Finding Robust Itemsets Under Subsampling

NIKOLAJ TATTI, HIIT, Aalto University, KU Leuven, University of Antwerp
FABIAN MOERCHEN, Amazon.com Inc
TOON CALDERS, Université Libre de Bruxellés, Eindhoven University of Technology

Mining frequent patterns is plagued by the problem of pattern explosion making pattern reduction techniques a key challenge in pattern mining. In this paper we propose a novel theoretical framework for pattern reduction. We do this by measuring the robustness of a property of an itemset such as closedness or non-derivability. The robustness of a property is the probability that this property holds on random subsets of the original data. We study four properties: an itemset being closed, free, non-derivable or totally shattered, and demonstrate how to compute the robustness analytically without actually sampling the data. Our concept of robustness has many advantages: Unlike statistical approaches for reducing patterns, we do not assume a null hypothesis or any noise model and in contrast to noise tolerant or approximate patterns, the robust patterns for a given property are always a subset of the patterns with this property. If the underlying property is monotonic, then the measure is also monotonic, allowing us to efficiently mine robust itemsets. We further derive a parameter-free technique for ranking itemsets that can be used for top-k approaches. Our experiments demonstrate that we can successfully use the robustness measure to reduce the number of patterns and that ranking yields interesting itemsets.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Data Mining

General Terms: Algorithms, Experimentation, Theory

Additional Key Words and Phrases: pattern reduction, robust itemsets, closed itemsets, free itemsets, non-derivable itemsets, totally shattered itemsets

ACM Reference Format:
ACM Trans. Datab. Syst. V, N, Article A (January YYYY), 27 pages.
DOI = 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

The research described in this paper builds upon and extends the work presented in the IEEE International Conference on Data Mining (ICDM IEEE 2011) [Tatti and Moerchen 2011].
Part of this work done while Nikolaj Tatti was employed by ADReM Research Group, Department of Mathematics and Computer Science, University of Antwerp and DTAI group, Department of Computer Science, Katholieke Universiteit Leuven, Leuven, Belgium. In addition, Fabian Moerchen was employed by Siemens Corporation , USA and Toon Calders was employed by Faculty of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands. Nikolaj Tatti was partly supported by a Post-Doctoral Fellowship of the Research Foundation — Flanders (FWO).
Authors' address: N. Tatti, Helsinki Insitute for Information Technology, Department of Information and Computer Science, Aalto University, Finland. F. Moerchen, Amazon.com Inc, 410 Terry Avenue North, Seattle, WA, USA. T. Calders, WIT group, Computer & Decision Engineering department, Université Libre de Bruxellés, Belgium

DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

1. INTRODUCTION

Frequent itemset mining was first introduced in the context of market basket analysis [Agrawal et al. 1993]. This problem can be defined as follows: a transaction is a subset of a given set of items A, and a transaction database is a set of such transactions. A subset X of A is a *frequent itemset* in a transaction database if the number of transactions containing all items of X exceeds a given threshold. Since its proposal, frequent itemset mining has been used to address many data mining problems such as association rule generation [Hipp et al. 2000], clustering [Wang et al. 1999], classification [Cheng et al. 2007], temporal data mining [Moerchen et al. 2010] and outlier detection [Smets and Vreeken 2011]. The mining of itemsets is a core step in these methods that often dominates the overall complexity of the problem. The number of frequent itemsets, however, can be extremely large even for moderately sized datasets; in worst case, the number of frequent itemsets is exponential in $|A|$. This explosion severely complicates manual analysis or further automated processing steps.

Therefore, researchers have proposed many solutions to reduce the number of patterns depending on the context in which the patterns are to be used or the process in which the data was generated. Examples of reduced pattern collections include: closed itemsets [Pasquier et al. 1999] to avoid redundant association rules, constrained itemsets [Pei et al. 2001], condensed representations [Calders et al. 2006] to answer frequency queries with limited memory, margin-closed itemsets [Moerchen et al. 2010] for exploratory analysis, and surprising itemsets [Brin et al. 1997; Tatti 2008] or top-k tiles [Geerts et al. 2004].

Many of reduction techniques have a drawback of being fragile. For example, a closed itemset can be defined as an itemset that can be written as the intersection of transactions; that is, all of its supersets are contained in strictly less transactions. Given a non-closed itemset X, adding a single transaction to the dataset containing only X will make X closed. In this paper we introduce a novel theoretical framework that uses this drawback to its advantage. Given a property of an itemset (closedness or non-derivability, for example) we can measure the *robustness* of this property. A property of X is robust if it holds for many datasets subsampled from the original data. We demonstrate that we can compute this measure analytically for several important classes of itemsets: closed [Pasquier et al. 1999], free [Boulicaut et al. 2003], non-derivable [Calders and Goethals 2007], and totally shattered itemsets [Mielikäinen 2005]. Computing robust itemsets under subsampling turns out to be practical for free, non-derivable, and totally shattered itemsets. Unfortunately, for closed itemsets the test for robustness is prohibitively expensive.

A possible drawback of our approach is that it depends on a parameter α, the probability of including a transaction in a subsample. In addition to providing reasonable guidelines on how to choose α we also introduce a technique making us independent of α. We show that there is a neighborhood near 1 in which the ranking of itemsets does not depend on α. We further demonstrate how we can compute this ranking without actually discovering the exact neighborhood or computing the measure for the itemsets. We give exact solutions for free, non-derivable, and totally shattered itemsets and provide practical heuristics for closed itemsets.

In the remainder of this paper we describe related work in Section 2. Itemsets robust under subsampling and algorithms to find them are described in Section 3. We discuss ordering itemsets based on large values of α in Sections 4–5. Section 6 demonstrates how the subsampling approach can reduce the number of reported itemsets significantly. We conclude our paper with a discussion in Section 7.

2. RELATED WORK

The design goal of condensed representations [Calders et al. 2006] of frequent itemsets is to be able to answer all possible frequency queries. For example, non-derivable itemsets [Calders and Goethals 2007] exclude any itemset whose support can be derived exactly from the supports of its subsets using logical rules. Other examples of such complete collections are the closed and the free itemsets which are based upon the notion of equivalence of itemsets. Two itemsets are equivalent if they are supported by exactly the same set of transactions. This notion of equivalence divides the frequent itemsets into equivalence classes. The unique maximal element of each equivalence class is a closed itemset [Pasquier et al. 1999]. No more items can be added to this set without losing some supporting transactions. The not necessarily unique minimal elements of the equivalence class are free itemsets [Boulicaut et al. 2003] or generators. No items can be taken out without adding transactions to their support set. Complete condensed representations such as those based upon the non-derivable, closed, and free sets allow the derivation of the support of all frequent itemsets. Such representations are useful because they are more compact, yet they still support further mining tasks such as the generation of association rules where the frequencies of all subsets of an itemset are needed to determine the confidence of all possible rules.

Nevertheless, even the number of closed and free itemsets can still be very large when the minimum support threshold is low. As for other tasks knowing the frequency of all frequent itemsets may be less useful because there is a large redundancy in the set of frequent itemsets. By using approximate methods the number of patterns can be further reduced; for instance by clustering itemsets representing similar sets of transactions [Xin et al. 2005], enforcing itemsets to have a minimum margin of difference in support [Moerchen et al. 2010], or ranking itemsets by significance [Brin et al. 1997; Gallo et al. 2007; Webb 2007; Tatti 2008].

In fault tolerant approaches the strict definition of support, requiring all items of an itemset to be present in a transaction is relaxed, see [Gupta et al. 2008; Calders et al. 2007; Uno and Arimura 2007; Luccese et al. 2010]. Rather, it is assumed that items can be present or absent at random in the transactions. These approaches can reveal important structures in noisy data that might otherwise get lost in a huge amount of fragmented patterns. One needs to be aware though that they report approximate support values and possibly list itemsets that are not observed as such in the dataset at all or with much smaller support. Also the design goal is not to reduce the number of reported patterns. Only Cheng et al. [2006] considers the combination of the two approaches and studies closedness in combination with fault tolerance.

Furthermore, a third class of techniques considers a statistical null hypothesis and ranks patterns according to how much their support deviates from their expected support under the null model [Brin et al. 1997; Gallo et al. 2007; Webb 2007; Tatti 2008]. Unlike these approaches, we do not assume a statistical null hypothesis. We also do not assume any noise model, such as flipping the values of a matrix independently. Instead our goal is to study robustness of a given property based on subsampling transactions.

The idea of using random databases to assess data mining results has been proposed in [Gionis et al. 2007; Hanhijärvi et al. 2009; De Bie 2011]. The goal is to first infer some (simple) background information from a dataset, and then consider all datasets that have the same statistics. A data mining result is then deemed interesting only if it appears in a small number of these datasets. Interestingly enough, this is the opposite of what we are considering to be important; that is, we want to find itemsets that satisfy the predicate in many random subsets of the data. This philosophical difference can be explained by completely orthogonal randomizations. The authors in the aforementioned papers sample random datasets from simple statistics, that is, they ignore on purpose

complex interactions between items, and try to explain mining results with simple information. Our goal is not to explain results but rather to test whether our results are robust by testing how data mining results change if we remove transactions.

An idea using random datasets to compute the smoothness of results has been proposed by Misra et al. [2012]. The idea is to measure how stable the results are by sampling random datasets from a distribution that favors datasets close to the original one, and computing the average deviation from the original result in the sampled datasets. Finally, stability of rankings has been studied in the context of networks, see for example [Ghoshal and Barabási 2011].

3. ROBUST ITEMSETS

In this section we define the robustness and describe how to compute it efficiently.

3.1. Notation and definitions

We begin by reviewing the preliminaries and introducing the notations used in the paper.

A *binary dataset* D is a set of transactions, tuples (tid, t) consisting of a transaction id and a binary vector $t \in \{0,1\}^K$ of length K. The ith element of a transaction corresponds to an *item* a_i; a 1 in the ith position indicates that the transaction contains the item, a 0 that it does not. We denote the collection of all items by $A = \{a_1, \ldots, a_K\}$.

If S is a set of binary vectors of length K, we will write $D \cap S$ to denote $\{(tid, t) \in D \mid t \in S\}$.

An itemset X is a subset of A. Given a binary vector t of length K and an itemset X, we define t_X to be the binary vector of length $|X|$ obtained by keeping only the positions corresponding to the items in X.

Given an itemset $X = (x_1, \ldots, x_N)$ and a binary vector v of length N, we define the *support*

$$sp(X = v; D) = |\{(tid, t) \in D \mid t_X = v\}|$$

to be the number of transactions in D, where the items in X obtain the values given in v. We often omit D from the notation, when it is clear from the context. In addition, if v contains only 1s, we simply write $sp(X)$. Note that $sp(X)$ coincides with the traditional definition of a support for X. Discovering frequent itemsets, that is, itemsets whose support exceeds some given threshold is a well-studied problem.

Example 3.1. Throughout the paper we will use the following dataset D as a running example:

$$D = \begin{bmatrix} 1\colon 0\ 0\ 0\ 0\ 1 \\ 2\colon 0\ 1\ 0\ 1\ 1 \\ 3\colon 1\ 1\ 1\ 1\ 1 \\ 4\colon 0\ 1\ 0\ 1\ 1 \\ 5\colon 1\ 1\ 1\ 1\ 1 \\ 6\colon 1\ 0\ 0\ 0\ 0 \end{bmatrix}$$

D contains 5 items, a, b, c, d, and e, and 6 transactions. For this dataset we have $sp(ab) = 2$, and $sp(ab = [1,0]) = 1$.

