FSSD - A Fast and Efficient Algorithm for Subgroup Set Discovery

Adnene Belfodil, Aimene Belfodil, Anes Bendimerad, Philippe Lamarre, Céline Robardet, Mehdi Kaytoue, Marc Plantevit

FSSD - A Fast and Efficient Algorithm for Subgroup Set Discovery

Adnene Belfodil[*,1], Aimene Belfodil[*,1,2], Anes Bendimerad[1],
Philippe Lamarre[1], Céline Robardet[1], Mehdi Kaytoue[1,3], Marc Plantevit[4]

[1] Univ Lyon, INSA Lyon, CNRS, LIRIS UMR 5205, F-69621, LYON, France
[2] Mobile Devices Ingeniérie, 100 Avenue Stalingrad, 94800, Villejuif, France
[3] Infologic, 99 avenue de Lyon, 26500 Bourg-Lès-Valence, France
[4] Univ Lyon, CNRS, LIRIS UMR 5205, F-69621, LYON, France
firstname.surname@liris.cnrs.fr

***Abstract*—Subgroup discovery (SD) is the task of discovering interpretable patterns in the data that stand out w.r.t. some property of interest. Discovering patterns that accurately discriminate a class from the others is one of the most common SD tasks. Standard approaches of the literature are based on local pattern discovery, which is known to provide an overwhelmingly large number of redundant patterns. To solve this issue, pattern set mining has been proposed: instead of evaluating the quality of patterns separately, one should consider the quality of a pattern set as a whole. The goal is to provide a small pattern set that is diverse and well-discriminant to the target class. In this work, we introduce a novel formulation of the task of diverse subgroup set discovery where both discriminative power and diversity of the subgroup set are incorporated in the same quality measure. We propose an efficient and parameter-free algorithm dubbed `FSSD` and based on a greedy scheme. `FSSD` uses several optimization strategies that enable to efficiently provide a high quality pattern set in a short amount of time.**

***Index Terms*—Pattern Mining, Subgroup Discovery.**

I. Introduction

Data science can be seen as a language that allows, among others, the analysis of (large amount of) data [43]. Following the same philosophy of Exploratory Data Analysis (EDA) [47], one of the main tasks of data science is the discovery of interpretable and understandable patterns in the data, a task known as data mining [11]. Subgroup discovery (SD) [27], [50] is a popular data mining task that aims, amongst many other possibilities, to discover patterns describing regions in a dataset where the distribution of the target variable significantly deviates from the norm. For instance, a subgroup defined by the description $age \geq 65 \wedge smoker = true$ fosters the target variable "*lung cancer*" prevalence in some *patient-dataset*.

In SD, as in most of local pattern mining approaches, an important problem that one has to consider is the huge number of patterns (subgroups) that can be produced. Many of these subgroups are redundant and convey very similar information. This issue notably becomes more serious when the number and/or the domain of attributes is large. This challenge led to the definition of the *pattern set mining problem* [8], [20], [28], [40]. While *local pattern mining* seeks for patterns where each of them satisfies local constraints individually, *pattern set mining* aims to find a small set of patterns that together satisfy global constraints. These global constraints are used to guarantee the diversity in the returned pattern set. Different approaches exist in the literature to tackle the problem of pattern set mining:

(1) Non-diversified top-k approach. Such algorithms output the top-k best subgroups (e.g., `BSD` [32] and `MiSoSouP` [41]). Although this allows to control the number of returned patterns, this does not solve the redundancy problem since the top-k patterns can be very similar and describe only a few of all the local optima (lack of diversity).

(2) Exhaustive approach. Such methods aim to find the best possible solution. The pioneering work of [40] proposes a level wise algorithm to explore the search space, and [21], [26], [38] use a *Constraint Programming (CP)* solver to retrieve the k-pattern set. In large and complex datasets, the exhaustive selection of the best k-pattern set becomes unfeasible, even if pruning strategies are used. As the number of possible pattern sets is exponential in the size of the set of local patterns, which is itself huge, the computational efficiency of pattern set mining is a very challenging task. In fact, the general problem of pattern set discovery is NP-hard [35].

(3) Two-step approach. These algorithms start by (a) generating a collection of local patterns by some exhaustive or heuristic technique followed by (b) a heuristic selection of a smaller subset of complementary patterns [5], [8], [44]. However, since the number of local patterns can be large, these algorithms need a huge amount of memory to store them. Moreover, post-processing the set of all mined local patterns can be time-consuming. Noteworthy algorithms belonging to this family are: `DSSD` [48] and `MCTS4DM` [7].

(4) Sequential covering approach. Algorithms proposed in [9], [31], [46] use the so-called *sequential covering* strategy for subgroup set discovery. These algorithms repeatedly and heuristically (i.e. beam search) look for one subgroup; add it to the already constituted subgroup set; then re-weight or

[*] Both authors contributed equally to this work.

delete the already covered objects in the database in the aim of reducing their impact in the next iterations. This approach allows to implicitly consider the diversity of the produced subgroup set, conversely to some two-step approaches where an additional "diversity" measure is used.

Contributions. In this paper, we investigate the problem of *subgroup set discovery*, i.e. "discovery of a non-redundant set of high-quality subgroups." [48]. This can be seen as a particular instance of the more general task of *pattern set mining*. After formally introducing the notion of *SD-compatible* quality measures (e.g. WRAcc) that locally evaluate subgroups interestingness, we extend their definitions to subgroup sets. This allows to evaluate the interestingness of a subgroup set in terms of its discriminative power and diversity. We propose a greedy scheme to maximize these measures by incrementally augmenting the subgroup set with subgroups providing the highest *marginal contribution*. In this respect, we devise a branch-and-bound algorithm, dubbed FSSD, a *parameter-free* and *memory efficient* algorithm tailored for diverse subgroup set discovery. It exploits (i) *tight optimistic estimates* on the marginal contributions of SD-compatible measures to prune unpromising branches, (ii) closure operators to efficiently explore the search space and (iii) successive removal of already covered objects (i.e. *sequential covering*) to significantly reduce candidate subgroups.

Outline. The remainder of this paper is organised as follows. Sec. II recalls preliminary notions. Sec. III introduces the problem of k-diverse subgroup set discovery. Sec. IV presents the generic greedy framework. Sec. V introduces FSSD. While the approximation ratio to the best possible solution of the proposed problem is not theoretically bounded, Sec. VI reports a thorough empirical study that demonstrates FSSD efficiency and effectiveness. Sec. VII concludes the paper.

II. Preliminaries

In this paper, for any set E, $\wp(E)$ denotes the powerset of E. For $f : E \to F$ and a subset $A \subseteq E$, the image of A by f is denoted by $f[A]$ (i.e. $f[A] = \{f(a) \mid a \in A\}$). Details on order-theoretic notions can be found in [42].

A. Input dataset.

A *dataset* is a pair $(\mathcal{G}, \mathcal{A})$ where $\mathcal{G}$ is a set of objects and $\mathcal{A} = \{a_i\}_{1 \leq i \leq m}$ is an ordered set of attributes. Each attribute $a \in \mathcal{A}$ can be seen as a mapping $a : \mathcal{G} \to \mathcal{R}_a$ where $\mathcal{R}_a$ is called the *domain* of the attribute. For instance, $\mathcal{R}_a$ is given by $\mathbb{R}$ if a is *numerical*, by a finite set of categories C_i if a is *nominal* (*categorical*) or by $\{0, 1\}$ if a is *Boolean*. Each object $g \in \mathcal{G}$ is labeled as either *positive* or *negative*. It means that $\mathcal{G}$ is partitioned into two subsets $\mathcal{G}^+$ and $\mathcal{G}^-$ enclosing respectively positive and negative instances. The proportion $\alpha := |\mathcal{G}^+|/|\mathcal{G}|$ is called the positive prevalence. Fig. 1 (left) depicts a dataset containing 3 attributes of distinct types where $\mathcal{G}^+ = \{g_2, g_3, g_5\}$ is the set of individuals having a high income.

B. Pattern Language

A *pattern* d is a *constrained selector* of a subset of objects of the dataset using their attribute values. We refer to the set of all possible patterns that we want to explore by the *pattern language* and we denote it $\mathcal{D}$. In our case, $D = \bigtimes_{i=1}^{m} \mathcal{D}_i$ where $\mathcal{D}_i$ is given by the set of all possible intervals in $\mathbb{R}$ if a_i is numerical, the set $\{C_i, \emptyset\} \cup \{\{c\} \mid c \in C_i\}$ if a_i is nominal or $\{\{0, 1\}, \{1\}\}$ if a_i is Boolean (i.e. if negative item $\{0\}$ needs to be considered, attribute a_i needs to be considered as nominal). A pattern $d \in \mathcal{D}$ is then given by a sequence of restrictions over each attribute (i.e. $d = (d_i)_{1 \leq i \leq m}$). For example, in Fig. 1 (right) pattern $d = (\{0, 1\}, \{M\}, \mathbb{R})$ is read "**individual that are** ~~married or not,~~ **men** ~~and having any age~~". Note that we considered here the exhaustive search space for the numerical attributes as in [7], [25] and no *apriori* discretization is performed.

