Towards Stable Significant Subgroup Discovery*

Jyoti[1][0000−0003−1595−3173], Aleksey Buzmakov[2][0000−0002−9317−8785], and Sriram Kailasam[3][0000−0002−2218−8660]

[1] Indian Institute of Technology Mandi, India
jjangra2@gmail.com
[2] National Research University Higher School of Economics, Perm, Russia
AVBuzmakov@hse.ru
[3] Indian Institute of Technology Mandi, India
sriramk@iitmandi.ac.in

Abstract. Discovering subgroups with significant association with binary class labels has wide applications in drug discovery, market basket analysis, etc. The state-of-the-art technique, TopKWY, which mines the top-k significant subgroups does not scale to large datasets, especially, when the search space of concepts is very large. In this paper, we propose SD-SOFIA, an algorithm that mines stable significant subgroups rather than just significant subgroups. SD-Sofia is able to mine the same significant subgroup or subgroup with comparable quality to TopKWY by navigating only a reduced search space. We have verified the result in 19 real-world datasets. This insight gives us an opportunity to design efficient and scalable algorithm for finding statistically significant subgroup in large datasets. The quality of the pattern mined and the time taken by our algorithm is governed by the initial delta threshold value. From experiments, we show that when initial delta threshold value is set between 0.5 to 3 percent of the number of objects in the dataset, our algorithm generates a pattern with comparable quality as TopKWY.

Keywords: Stable significant pattern mining · Subgroup discovery · Delta stable · SD-SOFIA.

1 Introduction

Subgroup discovery has many applications in drug discovery, market basket analysis, and technical domains [1]. Unlike concept mining in Formal Concept Analysis (FCA) [6], where we find closed patterns from unlabeled datasets, subgroup discovery aims to mine patterns having a high association with a class label. A special subarea of subgroup discovery, called significant pattern mining, mines patterns which are considered significant by means of some statistical test [8].

The total number of statistically significant patterns in a dataset can be large. Thus, one may be interested in mining only the most significant patterns.

* Supported by SPARC, a Government of India Initiative under grant no. SPARC/2018-2019/P682/SL.

TopKWY [8] is the state-of-the-art algorithm that can directly mine the top-k significant patterns without exploring all the significant patterns. It automatically adjusts the significance threshold and the support threshold during concept exploration and thereby attains better pruning of the search space. Even with these optimizations, TopKWY can take significant amount of time for large datasets due to the large search space of closed patterns. In practical applications, where one is interested in mining just a few, most significant patterns, time is the primary concern. For example, in subgroup discovery mining, only one target concept of considerable quality is required; it is crucial to mine this concept as fast as possible.

In this paper, we propose SD-SOFIA, an algorithm that mines stable significant subgroups rather than just significant subgroups. It is derived from SOFIA [5, 3], an algorithm for mining patterns whose Δ-measure is greater than a specified threshold. SD-SOFIA uses value obtained from an optimistic estimator, in addition to the Δ-measure to expand the most promising concepts w.r.t. a subgroup discovery task. This leads to faster discovery of the most stable significant subgroup. We show that in standard datasets, SD-SOFIA can mine the most stable significant pattern with more or less the same quality as the most significant pattern found by TopKWY from the reduced search space of Δ-stable patterns.

2 Background and Related Work

Formal context is defined as triplet $\mathbb{K} = (G, M, I)$, where G is a set of objects, M is a set of attributes (or descriptions), and $I \subseteq G \times M$ is a binary relation between G and M [6]. A pattern (or itemset) B is a subset of the attribute set M and said to be closed if there exists no superset of B which is present in all the objects containing B. Support of a pattern B is defined as the number of objects containing B.

As the number of closed patterns is extremely large, pattern mining algorithms usually mine only patterns w.r.t. an interestingness measure. One of the most promising interestingness measure is stability but it is hard to compute [4]. So an estimate of stability; Δ-measure is proposed in [4]. Given a closed pattern $B \subseteq M$, Δ-measure is defined as $\mathit{\Delta}(B) = \min_{A \supset B}(\mathit{Supp}(B) - \mathit{Supp}(A))$. Here, Supp$(X)$ is a function that returns the support of the pattern X. This paper introduces a new algorithm by exploring the search space of Δ-stable patterns, i.e., closed patterns with Δ-measure greater than a specified threshold θ.