We say that a function f mapping an itemset X to a real number $f(X)$ is *monotonically decreasing* if for each $Y \subseteq X$ we have $f(Y) \geq f(X)$. A classic pattern mining task is to discover all itemsets of having $f(X) \geq \rho$ given a threshold ρ and a function f mapping an itemset to a real number. If this function turns out to be monotonically decreasing, then we can use efficient pattern mining algorithms to discover *all* patterns satisfying this criterion.

Our next step is to define 4 different properties for itemsets. These are closed, free, non-derivable, and totally shattered itemsets. The goal of this work is to study how to introduce a measure of robustness for these properties.

Closed Itemsets. An itemset X is said to be *closed*, if there is no $Y \supsetneq X$ such that $sp(X) = sp(Y)$, i.e., X is maximal w.r.t. set inclusion among the itemsets having the same support. We define a predicate

$$\sigma_c(X; D) = \begin{cases} 1 & \text{if } X \text{ is closed in } D, \\ 0 & \text{otherwise} \quad . \end{cases}$$

Every closed itemset corresponds to the intersection of a subset of transactions in D and vice versa.

Free Itemsets. An itemset X said to be *free* if there is no $Y \subsetneq X$ such that $sp(X) = sp(Y)$, i.e., free itemsets are minimal among the itemsets having the same support. We define a predicate

$$\sigma_f(X; D) = \begin{cases} 1 & \text{if } X \text{ is free in } D, \\ 0 & \text{otherwise} \quad . \end{cases}$$

A vital property of free itemsets is that they constitute a downward closed collection allowing efficient mining with an Apriori-style algorithm (see Theorem 1 in [Boulicaut et al. 2000]). That is, if an itemset X is free, all its subsets are free as well.

Example 3.2. The closed itemsets in our running example are a, e, bde, and $abcde$. On the other hand, the itemsets $\emptyset$, a, b, c, d, e, ab, ad, and ae are free.

Non-derivable Itemsets. An itemset X is said to be *derivable*, if we can derive its support from the supports of the proper subsets of X, otherwise an itemset is called *non-derivable*. We define a predicate

$$\sigma_n(X; D) = \begin{cases} 1 & \text{if } X \text{ is non-derivable in } D, \\ 0 & \text{otherwise} \quad . \end{cases}$$

Non-derivable itemsets form a downward closed collection (Corollary 3.4 in [Calders and Goethals 2007]), hence we can mine them using an Apriori-style approach.

Totally Shattered Itemsets. We say that an itemset X is *totally shattered* if $sp(X = v) > 0$ for all possible binary vectors v. In other words, every possible combination of values for X occur in D. Again, we define a predicate

$$\sigma_s(X; D) = \begin{cases} 1 & \text{if } X \text{ is totally shattered in } D, \\ 0 & \text{otherwise} \quad . \end{cases}$$

Totally shattered itemsets are related to the VC-dimension [Mielikäinen 2005],and we can show that a totally shattered itemset is always free and non-derivable (but not the other way around).

Example 3.3. Itemset ab in the running example is totally shattered. Itemset ac is non-derivable but not totally shattered because $sp(ac = [0, 1]) = 0$.

It is easy to see from the definition that totally shattered itemsets constitute a downward closed collection, hence they are easy to mine using an Apriori-style approach.

3.2. Measuring robustness

In this section we propose a measure of robustness for itemsets with a predicate σ. The idea is to sample random subsets from a given dataset and measure how often the predicate $\sigma(X)$ holds in a random dataset. Intuitively we consider an itemset robust if the predicate is true for many subsets of the database.

In order to define the measure formally, we first define a probability for a subset of D.

Definition 3.4. Given a binary dataset D, and a real number α, $0 \leq \alpha \leq 1$, we define a random dataset D_α obtained from D by keeping each transaction with probability α, or otherwise discarding it. More formally, let S be a subset of D. The probability of $D_\alpha = S$ is equal to

$$p(D_\alpha = S) = \alpha^{|S|}(1-\alpha)^{|D|-|S|} \quad . \tag{1}$$

We can now define the robustness of an itemset X as the probability of $\sigma(X)$ being true in a random dataset.

Definition 3.5. Given a binary dataset D, a real number α, and an itemset predicate σ, we define the robustness to be the probability that $\sigma(X; D_\alpha) = 1$, that is,

$$r(X; \sigma, D, \alpha) = p(\sigma(X; D_\alpha) = 1) = \sum_{\sigma(X;S)=1} p(D_\alpha = S) \quad .$$

For notational clarity, we will omit D and α when they are clear from the context.

Example 3.6. Consider itemset ab in our running example. Let $\alpha = 1/3$. Note that $sp(ab = [0,0]) = sp(ab = [1,0]) = 1$ and $sp(ab = [0,1]) = sp(ab = [1,1]) = 2$. In order for ab to still be totally shattered on a subset each of these supports needs to stay greater than zero. The probability of this event is equal to

$$1/3 \times 1/3 \times (1 - 2/3 \times 2/3) \times (1 - 2/3 \times 2/3) = 25/729,$$

because for the first two cases we need to sample the single transaction upholding the property and for the other two cases we need to make sure we do not skip both transactions we need to uphold the property.

Our main goal is to mine itemsets for which the robustness measure exceed some given threshold ρ, that is, find all itemsets for which $r(X; \sigma, D, \alpha) \geq \rho$.

In order to mine all significant patterns we need to show that the robustness measure is monotonically decreasing. This is indeed the case if the underlying predicate is monotonically decreasing.

PROPOSITION 3.7. *Let σ be a monotonically decreasing predicate. Then $r(X; \sigma, D, \alpha)$ is also monotonically decreasing.*

PROOF. Let Y and X be itemsets such that $Y \subset X$. Then

$$r(X; \sigma, D, \alpha) = \sum_{\sigma(X;S)=1} p(D_\alpha = S) \leq \sum_{\sigma(Y;S)=1} p(D_\alpha = S) = r(Y; \sigma, D, \alpha)\,,$$

which proves the proposition. □

As pointed out in Section 3.1, predicates for free, non-derivable, and totally shattered itemsets are monotonically decreasing. However, the predicate for closedness is not monotonically decreasing.

We will finish this section by considering how robustness depends on α. If we set $\alpha = 1$, then $r(X; \sigma, D, \alpha) = \sigma(X; D)$. Naturally, we expect that when we lower α, the robustness would decrease. This holds for predicates that satisfy a specific property.

Definition 3.8. We say that a predicate σ is *monotonic w.r.t. deletion* if for each itemset X, each dataset D, and each transaction $t \in D$ it holds that if $\sigma(X; D) = 0$, then $\sigma(X; D - t) = 0$.

PROPOSITION 3.9. *Let σ be a predicate monotonic w.r.t. deletion. Then $r(X; \sigma, D, \alpha) \leq r(X; \sigma, D, \beta)$, for $\alpha \leq \beta$.*

PROOF. We will prove the proposition by induction over $|D|$. The proposition holds trivially for $|D| = 0$. Assume that the theorem holds for $|D| = N$ and let D be a dataset with $|D| = N + 1$.

Fix $t \in D$ and define a new predicate $\sigma_t(X; S) = \sigma(X; S \cup \{t\})$, where S is a dataset. σ_t is monotonic w.r.t deletion. Otherwise, if there is a dataset S, a transaction $u \in S$ an itemset Y violating the monotonicity, then $S \cup \{t\}$, the same transaction u, and the itemset Y will violate the monotonicity for σ.

Moreover, since σ is monotonic w.r.t deletion, it holds that $\sigma(X; S) \leq \sigma_t(X; S)$. This in turns implies that

$$r(X; \sigma, S, \alpha) \leq r(X; \sigma_t, S, \alpha) \quad . \tag{2}$$

Let us write $D' = D - \{t\}$. Then we have,

$$\begin{aligned} r(X; \sigma, D, \alpha) &= (1 - \alpha) r(X; \sigma, D', \alpha) + \alpha r(X; \sigma_t, D', \alpha) \\ &\leq (1 - \beta) r(X; \sigma, D', \alpha) + \beta r(X; \sigma_t, D', \alpha) \\ &\leq (1 - \beta) r(X; \sigma, D', \beta) + \beta r(X; \sigma_t, D', \beta) \\ &= r(X; \sigma, D, \beta) \,, \end{aligned}$$

where the first inequality holds because of Equation 2 and the second inequality holds because of induction assumption. This proves the proposition. □

It turns out that all the predicates we considered in Section 3.1 are monotonic w.r.t. deletion.

PROPOSITION 3.10. *Predicates σ_c, σ_f, σ_n, and σ_s are monotonic w.r.t. deletion.*

In order to prove the case for non-derivable itemsets we will need the following technical lemma. We will also use this lemma later on.

LEMMA 3.11. *An itemset X is derivable if and only if there are two vectors v and w of length $|X|$ with v having odd number of* 0*s and w having even number of* 0*s such that $sp(X = v) = sp(X = w) = 0$.*

PROOF. Let O be the set of binary vectors of length $|X|$ having odd number of 0s and let E be the set of binary vectors of length $|X|$ having even number of 0s.

An alternative way of describing non-derivable itemsets is to compute the following quantities

$$u = sp(X) + \min_{x \in O} sp(X = x) \quad \text{and} \quad l = sp(X) - \min_{x \in E} sp(X = x) \quad .$$

We can show that $l \leq sp(X) \leq u$, and that both u and l can be computed from proper subsets of X with the inclusion-exclusion principle (see [Calders and Goethals 2007]). We also know that an itemset is derivable if and only if $u = l$ (see [Calders and Goethals 2007]). This is because we know then that $l = sp(X) = u$.

Let $v = \arg\min_{x \in O} sp(X = x)$ and $w = \arg\min_{x \in E} sp(X = x)$. This implies that $0 = u - l = sp(X = v) + sp(X = w)$, which proves the lemma. □

PROOF OF PROPOSITION 3.10. An itemset is not totally shattered if there is a binary vector v such that $sp(X = v; D) = 0$. This immediately implies that $sp(X = v; D - \{t\}) = 0$. Thus σ_s is monotonic w.r.t. deletion. Similarly, Lemma 3.11 implies that σ_n is monotonic w.r.t. deletion.

An itemset X is not free, if there is $x \in X$ such that there is no transaction $u \in D$ for which $u_x = 0$ and $u_y = 1$ for all $y \in X - \{x\}$. If this holds in D, then it holds for $D - \{t\}$. This makes σ_f monotonic w.r.t. deletion. Similarly, an itemset X is not closed, if there is $x \notin X$ such that there is no transaction $u \in D$ for which $u_x = 0$ and $u_y = 1$ for all $y \in X$. If this holds in D, then it holds for $D - \{t\}$. This makes σ_c monotonic w.r.t. deletion. □

Table I. Computational complexity of robustness and orders. Computing measures is explained in Section 3.3. Computing orders is explained in Section 4. K is the number of items, $|\mathcal{C}|$ is the number of frequent closed itemsets.

predicate	measure	order	order estimate
free	$O(\|X\|)$	$O(\|X\|)$	–
totally shattered	$O(2^{\|X\|})$	$O(2^{\|X\|})$	–
closed	$O(2^{K-\|X\|})$	$O(2^{K-\|X\|})$	$O(\|\mathcal{C}^2\|)$
non-derivable	$O(2^{\|X\|})$	$O(\|D\|2^{\|X\|})$	–

Example 3.12. The itemset bd is not closed because its superset bde is always observed when bd is observed. No matter which transaction we delete (one with or without bde) this will not change. Note, however, that bde can become non-closed if transactions 2 and 4 are deleted because then $abcde$ will have the same support of 2.