As when we deal with itemsets, patterns in $\mathcal{D}$ are ordered from the most general one to the most restrictive one by an order relation $\sqsubseteq$. In our case, for two patterns $c = (c_i)_{1 \leq i \leq m} \in \mathcal{D}$ and $d = (d_i)_{1 \leq i \leq m} \in \mathcal{D}$, we have: $c \sqsubseteq d \Leftrightarrow \forall i \in [\![1, m]\!] \, (c_i \supseteq d_i)$. In Fig. 1 (right), pattern $c = (\{0, 1\}, \{M\}, \mathbb{R})$ is less restrictive than pattern $d = (\{0, 1\}, \{M\}, [27, 65])$ (i.e. $c \sqsubseteq d$) since d has an additional constraint on the age attribute. One can show that the partially ordered set $(\mathcal{D}, \sqsubseteq)$ is a meet-semilattice (the meet is denoted $\sqcap$). It means that for all $c, d \in \mathcal{D}$, there is maximum common lower bound $c \sqcap d$ (i.e. $(\forall e \in \mathcal{D}) \, e \sqsubseteq c \wedge e \sqsubseteq d \Leftrightarrow e \sqsubseteq c \sqcap d$).

Given the pattern language explained above, we should link objects in $\mathcal{G}$ to descriptions in $\mathcal{D}$. This is done by the *cover* relationship. A description $d \in \mathcal{D}$ is said to *cover* $g \in \mathcal{G}$ iff $\forall i \in [\![1, m]\!] : a_i(g) \in d_i$. Each object $g \in \mathcal{G}$ is now mapped to the most-restrictive description $\delta(g) \in \mathcal{D}$ covering it: $\forall d \in \mathcal{D}$, d *covers* g iff $d \sqsubseteq \delta(g)$. Fig. 1 illustrates this mappings between objects and descriptions.

C. Pattern Structure

Pattern structure is a generalization of the Formal Concept Analysis (FCA) framework [13] to complex attributes such as numerical and nominal ones (see [14] for further details). Since $(\mathcal{D}, \sqsubseteq)$ is a meet-semilattice, the triple $(\mathcal{G}, (\mathcal{D}, \sqsubseteq), \delta)$ is pattern structure. This triple contains all information we need to search for patterns in a dataset and allows us to use an algorithm like Close-by-One (CbO) [29] to exhaustively enumerate them. Two mappings associated to a pattern structure are important to keep in mind: (1) $ext : \mathcal{D} \to \wp(\mathcal{G}), d \mapsto \{g \in \mathcal{G} \mid d \sqsubseteq \delta(g)\}$ called the *extent*. It associates to each pattern $d \in \mathcal{D}$ the set of objects in $\mathcal{G}$ for which d hold. (2) $int : \wp(\mathcal{G}) \to \mathcal{D}, E \mapsto \bigsqcap_{g \in E} \delta(g)$ associates to each subset $E \subseteq \mathcal{G}$ the most-restrictive pattern $d \in \mathcal{D}$ covering them. Note that ext is order-reversing: i.e. if $c \sqsubseteq d$ then $ext(d) \subseteq ext(c)$ (i.e. if d is more restrictive than c, then d covers less objects than c), property from which (anti)-monotonicity of some measures ensues. Note also that $int \circ ext$ and $ext \circ int$ are closure operators since the pair (ext, int) forms a Galois connection.

	$married?$	sex	age	**income**
g_1	×	I	23	$\leq 50k$
$\mathbf{g_2}$	×	M	27	$> 50k$
$\mathbf{g_3}$		M	65	$> 50k$
g_4		F	54	$\leq 50k$
$\mathbf{g_5}$	×	F	43	$> 50k$
g_6		M	13	$\leq 50k$

	mapping δ	**income**
g_1	$(\{1\},\{I\},[23,23])$	$\leq 50k$
$\mathbf{g_2}$	$(\{1\},\{M\},[27,27])$	$> 50k$
$\mathbf{g_3}$	$(\{0,1\},\{M\},[65,65])$	$> 50k$
g_4	$(\{0,1\},\{F\},[54,54])$	$\leq 50k$
$\mathbf{g_5}$	$(\{1\},\{F\},[43,43])$	$> 50k$
g_6	$(\{0,1\},\{M\},[13,13])$	$\leq 50k$

Descriptions and their extents:

$c = (\{0,1\},\{M\},\mathbb{R}), ext(c) = \{g_2, g_3, g_6\}$

$d = (\{0,1\},\{M\},[27,65]), ext(d) = \{g_2, g_3\}$

Fig. 1: A labeled dataset **(left)**, objects and their descriptions **(center)** and some descriptions in $\mathcal{D}$ **(right)**

III. Problem Definition

A. Subgroups and their interestingness.

A *subgroup* s is any subset of objects in $\mathbb{S} = ext[\mathcal{D}] = \{ext(d) \mid d \in \mathcal{D}\}$. We choose to describe subgroups by their *intents* since they provide the most specific (most insightful) description. For instance, in Figure 1, the set $\{g_2, g_3\}$ is a subgroup whose intent is $d = (\{0,1\},\{M\},[27,65])$. Indeed, any refinement of the description d will drop at least one object.

A measure $\phi : \wp(\mathcal{G}) \times \wp(\mathcal{G}) \to \mathbb{R}; (s, U) \mapsto \phi(s, U)$ defines a mapping that evaluates some property of a subgroup $s \in \wp(\mathcal{G})$ in the sub-dataset $U \subseteq \mathcal{G}$. For the sake of clarity, if the second parameter U is omitted, then U is the whole set of objects $\mathcal{G}$. The very basic measure is the relative support : $relsup : s, U \mapsto |s \cap U|/|U|$.

The relative support is monotonic w.r.t. parameter 1 (i.e. $s \subseteq t \Rightarrow relsup(s, U) \leq relsup(t, U)$). Using the relative support, the *true positive rate* (tpr) and the *false positive rate* (fpr) of a subgroup s are defined respectively as follows $tpr(s, U) = relsup(s, \mathcal{G}^+ \cap U)$ and $fpr(s, U) = relsup(s, \mathcal{G}^- \cap U)$. Every objective measure ([17]) can be written using tpr, fpr and some constants. Properties of measures w.r.t. the task have been thoroughly studied in the literature [12], [24], [33], [39]. For a SD task, we define *SD-compatible* measures.

Definition 1. A measure ϕ is said to be *SD-compatible* if the following property $\forall s_1, s_2 \in \mathbb{S}$: if $tpr(s_1) \geq tpr(s_2)$ and $fpr(s_1) \leq fpr(s_2)$ then $\phi(s_1) \geq \phi(s_2)$. In other words, if s_1 dominates s_2 in the ROC space (i.e. tpr-vs.-fpr space), then $\phi(s_1) \geq \phi(s_2)$.

Almost all usual measures in SD are SD-compatibles. For instance, Weighted Relative Accuracy (WRAcc), Accuracy, correlation coefficient, Cohen's kappa, m-estimate, G-measure and F_β-measure among others are SD-compatible (see Appendix C). Please note that, preferably, other properties must hold for a measure in order to be *useful* for a SD task (e.g. constant in independence case, i.e. $tpr = fpr$ [33]).

B. Subgroup Sets Interestingness.

A *subgroup set* is a set $\mathcal{S} \subseteq \mathbb{S}$. Following [40], subgroup sets are interpreted as *disjunctions of subgroups*. Accordingly, the set of objects covered by a *subgroup set* is given by the union of its subgroups. To evaluate the interestingness of a *subgroup set* $\mathcal{S}$, we extend the quality measure ϕ definition as follows $\phi(\mathcal{S}) = \phi\left(\bigcup_{s \in \mathcal{S}} s\right)$ since the description associated to $\mathcal{S}$ is the disjunction of the intent of each subgroup. Such a formulation gives higher quality $\phi(\mathcal{S})$ to subgroups in $\mathcal{S}$ covering different regions.

