Mach Learn (2007) 68: 1–33
DOI 10.1007/s10994-007-5006-x

Discovering Significant Patterns

Geoffrey I. Webb

Received: 23 May 2005 / Revised: 13 February 2007 /
Accepted: 14 February 2007 / Published online: 14 April 2007
Springer Science+Business Media, LLC 2007

Abstract Pattern discovery techniques, such as association rule discovery, explore large search spaces of potential patterns to find those that satisfy some user-specified constraints. Due to the large number of patterns considered, they suffer from an extreme risk of type-1 error, that is, of finding patterns that appear due to chance alone to satisfy the constraints on the sample data. This paper proposes techniques to overcome this problem by applying well-established statistical practices. These allow the user to enforce a strict upper limit on the risk of experimentwise error. Empirical studies demonstrate that standard pattern discovery techniques can discover numerous spurious patterns when applied to random data and when applied to real-world data result in large numbers of patterns that are rejected when subjected to sound statistical evaluation. They also reveal that a number of pragmatic choices about how such tests are performed can greatly affect their power.

Keywords Pattern discovery · Statistical evaluation · Association rules

1 Introduction

This paper addresses the problem of using statistical hypothesis tests to screen individual patterns in the context of discovering large numbers of patterns from a single set of data. This problem arises in *pattern discovery*, as exemplified by *association rule discovery* (Agrawal et al. 1993), *k-optimal* or *k-top rule discovery* (Webb 1995; Scheffer and Wrobel 2002; Webb and Zhang 2005), *contrast* or *emerging pattern discovery* (Bay and Pazzani 2001; Dong and Li 1999), *subgroup discovery* (Klösgen 1996), *interesting itemset discovery* (Jaroszewicz and Simovici 2004) and *impact* or *quantitative rule discovery* (Aumann 1999; Webb 2001; Zhang et al. 2004). All these techniques search large spaces of possible patterns $\mathcal{P}$ and return all patterns that satisfy user-defined constraints. Such a process entails evaluating numerous

Editor: Johannes Fürnkranz.

G.I. Webb (✉)
Faculty of Information Technology, Monash University, PO Box 75, Clayton, Vic., 3800, Australia
e-mail: webb@infotech.monash.edu

patterns $\rho \in \mathcal{P}$ against a set of sample data D drawn from a distribution Θ, seeking those ρ that satisfy user-specified constraints ϕ with respect to Θ. Each time such an evaluation is performed there is a risk that the pattern ρ will satisfy ϕ with respect to the sample data D and hence that it will be inferred that ρ satisfies ϕ with respect to Θ, even though this inference is incorrect. Even if this risk is tightly bounded for each ρ, by considering numerous patterns, the risk of accepting at least one erroneous pattern can grow large. For example, suppose the risk is bounded by applying a statistical test for ϕ, accepting each pattern only if the significance level falls below critical value κ. If n independent patterns are assessed, the risk of accepting at least one erroneous pattern by chance is $1 - (1 - \kappa)^n$, which, for even relatively small values of κ and n, rapidly approaches 1.0. This problem is closely related to the *multiple comparisons problem* (Jensen and Cohen 2000) and the problem of *oversearch* (Quinlan and Cameron-Jones 1995).

Most pattern discovery systems do little to control this risk. While a number of approaches to controlling this risk have been developed (for example, Megiddo and Srikant 1998; Bay and Pazzani 2001; Webb 2002), each has limitations as detailed in Sect. 4.

The current paper investigates two approaches to applying statistical tests in pattern discovery. The first applies a Bonferroni correction for multiple tests (Shaffer 1995), dividing the global significance level α by the number of patterns in the search space in order to obtain the critical value κ. The second divides the available data into *exploratory* and *holdout* sets. The exploratory data are used for pattern discovery. The holdout data are then used to assess the patterns so discovered, with any set of statistical hypothesis tests the user desires being applied to each pattern in turn. The critical value used with the hypothesis tests is adjusted for the number of patterns tested using a correction for multiple tests such as the Bonferroni correction.

These approaches

- can apply any type of statistical hypothesis test to each of the individual patterns discovered by a system;
- can apply any number of such hypothesis tests to each of the patterns; and
- provide precise control over the experimentwise risk of type-1 error. That is, under the assumption that D is an iid sample from Θ, they allow the experimenter to establish a precise upper-bound on the risk of any of the multiple hypothesis tests falsely rejecting a null hypothesis and thus on the risk of falsely accepting a pattern.

Further, the holdout-evaluation approach can be applied as a simple wrapper to any pattern discovery system.

These approaches to handling multiple comparisons are well established in statistical theory and practice (Shaffer 1995). The contribution of the current work is to recognize their applicability in the context of the massive search spaces frequently explored in pattern discovery, to highlight the critical need for such techniques, and to investigate their relative strengths and weaknesses in this context.

The use of holdout evaluation should be very familiar to most machine learning researchers, who frequently use a similar process, dividing data into *training* and *test* sets, forming models from the former and obtaining unbiased estimates of their expected performance by observing their performance against the latter. The differences are that rather than obtaining an unbiased estimate of the quality of a single model, such as its error or ROC curve, we are seeking to either accept or reject each of many patterns, and we are doing so in a manner that strictly controls the risk of false discoveries.

The paper is organized as follows. Section 2 describes pattern discovery. Section 3 provides a formal problem statement. Section 4 discusses previous approaches to statistical testing within pattern discovery and outlines their limitations. Section 5 presents the new approaches. Section 6 summarises a series of experiments that assess key attributes of the new technique and compare its performance to that of previous approaches. Details of the experiments are presented in Appendices 1 and 2. Section 7 provides a general discussion including directions for future research. Section 8 concludes the paper with a summary and concluding remarks.

The holdout-evaluation technique was first presented in Webb (2003) and the direct-adjustment technique first presented in Webb (2006). Experiment 1 in the current paper, and Sect. 6.1 that discusses it, is reproduced from Webb (2006) and Experiments 2, 3 and 4 are more extensive variants of experiments in that previous paper. The discussions of previous approaches in Sect. 4 and the new approaches in Sect. 5 are substantially expanded upon those in the earlier papers. The general discussion presented in Sect. 7 is entirely new.

2 Pattern Discovery

Most machine learning systems learn a single model from the available data. The model learned is usually that expected to maximize accuracy or some other metric on unseen future data. Many systems that learn explicit models, such as decision tree (Quinlan 1993) or decision rule (Michalski 1983) learners, do so by searching a space of alternative models to select the model that appears to perform best with respect to the available data.

Frequently during such a search through the space of models the learner will encounter alternatives that perform equally well or almost as well as each other on the available data. For example, during decision tree induction it is common to find multiple alternative splits at a node all of which score as well or nearly as well on the split selection criterion.

A machine learning system must inevitably make arbitrary choices between such alternative models. Quite simply, it must choose one of these models and has no non-arbitrary basis on which to do so. This is acceptable if there is no other source of information about the potential models. If several alternatives appear equally good, there is no basis on which to prefer any one over the others and it does not matter which is selected.

However, in some contexts there are factors not available to the learning system that can help distinguish between the models.

