Efficient Discovery of Significant Patterns with Few-Shot Resampling

Leonardo Pellegrina
Dept. of Information Engineering, University of Padova
Padova, Italy
leonardo.pellegrina@unipd.it

Fabio Vandin
Dept. of Information Engineering, University of Padova
Padova, Italy
fabio.vandin@unipd.it

ABSTRACT

Significant pattern mining is a fundamental task in mining transactional data, requiring to identify *patterns* significantly associated with the value of a given feature, the *target*. In several applications, such as biomedicine, basket market analysis, and social networks, the goal is to discover patterns whose association with the target is defined with respect to an underlying population, or process, of which the dataset represents only a collection of observations, or samples. A natural way to capture the association of a pattern with the target is to consider its *statistical significance*, assessing its deviation from the (null) hypothesis of independence between the pattern and the target. While several algorithms have been proposed to find statistically significant patterns, it remains a computationally demanding task, and for complex patterns such as subgroups, no efficient solution exists.

We present FSR, an efficient algorithm to identify statistically significant patterns with rigorous guarantees on the probability of false discoveries. FSR builds on a novel general framework for mining significant patterns that captures some of the most commonly considered patterns, including itemsets, sequential patterns, and subgroups. FSR uses a small number of resampled datasets, obtained by assigning i.i.d. labels to each transaction, to rigorously bound the supremum deviation of a quality statistic measuring the significance of patterns. FSR builds on novel tight bounds on the supremum deviation that require to mine a small number of resampled datasets, while providing a high effectiveness in discovering significant patterns. As a test case, we consider significant subgroup mining, and our evaluation on several real datasets shows that FSR is effective in discovering significant subgroups, while requiring a small number of resampled datasets.

1 INTRODUCTION

Pattern mining is a fundamental task in data mining that, in its most common definition [17], requires to find patterns that occur more often than a given frequency threshold in a database of transactions. Pattern mining finds applications in several areas such as market basket analysis [2], graph databases [1, 3, 11], and the analysis of spatial and temporal data [9, 10, 18].

Significant pattern mining [16, 30] is an extension of pattern mining that, in its most general formulation, requires to discover *patterns* with a *significant association* with a binary label from a dataset consisting of a collection of elements, where each element comprises the values of *features*, which may be categorical, binary, or continuous, and the value of the binary label of interests, also called the *target*. Such formulation captures various types of patterns, such as *itemsets*, when all features are binary, or *subgroups* [4], with more general features. This task finds applications in a wide range of domains, such as market basket analysis, medicine, and molecular biology, where finding reliable associations is paramount.

Significance is usually assessed using the *statistical hypothesis testing* framework. In such framework one defines a measure of *quality* for patterns, and assumes the *null hypothesis* of no association between a pattern and the target label. The *significant patterns* are then the ones with quality that significantly deviate from the *null distribution*, that is, the distribution of the quality under the null hypothesis. The deviation from the null distribution is usually measured by a *p-value*, that is the probability, under the null distribution, that the pattern has quality as large as the one observed in the dataset.

A major complication in the use of the statistical hypothesis testing framework in data mining is given by the huge number of candidate patterns that are considered, resulting in a *multiple hypothesis testing* problem. With a huge number of candidate patterns, some non-significant patterns display a substantial deviation from the null distribution just by chance. Therefore, it is critical to account for testing multiple hypotheses when mining significant patterns, in order to avoid reporting a large number of spurious discoveries. Several methods have been proposed to deal with multiple hypothesis testing [6, 7, 46]. While these methods provide various guarantees, the one most commonly considered is the Family-Wise Error Rate (FWER), which is the probability of reporting in output one or more false discoveries.

Current approaches for significant pattern mining with guarantees on the FWER belong to one of two classes. The first class is given by approaches that assess the significance of each single pattern (e.g., through a *p*-value or related quantities), and then perform an analytical correction to account for multiple hypothesis testing [43–45]. A widely used procedure in this class is given

by Bonferroni correction [6], which computes a *corrected p-value* by multiplying the *p*-value $p_{\mathcal{P}}$ of a pattern $\mathcal{P}$ by the number h of candidate hypotheses. If patterns with corrected *p*-value $h \times p_{\mathcal{P}}$ below a threshold α are flagged as significant, the FWER of the output is guaranteed to be $\leq \alpha$. While these approaches and their improved versions [28, 40] are fairly efficient, thanks to the use of analytical derivations, they suffer from a low *statistical power*, that is they often fail in identifying significant patterns, due to the multiple hypothesis corrections that must provide guarantees for every situation, independently of the observed data.

The second class is given by approaches that use the dataset to estimate the overall distribution of the patterns' statistics (or corresponding measures, such as the *p*-values) under the null hypothesis [25, 32, 39]. The distribution is estimated using permuted versions of the data, obtained by keeping the features in each element of the dataset fixed and randomly permuting the target labels among elements. For example, the Westfall-Young (WY) method [46] uses permuted datasets to estimate the quantiles of the *smallest p*-value under the null hypotheses (or, equivalently, largest qualities or test statistics), and uses such estimate to derive a corrected threshold to flag patterns as significant. These approaches usually improve the statistical power for detecting significant patterns compared to approaches in the first class, but are often computationally demanding, since they need to mine a large number of permuted datasets to obtain good estimates of the overall distribution of the patterns' statistics. While this can be achieved fairly efficiently for simple patterns such as itemsets [25, 32], the overall approach is impractical for more complex patterns such as subgroups, for which mining even a single dataset is extremely time-consuming.

An additional limitation of permutational approaches is that they focus on *conditional testing* [15]. In conditional testing one assumes that the variables of interest, in our case the frequency of patterns and the fraction of elements with target label 1, are the same in *every* dataset from the null distribution. In contrast, in *unconditional testing* [5] one assumes that the variables of interest are the realization of corresponding random variables. Conditional testing and unconditional testing capture different assumptions regarding how data is generated and collected, that is, whether the variables of interest would be the same in different repetitions of the experiment. The choice between the two types of testing depends on the specific scenarios. However, in practice conditional tests are often used for computational reasons, since unconditional tests are much more demanding from the computational standpoint due to the need to account for uncertainties in the observed quantities. In fact, while for simple patterns such as itemsets [31] significant pattern mining procedures with (partial) unconditional testing have been designed, for more complex patterns such as subgroups no unconditional testing procedure is available.

1.1 Contributions

This work focuses on the efficient discovery of significant patterns. Our contributions are four-fold. Firstly, we propose the first general framework to discover significant patterns that can be used for both conditional and unconditional testing. Our framework is based on a natural definition of a pattern's quality that captures its significance, and applies to any type of pattern for which the *appearance*

of the pattern in an element of the dataset is well defined. Such patterns include widely used patterns such as itemsets, subgroups, sequential patterns, and subgraphs. Second, we propose FSR, an algorithm for the efficient discovery of a rigorous approximation of significant patterns while controlling the *Family-Wise Error Rate*, which is the probability of reporting even a single false discovery. FSR uses a *few-shot resampling approach*, that is, it mines a small number of *resampled* datasets, obtained by keeping the features of each element of the dataset fixed and assigning i.i.d. values to the target. Moreover, FSR can leverage any existing algorithm for mining the patterns of interest. Third, we provide novel tight theoretical results relating the distribution of patterns' qualities under the conditional and the unconditional distributions, and relating the estimated maximum deviation of patterns' qualities in resampled datasets with their (unknown) true quality in the corresponding distribution. These results are crucial in making our approach practical, since they imply that mining a small number of resampled datasets is enough to identify significant patterns. Fourth, we consider significant subgroups mining as a test case in our extensive empirical evaluation. We use our algorithm FSR to derive the first approach for mining significant subgroups with unconditional testing, and an approach for the conditional testing scenario which is much more efficient than permutation testing while maintaining an extremely high power. For the most challenging datasets, FSR is the only approach that allows to mine significant subgroups within reasonable time. More importantly, we remark that the considered test case of subgroups is well representative of other settings. In fact, we expect the sensible improvements obtained by FSR to transfer to other cases (i.e., other pattern types), given the generality of our framework and the characteristics of our permutational approaches, which are shared by all significant pattern tasks.

Due to space constraints, we defer some of the proofs and additional results to the Appendix in the online extended version [33].

2 RELATED WORKS

Our work focuses on efficiently mining significant patterns. We now discuss the previous works most related to our contributions. We refer the reader to recent comprehensive reviews and tutorials for an overview of commonly used techniques for mining significant patterns [16, 30].

Several approaches [43–45] have used general methods for multiple hypotheses testing, such as Bonferroni [7] and Holm methods [19], within significant pattern mining. Such methods result in low *statistical power*, since they correct the *p*-value, or a measure of the significance of a pattern, by the number of candidate hypotheses (i.e., the number of patterns), which is extremely large in data mining applications. LAMP [28, 40] is a recently introduced method that partially addresses such issue by selecting a subset of patterns for testing, while discarding the patterns with no chance of being significant. Such approach leads to identifying an improved (i.e., smaller) correction factor, resulting in higher statistical power while controlling the FWER. However, since each selected pattern is still tested as a distinct hypothesis, LAMP still leads to reduced statistical power [25, 39]. Our algorithm FSR instead uses resampled datasets, taking into account dependencies among patterns, to achieve high statistical power.

Several permutation-based methods have been proposed to identify patterns significantly associated with a binary target label. [25, 32, 39] use the Westfall-Young (WY) permutation test [46]. The WY permutation procedure requires to estimate the δ-quantile of the distribution of the minimum (overall all patterns) p-value (or, equivalently, of the distribution of the maximum deviation, over all patterns, of the measure of significance). The use of such quantiles makes WY permutation testing more powerful than LAMP, but also more computationally expensive, since the estimation of such quantiles requires to mine a large number of permuted datasets. For these reasons, all such methods may need impractical resources. Our algorithm FSR instead requires to mine a small number of resampled datasets, leading to an efficient approach even for complex patterns such as subgroups. In addition, permutation-based methods only consider the conditional distribution, where patterns' frequencies and the fraction of elements with target label equal to 1 are assumed fixed in all permuted datasets. Our approach instead works for both the conditional and the unconditional distribution.

The identification of significant patterns with unconditional testing has received scant attention. This is due, in part, to the higher computational cost required for assessing the significance of even a single hypothesis in the unconditional setting, for example using Barnard's test [5]. As far as we know, the only approach for mining significant patterns in a *partially* unconditional setting is SPuManTE [31]. In the partially unconditional setting considered by SPuManTE only the frequencies of the patterns are not fixed, while the target is fixed by design. In contrast our algorithm FSR considers a *fully* unconditional setting, where the fraction of elements with target label equal to 1 is not fixed either. Moreover, SPuManTE leverages specialized techniques for mining significant itemsets that cannot be easily generalized to other pattern types, while our approach applies directly to several pattern languages, including itemsets, sequential patterns, and subgroups.

Other works, orthogonal to ours, consider improving the diversity or limiting redundancy of the output [12, 21, 41].

3 PRELIMINARIES

We consider a *dataset* $\mathcal{D}$ as a collection of m transactions $\mathcal{D} = \{(s_1, \ell_1), \ldots, (s_m, \ell_m)\}$, where each transaction $(s, \ell) \in \mathcal{D}$ is composed by a set s of d *features*, either binary, categorical, or continuous, and a binary *target* variable $\ell \in \{0, 1\}$. More generally, we assume that s belongs to a domain $\mathcal{X}$. By defining the multisets $\mathcal{A} = \{s_1, \ldots, s_m\}$ and $\mathcal{T} = \{\ell_1, \ldots, \ell_m\}$, a dataset $\mathcal{D}$ is also represented by the pair $\mathcal{D} = (\mathcal{A}, \mathcal{T})$. We assume to have a *language* $\mathcal{L}$ containing the *patterns* of potential interest. This scenario captures widely used pattern mining tasks, such as: itemset mining, where all features correspond to (binary) *items* and the language $\mathcal{L}$ corresponds to the set of all *itemsets*, i.e., (non-empty) subsets of items; subgroup mining, where the language $\mathcal{L}$ contains all *subgroups*, i.e., the sets of conjunctions with at most z conditions over features from $\mathcal{A}$, where each condition is either an equality, on a categorical feature, or an interval, on a continuous feature.