3.3. Computing the measure

In this section we demonstrate how to compute the robustness measure for the predicates. Computing the measure directly from the definition is impractical since D has $2^{|D|}$ different subsamples. It turns out that computing free, non-derivable, and totally shattered itemsets has practical formulas while the robustness measure for closed itemsets has no practical formulation (see Table I).

We will first demonstrate how to compute robustness for free and totally shattered itemsets. In order to do that we introduce the following function: Given an itemset X and a set of binary vectors $V \subseteq \{0,1\}^{|X|}$ we define

$$o(X, V, \alpha) = \prod_{v \in V} 1 - (1-\alpha)^{sp(X=v)} \quad .$$

Intuitively, $o(X, V, \alpha)$ denotes the probability of the following event: for every vector $v \in V$, $sp(X = v; D_\alpha) > 0$. Note that since every transaction can support at most one $X = v$, the events $sp(X = v; D_\alpha) > 0$ are independent from each other. Note that we can compute $o(X, V, \alpha)$ in $O(|V|)$ time. Our next step is to show that robustness for free itemsets can be expressed with with $o(X, V, \alpha)$ for a certain set of vectors V.

PROPOSITION 3.13. *Given an itemset X, let V be the set of $|X|$ vectors having $|X|-1$ ones and one 0. The robustness of a free itemset is $r(X; \sigma_f, \alpha) = o(X, V, \alpha)$.*

PROOF. Given an item $x \in X$, define an event $T_x = sp(X - \{x\}; D_\alpha) > sp(X; D_\alpha)$. X is still free in D_α if T_x is true for all $x \in X$. T_x is true if and only if D_α contains a transaction t with $t_x = 0$ and $t_y = 1$ for $y \in X - \{x\}$. There are $sp(X = v; D)$ such transactions, where $v \in V$ is the vector for which $v_x = 0$. $p(T_x)$ is the probability of not removing all these transactions, thus

$$p(T_x) = 1 - (1-\alpha)^{sp(X=v;D)} \quad .$$

Since each of these transaction is missing only one $x \in X$, there are no common transactions between different events T_x, making them independent. Thus, we can conclude $r(X; \sigma_f, \alpha) = \prod_{x \in X} p(T_x) = o(X, V, \alpha)$. □

A similar result also holds for totally shattered itemsets.

PROPOSITION 3.14. *Given an itemset X, let V be the set of all binary vectors of length $|X|$. The robustness of a totally shattered itemset is $r(X; \sigma_s, \alpha) = o(X, V, \alpha)$.*

PROOF. Given a binary vector $v \in V$, define an event $T_v = sp(X = v; D_\alpha) > 0$. X is still totally shattered in D_α if T_v is true for all $v \in V$. $p(T_v)$ is the probability of

not removing all these transactions, thus $p(T_v) = 1 - (1-\alpha)^{sp(X=v;D)}$. Again, since no transaction can contribute to different T_v being true, the random variables are independent and we obtain $r(X;\sigma_s,\alpha) = \prod_{v\in V} p(T_v) = o(X,V,\alpha)$. □

Note that the formula in Proposition 3.14 corresponds directly to Example 3.6.

Let us now consider non-derivable itemsets. The analytic formula is somewhat more complicated than for free or totally shattered itemsets, although, the principle remains exactly the same.

PROPOSITION 3.15. *Given an itemset X, let V be the set of binary vectors of length $|X|$ having odd number of ones. Similarly let W be the set of binary vectors of length $|X|$ having even number of ones. The robustness of a non-derivable itemset is*

$$r(X;\sigma_n,\alpha) = 1 - (1 - o(X,\alpha,V))(1 - o(X,\alpha,W)) \quad .$$

PROOF. Let us define the event T_V to be that there is no $v \in V$ such that $sp(X = v) = 0$. Similarly, let T_W be the event that there is no $w \in W$ such that $sp(X = w) = 0$. According to Lemma 3.11, an itemset X is derivable if T_V and T_W are both false.

Using the same argument as with Proposition 3.14, we see that $p(T_V) = o(X,\alpha,V)$. Similarly, $p(T_W) = o(X,\alpha,W)$. Since $V \cap W = \emptyset$, events T_V and T_W are independent. Hence, $r(X;\sigma_n,\alpha)$ is equal to

$$1 - p(\neg T_V \wedge \neg T_W) = 1 - (1 - p(T_V)(1 - p(T_W)) \quad .$$

This completes the proof. □

We will now consider closed itemsets. Unlike for the free/totally shattered itemsets, there is an exponential number of terms in the expression for the robustness. The key problem is that while we can write the robustness in a similar fashion as we did in the proofs of the previous propositions, the events $sp(X \cup \{y\}) < sp(X)$ for all $y \in A \setminus X$, will no longer be independent, and hence we cannot multiply the probabilities of the individual events. Indeed, in our running example, bde is a closed itemset. The events $sp(abde;D_\alpha) < sp(bde;D_\alpha)$ and $sp(bcde;D_\alpha) < sp(bde;D_\alpha)$ are clearly dependent since both events occur in exactly the same subsamples, namely those that contain at least one of the transactions 3 and 5.

PROPOSITION 3.16. *The robustness of a closed itemset is*

$$r(X;\sigma_c,\alpha) = \sum_{Y\supseteq X} (-1)^{|Y|-|X|}(1-\alpha)^{sp(X)-sp(Y)} \quad .$$

PROOF. Given an item $y \notin X$, define an event $E_y = sp(X \cup \{y\}; D_\alpha) = sp(X;D_\alpha)$. Itemset X is still closed in D_α if all E_y are false, thus $r(X;\sigma_c,\alpha)$ is equal to

$$1 - p\Big(\bigvee_{y\notin X} E_y\Big) = \sum_{Z\subseteq(A\setminus X)} (-1)^{|Z|} p\Big(\bigwedge_{y\in Z} E_y\Big),$$

where the equality follows from the inclusion-exclusion principle. Through this transformation we now need to determine the probability of all E_y, $y \in Z$ simultaneously being true. For this all $sp(X) - sp(Z \cup X)$ transactions containing X but not Z must have been excluded from D_α, hence

$$p\Big(\bigwedge_{y\in Z} E_y\Big) = (1-\alpha)^{sp(X)-sp(Z\cup X)} \quad .$$

Substituting this above and writing $Y = X \cup Z$ leads to the proposition. □

Example 3.17. In our running example, we have $sp(bde) = 4$. This itemset has 3 superitemsets having the supports $sp(abde) = sp(bcde) = sp(abcde) = 2$. Hence, the measure $r(bde; \sigma_c, \alpha)$ is equal to

$$1 - (1-\alpha)^{4-2} - (1-\alpha)^{4-2} + (1-\alpha)^{4-2} = 1 - (1-\alpha)^2,$$

where itemsets bde, $abde$, $bcde$, and $abcde$ correspond to the terms in the given order.

Unlike with the other predicates, analytic robustness for closed itemsets cannot be be computed in practice since there are $2^{K-|X|}$ terms in the analytic solution. It turns out that we cannot do much better as computing robustness is **NP**-hard.

PROPOSITION 3.18. *The following* Robustness of a Closed Itemset (RCI) *problem is* ***NP****-hard:*

> *For a given database D over the set of items A, parameters $\alpha, \rho \in [0,1]$, and itemset $X \subseteq A$, decide if $r(X; \sigma_c, D, \alpha) \geq \rho$.*

PROOF. We will reduce the well-known **NP**-complete vertex cover problem to the RCI problem. Let $G(V, E)$ be a graph. For every vertex $v \in V$, we will create a unique transaction with identifier tid_v. The set of items over which the transactions will be defined is the set of edges $E = \{e_1, \ldots, e_K\}$. Let $t_v = [t_{v1}, \ldots, t_{vK}]$ denote the binary vector of length $|E|$ defined as: for all $i = 1, \ldots, K$,

$$t_{vi} = 1 \quad \text{if and only if} \quad e_i \text{ is not incident with } v \quad .$$

The transaction database D is now defined as

$$D = \{(tid_v, t_v) \mid v \in V\} \quad .$$

The itemset X in the RCI-problem will be the empty set, $X = \emptyset$. Before we specify α and ρ, we show the following property:

LEMMA 3.19. *Let $S \subseteq D$; $\emptyset$ is closed in S if and only if $V_S = \{v \in V \mid (tid_v, t_v) \in S\}$ is a vertex cover of G.*

PROOF. If $\emptyset$ is closed in S, then for every e there is $t \in D$ such that $t_e = 0$, otherwise $sp(e) = sp(\emptyset)$. Hence, for all $e \in E$ there must exist at least one $v \in V_S$ $t_{ve} = 0$, that is, e must be incident with v. Since e was chosen arbitrary, this implies that every edge in E is covered by at least one node in V_S and hence V_S is a vertex cover of G. □

This relation between the closedness of $\emptyset$ in a subsample S and V_S being a vertex-cover allows us to establish the following relation between the robustness of $\emptyset$ in D and the existence of a vertex-cover of size k, that holds for any $\alpha \in [0,1]$.

LEMMA 3.20. *If G has a vertex cover of size k,*

$$r(\emptyset; \sigma_c, D, \alpha) \geq \alpha^k (1-\alpha)^{|D|-k}$$

otherwise,

$$r(\emptyset; \sigma_c, D, \alpha) \leq \sum_{j=k+1}^{|D|} \alpha^j (1-\alpha)^{|D|-j} \binom{|D|}{j} \quad .$$

PROOF. Indeed, let VC be a vertex cover of G, then $\emptyset$ is closed in $S = \{(tid_v, t_v) \mid v \in VC\}$. The probability that a randomly selected sample equals S is equal to

$$L = \alpha^k (1-\alpha)^{|D|-k},$$

which is a lower bound on the robustness of $\emptyset$. Otherwise, if there does not exist a vertex cover of size k, this implies that $\emptyset$ is not closed in any subsample S of size k or less. Therefore, the probability mass of all subsamples with at least $k+1$ transactions

$$U = \sum_{j=k+1}^{|D|} \alpha^j (1-\alpha)^{|D|-j} \binom{|D|}{j}$$

is an upper bound on the robustness of $\emptyset$. □

The proof now concludes by carefully choosing α such that $U \leq L$, and selecting ρ such that $U \leq \rho \leq L$; in that way, the robustness of the closedness of $\emptyset$ exceeds L and hence ρ if G has a vertex cover of size k or less, and otherwise the robustness is below U, and hence also below ρ. The last step in the proof is hence to show that we can always pick α such that $U \leq L$. It can easily be seen that $\alpha = 2^{-(|D|+1)}$ satisfies this condition: Since $\alpha \leq 1/2$, we can now bound U by

$$\sum_{j=k+1}^{|D|} \alpha^j (1-\alpha)^{|D|-j} \binom{|D|}{j} \leq \sum_{j=k+1}^{|D|} \alpha^{k+1} (1-\alpha)^{|D|-k-1} \binom{|D|}{j} = 2^{|D|} \alpha^{k+1} (1-\alpha)^{|D|-k-1} \quad .$$

The right hand-side is smaller than L if and only if $1-\alpha \geq 2^{|D|}\alpha$. Note that for our choice of α, we have $1-\alpha = 1-\alpha = 2^{-(|D|+1)} \geq 1/2 = 2^{|D|}\alpha$.