C. Problem - Subgroup Set Discovery.

Problem Statement. Let be a dataset $(\mathcal{G}, \mathcal{A})$, a cardinality constraint $k \in \mathbb{N}$ and a SD-compatible measure ϕ. Output one subgroup set from $\text{argmax}_{\mathcal{S} \subseteq \mathbb{S}, |\mathcal{S}| \leq k} \phi(\mathcal{S})$.

To the best of the authors' knowledge, this is a novel way to present the problem of diverse subgroup set discovery. It has the advantage that both discriminative power and diversity of the subgroup set are incorporated into the same measure. It is worth to note that Problem III-C, even when $\mathbb{S}$ is considered as an input, is NP-Hard. For the convenience of the reader, we delay the proof to Appendix A. One naive (exact) solution for this problem is: (1) look for all possible subgroups in the first phase then (2) test all possible sets of subgroups which size is below k. However, such a solution is practically unfeasible.

IV. A Greedy Solution

To approximate the exact solution of problem III-C, we rely on a greedy optimization algorithm. Before presenting such a solution, we present below in Definition 2 the notion of marginal contribution associated to some quality measure.

Definition 2. For a measure ϕ and a subgroup set $\mathcal{S} \subseteq \mathbb{S}$, the *marginal contribution* to $\mathcal{S}$ is the function ϕ_S:

$$\phi_{\mathcal{S}} : \wp(\mathcal{G}) \to \mathbb{R}; s \mapsto \phi\left(\bigcup \mathcal{S} \cup s\right) - \phi\left(\bigcup \mathcal{S}\right)$$

The marginal contribution $\phi_{\mathcal{S}}$ quantifies the quality ϕ gain that a subgroup s brings to the subgroup set $\mathcal{S}$ if added.

A. The Greedy Scheme.

Algorithm 1 presents the general scheme of a greedy solution [36]. It starts by an empty subgroup set (line 1). Next, it incrementally constitutes the subgroup set $\mathcal{S}$ by successively adding the subgroup s^* providing the top marginal contribution (line 3). Algorithm 1 stops when $|\mathcal{S}| = k$ or there is no subgroup providing a positive marginal contribution.

When studying the quality of a greedy approach solution, one need to evaluate the so-called *approximation ratio*. However, since quality measures $\phi : \wp(\mathbb{S}) \to \mathbb{R}$ extended to

Algorithm 1: Greedy solution for Problem III-C

Input: $(\mathcal{G}, \mathcal{A})$ labeled dataset, $k \in \mathbb{N}$, a measure ϕ

1 $\mathcal{S} \leftarrow \{\}$

2 **while** $|\mathcal{S}| < k$ **do**

3 Look for $s^* \in \text{argmax}_{s \in \mathbb{S}} \phi_{\mathcal{S}}(s)$

4 **if** $\phi_{\mathcal{S}}(s^*) > 0$ **then** $\mathcal{S} \leftarrow \mathcal{S} \cup \{s^*\}$ **else Break**

5 **return** $\mathcal{S}$

	a	b	c	**class**
g_1	$\times$			$+$
g_2		$\times$		$+$
g_3		$\times$		$+$
g_4	$\times$	$\times$		$-$
g_5	$\times$	$\times$		$-$
g_6			$\times$	$-$

TABLE I: Dataset with boolean descriptive attributes

subgroup sets are not necessarily sub-modular [36], one cannot use the usual lower bound on the approximation ratio, i.e. $1 - \frac{1}{e}$.

Definition 3. Let ϕ be a measure, k be the cardinality constraint and S_{greedy} be the output of Algorithm 1 w.r.t. k. The approximation ratio ρ of S_{greedy} is given by:

$$\rho(S_{greedy}) = \frac{\phi(\mathcal{S}_{greedy})}{\max_{\mathcal{S}\in\mathbb{S},|\mathcal{S}|\leq k}\phi(S)}$$

N.B. $\max_{\mathcal{S}\in\mathbb{S},|S|\leq k}\phi(\mathcal{S}) = 0 \Rightarrow \rho(\mathcal{S}_{greedy}) = 1$.

Unfortunately, the quality of the greedy solution cannot be theoretically guaranteed (i.e. lower-bounded) for the general formulation of Problem III-C. Indeed, consider Problem III-C with dataset $(\mathcal{G}, \mathcal{A})$ depicted in Table I, WRAcc quality measure and $k = 2$. We consider the itemset search space (i.e. attributes in $\mathcal{A}$ are Booleans). Hence, the description language is (isomorphic to) $\wp(\{a, b, c\})$. For an itemset $i \in \wp(\{a, b, c\})$, we denote by s_i the subgroup $ext(i)$. One can show that for all itemsets $i \in \wp(\{a, b, c\})$, we have $tpr(s_i) \leq fpr(s_i)$. Thus, $\mathrm{WRAcc}(s_i) \leq 0$. Algorithm 1 outputs thus $\mathcal{S}_{greedy} = \emptyset$ since the top gain is below 0. Let be the $2-$sized subgroup set $\mathcal{S} = \{s_{\{a\}}, s_{\{b\}}\}$. We have $\mathrm{WRAcc}(\mathcal{S}) > 0$ since $tpr(\mathcal{S}) > fpr(\mathcal{S})$. Hence, the top subgroup set $\mathcal{S}^*$ has $\mathrm{WRAcc}(\mathcal{S}^*) > 0$. Then $\rho(\mathcal{S}_{greedy}) = 0$. Other measures are concerned by the non-existence of the theoretical bound on ρ such as Klösgen measure, linear correlation coefficient and Cohen's Kappa. Here, we particularly considered Problem III-C instance on a boolean dataset. We show in Appendix B that no approximation guarantee can be ensured by Algorithm 1 when attribute-value datasets are considered, i.e. categorical and numerical attributed datasets.

V. FSSD - Fast and Efficient Subgroup Set Discovery

There are several ways to implement line 3 in Algorithm 1. The easiest one is to look for all subgroups in the dataset using some known pattern mining algorithm (for example SD-Map [2]) beforehand. We refer to such a solution by BASELINE algorithm. This BASELINE algorithm suffers from many drawbacks. For instance, it needs to store all found patterns in memory before post-processing them at the end. We need to find hence a fast and memory-efficient way to implement line 3; i.e. the step of finding the top-gain subgroup. If we do not want to store all found subgroups before post-processing them, one should explore at each step of the algorithm the whole search space $\mathbb{S}$. Therefore, in order to optimize line 3, one need to look for less subgroups (i.e. explore subgroups in some $\mathbb{S}_i \subseteq \mathbb{S}$) at each iteration $i \in 1...k$ to find the top-gain one. Starting from here, we will explain how to build these smaller search spaces $\mathbb{S}_i$ at each iteration.

A. Ignore already-covered instances.

Consider some iteration $i \in 1...k$ in Algorithm 1 with some already constituted subgroup set $\mathcal{S}_{i-1} \subseteq \mathbb{S}$. We draw the reader attention to the following fact: looking for the top-gain subgroup in pattern structure $\mathbb{P} = (\mathcal{G}, (\mathcal{D}, \sqsubseteq), \delta)$ is equivalent to looking for it in pattern structure $\mathbb{P}_{i-1} = (\mathcal{G}_{\mathcal{S}_{i-1}}, (\mathcal{D}, \sqsubseteq), \delta)$ associated to the set of non-already covered instances $\mathcal{G}_{\mathcal{S}_{i-1}} = \mathcal{G} \setminus \bigcup \mathcal{S}_{i-1}$. Indeed, consider one top-gain subgroup s^* in $\mathbb{P}_{i-1}$ (i.e. $s^* \subseteq \mathcal{G}_{\mathcal{S}_{i-1}}$ and s^* not necessarily belongs to $\mathbb{S}$), subgroup $ext(int(s^*)) \in \mathbb{S}$ maximizes the marginal contribution $\phi_{\mathcal{S}_{i-1}}$. Supposing the converse, that is $\exists t \in \mathbb{S}$ s.t. $\phi_{\mathcal{S}_{i-1}}(t) > \phi_{\mathcal{S}_{i-1}}(ext(int(s^*)))$, allow to build another subgroup $s^{**} = t \cap \mathcal{G}_{\mathcal{S}_{i-1}}$ in pattern structure $\mathbb{P}_{i-1}$ better than s^* which is contradictory under the hypothesis that s^* maximizes $\phi_{\mathcal{S}_{i-1}}$. Using this observation, we build a smaller subgroup search space $\mathbb{S}^i$ at each iteration i: i.e. $\mathbb{S}^i = \left\{ext(int(s \cap \mathcal{G}_{\mathcal{S}_{i-1}})) \mid s \in \mathbb{S}\right\}$. Note that $\mathcal{S}_0 = \emptyset$.

