Data Min Knowl Disc
DOI 10.1007/s10618-008-0111-4

Adequate condensed representations of patterns

Arnaud Soulet · Bruno Crémilleux

Received: 20 June 2008 / Accepted: 24 June 2008

Abstract Patterns are at the core of the discovery of a lot of knowledge from data but their uses are limited due to their huge number and their mining cost. During the last decade, many works addressed the concept of condensed representation w.r.t. frequency queries. Such representations are several orders of magnitude smaller than the size of the whole collections of patterns, and also enable us to regenerate the frequency information of any pattern. In this paper, we propose a framework for condensed representations w.r.t. a large set of new and various queries named *condensable functions* based on interestingness measures (e.g., frequency, lift, minimum). Such condensed representations are achieved thanks to new closure operators automatically derived from each condensable function to get *adequate condensed representations*. We propose a generic algorithm MicMac to efficiently mine the adequate condensed representations. Experiments show both the conciseness of the adequate condensed representations and the efficiency of our algorithm.

Keywords Pattern mining · Condensed representation · Closure operator

Responsible editors: Walter Daelemans, Bart Goethals, and Katharina Morik.

A. Soulet (✉)
LI, Université François Rabelais de Tours, 3 place Jean Jaurès, 41029 Blois, France
e-mail: arnaud.soulet@univ-tours.fr

B. Crémilleux
GREYC-CNRS, Université de Caen, Campus Côte de Nacre, 14032 Caen Cedex, France
e-mail: bruno.cremilleux@info.unicaen.fr

1 Introduction

It is well-known that a current challenge in Knowledge Discovery in Databases (KDD) is to cope with the "pattern flooding which follows data flooding" that is unfortunately so typical in exploratory KDD processes. Indeed, a range of powerful techniques for producing local patterns has been developed over the last decade (Morik et al. 2005), but the overwhelming number of produced patterns hinders their uses. Such massive output hampers the individual analysis performed by end-users of data whereas collections of local patterns can capture subtle relationships in the data leading to the discovery of precious nuggets of knowledge.

One solution to this problem relies on the *condensed representation* principle. The idea is to compute a representation $\mathcal{R}$ of the produced patterns which is lossless: the whole collection of patterns can be efficiently derived from $\mathcal{R}$ and $\mathcal{R}$ has to be as concise as possible. This approach has been mainly developed in the context of the frequency measure (Calders et al. 2004) and there are very few works addressing other measures (Giacometti et al. 2002; Soulet et al. 2004). In real-world applications, we claim that interestingness of patterns is evaluated by various user-defined measures (e.g., frequency, confidence, lift, minimum). Combinations of such measures and constraints coming from these measures bring a rich quantitative information. This paper extends the concept of pattern condensed representations to a broad spectrum of measures and constraints (see examples in Table 1) and enables us the efficient discovery of various kinds of patterns (e.g., satisfying a utility-based function combining minimal and maximal restrictions (Yao et al. 2004, see f_6).

Equivalence classes are at the core of the pattern condensed representations. They have a nice property to summarize information: an equivalence class can be uniquely and concisely represented by a closed pattern (Pasquier et al. 1999) or a set of generators (Boulicaut et al. 2003; Calders et al. 2004). There is a direct link between the generators and the closed patterns: any closed pattern is the closure of at least one generator of its equivalence class. The closure operator is at the root of the definition of the equivalence classes and thus the condensation. In Sect. 2.2, we will see that works in literature use the Galois closure. It is appropriate to frequency based measures but unfortunately not to other measures. In this paper, we bring a major improvement by proposing a

Table 1 Examples of condensable functions (γ, ρ, α are thresholds)

Condensable function	Category
f_1: $X \mapsto freq(X, \mathcal{D})$	Measure
f_2: $X \mapsto min(X.val)$	Measure
f_3: $X \mapsto (min(X.val), freq(X, \mathcal{D}))$	Measure/measure
f_4: $X \mapsto freq(X, \mathcal{D}) \geq \gamma$	Constraint
f_5: $X \mapsto freq(X, \mathcal{D}_1)/freq(X, \mathcal{D}_2) \geq \rho$	Constraint
f_6: $X \mapsto ((min(X.val) + max(X.val))/2 \geq \alpha, freq(X, \mathcal{D}))$	Constraint/measure
f_7: $X \mapsto freq(X, \mathcal{D}_1)/freq(X, \mathcal{D})$	Measure
f_8: $X \mapsto (freq(X, \mathcal{D}_1) \times \lvert\mathcal{D}\rvert)/(freq(X, \mathcal{D}) \times \lvert\mathcal{D}_1\rvert)$	Measure
f_9: $X \mapsto (max(X.val), freq(X, \mathcal{D}))$	Measure/measure
f_{10}: $X \mapsto freq_\vee(X, \mathcal{D})$	Measure

new closure operator which is adequate for a large set of various measures. This is the key point which enables us to extend the principle of the condensed representations. However, the user is generally also interested in the values of an evaluation measure associated to patterns. For that reason, a condensed representation must allow to infer not only the patterns, but also the values of an evaluation measure such as the frequency without accessing the data. We will see that our approach provides such kind of information for all handled measures. Finally, the success of condensed representations also lies in the efficiency of their mining: it is faster to extract a condensed representation than the whole collection of patterns, especially in dense and/or correlated data (Goethals and Zaki 2003). Experiments demonstrate the scalability of our approach and that we are able to mine patterns under measures and constraints for which only naive methods are available. More interestingly, these naive methods may fail in large databases.

Condensed representations have a lot of applications and their use is not limited to obtain frequent patterns more efficiently. First, they make interactive KDD processes more easily: they have suitable properties to infer relevant patterns directly from the condensed representations and they can be used as cache mechanisms. It is a way to interact faster with queries than direct mining. Second, their properties are useful for higher KDD tasks. For instance, the generators and closed patterns are used to produce non-redundant (Zaki 2000a) or informative rules (Bastide et al. 2000). It is also possible to exploit condensed representations for classification (Crémilleux and Boulicaut 2002). Condensed representations are a key concept of inductive databases (Imielinski and Mannila 1996): this paradigm is based on declarative queries instead of procedural techniques and storing local patterns as intermediate results is then a major step.

Contributions. The main goal of this paper is to present a generic framework for pattern condensed representations. We introduce the notion of *condensable function* which is at the core of the definition of this framework. This latter covers a broad spectrum of functions including measures and constraints having no suitable property of monotonicity (see Sect. 2.2). Condensed representations are achieved thanks to a new closure operator which depends on a condensable function f and thus this operator automatically becomes adequate to f. Based on this closure operator, we propose the MicMac algorithm. It mines the condensed representations for any condensable function, so-called *adequate condensed representations*. This algorithm is efficient and the adequate condensed representations are concise.