The binary representation of the numbers α and ρ are polynomial in the size of the original vertex cover problem and the reduction can be carried out in polynomial time. □

4. ORDERING PATTERNS

The robustness measure depends on the parameter α. In this section we propose a parameter-free approach. The idea is to study how the measure is behaving when α is *close* to 1. We can show that there is a (small) neighborhood close to 1, where the *ranking* of itemsets does not depend on α, that is, there exists $\beta < 1$ such that if $\alpha, \alpha' \in [\beta, 1]$ $r(X;\sigma,D,\alpha) \leq r(Y;\sigma,D,\alpha)$ if and only if $r(X;\sigma,D,\alpha') \leq r(Y;\sigma,D,\alpha')$.

We will show how to compute the ranking in this region without actually computing the measure or determining β. We can use this ranking to select top-k robust itemsets.

In this section we will first give first formal definition, and discuss the theoretical properties of the ranking. In the next section we demonstrate how we can compute the order in practice, that is, how to avoid determining β and computing the actual robustness.

4.1. Measuring robustness when α approaches 1

When $\alpha = 1$ then $D_\alpha = D$ with probability 1 and the measure is equivalent to the underlying predicate, providing only a crude ranking: itemsets that satisfy the predicate vs. itemsets that do not. If we make α slightly smaller the measure will decrease a little bit for each itemset. The amount of this change will vary from one itemset to another based on how likely removing only very few transactions will break the predicate for this itemset. We can use the magnitude of this change to obtain a more fine-grained ranking by robustness. The key result for this is that there is a small neighborhood below 1 in which the ranking of itemsets based on the measure does not depend on α.

PROPOSITION 4.1. *Given a predicate σ and a dataset D, there exists a number $\beta < 1$ such that*

$$r(X;\sigma,D,\alpha) \leq r(Y;\sigma,D,\alpha) \quad \textit{if and only if} \quad r(X;\sigma,D,\alpha') \leq r(Y;\sigma,D,\alpha')\,,$$

for any itemset X and Y and $\beta \leq \alpha \leq 1$, $\beta \leq \alpha' \leq 1$.

PROOF. Fix X and Y and consider

$$f(\alpha) = r(X;\sigma,D,\alpha) - r(Y;\sigma,D,\alpha) \quad .$$

Since the measure is a finite sum of probabilities that are, according to Eq. 1, polynomials of α, the function f is a polynomial. This implies that f can have only a finite number of 0s, of $f = 0$. Consequently there is a neighborhood $N = [\beta, 1]$ such that either $f(\alpha) \geq 0$ for any $\alpha \in N$, or $f(\alpha) \leq 0$ for $\alpha \in N$. Since there is only a finite number of itemsets, we can take the maximum of all βs to prove the theorem. □

Proposition 4.1 allows us to define an order for itemsets based on the measure for $\alpha \approx 1$.

Definition 4.2. Given a predicate σ, and a dataset D, we say that $X \preceq_\sigma Y$, where X and Y are itemsets, if there exists $\beta < 1$ such that $r(X;\sigma,D,\alpha) \leq r(Y;\sigma,D,\alpha)$ for any α such that $\beta \leq \alpha \leq 1$. Moreover, if $r(X;\sigma,D,\alpha) < r(Y;\sigma,D,\alpha)$ for some $\alpha \geq \beta$, then we write $X \prec_\sigma Y$.

Note that Proposition 4.1 implies that $\preceq_\sigma$ is a total linear order. That is, we can use this relation to order itemsets.

4.2. Properties of the order

In this section we will study the properties of the order. Namely, we will show two properties:

— We will show in Proposition 4.4 that robustness for $\alpha \approx 1$, essentially measures how many transactions we need to remove in order to make the predicate fail. The more transactions are needed, the more robust is the itemset.
— We will show in Proposition 4.9 that when we increase the number of transactions, then a ranking based on robustness *for any fixed* α will become equivalent with the ranking based on $\prec_\sigma$.

First, we will need the following key lemma that can be proven by elementary real analysis.

LEMMA 4.3. *Let $f(x) = \sum_{i=0}^{N} a_i x^i$ be a non-zero polynomial. Let k be the first index such that $a_k \neq 0$ If $a_k > 0$, then there is a $\beta > 0$ such that $0 \leq x \leq \beta$ implies $f(x) \geq 0$. Similarly, if $a_k < 0$, then there is a $\beta > 0$ such that $0 \leq x \leq \beta$ implies $f(x) \leq 0$.*

The lemma essentially says that if we express the robustness as a polynomial of $1 - \alpha$, then we can determine the order by studying the coefficients of the polynomial.

Our first application of this lemma is a characterization of the order. Assume two itemsets X and Y. Assume that we need to remove n transactions in order to make the predicate $\sigma(Y)$ fail and that we can fail $\sigma(X)$ by removing less than n transactions. Then it holds that $X \prec_\sigma Y$. The following proposition generalizes this idea.

PROPOSITION 4.4. *Let σ be a predicate, X and Y two itemsets, and D a dataset. Define a vector $c(X)$ of length $|D|$ such that $c_k(X)$ is the number of subsamples of D with $|D| - k$ transactions failing the predicate $\sigma(X)$. Similarly, define $c(Y)$. Then, $c(X) = c(Y)$ implies that $r(X;\sigma,D,\alpha) = r(Y;\sigma,D,\alpha)$ for any α. If $c(X)$ is larger than $c(Y)$ in lexicographical order, then $X \prec_\sigma Y$.*

PROOF. Let us first write the robustness of X using the vector $c_k(X)$. We have,

$$1 - r(X;\sigma,D,\alpha) = p(\sigma(X;D_\alpha)=0) = \sum_{k=0}^{|D|} p(\sigma(X;D_\alpha)=0, |D_\alpha| = |D|-k)$$
$$= \sum_{k=0}^{|D|}(1-\alpha)^k\alpha^{|D|-k}\sum_{\substack{S\subseteq D\\ |S|=|D|-k}} 1-\sigma(X;S) = \sum_{k=0}^{|D|}(1-\alpha)^k\alpha^{|D|-k}c_k(X) \quad .$$

If $c_k(X) = c_k(Y)$ it follows immediately that the robustness for X and Y are identical.

Assume now that $c(X)$ is larger than $c(Y)$ in lexicographical order. That it, there is l such that $c_l(X) > c_l(Y)$ and $c_k(X) = c_k(Y)$ for $k < l$. We have

$$r(Y;\sigma,D,\alpha) - r(X;\sigma,D,\alpha) = \sum_{k=l}^{|D|}(1-\alpha)^k\alpha^{|D|-k}(c_k(X)-c_k(Y))$$
$$= (c_l(X)-c_l(Y))(1-\alpha)^l + f(1-\alpha),$$

where $f(x)$ is a polynomial such that the degree of an individual term in f is bigger than l. Lemma 4.3 now proves the proposition. □

Interestingly enough, if we would define the order based on $\alpha \approx 0$, then we have a similar result with the difference that instead of deleting transactions we would be adding them. We would rank Y higher than X if we can satisfy $\sigma(Y)$ with less transactions than the number of transactions needed to satisfy $\sigma(X)$.

Ranking itemsets based on how many transactions can be deleted is similar to the breakdown point that measures robustness of statistical estimators. The breakdown point for estimators such as the mean is the number of observations that can be made arbitrarily large before the estimator becomes arbitrarily large as well. The breakdown value of the mean is 1, it becomes infinity as soon as one observation is set to infinity. In contrast the median can handle just under half of the observations to be set to infinity before it breaks down.

We will next show that, in essence, for large datasets the robustness for *any* $\alpha > 0$ will produce the same ranking as the order defined for α close to 0. For this we will consider predicates only of certain type. The reason for this is to avoid some pathological predicates, for example, $\sigma(X;D) = 1$ if $|D|$ is even, and 0 otherwise.

Definition 4.5. Let σ be a predicate. Let K be the number of items and let X be an itemset. We say that σ is a *monotone CNF predicate* if there is a collection $\{B_i\}_1^L$ of sets of binary vectors of length K, (possibly) depending on X and K such that

$$\sigma(X;D) = \begin{cases} 1 & \text{if } D\cap B_i \neq \emptyset \text{ for each } i = 1,\ldots,L,\\ 0 & \text{otherwise,}\end{cases}$$

that is, in order to $\sigma(X;D) = 1$, D must contain a transaction from each B_i.

Every predicate we consider in this paper is in fact a monotone CNF predicate.

PROPOSITION 4.6. *Predicates σ_c, σ_f, σ_n, and σ_s are monotone CNF predicates.*

PROOF. Fix an itemset $X = x_1\cdots x_N$, and K, the total number of items. Let $\Omega = \{0,1\}^K$ be the collection of all binary vectors of length K.

Free itemsets. Let $B_i = \left\{t \in \Omega \mid t_{x_i} = 0, t_{x_j} = 1, j \neq i, 1 \leq j \leq N\right\}$ for $i = 1, \ldots, N$. In order to X to be free in D, we must have $D \cap B_i \neq \emptyset$. Otherwise, $sp(X) = sp(X \setminus \{x_i\})$, making X not free.

Closed itemsets. Define $K - N$ sets by $B_i = \left\{t \in \Omega \mid t_{x_j} = 1, t_i = 0, 1 \leq j, \leq N\right\}$ for $i \notin X$. X is closed in D if and only if $D \cap B_i \neq \emptyset$. Otherwise, $sp(X) = sp(X \cup \{x_i\})$, making X not closed.

Totally shattered itemsets. Define 2^K sets by $B_u = \left\{t \in \Omega \mid t_{x_j} = u_j, 1 \leq j, \leq N\right\}$ for each $u \in \{0,1\}^K$. The proposition follows directly from the definition.

Non-derivable itemsets. Let $C_u = \left\{t \in \Omega \mid t_{x_j} = u_j, 1 \leq j, \leq N\right\}$ for each $u \in \{0,1\}^K$. Define 4^{K-1} sets by $B_{u,v} = C_u \cup C_v$, where $u, v \in \{0,1\}^K$, u has odd number of 1s and v has even number of 1s. The proposition follows directly Lemma 3.11. □

Example 4.7. In our running example, an itemset bde is closed if and only D contains at least one transaction from $B_1 = \{(0,1,1,1,1), (0,1,0,1,1)\}$ and from $B_2 = \{(1,1,0,1,1), (0,1,0,1,1)\}$. The dataset does contain $(0,1,0,1,1)$ making bde closed.

In order to prove the main result we need the following lemma showing that the robustness of a monotone CNF predicate can be expressed in a certain way. We can then exploit this expression in Proposition 4.9.

LEMMA 4.8. *Let σ be a monotone CNF predicate and let X be an itemset. Let K be the number of itemsets. Then there is a set of coefficients $\{c_i\}_1^N$ and a collection $\{S_i\}_1^N$ of sets of binary vectors of length K such that*

$$r(X; \sigma, D, \alpha) = \sum_{i=1}^{N} c_i (1-\alpha)^{|D \cap S_i|} \quad .$$

PROOF. Let S be a set of binary vectors of length K. The probability of a random subsample D_α not having a transaction from S is equal to

$$p(D_\alpha \cap S = \emptyset) = (1-\alpha)^{|D \cap S|} \quad .$$

We can rewrite the robustness using the inclusion-exclusion principle,

$$\begin{aligned} r(X; \sigma, D, \alpha) &= 1 - p(\sigma(X; D_\alpha) = 0) = 1 - p(D_\alpha \cap B_1 = \emptyset \vee \cdots \vee D_\alpha \cap B_L = \emptyset) \\ &= 1 - \sum_{i=1}^{L} p(D_\alpha \cap B_i = \emptyset) + \sum_{1 \leq i < j \leq L} p(D_\alpha \cap ((B_i \cup B_j) = \emptyset) + \cdots \\ &= 1 - \sum_{i=1}^{L} (1-\alpha)^{|D \cap B_i|} + \sum_{1 \leq i < j \leq L} (1-\alpha)^{|D \cap (B_i \cup B_j)|} + \cdots \end{aligned}$$

The right-hand side of the equation has the correct form, proving the lemma. □

We are now ready to state the main result of this subsection. Assume that we have a dataset D and we create a new larger dataset R by sampling transactions with replacement from D. The dataset R has the same characteristics as D, it is only larger. Then if we have two itemsets X and Y such that $X \prec_\sigma Y$, then on average we will have $r(X; \sigma, R, \alpha) < r(Y; \sigma, R, \alpha)$ for *any* $\alpha > 0$ assuming that $|R|$ is large enough.