In subgroup discovery, a binary class label $c \in \{0, 1\}$ is associated with each object. Following the authors of TopKWY [8], we rely on Fisher exact test to evaluate the association between a concept and the class labels. Moreover, this test allows for efficient branch cutting during the search for the best concept w.r.t. its association to class labels. This test gives the p-value in the interval $[0, 1]$. The smaller the p-value is, the better the significance of the concept.

2.1 Related Work

In significant pattern mining just fixing the significance threshold to some value may lead to large number of false positives while testing multiple hypotheses. Llinares et al. [7] addressed this issue by first finding the appropriate significance threshold and then discovering significant patterns w.r.t. this threshold. Although, it works correctly for multiple hypothesis testing, the search space of significant patterns can still be huge. Pellegrina et al. [8] in their algorithm TopKWY performs automatic adjustment of the upper bound on the support of significant patterns. Thus, it prunes the insignificant (or untestable) patterns faster. However, it has been observed that in large datasets, this upper bound gets adjusted to a very small value. Thereby, TopKWY takes huge amount of time to mine the most significant pattern. In this paper, we present an algorithm SD-SOFIA, which is based on SOFIA algorithm [5, 3] to mine the most Δ-stable significant pattern. As our algorithm considers only Δ-stable concepts, it can potentially give rise to results having lower quality compared to TopKWY. However, from experiments on standard datasets we observe that SD-SOFIA gives patterns with comparable quality from the reduced search space of Δ-stable patterns.

3 Proposed Algorithm

Let us first fix some order on attributes M. For the sake of simplicity in this paper, by projection i we mean the dataset limited to having only first i attributes. Let M_i be the first i attributes from M.

The recent algorithm SOFIA [3] is based on Δ-stable concepts in projection i and the new attribute $i+1$ computes Δ-stable context in the projection $i+1$. It should be noticed, that this strategy is limited for subgroup discovery, since it finds preimages of all concepts from projection i to projection $i+1$ simultaneously. However, the most commonly used strategy in subgroup discovery is expansion of the most promising concept [2]. This allows for earlier finding of concepts with high quality Q improving the efficiency of branch cutting.

In algorithm SOFIA, finding preimages of a concept does not depend on other concepts and, thus, concepts that are stored in $\mathcal{P}$ can correspond to different projections. Thus, it is not necessary to move all concepts from projection i to projection $i+1$, i.e., only the most promising concept can be moved to the next (w.r.t. this concept) projection. This procedure is shown in Algorithm 1. All concepts are stored in the queue. The queue can contain concepts from different projections, thus, a concept is denoted as $(A, B \mid i)$, where A and B are the extent and the intent of the concept correspondingly, and i is the projection it is computed in. The corresponding elements of a concept $c = (A, B \mid i)$ can be extracted by means of functions $\mathrm{Ext}(c)$, $\mathrm{Int}(c)$, and $\mathrm{Proj}(c)$ for the extent, the intent, and the projection number of c correspondingly.

In line 2, Algorithm SD-SOFIA initializes the queue with the only available concept in projection 0. This concept is $(G, \emptyset)$. Indeed, since M_0 contains no