B. Ignore non closed-on-the-positive subgroups.

The closed on the positive (cotp for short) concept was introduced with the concept of relevance in [15], [16]. Informally, cotp are subgroups for which the addition of any constraint to their intent results in dropping at least one positive object (reducing the true positive rate). Formally, a subgroup $s \in \mathbb{S}$ is cotp iff $s = ext(int(s \cap \mathcal{G}^+))$. Mapping $s \mapsto ext(int(s \cap \mathcal{G}^+))$ and $d \mapsto int(ext(d) \cap \mathcal{G}^+)$ are closure operators. Given an arbitrary subgroup $s \in \mathbb{S}$, subgroup $s^+ = ext(int(s \cap \mathcal{G}^+))$ covers the same positive instances as s ($tpr(s^+) = tpr(s)$) but may cover less negative instances ($fpr(s^+) \leq fpr(s)$). Hence, when optimizing a *SD-compatible* measure ϕ, cotp must be preferred: i.e. $(\forall s \in \mathbb{S})$ $\phi(s^+) \geq \phi(s)$. In Figure 1, subgroup $ext(c)$ is not cotp while subgroup $ext(d) = ext(c) \cap \mathcal{G}^+$ is cotp. Hence, using this second observation along with the first obtained beforehand, we get $\mathbb{S}^i = \left\{ext(int(s \cap \mathcal{G}_{\mathcal{S}_{i-1}} \cap \mathcal{G}^+)) \mid s \in \mathbb{S}\right\}$.

C. Ignore unpromising branches.

When exploring the search space of subgroups $\mathbb{S}^i$, one can exploit the order of visit of the subgroups and properties of the quality measure ϕ to ignore unpromising parts of the search space. For instance, if subgroups are explored in a top-down fashion (from larger subgroups to smaller ones), a usual way is to build an *optimistic estimate* following [19], [49].

Definition 4. We say that a measure ϕ has an *optimistic estimate* iff there exists some function $\phi^{oe} : \wp(\mathcal{G}) \to \mathbb{R}$ s.t.

$$\forall s \in \wp(\mathcal{G})\, \forall t \in \wp(\mathcal{G}) \text{ s.t. } t \subseteq s : \phi(t) \leq \phi^{oe}(s)$$

Moreover, ϕ^{oe} is said to be a *tight* iff:

$$\forall s \in \wp(\mathcal{G})\, \exists t \in \wp(\mathcal{G}) \text{ s.t. } t \subseteq s : \phi(t) = \phi^{oe}(s)$$

Hence, if an optimistic estimate $\phi^{oe}_{\mathcal{S}}$ is defined for the marginal contribution $\phi_{\mathcal{S}}$, a simple *branch-and-bound* technique can be used in any *top-down depth-first-search (DFS)* algorithm exploring elements of $\mathbb{S}^i$ to find the top-gain subgroup in $\mathbb{S}^i$. Indeed, consider during exploration of $\mathbb{S}^i$, the current top-gain subgroup is s^*. Then, whenever we visit a subgroup s s.t. $\phi^{oe}_{\mathcal{S}_{i-1}}(s) < \phi_{\mathcal{S}_{i-1}}(s^*)$, we ignore subgroups $t \subseteq s$ since $\phi_{\mathcal{S}_{i-1}}(t) \leq \phi^{oe}_{\mathcal{S}_{i-1}}(s) < \phi_{\mathcal{S}_{i-1}}(s^*)$. Theorem 1 states a *tight optimistic estimate* for the marginal contribution of an arbitrary SD-compatible measure.

Theorem 1. *Let $\mathcal{S}$ be a subgroup set and let ϕ be a SD-compatible measure. The marginal contribution $\phi_{\mathcal{S}}$ has a tight optimistic estimate given by: $\phi^{oe}_{\mathcal{S}} : s \mapsto \phi_{\mathcal{S}}(s \cap \mathcal{G}^+)$.*

Proof. Let $t \subseteq s$, we need to show that the quantity $\phi_{\mathcal{S}}(s \cap \mathcal{G}^+) - \phi_{\mathcal{S}}(t) \geq 0$. We have $\phi_{\mathcal{S}}(s \cap \mathcal{G}^+) - \phi_{\mathcal{S}}(t) = \phi(\bigcup \mathcal{S} \cup (s \cap \mathcal{G}^+)) - \phi(\bigcup \mathcal{S} \cup t)$. Clearly, $t \cap \mathcal{G}^+ \subseteq s \cap \mathcal{G}^+$. It follows that: (i) $tpr(\bigcup \mathcal{S} \cup t) \leq tpr(\bigcup \mathcal{S} \cup (s \cap \mathcal{G}^+))$. Moreover, $fpr(\bigcup \mathcal{S} \cup (s \cap \mathcal{G}^+)) = fpr(\bigcup \mathcal{S})$, since $\mathcal{G}^+ \cap \mathcal{G}^- = \emptyset$. Given that $\bigcup S \subseteq \bigcup S \cup t$, we have: (ii) $fpr(\bigcup \mathcal{S} \cup t) \geq fpr(\bigcup \mathcal{S} \cup (s \cap \mathcal{G}^+))$. From (i) and (ii) and given that ϕ is SD-compatible, we obtain: $\phi_{\mathcal{S}}(s \cap \mathcal{G}^+) - \phi_{\mathcal{S}}(t) \geq 0$. The tightness is obtained directly from $\phi^{oe}_{\mathcal{S}}$ definition since $s \cap \mathcal{G}^+ \subseteq s$. □

N.B. Optimistic estimates of any SD-compatible measure ϕ can be obtained from Theorem 1 when $\mathcal{S} = \emptyset$.

WRAcc measure. We draw a particular attention to WRAcc measure as it is one of the most frequently used measure in SD. The WRAcc marginal contribution to $\mathcal{S}$ can be reformulated as follows (with $\alpha = |\mathcal{G}^+|/|\mathcal{G}|$ and $\mathcal{G}_{\mathcal{S}} = \mathcal{G} \setminus \bigcup \mathcal{S}$):

$$\text{WRAcc}_{\mathcal{S}} : s \mapsto \alpha \cdot (1-\alpha) \cdot \big[tpr\,(\mathcal{G}_{\mathcal{S}}, \mathcal{G}) \cdot tpr\,(s, \mathcal{G}_{\mathcal{S}}) - fpr\,(\mathcal{G}_{\mathcal{S}}, \mathcal{G}) \cdot fpr\,(s, \mathcal{G}_{\mathcal{S}}) \big]$$

Using Theorem 1, its associated optimistic estimate is:

$$\text{WRAcc}^{oe}_{\mathcal{S}} : s \mapsto \alpha \cdot (1-\alpha) \cdot tpr(\mathcal{G}_{\mathcal{S}}, \mathcal{G}) \cdot tpr(s, \mathcal{G}_{\mathcal{S}})$$

D. Algorithm FSSD.

Algorithm 2, dubbed FSSD for Fast and Efficient Algorithm for Subgroup Set Discovery, is basically the combination of the three optimizations presented beforehand. This algorithm provides a greedy solution to Problem III-C given that ϕ is SD-compatible. It follows the same schema of Algorithm 1 where step 3 is revisited as follows: At a given iteration i for the already subgroup set $\mathcal{S}_{i-1}$, it looks for the top-gain subgroup s maximizing $\phi_{\mathcal{S}_{i-1}}$ in the space $\mathbb{S}_i$ presented at the end of Sec. V-B. This is done by leveraging the closure-on-the-positive operator $d \mapsto int(ext(d) \cap \mathcal{G}^+)$ in pattern structure $(\mathcal{G}_{\mathcal{S}_{i-1}}, (\mathcal{D}, \sqsubseteq), \delta)$. The explorations of subgroups in $(\mathcal{G}_{\mathcal{S}}, (\mathcal{D}, \sqsubseteq), \delta)$ is done in a *top-down depth-first-search (DFS)* fashion (i.e. subgroups with larger support are visited first). Details of equivalent exploration can be found in [3], [25]. The employed search strategy for the top-gain subgroup at each iteration i follows a branch-and-bound technique using the optimistic estimate of the marginal contribution $\phi^{oe}_{\mathcal{S}_{i-1}}$ as explained in Sec. V-C.

