pay is computational complexity. The good news are that (a) the submitted inductive queries on DRBS patterns might involve further user-defined constraints which can be used for efficient pruning, and (b) one can look for more efficient data structures and thus a more efficient implementation of the DR-Miner generic algorithm. A pragmatic usage of available algorithms is indeed to extract some bi-sets, e.g., formal concepts, and then select some of them (say $B = (X, Y)$) for further extensions towards fault-tolerant patterns: it becomes, e.g., the computation of a DRBS pattern (say $B' = (X', Y')$ such that the constraint $B \subseteq B'$ is enforced. Also, a better characterization of FBS pattern class might be useful for huge database processing.

Acknowledgements. The authors thank P. François and B. Crémilleux who provided meningitis. J. Besson's work is funded by INRA. This research is partly funded by ACI CNRS MD 46 Bingo and by EU contract IQ FP6-516169 (FET arm of the IST programme).

References

1. C. Antunes and A.L. Oliveira. Constraint relaxations for discovering unknown sequential patterns. In *Revised selected and invited papers KDID'04*, volume 3377 of *LNCS*, pages 11–32. Springer-Verlag, 2005.
2. J. Besson, C. Robardet, and J-F. Boulicaut. Mining formal concepts with a bounded number of exceptions from transactional data. In *Revised selected and invited papers KDID'04*, volume 3377 of *LNCS*, pages 33–45. Springer-Verlag, 2004.
3. J. Besson, C. Robardet, and J-F. Boulicaut. Approximation de collections de concepts formels par des bi-ensembles denses et pertinents. In *Proceedings CAp 2005*, pages 313–328. PUG, 2005. An extended and revised version in English is submitted to a journal.
4. J. Besson, C. Robardet, J-F. Boulicaut, and S. Rome. Constraint-based concept mining and its application to microarray data analysis. *Intelligent Data Analysis*, 9(1):59–82, 2005.
5. M. Bistarelli and F. Bonchi. Interestingness is not a dichotomy: introducing softness in constrained pattern mining. In *Proceedings PKDD'05*, volume 3721 of *LNCS*, pages 22–33. Springer-Verlag, 2005.
6. J-F. Boulicaut. Inductive databases and multiple uses of frequent itemsets: the cInQ approach. In *Database Technologies for Data Mining - Discovering Knowledge with Inductive Queries*, volume 2682 of *LNCS*, pages 1–23. Springer-Verlag, 2004.
7. J-F. Boulicaut, A. Bykowski, and C. Rigotti. Approximation of frequency queries by mean of free-sets. In *Proceedings PKDD'00*, volume 1910 of *LNAI*, pages 75–85. Springer-Verlag, 2000.
8. J-F. Boulicaut, A. Bykowski, and C. Rigotti. Free-sets: a condensed representation of boolean data for the approximation of frequency queries. *Data Mining and Knowledge Discovery journal*, 7 (1):5–22, 2003.

9. C. Bucila, J. E. Gehrke, D. Kifer, and W. White. Dualminer: A dual-pruning algorithm for itemsets with constraints. *Data Mining and Knowledge Discovery journal*, 7 (4):241–272, 2003.
10. L. De Raedt. A perspective on inductive databases. *SIGKDD Explorations*, 4(2):69–77, 2003.
11. I. S. Dhillon, S. Mallela, and D. S. Modha. Information-theoretic co-clustering. In *Proceedings ACM SIGKDD 2003*, pages 89–98, Washington, USA, 2003. ACM Press.
12. P. François, C. Robert, B. Cremilleux, C. Bucharles, and J. Demongeot. Variables processing in expert system building: application to the aetiological diagnosis of infantile meningitis. *Med Inform*, 15(2):115–124, 1990.
13. F. Geerts, B. Goethals, and T. Mielikäinen. Tiling databases. In *Proceedings DS'04*, volume 3245 of *LNAI*, pages 278–289. Springer-Verlag, 2004.
14. A. Gionis, H. Mannila, and J. K. Seppänen. Geometric and combinatorial tiles in 0-1 data. In *Proceedings PKDD'04*, volume 3202 of *LNAI*, pages 173–184. Springer-Verlag, 2004.
15. B. Goethals and M. Zaki. *Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations FIMI 2003*. CEUR-WS, Melbourne, USA, 2003.
16. T. Imielinski and H. Mannila. A database perspective on knowledge discovery. *Communications of the ACM*, 39(11):58–64, 1996.
17. S. O. Kuznetsov and S. A. Obiedkov. Comparing performance of algorithms for generating concept lattices. *Journal of Experimental and Theoretical Artificial Intelligence*, 14 (2-3):189–216, 2002.
18. J. Pei, A. K. H. Tung, and J. Han. Fault-tolerant frequent pattern mining: Problems and challenges. In *SIGMOD wokshop DMKD*. ACM workshop, 2001.
19. R. Pensa and J-F. Boulicaut. From local pattern mining to relevant bi-cluster characterization. In *Proceedings IDA'05*, volume 3646 of *LNCS*, pages 293–304. Springer-Verlag, 2005.
20. R. G. Pensa, C. Robardet, and J.-F. Boulicaut. A bi-clustering framework for categorical data. In *Proceedings PKDD'05*, volume 3721 of *LNCS*, pages 643–650. Springer-Verlag, 2005.
21. C. Robardet, B. Crémilleux, and J.-F. Boulicaut. Characterization of unsupervized clusters by means of the simplest association rules: an application for child's meningitis. In *Proceedings IDAMAP'02 co-located with ECAI'02*, pages 61–66, Lyon, F, 2002.
22. J. K. Seppänen and H. Mannila. Dense itemsets. In *Proceedings ACM SIGKDD'04*, pages 683–688, Seattle, USA, 2004. ACM Press.
23. G. Stumme, R. Taouil, Y. Bastide, N. Pasqier, and L. Lakhal. Computing iceberg concept lattices with TITANIC. *Journal of Data and Knowledge Engineering*, 42 (2):189–222, 2002.
24. R. Wille. Restructuring lattice theory: an approach based on hierarchies of concepts. In I. Rival, editor, *Ordered sets*, pages 445–470. Reidel, 1982.
25. C. Yang, U. Fayyad, and P. S. Bradley. Efficient discovery of error-tolerant frequent itemsets in high dimensions. In *Proceedings ACM SIGKDD'01*, pages 194–203. ACM Press, 2001.