

Significant Pattern Mining on Continuous Variables

Mahito Sugiyama

ISIR, Osaka University; JST PRESTO

mahito@ar.sanken.osaka-u.ac.jp

Karsten M. Borgwardt

D-BSSE, ETH Zürich

karsten.borgwardt@bsse.ethz.ch

Abstract

Significant pattern mining, the search for sets of binary features that are statistically significantly enriched in a class of objects, is of fundamental importance in a wide range of applications from economics to statistical genetics. Still, all existing approaches make the restrictive assumption that the features are *binary* and require a binarization of continuous data during preprocessing, which often leads to a loss of information.

Here, we solve the open problem of *significant pattern mining* on *continuous* variables. Our approach detects all patterns that are statistically significantly associated with a class of interest, while rigorously correcting for multiple testing. Key to this approach is the use of Spearman’s rank correlation coefficient to represent the frequency of a pattern. Our experiments demonstrate that our novel approach detects true patterns with higher precision and recall than competing methods that require a prior binarization of the data.

1 Introduction

The original goal of *pattern mining* (Agrawal et al., 1993; Agrawal and Srikant, 1994) is to efficiently enumerate frequent combinations of binary features, that is, *patterns*, from the massive amount of candidate patterns in a database (Aggarwal and Han, 2014). Various types of patterns have been analyzed in applications: *itemsets* (Han et al., 2000), combinations of binary variables, used in market basket analysis to find frequently copurchased items, *subgraphs* (Yan and Han, 2002), used in drug discovery to detect commonly occurring substructures in a set of molecules modeled as graphs, and *sequences* (Pei et al., 2001), employed, for instance, in DNA sequence analysis and customer behavior analysis.

As an extension of the traditional pattern mining problem, *significant (discriminative) pattern mining* is recently emerging, which tries to find patterns enriched in one class relative to another class. The objective is to mine patterns that are statistically significantly associated with class membership while correcting for multiple testing to ensure rigorous control of the *family-wise error rate (FWER)*, that is, the probability to detect one or more false positive patterns. Significant pattern mining has been actively studied and applied to various types of data, including itemsets (Terada et al., 2013a,b; Llinares-López et al., 2015b; Papaxanthos et al., 2016; Terada et al., 2016), association rules (Webb, 2007; Hämmäläinen, 2012), subgraphs (Sugiyama et al., 2015), and intervals (Llinares-López et al., 2015a), as the method can provide *p-values* that are indispensable in scientific fields such as biology and medicine.

Common to all significant pattern mining approaches is that they assume binary (or binarized) variables. How to directly perform significant pattern mining on *continuous* data is still an open problem. Several application domains would benefit from such a technique, for instance, microarray data analysis to find highly co-expressed gene combinations (Stuart et al., 2003), brain activity analysis to detect a specific brain region from

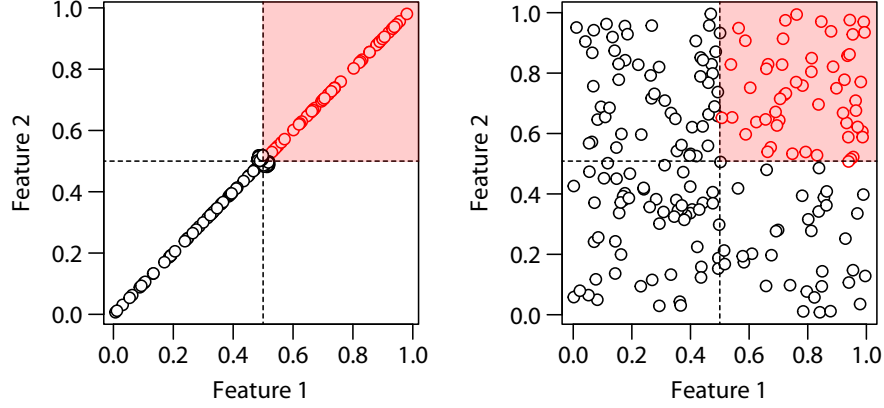


Figure 1: Although there is a clear correlation between Features 1 and 2 in the left panel and none in the right panel, after median-based binarization of Feature 1 and Feature 2, the frequency of the pattern (that is the number of points for which $\text{Feature 1} * \text{Feature 2} = 1$, or the number of points in the red box) will be exactly the same in both examples.

fMRI images (Davatzikos et al., 2005), and combinations of product features that affect marketing (Coussement et al., 2013). One can use significant pattern mining techniques (Terada et al., 2013a; Llinares-López et al., 2015b) with prior binarization of the data in these problems, but the common strategies for binarization cause at least one of the two following problems:

First, one common approach (Kaytoug et al., 2011b,a) is to record all N values of a feature and to then represent each feature value as a N -dimensional binary vector, whose entries indicate whether this feature value is less than each of the other feature values or not. As a result, a large sample size turns into a large number of binarized features, which significantly deteriorates the efficiency of pattern mining and generates highly redundant patterns. Second, even if we avoid the above efficiency problem by using the median of each feature as a binarization threshold per feature, this coarse binarization approach then cannot distinguish correlated from uncorrelated feature combinations (see Figure 1 for an example).

In this paper, we overcome these two drawbacks and enable significant pattern mining directly on continuous variables. More specifically, given a multivariate dataset with continuous values, the goal is to enumerate all significant feature combinations (patterns) without any binarization, whose multiple-testing-corrected p -values for association with the class label are below the significance level so that the FWER is controlled. Our approach is to use the *rank order statistics* to represent the frequency of a pattern, which is known as *copula support* (Tatti, 2013). We show that our frequency is easy to compute, well-defined in the sense of probability theory (Equation (4)), and can be interpreted as a multi-dimensional extension of Spearman’s pairwise rank correlation coefficient (Theorem 1), which means that we can solve the second problem described above and distinguish correlated and uncorrelated feature combinations (Figure 1). Moreover, we compute p -values by a *likelihood ratio test* using the frequencies of patterns, and show that *Tarone’s testability trick* (Tarone, 1990) can be used to prune untestable patterns that can never be significant, which enables us to solve the first problem and achieve efficient pattern mining. Our contribution allows to find all significant feature combinations and to compute their p -values directly on continuous data.

This paper is organized as follows: We define our problem and introduce the rank based frequency and analyze the connection to Spearman’s rank correlation in Section 2. Then we extend our approach to significant pattern mining and show how to correct for multiple testing, followed by the presentation of an enumeration algorithm in Section 3. After reviewing related work in Section 4, we experimentally validate

our method in Section 5 and summarize our findings in Section 6.

2 Frequent Pattern Mining

First we consider the problem of frequent pattern mining on continuous variables to introduce the frequency of a pattern, which is fundamental for the pattern mining problem and required to achieve significant pattern mining. Given a set $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ of N samples as a dataset, which corresponds to a set of transactions in the context of frequent itemset mining. Each sample $\mathbf{x}_i$ is an n -dimensional real vector $\mathbf{x}_i = (x_i^1, x_i^2, \dots, x_i^n) \in \mathbb{R}^n$. For each feature $j \in \{1, 2, \dots, n\}$, we denote by $\mathbf{x}^j$ the N -dimensional vector composed of the j th feature of the dataset D , that is, $\mathbf{x}^j = (x_1^j, x_2^j, \dots, x_N^j)$.

The goal of frequent pattern mining is to find frequently occurring *patterns*, each of which is an element J of the power set 2^V of the set of feature indices $V = \{1, 2, \dots, n\}$. More specifically, given a user specified threshold $\sigma \in \mathbb{R}$, enumerate the set of *frequent patterns* $\mathcal{F} \subseteq 2^V$ such that $\mathcal{F} = \{J \in 2^V \mid \eta(J) \geq \sigma\}$. In this setup, the *frequency* $\eta : 2^V \rightarrow [0, 1]$ should satisfy the *anti-monotonicity* with respect to the inclusion relationship: $\eta(J) \leq \eta(K)$ if $J \supseteq K$, which is essential to efficiently enumerate $\mathcal{F}$ based on the well-known Apriori principle (Agrawal and Srikant, 1994). The basic strategy is as follows: Start from the empty set $\emptyset$, enumerate frequent patterns according to the order of the inclusion relationship and prune all patterns $K \supseteq J$ if the condition $\eta(J) < \sigma$ is satisfied.