Algorithm 2: FSSD Algorithm

Input: $(\mathcal{G}, \mathcal{A})$ labeled dataset, $k \in \mathbb{N}$

1 $\mathcal{S}$, $\mathcal{G}_{\mathcal{S}}$, $\leftarrow \emptyset$, $\mathcal{G}$ // $\mathcal{G}_{\mathcal{S}}$ is the set of considered objects
2 **while** $|\mathcal{S}| < k$ **do**
3 Find cotp subgroup s^* providing the maximal gain ϕ_S in pattern structure $(\mathcal{G}_{\mathcal{S}_i}, (\mathcal{D}, \sqsubseteq), \delta)$ using optimistic estimate in a branch-and-bound scheme.
4 **if** $\phi_S(ext(int(s^*))) \leq 0$ **then Break**
5 S, $\mathcal{G}_{\mathcal{S}} \leftarrow S \cup \{ext(int(s^*))\}$, $\mathcal{G}_{\mathcal{S}} \backslash s^*$
6 **return** $\mathcal{S}$

E. Discussion.

FSSD presents many advantages comparing to literature algorithm: it is a *parameter free* algorithm (apart from the cardinality constraint) handling any quality measure that is SD-compatible. Moreover, FSSD is *memory-efficient* since it does not need to store all found patterns before post-processing them as done by DSSD [48] or MCTS4DM [7]. Note also that, in the first iteration (i.e. $\mathcal{S} = \emptyset$), FSSD finds the top quality subgroup existing in the dataset w.r.t. to the pattern language.

VI. Empirical Study

We report our experimental study to evaluate the effectiveness of FSSD implemented in Python 3.7.2. The source code and supplementary experiments are made available in https://github.com/Adnene93/FSSD. We consider a variety of datasets (Tab. II) involving categorical, ordinal and/or continuous numerical attributes from the UCI repository. Experiments are performed using WRAcc.

First, we show and analyze an example of subgroup set returned by FSSD. Second, we study how well FSSD approximates the optimal solution in the benchmark datasets. Third, we compare FSSD to the naive two step greedy solution (dubbed BASELINE) in terms of both memory and time

id	dataset	rows	class	$\frac{\lvert\mathcal{G}^+\rvert}{\lvert\mathcal{G}\rvert}$	#attrs (cat./num.)
D01	*abalone*	4177	M	0.37	(0/8)
D02	*adult*	32561	$\geq 50K$	0.24	(8/6)
D03	*autos*	195	3	0.12	(11/14)
D04	*balance*	625	B	0.08	(0/4)
D05	*breastCancer*	683	4	0.35	(0/9)
D06	*BreastTissue*	106	car	0.20	(0/10)
D07	*CMC*	1473	2	0.23	(0/9)
D08	*credit*	666	+	0.45	(9/6)
D09	*dermatology*	358	3	0.20	(0/34)
D10	*glass*	214	3	0.08	(0/10)
D11	*haberman*	306	2	0.26	(0/3)
D12	*iris*	150	V	0.33	(0/4)
D13	*mushrooms*	8124	p	0.48	(22/0)
D14	*sonar*	208	R	0.47	(0/60)
D15	*TicTacToe*	958	-	0.35	(9/0)

TABLE II: Benchmark datasets and their characteristics: number of rows, the considered class and its prevalence $\frac{|\mathcal{G}^+|}{|\mathcal{G}|}$, number of attributes (categorical / numerical attributes).

efficiency. Fourth, we confront FSSD against three state-of-the-art algorithms: two *Two-steps approach* algorithms DSSD [48], MCTS4M [7], and one *sequential covering approach* algorithm CN2−SD [31].

Table III reports a subgroup set of size $k = 5$ returned by FSSD when executed on *Haberman* (*D11*). This results set covers $210/306 \approx 68.6\%$ records of the dataset, with a $tpr \approx 92.6\%$ and a $fpr \approx 0.6$. The WRAcc of the set equals 0.063. Notice that the identified subgroups are completely disjoint. This example demonstrates the ability of FSSD to uncover a subgroup set that is both discriminant and diversified (see Fig. 2 **(left)**). It is worth mentioning that FSSD can return overlapping subgroups as depicted in Fig. 2 **(right)**.

In order to evaluate the quality of FSSD' results, we compute the approximation ratio ρ of FSSD for the benchmark datasets. This requires to retrieve the ground truth (the optimal solution) for each of the 15 datasets. In order to make the calculation of the ground truth computationally possible, we have reduced the number of attributes. We specify the number of considered attributes for each dataset in its affiliated id. For example, D07-4 refers to the 7^{th} dataset and the number of attributes is 4. Attributes are selected in the following order: first, categorical, and then, numerical. Categorical (resp. numerical) attributes are sorted in an ascending order according to the size of their domain. Table IV reports the results. For most datasets, the approximation ratio is very high. FSSD succeeds to approximate the ground truth by at least 87% for all the datasets except for D10-3, where the ratio is acceptable (62%).

Table V presents the time and memory usage of FSSD and the BASELINE on all the datasets. The number of attributes is limited so that the BASELINE succeeds to mine the subgroup set in the limit of 30 minutes and without exceeding the machine memory limit. FSSD is faster than the BASELINE in all the configuration. Furthermore, FSSD consumes significantly less memory than the BASELINE (10 times less in average). The memory usage of the BASELINE is considerably impacted by the number of local subgroups, since it needs to store their complete list in order to process them.

We compare FSSD ($\text{Depth}_{\max}$=8) against the competitive approaches: DSSD ($\text{Depth}_{\max}$=8, BeamWidth=5, j=10000 and coverBeamMultiplier=0.9), MCTS4DM (nb_{iter}=50000 and maxRedundancy=0.25), and CN2−SD ($\text{Depth}_{\max}$=8, BeamWidth=50 and $\text{min\_sup\_pos} \geq |\mathcal{G}^+|/10$). We report in Table VI the runtime and quality of results for each approach on the benchmark datasets. For each dataset, we limit the number of attributes to the maximum number so that the four algorithms succeed to finish within one hour. FSSD provides the best qualities for all the datasets except D03, where DSSD performs better. Particularly, the difference is notable when comparing with CN2−SD and MCTS4DM. Regarding the runtime, FSSD is generally faster than DSSD, MCTS4DM, and CN2−SD in most configurations except for D01 and D14 where FSSD managed to find a better subgroup set in terms of the quality.

Subgroup Intent	support	W_{Gain}
"$age \in [43, 83] \wedge operation\_year \in [58, 65]$"	179	0.041
"$age \in [34, 46] \wedge operation\_year \in [66, 69]$"	20	0.009
"$age \in [54, 61] \wedge operation\_year \in [68, 68]$"	4	0.006
"$age \in [41, 41] \wedge operation\_year \in [60, 64]$"	3	0.004
"$age \in [52, 52] \wedge operation\_year \in [66, 69]$"	4	0.003

TABLE III: A subgroup set of size 5, $WRAcc \approx 0.063$ and $support = 210$ identified by FSSD in the dataset $D11$ projected on two out of three attributes.

id	ρ	id	ρ	id	ρ	id	ρ
D01-1	99.7%	D05-1	100%	D09-5	99.7%	D13-7	99.6%
D02-3	100 %	D06-1	90.54%	D10-3	62.2%	D14-1	100 %
D03-7	87.3%	D07-4	99.66%	D11-1	100%	D15-4	100 %
D04-2	100 %	D08-7	99.62%	D12-1	100 %		

TABLE IV: Approximation ratios ρ of the Top3 obtained after running FSSD on each of the 15 reduced datasets.

id	BASELINE		FSSD	
	t(s)	M(MiB)	t(s)	Mem.(MiB)
D01-5	0.37	22.73	**0.21**	**15.34**
D02-10	120.31	426.16	**8.70**	**40.09**
D03-10	34.37	37.10	**0.09**	**9.13**
D04-4	0.77	25.60	**0.29**	**1.55**
D05-9	2.63	103.52	**0.06**	**2.58**
D06-10	38.01	23.98	**0.04**	**5.51**
D07-9	97.54	2556.15	**45.62**	**7.42**
D08-10	3.76	77.22	**0.26**	**3.33**
D09-10	34.46	338.61	**0.06**	**2.32**
D10-10	11.89	26.11	**0.07**	**9.46**
D11-3	33.78	1141.01	**7.16**	**2.58**
D12-4	15.78	341.79	**0.02**	**1.29**
D13-10	149.25	457.50	**3.96**	**33.52**
D14-10	36.97	1533.13	**17.77**	**4.08**
D15-9	2.68	20.77	**0.27**	**2.06**

TABLE V: Comparison of the runtime -t(s)- and memory consumption -Mem.(MiB)- of FSSD vs. BASELINE for a Top5 subgroup set discovery task on the benchamrk datasets.