The outline of this paper is as follows. Section 2 sketches basic definitions and related work. In Sect. 3, we propose our generic framework for pattern condensed representations according to any condensable function. Section 4 defines the adequate condensed representations and the MicMac algorithm. Section 5 provides in depth experimental results.

2 Context and related work

2.1 Basic definitions

Let $\mathcal{I}$ be a set of distinct literals called *items*, an itemset (or pattern) is a non-null subset of $\mathcal{I}$. The language of itemsets corresponds to $\mathcal{L}_{\mathcal{I}} = 2^{\mathcal{I}} \setminus \emptyset$. A transactional

$\mathcal{D}$

Trans.	Items						
t_1	A	B			E	F	$\mathcal{D}_1$
t_2	A				E		
t_3	A	B	C	D			
t_4	A	B	C	D	E		$\mathcal{D}_2$
t_5				D	E		
t_6			C			F	

Item	A	B	C	D	E	F
val	50	30	75	10	30	15

Fig. 1 Example of a transactional context $\mathcal{D}$

dataset is a multi-set of itemsets of $\mathcal{L}_\mathcal{I}$. Each itemset, usually called *transaction*, is a database entry. For instance, Fig. 1 gives a transactional dataset $\mathcal{D}$ where 6 transactions $t_1, \ldots, t_6$ are described by 6 items $A, \ldots, F$.

Pattern mining aims at discovering information from all the patterns or a subset of $\mathcal{L}_\mathcal{I}$. More precisely, constraint-based mining task selects all the itemsets of $\mathcal{L}_\mathcal{I}$ present in $\mathcal{D}$ and satisfying a predicate which is named *constraint*. A constraint expresses the interest of the user to focus on the most promising patterns according to his point of view. Introduced in Agrawal and Srikant (1994), the minimal frequency constraint provides the itemsets having a frequency exceeding a given minimal threshold $\gamma > 0$: $freq(X, \mathcal{D}) \geq \gamma$. The *(conjunctive) frequency* of an itemset X, denoted by $freq(X, \mathcal{D})$, is the number of transactions in $\mathcal{D}$ containing X. Many works (Ng et al. 1998) replace the frequency by other interestingness measures to evaluate the relevance of itemsets. These measures are defined from different primitive functions. For instance, the function $freq_\vee$ denotes the disjunctive frequency of an itemset (i.e., the number of transactions containing at least one item of X), and $count$ the cardinality of X. From Fig. 1, $freq(AC, \mathcal{D}) = 2$, $freq_\vee(AC, \mathcal{D}) = 5$ and $count(AC) = 2$. Additional information (such as numeric values associated to items) can be used. Given a function $val : \mathcal{I} \rightarrow \Re$, we extend it to an itemset X and note $X.val$ the multiset $\{val(i) | i \in X\}$. This kind of function is used with the usual SQL-like primitives sum, min and max. For instance, $min(X.val)$ is the minimal val of each item of X. From Fig. 1, $min(AC.val) = 50$.

2.2 Related work

In the literature, there are several families of condensed representations. Introduced in Mannila and Toivonen (1997), the *borders* (or boundaries) are condensed representations of (anti-)monotone constraints (this notion directly stems from the field of concept-learning (Mitchell 1982) where boundaries sum up version spaces). These sets of minimal or maximal itemsets w.r.t. a specialization relation are enough to determine whether an itemset satisfies monotone and/or anti-monotone constraints. Borders are concise representations but unfortunately, it is impossible to directly regenerate values of interestingness measures for any itemset from only the borders. There are several works on *condensed representations for frequent itemsets* which propose compact representations adequate to the frequency function by introducing new kind of patterns (e.g., disjunction-free sets (Bykowski and Rigotti 2003), Non-Derivable Itemsets (Calders and Goethals 2002), k-free itemsets (Calders and Goethals 2003), itemsets

with negation (Kryszkiewicz 2005)). Nevertheless, these condensed representations mainly use the inclusion-exclusion principle (see Calders et al. (2004) for more details) to regenerate the frequency of any itemset. As this technique requires a computation from several itemset frequencies of the condensed representation, it may be time consuming. As the inclusion-exclusion principle is specific to the frequency, a strong limitation of this approach is that it cannot be extended to other measures such as min, sum, etc.

Many condensed representations rely on a closure operator (Birkhoff 1967) and they are called *closure-based condensed representations*. They are related to lattice theory (Birkhoff 1967). The most popular closure operator is very likely the Galois closure: $h(X) = \{i \in \mathcal{I} | freq(X \cup \{i\}, \mathcal{D}) = freq(X, \mathcal{D})\}$. The function h defines equivalence classes in $\mathcal{L}_{\mathcal{I}}$: two itemsets X and Y are in the same equivalence class iff $h(X) = h(Y)$, and one can straightforwardly prove that $freq(X, \mathcal{D}) = freq(Y, \mathcal{D})$. The key idea of closure-based condensed representations is to select a representative for each equivalence class (closed patterns (Pasquier et al. 1999) or generators (Boulicaut et al. 2003) also called *free* or *key* itemsets). There are a few extensions of the closure-based condensed representations to measure other than the frequency. In Soulet et al. (2004) and Li et al. (2007), it is shown that the previous condensed representations are adequate to any frequency-based measure (e.g., lift, growth rate). The essential patterns constitute a condensed representation adequate to the disjunctive frequency (Casali et al. 2005). The closure operator h is adapted in Gasmi et al. (2007) to deal with negative items (condensed representations include literal sets such as $A\overline{B}$ where $freq(A\overline{B}, \mathcal{D})$ is the number of transactions containing A and not B). We refer to the literature for more details (e.g., adding a border, computing several measures for each itemset) concerning the various existing condensed representations.

2.3 Problem statement

To the best of our knowledge, all the proposed closure-based condensed representations are dedicated to the frequency (conjunctive, disjunctive or negative) or frequency-based measure. Other measures have received very little attention: only monotone (i.e., monotonically decreasing or increasing) functions are addressed in Giacometti et al. (2002). In this paper, we go further by proposing a generic framework for closure-based condensed representations which encompasses these measures and deals with many others having no suitable properties such as monotonicity. We think that generalizing closure-based condensed representations will offer many exciting new tools for KDD (e.g., minimal rules w.r.t. min), similarly there are many uses stemming from the frequency.