2.1 Rank Based Frequency

We show that the frequency $\eta(J)$ of the pattern J , which determines the importance of J , is naturally induced from a discrete probability distribution with the alphabet 2^V , which has not been analyzed until now.

Let us use a function $p : 2^V \rightarrow [0, 1]$ to represent a distribution on 2^V such that $\sum_{J \in 2^V} p(J) = 1$, which gives probability $p(J)$ to each pattern J . Then the frequency $\eta(J)$ is obtained as the sum of probabilities of patterns that include the pattern J , i.e.,

$$\eta(J) = \sum_{K \supseteq J} p(K).$$

For example, in itemset mining with binary variables $X^1, X^2, \dots, X^n$, the probability of each pattern $J \in 2^V$ is given as

$$p_{\text{binary}}(J) = \Pr \left(\prod_{j \in J} (X^j = 1) \prod_{j \in V \setminus J} (X^j = 0) \right),$$

resulting in $\sum_{J \in 2^V} p_{\text{binary}}(J) = 1$. The frequency is obtained from a given dataset $B = \{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_N\} \subseteq \{0, 1\}^n$ as

$$\begin{aligned} \eta_{\text{binary}}(J) &= \sum_{K \supseteq J} p_{\text{binary}}(K) = \Pr \left(\prod_{j \in J} (X^j = 1) \right) \\ &= \frac{1}{N} \sum_{i=1}^N \prod_{j \in J} z_i^j = \frac{|\{\mathbf{z} \in B \mid \Sigma(\mathbf{z}) \supseteq J\}|}{N}, \end{aligned} \quad (1)$$

where $\Sigma(\mathbf{z})$ is the set of indices for “1” of $\mathbf{z} \in \{0, 1\}^n$.

We apply this formulation to pattern mining from continuous data. To estimate the distribution p from a given continuous dataset D , we use the *rank product statistics*, which is a nonparametric statistical approach (Koziol, 2010; Heskes et al., 2014) and actively used in gene expression data analysis (Breitling et al., 2004; Eisinga et al., 2013). For each feature $j \in V$, let $r(x_i^j)$ be the *rank* of x_i^j with respect to components of the vector $\mathbf{x}^j = (x_1^j, x_2^j, \dots, x_N^j)$, that is, $r(x_i^j) = k$ if x_i^j is the k th smallest value among N values $x_1^j, x_2^j, \dots, x_N^j$, yielding the increasing sequence

$$r^{-1}(1) < r^{-1}(2) < \dots < r^{-1}(N).$$

We always assume that there is no tie in D , which is a natural assumption for continuous data. Assume that a mapping $\pi : \mathbb{R} \rightarrow [0, 1]$ gives a normalized ranking as

$$\pi(x_i^j) = \frac{r(x_i^j)}{N}. \quad (2)$$

Then we can obtain the probability distribution as

$$p(J) = \frac{1}{N} \sum_{i=1}^N \left(\prod_{j \in J} \pi(x_i^j) \prod_{j \in (V \setminus J)} (1 - \pi(x_i^j)) \right). \quad (3)$$

Nakahara et al. (Nakahara et al., 2003) showed that this satisfies $p(J) \in [0, 1]$ for all $J \in 2^V$ and $\sum_{J \in 2^V} p(J) = 1$. Although it is not feasible to compute $p(J)$ for every pattern $J \in 2^V$ to get the frequency $\eta(J)$ due to combinatorial explosion, interestingly, we can derive the following equation by a simple calculation:

$$\eta(J) = \sum_{K \supseteq J} p(K) = \frac{1}{N} \sum_{i=1}^N \prod_{j \in J} \pi(x_i^j). \quad (4)$$

This means that we can directly compute $\eta(J)$ without computing each $p(K)$ with $K \supseteq J$, whose time complexity is $O(N|J|)$, which is independent of n . Note the similarity between the definition in Equation (4) and that for binary variables in Equation (1), where the only difference is either each value is binary $z_i^j \in \{0, 1\}$ or the normalized ranking $\pi(x_i^j) \in [0, 1]$. This indicates that the formulation is a direct relaxation of pattern mining on binary variables.

Almost the same frequency was first introduced by Tatti (2013) as *copula support*, where $\pi(x_i^j)$ was defined as not in Equation (2) but $(r(x_i^j) - 1)/(N - 1)$. The difference comes from either the frequency is derived from the distribution over *patterns* in our case or that over *thresholds* in his case.

2.2 Interpretation

We firstly show that the frequency in Equation (4) can be viewed as the multidimensional extension of *Spearman's rank correlation coefficient* (Spearman, 1904), which is popularly used as a nonparametric method to measure the association between continuous variables. The Spearman's correlation coefficient between a pair features $j, j' \in \{1, 2, \dots, n\}$ is defined as

$$\rho(j, j') = 1 - \frac{6}{N(N^2 - 1)} \sum_{i=1}^N \left(r(x_i^j) - r(x_i^{j'}) \right)^2. \quad (5)$$

We show that our frequency $\eta(J)$ with $J = \{j, j'\}$ is essentially the same as $\rho(j, j')$.

Theorem 1. For any pattern $J = \{j, j'\} \in 2^V$ with $|J| = 2$,

$$\eta(J) = C \cdot \rho(j, j') + C', \quad \text{where } C = \frac{N^2 - 1}{3N^2} \text{ and } C' = \frac{N + 1}{2N^2}.$$

Proof. From the definition of Spearman's rank correlation in Equation (5), we have

$$1 - \rho = \frac{6}{N(N^2 - 1)} \sum_{i=1}^N \left(r(x_i^j) - r(x_i^{j'}) \right)^2,$$

$$\eta(J) = \frac{N^2 - 1}{3N^2}(\rho - 1) + \frac{1}{N} \sum_{i=1}^N \left(\frac{i}{N} \right)^2,$$

hence $C = (N^2 - 1)/3N^2$ follows. Since $\sum_{i=1}^N (i/N)^2 = (N + 1)(2N + 1)/6N$, it follows that

$$C' = \frac{(N + 1)(2N + 1)}{6N} - \frac{N^2 - 1}{3N^2} = \frac{N + 1}{2N^2}. \quad \square$$

Therefore, using $\eta(J)$ defined in Equation (4) in the frequent pattern mining formulation we can enumerate sets of correlated features as frequent patterns.

3 Significant pattern mining

Now we extend our formulation to significant pattern mining, where a given dataset is *supervised*, that is, for each sample $\mathbf{x}_i \in D$ a corresponding binary class label $y_i \in \{0, 1\}$ is additionally given. The goal is to find patterns whose frequencies are statistically significantly associated with the class partitioning¹.

Our contribution is to theoretically prove the applicability of *Tarone's testability trick* (Tarone, 1990) to our frequency of continuous values, which is indispensable to efficiently mine significant patterns (Llinares-López et al., 2015b; Terada et al., 2013a; Sugiyama et al., 2015) and has been applied to only statistical tests for discrete values to date, including a Fisher's exact test, a χ^2 test, and a Cochran–Mantel–Haenszel (CMH) test.

3.1 Statistical Testing

To test the significance of each pattern J , Fisher's exact test has been used as the standard statistical test in recent studies (e.g. (Llinares-López et al., 2015b; Terada et al., 2013a; Sugiyama et al., 2015)), which applies to only discrete data. Instead of Fisher's exact test, we use a *likelihood ratio test* (Fisher, 1922), which is a generalized χ^2 -test and often called G-test (Woolf, 1957) for testing the independence in a contingency table.