Given a transaction (s, ℓ), we use the notation $\mathcal{P} \in s$ to say that the set s of features *supports* pattern $\mathcal{P}$, where the meaning of $\mathcal{P} \in s$ depends on the specific data mining task. For example, for itemset mining, $\mathcal{P} \in s$ means that the pattern $\mathcal{P}$ is contained in

the set s, while for subgroup mining it means that the conditions defined by $\mathcal{P}$ are all satisfied by the features of s. We define the set $C_{\mathcal{P}}(\mathcal{D})$ of transactions in the dataset $\mathcal{D}$ that *support* a pattern $\mathcal{P}$ as $C_{\mathcal{P}}(\mathcal{D}) = \{(s, \ell) \in \mathcal{D} : \mathcal{P} \in s\}$. The *frequency* $f_{\mathcal{P}}(\mathcal{D})$ of a pattern $\mathcal{P}$ in the dataset $\mathcal{D}$ is the fraction of transactions of $\mathcal{D}$ that support $\mathcal{P}$: $f_{\mathcal{P}}(\mathcal{D}) = \frac{1}{m} \sum_{i=1}^{m} \mathbb{1} [\mathcal{P} \in s_i] = \frac{|C_{\mathcal{P}}(\mathcal{D})|}{m}$.

Finally, let $\mu(\mathcal{D})$ denote the average value of the target ℓ for transactions in the dataset $\mathcal{D}$: $\mu(\mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(s,\ell) \in \mathcal{D}} \ell$.

3.1 Significant Patterns

Our goal is to find *significant patterns*, where a pattern $\mathcal{P}$ is *significant* if the presence of $\mathcal{P}$ in a transaction is associated with the target variable of the transaction being 1. In particular, we consider the significance of a pattern in the framework of *statistical significance*, assuming that the transactions $\{(s_1, \ell_1), \ldots, (s_m, \ell_m)\}$ constituting the dataset $\mathcal{D}$ are *samples* from an unknown distribution γ. A pattern $\mathcal{P}$ is associated with the target variable if the probability of the event "$\mathcal{P} \in s$ *and* $\ell = 1$" is higher than the corresponding probability when the event "$\mathcal{P} \in s$" and the event "$\ell = 1$" are independent. Formally, this corresponds to consider the following *null hypothesis* for a pattern $\mathcal{P}$:

$$\Pr_{(s,\ell) \sim \gamma} (\mathcal{P} \in s \wedge \ell = 1) = \Pr_{(s,\ell) \sim \gamma} (\mathcal{P} \in s) \Pr_{(s,\ell) \sim \gamma} (\ell = 1).$$

Since we are interested in patterns with a significant association with the target variable being 1, we are interested in finding patterns for which the following *alternative hypothesis* holds:

$$\Pr_{(s,\ell) \sim \gamma} (\mathcal{P} \in s \wedge \ell = 1) > \Pr_{(s,\ell) \sim \gamma} (\mathcal{P} \in s) \Pr_{(s,\ell) \sim \gamma} (\ell = 1).$$

To this end, we define the *quality*[1] of a pattern $\mathcal{P}$ as

$$q_{\mathcal{P}} = \Pr_{(s,\ell) \sim \gamma} (\mathcal{P} \in s \wedge \ell = 1) - \Pr_{(s,\ell) \sim \gamma} (\mathcal{P} \in s) \Pr_{(s,\ell) \sim \gamma} (\ell = 1).$$

Note that the alternative hypothesis is equivalent to $q_{\mathcal{P}} > 0$, and the null hypothesis is equivalent to $q_{\mathcal{P}} = 0$. Therefore, finding significant patterns is equivalent to finding patterns with quality $q_{\mathcal{P}} > 0$.

For example, consider the study of the association between the characteristics of the users of an online social network and users' interests in a given topic. In this case, each user is a transaction, users' characteristics are the features, and being interested or not in the topic defines the target variable. Significant patterns in this example are associations between users' characteristics and users' interests that are significantly stronger than expected under the null hypothesis of independence between characteristics and interests. For example, if the probability that a user has a given binary feature f is 0.3, and the probability that a user is interested in the topic is 0.5, then under the null hypothesis of independence we have that the probability that a user has feature f *and* is interested in the topic is 0.15. Therefore, the feature f is *significantly associated* with the topic of interest if the *actual* probability that a user has feature f *and* is interested in the topic is > 0.15.

Task Definition. Given a dataset $\mathcal{D}$ and the corresponding pattern language $\mathcal{L}$, our goal is to identify significant patterns, that

---

[1]The quality $q_{\mathcal{P}}$ is often called *leverage* for general patterns [16], but we use the term *quality* given its relation to the 1-quality commonly employed to find interesting subgroups.

is, patterns $\mathcal{P} \in \mathcal{L}$ with quality $q_{\mathcal{P}} > 0$. Since the distribution γ is unknown and we have access only to the dataset $\mathcal{D}$ comprising transactions sampled from γ, we cannot hope to discover *all* significant patterns without errors. As a consequence, we must resort to approximations. In particular, define the subset $\mathcal{L}^{\star}$ of the language $\mathcal{L}$ of patterns for which the null hypothesis holds:

$$\mathcal{L}^{\star} = \{\mathcal{P} \in \mathcal{L} : q_{\mathcal{P}} = 0\}.$$

Our task is then to produce a subset $O \subseteq \mathcal{L}$ of all patterns in the language with *Family-Wise Error Rate* (FWER) below a user-defined threshold δ, such that the probability that O contains at least one element from $\mathcal{L}^{\star}$ is at most δ:

$$\Pr_{\mathcal{D}} \left(O \cap \mathcal{L}^{\star} \neq \emptyset \right) \leq \delta. \tag{1}$$

Note that (1) implies that O is *false discovery free* approximation, which have previously been considered for other pattern mining tasks [36, 37].

Since $q_{\mathcal{P}}$ depends on the unknown distribution γ, we define a statistic $\bar{q}_{\mathcal{P}}(\mathcal{D})$ that corresponds to an estimate of $q_{\mathcal{P}}$ from data and that we will use in our algorithm FSR to identify significant patterns. For each pattern $\mathcal{P}$, we define the functions $f_{\mathcal{P}}$ and $g_{\mathcal{P}}$ as follows: each $f_{\mathcal{P}} : \mathcal{X} \to \{0, 1\}$ is defined as $f_{\mathcal{P}}(s) = \mathbb{1}\,[\mathcal{P} \in s]$, such that $f_{\mathcal{P}}(s) = 1$ if $\mathcal{P} \in s$, and $f_{\mathcal{P}}(s) = 0$ otherwise; each $g_{\mathcal{P}} : \mathcal{X} \times \{0, 1\} \to [-\mu(\mathcal{D}), 1 - \mu(\mathcal{D})]$ is defined as $g_{\mathcal{P}}(s, \ell) = f_{\mathcal{P}}(s)(\ell - \mu(\mathcal{D}))$. Then, the estimate $\bar{q}_{\mathcal{P}}(\mathcal{D})$ of $q_{\mathcal{P}}$ for pattern $\mathcal{P}$ from the dataset $\mathcal{D}$ is

$$\bar{q}_{\mathcal{P}}(\mathcal{D}) = \frac{1}{m} \sum_{i=1}^{m} g_{\mathcal{P}}(s_i, \ell_i).$$

Interestingly, $\bar{q}_{\mathcal{P}}(\mathcal{D})$ corresponds to the 1-quality commonly used in subgroup mining [4] (see Appendix A.1), even if the way it is used in our algorithm FSR (see Section 4) is different from its usual application, due to our focus on statistically significant patterns.

Intuitively, we expect significant patterns to have a sufficiently high empirical quality $\bar{q}_{\mathcal{P}}(\mathcal{D})$ measured on the data $\mathcal{D}$, since $\bar{q}_{\mathcal{P}}(\mathcal{D})$ estimates the (true) quality $q_{\mathcal{P}}$ of $\mathcal{P}$. To take into account the multiple hypothesis testing issue described above, we need to identify a *threshold* ε such that reporting in output all patterns with quality $\geq \varepsilon$ has bounded FWER. Note that ε should be as small as possible in order to have high statistical power (i.e., to report the largest set of results with guarantees on false discoveries). A critical quantity we study to address this issue is the *supremum deviation* of the empirical qualities of non-significant patterns, defined as

$$\sup_{\mathcal{P} \in \mathcal{L}^{\star}} \{\bar{q}_{\mathcal{P}}(\mathcal{D})\}. \tag{2}$$

We address this challenge with FSR: we derive novel analytical bounds on the concentration of the supremum deviation (2), and build on these tools an efficient few-shot resampling algorithm to sharply estimate it. FSR achieves high statistical power while scaling to large datasets and complex languages. Our approach applies to both conditional and unconditional testing, both of great interest in data mining.

3.2 Conditional and Unconditional Testing

When assessing the statistical significance of a pattern $\mathcal{P}$, one has to choose between *conditional* and *unconditional* tests. A conditional test assumes that the data-generating process represented by the unknown distribution γ only produces datasets with m transactions in which both the frequency $f_{\mathcal{P}}(\mathcal{D})$ of pattern $\mathcal{P}$ and the fraction $\mu(\mathcal{D})$ of transactions with target value 1 are the same as in the observed dataset; that is, it *conditions* on the observed variables of interest. In contrast, *unconditional tests* assume that $f_{\mathcal{P}}(\mathcal{D})$ and $\mu(\mathcal{D})$ are the realization of corresponding random variables. Unconditional tests therefore assess the association between a pattern and class labels considering also scenarios (i.e., datasets) where all frequencies of the patterns and/or the average target value may differ from what is observed in the data. Equivalently, conditional tests and unconditional tests are based on different assumptions regarding how data is generated and collected, namely, whether the variables of interest would be the same in a different repetition of the experiment (conditional tests) or not (unconditional tests).

Consider for example the scenario of online social networks described in Section 3.1, and assume for simplicity that we are interested in associations between the single features and the target. If the data is collected so that the total number of transactions for each value of the target is fixed and the fraction of transactions with given values of the features is fixed as well, then conditional testing is more appropriate. If instead one collects the data without constraints on the features/target values (e.g., simply collecting as many transactions as possible), then unconditional testing is more appropriate.

Conditional tests and unconditional tests are therefore both valid and of interest for data mining applications, and the choice between the two classes depends on the specific scenario, even if in practice conditional tests are usually preferred for computational reasons, since unconditional tests need to take into account more uncertainties in the observed quantities. In what follows, we introduce a general algorithm to identify significant patterns for both conditional testing and unconditional testing. Our algorithm is extremely efficient in both cases due to the use of few-shot resampling.

4 FSR ALGORITHM

We now describe our algorithm FSR (Algorithm 1) to find significant patterns for both conditional testing and unconditional testing. We first present the general approach, that is common to both testing scenarios, and then present the details for conditional testing in Section 4.1, and the details for unconditional testing in Section 4.2.

In a nutshell, FSR identifies significant patterns from a dataset $\mathcal{D} = \{(s_1, \ell_1), \ldots, (s_m, \ell_m)\}$ using a few-shot resampling approach to compute rigorous probabilistic bounds to the deviation of the estimated qualities $\bar{q}_{\mathcal{P}}(\mathcal{D})$ of the patterns, under the null hypothesis of no association between the patterns and the target label. It then reports in output all patterns with estimated quality $\bar{q}_{\mathcal{P}}(\mathcal{D})$ above such deviation.