Algorithm 1: Algorithm SD-SOFIA identifying the Δ-measure threshold θ and the corresponding best concept w.r.t. a quality function Q.

```
1  Function FindBestConcept()
2      queue.Push (($G, \perp \mid 0$));                /* Projection number 0 */
3      while not queue.isEmpty() do
4          $c \leftarrow$ queue.PopTheMostPromising ();
5          $\{c_i\} \leftarrow$ Preimages($c$);
6          foreach $cc \in \{c_i\}$ do
7              if Proj($cc$) = $|M|$ then
8                  best.Register($cc$);
9                  next;
10             if not $\Delta_{\mathrm{Proj}(cc)}(cc) < \theta$ then
11                 next;
12             if not best.IsPromising($cc$) then
13                 next;
14             queue.Push($cc$);
```

attribute, the only available intent is $\emptyset$. The algorithm iterates while queue is not empty. The concepts are extracted one by one (line 4) w.r.t. their potential, i.e., the value of the optimistic estimate $\overline{Q}$ of the quality function Q. Then in line 5 the preimages of the most potential concept $c = (A, B \mid i)$ are computed. Since projection ψ_{i+1} preserves one more attribute than projection ψ_i, there are only two possible preimages: $c_1 = (A, B^{\downarrow\uparrow} \mid i+1)$ and $c_2 = ((B \cup \{i\})^{\downarrow}, (B \cup \{i\})^{\downarrow\uparrow} \mid i+1)$. The preimages c_1 and c_2 can coincide.

Then every preimage cc of the concept c is processed in lines 7–14. First in lines 7–9 it is verified that cc is already in the last projection $\psi_{|M|}$. Only in this case the final Δ-measure value for this concept is known. Thus, only in this moment it is possible to decide if this concept can be reported as the best concept. Then in lines 10-13 the concept cc is checked if it can produce stable concepts and if it can produce a better concept than the already found one. Here, in line 10 the Δ-measure is indexed with the projection number, since the Δ-measure even for the same concept in different projection can change.

The correctness of the algorithm follows from the fact that different concepts can be stored in different projections but every concept is passing exactly the same strategy as in algorithm SOFIA.

4 Experimental Evaluation

We compare the performance of SD-SOFIA with TopKWY [8] in terms of pattern quality and total execution time on standard datasets available from FIMI[4],

[4] http://fimi.ua.ac.be

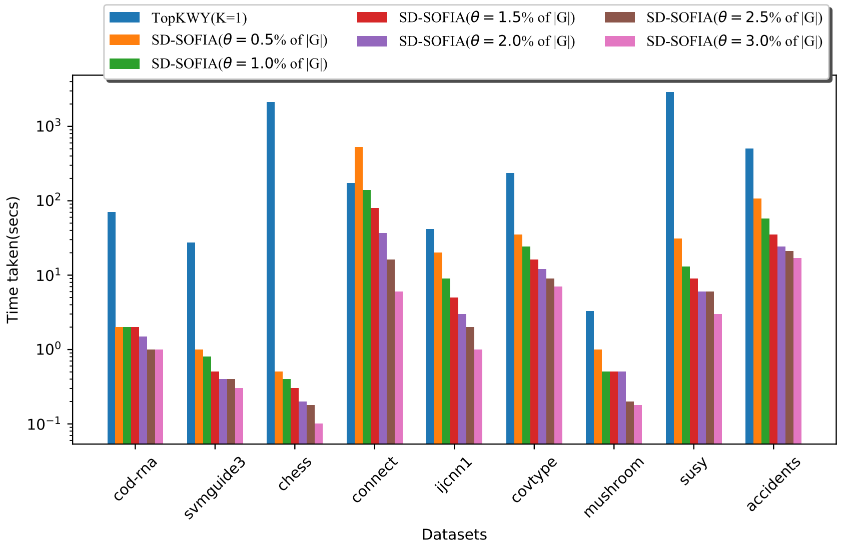

Fig. 1: The execution time of TopKWY and SD-SOFIA (without permutation testing) with different values of Δ-stability threshold (θ).

UCI[5], SPMF[6], and libSVM[7]. We did experiment on all 19 datasets used in TopKWY [8] with the number of attributes varying from 16 to 330285 and the number of objects varying from 1243 to 5000000. Same as TopKWY, SD-SOFIA is implemented in C++. We performed the experiments on a machine having 10-cores, 2.30GHz Intel(R) Xeon CPU, 128GB RAM; the OS is Ubuntu 16.04.6 LTS.