Suppose that N_l is the sample size in the class $l \in \{0, 1\}$ and let $r_l = \frac{N_l}{N}$. For each pattern $J \in 2^V$, let $\mathbf{p}_E$ be the expected distribution of its frequency under the null hypothesis that J is independent of class partitioning. The distribution $\mathbf{p}_E$ is represented as a four-dimensional vector:

$$\mathbf{p}_E = \left(r_0\eta(J), r_1\eta(J), r_0(1 - \eta(J)), r_1(1 - \eta(J)) \right),$$

¹This setting can be directly generalized to more than two classes when considering one-versus-rest classification.

Table 1: Contingency tables.

(a) Expected distribution.

	Occurrence of J	Non-occurrence of J	Total
$y = 0$	$r_0 \eta(J)$	$r_0(1 - \eta(J))$	r_0
$y = 1$	$r_1 \eta(J)$	$r_1(1 - \eta(J))$	r_1
Total	$\eta(J)$	$1 - \eta(J)$	1

(b) Observed distribution.

	Occurrence of J	Non-occurrence of J	Total
$y = 0$	$\eta_0(J)$	$r_0 - \eta_0(J)$	r_0
$y = 1$	$\eta_1(J)$	$r_1 - \eta_1(J)$	r_1
Total	$\eta(J)$	$1 - \eta(J)$	1

which corresponds to cells in the 2×2 contingency table shown in Table 1a and obtained from marginals $\eta(J)$, r_0 , and r_1 . Let $\eta_l(J)$ be the frequency in the class l , which is given as

$$\eta_l(J) = \frac{1}{N} \sum_{i=1}^N \delta(y_i, l) \prod_{j \in J} \pi(x_i^j),$$

where $\delta(x, y)$ is the Kronecker delta such that $\delta(x, y) = 1$ if $x = y$ and $\delta(x, y) = 0$ otherwise, and define the observed distribution as

$$\mathbf{p}_O = \left(\eta_0(J), \eta_1(J), r_0 - \eta_0(J), r_1 - \eta_1(J) \right),$$

which corresponds to the table in Table 1b. This probability distribution is always well-defined as $r_0 > \eta_0(J)$ and $r_1 > \eta_1(J)$ always hold by definition.

The difference between two distributions $\mathbf{p}_O$ and $\mathbf{p}_E$ can be measured by the Kullback–Leibler (KL) divergence, given as

$$D_{\text{KL}}(\mathbf{p}_O, \mathbf{p}_E) = \sum_i p_O^i \log \frac{p_O^i}{p_E^i},$$

and the null hypothesis H_0 and the alternative hypothesis H_1 are

$$H_0 : D_{\text{KL}}(\mathbf{p}_O, \mathbf{p}_E) = 0, \quad H_1 : D_{\text{KL}}(\mathbf{p}_O, \mathbf{p}_E) \neq 0.$$

This is known as G-test and the statistics is $\lambda = 2N D_{\text{KL}}(\mathbf{p}_O, \mathbf{p}_E)$, which follows the χ^2 -distribution with the degree of freedom 1. Therefore we can compute the p -value for each pattern $J \in 2^V$ as the upper-tail probability greater than λ .

3.2 Multiple testing correction

A pattern $J \in 2^V$ is (statistically) *significant* if its p -value is smaller than the predetermined threshold α , which means that the false positive rate is controlled under α (a typical setting is $\alpha = 0.05$ or 0.01). However,

if we test every pattern in 2^V with $|V| = N$, $\alpha \cdot 2^N$ false positive patterns could occur, which is often not acceptable as 2^N is massive.

To solve the problem, we need to correct the threshold α and control the *family-wise error rate* (FWER), which is the probability that at least one pattern is a false positive. A popular approach is *Bonferroni correction* (Bonferroni, 1936), where the corrected threshold δ_{Bon} is given as $\delta_{\text{Bon}} = \alpha/2^N$ and the resulting FWER is guaranteed to be smaller than α . However, it is well known that this method is too conservative, that is, the FWER is much smaller than α , which increases the false negative rate. A more serious problem is that we still need to compute the p -value for *all* patterns to check whether or not they are significant as the anti-monotonicity does not hold for the p -values, which is not feasible in most cases.

Here we use the *Tarone's testability trick* (Tarone, 1990), which have been successfully used in recent studies for significant pattern mining (Llinares-López et al., 2015a,b; Sugiyama et al., 2015; Terada et al., 2013a,b). Let $\psi(J)$ be the *minimum achievable p-value* of a pattern J under fixed marginals $\eta(J)$, r_0 , and r_1 . Suppose that $J_1, J_2, \dots$ is a sequence of patterns, which is sorted in increasing order according to $\psi(J_i)$ as

$$\psi(J_1) \leq \psi(J_2) \leq \psi(J_3) \leq \dots$$

Let k be the natural number such that

$$k \cdot \psi(J_k) < \alpha \quad \text{and} \quad (k+1) \cdot \psi(J_{k+1}) \geq \alpha. \quad (6)$$

Tarone proved that any hypotheses J_i with $i > k$ do not increase the resulting FWER, hence we can discard them (Tarone, 1990). The top- k patterns $J_1, J_2, \dots, J_k$ are called *testable patterns*, and the corrected significance level is given as $\delta_{\text{Tarone}} = \alpha/k$. Thus each testable pattern J_i with $i = 1, 2, \dots, k$ is statistically significant if

$$p\text{-value}(J_i) < \delta_{\text{Tarone}} = \frac{\alpha}{k}.$$

3.3 Minimum Achievable p-value

First we show that the minimum achievable p -value $\psi(J)$ with its fixed frequency $\eta(J)$ exists and can be analytically computed. The minimum p -value is obtained when the KL-divergence from p_O to p_E is maximized. Let $p_O|_x$ be the distribution

$$p_O|_x = \left(x, \eta(J) - x, r_0 - x, r_1 - (\eta(J) - x) \right).$$

Our goal is to find $p_O|_x$ that maximizes $D_{\text{KL}}(p_O|_x, p_E)$ with the constraint $0 < x < \min\{\eta, r_0\}$. Let us denote by

$$\xi(J) = \max_{0 < x < \min\{\eta, r_0\}} D_{\text{KL}}(p_O|_x, p_E).$$

In the following, we fix J and denote by $\eta = \eta(J)$ to simplify the notation and always assume that $r_0 \leq r_1$ and $\eta \leq 0.5$. We immediately obtain results for the case $\eta > 0.5$ by letting $\eta = 1 - \eta$.

Theorem 2 (Maximum achievable KL divergence).

$$\begin{aligned} \xi(J) &= \max_{0 < x < \min\{\eta, r_0\}} D_{\text{KL}}(p_O|_x, p_E) = \lim_{x \rightarrow \min\{\eta, r_0\}} D_{\text{KL}}(p_O|_x, p_E) \\ &= \eta \log \frac{1}{r_0} + (r_0 - \eta) \log \frac{r_0 - \eta}{r_0(1 - \eta)} + (1 - r_0) \log \frac{1}{(1 - \eta)} \end{aligned} \quad (7)$$

if $\eta = \eta(J) < r_0$ and swap η for r_0 otherwise.