FSR considers a collection $\mathcal{R}^{\star} = \{\mathcal{D}_1^{\star}, \ldots, \mathcal{D}_c^{\star}\}$ of $c \geq 1$ i.i.d. *resampled datasets*, each obtained by *resampling* the target labels of $\mathcal{D}$ while maintaining the same features of the transactions of $\mathcal{D}$. Each resampled dataset is $\mathcal{D}_j^{\star} = \{(s_1, \xi_{1,j}), \ldots, (s_m, \xi_{m,j})\}$, where $\xi_{i,j}$ are i.i.d. random variables with

$$\xi_{i,j} \sim Bern(p), \forall i \in [1, m], \forall j \in [1, c],$$

and $Bern(p)$ is the Bernoulli random variable with parameter p (i.e., it is 1 with probability p, and 0 otherwise).

We now describe the general approach followed by FSR (Algorithm 1). FSR starts by computing an upper bound ε_T to the deviation between the average target value $\mu(\mathcal{D})$ observed in $\mathcal{D}$ and its expected value $\mu = \mathbb{E}_{\mathcal{D}}[\mu(\mathcal{D})]$ under the null hypothesis (line 1). Note that μ is the probability that a sample from the unknown distribution γ has target label ℓ equal to 1, that is, $\mu = \Pr_{(s,\ell)}(\ell = 1)$. The computation of ε_T is performed by the procedure boundTarget and it depends on whether one is interested in conditional testing or in unconditional testing. The details of boundTarget for the two settings are described in Section 4.1 and in Section 4.2, respectively. Then FSR uses ε_T to obtain the upper bound $\hat{\mu}$ (line 2) and lower bound $\check{\mu}$ (line 3) to μ (we trivially assume $0 \leq \check{\mu} \leq \hat{\mu} \leq 1$). It then uses the procedure resampleTarget to generate c resampled datasets $\mathcal{R}^{\star}$ (line 4) assigning to each transaction in the datasets of $\mathcal{R}^{\star}$ the target label 1 with probability $p = \hat{\mu}$ (i.e, the upper bound to μ). Note that the resampleTarget procedure is the same for both conditional and unconditional testing. The algorithm then computes, from each of the c resampled datasets of $\mathcal{R}^{\star}$, an estimate of the maximum deviation of the empirical quality $\bar{q}_{\mathcal{P}}(\mathcal{D})$ for non-significant patterns. This is achieved by computing, for every resampled dataset $\mathcal{D}_j^{\star}$, the quantity $\sup_{\mathcal{P} \in \mathcal{L}} \bar{q}_{\mathcal{P}}(\mathcal{D}_j^{\star}, \check{\mu})$, where $\bar{q}_{\mathcal{P}}(\mathcal{D}_j^{\star}, \check{\mu}) = \frac{1}{m} \sum_{i=1}^{m} f_{\mathcal{P}}(s_i)(\xi_{i,j} - \check{\mu})$. The value $\sup_{\mathcal{P} \in \mathcal{L}} \bar{q}_{\mathcal{P}}(\mathcal{D}_j^{\star}, \check{\mu})$ can be interpreted as an empirical estimate of the maximum quality of non-significant patterns, measured from datasets sampled from the null distribution. Note that we use $\bar{q}_{\mathcal{P}}(\mathcal{D}_j^{\star}, \check{\mu})$ as $\check{\mu}$ is a lower bound to μ; consequently, $\bar{q}_{\mathcal{P}}(\mathcal{D}_j^{\star}, \check{\mu})$ provides a proper *upper bound* to such maximum deviation. Then, note that it is necessary to consider the supremum over the language $\mathcal{L}$, since the set of true null hypothesis $\mathcal{L}^{\star}$ is unknown. We remark that the computation of $\sup_{\mathcal{P} \in \mathcal{L}} \bar{q}_{\mathcal{P}}(\mathcal{D}_j^{\star}, \check{\mu})$ can be performed with fast pattern enumeration strategies, similar to the ones leveraged by previous methods for frequent and significant pattern mining; in fact, FSR can be combined with any efficient exploration procedure, such as the ones that explore the search space of the pattern language of interest using pruning bounds, for instance depth-first [25, 28, 39] or best-first searches [29, 32, 34]. The algorithm then stores the empirical deviations $\sup_{\mathcal{P} \in \mathcal{L}} \bar{q}_{\mathcal{P}}(\mathcal{D}_j^{\star}, \check{\mu})$ in the variables d_j. Then, the average maximum deviation $\tilde{d}(\mathcal{R}^{\star}, \check{\mu})$ over the c resampled datasets is computed (line 6) as the mean of the values $\{d_j, j \in [1, c]\}$, and it is then used to compute a rigorous upper bound ε to the supremum deviation $\sup_{\mathcal{P} \in \mathcal{L}^{\star}} \{\bar{q}_{\mathcal{P}}(\mathcal{D})\}$ (Eq. (2)) under the null hypothesis (line 7) through the procedure boundStatistic. The implementation of this procedure, i.e., the returned value of ε, depends on whether conditional testing or unconditional testing is considered, and it is described in Section 4.1 and Section 4.2, respectively. In both cases, we use advanced concentration bounds [8, 26] that allow us to obtain small values of ε with a small number c of resampled datasets. Finally, the set of patterns with estimated quality $\bar{q}_{\mathcal{P}}(\mathcal{D})$ greater than $\varepsilon + \varepsilon_T f_{\mathcal{P}}(\mathcal{D})$ is reported in output (lines 8-9).

Note that while the computation of ε_T and ε depends on whether conditional testing or unconditional testing is considered, the overall approach followed by FSR is the same in both cases. In particular, for both cases FSR relies on the resampled datasets $\mathcal{R}^{\star}$ to estimate the maximum deviation, over all patterns $\mathcal{P} \in \mathcal{L}$, of the estimate

---

Algorithm 1: FSR

Input: Pattern language $\mathcal{L}$; dataset $\mathcal{D}$ of m transactions; $c \geq 1$; $\delta \in (0, 1)$.

Output: Set $O \subseteq \mathcal{L}$ of significant patterns with FWER $\leq \delta$.

1 $\varepsilon_T \leftarrow$ boundTarget$(\mu(\mathcal{D}), m, \delta)$;

2 $\hat{\mu} \leftarrow \mu(\mathcal{D}) + \varepsilon_T$;

3 $\check{\mu} \leftarrow \mu(\mathcal{D}) - \varepsilon_T$;

4 $\mathcal{R}^{\star} \leftarrow$ resampleTarget$(\mathcal{D}, c, \hat{\mu})$;

5 **forall** $j \in [1, c]$ **do** $d_j \leftarrow \sup_{\mathcal{P} \in \mathcal{L}} \{\bar{q}_{\mathcal{P}}(\mathcal{D}_j^{\star}, \check{\mu})\}$;

6 $\tilde{d}(\mathcal{R}^{\star}, \check{\mu}) \leftarrow \frac{1}{c} \sum_{j=1}^{c} d_j$;

7 $\varepsilon \leftarrow$ boundStatistic$(\mathcal{D}, \mathcal{L}, \tilde{d}(\mathcal{R}^{\star}, \check{\mu}), \delta)$;

8 $O \leftarrow \{\mathcal{P} \in \mathcal{L} : \bar{q}_{\mathcal{P}}(\mathcal{D}) \geq \varepsilon + \varepsilon_T f_{\mathcal{P}}(\mathcal{D})\}$;

9 **return** O;

$\bar{q}_{\mathcal{P}}(\mathcal{D})$ from the (unknown) value $q_{\mathcal{P}}$ under the null hypothesis. Moreover, the exact same procedure is used to generate the resampled datasets. Note that our approach is similar to permutation approaches for significant pattern mining, which use permuted datasets to estimate the significance of patterns, but with two crucial differences. First, our datasets are obtained by resampling the target values, and not by permuting them, which allows us to obtain rigorous bounds for both conditional and unconditional testing, while permutation approaches can be used for conditional testing only. Second, since our analysis depends on the *expectation* of the maximum deviation, we can employ advanced bounds on the concentration of the expected value of functions of independent random variables; this allows us to use a small number c of resampled datasets, as shown by our analysis and experimental evaluation. This is in contrast with permutation approaches (e.g., the ones based on WY permutation testing [25, 32, 39]) that instead estimate the quantiles of the distribution of the maximum deviation using a large number of permutations (see also Section A.2 in Appendix for a more detailed comparison).

4.1 FSR for Conditional Testing

We now describe the details of procedures boundTarget and boundStatistic for the version of FSR that uses conditional testing, which we refer to as FSR-C. As a reminder, a conditional test for our problem assumes that the average target value μ and patterns frequencies $f_{\mathcal{P}}(\mathcal{D})$ are fixed, for all $\mathcal{P} \in \mathcal{L}$, to the values observed in the dataset $\mathcal{D}$.

For boundTarget, since the average target value μ is fixed to the value $\mu(\mathcal{D})$ observed in the dataset $\mathcal{D}$, the bound ε_T on its deviation from the expectation is 0, that is, boundTarget simply returns 0. Note that this implies that the output of FSR-C consists of all patterns in $\mathcal{L}$ with $\bar{q}_{\mathcal{P}}(\mathcal{D}) \geq \varepsilon$ (line 8).

For boundStatistic, we now show how to compute a rigorous probabilistic bound ε to the supremum deviation $\sup_{\mathcal{P} \in \mathcal{L}^{\star}} \{\bar{q}_{\mathcal{P}}(\mathcal{D})\}$ (Eq. (2)) under the null hypothesis. The bound is obtained by computing the average maximum deviation $\tilde{d}(\mathcal{R}^{\star}, \check{\mu})$ over c resampled datasets, and then applying advanced concentration results [8]. Note that since $\varepsilon_T = 0$, in this case $\tilde{d}(\mathcal{R}^{\star}, \check{\mu}) = \tilde{d}(\mathcal{R}^{\star}, \mu(\mathcal{D}))$.

Since FSR-C is based on resampled datasets, where each target label is sampled independently, our estimate $\tilde{d}(\mathcal{R}^{\star}, \mu(\mathcal{D}))$ of the expected maximum deviation is not based on the conditional distribution assumed by conditional tests, where the fraction of target labels equal to 1 is exactly $\mu(\mathcal{D})$ in *every* dataset (while in our resampled datasets such fraction may vary, and it is equal to $\mu(\mathcal{D})$ only in expectation). We therefore need to relate the supremum deviation of the observed quality of patterns on resampled datasets with the one observed in datasets sampled from the conditional distribution. Interestingly, we show that the resampling and conditional distributions are *closely related*, in the sense that high probability bounds for the former also apply to the latter.

We first prove a general result (Lemma 1) relating the expectation of monotone functions of permutations of binary vectors, corresponding to the conditional distribution, with the expectation taken w.r.t. to independent resamples, corresponding to resampled datasets. For an integer k with $0 \le k \le m$, define the set $B(k)$ of binary vectors with k entries equal to one as

$$B(k) = \left\{ \mathbf{v} \in \{0, 1\}^m, \sum_{i=1}^m \mathbf{v}_i = k \right\},$$

and let $U(B(k))$ be the uniform distribution over the set $B(k)$. Equivalently, $U(B(k))$ corresponds to the set of uniform permutations of a binary vector with k ones. Then, define $I(p)$ as a probability distribution over $\{0, 1\}^m$, such that each entry $\mathbf{v}_i$ of a random vector $\mathbf{v}$ taken from $I(p)$ is an i.i.d. Bernoulli r.v. with $\Pr(\mathbf{v}_i = 1) = p = 1 - \Pr(\mathbf{v}_i = 0)$, for some $p \in [0, 1]$.

Lemma 1. *Let $f : \{0, 1\}^m \to \mathbb{R}$ be a nonnegative function such that $\mathbb{E}_{\mathbf{v} \sim U(B(k))}[f(\mathbf{v})]$ is either monotonically increasing or monotonically decreasing in k. It holds*

$$\mathbb{E}_{\mathbf{v} \sim U(B(k))}[f(\mathbf{v})] \le 2 \mathbb{E}_{\mathbf{v} \sim I(k/m)}[f(\mathbf{v})].$$

To prove Lemma 1, we need the following technical result, providing bounds to the probability that a Binomial random variable exceeds its expectation [20].