Obviously, the closure operator is at the core of the closure-based condensed representations. The Galois closure is broadly the most often used in the literature. The key idea of this paper is to use a closure operator adequate to a function f. Indeed, a closure operator adequate to a function f_1 is generally not adequate to another function f_2. For instance, the Galois closure h which is adequate to the frequency is inadequate for the function min. In Fig. 1, we can see that $h(AC) = ABCD$

and $freq(AC, \mathcal{D}) = freq(h(AC), \mathcal{D}) = freq(ABCD, \mathcal{D})$, but $min(AC.val) \neq min(ABCD.val)$ since $min(AC.val) = 50$ and $min(ABCD.val) = 10$). Given a function f, the challenge is to find a closure operator h_f such that $f(X) = f(h_f(X))$. A naive technique would consider the identity function $id\colon X \mapsto X$. It is an adequate closure operator for any function f because id is a closure operator and for all X, one has $f(id(X)) = f(X)$. But, each itemset is its own representative and thus the corresponding representation is not condensed at all. The next section shows how to tackle this issue for a very large set of functions, the condensable functions.

3 Framework of condensable functions

3.1 Definition of condensable functions

Itemsets condensation comes from dependencies between itemsets. We define the notion of *preserving function* which reveals such a dependence between an itemset and its specializations. This dependence will enable us the summarization achieved by the condensed representations proposed in this paper.

Definition 1 (*Preserving function*) Let E be a set. A function $p\colon \mathcal{L}_{\mathcal{I}} \to E$ is preserving iff for each $i \in \mathcal{I}$ and for each $X \subseteq Y$ if $p(X \cup \{i\}) = p(X)$ then $p(Y \cup \{i\})$ equals to $p(Y)$.

A function p is a preserving function whenever the addition of an item i does not modify $p(X)$, then the addition of i does not modify the value of p for any specialization of X. Many functions are preserving: $freq$, $freq_{\vee}$, $count$, min, max, sum, etc. For the purpose of illustration, we show that min is a preserving function. Let $X \in \mathcal{L}_{\mathcal{I}}$ be an itemset and $i \in \mathcal{I}$ be an item such that $min(X.val) = min(X \cup \{i\}.val)$, $i.val$ is then greater than or equal to $min(X.val)$. Now, let $Y \supseteq X$, one has $min(Y.val) \leq min(X.val)$ because min decreases with X. As $i.val \geq min(X.val)$, we obtain $i.val \geq min(Y.val)$ and conclude that $min(Y \cup \{i\}.val) = min(Y.val)$.

The property of condensation expressed by the preserving functions still holds when they are combined as *condensable functions* formally defined as follows:

Definition 2 (*Condensable function*) Let E be a set. A function $f\colon \mathcal{L}_{\mathcal{I}} \to E$ is condensable iff there exist a function F and k preserving functions $p_1, \ldots, p_k$ such that $f = F(p_1, \ldots, p_k)$.

Basically, a condensable function is a compound of preserving functions, there is no restriction on F. We argue that the set of condensable functions is very broad and general. Table 1 provides examples of condensable functions. First, according to the nature of E, a condensable function is a constraint (if $E = \{false, true\}$: cf. f_4 and f_5) or a measure (if $E \subseteq \Re$: cf. f_1, f_2, f_7, f_8 and f_{10}). Besides, several measures or constraints can be jointly considered (e.g., f_3, f_6, f_9). Second, the set of condensable functions obviously includes all the preserving functions (e.g., f_1, f_2, f_{10}), but it is much larger and also contains many non-preserving functions. For instance, most of interestingness measures are condensable (e.g., growth rate f_5, confidence f_7, lift f_8)

because these measures are combinations of the frequency which is preserving. As our framework on condensed representations is based on the condensable functions which are very various, it clearly encompasses the current approaches on closure-based condensed representations dealing with condensable functions (e.g., $freq$ (Pasquier et al. 1999; Boulicaut et al. 2003), $freq_\vee$ (Casali et al. 2005), frequency-based measures (Soulet et al. 2004; Li et al. 2007)).

3.2 Adequate closure operators for condensable functions

This section shows how properties of condensable functions enable us to design a proper closure operator adequate to a condensable function. This adequate closure operator is at the heart of the conciseness of the condensed representation. Let X be an itemset. Intuitively, a closure operator completes X with all the items which do not affect $f(X)$. For instance, with the conjunctive frequency, the closure of X gathers all the items i such that $freq(X \cup \{i\}, \mathcal{D}) = freq(X, \mathcal{D})$. In the case of a condensable function $f = F(p_1, \ldots, p_k)$, we merely have to consider simultaneously the different preserving function p_j. We formally define the *adequate closure operator* for any condensable function as follows (Theorem 1 and Theorem 2 below show that $\mathtt{cl}_f$ is adequate to f and is a closure operator):

Definition 3 (*Adequate closure operator*) Let $f = F(p_1, \ldots, p_k)$ be a condensable function, the closure operator adequate to f, denoted by $\mathtt{cl}_f$, is defined as below:

$$\mathtt{cl}_f : X \mapsto \{i \in \mathcal{I} | \forall j \in \{1, \ldots, k\},\ p_j(X \cup \{i\}) = p_j(X)\}$$

Definition 3 says that an item i belongs to $\mathtt{cl}_f(X)$ iff each preserving function p_j remains constant with the addition of i. We illustrate Definition 3 with few examples. Of course, $\mathtt{cl}_{freq(X,\mathcal{D})}$[1] exactly corresponds to the Galois closure operator h (cf. Sect. 2.2). We can also consider new condensable functions such as min. We get $\mathtt{cl}_{min(X.val)}(X) = \{i \in \mathcal{I} | min(X.val) = min(X \cup \{i\}.val)\}$. From Fig. 1, as an example, one has $\mathtt{cl}_{min(X.val)}(BDE) = ABCDEF$ because the addition of any item does not modify the value $min(BDE.val)$.