- Experts may be able to bring to bear background knowledge of a form that would be difficult to encode and make available to a learning system.
- Alternative models may have different levels of desirability because they use different attributes that represent tests with differing costs to evaluate (Turney 2000). For example, in a medical diagnosis task it might be possible to form alternative models of equivalent power, one using a simple blood test and the other using an expensive MRI scan. Clearly the former model will be more desirable. However, most learners are unable to take account of the costs of tests during a learning process.
- Some models may be socially or politically unacceptable, for example because they do not fit into senior management's conceptualization of the business.
- Some models may be unusable due to legislated restrictions.
- Alternative models may be more or less acceptable to a user population simply because one is consistent with existing knowledge and beliefs and the other is not.
- Finally, in some applications it may be possible to deploy multiple models and hence be unnecessary to derive just one.

For these reasons it is often desirable to find all models that satisfy some criteria with regard to the data. In such pattern discovery tasks, the user specifies constraints on the models to be discovered and the system discovers all models that satisfy those constraints. For many pattern discovery techniques the models take the form of rules. However, the same underlying techniques may be applied to other forms of model of regularities in the data, such as itemsets (Agrawal et al. 1993), sequential patterns (Agrawal and Srikant 1995) and sub-graphs (Kuramochi and Karypis 2001). The best known example of pattern discovery is *association rule discovery* (Agrawal et al. 1993).

3 Problem Statement

As outlined in the introduction, pattern discovery seeks to identify patterns $\rho \in \mathcal{P}$ that satisfy constraints ϕ with respect to distribution Θ. However, whether ρ satisfies ϕ with respect to Θ is assessed by reference to sample data D drawn from Θ. Although the principles extend directly to further contexts, the current research limits consideration to two types of data, *transactional data* and *attribute-value data*, and one type of pattern, *rules*.

For both data types, D is a multiset of n records and each record $R \in D$ is a set of items $R \subseteq I$. For transactional data, items are atomic terms. For attribute-value data, there exists a set of a attributes $A_1 \ldots A_a$, each attribute A_i has a domain of $\#A_i$ values $\mathrm{dom}(A_i)$, each item is an attribute-value pair denoted as $A_i = v$, where $v \in \mathrm{dom}(A_i)$, and each record $R \in D$ contains exactly one item for each attribute.

Rules take the form $X \to y$, where $X \subseteq I$, $|X| \geq 1$ and $y \in I$. X is called the *antecedent* and y the *consequent* of the rule. For attribute-value data, $X \cup \{y\}$ may contain no more than one item for any one attribute.

Association rule discovery finds all rules that satisfy specified constraints ϕ specified as a minimum *support* (Agrawal et al. 1993), together with other constraints, if desired, such as minimum *confidence* (Agrawal et al. 1993), *lift* (International Business Machines 1996), or *leverage* (Piatetsky-Shapiro 1991). These terms are defined with respect to a rule $X \to y$ and dataset D as follows:

- $\mathit{coverage}(X \to y)$ is $|\{R \in D : X \subseteq R\}|$;
- $\mathit{support}(X \to y)$ is $|\{R \in D : X \cup \{y\} \subseteq R\}|$;
- $\mathit{confidence}(X \to y) = \mathit{support}(X \to y)/\mathit{coverage}(X \to y)$. This can be viewed as a maximum likelihood estimate of the conditional probability $P(y \mid X)$;
- $\mathit{lift}(X \to y) = \mathit{confidence}(X \to y)/\mathit{confidence}(\emptyset \to y)$.
- $\mathit{leverage}(X \to y) = \mathit{support}(X \to y) - \mathit{coverage}(X \to y) \times |\{R \in D : y \in R\}| \,/\, |D|$, This measure represents the difference between the support and the support that would be expected if X and y were independent.

Each assessment of whether a given ρ satisfies ϕ is accompanied by a risk that the pattern ρ will satisfy ϕ with respect to the sample data D but not with respect to Θ. Most pattern discovery systems fail to effectively control this risk.

Statistical hypothesis tests are applicable to such a scenario. To apply such a test it is necessary to specify a *null hypothesis*, in our context the hypothesis that the negation of ϕ is true. If the discovery process "discovers" a pattern ρ that satisfies the null hypothesis, ρ is considered to be a *false discovery* or equivalently, a *type-1 error*. Any pattern ρ that is not "discovered" and does not satisfy the null hypothesis is called a *type-2 error*.

The techniques presented herein allow arbitrary statistical hypothesis tests to be applied during pattern discovery in a manner that allows the user to place a strict upper bound on the risk of any pattern being accepted that is a false discovery.

4 Previous Approaches to Avoiding False Discoveries in Pattern Discovery

The original formulation of association rule discovery sought all rules that satisfied user specified constraints on minimum-support and minimum-confidence, with the justification that "confidence is a measure of the rule's strength" while "support corresponds to statistical significance" (Agrawal et al. 1993). From this it might be inferred that support and confidence are measures used to select rules that represent strong positive associations in the distribution from which the sample data are drawn. In our framework this means the ϕ should be regarded as *strong positive association* rather than the explicit minimum-support and confidence constraints.

There is evidence this approach can be successful at avoiding false discoveries when applied to the type of sparse transaction data for which it was designed (Megiddo and Srikant 1998). However, pattern discovery is widely applied to many other forms of data and, as is shown below in Sect. 6, the enforcement of a minimum-support constraint both fails to ensure statistical significance in many contexts and can also lead to many significant patterns being overlooked. Further, different data analysis tasks will call for different sets of constraints ϕ, and the support-confidence framework provides little scope for adjusting to arbitrary such constraints.

There have been numerous proposals to identify and discard association rules that are unlikely to be of interest. These include constraints on minimum lift (International Business Machines 1996) leverage (Piatetsky-Shapiro 1991) and improvement (Bayardo et al. 2000). They also include the identification and rejection of redundant (Bastide et al. 2000; Zaki 2000) and derivable (Calders and Goethals 2002) rules.

Redundant rules are those such as $\{pregnant, female\} \rightarrow oedema$ that include items in the antecedent that are entailed by the other elements of the antecedent, as is the case with *pregnant* entailing *female*. Redundant rule constraints discard rules $x \rightarrow y$ for which $\exists z \in x : support(x \rightarrow y) = support(x - z \rightarrow y)$.

A *minimum improvement* constraint (Bayardo et al. 2000) is more powerful than a redundant rules constraint. The improvement of rule $x \rightarrow y$ is defined as

$$improvement(x \rightarrow y) = confidence(x \rightarrow y) - \max_{z \subset x}(confidence(z \rightarrow y)). \qquad (1)$$

We use the term *productive* to denote rules with *improvement* greater than zero.

The improvement of a redundant rule cannot be greater than 0.0, and hence a constraint that rules must be productive will discard all redundant rules. In addition to redundant rules, a constraint that rules must be productive can discard rules that include items in the antecedent that are independent of the consequent given the remaining items in the antecedent.

However, these approaches make no assessment of the likelihood that an observed pattern is a chance artifact of the sample data rather than a consistent pattern in the distribution from which the data are drawn. That is, no assessment is made of the likelihood that an apparently redundant rule is truly redundant, or whether a rule with positive improvement on the sample data does not have negative improvement in the distribution from which these data are drawn.