Lemma 2. *Let μm be an integer. Then it holds*

$$\Pr_{\mathbf{v} \sim I(\mu)}\left(\sum_{i=1}^m \mathbf{v}_i > \mu m \right) < \frac{1}{2} < \Pr_{\mathbf{v} \sim I(\mu)}\left(\sum_{i=1}^m \mathbf{v}_i \ge \mu m \right).$$

We now prove Lemma 1.

PROOF OF LEMMA 1. We prove the result assuming that $\mathbb{E}_{\mathbf{v} \sim U(B(k))}[f(\mathbf{v})]$ is monotonically increasing in k, as the other case is analogous. First, we note that

$$\mathbb{E}_{\mathbf{v} \sim U(B(k))}[f(\mathbf{v})] = \mathbb{E}_{\mathbf{v} \sim I(k/m)}\left[f(\mathbf{v}) \mid \sum_{i=1}^m \mathbf{v}_i = k \right].$$

Therefore, we have

$$\mathbb{E}_{\mathbf{v} \sim I(k/m)}[f(\mathbf{v})]$$
$$= \sum_{j=0}^m \mathbb{E}_{\mathbf{v} \sim I(k/m)}\left[f(\mathbf{v}) \mid \sum_{i=1}^m \mathbf{v}_i = j \right] \Pr_{\mathbf{v} \sim I(k/m)}\left(\sum_{i=1}^m \mathbf{v}_i = j \right)$$
$$\ge \sum_{j=k}^m \mathbb{E}_{\mathbf{v} \sim I(k/m)}\left[f(\mathbf{v}) \mid \sum_{i=1}^m \mathbf{v}_i = j \right] \Pr_{\mathbf{v} \sim I(k/m)}\left(\sum_{i=1}^m \mathbf{v}_i = j \right)$$
$$\ge \mathbb{E}_{\mathbf{v} \sim I(k/m)}\left[f(\mathbf{v}) \mid \sum_{i=1}^m \mathbf{v}_i = k \right] \sum_{j=k}^m \Pr_{\mathbf{v} \sim I(k/m)}\left(\sum_{i=1}^m \mathbf{v}_i = j \right)$$
$$= \mathbb{E}_{\mathbf{v} \sim U(B(k))}[f(\mathbf{v})] \Pr_{\mathbf{v} \sim I(k/m)}\left(\sum_{i=1}^m \mathbf{v}_i \ge k \right)$$
$$\ge \mathbb{E}_{\mathbf{v} \sim U(B(k))}[f(\mathbf{v})] \frac{1}{2},$$

where the last inequality follows from Lemma 2. □

We make use of Lemma 1 to prove the following result.

Theorem 3. *Define the constant $\bar{\mu} = \mu(\mathcal{D})$ and let $k = \bar{\mu}m$. For any $z \ge 0$, it holds*

$$\Pr_{\mathbf{v} \sim U(B(k))}\left(\sup_{\mathcal{P} \in \mathcal{L}^{\star}} \frac{1}{m} \sum_{i=1}^m f_{\mathcal{P}}(s_i)(\mathbf{v}_i - \bar{\mu}) \ge z \right)$$
$$\le 2 \Pr_{\mathbf{v} \sim I(\bar{\mu})}\left(\sup_{\mathcal{P} \in \mathcal{L}^{\star}} \frac{1}{m} \sum_{i=1}^m f_{\mathcal{P}}(s_i)(\mathbf{v}_i - \bar{\mu}) \ge z \right).$$

PROOF OF THEOREM 3. Define the function $g : \{0, 1\}^m \to \{0, 1\}$

$$g(\mathbf{v}) = \mathbb{1}\left[\sup_{\mathcal{P} \in \mathcal{L}^{\star}} \frac{1}{m} \sum_{i=1}^m f_{\mathcal{P}}(s_i)(\mathbf{v}_i - \bar{\mu}) \ge z \right].$$

In order to apply Lemma 1, we prove that $\mathbb{E}_{\mathbf{v} \sim U(B(k))}[g(\mathbf{v})]$ is nondecreasing with k. This is equivalent to show that

$$\mathbb{E}_{\mathbf{v} \sim U(B(k))}[g(\mathbf{v})] \le \mathbb{E}_{\mathbf{v} \sim U(B(k'))}[g(\mathbf{v})],$$

where $k' = k + j$, for any pair of integers $k \in [0, m]$ and $j \in [0, m - k]$. To do so, we build a coupling $\pi(k, k')$ between the two distributions $U(B(k))$ and $U(B(k'))$ as follows. For any $\mathbf{v}$ taken from $U(B(k))$, define $\mathbf{v}'$ as a copy of $\mathbf{v}$ (i.e., such that $\mathbf{v}_i = \mathbf{v}'_i, \forall i \in [1, m]$), that is modified according to the following procedure: for j times, select uniformly at random an index i such that $\mathbf{v}'_i = 0$, and set $\mathbf{v}'_i$ to 1. Denote the pair $\mathbf{v}, \mathbf{v}'$ sampled according to $\pi(k, k')$ as the output of this procedure. It is immediate to observe that $\mathbf{v}' \sim U(B(k'))$ (i.e., that the marginal distribution of $\mathbf{v}'$ is $U(B(k'))$), and that $\mathbf{v}_i \le \mathbf{v}'_i, \forall i \in [1, m]$. This implies that, for any pair of vectors $\mathbf{v}, \mathbf{v}' \sim \pi(k, k')$, it holds

$$\sup_{\mathcal{P} \in \mathcal{L}^{\star}} \frac{1}{m} \sum_{i=1}^m f_{\mathcal{P}}(s_i)(\mathbf{v}_i - \bar{\mu}) \le \sup_{\mathcal{P} \in \mathcal{L}^{\star}} \frac{1}{m} \sum_{i=1}^m f_{\mathcal{P}}(s_i)(\mathbf{v}'_i - \bar{\mu}).$$

A consequence of this fact is

$$\mathbb{E}_{\mathbf{v}'\sim U(B(k+j))}\left[g(\mathbf{v}')\right]$$

$$= \Pr_{\mathbf{v}'\sim U(B(k+j))}\left(\sup_{\mathcal{P}\in\mathcal{L}^\star}\frac{1}{m}\sum_{i=1}^m f_\mathcal{P}(s_i)(\mathbf{v}'_i-\bar\mu)\geq z\right)$$

$$= \Pr_{\mathbf{v},\mathbf{v}'\sim\pi(k,k')}\left(\sup_{\mathcal{P}\in\mathcal{L}^\star}\frac{1}{m}\sum_{i=1}^m f_\mathcal{P}(s_i)(\mathbf{v}'_i-\bar\mu)\geq z\right)$$

$$\geq \Pr_{\mathbf{v},\mathbf{v}'\sim\pi(k,k')}\left(\sup_{\mathcal{P}\in\mathcal{L}^\star}\frac{1}{m}\sum_{i=1}^m f_\mathcal{P}(s_i)(\mathbf{v}_i-\bar\mu)\geq z\right)$$

$$= \Pr_{\mathbf{v}\sim U(B(k))}\left(\sup_{\mathcal{P}\in\mathcal{L}^\star}\frac{1}{m}\sum_{i=1}^m f_\mathcal{P}(s_i)(\mathbf{v}_i-\bar\mu)\geq z\right)$$

$$= \mathbb{E}_{\mathbf{v}\sim U(B(k))}\left[g(\mathbf{v})\right].$$

Therefore, g is nonnegative and $\mathbb{E}_{\mathbf{v}\sim U(B(k))}\left[g(\mathbf{v})\right]$ is nondecreasing in k. We apply Lemma 1 to the function g, obtaining the statement. □

We note that Theorem 3 is precisely what we seek: it implies that large supremum deviations that are unlikely in the independent resamples distribution $\mathbf{v}\sim I(\bar\mu)$ are also unlikely in the conditional distribution $\mathbf{v}\sim U(B(k))$. By using Theorem 3, we prove the following result, that implies strong concentration of the supremum deviation of pattern qualities w.r.t. their expectations, taken w.r.t. independent resamples of the target labels rather than permutations.

THEOREM 4. *Define $\bar\mu = \mu(\mathcal{D})$, and*

$$\omega = (1-\bar\mu)\min\left\{\bar\mu,\ \sup_{\mathcal{P}\in\mathcal{L}}\frac{1}{m}\sum_{i=1}^m f_\mathcal{P}(s_i)\right\}.$$

With probability at least $1-\delta$ over $\mathbf{v}\sim U(B(k))$, it holds

$$\sup_{\mathcal{P}\in\mathcal{L}^\star}\frac{1}{m}\sum_{i=1}^m f_\mathcal{P}(s_i)(\mathbf{v}_i-\bar\mu)$$

$$\leq \mathbb{E}_{\mathbf{v}\sim I(\bar\mu)}\left[\sup_{\mathcal{P}\in\mathcal{L}^\star}\frac{1}{m}\sum_{i=1}^m f_\mathcal{P}(s_i)(\mathbf{v}_i-\bar\mu)\right] + \sqrt{\frac{2\omega\log\left(\frac{2}{\delta}\right)}{m}}.$$

To prove Theorem 4, we need the following technical result regarding the concentration of functions of independent random variables.

THEOREM 5 (THEOREM 6.7 OF [8]). *Let $g:\mathcal{Y}^m\to\mathbb{R}$ be a function, and let $X = (X_1,\ldots,X_m)\in\mathcal{Y}^m$ a collection of m independent random variables. Define $\bar X^j = (X_1,\ldots,X_{j-1},X'_j,X_{j+1},\ldots,X_m)\in\mathcal{Y}^m$ as a copy of X, where its j-th element X_j is replaced by an independent copy X'_j. Assume that, for some constant $q\geq 0$,*

$$\sum_{j=1}^m\mathbb{E}\left[\left(g(X)-g(\bar X^j)\right)^2_+\mid X\right]\leq q$$

holds almost surely. Then, it holds, for all $t\geq 0$,

$$\Pr\left(g(X)\geq\mathbb{E}_X\left[g(X)\right]+t\right)\leq\exp(-t^2/(2q)).$$

We use these results to prove Theorem 4.