Theorem 1 indicates that $\mathtt{cl}_f$ is adequate to f:

Theorem 1 (Adequate property) *Let f be a condensable function and X be an itemset, one has $f(X) = f(\mathtt{cl}_f(X))$.*

Proof Let $X \in \mathcal{L}_\mathcal{I}$ and $n \geq 0$ such that $\mathtt{cl}_f(X) = X \cup \{i_1, \ldots, i_n\}$. If $n = 0$, one has $\mathtt{cl}_f(X) = X$ and then $f(X) = f(\mathtt{cl}_f(X))$. Assuming that for any $n > 0$, we have $f(X) = f(X \cup \{i_1, \ldots, i_n\})$. As $i_{n+1} \in \mathtt{cl}_f(X)$, we have $p_j(X \cup \{i_{n+1}\}) = p_j(X)$ for each $j \in \{1, \ldots, k\}$ (Definition 3) and $f(X \cup \{i_{n+1}\}) = F(p_1(X \cup \{i_{n+1}\}), \ldots, p_k(X \cup \{i_{n+1}\})) = F(p_1(X), \ldots, p_k(X)) = f(X)$. Moreover, $X \cup \{i_1, \ldots, i_n\}$ is a specialization of X and Definition 1 gives that $p_j(X \cup \{i_1, \ldots, i_n\}) = p_j(X \cup \{i_1, \ldots, i_{n+1}\})$ for each $j \in \{1, \ldots, k\}$. We straightforwardly obtain that $f(X \cup \{i_1, \ldots, i_n\}) =$

[1] To alleviate the notation, $\mathtt{cl}_{f(X)}$ refers to $\mathtt{cl}_{X \mapsto f(X)}$.

$f(X \cup \{i_1, \ldots, i_{n+1}\})$. As $f(X)$ equals $f(X \cup \{i_1, \ldots, i_n\})$ by hypothesis, we obtain that $f(X) = f(X \cup \{i_1, \ldots, i_{n+1}\})$. By induction, we conclude that Theorem 1 is correct. □

This theorem ensures us that $\mathtt{cl}_f$ is adequate to the function f. It is also crucial for proving Theorem 2 and justifying the adequacy of the condensed representations of our generic framework. The following example illustrates the practical impact of the link between f and $\mathtt{cl}_f$. From Fig. 1, we have $min(BDE.val) = min(\mathtt{cl}_{min}(BDE).val) = min(ABCDEF.val) = 10$. However, we note that $ABCDEF$ is not present in $\mathcal{D}$. More generally, $\mathtt{cl}_{min}$ is not adequate to the frequency (i.e., $freq(X, \mathcal{D}) \neq freq(\mathtt{cl}_{min}(X), \mathcal{D})$). We can jointly consider the functions min and $freq$ for handling this phenomenon (this is formulated by the function f_3 in Table 1). Then, for the itemset BDE, we obtain that $\mathtt{cl}_{f_3}(BDE) = ABCDE$. We still have $min(BDE.val) = min(ABCDE.val) = 10$, but $freq(BDE, \mathcal{D}) = freq(ABCDE, \mathcal{D}) = 1$.

Given a condensable function f, we now prove that $\mathtt{cl}_f$ is a closure operator:

Theorem 2 (Closure operator) *Let f be a condensable function, $\mathtt{cl}_f$ is a closure operator.*

Proof Extensive: Let $X \in \mathcal{L}_\mathcal{I}$ and $i \in X$. As we have $p_j(X \cup \{i\}) = p_j(X)$ for each $j \in \{1, \ldots, k\}$, we have $i \in \mathtt{cl}_f(X)$. Idempotent: Let $X \in \mathcal{L}_\mathcal{I}$ and $i \in \mathtt{cl}_f(\mathtt{cl}_f(X))$. One has $p_j(\mathtt{cl}_f(X) \cup \{i\}) = p_j(\mathtt{cl}_f(X))$ for each $j \in \{1, \ldots, k\}$ (see Definition 3). As $p_j(\mathtt{cl}_f(X)) = p_j(X)$ (see the proof of Theorem 1), we straightforwardly deduce that $p_j(X \cup \{i\}) = p_j(\mathtt{cl}_f(X) \cup \{i\})$ by a similar induction done in the proof of Theorem 1. Monotonically increasing: Let $X \subseteq Y$ and $i \in \mathcal{I}$ such that $i \in \mathtt{cl}_f(X)$. First, we have $p_j(X \cup \{i\}) = p_j(X)$ for each $j \in \{1, \ldots, k\}$ (Definition 3). As each function p_j is preserving (Definition 1), we obtain that $p_j(Y \cup \{i\}) = p_j(Y)$ because Y is a specialization of X. Thus, the item i belongs to $\mathtt{cl}_f(Y)$. □

The proof of this theorem clearly highlights the great role of Definition 1. The property given by the definition of preserving functions ensures that $\mathtt{cl}_f$ monotonically increases and then provides a closure operator. Theorem 2 enables us to use well-known properties on equivalence classes designed by a closure operator in order to define and mine condensed representations adequate to condensable functions as we will see in the next section.

4 Mining adequate condensed representations

4.1 Definition of adequate condensed representations

This section defines condensed representations adequate to condensable functions, so-called adequate condensed representations. The principle is similar to the one performed by the usual closure-based condensed representations, except that the adequate closure operator $\mathtt{cl}_f$ is used (i.e., given a condensable function f, an itemset X is in the same equivalence class as an itemset Y iff $\mathtt{cl}_f(X) = \mathtt{cl}_f(Y)$). All the

itemsets belonging to the same equivalence class have the same value for f (Theorem 1). As for usual closure-based condensed representations, we have to select at least one itemset in each equivalence class, typically the free itemsets (i.e., generators) or closed itemsets (cf. Sect. 2.2). We then extend the usual definitions of the free and closed itemsets stemmed from h to the adequate closure operator:

Definition 4 (*Adequate free and closed itemsets*) Let $f = F(p_1, \ldots, p_k)$ be a condensable function, an itemset X is a free (resp. closed) itemset adequate to f iff for each $Y \subset X$ (resp. $X \subset Y$), one has $\mathtt{cl}_f(Y) \neq \mathtt{cl}_f(X)$.

In other words, an adequate free/closed itemset is an itemset having its neighbors (immediate generalizations/specializations) with a different value for at least one preserving function p_j. Thus, we can straightforwardly deduce that an itemset X is an adequate closed itemset iff $\mathtt{cl}_f(X) = X$: an adequate closed itemset is the maximal element of its equivalence class. For instance, $ABCDE$ is not a closed itemset adequate to min (because $\mathtt{cl}_{min}(ABCDE) = ABCDEF$) and then, $ABCDEF$ is a closed itemset adequate to min. On the contrary, an adequate free itemset is a minimal element of its equivalence class. For example, BD is a free itemset adequate to f_3 because there is no smaller itemset (B or D) having simultaneously the same frequency and the same value for min.