Some systems seek to make such an assessment by applying a statistical significance test before accepting a pattern. Examples include Brin et al.'s (1997) correlation rules, Liu et al.'s (1999) pruning technique, version 1.3 of *Magnum Opus* (Webb 2002) and Zhang et al.'s (2004) significant statistical quantitative rules. As indicated in the introduction, this approach has high risk of type-1 error, that is, of erroneously accepting spurious patterns.

Some systems, such as *STUCCO* (Bay and Pazzani 2001), apply a statistical test with a correction for multiple comparisons. This system seeks to control type-1 error by applying a Bonferroni-like adjustment to the critical value used with a statistical test during pattern discovery. Rather than dividing the base critical value, α, by the number of tests performed, they apply tests in batches (tests for all rules with a given number n of conditions in the antecedent), and use a critical value of $\alpha/(2^n \times m)$, where m is the number of rules tested in batch n.

The application of either a standard Bonferroni adjustment for the total number of rules tested or this modified adjustment is not statistically sound because the number of rules tested m is less than the number of rules considered. This is because the tests are only applied to rules that pass all other constraints, including a constraint that the joint frequency of the antecedent and consequent (the support) must differ from the expected frequency ($\mathit{coverage}(X \rightarrow Y) \times \mathit{support}(\emptyset \rightarrow Y)$) by a user-specified constant. This has the effect of selecting rules that are most likely to pass the statistical test. However, it does not reduce the risk of encountering those rules by chance during the exploration of a large number of alternatives. To maintain strict control over the risk of type-1 error, the critical value should be adjusted not by the number of times the statistical test is applied, but rather by the number of patterns from which those to be tested are selected.

An alternative approach has been suggested by Megiddo and Srikant (1998). Given a transaction dataset D, repeated tests are performed on datasets D'_i that sample from a synthetic distribution Θ'. This distribution instantiates the null hypothesis with respect to which a hypothesis test evaluates ϕ, while retaining all other relevant properties of the original data. The data mining system can then be applied to the D'_i. As the null hypothesis is enforced, any patterns discovered represent type-1 error. The parameters of the system are manipulated so that less than critical value α runs produce any rules. Then, if the system is subsequently run on the original data with those settings and produces rules, the probability of obtaining such rules under the null hypothesis is less than α and hence the rules may be accepted. As a single statistical test is applied to all rules produced by a run of the system, rather than individual tests being applied to each rule in isolation, the problem of multiple comparisons is avoided.

In Megiddo and Srikant's (1998) technique the data are randomly generated from the set of items I of the original data such that for each record $R' \in D'$ and item $i \in I$, $P(i \in R') = P(i \in R)$, where $P(i \in R)$ is shorthand for the probability that an $R \in D$ drawn at random will contain i. This process generates data drawn from a distribution in which each $i \in I$ has the same frequency as in D, but in which each i is independent of the other.

This is a very powerful approach. However, it requires that it be possible to create a single synthetic distribution Θ' that instantiates the null hypothesis with respect to all hypothesis tests that are required. This is feasible with respect to a test that the elements of a rule are not all independent of one another. The null hypothesis is then that all conditions in the rule are independent. An appropriate Θ' can be generated using Megiddo and Srikant's (1998) technique or, for attribute-value data, by Monte Carlo sampling whereby the attribute values are assigned randomly to the data records while retaining the original frequency of each value. Hence, all characteristics of the data will be retained except that the value of one attribute for a record will not influence the values of the other attributes for that record.

It is also feasible with respect to a test that the consequent is not independent of the antecedent, if the consequent is constrained to a single prespecified target variable. In this case the null hypothesis is that the consequent is independent of the antecedent. An appropriate Θ' can be generated by Monte Carlo sampling whereby only the target variable is randomized in the data records while retaining the original frequency of each value.

However, there are other hypothesis tests for which it does not appear possible to apply this approach. For example, it does not appear possible to produce a synthetic distribution that establishes a suitable null hypothesis to test that the support of each rule meets a minimum support constraint with respect to the distribution Θ from which the sample data D has been drawn.

Another example is that it will often be desirable to test whether a pattern contains any condition c such that c is independent of all other conditions to which the pattern refers. This can be important to assess with association rules. If there is a rule $X \rightarrow Y$ with confidence τ, then for any condition c that is independent of both X and Y, $X \cup \{c\} \rightarrow Y$ will also have confidence τ (except insofar as sampling effects may vary the observed confidence for a given data set). Hence, depending upon the specific constraints in use, for every rule $X \rightarrow Y$ representing an interaction between conditions that is of interest, a spurious rule may be generated for every combination of conditions that is independent of X and Y. For example, consider the hypothetical example of a rule *tea* → *coffee* that captures an interdependence between *tea* and *coffee*. As *tea* and *coffee* are not independent it follows that

$$P(\mathit{coffee} \,|\, \mathit{tea}) \neq P(\mathit{coffee}) \tag{2}$$

Now consider any third factor, say *footwear*, that is independent of both *tea* and *coffee* (both individually and in combination). As *footware* is independent of both *tea* and *coffee* it follows that

$$P(\mathit{coffee} \,|\, \mathit{tea}, \mathit{footware}) = P(\mathit{coffee} \,|\, \mathit{tea}). \tag{3}$$

From (2) and (3) it follows that

$$P(\mathit{coffee} \,|\, \mathit{tea}, \mathit{footware}) \neq P(\mathit{coffee}). \tag{4}$$

Thus the antecedent of *tea* & *footware* → *coffee* is not independent of the consequent, even though *footware* is independent of both *tea* and *coffee*.

It may be desirable to apply a hypothesis test to each pattern to guard against such a possibility. However, for a pattern relating to more than two conditions there is no single manipulation of the data that can establish the null hypothesis. If the pattern contains three conditions a, b and c, then the null hypothesis is that any one of a, b or c is independent of the other two.

A related line of research is the use of shrinkage estimates, or Bayesian smoothing, to provide conservative estimates of the true probability of a conjunction of conditions. Examples of this approach include DuMouchel and Pregibon (2001) Empirical Bayes Screening, *Magnum Opus*'s m-estimates (Webb 2005) and Scheffer (1995) Bayesian Frequency Correction. These approaches can be very effective at reducing the overestimates of measures such as support or confidence that can occur for rules with low support, and hence reduce type-1 error with respect to minimum support or confidence (or similar) constraints. However, they do not provide a general mechanism for applying hypothesis tests to discovered rules, and do not take account of the search space size.

So, to summarize, existing techniques either have high risks of type-1 error, or are only applicable to a limited range of hypothesis tests. This is clearly of concern. In many pattern discovery contexts both type-1 and type-2 error are undesirable. We do not want to overlook knowledge-nuggets. Nor, however, do we want to present as knowledge-nuggets patterns that are actually knowledge-dross. My previous response to this problem has been that we should accept that pattern discovery is inherently statistically unsound. In order to avoid high levels of type-2 error we should not apply a correction for multiple comparisons. Rather, we should expect type-1 error and always seek to assess independently the quality of the rules that are discovered. In hypothesis testing terminology, we should view pattern discovery as a hypothesis generation process and then seek independently to test those hypotheses.

However, this line of reasoning leads naturally to the question of whether it is possible to automate the process of independent hypothesis testing. It turns out that it is indeed straightforward to do so. As outlined in the introduction, one need only divide the available data into *exploratory* and *holdout* sets, use the former for pattern discovery and then use the latter to assess the soundness of the rules so discovered.