PROOF OF THEOREM 4. Define the function $g:\{0,1\}^m\to\mathbb{R}$ as

$$g(\mathbf{v}) = \sup_{\mathcal{P}\in\mathcal{L}^\star}\frac{1}{m}\sum_{i=1}^m f_\mathcal{P}(s_i)(\mathbf{v}_i-\bar\mu).$$

We first note that, from the result of Theorem 3, it is sufficient to show that, choosing the constants t and z as

$$t = \sqrt{\frac{2\omega\log(\frac{2}{\delta})}{m}},\quad z = \mathbb{E}_{\mathbf{v}\sim I(\bar\mu)}\left[g(\mathbf{v})\right]+t,\qquad(3)$$

it holds

$$\Pr_{\mathbf{v}\sim I(\bar\mu)}\left(g(\mathbf{v})\geq z\right)\leq\delta/2.$$

To do so, we make use of Theorem 5. Define $\bar{\mathbf{v}}^j$ a copy of $\mathbf{v}$, where its j-th element $\mathbf{v}_j$ is replaced by an independent copy $\mathbf{v}'_j$. We want to upper bound

$$\sum_{j=1}^m\mathbb{E}\left[\left(g(\mathbf{v})-g(\bar{\mathbf{v}}^j)\right)^2_+\mid\mathbf{v}\right]$$

below some constant $q\geq 0$. Define $\mathcal{P}^\star$ as one of the elements of $\mathcal{L}^\star$ that achieve the supremum in $g(\mathbf{v})$. We observe that

$$\left(g(\mathbf{v})-g(\bar{\mathbf{v}}^j)\right)_+$$

$$\leq \left(\frac{1}{m}\sum_{i=1}^m f_{\mathcal{P}^\star}(s_i)(\mathbf{v}_i-\bar\mu)-\frac{1}{m}\sum_{i=1}^m f_{\mathcal{P}^\star}(s_i)(\bar{\mathbf{v}}^j_i-\bar\mu)\right)_+$$

$$= \left(\frac{1}{m}f_{\mathcal{P}^\star}(s_j)(\mathbf{v}_j-\bar\mu)-\frac{1}{m}f_{\mathcal{P}^\star}(s_j)(\mathbf{v}'_j-\bar\mu)\right)_+$$

$$= \left(\frac{1}{m}f_{\mathcal{P}^\star}(s_j)(\mathbf{v}_j-\mathbf{v}'_j)\right)_+.\qquad(4)$$

We now observe that $\mathbf{v}'_j = 0$ is the only possible value of $\mathbf{v}'_j$ that makes (4) be > 0. Therefore, we have

$$\sum_{j=1}^m\mathbb{E}\left[\left(g(\mathbf{v})-g(\bar{\mathbf{v}}^j)\right)^2_+\mid\mathbf{v}\right]$$

$$\leq \sum_{j=1}^m\mathbb{E}\left[\left(\left(\frac{1}{m}f_{\mathcal{P}^\star}(s_j)(\mathbf{v}_j-\mathbf{v}'_j)\right)_+\right)^2\mid\mathbf{v}\right]$$

$$= (1-\bar\mu)\sum_{j=1}^m\left(\frac{1}{m}f_{\mathcal{P}^\star}(s_j)\mathbf{v}_j\right)^2$$

$$= \frac{(1-\bar\mu)}{m}\sum_{j=1}^m\left(\frac{1}{m}f_{\mathcal{P}^\star}(s_j)\mathbf{v}_j\right)$$

$$\leq \frac{(1-\bar\mu)}{m}\sup_{\mathcal{P}\in\mathcal{L}^\star}\sum_{j=1}^m\left(\frac{1}{m}f_\mathcal{P}(s_j)\mathbf{v}_j\right)\leq\frac{\omega}{m}.$$

We apply Theorem 5 to the function g with $q = \omega/m$, obtaining that

$$\Pr_{\mathbf{v}\sim I(\bar\mu)}\left(g(\mathbf{v})\geq\mathbb{E}_{\mathbf{v}\sim I(\bar\mu)}\left[g(\mathbf{v})\right]+t\right)\leq\exp(-mt^2/(2\omega)).$$

Setting t as in (3), it is immediate to observe that the probability above is $\leq\delta/2$, obtaining the statement. □

Note that Theorem 4 provides a probabilistic upper bound, holding with probability at least $1 - \delta$, to the maximum observed value of the pattern quality $\sup_{\mathcal{P} \in \mathcal{L}^\star} \frac{1}{m} \sum_{i=1}^{m} f_{\mathcal{P}}(s_i)(\mathbf{v}_i - \bar{\mu})$ when the conditional distribution $\mathbf{v} \sim U(B(k))$ is considered, in terms of the expectation of the maximum observed value of the pattern quality according to the resampled distribution $\mathbf{v} \sim I(\bar{\mu})$. Then, the following result proves that the estimation $\tilde{d}(\mathcal{R}^\star, \breve{\mu})$ of the expected deviation in the upper bound above is very accurate.

THEOREM 6. *With probability at least* $1 - \delta/4$ *over* $\mathcal{R}^\star$, *it holds*

$$\mathbb{E}_{\mathbf{v} \sim I(\bar{\mu})}\left[\sup_{\mathcal{P} \in \mathcal{L}^\star} \frac{1}{m} \sum_{i=1}^{m} f_{\mathcal{P}}(s_i)(\mathbf{v}_i - \bar{\mu})\right] \leq \tilde{d}(\mathcal{R}^\star, \breve{\mu}) + \sqrt{\frac{\log\left(\frac{4}{\delta}\right)}{2cm}}.$$

By combining the results above, we prove the guarantees provided by FSR for the task of finding significant patterns when conditional testing is used. In particular, the following Corollary proves that, when boundStatistic returns the value ε defined below, then the output set O of significant patterns returned by FSR has FWER bounded by the user-defined parameter δ.

COROLLARY 7. *Fix* $\delta \in (0, 1)$ *and* $c \geq 1$. *Let* O *be the output of FSR with input parameters* $\mathcal{L}, \mathcal{D}, c, \delta$, *and let*

$$\varepsilon = \tilde{d}(\mathcal{R}^\star, \breve{\mu}) + \sqrt{\frac{2\omega \log\left(\frac{4}{\delta}\right)}{m}} + \sqrt{\frac{\log\left(\frac{4}{\delta}\right)}{2cm}}$$

be the value returned by boundStatistic *in line 7. Then, the set* O *has FWER* $\leq \delta$ *under the conditional null distribution.*

Note that the value of ε returned by boundStatistic requires to compute $\sup_{\mathcal{P} \in \mathcal{L}}\{\bar{q}_{\mathcal{P}}(\mathcal{D}_j^\star, \breve{\mu})\}$, which is costly, but only needs to be performed on c resampled datasets. As we will show in our experimental evaluation (see Section 5), small values of c suffice. Moreover, the maximum frequency of a pattern in the language $\mathcal{L}$ is required in order to compute ω (see Theorem 4). Such maximum frequency can be computed very efficiently in most data mining tasks. For example: in itemset mining it corresponds to the frequency of the most frequent item; in subgroup mining, it is equal to 1 whenever a continuous feature is present and the conditions defining the pattern language $\mathcal{L}$ include inequalities.

Power analysis. The results above show that FSR rigorously controls the probability of *false positives* (i.e., patterns for which the null hypothesis hold but are wrongly reported in output as significant). However, they do not provide guarantees on the *power* of FSR, that is, its ability to report patterns $\mathcal{P}$ with sufficiently high quality $q_{\mathcal{P}}$. The following result provides guarantees on the power of FSR-C, the version of FSR that uses conditional testing, for the pattern language of subgroups. Our analysis is based on a probabilistic upper bound to $\tilde{d}(\mathcal{R}^\star, \breve{\mu})$, obtained from bounds to the *pseudodimension* [24, 35, 38] of subgroups, and an advanced concentration bound for sums of *dependent random variables* [13], that hold under mild (but necessary) assumptions on the distribution of alternative hypotheses (the set of patterns with $q_{\mathcal{P}} > 0$). More precisely, we assume that the target labels of the transactions that support a pattern $\mathcal{P}$ with $q_{\mathcal{P}} > 0$ are distributed according to a *noncentral hypergeometric distribution* [42], i.e., a biased version of the standard hypergeometric distribution. We provide the proofs and additional details in Appendix A.5.

THEOREM 8. *Fix* $\delta \in (0, 1)$ *and* $c, z \geq 1$. *Let* O *be the output of FSR with input parameters* $\mathcal{L}, \mathcal{D}, c$, *and* δ, *where* $\mathcal{L}$ *is the language of subgroups composed by conjunctions with at most* z *conditions over* d *continuous features. Then, with probability at least* $1 - \delta$, O *contains all patterns with quality* $q_{\mathcal{P}}$ *satisfying*

$$q_{\mathcal{P}} \geq \sqrt{\frac{2\hat{\omega}z \ln\left(\frac{e^3 dm^2}{4z^3}\right)}{m}} + \sqrt{\frac{\ln\left(\frac{2}{\delta}\right)}{2cm}} + \frac{z \ln\left(\frac{e^3 dm^2}{4z^3}\right)}{3m}$$

$$+ \sqrt{\frac{2\omega \log\left(\frac{4}{\delta}\right)}{m}} + \sqrt{\frac{\log\left(\frac{4}{\delta}\right)}{2cm}} + \sqrt{\frac{2\hat{f}(\mathcal{D})z \ln\left(\frac{e^3 dm^2}{2z^3 \delta}\right)}{m}},$$

where $\hat{f}(\mathcal{D}) = \sup_{\mathcal{P} \in \mathcal{L}} f_{\mathcal{P}}(\mathcal{D})$ *and* $\hat{\omega} = \hat{\mu}(1 - \breve{\mu})\hat{f}(\mathcal{D})$.

4.2 FSR for Unconditional Testing

We now describe the details of procedures boundTarget and boundStatistic for the version of FSR that uses unconditional testing, which we refer to as FSR-U. As a reminder, in our scenario an unconditional test assumes that the average target value and patterns frequencies $f_{\mathcal{P}}(\mathcal{D})$ are observations of random variables, whose expected values are unknown. In particular, here we consider the transactions $\{(s_1, \ell_1), \ldots, (s_m, \ell_m)\}$ constituting the dataset $\mathcal{D}$ as *i.i.d.* samples from an unknown distribution γ. This scenario is more complex than the conditional testing one, since we need to account for the unknown deviation of all observed values when computing a bound to the maximum pattern quality under the null hypothesis. However, we show that the use of resampled datasets allow us to efficiently take into account such deviations.

For boundTarget, recall that μ is the the probability that a sample from the unknown distribution γ has target label ℓ equal to 1, such that $\mu = \mathbb{E}_{\mathcal{D}}[\mu(\mathcal{D})] = \Pr_{(s,\ell) \sim \gamma}(\ell = 1)$. More importantly, in this setting μ is unknown (as γ is). However, given that the samples in $\mathcal{D}$ are i.i.d., the following result provides a probabilistic bound ε_T to the deviation of the observed value $\mu(\mathcal{D})$ from its expectation μ.

LEMMA 9. *Let* $\mathcal{D}$ *be a collection of* m *samples taken i.i.d. from* γ. *For* $\delta \in (0, 1)$, *it holds with probability* $\geq 1 - \delta/4$

$$|\mu(\mathcal{D}) - \mu| \leq \varepsilon_T \doteq \sqrt{\frac{2 \min\left\{\mu(\mathcal{D}), \frac{1}{4}\right\} \ln\left(\frac{8}{\delta}\right)}{m}} + \frac{2 \ln\left(\frac{8}{\delta}\right)}{m}.$$

boundTarget returns the value ε_T as defined in Lemma 9, which requires the values $\mu(\mathcal{D}), \delta$, and m.

For boundStatistic, the computation of the upper bound ε to the maximum deviation between the observed pattern qualities and their expectations is more involved than in the conditional case. We show that the quality $q_{\mathcal{P}}$ of a pattern $\mathcal{P}$, which depends on several unknown quantities (see Section 3.1), can be sharply estimated from a dataset $\mathcal{D}$ *provided* that μ is known, which is *not* the case for the unconditional distribution. We then show that using $\mu(\mathcal{D})$ in place of μ provides a good estimate of $q_{\mathcal{P}}$, and prove a probabilistic upper bound on the difference between the two estimates.

First, we introduce the function family that we use to analyze the supremum deviation of the empirical quality of patterns from the dataset $\mathcal{D}$. We define the family of functions $g_{\mathcal{P}}^* : \mathcal{X} \times \{0, 1\} \to [-\mu, 1 - \mu]$, where $g_{\mathcal{P}}^*$ is defined as $g_{\mathcal{P}}^*(s, \ell) = f_{\mathcal{P}}(s)(\ell - \mu)$. Note that $g_{\mathcal{P}}^*(s, \ell) = f_{\mathcal{P}}(s)(\ell - \mu)$ corresponds to the function $g_{\mathcal{P}}(s, \ell)$ used by the FSR statistic where the unknown value μ is used in

place of its estimate $\mu(\mathcal{D})$ obtained from dataset $\mathcal{D}$. Define the estimator $q_{\mathcal{P}}(\mathcal{D})$ of the quality $q_{\mathcal{P}}$ of $\mathcal{P}$ from $\mathcal{D}$ as the average of $g_{\mathcal{P}}^*$ over $\mathcal{D}$, that is

$$q_{\mathcal{P}}(\mathcal{D}) = \frac{1}{m} \sum_{i=1}^{m} g_{\mathcal{P}}^*(s_i, \ell_i).$$

Note that $\mathbb{E}_{\mathcal{D}}\left[q_{\mathcal{P}}(\mathcal{D})\right] = q_{\mathcal{P}}$, as $q_{\mathcal{P}}(\mathcal{D})$ is an unbiased estimator of $q_{\mathcal{P}}$. However, $q_{\mathcal{P}}(\mathcal{D})$ depends on the unknown quantity μ. Even if $q_{\mathcal{P}}(\mathcal{D}) \neq \bar{q}_{\mathcal{P}}(\mathcal{D})$ (since $\mu(\mathcal{D})$ may be $\neq \mu$), FSR-U exploits the fact that $\mu(\mathcal{D})$ is sharply concentrated around μ, as proved in Lemma 9. This implies that the maximum deviation $\sup_{\mathcal{P} \in \mathcal{L}} |q_{\mathcal{P}}(\mathcal{D}) - \bar{q}_{\mathcal{P}}(\mathcal{D})|$ for all patterns $\mathcal{P} \in \mathcal{L}$ can be sharply estimated from $\mathcal{D}$. To this aim, we prove the following result.