PROPOSITION 4.9. *Let σ be a monotone CNF predicate and let D be a dataset. Let X and Y be itemsets such that $X \prec_\sigma Y$ in D. Let q be the empirical distribution of D*

and let R_m be a dataset of m random transactions drawn from q. Let $0 < \alpha < 1$. Then there is M such that

$$\mathrm{E}\left[r(X;\sigma,R_m,\alpha)\right] < \mathrm{E}\left[r(Y;\sigma,R_m,\alpha)\right]$$

for $m > M$.

PROOF. Let us write $\beta = 1 - \alpha$. Lemma 4.8 says that we can write the difference in robustness as

$$r(Y;\sigma,R_m,\alpha) - r(X;\sigma,R_m,\alpha) = \sum_{i=1}^{N} c_i \beta^{|R_m \cap S_i|}$$

for certain coefficients $\{c_i\}_1^N$ and sets of binary vectors $\{S_i\}_1^N$. Let $d_k = \sum_{|D\cap S_i|=k} c_i$. Since $X \prec_\sigma Y$, Lemma 4.3 implies that there is l such that $d_l > 0$ and $d_k = 0$ for $k < l$.

Let S be a set of binary transactions, and let $k = |S \cap D|$, that is, the probability of generating a random transaction belonging to S is $q(t \in S) = k/|D|$. We have

$$\mathrm{E}\left[\beta^{|S\cap R_m|}\right] = \sum_{j=0}^{m} \beta^j \binom{m}{j} q(t\in S)^j (1 - q(t\in S))^{m-j} = \left(\beta\frac{k}{|D|} + 1 - \frac{k}{|D|}\right)^m \quad .$$

We will write t_k as shorthand for the right-side hand of the equation. Note that since $\beta < 1$, we have $t_{k+1} < t_k$. We can write the expected difference between robustness as

$$\mathrm{E}\left[\sum_{i=1}^{N} c_i \beta^{|R_m\cap S_i|}\right] = \sum_{k=0}^{|D|} d_k t_k^m = d_l t_l^m + \sum_{k=l+1}^{|D|} d_k t_k^m = t_l^m\big(d_l + \sum_{k=l+1}^{|D|} d_k (t_k/t_l)^m\big) \quad .$$

Since $t_k/t_l < 1$, the terms $(t_k/t_l)^m$ approach 0 as m goes to infinity. Hence, there is M such that the sum in the right-hand side of the equation is larger than $-d_l$ for $m > M$. This guarantees that the difference is positive proving the proposition. □

This proposition suggests that ranking based on a fixed α and a parameter-free ranking will eventually agree if the dataset is large enough. In other words, β in Proposition 4.1 will get smaller (on average) as the size of the dataset increases. We will see this phenomenon later on in Propositions 5.8 and 5.9.

5. COMPUTING ORDER IN PRACTICE

In this section we demonstrate how we can compute the ranking for free, non-derivable, and totally shattered itemsets and how we can estimate the ranking for closed itemsets. For computational complexity see Table I.

5.1. Free and totally shattered itemsets

In this section we will demonstrate that we can compute the order for free and totally shattered itemsets without finding an appropriate α. We will do this by analyzing the coefficients of the measure viewed as a polynomial of $1 - \alpha$.

Note that for free and totally shattered itemsets these polynomials are given in Proposition 3.13 and Proposition 3.14. In order to obtain the coefficients of the polynomial we can simply expand the polynomials. However, the polynomials in Proposition 3.13 and Proposition 3.14 are regular enough so that we can compute the order without expanding the polynomials. In order to do so we need the following definition for ordering sequences.

Definition 5.1. Given two non-decreasing sequences $s = s_1, \ldots, s_K$ and $t = t_1, \ldots, t_N$, we write $s \prec t$ if either there is $s_n < t_n$ and $s_i = t_i$ for all $i < n$ or t is a proper prefix sequence of s, that is, $s_i = t_i$ for $i \leq N < K$. We write $s \preceq t$, if $s = t$ or $s \prec t$.

The following proposition will allow us to order itemsets without expanding the polynomials in Propositions 3.13–3.14.

PROPOSITION 5.2. *Assume two polynomials*

$$f(\alpha) = \prod_{i=1}^{K}(1-(1-\alpha)^{s_i}) \quad \textit{and} \quad g(\alpha) = \prod_{i=1}^{N}(1-(1-\alpha)^{t_i}),$$

where $s = s_1, \ldots, s_K$ *and* $t = t_1, \ldots, t_N$ *are non-decreasing sequences of integers,* $s_i, t_i \geq 0$. *If* $t \preceq s$, *then there is a* $\beta < 1$ *such that* $\beta \leq \alpha \leq 1$ *implies* $f(\alpha) \geq g(\alpha)$.

PROOF. The case $s = t$ is trivial. Hence we assume that $s \neq t$. If $s_1 = 0$ or $t_1 = 0$, then $f(\alpha) = 0$ or $g(\alpha) = 0$, and the result follows, hence we will assume that $s_i, t_i > 0$.

Let $\{a_i\}$ and $\{b_i\}$ be coefficients such that

$$f(\alpha) = \sum_i a_i(1-\alpha)^i \quad \text{and} \quad g(\alpha) = \sum_i b_i(1-\alpha)^i \quad .$$

Let I_n be the collection of all subsequences of s that sum to n,

$$I_n = \Big\{u \mid u \text{ is a subsequence of } s,\ \sum_{i=1}^{|u|} u_i = n\Big\} \quad .$$

Similarly, let J_n be the collection of all subsequences of t that sum to n. We can rewrite $f(\alpha)$ as

$$f(\alpha) = \sum_{\substack{u \text{ is a} \\ \text{subseq. of } s}} (-1)^{|u|}(1-\alpha)^{\sum_i u_i}$$

which implies that

$$a_n = \sum_{u \in I_n}(-1)^{|u|} \quad \text{and similarly} \quad b_n = \sum_{u \in J_n}(-1)^{|u|} \quad .$$

Assume that $t \prec s$. If s is a prefix sequence of t, then

$$g(\alpha) = f(\alpha) \prod_{i=K+1}^{N}(1-(1-\alpha)^{t_i}) \leq f(\alpha),$$

which proves the proposition. Let n be as given in Definition 5.1. For every $i < t_n < s_n$, the subsequences in I_i and J_i contain subsequences from s and t with indices smaller than n. Since s and t are identical up to n, then it follows that $I_i = J_i$ and consequently $a_i = b_i$. Let $u \in I_{t_n}$. Assume that $|u| > 1$. Since, we assume that $s_i > 0$, u is a subsequence of $s_1, \ldots, s_{n-1}$. This means that we will find the same subsequence in J_{t_n}. Let A be the number of singleton sequences in I_{t_n}, $A = |\{u \in I_{t_n} \mid |u| = 1\}|$, and let B be the number of singleton sequences in J_{t_n}. These singleton sequences correspond to the entries in s and t having the same value as t_n. Since s and t are identical up to n and s does not contain t_n after s_n, it holds that $B > A$. We have now $a_{t_n} - b_{t_n} = B - A > 0$. Lemma 4.3 now implies that $f(1-x) \geq g(1-x)$, when x is close to 0. Write $\alpha = 1 - x$ to complete the proof. □

The polynomials in Propositions 3.13–3.14 have the form used in Proposition 5.2. Consequently, we can use the proposition to order itemsets. In order to do that we need the following definitions.

Definition 5.3. Given a dataset D and an itemset X, we define a *free margin vector* $mv(X; D, \sigma_f)$ to be the sequence of $|X|$ integers $sp(X = v; D)$, where v is a binary vector having $|X| - 1$ ones, *ordered* in the increasing order.

Similarly, we define a *totally shattered margin vector* $mv(X; D, \sigma_s)$ to be a sequence of $2^{|X|}$ integers $sp(X = v; D)$ *ordered* in the increasing order.

COROLLARY 5.4. *Given itemsets X and Y and a dataset D, $X \preceq_{\sigma_f} Y$ if and only if* $mv(X; D, \sigma_f) \preceq mv(Y; D, \sigma_f)$.

COROLLARY 5.5. *Given itemsets X and Y and a dataset D, $X \preceq_{\sigma_s} Y$ if and only if* $mv(X; D, \sigma_f) \preceq mv(Y; D, \sigma_s)$.

Example 5.6. In our running example, $sp(ab = [1, 0]) = 1$ and $sp(ab = [0, 1]) = 2$, hence the free margin vector is equal to $mv(ab; \sigma_f) = [1, 2]$. Similarly, we have $sp(ae = [1, 0]) = 1$ and $sp(ae = [0, 1]) = 3$, hence the free margin vector is equal to $mv(ae; \sigma_f) = [1, 3]$. Hence, we conclude that $ab \prec_{\sigma_f} ae$.

Margin vectors are useful to determine the order of robust itemsets. However, we can also use them to provide a bound for β given in Definition 4.2. More specifically, the further the margin vectors are from each other the lower α can be such that the robustness still agrees with the order. To make this formal, we will need the following definition.

Definition 5.7. Assume two non-decreasing sequences $s = s_1, \ldots, s_K$ and $t = t_1, \ldots, t_N$ such that $s \preceq t$. Let n be the first index such that $s_n < t_n$, we define $d(s, t) = t_n - s_n$. If no such such index exist, that is, t is a prefix sequence of s, we define $d(s, t) = \infty$.

The following propositions state that the larger $d(s, t)$, the lower α can be. This reflects the result of Proposition 4.9: large datasets will result in large differences in margin vectors, allowing α to be small.

PROPOSITION 5.8. *Assume itemsets X and Y and a dataset D such that $X \preceq_{\sigma_f} Y$. Let $d = d(mv(X; D, \sigma_f), mv(Y; D, \sigma_f))$. Then*

$$r(X, \sigma_f, D, \alpha) \leq r(Y, \sigma_f, D, \alpha) \quad \textit{for} \quad \alpha \geq 1 - \frac{1}{\sqrt[d]{|Y| + 1}} \quad .$$

PROPOSITION 5.9. *Assume itemsets X and Y and a dataset D such that $X \preceq_{\sigma_s} Y$. Let $d = d(mv(X; D, \sigma_s), mv(Y; D, \sigma_s))$. Then*

$$r(X, \sigma_f, D, \alpha) \leq r(Y, \sigma_f, D, \alpha) \quad \textit{for} \quad \alpha \geq 1 - \frac{1}{\sqrt[d]{2^{|Y|} + 1}} \quad .$$

Both propositions follow immediately from the following proposition.