Following on, we focus on the condensed representations based on adequate closed itemsets (note that our algorithm proposed in next section also extracts all the adequate free itemsets). We exploit a property of the closure operator for a selector, i.e. a function s: $\mathcal{L}_\mathcal{I} \rightarrow \mathcal{R}$ (which maps any itemset $X \in \mathcal{L}_\mathcal{I}$ to its representative $s(X)$ in the condensed representation $\mathcal{R}$). An adequate condensed representation gathers all the adequate closed itemsets:

Theorem 3 (Adequate condensed representation) *Let f be a condensable function, the set of all the closed itemsets adequate to f, denoted by $\mathcal{RC}_f$, is a condensed representation adequate to f. The function $sc_f : X \mapsto min_\subseteq\{Y \in \mathcal{RC}_f | X \subseteq Y\}$ is a selector of this representation:*

$$\forall X \in \mathcal{L}_\mathcal{I},\ sc_f(X) \in \mathcal{RC}_f \text{ and } f(X) = f(sc_f(X))$$

Proof Let f be a condensable function. We have $\mathcal{RC}_f = \{X \in \mathcal{L}_\mathcal{I} | \forall Y \supset X, \mathtt{cl}_f(Y) \neq \mathtt{cl}_f(X)\}$. Let X be an itemset and $Y = sc_f(X)$. First, one has $X \subseteq \mathtt{cl}_f(X)$ because $\mathtt{cl}_f$ is idempotent. Second, by definition of sc_f, we have $X \subseteq Y$ and then, $\mathtt{cl}_f(X) \subseteq \mathtt{cl}_f(Y)$ because $\mathtt{cl}_f$ monotonically increases. By definition of $\mathcal{RC}_f$, Y is an adequate closed itemset: $Y = \mathtt{cl}_f(Y)$. Thereby, one has $X \subseteq \mathtt{cl}_f(X) \subseteq Y$. As $\mathtt{cl}_f(X)$ is a closed itemset adequate to f, $\mathtt{cl}_f(X)$ belongs to $\mathcal{RC}_f$. We deduce that $Y = \mathtt{cl}_f(X)$ because Y is the minimal closed superset of X. Finally, we obtain that $sc_f(X) = \mathtt{cl}_f(X)$ and Theorem 1 gives $f(X) = f(\mathtt{cl}_f(X))$. Thus, we conclude that $\mathcal{RC}_f$ with the selector sc_f is a condensed representation adequate to f. □

This theorem is significant because it ensures to get a condensed representation. We will see in the experiments that the sizes of adequate condensed representations

Table 2 $\mathcal{RC}_{f_3}$: adequate closed itemsets and their f_3 values

$X \in \mathcal{RC}_{f_3}$	$f_3(X)$	$X \in \mathcal{RC}_{f_3}$	$f_3(X)$	$X \in \mathcal{RC}_{f_3}$	$f_3(X)$	$X \in \mathcal{RC}_{f_3}$	$f_3(X)$
A	(50, 4)	F	(15, 2)	CF	(15, 1)	$ABCD$	(10, 2)
C	(75, 3)	AB	(30, 3)	DE	(10, 2)	$ABCE$	(30, 1)
D	(10, 3)	AC	(50, 2)	ABC	(30, 2)	$ABEF$	(15, 1)
E	(30, 4)	AE	(30, 3)	ABE	(30, 2)	$ABCDE$	(10, 1)

are smaller (and, in general, much smaller) than the whole collection of itemsets. On our pedagogical (and small) example, the condensed representation based on closed itemsets adequate to f_3 (i.e., according to min and $freq$) coming from the data in Fig. 1 is the set $\mathcal{RC}_{(min,freq)} = \mathcal{RC}_{f_3}$ given in Table 2.

These 16 adequate closed itemsets summarize both the frequency and the min value of the 40 itemsets present in the dataset $\mathcal{D}$. Moreover, the selector $sc_f(X)$ returns the minimal closed itemset adequate to f and including X. This selector is important because it does not require to get back to the dataset. For instance, $sc_{f_3}(BDE)$ directly returns its representative $ABCDE$ which also corresponds to $\mathtt{cl}_{f_3}(BDE)$ in the dataset $\mathcal{D}$.

4.2 MicMac: an algorithm to mine adequate condensed representations

MicMac (MIning ADequate Condensed representations) is a levelwise algorithm which produces the condensed representation adequate to a condensable function. More precisely, given a dataset $\mathcal{D}$ and a condensable function f, MicMac returns all the adequate free itemsets, each adequate free itemset X being associated with its adequate closed itemset $\mathtt{cl}_f(X)$ and its value $f(X)$.

Before detailing the algorithm, we give a property speeding up the mining of adequate condensed representations. Indeed, Property 1 shows that the freeness adequate to f is an anti-monotone constraint and thus MicMac can benefit from the monotonicity property to prune the search space, since MicMac follows the framework of the levelwise algorithms.[2]

Property 1 (Anti-monotonicity of adequate freeness) *Let f be a condensable function, the freeness adequate to f is an anti-monotone constraint.*

Proof Let f be a condensable function. Let $X \in \mathcal{L}_{\mathcal{I}}$ be a non-free itemset w.r.t. f. Let Y be specialization of X (i.e., $Y \supseteq X$). There exists $Z \subset X$ such that $\mathtt{cl}_f(Z) = \mathtt{cl}_f(X)$ because X is not free. One has $\mathtt{cl}_f(Z \cup (Y \backslash X)) = \mathtt{cl}_f(\mathtt{cl}_f(Z) \cup (Y \backslash X))$ (idempotent and monotonically increasing properties of $\mathtt{cl}_f$). As $\mathtt{cl}_f(Z) = \mathtt{cl}_f(X)$, we obtain that $\mathtt{cl}_f(\mathtt{cl}_f(Z) \cup (Y \backslash X)) = \mathtt{cl}_f(\mathtt{cl}_f(X) \cup (Y \backslash X))$. Idempotent and monotonically increasing properties of $\mathtt{cl}_f$ gives $\mathtt{cl}_f(\mathtt{cl}_f(X) \cup (Y \backslash X)) = \mathtt{cl}_f(X \cup$

[2] Let us recall that a constraint q is anti-monotone iff whenever X satisfies q, any generalization of X (i.e., $Y \subseteq X$) also satisfies q. Such constraints provide powerful pruning conditions in the search space (Mannila and Toivonen 1997).