Proof. Let $f(x) = D_{\text{KL}}(\mathbf{p}_O|_x, \mathbf{p}_E)$. We have

$$f(x) = x \log \frac{x}{r_0 \eta} + (\eta - x) \log \frac{\eta - x}{r_1 \eta} + (r_0 - x) \log \frac{r_0 - x}{r_0(1 - \eta)} + (r_1 - \eta + x) \log \frac{r_1 - \eta + x}{r_1(1 - \eta)}.$$

The second derivative of $f(x)$ is given as

$$\frac{\partial^2 f}{\partial x^2} = \frac{1}{r_0 - x} + \frac{1}{r_1 - \eta + 1} + \frac{1}{\eta - x} + \frac{1}{x},$$

which is always positive from the constraint $0 < x < \min\{\eta, r_0\}$. Thus $f(x)$ is maximized when x goes to 0 or $\min\{\eta, r_0\}$. The limit of $f(x)$ is obtained as

$$\lim_{x \rightarrow a} f(x) = \begin{cases} \eta \log \frac{1}{r_1} + r_0 \log \frac{1}{(1 - \eta)} + (r_1 - \eta) \log \frac{r_1 - \eta}{r_1(1 - \eta)} & \text{if } a = 0, \\ \eta \log \frac{1}{r_0} + (r_0 - \eta) \log \frac{r_0 - \eta}{r_0(1 - \eta)} + (1 - r_0) \log \frac{1}{(1 - \eta)} & \text{if } a = \eta, \\ r_0 \log \frac{1}{\eta} + (\eta - r_0) \log \frac{\eta - r_0}{\eta(1 - r_0)} + (1 - \eta) \log \frac{1}{(1 - r_0)} & \text{if } a = r_0. \end{cases}$$

Since $\lim_{x \rightarrow r_0} f(x)$ is exactly the same as $\lim_{x \rightarrow \eta} f(x)$ by alternating r_0 and η , we assume $\eta < r_0$ without loss of generality. From the condition $r_1 = 1 - r_0$, the difference $\delta = \lim_{x \rightarrow \eta} f(x) - \lim_{x \rightarrow 0} f(x)$ follows that

$$\begin{aligned} \delta &= -r_0 \log r_0 + r_1 \log r_1 + (r_0 - \eta) \log(r_0 - \eta) - (r_1 - \eta) \log(r_1 - \eta) \\ &= -r_0 \log r_0 + (1 - r_0) \log(1 - r_0) + (r_0 - \eta) \log(r_0 - \eta) - (1 - r_0 - \eta) \log(1 - r_0 - \eta). \end{aligned}$$

The partial derivative of δ with respect to η is

$$\frac{\partial \delta}{\partial \eta} = \log(1 - \eta - r_0) - \log(r_0 - \eta),$$

which is always positive as $(1 - \eta - r_0) - (r_0 - \eta) = 1 - 2r_0 \geq 0$ with the condition $r_0 \leq 0.5 \leq r_1$. Hence the difference δ takes the minimum value at $\eta = 0$ and we obtain $\delta \geq -r_0 \log r_0 + (1 - r_0) \log(1 - r_0) + r_0 \log r_0 - (1 - r_0) \log(1 - r_0) = 0$. Therefore $f(x)$ takes the maximum value at $x = \eta$ and given as

$$\xi(J) = \max f(x) = \lim_{x \rightarrow \eta} f(x) = \eta \log \frac{1}{r_0} + (r_0 - \eta) \log \frac{r_0 - \eta}{r_0(1 - \eta)} + (1 - r_0) \log \frac{1}{(1 - \eta)}.$$

When $\eta > r_0$, we obtain $\xi(J)$ by swapping r_0 for η . □

Next we analyze the behavior of $\xi(J) = \max D_{\text{KL}}(\mathbf{p}_O|_x, \mathbf{p}_E)$ as a function of η with fixed r_0 and r_1 . Note that, although $\xi(J)$ is originally defined as a function of a pattern J , it depends on only its frequency $\eta = \eta(J)$. Assume that $r_0 \leq r_1$. When $\eta < r_0$, we have

$$\frac{\partial \xi}{\partial \eta} = \log \frac{1}{r_0} - \log \frac{r_0 - \eta}{r_0(1 - \eta)} = \log \frac{1 - \eta}{r_0 - \eta} > 0,$$

hence $\xi(J)$ monotonically increases as η increases. When $r_0 < \eta < 0.5$, we have

$$\frac{\partial \xi}{\partial \eta} = \log \frac{\eta - r_0}{\eta(1 - r_0)} - \log \frac{1}{1 - r_0} = \log \frac{\eta - r_0}{\eta} < 0,$$

thereby $\xi(J)$ monotonically decreases as η increases. We illustrate $\xi(J)$ with $r_0 = 0.3$ and the corresponding minimum achievable p -value $\psi(J)$ with $N = 100$ in Figure 2.

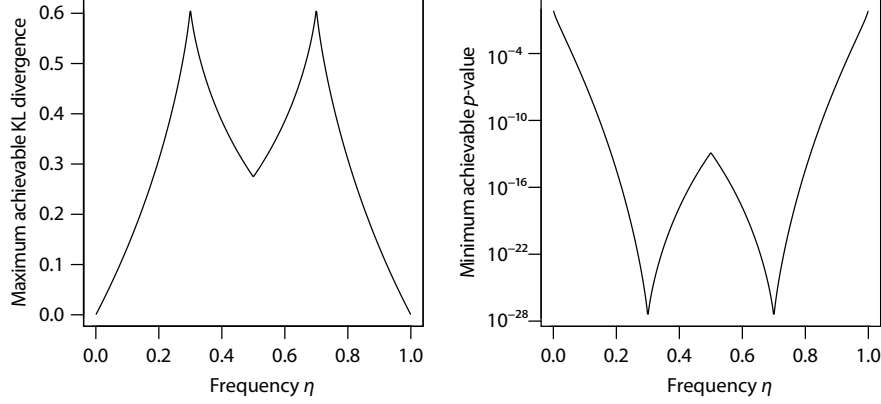


Figure 2: The maximum achievable KL divergence (left-hand side) and the corresponding minimum achievable p -value (right-hand side) with respect to changes in η when $r_0 = 0.3$.

3.4 Enumeration Algorithm

Since we obtain the minimum achievable p -value ψ which has the same property as that for Fisher’s exact test (Llinares-López et al., 2015a), we can employ the enumeration strategy proposed for significant itemset mining (Minato et al., 2014) or subgraph mining (Sugiyama et al., 2015).

We use an Apriori-like algorithm to enumerate patterns $J_1, J_2, \dots, J_k$ such that $\psi(J_1) \leq \psi(J_2) \leq \dots \leq \psi(J_k)$ and k satisfies the condition (6). Let C be a set of candidate patterns, which is the set of frequent patterns such that

$$C = \{ J \in 2^V \mid \eta(J) \geq \sigma \} \quad (8)$$

for a current threshold σ . Since the minimum achievable p -value $\psi(J)$ takes the minimum value when $\eta(J) = r_0$ and is monotonically decreasing as $\eta(J)$ decreases if $\eta(J) \leq r_0$, if $\sigma < r_0$, it is always guaranteed that

$$C \supseteq \mathcal{T} = \{ J \in 2^V \mid \psi(J) \leq \psi_\sigma \} \quad (9)$$

where ψ_σ is the minimum achievable p -value for the frequency σ . This means that the top- $|\mathcal{T}|$ patterns with respect to the increasing order according to ψ is certainly included in C . Thus, if the condition (6) is satisfied for some $k \leq |\mathcal{T}|$, the top- k patterns in $\mathcal{T}$ are the testable patterns.

We show our enumeration algorithm of testable patterns in Algorithm 1. First we set the frequency threshold $\sigma = 0$ and start pattern enumeration in depth-first search. Whenever we find a new frequent pattern J such that $\eta(J) > \sigma$, we update the candidate sets C and $\mathcal{T}$ given in Equations (8) and (9), respectively, and check the condition $|\mathcal{T}| \cdot \psi_\sigma < \alpha$. If it is not satisfied, the threshold σ is too low, thus we update σ as $\sigma = \min_{K \in C} \eta(K)$ and remove the pattern $K_{\min} = \operatorname{argmin}_{K \in C} \eta(K)$ from the two sets C and $\mathcal{T}$. Finally, when the algorithm stops, the set $\mathcal{T}$ is the set of testable patterns, hence each pattern $J \in \mathcal{T}$ is significant if

$$p\text{-value}(J) < \frac{\alpha}{|\mathcal{T}|}.$$

Note that, in practice the set $\mathcal{T}$ of testable patterns can be massive and it is hard to store all testable patterns in the main memory. In such a case, we just count the number of testable patterns in the function TESTABLEPATTERNENUMERATION, and after the termination of the algorithm with the final frequency $\sigma = \sigma_{\text{final}}$, we repeat the same function with fixing the frequency threshold $\sigma = \sigma_{\text{final}}$ and check whether or not $p\text{-value}(J) < \alpha/|\mathcal{T}|$ for each frequent pattern.