Further, it appears to have previously been presumed that applying a statistical correction for the number of patterns in the search spaces typically considered during pattern discovery would result in such low critical values that no patterns would be accepted. This turns out to be untrue.

While the use of holdout assessment and of a Bonferroni adjustment for the size of the search space in pattern discovery are both conceptually simple, there are a number of non-trivial technical issues to be addressed. The next section describes in detail these statistically sound approaches to pattern discovery.

5 Statistically Sound Pattern Discovery

We seek the capacity to apply arbitrary hypothesis tests to all patterns found by an arbitrary pattern discovery process while maintaining strict control over the experimentwise risk of type-1 error. In this section we first describe in outline two techniques that provide this capacity. We then provide detailed discussions of a number of specific issues that arise in practice when seeking to apply these techniques.

5.1 Proposed Techniques

Before explaining our techniques we need to provide a short overview of relevant established statistical techniques for controlling the risk of type-1 error in the context of multiple hypothesis tests.

Each hypothesis test returns a p-value, the probability that the observed data or more extreme would be obtained if the null hypothesis were true. In the pattern discovery context the null hypothesis will usually correspond to the pattern being spurious or a chance artifact of no real interest. Thus, the pattern is accepted only if the null hypothesis is rejected. In classical hypothesis testing, the null hypothesis is rejected if the p-value is no greater than a predefined *critical value*, commonly 0.05. However, as outlined in the Introduction, if many such hypothesis tests are performed, it becomes a near certainty that some will be accepted in error.

The classical approach rectifying this problem is the Bonferroni adjustment (Shaffer 1995) that bounds the risk of any type-1 error at α when performing n hypothesis tests by using a critical value $\kappa = \alpha/n$ for each hypothesis test. This turns out to be needlessly strict, however. The Holm procedure (Holm 1979) is more powerful than the Bonferroni adjustment while still guaranteeing that the risk of any type-1 error is no more than α. The procedure takes the p-values from the n hypothesis tests and orders them from the lowest, p_1, to the highest, p_n. It then sets $\kappa = \max(p_i : \forall 1 \leq j \leq i, p_j \leq \alpha/(n-j+1))$, that is, the highest p_i such that all p_j up to and including p_i pass the test $p_j \leq \alpha/(n-j+1)$. If no p_i satisfies this test then $\kappa = 0.0$.

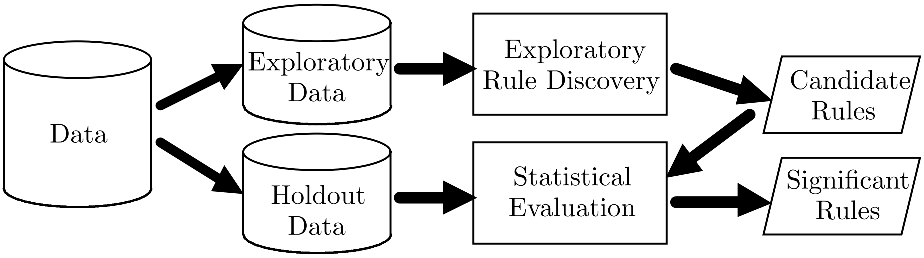

Fig. 1 The holdout evaluation process. Reproduced from Webb (2006)

The Bonferroni and Holm procedures both make no assumptions about whether the hypothesis tests are correlated. That is, both tests guarantee that the experimentwise error will be no more than α irrespective of whether there are correlations between the hypothesis tests. This property is very important in the context of pattern discovery, as often many of the patterns considered will be related to one another and some statistical tests will be mutually exclusive, which is the most extreme form of correlation between tests that a multiple testing procedure may encounter. For example, a hypothesis that *pregnant* and *gender* = *female* are positively correlated will be mutually exclusive with a hypothesis that *pregnant* and *gender* = *male* are positively correlated. While there are more powerful multiple testing procedures than the Holm procedure (Shaffer 1995), most make assumptions about the possible forms of correlation between the tests and hence fail to strictly bound the experimentwise risk of type-1 error in the pattern discovery context.

Two approaches present themselves to using these techniques for pattern discovery. Under the *direct adjustment* approach one determines the size of the search space, s and simply applies the adjustment directly, requiring all patterns to pass any statistical tests at the adjusted critical value α/s. Under the *holdout* approach, one first generates patterns without regard for the problem of multiple testing and then applies statistical tests on the candidate patterns so generated, ensuring that the tests are evaluated with respect to data that were not employed in the initial pattern generation process. This can be achieved by dividing the data into exploratory and holdout sets. A constrained number of patterns are found by analysis of the exploratory data. These are then evaluated against the holdout data through the application of statistical hypothesis tests using a correction for the number of patterns found, such as the Bonferroni adjustment. Both approaches may be applied with any statistical significance tests that should be desired. The holdout approach is illustrated in Fig. 1.

As both approaches use well established statistical procedures, there should be little doubt about their capacity to strictly control the risk of experimentwise type-1 error. What is less clear is the relative power of each technique. The power of the technique is the probability that a pattern will be discovered if it is a true pattern. Unfortunately, analysis of statistical power is only possible with respect to a hypothesis with a known effect-size. The stronger the effect and the greater the quantity of available data, the greater the power of a typical statistic.

Each technique has features that may provide an advantage relative to the other with respect to power. The direct adjustment approach will obtain an advantage from the use of all data for both pattern discovery and statistical evaluation. On the other hand, the holdout evaluation approach will obtain an advantage as the corrections to the critical value will usually be many orders of magnitude smaller and it will be possible to apply the more powerful Holm procedure. It does not appear feasible to apply the Holm procedure to the direct adjustment approach, as it requires that all significance levels be discovered before the adjusted critical value can be determined. If the search space is 10^{20}, for example, it will not be feasible to explicitly investigate all patterns in order to discover their significance levels. Further, the difference in adjustment will often be so small as to make little impact. For example, suppose 1 million true patterns were found in a search space of size 10^{20}. Rather than dividing α by 10^{20} for the 1 millionth pattern, the Holm procedure would divide it by 10^{20}–10^{6}, a very small variation from the straightforward Bonferroni adjustment.

Bounding the size of the search space will typically involve two conflicting pressures with respect to the power of the direct adjustment approach. A smaller search space is likely to reduce power due to excluding some true patterns from the space of possibilities that are considered. On the other hand, however, the smaller the search space the smaller the Bonferroni adjustment and hence the greater the power of the analysis with respect to those patterns within the search space.

The same factors will apply to the holdout approach, as a larger search space will usually generate more candidates and hence reduce the power of the Holm procedure. However, it is possible that the rate at which the number of candidates will increase will be lower than the rate at which the search space increases as more complex patterns often have weaker effect and hence less likelihood of being selected as a candidate. Hence, we hypothesize that increasing the size of the search space will tend to be more beneficial for the holdout than the direct adjustment approach.

The following sub-sections examine in turn the issues that must be addressed to operationalize these schemes.

5.2 Correcting for Multiple Comparisons with Respect to a Single Pattern

Sometimes it will be desirable to subject each pattern to multiple hypothesis tests, with n null hypotheses $H_{0,1}, H_{0,2}, \ldots, H_{0,n}$. For example, to test whether any conjunct in the antecedent of a rule is independent of the remaining conjuncts, one hypothesis test might be applied for each of the n conjuncts.