Algorithm 1 MicMac

Input: A condensable function f, an anti-monotone constraint q_{AM} and a dataset $\mathcal{D}$
Output: All the free itemsets adequate to f and satisfying q_{AM} with their closure adequate to f and f values

1: $\mathcal{C}and_1 := \mathcal{I}$
2: $i := 1$
3: **while** $\mathcal{C}and_i \neq \emptyset$ **do**
4: $\quad \mathcal{F}ree_i := \{X \in \mathcal{L}_{\mathcal{I}} \mid X \in \mathcal{C}and_i$ and X is a free itemset adequate to f and satisfies $q_{AM}\}$
5: $\quad \mathcal{C}and_{i+1} := \{X \in \mathcal{L}_{\mathcal{I}} \mid \forall Y \subset X,$ one has $Y \in \bigcup_{j \leq i} \mathcal{F}ree_j\} \setminus \bigcup_{j \leq i} \mathcal{C}and_j$
6: $\quad i := i + 1$
7: **od**
8: **return** $\{(X, \mathtt{cl}_f(X), f(X)) \mid X \in \bigcup_{j<i} \mathcal{F}ree_j\}$

$(Y \setminus X)) = \mathtt{cl}_f(Y)$. Thus, Y is not a free itemset adequate to f and we conclude that Property 1 is correct. □

In other words, whenever X is a free itemset adequate to f, all the subsets of X are also free itemsets adequate to f. As an example, let us consider f_3. Since the itemset ABD is not free adequate to f_3 (because $\min(BD.val) = \min(ABD.val)$ and $freq(BD, \mathcal{D}) = freq(ABD, \mathcal{D})$), we are sure that $ABDC$, $ABDE$ and $ABDCE$ are not free. Experiments show that Property 1 really improves the mining of adequate condensed representations.

MicMac algorithm (see Algorithm 1) is clearly inspired from Close (Pasquier et al. 1999) and MinEx (Boulicaut et al. 2003). It demonstrates that closed itemsets mining algorithms can be easily adapted to the new closure operator $\mathtt{cl}_f$. As already said, MicMac is a levelwise algorithm (Mannila and Toivonen 1997) benefiting from the anti-monotone property of the adequate freeness (Property 1). In addition, the user can specify another anti-monotone constraint q_{AM} (e.g., the minimal frequency constraint) which is often useful in real-world applications.

Now we briefly detail this generate-and-test algorithm where the sets $\mathcal{C}and_i$ (resp. $\mathcal{F}ree_i$) contain all the candidates (resp. adequate free itemsets) of cardinality i. Line 1 initializes the candidates of length 1 (i.e., items). While there are candidates of length i, Line 4 computes all the adequate free itemsets of length i satisfying the constraint q_{AM} (test step). Line 5 generates the new candidates of length $i + 1$ (generate step). Finally, Line 8 returns the complete collection of the adequate free itemsets (with the corresponding adequate closed itemsets and values for f).

The following theorem proves the soundness and the correctness of MicMac:

Theorem 4 MicMac *algorithm is sound and correct.*

Proof The conjunction of two anti-monotone constraints (i.e., adequate freeness and q_{AM}) being again anti-monotone, levelwise algorithm (Mannila and Toivonen 1997) guarantees that all the free itemsets adequate to f and satisfying q_{AM} are extracted. As the algorithm also returns the adequate closure of all the free itemsets and their value f, MicMac is sound and correct. □

5 Experimental study

The aim of our experiments is to quantify the benefit brought by the adequate condensed representations both on conciseness and run-time. For this purpose, we conducted experiments on benchmarks coming from the UCI repository[3] with various condensable functions. All the tests were performed on a 3 GHz Xeon processor with Linux operating system and 1GB of RAM memory. The used prototype is an implementation of MicMac.

We consider all the condensable functions listed in Table 1 except for f_2 and f_{10}. Note that some condensable functions lead to identical results (same size and extraction run-time). Indeed, two condensable functions with the same set of preserving functions have the same adequate closure operator (i.e., $(F_1(p_1, \ldots, p_k)$ and $F_2(p_1, \ldots, p_k))$ $\Rightarrow (\mathtt{cl}_{F_1(p_1,\ldots,p_k)} = \mathtt{cl}_{F_2(p_1,\ldots,p_k)})$). So we gather results on functions f_1, f_4, f_5, f_7 and f_8 because these functions are defined from the same set of preserving functions (here, $freq$).

In the following, condensable functions using numeric values were applied on attribute values (noted *val* in the definitions of functions) randomly generated within the range [1,100]. Experiments were carried out on 21 benchmarks (given in Table 3) having various dimensions and density.

5.1 Conciseness of the adequate condensed representations

This section shows the conciseness of the adequate condensed representations with regard to the whole collection of itemsets. First, we compare the number of adequate closed itemsets (i.e., the number of itemsets of the adequate condensed representation) to the number of itemsets present at least once in the dataset (i.e., $freq(X, \mathcal{D}) \geq 1$). It means that we need to know the number of the whole collection of itemsets and we use Eclat (Zaki 2000b) for this purpose. We chose the prototype implemented by Borgelt and available on FIMI repository[4] because it is renowned for its effectiveness (Goethals and Zaki 2003). Nevertheless, the total numbers of itemsets on `mushroom` and `sick` remain unknown because the extractions performed by Eclat failed. Results are given in Table 3.

Obviously, the number of itemsets of an adequate condensed representation is always lower than the total number of itemsets. More interestingly, the adequate condensed representation are several orders of magnitude smaller than the total number of itemsets (10^3 for `horse`, more than 10^2 for `german`, `hepatic`, `vehicle`, etc.). Thus adequate condensed representation are concise. The most important gains of compression are obtained on the largest collections of itemsets. Another result is that the compression gain is approximatively the same for all the condensable functions, including condensable functions only based on the frequency as preserving function and therefore using the Galois closure as adequate closure operator. It means that, on the one hand, the benefit of the adequate condensed representations is similar to

[3] www.ics.uci.edu/~mlearn/MLRepository.html.

[4] http://fimi.cs.helsinki.fi/data/.