THEOREM 10. *Let $\mathcal{D}$ be a collection of m samples taken i.i.d. from γ. For $\delta \in (0, 1)$, with probability $\geq 1 - \delta/4$ it holds*

$$|q_{\mathcal{P}}(\mathcal{D}) - \bar{q}_{\mathcal{P}}(\mathcal{D})| \leq \varepsilon_T f_{\mathcal{P}}(\mathcal{D}), \forall \mathcal{P} \in \mathcal{L}.$$

PROOF. Using Lemma 9, $\forall \mathcal{P} \in \mathcal{L}$ it holds with prob. $\geq 1 - \delta/4$

$$|q_{\mathcal{P}}(\mathcal{D}) - \bar{q}_{\mathcal{P}}(\mathcal{D})| = \left| \frac{1}{m} \sum_{i=1}^{m} g_{\mathcal{P}}^*(s_i, \ell_i) - \frac{1}{m} \sum_{i=1}^{m} g_{\mathcal{P}}(s_i, \ell_i) \right|$$

$$= \left| \frac{1}{m} \sum_{i=1}^{m} f_{\mathcal{P}}(s_i)(\mu(\mathcal{D}) - \mu) \right| \leq \varepsilon_T f_{\mathcal{P}}(\mathcal{D}). \qquad \square$$

We remark that the definition of $q_{\mathcal{P}}(\mathcal{D})$ is crucial to the analysis of our algorithm FSR-U, as $q_{\mathcal{P}}(\mathcal{D})$ is an average of m independent random variables, while $\bar{q}_{\mathcal{P}}(\mathcal{D})$ is not (since it depends on $\mu(\mathcal{D})$, that is estimated from the observations in the whole dataset).

Recall the definition of the set $\mathcal{L}^\star$ of non-significant patterns: $\mathcal{L}^\star = \{\mathcal{P} \in \mathcal{L} : q_{\mathcal{P}} = 0\}$. To output significant patterns while controlling the FWER below δ, our goal is to bound the *supremum deviation* $\sup_{\mathcal{P} \in \mathcal{L}^\star} \{\bar{q}_{\mathcal{P}}(\mathcal{D})\}$ (Eq. (2)) below some value η with probability at least $1 - \delta$, for some $\delta \in (0, 1)$, and provide in output all patterns with $\bar{q}_{\mathcal{P}}(\mathcal{D}) \geq \eta$. To bound (2), we study the surrogate quantity $\sup_{\mathcal{P} \in \mathcal{L}^\star} \{q_{\mathcal{P}}(\mathcal{D})\}$, as Theorem 10 guarantees a small bound on $|q_{\mathcal{P}}(\mathcal{D}) - \bar{q}_{\mathcal{P}}(\mathcal{D})|$. Consider the collection $\mathcal{R}^\star = \{\mathcal{D}_1^\star, \ldots, \mathcal{D}_c^\star\}$ of $c \geq 1$ i.i.d. resampled datasets computed by FSR (line 4). We prove the following result.

THEOREM 11. *Let $\mathcal{D}$ be a dataset of m samples taken i.i.d. from a distribution γ, and $\mathcal{R}^\star$ a collection of $c \geq 1$ i.i.d. resamples of the target labels of $\mathcal{D}$. For any $\delta \in (0, 1)$, define $v_T \geq \mu(1 - \mu)$, $v \geq v_T \sup_{\mathcal{P} \in \mathcal{L}^\star} \{\mathbb{E}_{\mathcal{D}}\left[f_{\mathcal{P}}(\mathcal{D})\right]\}$, and ε as*

$$\tilde{d}(\mathcal{R}^\star, \check{\mu}) = \frac{1}{c} \sum_{j=1}^{c} \sup_{\mathcal{P} \in \mathcal{L}} \left\{\bar{q}_{\mathcal{P}}(\mathcal{D}_j^\star, \check{\mu})\right\}$$

$$\hat{r} = \tilde{d}(\mathcal{R}^\star, \check{\mu}) + \sqrt{\frac{\ln\left(\frac{4}{\delta}\right)}{2cm}}$$

$$\hat{d} = \hat{r} + \sqrt{\left(\frac{2v_T \ln\left(\frac{4}{\delta}\right)}{m}\right)^2 + \frac{2\hat{r} \ln\left(\frac{4}{\delta}\right)}{m}} + \frac{2v_T \ln\left(\frac{4}{\delta}\right)}{m}$$

$$\varepsilon \doteq \hat{d} + \sqrt{\frac{2 \ln\left(\frac{4}{\delta}\right)\left(v + 2\hat{d}\right)}{m}} + \frac{\ln\left(\frac{4}{\delta}\right)}{3m}. \tag{5}$$

With probability at least $1 - \delta$ over the choice of $\mathcal{D}$ and $\mathcal{R}^\star$ it holds $\sup_{\mathcal{P} \in \mathcal{L}^\star} \{q_{\mathcal{P}}(\mathcal{D})\} \leq \varepsilon$.

We use the following simple upper bound for both v_T and v: $v_T = v = \sup_{|x-\mu(\mathcal{D})| \leq \varepsilon_T} x(1 - x)$, and note that while slightly more refined bounds are possible, we omit them to improve readability.

Note that Theorem 11 does not require the knowledge of the unknown parameter μ, but only of an upper bound $\hat{\mu}$ and of a lower bound $\check{\mu}$. Theorem 11 allows to implement the procedure boundStatistic for the unconditional setting: boundStatistic computes ε as in Eq. (5). Analogously to the conditional case, evaluating ε requires to compute $\sup_{\mathcal{P} \in \mathcal{L}} \left\{\bar{q}_{\mathcal{P}}(\mathcal{D}_j^\star, \check{\mu})\right\}$, which is costly, but only needs to be performed on c resampled datasets, and c is in practice small (as we show in Section 5).

By combining the results in this section, we prove the guarantees provided by FSR for the task of finding significant patterns when unconditional testing is used. In particular, the following Corollary proves that, when boundStatistic returns the value ε defined in Eq. 5, then the output set O of significant patterns returned by FSR-U has FWER bounded by the user-defined parameter δ.

COROLLARY 12. *Fix $\delta \in (0, 1)$ and $c \geq 1$. Let O be the output of FSR with input parameters $\mathcal{L}$, $\mathcal{D}$, c, δ, and let ε as in Eq. (5). Then, the set O has FWER $\leq \delta$ under the unconditional null distribution.*

Power analysis. The following result provides guarantees on the *power* of FSR-U, the version of FSR that uses unconditional testing, building on the probabilistic upper bound to $\tilde{d}(\mathcal{R}^\star, \check{\mu})$ proved in Appendix A.5, and on concentration bounds based on the pseudodimension of the language of subgroups.

THEOREM 13. *Fix $\delta \in (0, 1)$ and $c, z \geq 1$. Let O be the output of FSR with input parameters $\mathcal{L}$, $\mathcal{D}$, c, and δ, where $\mathcal{L}$ is the language of subgroups composed by conjunctions with at most z conditions over d continuous features. Define $\hat{\omega} = \hat{\mu}(1 - \check{\mu}) \sup_{\mathcal{P} \in \mathcal{L}} f_{\mathcal{P}}(\mathcal{D})$, and let ε be defined as in Theorem 11, where $\hat{r}$ is replaced by*

$$\hat{r} = \sqrt{\frac{2\hat{\omega} z \ln\left(\frac{e^3 m^2 d}{4z^3}\right)}{m}} + \sqrt{\frac{\ln\left(\frac{3}{\delta}\right)}{2cm}} + \frac{z \ln\left(\frac{e^3 m^2 d}{4z^3}\right)}{3m} + \sqrt{\frac{\ln\left(\frac{4}{\delta}\right)}{2cm}}.$$

Then, define $f_{\mathcal{P}} = \mathbb{E}_{\mathcal{D}}[f_{\mathcal{P}}(\mathcal{D})]$. With probability at least $1 - \delta$, O contains all patterns $\mathcal{P}$ with quality $q_{\mathcal{P}}$ satisfying

$$q_{\mathcal{P}} \geq \varepsilon + 2\varepsilon_T \left(f_{\mathcal{P}} + \sqrt{\frac{z \ln\left(\frac{2ed}{z}\right) + \ln\left(\frac{3}{\delta}\right)}{2m}}\right) + \sqrt{\frac{z \ln\left(\frac{2ed}{z}\right) + \ln\left(\frac{3}{\delta}\right)}{2m}}.$$

5 EXPERIMENTS

This section presents the results of our experiments. The goal of our experimental evaluation is to assess FSR's capabilities of discovering significant patterns with high statistical power, analyzing efficiently large real-world datasets with few-shot resampling, in both conditional (FSR-C) and unconditional (FSR-U) settings.

Pattern Language. In our experimental evaluation we focus on the problem of discovering significant *subgroups* from large real-world datasets with mixed feature types (both categorical and continuous). The language $\mathcal{L}$ is composed of conjunctions of up to z conditions on the features of the data [4], where z is a fixed parameter (see below). These conditions are either equalities (for categorical features), inequalities, or intervals (on continuous features).

Datasets. We tested FSR on 12 standard benchmarks and real-world datasets to evaluate subgroup discovery algorithms from UCI[2]. The statistics of the datasets are described in Table 1. These datasets cover a wide range of sizes, dimensionalities, and application domains. The column z of Table 1 reports the maximum number of conjunction terms for the subgroups in the language $\mathcal{L}$ for each dataset.

Implementation of FSR. We implemented FSR in Python. The code and the scripts to reproduce all experiments are available online[3]. To mine subgroups, we make use of a fast depth-first enumeration algorithm included in the library pysubgroup[4] [23].

Baselines. Since our algorithm FSR is the first algorithm that can be used for mining significant patterns with both conditional and unconditional testing, we consider different baselines for conditional testing and for unconditional testing.

For conditional testing, we compare FSR-C with a variant of TopKWY [32], the state-of-the-art method for significant pattern mining with conditional testing, based on the WY permutation testing procedure [46]. The original implementation of TopKWY is only tailored to identify significant itemsets and subgraphs, and does not support subgroups from categorical and continuous features; however, we note that it is fairly simple to adapt its strategy to such case. In particular, we extended TopKWY to identify significant subgroups by estimating the distribution of the supremum deviation using permuted datasets (instead of the p-values as done in the original TopKWY implementation [32]). That is, our variant of TopKWY considers the same statistic of FSR-C (i.e., the supremum deviation), but instead of generating resamples and taking the average of supremum deviations as done by FSR-C, it efficiently computes its δ-quantiles considering permutations of the labels. In addition, in our variant of TopKWY we use pysubgroup to mine subgroups. Note that since our variant of TopKWY and FSR-C share the same, equally optimized, procedure to explore the search space of the language $\mathcal{L}$, all comparisons in terms of running times are fair. For TopKWY we use 10^3 permutations, a good trade-off in terms of running time and accuracy for estimating the δ-quantile for typical values of δ (e.g., 0.05) [25].