It is not necessary to adjust for multiple tests in this respect. This is because although multiple hypotheses are being tested, the outcome sought is simply an evaluation of whether any of the hypotheses is violated without seeking to identify which one. If the pattern is rejected when any of the hypothesis tests fails, then there is no need to adjust the critical value applied for the individual tests. In effect, a new null hypothesis is evaluated, the disjunction of $H_{0,1}, H_{0,2}, \ldots, H_{0,n}$. If each test is applied with critical value α and only n of the hypotheses are not correct then the risk of incorrectly accepting the pattern is the risk of all the single hypothesis tests that test the n hypotheses simultaneously failing to reject them while none of the other tests incorrectly rejects its hypothesis. This risk must be less than α as the probability of a conjunction of events cannot exceed the probability of any of the individual events. To illustrate this point, consider a rule with two conditions in its antecedent, $\{a, b\} \rightarrow c$. We wish to apply three hypothesis tests, one to evaluate each of (i) $\mathit{confidence}(\{a, b\} \rightarrow c) > \mathit{confidence}(\{a\} \rightarrow c)$, (ii) $\mathit{confidence}(\{a, b\} \rightarrow c) > \mathit{confidence}(\{b\} \rightarrow c)$ and (iii) $\mathit{confidence}(\{a, b\} \rightarrow c) > \mathit{confidence}(\{\} \rightarrow c)$. Suppose that (i) is true and (ii) and (iii) are false. If all three hypothesis tests are applied with a critical value of α, the probability that the rule will be accepted is the probability that all three will be accepted which is the probability of a true positive for (i) co-occurring with a false positive for both (ii) and (iii). Thus, when multiple hypotheses are tested to assess only whether any does not hold, it is the risk of type-2 rather than type-1 error that is increased by the multiple hypothesis testing.

For this reason, an adjustment is applied only with respect to the number of patterns to be assessed, not with respect to the total number of hypothesis tests that are to be performed.

5.3 Anticipating the Holdout Test

Under the holdout evaluation approach, we intend to subject the patterns discovered to statistical evaluation on a holdout set. In consequence, it is desirable to minimize the number of patterns so tested, as this will minimize the adjustment that is applied to the significance test.

To this end it may be desirable to anticipate patterns that are unlikely to pass the hypothesis test and to discard them prior to the holdout testing phase. We will use the generic term *filtering* for such a strategy. The application of such a filter can be regarded as the imposition of additional constraints to be applied at rule discovery time.

Filtering will inevitably involve a trade-off. The more patterns discarded prior to holdout testing, the lower the adjustment that need be applied, and hence the greater the number of remaining patterns that are likely to pass. However, for every such pattern discarded prior to holdout testing, there will be a risk that had it been subjected to the holdout test it would have passed. Clearly, it is desirable to find a process for anticipating the hypothesis test that results in more patterns being accepted that would otherwise have failed than patterns discarded that would otherwise have passed.

One approach to this is to apply the hypothesis tests during the pattern discovery process as well as during holdout evaluation. One possibility is to apply such a test with the raw α, that is, without using the Bonferroni adjustment. This might be expected to balance the risks of type-1 and type-2 error, as discussed in the introduction. However, given that α will be adjusted at the statistical evaluation phase, a case can be made for applying this adjustment also at the exploratory phase. If the strength of an interaction is not sufficient to pass such a test with respect to the exploratory data then the probability that it will pass it with respect to the holdout data must be low. Note that this approach does not lead to statistically sound rule discovery during the exploratory discovery phase, as the adjustment being applied relates to the number of patterns to be subjected to holdout evaluation, not to the number of patterns considered during pattern discovery. Typically the latter will be orders of magnitude greater than the former as many patterns will fail the test and hence be rejected.

There can be no single ‘correct’ solution to this dilemma. Any strategy that is adopted will imply a trade-off between the risks of type-1 and type-2 error, of accepting spurious patterns or of failing to discover potentially valuable patterns. The best solution will depend on the relative costs of type-1 and type-2 error for a specific application. A stronger filter is likely to increase type-2 error by discarding more potential patterns before holdout evaluation, whereas a weaker filter is likely to increase type-1 error by increasing the number of patterns subjected to holdout evaluation and hence decreasing the critical value employed in the hypothesis tests.

5.4 Determining the Size of the Search Space

Before a Bonferroni adjustment may be applied directly during pattern discovery, it is necessary to determine the size of the search space. The size of the search space will clearly depend on the type of pattern being explored. In the current paper we consider rules.

For market-basket data it is straightforward to determine the size of the search space. Recall that m is the total number of items and assume that the antecedent X must contain at least one item and that there is an upper bound $X_{\max}$ on the number of items it may contain.

There are m possible values for y, and for each y value there are $m-1$ items from which up to $X_{\max}$ X values are selected.

$$s = m \times \sum_{i=1}^{X_{\max}} C_i^{m-1} \tag{5}$$

where C_i^{m-1} is the number of combinations containing i out of $m-1$ items. So, for example, with the Retail dataset, used below, the number of items is 16 470 and hence with X limited to no more than 5 items the size of the rule space is 1.66×10^{23}.

For attribute-value data the situation is a little more complex, as no rule containing more than one item for a single attribute can be productive. Examples of such rules include $\{gender = male, gender = female\} \rightarrow occupation = dataminer$ and $gender = male, occupation = dataminer\} \rightarrow gender = female$. In consequence, all such rules should be excluded from the calculation. To calculate the total s we must first be able to calculate the number of combinations of values of a given subset of i attributes, *atts*. To do so we first order the attributes in arbitrary order from 1 to i and refer to the individual attributes using this order as $att_1 \ldots att_i$. We use intermediate values $c_{att,j,k}$ that each represent the total number of combinations of up to j items where items contain only values for attributes $att_1 \ldots att_k$.

$$c_{att,j,k} = \begin{cases} \#att_k, & j = 1,\ k = 1, \\ 0, & j > 1,\ k = 1, \\ c_{att,1,k-1} + \#att_k, & j = 1,\ k > 1, \\ c_{att,j,k-1} + \#att_k \times c_{att,j-1,k-1}, & \text{otherwise} \end{cases} \tag{6}$$

where $\#att_j$ is the number of values for attribute att_j.

$$s = \sum_{z \in a} \left(\#z \times \sum_{j=1}^{X_{\max}} c_{a-z,m,j} \right). \tag{7}$$

5.5 Computational Considerations

The computational impact of each of the two strategies will vary greatly depending upon the base pattern discovery algorithm with which they are employed and the properties of the search space for a given application.

The holdout approach will speed-up the pattern discovery process by reducing the amount of data that is considered. This saving will be offset, however, by the addition of the holdout process. The computational requirements of the holdout process will depend on the number of patterns to be assessed and the complexity of the statistical tests that are employed.

The direct adjustment approach will have an additional computational burden resulting from the application of a statistical test during search. The amount of additional computation this requires will also depend on the requirements of the specific tests being employed. Note that the tests need only be applied if a potential pattern passes all other constraints, so it is conceivable that no more statistical tests will be required under this approach than under the holdout approach.