Algorithm 1: Significant pattern mining algorithm.

```
1 SIGNIFICANTPATTERNMINING( $D, \alpha$ )
2    $\sigma \leftarrow 0$  // this is a global variable
3    $C \leftarrow \emptyset$ 
4   TESTABLEPATTERNENUMERATION( $\emptyset, 0, C, D, \alpha$ )
5    $\mathcal{T} \leftarrow \{K \in C \mid \psi(K) \leq \psi_\sigma\}$  // The set of testable patterns
6   SIGNIFICANCETESTING( $\mathcal{T}, \alpha$ )
7 TESTABLEPATTERNENUMERATION( $J, j_{\text{prev}}, C, D, \alpha$ )
8   foreach  $j \in \{j_{\text{prev}} + 1, \dots, n\}$  do
9      $J \leftarrow J \cup \{j\}$ 
10    Compute the frequency  $\eta(J)$  by Equation (4)
11    if  $\eta(J) > \sigma$  then
12      Compute  $\psi(J)$  from  $\xi(J)$  by Equation (7)
13       $C \leftarrow C \cup \{J\}$  // store  $\eta(J)$  and  $\psi(J)$  together
14       $\mathcal{T} \leftarrow \{K \in C \mid \psi(K) \leq \psi_\sigma\}$ 
15      while  $|\mathcal{T}| \cdot \psi_\sigma \geq \alpha$  do
16         $K_{\min} \leftarrow \operatorname{argmin}_{K \in C} \eta(K)$ 
17         $\sigma \leftarrow \min\{r_0, \eta(K_{\min})\}$ 
18         $C \leftarrow C \setminus \{K_{\min}\}$ 
19         $\mathcal{T} \leftarrow \mathcal{T} \setminus \{K_{\min}\}$ 
20      TESTABLEPATTERNENUMERATION( $J, j, C, D, \alpha$ )
21     $J \leftarrow J \setminus \{j\}$ 
22 SIGNIFICANCETESTING( $\mathcal{T}, \alpha$ )
23   foreach  $J \in \mathcal{T}$  do
24     if  $p\text{-value}(J) < \alpha / |\mathcal{T}|$  then
25       Output  $J$ 
```

4 Related Work

First we summarize studies on finding feature combinations from continuous data. The seminal work in the pattern mining community is *numerical pattern mining* by Kaytoue et al. (Kaytoue et al., 2011b,a). Their method, called MinIntChange, binarizes a given dataset by *interordinal scaling*, followed by applying closed itemset mining. MinIntChange uses an efficient pruning technique to speed-up the itemset mining process, while the resulting patterns are exactly the same as those obtained by a two-step procedure: binarizing a continuous dataset by interordinal scaling and applying a closed itemset mining algorithm such as LCM (Uno et al., 2004). In interordinal scaling, each binarized feature is in the form of “ $\leq a$ ” or “ $\geq a$ ”, where endpoints a are from a dataset, that is, $a \in \{x_1^j, x_2^j, \dots, x_N^j\}$ for a feature $j \in \{1, 2, \dots, n\}$. Thus, for an n -dimensional real-valued vector $\mathbf{x}_i \in D$, each element x_i^j is expanded as the $2N$ -dimensional binary vector such that

$$\left(x_i^{\leq x_1^j}, x_i^{\leq x_2^j}, \dots, x_i^{\leq x_N^j}, x_i^{\geq x_1^j}, x_i^{\geq x_2^j}, \dots, x_i^{\geq x_N^j} \right), \quad (10)$$

where each value for the binarized feature $x_i^{\leq a}$ is 1 if $x_i^j \leq a$ and 0 otherwise. As a result, the dataset D is converted into the binary dataset with $2nN$ features.

A relevant task has been studied in the area of *subgroup discovery* (Atzmueller, 2015; Novak et al., 2009; Herrera et al., 2011). While the motivation is close to that of pattern mining, they use not the frequency but other quality functions. Grosskreutz and Rüping were the first to study subgroup discovery from continuous data (Grosskreutz and Rüping, 2009). In their method, called MergeSD, each binarized feature is in the form of “ $(a, b]$ ”, where endpoints a and b are from a dataset. For an n -dimensional real-valued vector $\mathbf{x}_i$, each element x_i^j is expanded as the $(N(N - 1)/2)$ -dimensional binary vector such that

$$\left(x_i^{(x_1^j, x_2^j]}, x_i^{(x_1^j, x_3^j]}, \dots, x_i^{(x_1^j, x_N^j]}, x_i^{(x_2^j, x_3^j]}, \dots, x_i^{(x_{N-1}^j, x_N^j]} \right). \quad (11)$$

Thus a dataset D is converted into the binary dataset with $nN(N - 1)/2$ features. After binarization, they apply Apriori-like algorithms to find all feature combinations whose quality is above the user-specified threshold. Grosskreutz and Rüping (2009) used a density-based quality function, while van Leeuwen and Ukkonen (2016) used a mean deviation-based quality function.

Next we overview studies on significant pattern mining, which introduces the statistical significance into the task of contrast (discriminative) pattern mining (Dong and Bailey, 2013), where the objective is to find discriminative patterns with respect to class partitioning of a dataset. After seminal works on pattern mining that finds statistically significant association rules (Webb, 2007; Hämmäläinen, 2012), Terada et al. (2013a) were the first to achieve the multiple testing correction in itemset mining with controlling the FWER with successfully combining a pattern mining algorithm and the Tarone’s trick (Tarone, 1990). The enumeration algorithm has been improved in LAMP ver. 2 (Minato et al., 2014) and significant subgraph mining (Sugiyama et al., 2015). To date, significant pattern mining has been actively extended to various types of tests and data, including a Westfall-Young permutation test to treat independencies among patterns (Terada et al., 2013b; Llinares-López et al., 2015b), logistic regression (Terada et al., 2016) or a Cochran–Mantel–Haenszel (CMH) test (Papaxanthos et al., 2016) for categorical covariates, and hypothesis streams (Webb and Petitjean, 2016). However, none of the above studies succeeded to directly perform significant pattern mining on continuous variables without any binarization.

5 Experiments

We examine the efficiency and the effectiveness of our method using synthetic and real-world datasets. We used Amazon Linux AMI release 2016.09 and ran all experiments on a single core of 2.3 GHz Intel Xeon CPU E7-8880 v3 and 2.0 TB of memory. All methods were implemented in C/C++ with the Eigen library and compiled with gcc 4.8.3. We provide the code and data for experiments at <http://mahito.info/codes/kdd2017/>.

5.1 Comparison Partners

We use the following four comparison partners, where only the first one directly mines feature combinations from continuous data, while the other three methods binarize a given dataset and apply significant itemset mining to binarized data.

Bonferroni correction: To examine the contribution of the testability trick introduced in Section 3.3, we compare our method to Bonferroni correction. Since there is no method that can directly perform significant pattern mining on continuous variables, this is the unique comparison partner that also finds feature

combinations without any binarization. In this method, we use the same frequency with our proposed method in Equation (4) but enumerate all 2^V patterns instead of testable ones and compute their p -values to determine whether or not each of the patterns is statistically significant, that is, the p -value is smaller than $\alpha/2^{|V|}$.

Interordinal scaling + significant itemset mining: The next comparison partner represents the numerical pattern mining technique (Kaytoue et al., 2011a). We use the same binarization method, interordinal scaling given in Equation (10), as in numerical pattern mining, and apply significant itemset mining to the binarized data instead of frequent itemset mining. We use the state-of-the-art significant itemset mining algorithm LAMP ver.2², which incorporates the fastest frequent pattern mining algorithm LCM (Uno et al., 2004) to enumerate testable itemsets.