id	DSSD		MCTS4DM		CN2SD		FSSD	
	t(s)	Qual	t(s)	Qual	t(s)	Qual	t(s)	Qual
D01-5	**4.93**	0.06	30.28	0.05	11.75	0.04	72.09	**0.07**
D02-10	504	0.10	111.53	-	130	0.07	**237**	**0.11**
D03-10	1.77	**0.08**	788.79	-	3.54	-	**0.01**	0.07
D04-4	1.36	0.02	3.91	0.002	7.62	-	**0.28**	**0.03**
D05-9	2.30	0.17	2.28	0.05	3.58	-	**1.18**	**0.22**
D06-10	1.83	0.14	4.40	-	1.64	-	**0.01**	**0.16**
D07-9	2.95	0.06	**2.87**	0.06	20.19	0.03	133	**0.08**
D08-10	2.34	0.18	1864	-	7.24	-	**0.23**	**0.19**
D09-10	1.40	0.09	1.76	0.004	2.58	-	**0.03**	**0.16**
D10-10	2.08	0.06	2.66	0.02	2.92	-	**0.01**	**0.07**
D11-3	1.38	0.07	5.16	0.02	4.61	0.08	**0.35**	**0.09**
D12-4	1.34	0.20	2.59	0.20	1.35	0.20	**0.01**	**0.22**
D13-10	6.56	0.19	565.81	-	2.94	-	**0.54**	**0.23**
D14-10	**2.32**	0.11	9.09	0.08	10.71	-	933	**0.16**
D15-9	1.91	0.07	178.04	-	3.24	0.13	**0.24**	**0.17**

TABLE VI: Comparison of the runtime and qualities of the Top5 identified subgroups set by FSSD, DSSD, MCTS4DM and CN2SD. '-' corresponds to the cases when an algorithm fails to find a subgroup set with a strictly positive quality.

VII. Conclusion

We introduced in this paper a novel problem of diverse subgroup set discovery where both discriminative power and

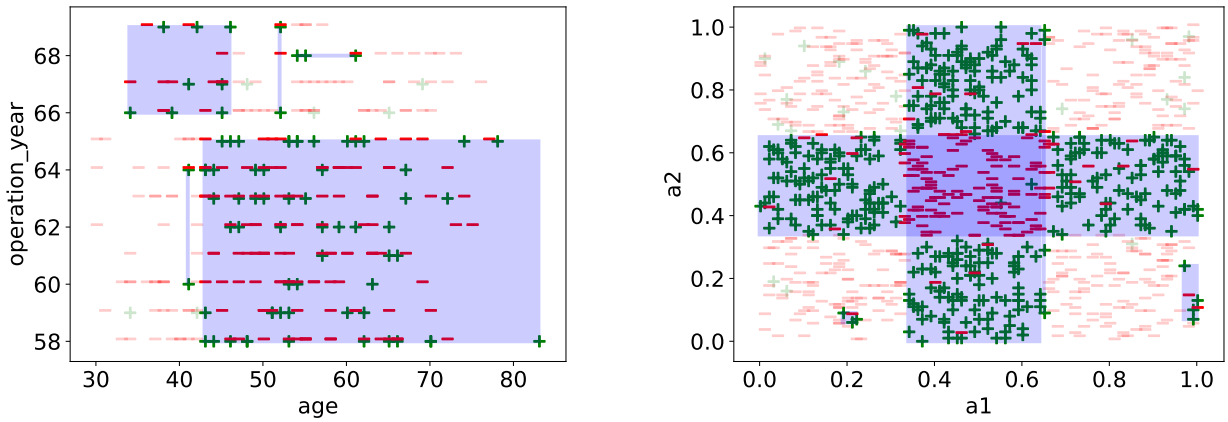

Fig. 2: **(left)** subgroup set found for dataset $D11$ illustrating Table III, **(right)** subgroup set found for a synthetic dataset with two numerical attributes. One can see that almost all positive instances are covered by the subgroup set.

diversity of the subgroup set are incorporated into the same measure (i.e. quality of subgroups union). We proposed then `FSSD` , a *parameter-free* and *memory efficient* greedy algorithm approximating the solution of the proposed problem. `FSSD` exploits closure operators and tight optimistic estimates on the marginal contributions of SD-compatible measures to efficiently explore and prune unpromising areas of the search space. We shown that unfortunately, for many SD-compatible measures like WRAcc, greedy algorithms cannot provide guarantees (lower bound) on the approximation ratio to the best possible solution. Nonetheless, empirical study gave evidence that `FSSD` (1) is a time and memory efficient algorithm, (2) is able to provide a diverse and high quality subgroup set and (3) provides a judicious trade-off between the runtime and the quality compared to the state-of-the-art approaches. A further improvement of `FSSD` performance may be achieved if only class-relevant patterns are sought. This requires a polynomial space and output-polynomial time algorithm for enumerating such patterns, which remains an open problem [18]. Other open questions that need to be further investigated are the following: is there an algorithm that ensure approximation guarantees for Problem III-C in a tractable way, i.e. is Problem III-C hard to approximate?

Aknowledgement. This work has been partially supported by the project *ContentCheck* **ANR-15-CE23-0025** funded by the French National Research Agency, the Association Nationale Recherche Technologie (**ANRt**) French program. The authors would like to thank the reviewers for their valuable remarks.

References

[1] Rakesh Agrawal, Tomasz Imielinski, and Arun N. Swami. Mining association rules between sets of items in large databases. In *ACM SIGMOD*, pages 207–216. ACM Press, 1993.

[2] Martin Atzmüller and Frank Puppe. Sd-map - A fast algorithm for exhaustive subgroup discovery. In *PKDD 2006*, pages 6–17, 2006.

[3] Adnene Belfodil, Sylvie Cazalens, Philippe Lamarre, and Marc Plantevit. Flash points: Discovering exceptional pairwise behaviors in vote or rating data. In *ECML/PKDD*, 2017.

[4] Aimene Belfodil, Adnene Belfodil, and Mehdi Kaytoue. Anytime subgroup discovery in numerical domains with guarantees. In *ECML/PKDD (2)*, volume 11052 of *Lecture Notes in Computer Science*, pages 500–516. Springer, 2018.

[5] Tijl De Bie, Kleanthis-Nikolaos Kontonasios, and Eirini Spyropoulou. A framework for mining interesting pattern sets. *SIGKDD Explorations*, 12(2):92–100, 2010.

[6] Mario Boley, Claudio Lucchese, Daniel Paurat, and Thomas Gärtner. Direct local pattern sampling by efficient two-step random procedures. In *KDD*, pages 582–590. ACM, 2011.

[7] Guillaume Bosc, Jean-François Boulicaut, Chedy Raïssi, and Mehdi Kaytoue. Anytime discovery of a diverse set of patterns with monte carlo tree search. *Data Min. Knowl. Discov.*, 32(3):604–650, 2018.

[8] Björn Bringmann and Albrecht Zimmermann. The chosen few: On identifying valuable patterns. In *ICDM*, pages 63–72, 2007.

[9] Peter Clark and Tim Niblett. The CN2 induction algorithm. *Machine Learning*, 3:261–283, 1989.

[10] Guozhu Dong and Jinyan Li. Efficient mining of emerging patterns: Discovering trends and differences. In *KDD*, pages 43–52. ACM, 1999.

[11] Usama M. Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. From data mining to knowledge discovery in databases. *AI Magazine*, 17(3):37–54, 1996.

[12] Johannes Fürnkranz and Peter A. Flach. ROC 'n' rule learning - towards a better understanding of covering algorithms. *Machine Learning*, 58(1):39–77, 2005.

[13] B. Ganter and R. Wille. *Formal Concept Analysis*. Springer, 1999.

[14] Bernhard Ganter and Sergei O. Kuznetsov. Pattern structures and their projections. In *ICCS*, pages 129–142, 2001.

[15] Gemma C. Garriga, Petra Kralj, and Nada Lavrac. Closed sets for labeled data. In *PKDD*, pages 163–174, 2006.

[16] Gemma C. Garriga, Petra Kralj, and Nada Lavrac. Closed sets for labeled data. *Journal of Machine Learning Research*, 9:559–580, 2008.

[17] Liqiang Geng and Howard J. Hamilton. Interestingness measures for data mining: A survey. *ACM Comput. Surv.*, 38(3):9, 2006.

[18] Henrik Grosskreutz. Class relevant pattern mining in output-polynomial time. In *SDM*, pages 284–294. SIAM / Omnipress, 2012.

[19] Henrik Grosskreutz, Stefan Rüping, and Stefan Wrobel. Tight optimistic estimates for fast subgroup discovery. In *ECML/PKDD 2008*, pages 440–456, 2008.