The direct adjustment approach may also result in some computational savings through the introduction of new opportunities to prune the search space. Within the KORD algorithm (Webb and Zhang 2005) that underlies the *Magnum Opus* software used in our experiments, some pruning of the search space can be obtained by considering at a node in the search space whether any specialization of the current rule could pass the significance test. For the statistical tests employed in the current work, this can be determined by assessing whether a rule with the same support as the current rule but with confidence = 1.0 would pass the test. As this represents the greatest power that a specialization of the current rule could possibly have, if such a rule would not pass the test then no specialization of the current rule will.

6 Empirical Investigations

The techniques that we are proposing all use standard statistical procedures and hence there should be no question as to whether they will provide strict control over type-1 error (Shaffer 1995). It is important, however, to assess whether previous techniques actually suffer in practice the high risk of type-1 error suggested by the analysis above and the relative power of each of the two techniques. This section presents a number of experiments that investigate these issues. The current section provides an overview of the experimental outcomes.

Section 6.1 investigates the susceptibility of pattern discovery techniques to making false discoveries and the capacity of the proposed techniques to prevent them. Section 6.2 explores the factors that affect the power of each of the approaches. Section 6.3 investigates the need for and efficacy of the techniques with real-world data. Section 6.4 assesses the computational efficiency of the techniques when applied to real-world data. Finally, Sect. 6.5 looks at the number of statistically significant patterns that are embodied in a number of real-world data sets.

All experiments were carried out with the *Magnum Opus* (Webb 2005) pattern discovery system. *Magnum Opus* was used with its default settings unless otherwise specified. By default it finds the 100 rules that maximize leverage with respect to the exploratory data within any other user specified constraints. By default the rule antecedents consist of a conjunction of one to four attribute-value tests (tests of the form $x = v$, where x is an attribute and v is a value of that attribute) and the consequents consist of a single attribute-value test.

The experiments use the Fisher exact test for productive rules, described in Appendix 1.

6.1 Establishing the Need

Experiment 1 investigated the need for statistical control over the risk of Type-1 error during pattern discovery. Random data were generated containing 10 000 records, each comprising values for 100 binary variables. The two values for each variable were equiprobable. All variables in this data were independent of each other and hence any patterns discovered must be false discoveries. 100 such data sets were generated. *Magnum Opus* was applied to each data set using each of the following set of parameters. For all settings the maximum antecedent size was set to the default value of 4.

Non-redundant: find the 1000 non-redundant rules with the highest leverage.

Productive: find the 1000 productive rules with the highest leverage.

Significance=0.05: find the 1000 rules with the highest leverage that pass a significance test at the 0.05 significance level.

Direct adjustment: find the 1000 rules that pass a significance test at the 4.06×10^{-12} significance level that results from applying a Bonferroni correction to a raw significance level of 0.05 with a search space of 1.23×10^{10} rules.

Non-redundant+holdout: find the 1000 non-redundant rules with the highest leverage from half the data and then validate the rules using the remaining holdout data.

Table 1 Support, confidence and leverage of rules found from random data

	Support			Confidence			Leverage		
Treatment	min	mean	max	min	mean	max	min	mean	max
Non-redundant	320	950	2688	0.490	0.537	0.618	0.0044	0.0050	0.0116
Productive	320	950	2688	0.490	0.537	0.618	0.0044	0.0050	0.0116
Sig. 0.05	320	860	2688	0.489	0.537	0.618	0.0042	0.0050	0.0116

Productive+holdout: find the 1000 productive rules with the highest leverage from half the data and then validate the rules using the remaining holdout data.

Unadjusted+holdout: find the 1000 rules with the highest leverage that pass a significance test at the 0.05 significance level from half the data and then validate the rules using the remaining holdout data.

The non-redundant, productive and significance = 0.05 treatments all resulted in discovery of 1000 rules for every dataset. Table 1 shows the minimum, mean and maximum support, confidence and leverage for each of these treatments. As can be seen, some rules had substantial support, confidence and leverage. For this task there were almost no differences in the rules discovered by the non-redundant and productive approaches because almost all rules with the highest leverage were productive on the sample data.

These results illustrate the high risk of false discoveries that simple theoretical analysis shows must exist unless appropriate allowance is made for the multiple-tests problem.

Neither the direct adjustment nor any of the three holdout approaches found any rules. Given that all four of these approaches controlled the experimentwise risk of false discoveries at the 0.05 level, it would have been reasonable to expect false discoveries for up to 5 of the 100 datasets under each of these treatments. However, the Bonferroni adjustment applied by the direct-adjustment approach is known to be very conservative. Further, many of the null hypotheses tested are strongly positively correlated as they relate to correlations between the same sets of items. This greatly reduces the experimentwise risk of false discoveries, but means that if any false discoveries occur, there are likely to be many.

These results demonstrate that both the direct-adjustment and holdout approaches can prevent false discoveries.

6.2 Assessing the Relative Power of the Approaches

As discussed in Sect. 5, the power of a statistical analysis is the probability that it will reject the null hypothesis if the null hypothesis is false. Recast in the pattern discovery context, this translates into the probability that a true pattern will be discovered. The relative power of the two approaches is going to depend upon many factors for any given application, including the size of the search space, the quantity of data, and the number of patterns found during the exploratory stage of the holdout evaluation approach.

6.2.1 Experiments

To explore these issues we investigated manipulating each of these factors in two contexts, one where increasing the search space provided access to more true patterns and the other where it did not. We manipulated the size of the search space by changing the maximum number of items in the antecedent.

Experiment 2 established a situation where increases to the size of the search space do not provide access to additional true patterns. We generated random data for ten pairs of binary variables x_0 and y_0 through to x_9 and y_9. Each x_i was generated at random with each value being equiprobable. The probability of $y_i = 1$ was $1.0 - i \times 0.05$ if $x_i = 1$, $0.0 + i \times 0.05$ otherwise. This gives rise to a total of forty valid (productive) rules of the forms $x_i = 0 \to y_i = 0$, $x_i = 1 \to y_i = 1$, $y_i = 0 \to x_i = 0$ and $y_i = 1 \to x_i = 1$. These rules differ greatly in the ease with which they may be detected, those for x_0 and y_0 representing very strong correlations and being straightforward to detect and those for x_9 and y_9 representing relatively weak correlations and being correspondingly difficult to detect. As all valid rules have only one item in the antecedent, any increase in the maximum allowed size of the antecedent serves to increase the search space without increasing the number of valid rules in the search space. For all other combinations of parameters, we varied the maximum allowed antecedent size through each size from 1 through to 5.

We varied the quantity of data by generating datasets of the following sizes, 250, 500, 1000, 2000, 4000, 8000 and 16 000. These sizes were selected by experimentation as those providing the most interesting variations in performance. We generated 100 random datasets at each of these sizes. Each larger dataset was generated by appending additional data onto the immediately smaller dataset.

For the holdout treatments, half of each dataset was used for exploration and the remaining half for statistical evaluation. For holdout evaluation, we sought alternatively the 100, 1000 and 10 000 rules with the highest leverage during the exploratory stage. We also varied whether a statistical test was applied during the exploratory stage. The *None* treatments found the highest leverage rules without regard for whether they were productive. The *Sig* treatments found the highest leverage rules out of those that passed the Fisher exact test for productivity at the 0.05 level, without any adjustment for the number of patterns considered.