PROPOSITION 5.10. *Given two non-decreasing sequences $s = s_1, \ldots, s_K$ and $t = t_1, \ldots, t_N$ such that $s \preceq t$, let $d = d(s, t)$. Then*

$$\prod_{i=1}^{K} (1 - (1 - \alpha)^{s_i}) \leq \prod_{i=1}^{N} (1 - (1 - \alpha)^{t_i}) \quad \textit{for} \quad \alpha \geq 1 - \frac{1}{\sqrt[d]{N + 1}} \quad .$$

PROOF. If t is a prefix sequence of s, then the inequality holds for any α. Assume that t is not a prefix sequence and let n be the first index such that $s_n < t_n$. Write

$\beta = 1 - \alpha$. We can upper bound the left-hand side by

$$\prod_{i=1}^{K}(1-\beta^{s_i}) \leq (1-\beta^{s_n}) \prod_{i=1}^{n-1}(1-\beta^{s_i})$$

and lower bound the right-hand side by

$$\prod_{i=1}^{N}(1-\beta^{t_i}) \geq (1-\beta^{s_n+d})^N \prod_{i=1}^{n-1}(1-\beta^{s_i}) \quad .$$

Hence it is sufficient to show that

$$(1-\beta^{s_n}) \leq (1-\beta^{s_n+d})^N \quad \textbf{or} \quad \log(1-\beta^{s_n}) \leq N\log(1-\beta^{s_n+d}) \quad .$$

We apply the inequalities $-x \leq \log(1-x) \leq -x/(1-x)$ which gives us

$$-\beta^{s_n} \leq -N\frac{\beta^{s_n+d}}{(1-\beta^{s_n+d})} \quad \textbf{or} \quad 1 \geq N\beta^d + \beta^{s_n+d} \quad .$$

Since $\beta^d \geq \beta^{s_n+d}$ it is sufficient to have $1 \geq (1+N)\beta^d$. This is true for $\beta \leq \frac{1}{\sqrt[d]{N+1}}$. □

5.2. Closed itemsets

In this section we will introduce a technique for estimating the ranking for closed itemsets. As the measure for closed itemsets has a different form than for free or totally shattered itemsets we are forced to seek for alternative approaches. We approach the problem by first expressing the coefficients of the polynomial with supports of closed itemsets. Then we estimate the polynomial by considering only the most frequent closed itemsets.

Let us consider Proposition 3.16. Let a_k be the coefficient for the kth term of the polynomial for $r(X;\sigma_c,\alpha)$ given in Proposition 3.16. If we can compute these numbers efficiently, we can use Lemma 4.3 to find the ranking.

We will do this by first expressing a_k using closed itemsets. In order to do that let $cl(X)$ be the closure of an itemset X. Let us define

$$e(Y,X) = \sum_{\substack{Z \supseteq X,\\ cl(Z)=Y}} (-1)^{|Z|+|X|}$$

to be the alternating sum over all itemsets containing X and having Y as their closure. Since all the itemsets having the same closure will have the same support we can write the coefficients a_k using $e(Y,X)$,

$$a_k = \sum_{\substack{Y \supseteq X,\\ sp(X)-sp(Y)=k}} (-1)^{|Y|+|X|} = \sum_{\substack{Y \supseteq X, Y=cl(Y)\\ sp(X)-sp(Y)=k}} e(Y,X) \quad . \tag{3}$$

To compute $e(Y,X)$, first note that $e(X,X)=1$. If $Y \neq X$, then using the following identity

$$\sum_{\substack{Y \supseteq Y' \supseteq X\\ Y'=cl(Y')}} e(Y',X) = \sum_{Y \supseteq Z \supseteq X} (-1)^{|Z|+|X|} = 0$$

we arrive to

$$e(Y,X) = -\sum_{\substack{Y \supsetneq Y' \supseteq X\\ Y'=cl(Y')}} e(Y',X) \quad . \tag{4}$$

Thus, we can compute $e(Y, X)$ from $e(Y', X)$, where Y' is a closed subset of Y. This is convenient, because when computing $e(Y, X)$, say for a_k, we have already computed all the subsets of Y for previous coefficients.

Example 5.11. Consider itemset e in our running example. There are two closed supersets of e, namely bde and $abcde$, having the supports 4 and 2, respectively. Using the update equations, we see that $e(e, e) = 1$, $e(bde, e) = -1$, and $e(abcde, e) = 0$. As $sp(e) = 5$, we see that the non-zero coefficients a_i are $a_0 = 1$ and $a_1 = -1$.

The problem with this approach is that we can still have an exponential number of closed itemsets. Hence, we chose to estimate the ranking by only using *frequent* closed itemsets and estimate the remaining itemsets to have a support of 0.

This estimation is achieved by removing all closed non-frequent itemsets from the sums of Eqs. 3 and 4 and adding an itemset containing all the items and having the support 0. The code for this estimation is given in Algorithm 1.

Algorithm 1: Algorithm for estimating coefficients of the polynomial given in Proposition 3.16.

input : X an itemset, $\mathcal{C}$, frequent closed itemsets
output : $\{a_k\}$, coefficients of the polynomial

if $A \notin \mathcal{C}$ **then** add A to $\mathcal{C}$ with $sp(A) = 0$;
$\mathcal{C} \leftarrow \{Y \in \mathcal{C} \mid X \subseteq Y\}$;
$\mathcal{L} \leftarrow$ sets in $\mathcal{C}$ ordered by the subset relation;
$e(X, X) \leftarrow 1$;
for $Y \in \mathcal{L}$ **do**
    $e(Y, X) \leftarrow -\sum_{Z \in \mathcal{C}, Z \subsetneq Y} e(Z, X)$;
    $k \leftarrow sp(X) - sp(Y)$;
    $a_k \leftarrow a_k + e(Y, X)$;

Algorithm 1 takes $O(|\mathcal{C}|^2)$ time. In practice, this is much faster because an average itemset does not have that many supersets.

Now that we have a way of estimating a_k from frequent closed itemsets, we can, given two itemsets X and Y, search the smallest k for which the coefficients differ in order to apply Lemma 4.3. Note that if the index of the differing coefficient, say k, is such that $sp(X) - k$ is larger or equal to the support threshold, then a_k is correctly computed by our estimation, and our approximation yields a correct ranking.

5.3. Non-derivable itemsets

In this section we will discuss how to compute the ranking non-derivable itemsets. The ranking for non-derivable is particularly difficult because we cannot use Proposition 5.2 to avoid expanding the polynomial given in Proposition 3.15. We can, however, expand the polynomial since, due to Eq. 1, it only has $|D|$ terms. Once we have expanded the polynomial, we can use Lemma 4.3 to compare the itemsets.

First note that we can rewrite the measure as

$$r(X; \sigma_n, \alpha) = o(X, \alpha, V) + o(X, \alpha, W) - o(X, \alpha, U)\,, \tag{5}$$

where U consists of all binary vectors of length $|X|$, V is the subset of U containing vectors having odd number of ones, and $W = U \setminus V$.

Next, we will show how to expand a term $o(X, \alpha, S)$ for any set of binary vectors S. Once we are able to do that, we can expand each term in Eq. 5 individually to compute

the final coefficients. In order to do that, we will use the identity

$$(1-x^a)\sum_{i=0}^{N} c_i x^i = \sum_{i=1}^{N+a} (c_i - c_{i-a})x^i,$$

where in the right-hand side we define $c_i = 0$ for $i < 0$ or $i > N$. This gives us a simple iterative procedure, given in Algorithm 2: For each $v \in S$, we shift the current coefficients by $sp(X = v)$ and subtract the result from the current coefficients.

Algorithm 2: EXPAND(X, D, S), expands the polynomial $o(X, D, S)$

input : X, an itemset, D, a dataset, S a set of vectors
output : $\{c_i\}_1^{|D|}$ set of coefficients of the polynomial $o(X, D, S)$

```
c_i ← 0 for i = 0, ..., |D|;
c_0 ← 1;
foreach v ∈ S do
    s ← sp(X = v);
    n_i ← 0 for i = 0, ..., s − 1;
    n_i ← c_{i−s} for i = s, ..., |D|;
    c_i ← c_i − n_i for i = 0, ..., |D|;
return {c_i}_1^{|D|};
```

The highest degree in the polynomial will be $\sum_{v \in S} sp(X = v)$. Since, each v is unique in S, this number is bounded by $|D|$. This means that we have to consider only $|D|$ coefficients and that the computational complexity of EXPAND is $O(|S||D|)$. Consequently, computing the coefficients in Eq. 5 will take $O(|U||D|) = O(2^{|X|}|D|)$ time. We can further speed this up by using sparse vectors, and computing the terms in a lazy fashion during the comparison.

Example 5.12. Consider itemset ac in our running example. We have $sp(ac = (0,0)) = 3$, $sp(ac) = 2$, $sp(ac = (1,0)) = 1$, and $sp(ac = (0,1)) = 0$. Let $V = \{(0,1),(1,0)\}$, $W = \{(0,0),(1,1)\}$ and $U = V \cup W$. Since $sp(ac = (0,1)) = 0$, both $o(X, \alpha, V)$ and $o(X, \alpha, U)$ are 0. We have

$$o(X, \alpha, W) = (1-(1-\alpha)^3)(1-(1-\alpha)^2) = 1-(1-\alpha)^2-(1-\alpha)^3+(1-\alpha)^5 \quad .$$

Consequently, EXPAND will return $(1, 0, -1, -1, 0, 1, 0)$ as coefficients.

6. EXPERIMENTS

In this section we present our experiments.

— We study typical behavior of robustness for free, totally shattered, and non-derivable itemsets as a function of α.
— We test how similar the rankings are based on robustness and based on the order $\prec_\sigma$.
— We test how the ranking of robust closed itemsets changes under the effect of noise.

In addition, we provide examples of top-k robust closed and free itemsets.

6.1. Datasets

We used datasets from three repositories. The 8 FIMI [Goethals and Zaki 2003] datasets include large transaction datasets derived from traffic data, census data, and retail data. Two datasets are synthetically generated to simulate market basket data. The datasets from the UCI Machine Learning Repository [Asuncion and Newman 2007]

represent classification problems from a wide variety of domains. We used the itemset representations of 29 datasets from the LUCS repository [Coenen 2003]. Finally we used 18 text datasets shipped with the Cluto clustering toolkit [Zhao and Karypis 2002] but converted to itemsets using a binary representation of words in documents discarding the term frequencies.

6.2. Reducing the number of patterns

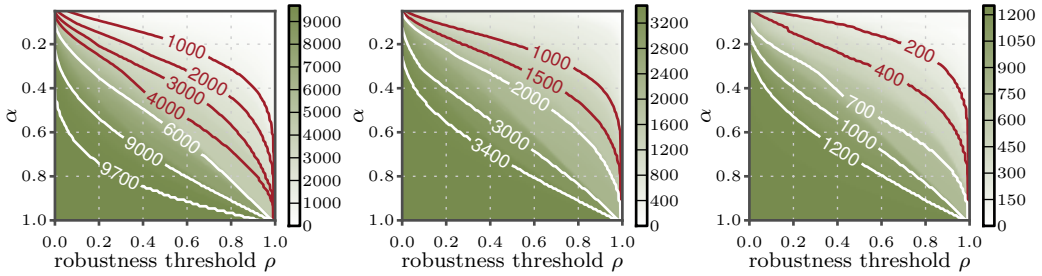
The goal of the first experiment is to show that this new constraint for itemsets can significantly reduce the number of itemsets reported in the results by removing itemsets that are spurious in the sense that they are unlikely to be observed on many subsamples. Throughout this section we will use α for the size of the data sample, ρ for the minimum robustness threshold, and τ for the minimum support threshold.