[20] Tias Guns, Siegfried Nijssen, and Luc De Raedt. Evaluating pattern set mining strategies in a constraint programming framework. In *PAKDD*, pages 382–394, 2011.

[21] Tias Guns, Siegfried Nijssen, and Luc De Raedt. k-pattern set mining under constraints. *IEEE TKDE.*, pages 402–418, 2013.

[22] Jon Hills, Luke M. Davis, and Anthony Bagnall. Interestingness measures for fixed consequent rules. In *IDEAL*, pages 68–75. Springer, 2012.

[23] Frederik Janssen and Johannes Fürnkranz. On trading off consistency and coverage in inductive rule learning. In *LWA*, volume 1/2006

of *Hildesheimer Informatik-Berichte*, pages 306–313. University of Hildesheim, Institute of Computer Science, 2006.
[24] Roberto J. Bayardo Jr. and Rakesh Agrawal. Mining the most interesting rules. In *KDD*, pages 145–154. ACM, 1999.
[25] Mehdi Kaytoue, Sergei O. Kuznetsov, and Amedeo Napoli. Revisiting Numerical Pattern Mining with Formal Concept Analysis. In *IJCAI*, pages 1342–1347, 2011.
[26] Mehdi Khiari, Patrice Boizumault, and Bruno Crémilleux. Constraint programming for mining n-ary patterns. In *CP 2010*, pages 552–567, 2010.
[27] Willi Klösgen. Explora: A multipattern and multistrategy discovery assistant. In *Advances in Knowledge Discovery and Data Mining*, pages 249–271. 1996.
[28] Arno J. Knobbe and Eric K. Y. Ho. Pattern teams. In *PKDD*, pages 577–584, 2006.
[29] Sergei O. Kuznetsov. Learning of simple conceptual graphs from positive and negative examples. In *PKDD*, pages 384–391, 1999.
[30] Nada Lavrac, Peter A. Flach, and Blaz Zupan. Rule evaluation measures: A unifying view. In *ILP*, volume 1634 of *Lecture Notes in Computer Science*, pages 174–185. Springer, 1999.
[31] Nada Lavrac, Branko Kavsek, Peter A. Flach, and Ljupco Todorovski. Subgroup discovery with CN2-SD. *JMLR*, pages 153–188, 2004.
[32] Florian Lemmerich, Mathias Rohlfs, and Martin Atzmüller. Fast discovery of relevant subgroup patterns. In *FLAIRS*, 2010.
[33] Philippe Lenca, Patrick Meyer, Benoît Vaillant, and Stéphane Lallich. On selecting interestingness measures for association rules: User oriented description and multiple criteria decision aid. *European Journal of Operational Research*, 184(2):610–626, 2008.
[34] Michael Mampaey, Siegfried Nijssen, Ad Feelders, and Arno J. Knobbe. Efficient algorithms for finding richer subgroup descriptions in numeric and nominal data. In *ICDM*, pages 499–508, 2012.
[35] Taneli Mielikäinen and Heikki Mannila. The pattern ordering problem. In *PKDD 2003*, pages 327–338, 2003.
[36] George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions - I. *Math. Program.*, 14(1):265–294, 1978.
[37] Miho Ohsaki, Shinya Kitaguchi, Kazuya Okamoto, Hideto Yokoi, and Takahira Yamaguchi. Evaluation of rule interestingness measures with a clinical dataset on hepatitis. In *PKDD*, volume 3202 of *Lecture Notes in Computer Science*, pages 362–373. Springer, 2004.
[38] Abdelkader Ouali, Albrecht Zimmermann, Samir Loudni, Yahia Lebbah, Bruno Crémilleux, Patrice Boizumault, and Lakhdar Loukil. Integer linear programming for pattern set mining; with an application to tiling. In *PAKDD 2017*, pages 286–299, 2017.
[39] Gregory Piatetsky-Shapiro. Discovery, analysis, and presentation of strong rules. In *Knowledge Discovery in Databases*, pages 229–248. AAAI/MIT Press, 1991.
[40] Luc De Raedt and Albrecht Zimmermann. Constraint-based pattern set mining. In *SDM*, pages 237–248. SIAM, 2007.
[41] Matteo Riondato and Fabio Vandin. Misosoup: Mining interesting subgroups with sampling and pseudodimension. In *KDD*, pages 2130–2139, 2018.
[42] S. Roman. *Lattices and Ordered Sets*. Springer New York, 2008.
[43] Arno Siebes. Data science as a language: challenges for computer science - a position paper. *I. J. Data Science and Analytics*, 6(3):177–187, 2018.
[44] Arno Siebes, Jilles Vreeken, and Matthijs van Leeuwen. Item sets that compress. In *SDM*, pages 395–406. SIAM, 2006.
[45] Pang-Ning Tan, Vipin Kumar, and Jaideep Srivastava. Selecting the right interestingness measure for association patterns. In *SIGKDD 2002*, pages 32–41. ACM, 2002.
[46] Ljupčo Todorovski, Peter Flach, and Nada Lavrač. Predictive performance of weighted relative accuracy. In *PKDD*, pages 255–264, 2000.
[47] John W. Tukey. *Exploratory data analysis*. Addison-Wesley series in behavioral science : quantitative methods. Addison-Wesley, 1977.
[48] Matthijs van Leeuwen and Arno J. Knobbe. Diverse subgroup set discovery. *Data Min. Knowl. Discov.*, 25(2):208–242, 2012.
[49] Geoffrey I. Webb. OPUS: an efficient admissible algorithm for unordered search. *J. Artif. Intell. Res.*, 3:431–465, 1995.
[50] Stefan Wrobel. An algorithm for multi-relational discovery of subgroups. In *PKDD*, pages 78–87, 1997.
[51] Tao Zhang. Association rules. In *PAKDD*, volume 1805 of *Lecture Notes in Computer Science*, pages 245–256. Springer, 2000.

APPENDIX

A. Problem III-C is NP-Hard.

Before giving the demonstration, let us consider a reformulation of Problem III-C presented below in Problem A.

Problem Statement (Problem A). Let be some set of objects $\mathcal{G}$ partitioned into $\mathcal{G}^+$ and $\mathcal{G}^-$. Let $\mathbb{S} \subseteq \wp(\mathcal{G})$ be all possible subgroups derived w.r.t. some pattern language and let $k \in \mathbb{N}^*$. Output one subgroup set from $\operatorname{argmax}_{\mathcal{S}\subseteq\mathbb{S},|\mathcal{S}|\leq k} \mathrm{WRAcc}(\mathcal{S})$

Problem A can be seen as a simplification of Problem III-C in the sense that (1) it does consider only the special problem of optimizing the WRAcc measure and (2) consider the set of all possible subgroups $\mathbb{S}$ as an input rather than as an intermediary output. Showing that Problem A is NP-hard allows us hence to show that Problem III-C is also NP-hard. Before showing the NP-hardness of Problem A, we recall below the Maximum Cover Problem (MCP) which is an NP-complete problem. The NP-hardness of MCP is a consequence of the NP-hardness of optimizing a submodular set function subject to a cardinality constraint [36].

Problem Statement (MCP). Let $\mathcal{G}$ be a finite universe, $\mathbb{S} \subseteq \wp(\mathcal{G})$ and $k \in \mathbb{N}^*$. Finds out a subset $\mathcal{S} \subseteq \mathbb{S}$ such that $|\mathcal{S}| \leq k$ for which $|\bigcup \mathcal{S}|$ is maximized. Formally, output one solution from $\operatorname{argmax}_{\mathcal{S}\subseteq\mathbb{S},|\mathcal{S}|\leq k} |\bigcup \mathcal{S}|$.

Proposition 1. Problem A is NP-hard.

Proof. Consider the following MCP problem with $\mathcal{G}^+$ is a finite set of elements, $\mathbb{S}^+ \subseteq \wp(\mathcal{G}^+)$ and $k \in \mathbb{N}^*$. We *reduce* (in a polynomial-time) this MCP problem to an instance of Problem A as follow: Build a set $\mathcal{G}$ s.t. $\mathcal{G} = \mathcal{G}^+ \cup \mathcal{G}^-$, $\mathcal{G}^+ \cap \mathcal{G}^- = \emptyset$ and $|\mathcal{G}^-| = |\mathcal{G}^+|$. Let $\mathbb{S} = \mathbb{S}^+$. It is clear that for all $\mathcal{S} \subseteq \mathbb{S}^+$, we have:

$$\mathrm{WRAcc}(\mathcal{S}) = \frac{1}{2 \cdot |\mathcal{G}|} \cdot \left|\bigcup \mathcal{S}\right|$$

where $\frac{1}{2\cdot|\mathcal{G}|}$ is a constant. Clearly, any solution of this instance of Problem A is also a solution of the MCP problem instance. Hence, since the MCP Problem is NP-hard then Problem A is NP-hard. □

Corollary 1. Problem III-C is NP-hard.