The *Direct* treatments applied a significance test during search with the critical value adjusted to allow for the size of the search space.

For the second experiment 15 binary variables were created, a, b, c, d, e and $x_0, x_1, \ldots x_9$. All variable values were randomly generated independently of one another, with each value equiprobable, except for e for which the probability of value 1 was 0.80 if all of a, b, c and d were 1 and 0.48 otherwise.

This generates a total of 83 productive rules, those with

- one or more of $a = 1, b = 1, c = 1$ and $d = 1$ in the antecedent and $e = 1$ in the consequent
- $e = 1$ and zero or more of $a = 1$, $b = 1$, $c = 1$ and $d = 1$ in the antecedent and one of $a = 1$, $b = 1$, $c = 1$ and $d = 1$ in the consequent,
- exactly one of $a = 0$, $b = 0$, $c = 0$ and $d = 0$ in the antecedent and $e = 0$ in the consequent, and
- $e = 0$ and zero or more of $a = 1$, $b = 1$, $c = 1$ and $d = 1$ in the antecedent and one of $a = 0$, $b = 0$, $c = 0$ and $d = 0$ in the consequent.

Note that this meant that each increase in the size of the search space but the last increased the number of productive rules that could be found.

Identical treatments were applied to those for Experiment 2 except that datasets sizes were varied from 1000 to 64 000, as larger datasets were required to find the more subtle patterns.

6.2.2 Results

We refer to the application of one of the 245 treatments to a single dataset as a *run*. Tables 4 and 5 in Appendix 2 present for each treatment the mean numbers of true and false discoveries per run, together with the number of runs for which any false discoveries occurred. This latter value corresponds to experimentwise error, whereas the mean number of false discoveries indicates the total number of patterns falsely discovered, rather than the number of runs in which false discoveries occurred. Figures 2 and 3 plot these outcomes against variations in the maximum antecedent size and the quantity of data available. We do not perform statistical evaluation of comparative performance as these are synthetic data and whether differences in performance reach statistical significance or not is a greater reflection on the data generated than on the alternative techniques. These experiments are not designed to test hypotheses, as the theoretical properties of the techniques we are employing are already well understood. Rather, they are designed to provide insight into the magnitude of the expected effects in practice.

The holdout and direct adjustment approaches both cap the risk of any false discovery at 0.05. For Experiment 2, out of the 245 treatments considered, just 2 had 6 runs with false discoveries. For Experiment 3, 2 of the 245 treatments had 7 runs with false discoveries and 1 had 6. Each outcome represents an aggregate result for 100 runs. If the probability of each event is 0.05, the probability is 0.23 of obtaining 7 or more events out of 100 trials, 0.38 of obtaining 6 or more out of 100 and 0.56 of 5 or more. For Experiment 2, out of the 24 500 runs performed, only 124 resulted in any false discoveries, an experimentwise error rate of approximately 0.005. For Experiment 3, only 238 out of 24 500 runs resulted in false discoveries, an experimentwise error rate of approximately 0.010. However, it is noteworthy that when an experiment resulted in any error it usually resulted in more than one. For Experiment 2 there were a total of 293 false discoveries, representing an average of 2.4 false discoveries for every run for which there were any false discoveries. For Experiment 3 there were 507 false discoveries, representing an average of 4.2 false discoveries for every run with any false discoveries. This effect is due to the null hypotheses for different rules being closely related to one another such that if one passes then another is likely to. Despite this effect, the total false discovery rate for Experiment 2 is 293/814 954, which is less than 0.001 and for Experiment 3 it is 507/460 800, which is less than 0.002.

Due to the very low experimentwise false discovery rates, the differences in false discoveries between treatments should not be considered meaningful, as they could readily be accounted for by the variance in results. The ridges observed in Figs. 2 and 3 at specific data sizes, such as that for Experiment 2 None 1000 at data size 8000, can be explained by the use of the same 100 datasets for each antecedent size and do not necessarily represent an effect relating to the specific data set size. If a false discovery with one item in the antecedent received strong support from a specific dataset, it would appear as a false discovery at all maximum antecedent size levels.

For Experiment 2, all treatments found all 40 productive rules on all 100 runs when the antecedent was restricted to 1 item and the dataset contained 16 000 or (except for None 1000 and None 10 000) 8000 records. For all treatments the true discovery rate declined as the search space was increased to include more false patterns. For the direct-adjustment technique this is because the adjusted critical value decreases as the search space increases, making it more difficult for a rule to be accepted. For the holdout techniques this is because of two effects. As the search space increases, the risk increases that spurious rules will be discovered with higher apparent leverage than the true rules and hence the true rules will be excluded from the set of candidate rules. Where a significance test is used as a filter, when the search space is small it will often be the case that fewer than the allowed maximum number of rules are found, and hence the adjustment applied during holdout evaluation is smaller and more patterns are able to pass the holdout test.

No treatment consistently found all 83 rules for Experiment 3, but it is clear that given sufficient data all would. For most conditions the number of true discoveries decreased when