Interval binarization + significant itemset mining: This method uses interval binarization given in Equation (11) as a representative of subgroup discovery. Since there is no method in subgroup discovery that can find significant feature combinations with controlling the FWER, we use the same binarization technique used in MergeSD (Grosskreutz and Rüping, 2009), followed by applying a significant itemset mining algorithm.

Median-based binarization + significant itemset mining: Finally, we employ median-based binarization as a binarization method before applying significant itemset mining. For each feature $j \in \{1, 2, \dots, n\}$, we pick up the median of $\mathbf{x}^j = (x_1^j, x_2^j, \dots, x_N^j)$, denoted by $\text{med}(j)$, and binarize each value x_i^j as a pair

$$\left(x_i^{\leq \text{med}(j)}, x_i^{> \text{med}(j)} \right),$$

where $x_i^{\leq \text{med}(j)} = 1$ if $x_i^j \leq \text{med}(j)$ and 0 otherwise, and $x_i^{> \text{med}(j)} = 1$ if $x_i^j > \text{med}(j)$ and 0 otherwise. Thus a given dataset is converted to the binarized dataset with $2n$ features, which is expected to be more efficient than the above two approaches.

5.2 Evaluation Criteria

We use the following measures to evaluate our method compared with the above partners.

Number of testable patterns: First we count the number $|\mathcal{T}|$ of testable patterns to examine the effectivity of the testability trick. Note that, in Bonferroni correction, this number is always 2^n as it checks all patterns. The smaller value means that the corresponding method prunes more unnecessary patterns with increasing the power of statistical tests.

Running time: To examine the efficiency of the set of methods, we measure the running time needed for enumerating all significant patterns. In the three binarization-based methods, we exclude the time used for binarization as the process is efficient enough and negligible compared to the pattern mining step.

Final Frequency threshold: We also check the final frequency threshold σ_{final} obtained at the enumeration step of testable patterns, that is, the set of testable pattern $\mathcal{T}$ coincides with the set of patterns $\{J \in 2^V \mid \psi(J) \leq \psi_{\sigma_{\text{final}}}\}$, where $\psi_{\sigma_{\text{final}}}$ is the minimum achievable p -value at the frequency σ_{final} . Since σ_{final} is always 0 in Bonferroni correction, a higher threshold means that we can prune more untestable patterns.

Precision and recall: To examine the quality of detected patterns, we compute precision and recall in experiments on synthetic data. We compare our method and binarization-based methods with excluding the Bonferroni correction method as it controls the FWER at the different level. Let $A \subseteq \{1, 2, \dots, n\}$ be the

²We used the implementation available at <https://www.bsse.ethz.ch/mlcb/research/machine-learning/wylight.html>

Table 2: Statistics of real data.

	N	N_0	n
ctg	2126	471	22
faults	1941	158	27
ionosphere	351	126	34
segment	2310	330	19
waveform	3353	1657	21
wdbc	569	212	30

Table 3: Results on real data.

data	Number of testable patterns			Runtime (sec.)		
	Proposed	Bonferroni	Binarization	Proposed	Bonferroni	Binarization
ctg	4.37×10^5	4.19×10^6	7.37×10^6	2.23×10^1	1.19×10^1	9.77×10^1
faults	3.53×10^6	1.34×10^8	9.61×10^6	1.78×10^2	2.61×10^1	2.71×10^3
ionosphere	1.57×10^5	1.72×10^{10}	9.66×10^6	3.96×10^0	2.71×10^1	$\geq 48\text{h}$
segment	1.78×10^5	5.24×10^5	5.25×10^5	7.81×10^0	1.12×10^0	1.39×10^1
waveform	1.98×10^5	2.10×10^6	2.08×10^7	1.61×10^1	3.37×10^1	7.92×10^1
wdbc	1.74×10^5	1.07×10^9	8.79×10^6	4.83×10^0	3.34×10^1	$\geq 48\text{h}$

data	Final frequency threshold	
	Proposed	Binarization
ctg	4.36×10^{-3}	6.50×10^{-2}
faults	3.28×10^{-3}	4.00×10^{-2}
ionosphere	3.52×10^{-2}	9.00×10^{-2}
segment	2.92×10^{-3}	4.50×10^{-2}
waveform	4.87×10^{-3}	1.45×10^{-1}
wdbc	2.29×10^{-2}	9.50×10^{-2}

features that are associated with class labels (see the next subsection for the definition of the association). In the binarization-based methods, A is the set of binarized features such that each of them is generated from a feature in the original A . Suppose that S is the set of significant patterns. For each pattern $J \in S$, J is true positive if $J \subseteq A$. Thus precision and recall are given as $TP/|S|$ and $TP/2^{|A|}$, respectively, where $TP = |\{J \in S \mid J \subseteq A\}|$.

5.3 Results on Synthetic Data

To examine the evaluation criteria, we generate synthetic data, varying the number N of samples from 100 to 100,000, the number n of features from 10 to 200, and setting the class ratio to $r_0 = 0.5$ or $r_0 = 0.1$, i.e., the number N_0 of samples in the minor class is $N/2$ or $N/10$. In each dataset, we generate 20 % of features that are positively correlated with the class labels, where the top- N_0 samples with respect to their original continuous values get the class label 0, while the others have the class label 1. These 20 %

of correlated features are used to compute the true positives. The other 80 % of features are uninformative features generated from the uniform distribution.

Results are plotted in Figure 3 for $r_0 = 0.5$ and Figure 4 for $r_0 = 0.1$. In the respective figure, we plot results with varying N while fixing $n = 20$ on the left column and those with varying n while fixing $N = 200$ on the right column. First, two binarization-based methods, interordinal scaling and interval binarization, do not finish their computation even for the smallest dataset with $N = 100$ and $n = 10$ after 48 hours, and they exceed the memory limit (2.0 TB) for larger datasets. We also tried a tiny dataset $N = 50$ and $n = 5$, then the interordinal scaling approach finishes about 24 hours. This is why they generate many binarized features as we showed in Section 4 and the binarized dataset is dense, resulting in too heavy computation in the itemset mining step. Therefore we exclude these two methods in the following analysis.

The advantage of our method (plotted in blue in the figures) is clear if we compare to the Bonferroni correction method (plotted in red) with respect to both the number of testable patterns and running time. In particular, if the number n of features gets larger and larger, the Bonferroni correction method becomes exponentially slower due to combinatorial explosion of the candidate space, while our method can finish mining in a reasonable time by pruning untestable patterns. We can also confirm the advantage of the testability trick from the final frequency thresholds, which are higher than 0 and hence infrequent patterns are pruned as untestable patterns.

In comparison with the median-based binarization method (plotted in green), our method has a clear advantage regarding the quality of detected significant patterns. In all cases, both precision and recall of our method are higher than those of the median-based binarization method. These results confirm the drawback of binarization based methods, which we discussed in the Introduction and illustrated in Figure 1, that is, the binarization based method cannot distinguish correlated and uncorrelated patterns. In both methods, precision drops when the number N of samples becomes large: As we gain more and more statistical power for larger N , many feature combinations, even those with very small dependence to the class labels, tend to reach statistical significance. In contrast, recall drops when the number n of features becomes large, due to the growing number of testable patterns, resulting in a loss of statistical power in finding significant patterns.

To summarize, we observe that our proposal is the only method that efficiently mines significant patterns by avoiding the combinatorial explosion of the number of tests and that effectively finds correlated features.

5.4 Results on Real Data

We evaluate our proposal on real-world datasets shown in Table 2, which are benchmark datasets for binary classification from the UCI repository³. Since the true causal features are unknown in those datasets, we evaluated the number of testable patterns, running time, and the final frequency thresholds.

We summarize results in Table 3, where we observe the same trend as before: First, our method is more efficient than the Bonferroni correction method by pruning untestable patterns. Second, although we cannot report precision and recall here, the number of testable patterns in our method is one or two orders of magnitude smaller than that in median-based binarization, which indicates that the median-based binarization method generates many redundant features.