Table 3 Size of the adequate condensed representations and run-time comparison on UCI benchmarks

Dataset	Number of itemsets					Run-time (s)				
	$\mathcal{RC}_{f_i}$				All	MicMac				Eclat
	$f_{1,4,5,7,8}$	f_3	f_9	f_6		$f_{1,4,5,7,8}$	f_3	f_9	f_6	
Abalone	24 k	26 k	27 k	29 k	57 k	0.67	0.77	0.82	0.96	**0.22**
Anneal	13 k	20 k	32 k	51 k	3,041 k	**0.63**	**0.98**	**1.66**	**2.12**	4.62
Austral	193 k	256 k	329 k	448 k	11,511 k	**12.96**	**17.24**	19.00	25.68	17.70
Cleve	85 k	115 k	104 k	151 k	3,030 k	4.11	6.35	5.45	8.25	**3.82**
Cmc	41 k	58 k	53 k	75 k	176 k	0.91	1.50	1.32	2.24	**0.32**
Crx	186 k	232 k	336 k	430 k	23,239 k	**20.77**	**25.55**	**29.45**	**34.02**	36.92
German	4,857 k	6,281 k	8,144 k	10,838 k	1,665M	**440.10**	**588.74**	**768.24**	**1043.46**	3105.12
Glass	7 k	10 k	8 k	13 k	212 k	**0.16**	**0.25**	**0.24**	0.42	0.28
Heart	81 k	94 k	104 k	126 k	2,490 k	3.76	5.34	5.22	6.11	**3.16**
Hepatic	125 k	251 k	157 k	333 k	82,881 k	**10.18**	**21.44**	**13.02**	**26.97**	120.86
Horse	152 k	226 k	234 k	375 k	286M	**18.48**	**26.63**	**27.97**	**43.91**	437.44
Hypo	947 k	1,568 k	1,023 k	1,734 k	113M	**311.25**	376.60	335.09	398.32	319.82
Lymp	50 k	135 k	66 k	188 k	57,571 k	**3.44**	**10.94**	**5.00**	**13.49**	82.06
Mushroom	239 k	706 k	481 k	1,335 k	?	**53.05**	**116.27**	**99.55**	**215.37**	fail
Page	46 k	59 k	58 k	77 k	390 k	1.14	1.57	1.55	2.20	**0.62**
Pima	27 k	29 k	31 k	35 k	78 k	0.58	0.73	0.76	0.89	**0.12**
Sick	4,713 k	6,142 k	6,176 k	8,814 k	?	**2261.02**	**2434.56**	**2371.44**	**2616.81**	fail
Tic-tac-toe	60 k	68 k	68 k	78 k	251 k	1.36	1.96	1.96	2.34	**0.39**
Vehicle	453 k	754 k	752 k	1,344 k	209M	**50.20**	**72.47**	**79.35**	**123.03**	372.43
Wine	27 k	43 k	38 k	68 k	2,034 k	**1.60**	2.46	**2.33**	3.97	2.44
Zoo	5 k	11 k	8 k	21 k	4,122 k	**0.27**	**0.68**	**0.42**	**1.20**	5.47

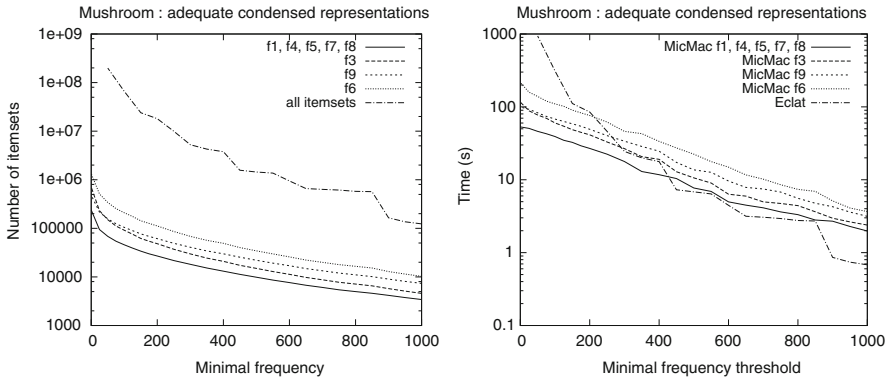

Fig. 2 Conciseness of adequate condensed representations (left) and run-time comparison between MicMac and Eclat (right) on `mushroom` according to γ

the one induced by the usual condensed representation of closed itemsets and, on the other hand, the adequate closure operators are suitable for all the condensable functions.

Let us have a look on the size of the adequate condensed representations according to the minimal frequency threshold γ. Figure 2 (left) plots the size of the adequate condensed representations on `mushroom` according to γ. Of course, the size of the adequate condensed representation increases when γ decreases because there are more frequent itemsets. But, the number of adequate closed itemsets increases more slowly than the number of itemsets (see the upper curve in Fig. 2: on `mushroom`, the compression gain is around 100 for the high values of γ whereas it reaches 1000 for the low values). The control of the increasing of the number of itemsets may be precious for searching rare itemsets.

5.2 Efficiency of MicMac

This section aims at measuring the run-time benefit brought by the adequate condensed representations w.r.t. Eclat. Keep in mind that Eclat mines all the itemsets present in the data and thereby its run-time is the same for all the condensable functions. A post-processing step is required to compute the values of the condensable functions on each pattern. The run-time of this step is low and it is not reported in our results. On the contrary, MicMac directly provides the values of the condensable functions on patterns and, obviously, produces only the patterns of the condensed representations (as said below, it also enables us in a straightforward way the uses of the condensed representations).

Run-times are displayed in Table 3. Best run-times are written in bold. Eclat failed twice due to a lack of memory whereas MicMac succeeded. First, MicMac run-times for mining the different adequate condensed representations were quite similar. Most of the time, MicMac is simultaneously better (or worst) than Eclat for all the condensable functions. Second, the global behavior of MicMac is very good: MicMac achieves the best running times on 10 datasets, Eclat on 7 datasets and results

are mixed on 4 datasets. It is very important to recall that the approach performed by MicMac is generic whereas Eclat only extracts frequent patterns. With MicMac, the user can address a broad set of condensable functions (even complex ones such as f_6 and not only measures based on (anti-)monotone properties). MicMac is able to mine adequate condensed representations in datasets which are intractable for this task with other techniques (e.g., `mushroom`, `sick`).

With regard to the role of the minimal frequency threshold γ, Fig. 2 (right) plots MicMac and Eclat run-times on `mushroom` according to γ. As expected, run-times increase when γ decreases because there are more frequent itemsets. More interestingly, these curves indicate that the higher compression gain, the more efficient MicMac w.r.t. Eclat is (we have seen in the previous section that the compression gain increases when γ decreases). When the adequate condensed representations are very concise, Property 1 drastically reduces the search space and thus improves the extraction. For this reason, MicMac appears to be very efficient with low frequency threshold on `mushroom`. This phenomenon (i.e., the effectiveness of the condensed representation approach increases according to the decreasing of γ) is also reported by the usual closed pattern mining methods (Goethals and Zaki 2003).