B. Greedy algorithm provides no guarantee.

We have investigated in Section IV of the paper that the greedy algorithm provides no guarantee on the approximation ratio. However, in this proof, we considered the language of itemsets. We show here that, still, even for datasets with exclusively categorical attributes (or numerical ones), there is no provided guarantee on the approximation ratio. Proposition 2 formalizes this observation.

Proposition 2. $\forall \epsilon > 0$, one can build an instance of Problem III-C such that all the attributes are categorical and solution $\mathcal{S}_{greedy}$ outputted from Algorithm 1 (i.e. greedy scheme) provide an approximation ratio $\rho(\mathcal{S}_{greedy}) < \epsilon$.

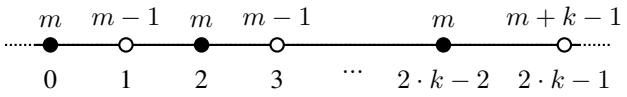

Fig. 3: Inexistence of approximation guarantee by the greedy algorithm for numerical datasets (Proposition 2 proof)

Proof. Let us build a parametric instance of Problem III-C where k is the cardinality constraint and the optimized measure is the WRAcc measure. Let be $m \in \mathbb{N}^*$, we build the dataset $(\mathcal{G}, \mathcal{A})$ as depicted in Fig. 3. That is: $\mathcal{G} = \{g_i\}_{0 \leq i < 2 \cdot k \cdot m}$ and $\mathcal{A} = \{a\}$ with a is numerical. The black (resp. white) dots refer to elements in $\mathcal{G}^+$ (resp. in $\mathcal{G}^-$). We write below each dot the value of the attribute a and above the dot the number of elements having the same value. For instance, in this dataset, there is m positive instances having for value 0 on attribute a followed by $m-1$ negative instances having for value 1, ... followed by m positive instances having for value $2 \cdot k - 2$ followed by $m + k - 1$ negative instances having for value $2 \cdot k - 1$. It is clear that there is $k \cdot m$ positive instances and $k \cdot m$ negative instances. Since the used patterns are interval patterns, with k is the cardinality constraints one can build the k-sized optimal subgroup set $\mathcal{S}^*$ for which $\text{WRAcc}(\mathcal{S}^*) = 0.25$ is optimal (i.e. regroups subgroups associated to constraints $0 \leq a \leq 0$, $2 \leq a \leq 2$, ... , $2 \cdot k - 2 \leq a \leq 2 \cdot k - 2$). Subgroup set $\mathcal{S}^*$ is optimal since its true positive rate is 1 and false positive rate is 0. That is: it takes all positive instances without covering any negative one. However, one can easily show that the greedy algorithm will output 1-sized subgroup set $\mathcal{S}_{greedy}$ containing the extent of pattern $0 \leq a \leq 2 \cdot k - 2$ since that this subgroup is the optimal one w.r.t. WRAcc. Hence: $\text{WRAcc}(\mathcal{S}_{greedy}) = 0.25 \cdot \left(1 - \frac{(k-1) \cdot (m-1)}{k \cdot m}\right)$. Thus, $\rho(\mathcal{S}_{greedy}) = \frac{k+m-1}{k \cdot m}$. It is easy to see now that for all $\epsilon > 0$ one can always compute k and m such that $\rho(\mathcal{S}_{greedy}) < \epsilon$.

Please note that this proof consider also categorical attributes. Indeed, one can transform the numerical dataset $(\mathcal{G}, \{a\})$ used here to the categorical dataset $(\mathcal{G}, \mathcal{A}_C)$ where $\mathcal{A}_C = \{a_i \mid i \in 0..2 \cdot k - 1\}$ s.t. the range of each a_i is given by $\mathcal{R}_{a_i} = \{\text{“} \leq i\text{”}, \text{“} > i\text{”}\}$. If an object $g \in \mathcal{G}$ has $a(g) = i$ in the first dataset then it has $a_j(g) = \text{“} \leq j\text{”}$ for all $j \geq i$ and $a_j(g) = \text{“} > j\text{”}$ for $j < i$. It is clear that the subgroups induced by $(\mathcal{G}, \mathcal{A}_C)$ are exactly the same as those induced the numerical dataset $(\mathcal{G}, \{a\})$.

One can follow an equivalent reasoning for other SD-compatible measures such as the measures highlighted in **bold** in Table VII. □

C. Quality measures in SD

Following the notations presented in Sec. II, Table VII presents different quality measures that are used in subgroup discovery. Except the support and the false positive rate, all these quality measures are *SD-Compatible* following Definition 1. Please note that measures are regrouped in blocs. Each bloc refer to *compatible measures*, i.e. measures ordering subgroups in the same way (see Definition 2.2 in [12]).

In order to compute an optimistic estimate (see Definition 4) for these measures, one can follow Theorem 1 where the false positive rate of the subgroup (x) is replaced by 0. The measures highlighted in **bold** refer to the measure having a non constant optimistic estimate.

Measure	**Definition**
False Positive Rate	$x := fpr(t) = \lvert t \cap \mathcal{G}^- \rvert / \lvert \mathcal{G}^- \rvert$
True Positive Rate	$y := tpr(t) = \lvert t \cap \mathcal{G}^+ \rvert / \lvert \mathcal{G}^+ \rvert$
Positive Prevalence	$\alpha := \lvert \mathcal{G}^+ \rvert / \lvert \mathcal{G} \rvert$
Dataset size	$n := \lvert \mathcal{G} \rvert$
Relative Support [1]	$s := \alpha \cdot y + (1 - \alpha) \cdot x$
Precision/Confidence [1]	$p := \alpha \cdot y / s$
Growth Rate [10]	$[(1-\alpha)/\alpha] \cdot [p/(1-p)] = y/x$
Ganascia index [22]	$2p - 1$
Sebag-Schoenaur [22]	$p/(1-p)$
ECE rate [22]	$2 - 1/p$
Ohsakis Conviction [37]	$(1-\alpha)^2/(1-p)$
Lift [22]	p/α
Mutual information [22]	$log(p/\alpha)$
One way support [17]	$p \cdot log(p/\alpha)$
Added Value [30]	$p - \alpha$
Certainty Factor [22]	$(p-\alpha)/(1-\alpha)$
Brin's Conviction [22]	$(1-\alpha)/(1-p)$
Zhang [51]	$(y-x)/sup\{y, x\}$
Odds Ratio [45]	$\Omega := [y \cdot (1-x)]/[x(1-y)]$
Yule's Q [45]	$(\Omega - 1) \,/\, (\Omega + 1)$
Yule's x [45]	$\left(\sqrt{\Omega} - 1\right) / \left(\sqrt{\Omega} + 1\right)$
Least contradiction [17]	$[\alpha \cdot y - (1-\alpha) \cdot x]/\alpha$
Accuracy [17]	$\alpha \cdot y - (1-\alpha) \cdot x + (1-\alpha)$
WRAcc [30]	$\alpha \cdot (1-\alpha) \cdot (y-x)$
Informedness [4]	$y - x$
Binomial Test [34]	$\alpha \cdot (1-\alpha) \cdot (y-x)/\sqrt{s}$
Klösgen$_\omega$ ($\omega \in [0,1]$) [23]	$\alpha \cdot (1-\alpha) \cdot s^{\omega-1} \cdot (y-x)$
Linear correlation [45]	$\sqrt{[(\alpha \cdot (1-\alpha)) \,/\, (s \cdot (1-s))]} \cdot (y-x)$
Cohen's kappa (κ) [45]	$\kappa := 2\alpha \cdot (1-\alpha)(y-x)/[\alpha + (1-2\alpha) \cdot s]$
Cosine/G-Measure [17]	$y/\sqrt{[y + (\{1/\alpha\} - 1) \cdot x]}$
m**-estimate** ($m > 0$) [12]	$\alpha \cdot [y + m/n]/[\alpha \cdot y + (1-\alpha) \cdot x + m/n]$
Discriminativity [6]	$n^2 \cdot \alpha \cdot (1-\alpha) \cdot y \cdot (1-x)$
$\mathbf{F}_\beta$ ($\beta \in [0, +\infty)$) [23]	$[(1+\beta^2) \cdot y]/[y + (\{1/\alpha\} - 1) \cdot x + \beta^2]$

TABLE VII: Quality of a subgroup t using its *false positive rate* x, its *true positive rate* y, positive prevalence α and the dataset size n.