6 Conclusion

In this paper, we have addressed the challenge of extending significant pattern mining to continuous features. Key to our approach is to represent the frequency of a pattern by means of Spearman’s rank correlation

³<http://archive.ics.uci.edu/ml/>

coefficient. Our experimental results demonstrate that our approach is superior in detecting true patterns compared to all pattern mining approaches that require a prior binarization.

Our work opens the door to many applications of significant pattern mining, in which the data is not adequately described by binary features, including large fields such as data analysis for high-throughput technologies in biology and medicine. Our work here addresses the problem of mining continuous *features*. Significant pattern mining on *continuous class labels* (or output variables) is an equally challenging and practically relevant problem, that we will tackle in future work.

References

- C. C. Aggarwal and J. Han, editors. *Frequent Pattern Mining*. Springer, 2014.
- R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Data Bases*, pages 487–499, 1994.
- R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, 1993.
- M. Atzmueller. Subgroup discovery. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 5(1):35–49, 2015.
- C. E. Bonferroni. Teoria statistica delle classi e calcolo delle probabilità. *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commerciali di Firenze*, 8:3–62, 1936.
- R. Breitling, P. Armengaud, A. Amtmann, and P. Herzyk. Rank products: a simple, yet powerful, new method to detect differentially regulated genes in replicated microarray experiments. *FEBS Letters*, 573(1-3):83–92, 2004.
- K. Coussement, K. W. De Bock, and S. A. Neslin, editors. *Advanced Database Marketing*. Routledge, 2013.
- C. Davatzikos, K. Ruparel, Y. Fan, D. G. Shen, M. Acharyya, J. W. Loughhead, R. C. Gur, and D. D. Langleben. Classifying spatial patterns of brain activity with machine learning methods: Application to lie detection. *NeuroImage*, 28(3):663–668, 2005.
- G. Dong and J. Bailey, editors. *Contrast Data Mining: Concepts, Algorithms, and Applications*. CRC Press, 2013.
- R. Eisinga, R. Breitling, and T. Heskes. The exact probability distribution of the rank product statistics for replicated experiments. *FEBS Letters*, 587(6):677–682, 2013.
- R. A. Fisher. On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 222(594-604):309–368, 1922.
- H. Grosskreutz and S. Rüping. On subgroup discovery in numerical domains. *Data Mining and Knowledge Discovery*, 19(2):210–226, 2009.
- W. Hämmäläinen. Kingfisher: an efficient algorithm for searching for both positive and negative dependency rules with statistical significance measures. *Knowledge and Information Systems*, 32(2):383–414, 2012.

- J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 1–12, 2000.
- F. Herrera, C. J. Carmona, P. González, and M. J. del Jesus. An overview on subgroup discovery: foundations and applications. *Knowledge and Information Systems*, 29(3):495–525, 2011.
- T. Heskes, R. Eisinga, and R. Breitling. A fast algorithm for determining bounds and accurate approximate p-values of the rank product statistic for replicate experiments. *BMC Bioinformatics*, 15(1):367, 2014.
- M. Kaytoue, S. O. Kuznetsov, and A. Napoli. Revisiting numerical pattern mining with formal concept analysis. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 1342–1347, 2011a.
- M. Kaytoue, S. O. Kuznetsov, A. Napoli, and S. Duplessis. Mining gene expression data with pattern structures in formal concept analysis. *Information Sciences*, 181:1989–2001, 2011b.
- J. A. Koziol. The rank product method with two samples. *FEBS Letters*, 584(21):4481–4484, 2010.
- F. Llinares-López, D. G. Grimm, D. A. Bodenham, U. Gieraths, M. Sugiyama, B. Rowan, and K. M. Borgwardt. Genome-wide detection of intervals of genetic heterogeneity associated with complex traits. *Bioinformatics*, 31(12):i240–i249, 2015a.
- F. Llinares-López, M. Sugiyama, L. Papaxanthos, and K. M. Borgwardt. Fast and memory-efficient significant pattern mining via permutation testing. In *Proceedings of the 21st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 725–734, 2015b.
- S. Minato, T. Uno, K. Tsuda, A. Terada, and J. Sese. A fast method of statistical assessment for combinatorial hypotheses based on frequent itemset enumeration. In *Machine Learning and Knowledge Discovery in Databases*, volume 8725 of *LNCS*, pages 422–436. Springer, 2014.
- H. Nakahara, S. Nishimura, M. Inoue, G. Hori, and S. Amari. Gene interaction in DNA microarray data is decomposed by information geometric measure. *Bioinformatics*, 19(9):1124–1131, 2003.
- P. K. Novak, N. Lavrač, and G. I. Webb. Supervised descriptive rule discovery: A unifying survey of contrast set, emerging pattern and subgroup mining. *The Journal of Machine Learning Research*, 10:377–403, 2009.
- L. Papaxanthos, F. Llinares-Lopez, D. Bodenham, and K. M. Borgwardt. Finding significant combinations of features in the presence of categorical covariates. In *Advances in Neural Information Processing Systems*, volume 29, pages 2271–2279, 2016.
- J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu. PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth. In *Proceedings of the 17th International Conference on Data Engineering*, pages 215–224, 2001.
- C. Spearman. The proof and measurement of association between two things. *The American journal of psychology*, 15(1):72–101, 1904.
- J. M. Stuart, E. Segal, D. Koller, and S. K. Kim. A gene-coexpression network for global discovery of conserved genetic modules. *Science*, 302(5643):249–255, 2003.

- M. Sugiyama, F. Llinares-López, N. Kasenburg, and K. M Borgwardt. Significant subgraph mining with multiple testing correction. In *Proceedings of 2015 SIAM International Conference on Data Mining*, pages 37–45, 2015.
- R. E. Tarone. A modified Bonferroni method for discrete data. *Biometrics*, 46(2):515–522, 1990.
- N. Tatti. Itemsets for real-valued datasets. In *2013 IEEE 13th International Conference on Data Mining*, pages 717–726, 2013.
- A. Terada, M. Okada-Hatakeyama, K. Tsuda, and J. Sese. Statistical significance of combinatorial regulations. *Proc. Natl. Acad. Sci. USA*, 110(32):12996–13001, 2013a.
- A. Terada, K. Tsuda, and J. Sese. Fast Westfall-Young permutation procedure for combinatorial regulation discovery. In *2013 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 153–158, 2013b.
- A. Terada, D. duVerle, and K. Tsuda. Significant pattern mining with confounding variables. In *Advances in Knowledge Discovery and Data Mining (PAKDD 2016)*, volume 9651 of *LNCS*, pages 277–289, 2016.
- T. Uno, T. Asai, Y. Uchida, and H. Arimura. An efficient algorithm for enumerating closed patterns in transaction databases. In *Discovery Science*, volume 3245 of *Lecture Notes in Computer Science*, pages 16–31, 2004.
- M. van Leeuwen and A. Ukkonen. Expect the unexpected – on the significance of subgroups. In T. Calders, M. Ceci, and D. Malerba, editors, *Discovery Science*, volume 9956 of *Lecture Notes in Computer Science*, pages 51–66, 2016.
- G. I. Webb. Discovering significant patterns. *Machine Learning*, 68(1):1–33, 2007.
- G. I. Webb and F. Petitjean. A multiple test correction for streams and cascades of statistical hypothesis tests. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1255–1264, 2016.
- Barnet Woolf. The log likelihood ratio test (the *G*-test). *Annals of human genetics*, 21(4):397–409, 1957.
- X. Yan and J. Han. gSpan: Graph-based substructure pattern mining. In *Proceedings of 2002 IEEE International Conference on Data Mining*, pages 721–724, 2002.