Regarding unconditional testing, we note that FSR-U is the first method to discover significant patterns in a (fully) unconditional setting. Therefore, a baseline that may seem reasonable to control the FWER is the standard Bonferroni correction: each pattern $\mathcal{P}$ is flagged as significant if the probability (under the null hypothesis) of observing a quality greater or equal than the one measured in the data is at most $\delta/|\mathcal{L}|$, where $|\mathcal{L}|$ is the size of the language $\mathcal{L}$ (see also Section 1). Note, however, that this simple method is not useful for subgroups as $|\mathcal{L}|$ is infinite (e.g., as the number of inequalities over a continuous feature is unbounded)[5]. Therefore, we compare FSR-U with a novel non-trivial baseline, that we call FSR-U-UB, that resolves this issue. We describe FSR-U-UB at high level and defer additional details to Appendix A.6. For FSR-U-UB, we use part of the concentration bounds developed for FSR-U to

---

[2]https://archive.ics.uci.edu/

[3]https://github.com/VandinLab/FSR

[4]https://github.com/flemmerich/pysubgroup

[5]We remark that, while there exist other correction procedures more powerful than Bonferroni (e.g., the Holm procedure), they all require to fix a finite set of hypothesis, therefore do not apply to our setting.

Table 1: Statistics of the datasets considered in our experiments. m is the number of transactions, d is the number of features (categorical/continuous), $\mu(\mathcal{D})$ is the fraction of transactions with target equal to 1, z is the maximum number of conjunction terms in the language $\mathcal{L}$.

$\mathcal{D}$	m	d	$\mu(\mathcal{D})$	z
abalone	4177	1/7	0.663	5
adult	32561	8/6	0.241	5
bank	41188	10/10	0.113	3
brain-cancer	862	22/1	0.421	5
cancer-rna-seq	801	0/20531	0.375	2
covtype	581012	0/54	0.365	3
gisette	7000	0/5000	0.500	2
HIGGS	11000000	0/28	0.529	3
kdd-cup	95370	73/405	0.050	2
mushroom	8124	22/0	0.482	5
SUSY	5000000	0/18	0.457	3
theorem-prover	3059	0/51	0.420	3

upper bound the supremum deviation in terms of its expectation (taken w.r.t. the resamples but conditionally on the transactions, see Theorem 11 and Appendix A.6, but instead of estimating the expected supremum deviation with c resamples (as done by FSR-U), we compute an upper bound to it via an union bound over the (finite) number of distinct subgroups that are *observed* in the data in at least one transaction, i.e., we consider the finite *projection* of $\mathcal{L}$ on $\mathcal{D}$, instead of *all* $|\mathcal{L}|$ possible subgroups. Note that such approach bounds the FWER while being much less conservative than Bonferroni, since it corrects for the size of the projection instead of the size of $\mathcal{L}$. By comparing FSR-U to FSR-U-UB we directly evaluate the advantage of computing bounds from resampled datasets that consider dependencies among patterns, as done by FSR-U.

Experimental setup. All the experiments were run on a machine with 2.30 GHz Intel Xeon CPU, 512 GB of RAM, on Ubuntu 20.04. In all experiments we use $\delta = 0.05$ (i.e., we control the FWER below 0.05). We repeated all experiments 10 times, and report averages ± stds over the 10 repetitions.

Impact of parameters on FSR. In the first set of experiments we evaluate the effect of the number of resamples c on the deviation bound ε computed by FSR and its running time. We consider both FSR-C and FSR-U, respectively designed to compute significant patterns with conditional and unconditional testing. To ease the presentation, for this first experiment we focus on 3 of the datasets we considered; the results for the other datasets are very similar. In Figure 1-(a) we show the deviation bounds computed by FSR-C for different values of c, while Figure 1-(b) is analogous for FSR-U. From these plots we clearly conclude that using $c = 10$ resamples is sufficient to obtain a small deviation bound, and that using more than 10 resamples is marginally beneficial as all curves flatten. Remarkably, for all datasets (and in particular for the larger dataset adult), and for both methods, even using *one* resample is enough to compute a meaningful deviation bound. This is in striking contrast with state-of-the-art methods based on permutation testing, that instead require a number of 10^3-10^4 permutations of the target
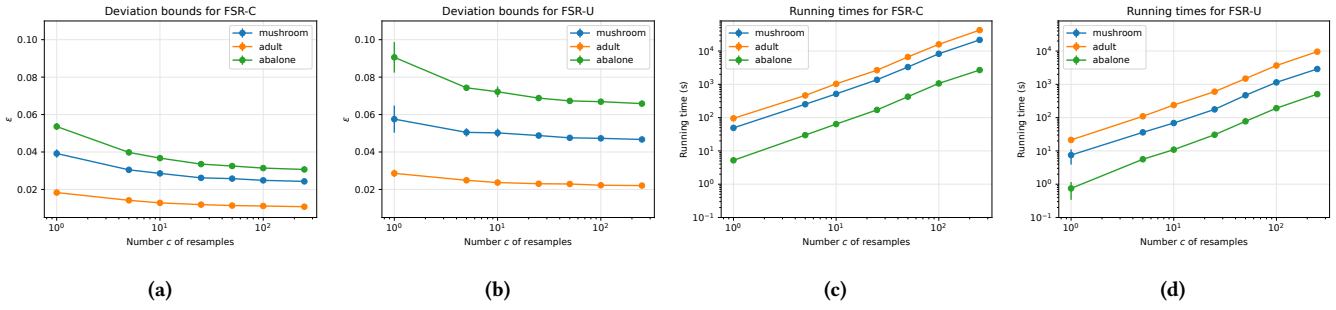
Figure 1: Effect of the number of resamples c on the deviation bounds (a)-(b) and running times (c)-(d) for FSR algorithms.
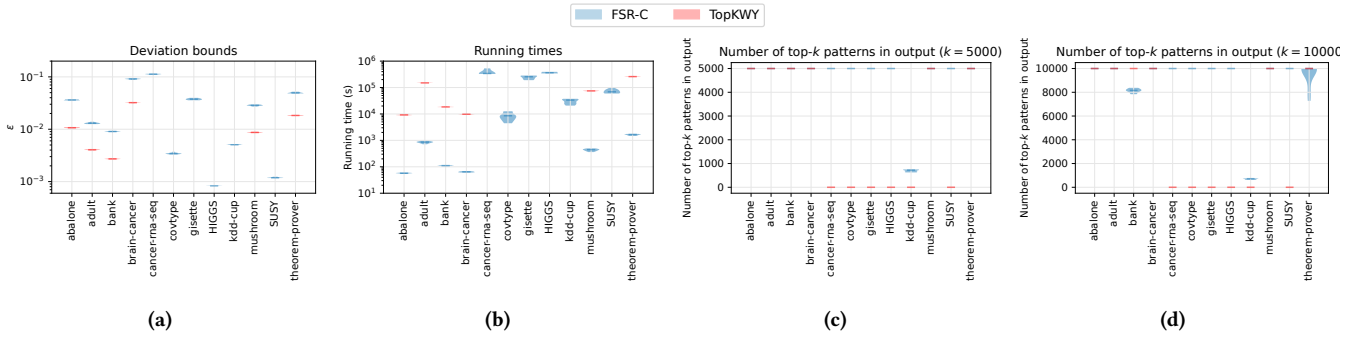

Figure 2: Comparison of FSR-C with TopKWY in terms of deviation bounds (a), running times (b), and number of results (c)-(d).
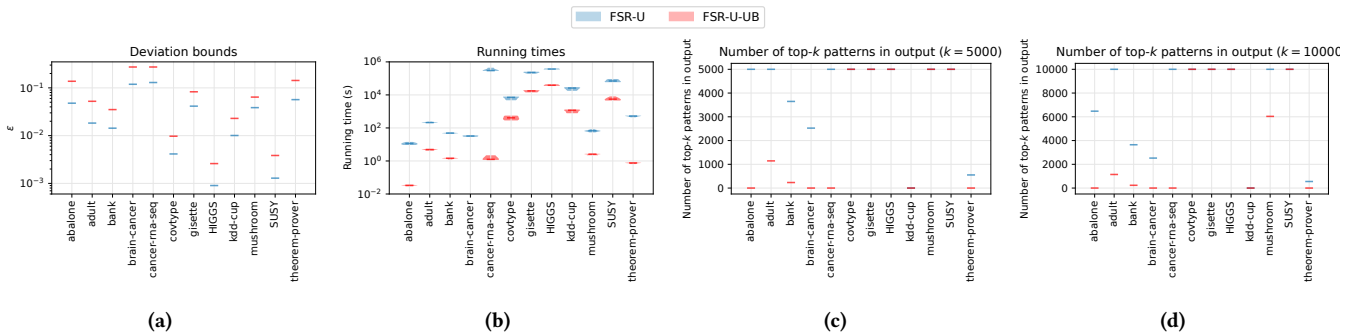

Figure 3: Comparison of FSR-U with the baseline FSR-U-UB (Appendix A.6) in terms of deviation bounds (a), running times (b), and number of results (c)-(d).

labels to properly estimate the δ-quantile. Furthermore, we observe that the deviation bounds computed by FSR-C, in the conditional setting, are smaller than the bounds computed by FSR-U, for the unconditional scenario; this confirms the fact that the assumptions made on the process generating the data have a sensible effect on this aspect, a consequence of properly taking into account the *uncertainty* of the collected data.

Figures 1-(c)-(d) show the running time of the two methods FSR-C and FSR-U as functions of c. Not surprisingly, we clearly observe that the running time increases linearly with the number of resamples c for both methods. Therefore, we expect that processing a small number of resampled datasets is extremely advantageous

in terms of running time. We also observe that FSR-U is faster than FSR-C for all values of c. This is due to the fact that FSR-C, by computing a smaller deviation bound, explores a wider portion of the pattern language $\mathcal{L}$. In any case, using at most 10 resamples is always feasible (since both algorithms terminate after at most 17 minutes for these datasets). From these observation, we fix c to 10 for FSR-C and FSR-U in all our experiments.

Evaluation of FSR-C. In this experiment we report the performance of FSR-C to identify significant patterns with conditional testing, comparing it with a variant of TopKWY, the state-of-the-art method to identify significant patterns with permutation testing. In Figure 2 we show the deviation bounds (a), the running times

(b), and the number of reported results (c)-(d) for both algorithms. From Figure 2-(a), we observe that the deviation bounds computed by FSR-C are larger than the ones computed by TopKWY. This is not surprising, as the deviation bound returned by TopKWY, that is based on the WY procedure, is a very accurate estimate of the optimal corrected threshold to bound the FWER (since the WY estimator converges asymptotically to it [27]); on the other hand, FSR-C computes an *upper bound* to such quantity with stronger probabilistic guarantees (i.e., the bound does not only converge *asymptotically* and *in expectation*, but rather holds in finite samples with high probability). While the deviation bounds computed by TopKWY are smaller than FSR-C, from Figure 2-(b) we observe a significant gap in terms of running time between the two methods. In fact, TopKWY requires two orders of magnitude more time than FSR-C to compute its deviation bound; this is mainly due to the fact that TopKWY has to process two orders of magnitude more permutations of the target label than FSR-C. Note that using 10^4 permutations (instead of 10^3, to have a more accurate estimation of the δ-quantile, as often done in practice) in TopKWY would result in an even more substantial gap.