6 Conclusion

By proposing the new notion of *adequate condensed representation*, this paper extends the paradigm of condensed representations to a broad spectrum of functions including interestingness measures and constraints. This framework encompasses the current methods since existing closure-based condensed representations (e.g., disjunctive/conjunctive frequency, lift) correspond to specific closure operators of our framework. Experiments show that sizes of the adequate condensed representations are smaller (and, in general, much smaller) than the total number of itemsets. Besides, MicMac efficiently mines such condensed representations even in difficult datasets which are intractable for this task with other techniques.

We think that adequate condensed representations open a new research direction on discovering both various and significant patterns which may lead to promising applications. We particularly intend to exploit the powerful semantic of the adequate closure operators to mix the notions of utility-based functions and non-redundant association rules. For instance, in the context of market basket analysis, we can consider the rule $X \rightarrow Y$ with a minimal body X with low prices attached to X (i.e., X is a free itemset adequate to $max(X.price) \leq \gamma_1$) and a maximal head Y with high prices (i.e., $X \cup Y$ is a closed itemset adequate to $min((X \cup Y).price) \geq \gamma_2$), then such a rule may indicate that the purchase of cheap products leads to the purchase of expensive products. More generally, this direction suggests many exciting new applications for KDD similarly there are many uses stemming from the condensed representations based on frequency.

Acknowledgments This work is partly supported by the ANR (French Research National Agency) funded project Bingo2 ANR-07-MDCO-014.

References

Agrawal R, Srikant R (1994) Fast algorithms for mining association rules in large databases. In: Bocca JB, Jarke M, Zaniolo C (eds) VLDB'94, proceedings of 20th international conference on very large data bases. Morgan Kaufmann, pp 487–499

Bastide Y, Pasquier N, Taouil R, Stumme G, Lakhal L (2000) Mining minimal non-redundant association rules using frequent closed itemsets. In: Lloyd JW, Dahl V, Furbach U, Kerber M, Lau K-K, Palamidessi C, Pereira LM, Sagiv Y, Stuckey PJ (eds) Computational logic, vol 1861 of LNCS. Springer, pp 972–986

Birkhoff G (1967) Lattices theory, vol 25. American Mathematical Society

Boulicaut J-F, Bykowski A, Rigotti C (2003) Free-sets: a condensed representation of boolean data for the approximation of frequency queries. Data Min Knowl Discov 7(1):5–22. Kluwer Academics Publishers

Bykowski A, Rigotti C (2003) DBC: a condensed representation of frequent patterns for efficient mining. Inf Syst 28(8):949–977

Calders T, Goethals B (2002) Mining all non-derivable frequent itemsets. In: Proceedings of the 6th European conference on principles of data mining and knowledge discovery (PKDD'02), pp 74–85

Calders T, Goethals B (2003) Minimal k-free representations of frequent sets. In: Proceedings of the 7th European conference on principles and practice of knowledge discovery in databases (PKDD'03), Springer, pp 71–82

Calders T, Rigotti C, Boulicaut J-F (2004) A survey on condensed representations for frequent sets. In: Boulicaut J-F, Raedt LD, Mannila H (eds) Constraint-based mining and inductive databases, European workshop on inductive databases and constraint based mining, vol 3848 of LNCS. Springer, pp 64–80

Casali A, Cicchetti R, Lakhal L (2005) Essential patterns: a perfect cover of frequent patterns. In: Tjoa AM, Trujillo J (eds) Data warehousing and knowledge discovery, 7th international conference, DaWaK 2005, proceedings, vol 3589 of LNCS. Springer, pp 428–437

Crémilleux B, Boulicaut JF (2002) Simplest rules characterizing classes generated by delta-free sets. In: 22nd international conference on knowledge based systems and applied artificial intelligence, pp 33–46

Gasmi G, Yahia SB, Nguifo EM, Bouker S (2007) Extraction of association rules based on literalsets. In: Song IY, Eder J, Nguyen TM (eds) Data warehousing and knowledge discovery, 9th international conference, DaWaK 2007, proceedings, vol 4654 of LNCS. Springer, pp 293–302

Giacometti A, Laurent D, Diop CT (2002) Condensed representations for sets of mining queries. In: Knowledge discovery in inductive databases, 1st international workshop, KDID 2002

Goethals B, Zaki MJ (eds) (2003) FIMI '03, frequent itemset mining implementations, proceedings, vol 90 of CEUR workshop proceedings. http://CEUR-WS.org

Imielinski T, Mannila H (1996) A database perspective on knowledge discovery. Commun ACM 39(11): 58–64

Kryszkiewicz M (2005) Generalized disjunction-free representation of frequent patterns with negation. J Exp Theor Artif Intell 17(1–2):63–82

Li J, Liu G, Wong L (2007) Mining statistically important equivalence classes and delta-discriminative emerging patterns. In: Berkhin P, Caruana R, Wu X (eds) KDD. ACM, pp 430–439

Mannila H, Toivonen H (1997) Levelwise search and borders of theories in knowledge discovery. Data Min. Knowl. Discov 1(3):241–258

Mitchell TM (1982) Generalization as search. Artif Intell 18(2):203–226

Morik K, Boulicaut J-F, AS (eds) (2005) Local pattern detection, vol 3539 of LNAI. Springer-Verlag

Ng RT, Lakshmanan LVS, Han J, Pang A (1998) Exploratory mining and pruning optimizations of constrained association rules. In: Haas LM, Tiwary A (eds) SIGMOD 1998, proceedings ACM SIGMOD international conference on management of data. ACM Press, pp 13–24

Pasquier N, Bastide Y, Taouil R, Lakhal L (1999) Discovering frequent closed itemsets for association rules. In: Database theory—ICDT '99, 7th international conference, proceedings, vol 1540 of LNCS. Springer, pp 398–416

Soulet A, Crémilleux B, Rioult F (2004) Condensed representation of EPs and patterns quantified by frequency-based measures. In: Post-proceedings of knowledge discovery in inductive databases, 3rd international workshop, KDID 2004, Pise, Springer

Yao H, Hamilton HJ, Butz CJ (2004) A foundational approach to mining itemset utilities from databases. In: Berry MW, Dayal U, Kamath C, Skillicorn DB (eds) Proceedings of the fourth SIAM international conference on data mining

Zaki MJ (2000a) Generating non-redundant association rules. In: KDD, pp 34–43

Zaki MJ (2000b) Scalable algorithms for association mining. IEEE Trans Knowl Data Eng 12(3):372–390