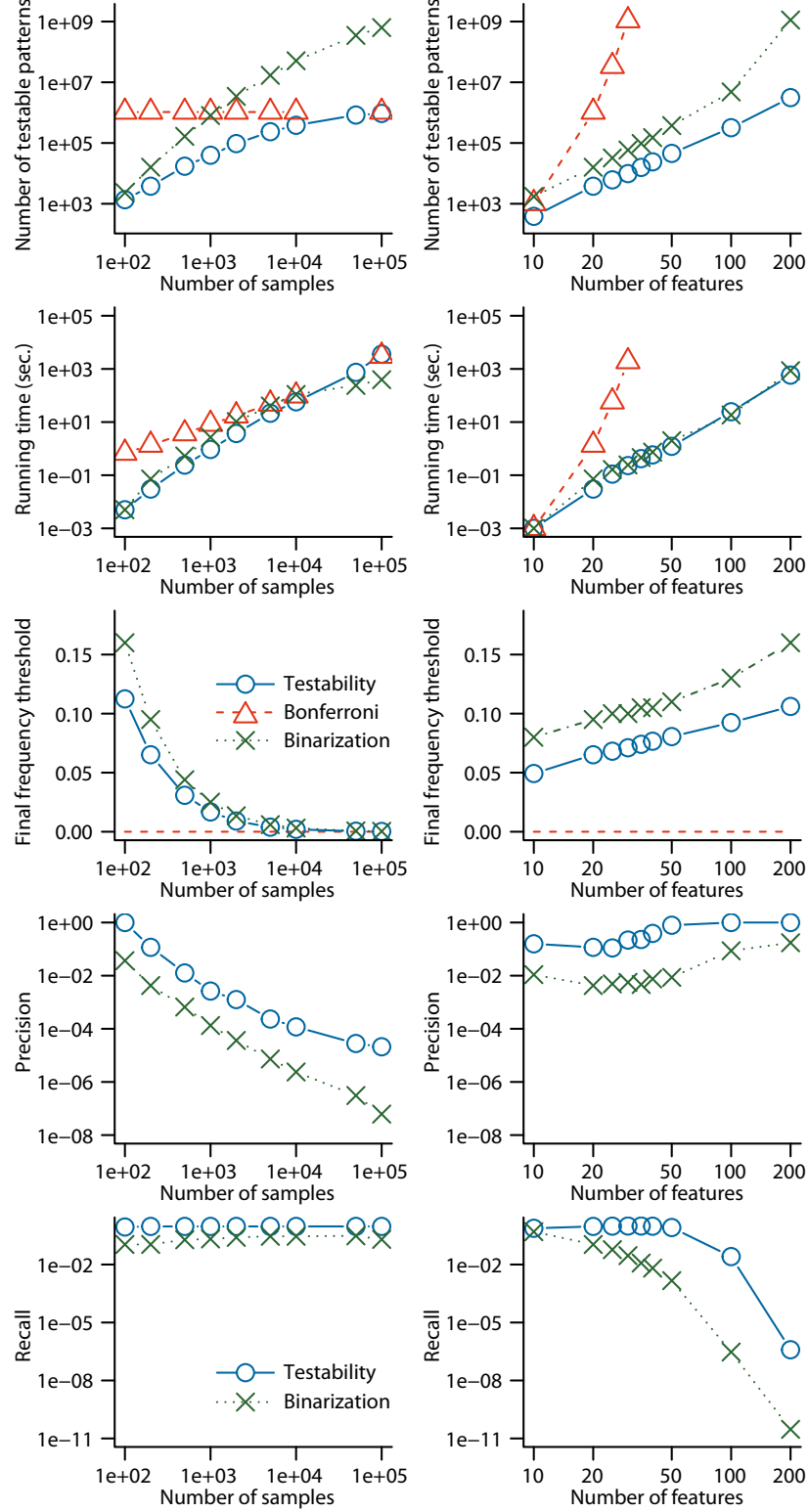


Figure 3: Results on synthetic data with the minor class ratio $r_0 = 0.5$. The number n of features is fixed at 20 in the left column and the number N of samples is fixed at 200 in the right column.

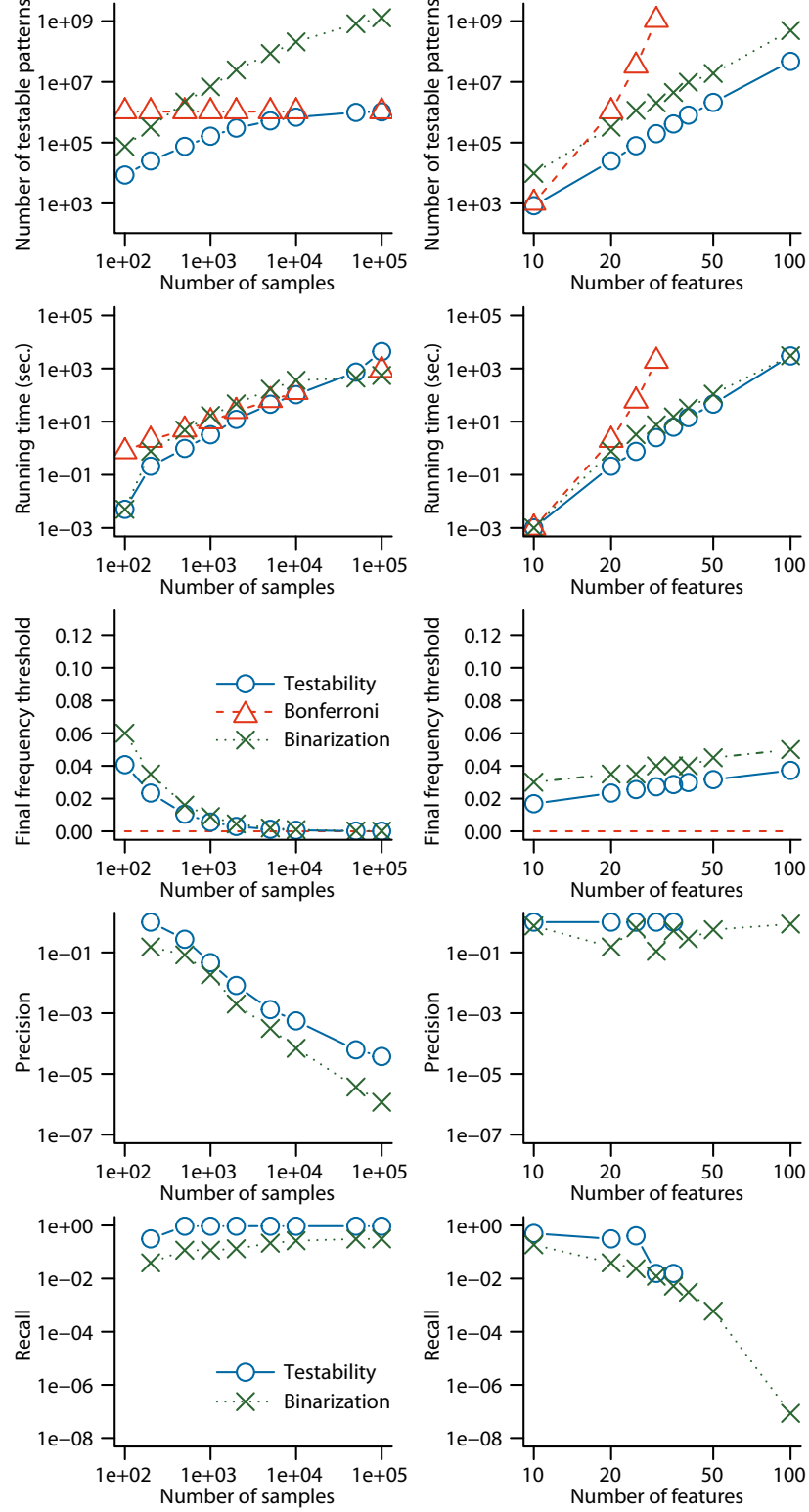


Figure 4: Results on synthetic data with the minor class ratio $r_0 = 0.1$. The number n of features is fixed at 20 in the left column and the number N of samples is fixed at 200 in the right column. Missing values in the plots of precision and recall mean that there is no significant pattern.