To mine the most significant pattern using TopKWY[8], we set the value of K to one while retaining default values for the other parameters. For SD-SOFIA, we vary Δ-stability threshold, θ, from 0.5% to 3%. The performance of SD-SOFIA depends on the Δ-stability threshold θ. Below 0.5%, SD-SOFIA ends up exploring the entire space of closed patterns and takes too long to complete. Above 3%, the quality of the resulting pattern in SD-SOFIA is quite poor.

Fig. 1 compares the execution time of TopKWY and SD-SOFIA (without permutation testing) on different datasets. x-axis shows the datasets and y-axis shows the execution time of the algorithms using logarithmic scale. As we have many datasets, we show the results only for 9 datasets. From Fig. 1, the execution time for SD-SOFIA decreases with an increase in θ value. Similar results are obtained for the remaining datasets. As θ value increases, the search

[5] https://archive.ics.uci.edu/ml/index.php
[6] http://www.philippe-fournier-viger.com/spmf
[7] https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets
[8] https://github.com/VandinLab/TopKWY

space of stable patterns gets shrunk. Hence, the execution time goes down; but the quality of the resulting pattern obtained from SD-SOFIA also decreases.

When the threshold is 0.5, for 13 out of 19 datasets, SD-SOFIA exactly matched the quality obtained by TopKWY, i.e. the most stable significant pattern was the same as the most significant pattern. Basically, we have a trade-off between time and quality. As the Δ threshold increases, the running time reduces, but the quality of the found pattern by SD-SOFIA also decreases and vice-versa. By a similar procedure to TopKWY, we also verified that, if the results of SD-SOFIA and TopKWY are different, the result found by SD-SOFIA is always statistically significant. With naively implemented permutation testing, SD-SOFIA's (for $\theta = 3\%$ of $|G|$) total execution time is comparable to TopKWY in 13 datasets, an order of magnitude better than TopKWY in 2 datasets and slower by an order of magnitude in 4 datasets.

5 Conclusion and Future Work

We have shown that stable patterns are good candidates for statistically significant patterns. This insight is useful to scale to large datasets by navigating only a reduced search space. From the results, it seems that 0.5-3 % of $|G|$ is a good choice for the threshold θ. Our future work is to improve efficiency and verify the statistical power of SD-SOFIA in noisy datasets. Another interesting extension is to mine the top-k stable significant patterns. We would also adopt efficient permutation testing from TopKWY in SD-SOFIA.

References

1. Atzmueller, M.: Subgroup discovery. WIREs: Data Mining and Knowledge Discovery **5**(1), 35–49 (2015)
2. Boley, M., Grosskreutz, H.: Non-redundant Subgroup Discovery Using a Closure System. In: Machine Learning and Knowledge Discovery in Databases. pp. 179–194. Springer, Berlin, Heidelberg (2009)
3. Buzmakov, A., Kuznetsov, S.O., Napoli, A.: Efficient Mining of Subsample-Stable Graph Patterns. In: 2017 IEEE International Conference on Data Mining (ICDM). pp. 757–762. New Orlean, LA, USA (2017)
4. Buzmakov, A., Kuznetsov, S.O., Napoli, A.: Scalable estimates of concept stability. In: Glodeanu, C.V., Kaytoue, M., Sacarea, C. (eds.) Formal Concept Analysis. pp. 157–172. Springer International Publishing, Cham (2014)
5. Buzmakov, A., Kuznetsov, S.O., Napoli, A.: Fast Generation of Best Interval Patterns for Nonmonotonic Constraints. In: Machine Learning and Knowledge Discovery in Databases, LNCS, vol. 9285, pp. 157–172. Springer (2015)
6. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer, 1st edn. (1999)
7. Llinares-López, F., Sugiyama, M., Papaxanthos, L., Borgwardt, K.: Fast and memory-efficient significant pattern mining via permutation testing. In: 21st ACM SIGKDD. p. 725–734. New York, NY, USA (2015)
8. Pellegrina, L., Vandin, F.: Efficient mining of the most significant patterns with permutation testing. In: 24th ACM SIGKDD. p. 2070–2079. ACM, New York, NY, USA (2018)