Our first question is how the parameters should be chosen. It is clear that if we choose α very close to 1, then even itemsets that would lose their predicate by removing only a few transactions still have a high likelihood of being found. We would thus expect most robustness values to be close to 1 when α is close to 1. This would make choosing a suitable ρ very difficult and might lead to problems due to floating point arithmetics. Similarly, choosing α close to 0 will cause most itemsets to have a very low likelihood of still being found, thus most robustness values will be close to 0. Thus choosing a medium α will be most useful to emphasize the quantitative difference between itemsets of various robustness.

As for the minimum robustness threshold ρ, the larger its value is, the stricter the filtering will be. Choosing the threshold is somewhat application dependent but it should not be close to zero, otherwise no reduction will be observed.

To confirm our reasoning we performed a parameter study for the itemset version of the *Zoo* dataset that describes 101 animals with 42 boolean attributes. This data contains 9 702 free itemsets, 3 476 non-derivable itemsets, and 1 252 totally shattered itemsets (at minimum support $\tau = 0.01$). The number of itemsets as a function of α and ρ is given in Figure 1. As expected

— for large α all but the largest ρ do not reduce the number of itemsets reported,
— as α becomes smaller, the itemsets are spread smoothly across the range of ρ allowing a meaningful quantitative evaluation,
— for small α almost no itemsets are reported even for very small ρ.



(a) free itemsets (b) non-derivable itemsets (c) totally shattered itemsets

Fig. 1. Number of free, non-derivable, and totally shattered itemsets on *Zoo* ($\tau = 0.01$) dataset as a function of α and ρ.

In order to evaluate if this holds for more datasets, we computed the number of free/non-derivable/totally shattered using different αs and normalized this by the

number of robust itemsets exceeding the minimal robustness threshold of 0.1. In order to minimize the variance of behavior of the robustness in a single dataset, we consider an average over all test datasets, which we give in Figure 2. We see the same phenomenon as in Figure 1. Large values of α induce a skewed distribution which becomes more balanced as we decrease the value of α. Consider $\alpha = 0.9$. Our test datasets typically contain a lot of itemsets having only one transaction keeping them from becoming non-free. This can be seen as a dip of the curve for $\alpha = 0.9$ at $\rho = 0.9$ in Figure 2(a). A second dip at $\rho = 0.81$ represents the itemsets that can be made non-free by deleting two transactions. As we make α smaller, these dips become less prominent.

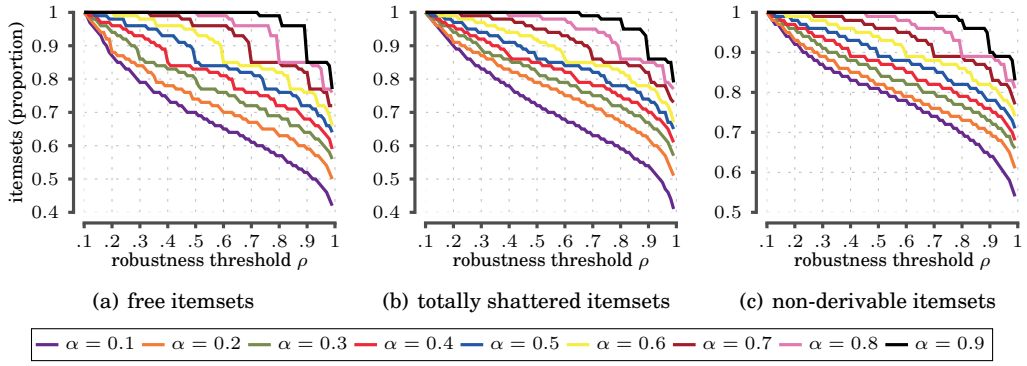


Fig. 2. Average of the number of free/totally shattered/non-derivable itemsets as a function of ρ normalized by the number of itemsets for $\rho = 0.1$. Average is taken over all test datasets

Based on this we chose $\alpha = 0.5$ and plotted the number of free itemsets as a function of ρ. Figure 3(a) shows that for the *Zoo* dataset there are many free itemsets with very different robustness values showing a rich structure that can be exploited to rank and reduce the number of itemsets. Similar results were observed for many of the UCI datasets. Figure 3(b) shows a representative example for the text datasets. While the distribution is much more skewed, a large ρ would also reduce the number of itemsets by about 50%. Finally, Figure 3(c) shows an example for a large transactional dataset with 88k transactions. Using $\alpha = 0.5$ generated a distribution where all values were close to one so we needed to set $\alpha = 0.01$ to better show the quantitative differences of the itemsets. This demonstrates that the more transactions a dataset contains, the more skewed the distribution for a fixed α will be.

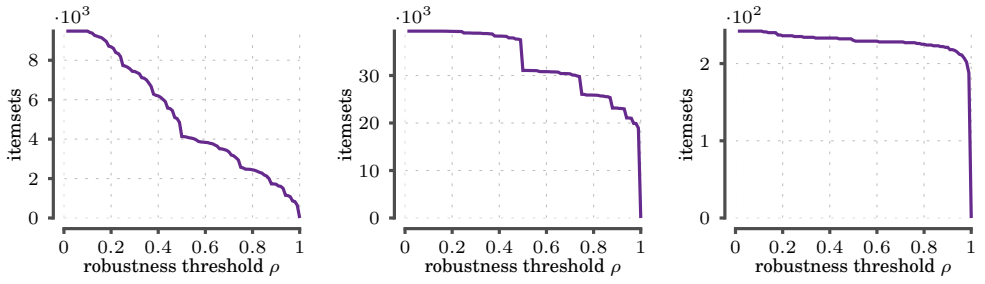


Fig. 3. Number of free itemsets as a function of ρ

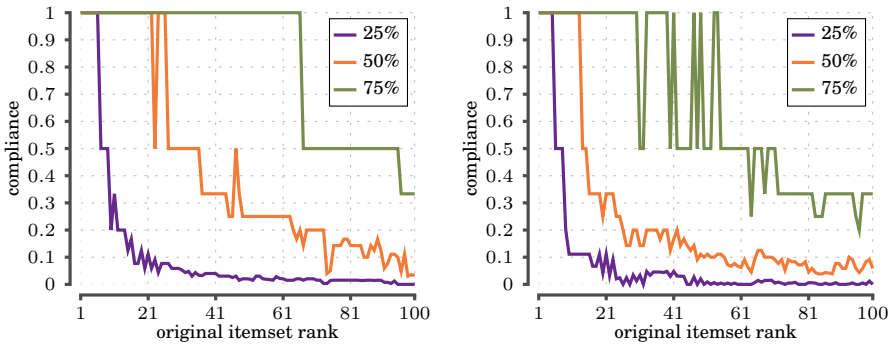
6.3. Effect of noise for robust closed itemsets

Our next experiment is to see how robust closed itemsets behave when a dataset is exposed to noise. Our expectation is that most robust itemsets will stay closed and be ranked higher while the ranking of the less robust itemsets will be more susceptible to noise.

In order to do this, we created from each dataset a synthetic dataset having the same dimensions by sampling from a distribution. The underlying distribution had the same margins as the original data but otherwise items were independent. We then mix the original data with the synthetic one, that is, an entry in a mixed dataset is an entry from the synthetic dataset with the probability η, and is an entry from the original dataset with the probability $1 - \eta$. We tested two different noise levels $\eta = 0.05$ and $\eta = 0.1$.

We mined approximately $10\,000$ frequent closed itemsets from each original dataset. If the dataset contained less than $10\,000$ itemsets, we set the threshold to one transaction. Using the same thresholds we mined closed itemsets from the mixed datasets. We sorted the itemsets using Algorithm 1.

Let X be an itemset ranked ith in the original data. Assume that X is ranked jth in the noisy data. We define compliance of X by $1/(|i - j| + 1)$. The compliance will be 1 if $i = j$ and decreases to 0 the longer is the distance. The reason for using this particular definition is that we can naturally set compliance to 0 if X is not found in the noisy data. The compliances for top-100 itemsets are given in Figure 4.



(a) noise $\eta = 0.05$ (b) noise $\eta = 0.1$

Fig. 4. Rank compliance of an itemset in a noisy data as a function of robustness in the original data. High compliance value imply that adding noise had little effect on the rank of an itemset. Median and quartiles are computed over all datasets.

From the figures we see that compliance stay high for robust itemsets and drop as we move further down the original ranking. That is, the more robust an itemset is, the less prone to noise it is. Adding more noise to the data implies less compliance. For example, for noise level $\eta = 0.05$, top-60 itemsets had a compliance of 0.25 or higher in half of the datasets. This means that their rank changed only by 3. On the other hand for noise level $\eta = 0.1$, top-50 itemsets had a compliance of 0.1 or higher in half of the dataset, in other words, ranks changed by 9.

6.4. Ranking without α

Our next experiment was to compare the parameter-free ranking described in Section 4 against the rankings based on quantitative robustness given specific values of α. We

expect that rankings are similar for large α values and difference increase when we lower α. For comparison we used the number of discordant pairs to calculate a distance of the rankings similar to Kendall's τ. A discordant pair is a pair of itemsets (X, Y) such that the first method ranks X higher than Y and the second method ranks Y higher than X. We normalize the number of observed discordant pairs by multiplying by $100/b$, where b is the maximum number of discordant pairs. Hence, we obtain a value between 0 and 100. If there are no ties in robustness, then $b = N(N-1)/2$, where N is the number of itemsets. However if ties are presented, that is, the robustness induces a bucket order, then $b = N(N-1)/2 - \sum_{i=1} B_i(B_i - 1)/2$, where B_i is the size of each bucket, set of itemsets having the same robustness. Values close to 0 mean that rankings are in agreement.

Typical examples are given in Table II for the *Mushroom* and *Zoo* datasets, along with the averages taken over all datasets. Surprisingly, the ranking distance is extremely small even for small values of α showing that the parameter free approach produces rankings similar to rankings under most α. Starting at $\alpha = 0.5$ for *Mushroom* and all α for *Zoo* only about 1% of pairs are discordant. We see that values increase as we lower α which is expected since the parameter-free approach is based on large α values.

Table II. Distance between parameter-free rankings and rankings based on α for *Mushroom* and *Zoo* datasets. Low values imply that rankings agree. Value range is 0–100.

	Mushroom ($\tau = 0.05$)			*Zoo* ($\tau = 0.01$)			All datasets		
α	free	ts	nd	free	ts	nd	free	ts	nd
0.1	0.30	0.82	0.39	14.91	7.26	7.62	4.35	4.25	2.80
0.2	0.078	0.27	0.089	10.01	8.69	4.31	2.59	2.67	2.11
0.3	0.017	0.11	0.044	5.94	5.40	2.23	1.46	1.63	1.45
0.4	0.0016	0.050	0.015	2.84	2.77	1.84	0.69	1.03	0.91
0.5	0.000032	0.027	0.0022	1.12	1.31	1.11	0.26	0.56	0.52
0.6	0.0000050	0.016	0.0023	0.32	0.58	0.56	0.082	0.25	0.27
0.7	0	0.0020	0.0011	0.017	0.20	0.25	0.013	0.073	0.11
0.8	0	0.0022	0	0	0	0.013	0.000074	0.013	0.031
0.9	0	0	0	0	0	0	0	0.00049	0.0039

6.5. Top-k closed and free itemsets

Closed itemsets are often used for tasks requiring interpretation of the itemsets, because as maximum elements of an equivalence class they offer the most detailed description. We studied the highest ranked closed itemsets for text datasets that are easily understood without domain knowledge. As an illustrative example, we used the *re0* news dataset from which we mine $2\,493$ closed itemsets with minimum support $\tau = 0.05$. We ordered these itemsets using the estimation technique given in Section 5.2 and list the top 45 itemsets in Table III. The ranking is different from the one using support, less frequent (but more robust) itemsets are commonly ranked higher that frequent itemsets. For example, 'bank pct rate' occurs before the much more frequent itemset 'bank pct' showing that 'bank pct' is only closed in the full dataset due to relatively few documents using it without also using 'rate'.