We now compare the two methods in terms of number of reported significant patterns. To do so, we follow a typical subgroup discovery analysis, based on the task of identifying the k most "interesting" subgroups in terms of quality: first, we mine the set of top-k subgroups with highest quality; then, we count the number of them that are flagged as significant by the two methods. We use $k \in \{5 \cdot 10^3, 10^4\}$. Figures 2-(c)-(d) show the number of patterns that are reported in output by both methods. We observe that, for $k = 5 \cdot 10^3$, for 6 of the 12 datasets we considered, both algorithms output all the top-k patterns, i.e., they have the same output; for such datasets TopKWY reports all top-k patterns also for $k = 10^4$, while FSR-C outputs the same set of results for all but two datasets, for which it reports more than 70% of them. For the remaining 6 datasets, which consist of the 4 largest datasets (in terms of the number m of transactions) and the 2 datasets with highest number of features, TopKWY could not complete in reasonable time (i.e., we stopped it after 10 days), while FSR-C finished the analysis while returning a large number of significant results (i.e., either all the top-k patterns or more than 600 of them for the kdd-cup dataset). For the most challenging datasets (HIGGS, with 11 millions transactions, and cancer-rna-seq, with $> 10^4$ continuous features), FSR-C concludes after 5 days, while TopKWY would require (approximately) 1.5 years of computation. Therefore, in such cases FSR-C *enables* the analysis of these challenging instances while identifying many significant patterns. These results confirm that FSR-C, while providing a slightly more conservative upper bound to the supremum deviation, is still capable of discovering the same (or almost the same) most significant patterns, while being more than two orders of magnitude faster, and reporting many patterns as significant for challenging instances out of reach for the state-of-the-art. We conclude that FSR-C provides an excellent trade-off between the number of patterns identified as significant and the computational requirement of the analysis.

Evaluation of FSR-U. We now evaluate the performance of FSR-U to discover significant patterns with unconditional testing. We show the results in Figure 3. We compare FSR-U with the baseline FSR-U-UB (described above) in terms of deviation bounds (a), running

time (b), and number of results (c)-(d). From Figure 3-(a) we clearly observe that FSR-U computes deviation bounds that are always smaller than FSR-U-UB. We conclude that processing the resamples of the target label provides much more accurate deviation bounds w.r.t. more standard techniques (i.e., a Bonferroni correction). This is a consequence of taking into account the dependencies among patterns when upper bounding the expected supremum deviation. In terms of running time, FSR-U always conclude in reasonable time (similarly to FSR-C), while FSR-U-UB is faster (since it does not consider any resamples). On the other hand, in many cases the baseline FSR-U-UB terminates quickly but without reporting anything in output: Figures 3-(c)-(d) show that for 5 datasets it does not report significant patterns, while for the other datasets it outputs a significantly smaller amount (e.g., for adult, FSR-U finds almost an order of magnitude more results than FSR-U-UB). This experiment shows that FSR-U is a practical and powerful method to discover significant patterns with unconditional testing, and that it significantly improves over more standard techniques such as Bonferroni correction.

Application to Neural Network interpretation. In this final experiment we evaluate a practical application of FSR to the task of Neural Network interpretation [14]. More precisely, we consider the MNIST dataset [22] and train a Convolutional Neural Network (CNN), with the goal of identifying correlations between the activation values of neurons with the predicted target. To do so, we evaluate the association of the activation values of neurons in a convolutional filter with a binary target, composed by drawings of digits composed by straight lines only (1 and 7), versus the other digits. While we defer most details of this experiment to Appendix A.7, we observed that FSR successfully identifies interpretable activation patterns of several neurons, while requiring orders of magnitude less time than previous methods (as discussed previously).

6 CONCLUSIONS

We presented FSR, a novel algorithm to identify statistically significant patterns with rigorous bounds on the FWER. FSR uses a few-shot resampling strategy, which leads to an efficient and practical approach that can be used for both conditional and unconditional testing. Our experimental evaluation shows that FSR is an effective and accurate method for significant patterns discovery, while significantly reducing the computational cost of state-of-the-art multiple comparisons procedures, such as permutation testing, that hardly scale the analysis to complex languages, such as subgroups, and large datasets. While the experiments presented in this work are focused on subgroups, we expect the relative improvements obtained by FSR to directly transfer to other pattern types, given the generality of our framework and of the design of our resampling procedures, which are shared by all types of patterns.

REFERENCES

[1] Charu C Aggarwal, Yao Li, Philip S Yu, and Ruoming Jin. 2010. On dense pattern mining in graph streams. *Proceedings of the VLDB Endowment* 3, 1-2 (2010), 975–984.

[2] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. 1993. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*. 207–216.

[3] Mohammad Al Hasan and Mohammed J Zaki. 2009. Output space sampling for graph patterns. *Proceedings of the VLDB Endowment* 2, 1 (2009), 730–741.

[4] Martin Atzmueller. 2015. Subgroup discovery. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 5, 1 (2015), 35–49.

[5] GA Barnard. 1945. A new test for 2× 2 tables. *Nature* 156, 3954 (1945).

[6] Yoav Benjamini and Yosef Hochberg. 1995. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)* 57, 1 (1995), 289–300.

[7] Carlo Bonferroni. 1936. Teoria statistica delle classi e calcolo delle probabilita. *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commericiali di Firenze* 8 (1936), 3–62.

[8] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. 2013. *Concentration inequalities: A nonasymptotic theory of independence.* Oxford university press.

[9] Lei Cao, Yizhou Yan, Samuel Madden, Elke A Rundensteiner, and Mathan Gopalsamy. 2019. Efficient discovery of sequence outlier patterns. *Proceedings of the VLDB Endowment* 12, 8 (2019), 920–932.

[10] Matteo Ceccarello and Johann Gamper. 2022. Fast and Scalable Mining of Time Series Motifs with Probabilistic Guarantees. *Proceedings of the VLDB Endowment* 15, 13 (2022), 3841–3853.

[11] Chen Chen, Cindy X Lin, Matt Fredrikson, Mihai Christodorescu, Xifeng Yan, and Jiawei Han. 2009. Mining graph patterns efficiently via randomized summaries. *Proceedings of the VLDB Endowment* 2, 1 (2009), 742–753.

[12] Sebastian Dalleiger and Jilles Vreeken. 2022. Discovering significant patterns under sequential false discovery control. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.*

[13] Devdatt P Dubhashi and Alessandro Panconesi. 2009. *Concentration of measure for the analysis of randomized algorithms.* Cambridge University Press.

[14] Jonas Fischer, Anna Olah, and Jilles Vreeken. 2021. What's in the Box? Exploring the Inner Life of Neural Networks with Robust Rules. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Vol. 139. PMLR, 3352–3362.

[15] Ronald A Fisher. 1922. On the interpretation of χ 2 from contingency tables, and the calculation of P. *Journal of the royal statistical society* 85, 1 (1922).

[16] Wilhelmiina Hämäläinen and Geoffrey I Webb. 2019. A tutorial on statistically sound pattern discovery. *Data Mining and Knowledge Discovery* 33 (2019), 325–377.

[17] Jiawei Han, Hong Cheng, Dong Xin, and Xifeng Yan. 2007. Frequent pattern mining: current status and future directions. *Data mining and knowledge discovery* 15, 1 (2007), 55–86.

[18] Nguyen Thi Thao Ho, Torben Bach Pedersen, et al. 2022. Efficient temporal pattern mining in big time series using mutual information. *Proceedings of the VLDB Endowment* 15, 3 (2022), 673–685.

[19] Sture Holm. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics* (1979), 65–70.

[20] Kumar Jogdeo and Stephen M Samuels. 1968. Monotone convergence of binomial probabilities and a generalization of Ramanujan's equation. *The Annals of Mathematical Statistics* 39, 4 (1968), 1191–1195.

[21] Janis Kalofolias, Mario Boley, and Jilles Vreeken. 2017. Efficiently discovering locally exceptional yet globally representative subgroups. In *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE.

[22] Yann LeCun and Corinna Cortes. 2010. MNIST handwritten digit database. http://yann.lecun.com/exdb/mnist/

[23] Florian Lemmerich and Martin Becker. 2019. pysubgroup: Easy-to-use subgroup discovery in python. In *ECML PKDD 2018*. Springer, 658–662.

[24] Yi Li, Philip M Long, and Aravind Srinivasan. 2001. Improved bounds on the sample complexity of learning. *J. Comput. System Sci.* 62, 3 (2001), 516–527.

[25] Felipe Llinares-López, Mahito Sugiyama, Laetitia Papaxanthos, and Karsten Borgwardt. 2015. Fast and memory-efficient significant pattern mining via permutation testing. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 725–734.

[26] Colin McDiarmid. 1989. On the method of bounded differences. *Surveys in combinatorics* 141, 1 (1989), 148–188.

[27] Nicolai Meinshausen, Marloes H Maathuis, and Peter Bühlmann. 2011. Asymptotic optimality of the Westfall-Young permutation procedure for multiple testing under dependence. *The Annals of Statistics* (2011), 3369–3391.

[28] Shin-ichi Minato, Takeaki Uno, Koji Tsuda, Aika Terada, and Jun Sese. 2014. A fast method of statistical assessment for combinatorial hypotheses based on frequent itemset enumeration. In *ECML PKDD 2014*. Springer.

[29] Leonardo Pellegrina, Cyrus Cousins, Fabio Vandin, and Matteo Riondato. 2022. MCRapper: Monte-Carlo Rademacher averages for poset families and approximate pattern mining. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 16, 6 (2022), 1–29.

[30] Leonardo Pellegrina, Matteo Riondato, and Fabio Vandin. 2019. Hypothesis Testing and Statistically-sound Pattern Mining. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '19)*. ACM, New York, NY, USA, 3215–3216.

[31] Leonardo Pellegrina, Matteo Riondato, and Fabio Vandin. 2019. SPuManTE: Significant Pattern Mining with Unconditional Testing. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '19)*. ACM, New York, NY, USA, 1528–1538.

[32] Leonardo Pellegrina and Fabio Vandin. 2020. Efficient mining of the most significant patterns with permutation testing. *Data Mining and Knowledge Discovery* 34 (2020), 1201–1234.

[33] Leonardo Pellegrina and Fabio Vandin. 2024. Efficient Discovery of Significant Patterns with Few-Shot Resampling (Extended version). *arXiv preprint arXiv:2406.11803* (2024). http://arxiv.org/abs/2406.11803

[34] Andrea Pietracaprina and Fabio Vandin. 2007. Efficient incremental mining of top-K frequent closed itemsets. In *International Conference on Discovery Science*. Springer, 275–280.

[35] David Pollard. 2012. *Convergence of stochastic processes.* Springer Science & Business Media.

[36] Matteo Riondato and Fabio Vandin. 2020. MiSoSouP: Mining interesting subgroups with sampling and pseudodimension. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 14, 5 (2020), 1–31.

[37] Diego Santoro, Andrea Tonon, and Fabio Vandin. 2020. Mining Sequential Patterns with VC-Dimension and Rademacher Complexity. *Algorithms* 13, 5 (2020), 123.

[38] Shai Shalev-Shwartz and Shai Ben-David. 2014. *Understanding Machine Learning: From Theory to Algorithms.* Cambridge University Press.

[39] Aika Terada, Hanyoung Kim, and Jun Sese. 2015. High-speed Westfall-Young permutation procedure for genome-wide association studies. In *Proceedings of the 6th ACM Conference on Bioinformatics, Computational Biology and Health Informatics*. 17–26.

[40] Aika Terada, Mariko Okada-Hatakeyama, Koji Tsuda, and Jun Sese. 2013. Statistical significance of combinatorial regulations. *Proceedings of the National Academy of Sciences* 110, 32 (2013), 12996–13001.

[41] Matthijs Van Leeuwen and Arno Knobbe. 2012. Diverse subgroup set discovery. *Data Mining and Knowledge Discovery* 25 (2012), 208–242.

[42] Kenneth Ted Wallenius. 1963. *Biased sampling: the noncentral hypergeometric probability distribution.* Technical Report. https://purl.stanford.edu/wh056vj9347

[43] Geoffrey I Webb. 2006. Discovering significant rules. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 434–443.

[44] Geoffrey I Webb. 2007. Discovering significant patterns. *Machine learning* 68 (2007), 1–33.

[45] Geoffrey I Webb. 2008. Layered critical values: a powerful direct-adjustment approach to discovering significant patterns. *Machine Learning* 71 (2008), 307–323.

[46] Peter H Westfall and S Stanley Young. 1993. *Resampling-based multiple testing: Examples and methods for p-value adjustment.* John Wiley & Sons.