Finally, we considered an alternative order by ranking itemsets based on how free they are. Note that a closed itemset is robust if the same transactions cannot be explained by a superset whereas a free itemset is robust if the same transactions cannot be explained by a subset. For example, a free singleton X will be ranked higher than singleton Y if X has *lower* support. The reason for this is that it requires less transactions to be removed in order to make Y non-robust, namely the transactions not containing Y. We present the top-45 free non-singleton itemsets from *re0* news dataset in Table IV.

Table III. Top-45 closed itemsets from *re0* ($\tau = 0.05$) dataset.

1.	pct	792	16.	week	310	31.	canada	117
2.	bank	702	17.	pct earlier	127	32.	pct month	261
3.	trade	485	18.	japan	318	33.	econom	295
4.	billion	552	19.	trade current	126	34.	billion dlr mln	116
5.	market	554	20.	dlr	472	35.	told bank	116
6.	billion dlr	346	21.	bank pct rate	287	36.	told nation	116
7.	offici	342	22.	dollar	336	37.	pct japan	115
8.	mln	420	23.	statem	122	38.	pct adjust	115
9.	nation	323	24.	committe	121	39.	billion current	115
10.	rate	566	25.	nation month	121	40.	european	114
11.	bank market	369	26.	ministri	120	41.	month japan	114
12.	foreign	331	27.	pct rise	269	42.	bank ad market	114
13.	pct figur	132	28.	bank pct	407	43.	action	114
14.	pct rate	418	29.	pct rate feb	119	44.	trade world	114
15.	month	391	30.	lead	118	45.	nation japan	114

These are frequent item pairs ab such that $sp(ab) \ll sp(a)$ and $sp(ab) \ll sp(b)$, that is, a non-robust free item pair ab would be such that if we would remove a singleton a (or b), then roughly the same transactions will still cover the pattern. An example of such non-robust free itemset is *bank assist*. This itemset is ranked as 2 465 out of 2 558 itemsets. The support of this itemset is 96 but the support of *assist* is 98, consequently there are only two documents in which *assist* occurs but not *bank*.

Table IV. Top-45 free non-singleton itemsets from *re0* ($\tau = 0.05$) dataset.

1.	billion rate	165	16.	bank billion	287	31.	govern dollar	92
2.	rate dlr	132	17.	billion pct	288	32.	foreign februari	87
3.	trade rate	146	18.	rise dlr	118	33.	januari dlr	81
4.	trade bank	154	19.	pct dlr	210	34.	monei dlr	81
5.	billion market	228	20.	trade billion	223	35.	dollar februari	89
6.	rate mln	109	21.	bank dlr	211	36.	rise japan	78
7.	trade pct	176	22.	govern februari	77	37.	februari japan	78
8.	bank pct	407	23.	govern mln	90	38.	dollar offici	96
9.	market rate	262	24.	month dlr	141	39.	rise offici	104
10.	trade mln	130	25.	trade dlr	222	40.	pct mln	184
11.	market dlr	186	26.	pct market	306	41.	februari dlr	94
12.	month mln	106	27.	market rise	133	42.	foreign mln	95
13.	trade market	203	28.	februari offici	83	43.	nation februari	89
14.	rise mln	109	29.	govern monei	77	44.	januari mln	90
15.	trade rise	115	30.	januari govern	77	45.	monei month	90

7. DISCUSSION

The experiments have shown that the number of itemsets can be largely reduced on many datasets when requiring a certain robustness. The fact that the results vary by dataset are another indication of the well known fact that itemset data with different structures (dense vs. sparse, many items vs. many transactions) behave very differently in mining tasks.

We conjecture that robust itemsets can be beneficial for post-processing techniques such as [Bringmann and Zimmermann 2009] or [Vreeken et al. 2011] that use itemsets as their input and remove redundancy in the pattern set. Robust itemsets can be used as an alternative input reducing their runtime without sacrificing performance. Also, robust itemsets could be used instead of closed-itemsets as seeds to the AC-Close algorithm for approximate itemset mining [Cheng et al. 2006] improving its efficiency that was criticized by Gupta et al. [2008].

The ranking of itemsets by robustness presents a new interestingness measure that can be used to choose the top-k itemsets for interpretation or other data mining tasks. The intuition of robustness should be easy to understand for analysts but which ranking is better for specific data mining tasks remains to be studied.

In particular it will be interesting to evaluate performance as features for classification tasks in contrast to direct mining of prediction tasks. For interpretable classifiers one would want itemsets to be long, thus use closed patterns. On the other hand the desire is for an itemset to be present in unseen data with high likelihood, so free itemset as the minimal elements of an equivalence class may generalize better. For both patterns we can ensure that they are present in many subsets of the training without actually sampling, potentially alleviating the need for nested cross validation.

8. SUMMARY

We have shown how robustness under subsampling for common classes of itemsets can be computed efficiently without actually sampling the data. The experimental results show that the number of reported itemsets can be largely reduced on many datasets, in other words spurious itemsets that would not have been found in many subsets of the data are removed. The approach can further be used to rank itemsets for top-k mining by robustness. Future work will investigate the effect of using robust itemsets on data mining tasks such as clustering, classification, and rule generation using itemsets.

REFERENCES

AGRAWAL, R., IMIELINSKI, T., AND SWAMI, A. N. 1993. Mining association rules between sets of items in large databases. In *SIGMOD*. 207–216.

ASUNCION, A. AND NEWMAN, D. 2007. UCI machine learning repository.

BOULICAUT, J.-F., BYKOWSKI, A., AND RIGOTTI, C. 2000. Approximation of frequency queries by means of free-sets. In *PKDD*. 75–85.

BOULICAUT, J.-F., BYKOWSKI, A., AND RIGOTTI, C. 2003. Free-sets: A condensed representation of boolean data for the approximation of frequency queries. *DMKD 7,* 1, 5–22.

BRIN, S., MOTWANI, R., AND SILVERSTEIN, C. 1997. Beyond market baskets: Generalizing association rules to correlations. In *SIGMOD*. 265–276.

BRINGMANN, B. AND ZIMMERMANN, A. 2009. One in a million: picking the right patterns. *KAIS 18,* 1, 61–81.

CALDERS, T. AND GOETHALS, B. 2007. Non-derivable itemset mining. *DMKD 14,* 1, 171–206.

CALDERS, T., GOETHALS, B., AND MAMPAEY, M. 2007. Mining itemsets in the presence of missing values. In *SAC*. 404–408.

CALDERS, T., RIGOTTI, C., AND BOULICAUT, J.-F. 2006. A survey on condensed representations for frequent sets. In *Constraint-Based Mining and Inductive Databases*. 64–80.

CHENG, H., YAN, X., HAN, J., AND HSU, C. 2007. Discriminative frequent pattern analysis for effective classification. In *ICDE*. 716–725.

CHENG, H., YU, P. S., AND HAN, J. 2006. AC-Close: Efficiently mining approximate closed itemsets by core pattern recovery. In *ICDM*. IEEE, 839–844.

COENEN, F. 2003. The LUCS-KDD discretised/normalised ARM and CARM data library.

DE BIE, T. 2011. Maximum entropy models and subjective interestingness: an application to tiles in binary databases. 1–40.

GALLO, A., DE BIE, T., AND CRISTIANINI, N. 2007. Mini: Mining informative non-redundant itemsets. In *ECMLPKDD*. 438–445.

GEERTS, F., GOETHALS, B., AND MIELIKÄINEN, T. 2004. Tiling databases. In *Proc. Discovery Science*. 278–289.

GHOSHAL, G. AND BARABÁSI, A.-L. 2011. Ranking stability and super-stable nodes in complex networks. *Nature Communications 2*.

GIONIS, A., MANNILA, H., MIELIKÄINEN, T., AND TSAPARAS, P. 2007. Assessing data mining results via swap randomization. *TKDD 1,* 3.

GOETHALS, B. AND ZAKI, M. 2003. FIMI '03, frequent itemset mining implementations. In *ICDM 2003 Workshop, FIMI*.

GUPTA, R., FANG, G., FIELD, B., STEINBACH, M., AND KUMAR, V. 2008. Quantitative evaluation of approximate frequent pattern mining algorithms. In *KDD*. 301–309.

HANHIJÄRVI, S., OJALA, M., VUOKKO, N., PUOLAMÄKI, K., TATTI, N., AND MANNILA, H. 2009. Tell me something I don't know: randomization strategies for iterative data mining. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2009)*. 379–388.

HIPP, J., GÜNTZER, U., AND NAKHAEIZADEH, G. 2000. Algorithms for association rule mining - a general survey and comparison. *SIGKDD Explorations 2,* 1, 58–64.

LUCCESE, C., ORLANDO, S., AND PEREGO, R.CASAS-GARRIGA, G. 2010. Mining top-k patterns from binary datasets in presence of noise. In *ICDM*.

MIELIKÄINEN, T. 2005. Transaction databases, frequent itemsets, and their condensed representations. In *KDID*. 139–164.

MISRA, G., GOLSHAN, B., AND TERZI, E. 2012. A framework for evaluating the smoothness of data-mining results. In *ECMLPKDD 2012*. 660–675.

MOERCHEN, F., THIES, M., AND ULTSCH, A. 2010. Efficient mining of all margin-closed itemsets with applications in temporal knowledge discovery and classification by compression. *KAIS*.

PASQUIER, N., BASTIDE, Y., TAOUIL, R., AND LAKHAL, L. 1999. Discovering frequent closed itemsets for association rules. In *ICDT*. 398–416.

PEI, J., HAN, J., AND LAKSHMANAN, L. V. S. 2001. Mining frequent itemsets with convertible constraints. In *ICDE*. 433–442.

SMETS, K. AND VREEKEN, J. 2011. The odd one out: Identifying and characterising anomalies. In *SDM*.

TATTI, N. 2008. Maximum entropy based significance of itemsets. *KAIS 17,* 1, 57–77.

TATTI, N. AND MOERCHEN, F. 2011. Finding robust itemsets under subsampling. In *11th IEEE International Conference on Data Mining, ICDM 2011*. 705–714.

UNO, T. AND ARIMURA, H. 2007. An efficient polynomial delay algorithm for pseudo frequent itemset mining. In *Discovery Science*. Springer, 219–230.

VREEKEN, J., VAN LEEUWEN, M., AND SIEBES, A. 2011. Krimp: mining itemsets that compress. *DMKD 23,* 1, 169–214.

WANG, K., XU, C., AND LIU, B. 1999. Clustering transactions using large items. In *CIKM*. 483–490.

WEBB, G. I. 2007. Discovering significant patterns. *Mach. Learn. 68,* 1, 1–33.

XIN, D., HAN, J., YAN, X., AND CHENG, H. 2005. Mining compressed frequent-pattern sets. In *VLDB*. 709–720.

ZHAO, Y. AND KARYPIS, G. 2002. Evaluation of hierarchical clustering algorithms for document datasets. In *CIKM*. 515–524.