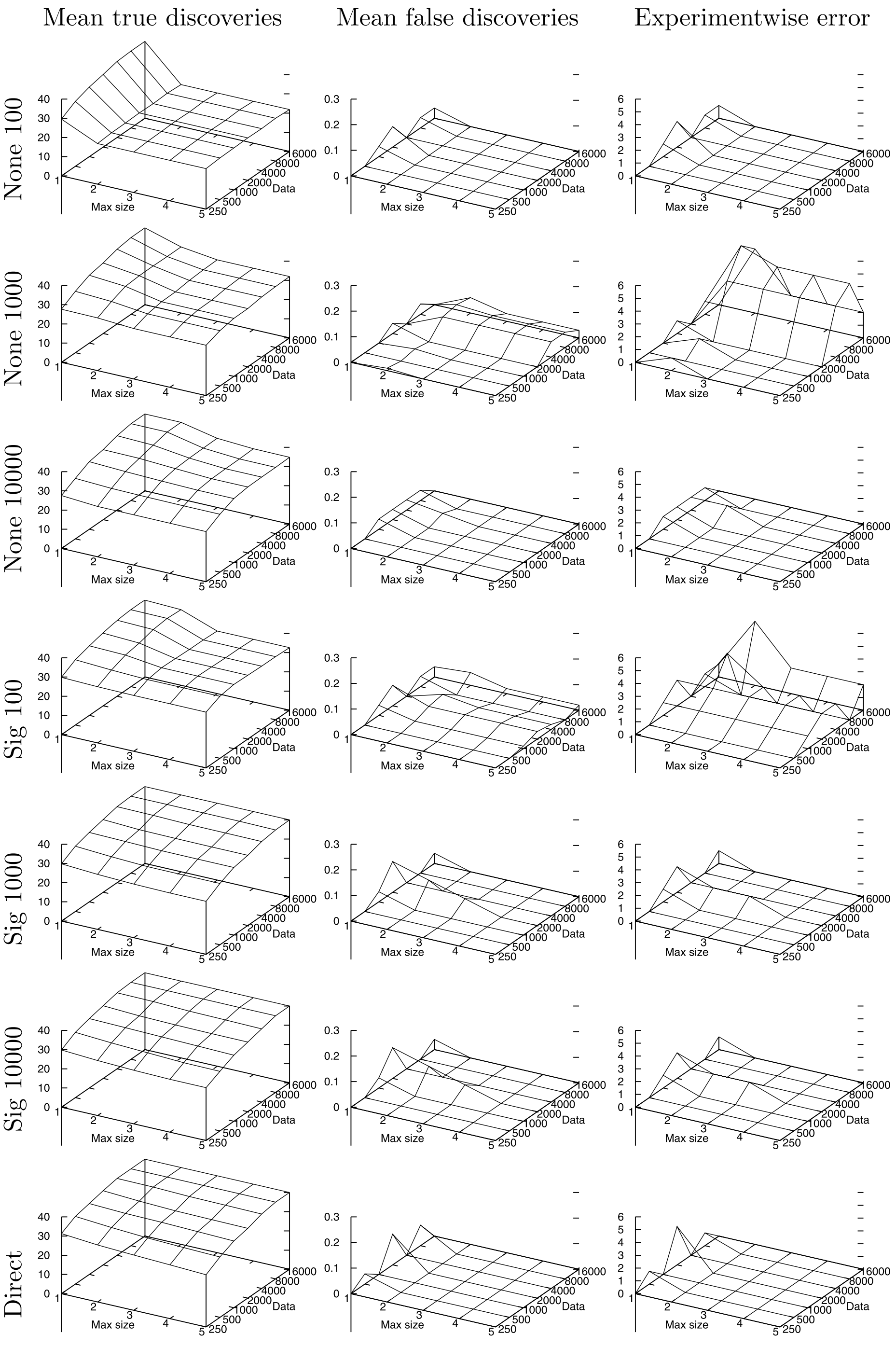

Fig. 2 Plots of results for Experiment 2

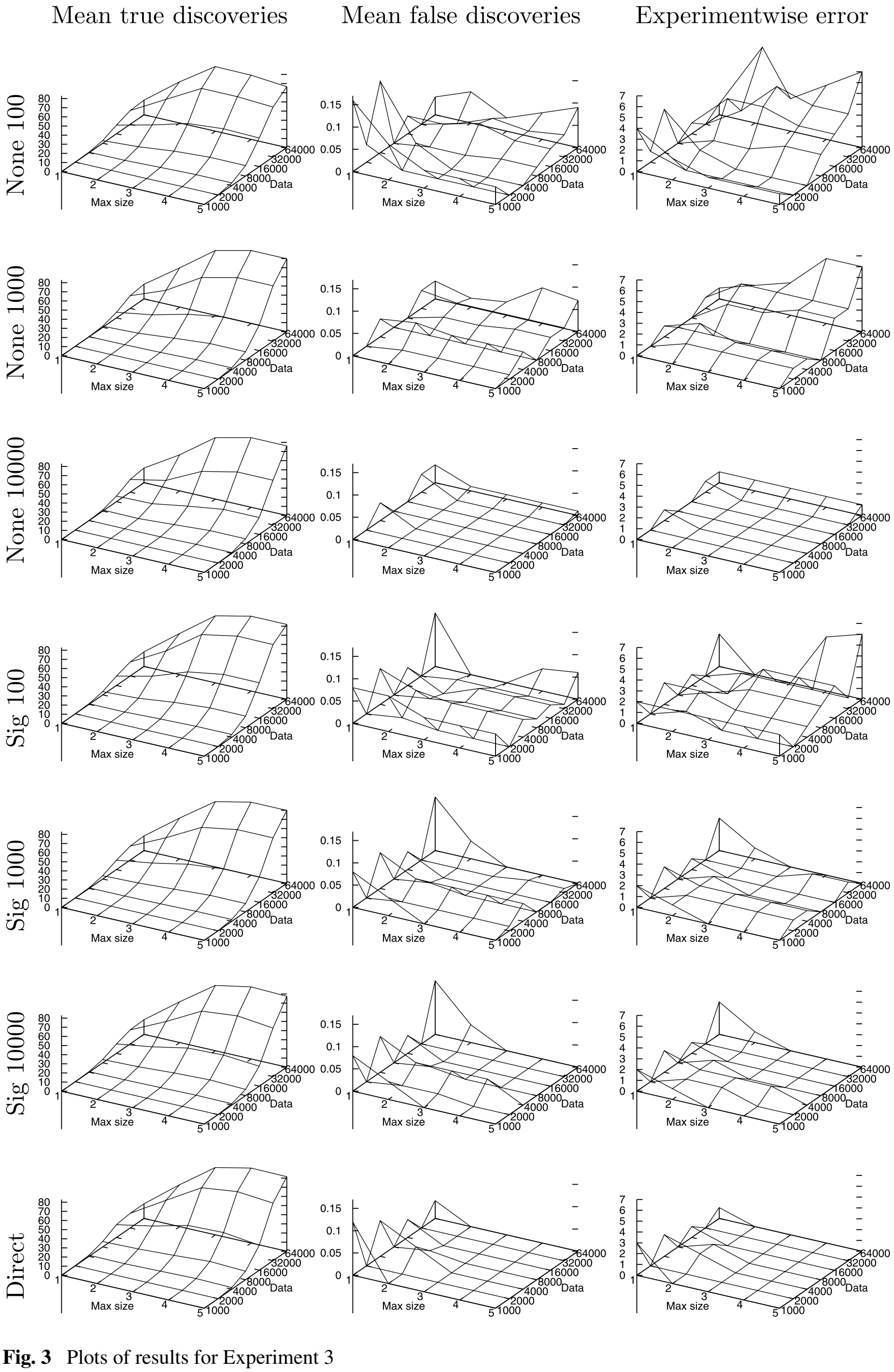

Fig. 3 Plots of results for Experiment 3

the search space was increased to include only regions that did not contain any true discoveries (the increase from *Max Size* 4 to 5).

For all treatments across both experiments the true discovery rate increased as the amount of data increased. This is because the statistical tests are more powerful with more data and are less likely to fail. However, while the true discovery rate increased, so did the false discovery rate for the holdout approaches.

For the holdout techniques, the number of true discoveries tended to increase as the number of candidates found during the exploratory stage increased. This is because the primary reason for rules not being discovered was that they were not found during the exploratory stage rather than their being excluded during holdout evaluation. Hence the decrease in the critical value caused by the increase in the number of candidates had less effect than the increased likelihood of a true rule being included in the set of candidates.

Turning next to the effect of filtering the candidates, in these experiments the application of a statistical test to filter the candidates (the Sig treatment) is very effective at increasing the number of true rules found (relative to the None treatment). This is primarily because it increases the number of true rules in the set of candidates subjected to holdout evaluation by excluding alternative rules with higher leverage that might otherwise replace them.

For Experiment 2, comparing the direct-adjustment significance testing to the most effective of the holdout approaches (Sig 10000), the former is more effective at small data sizes but the latter is more effective at larger data sizes. While the results suggest a similar effect for Experiment 3, even with data sizes of 64 000 the direct-adjustment approach does not hold a clear advantage over Sig 100.

6.3 Experiments with Real World Data

We turn next to the issue of how the techniques perform on real-world data. The same seven treatments were used as for Experiment 1. Experiments were conducted using eight of the largest attribute-value datasets from the UCI machine learning (Newman et al. 2006) and KDD (Hettich and Bay 2006) repositories together with the BMS-WebView-1 (Zheng et al. 2001) and Retail (Brijs et al. 1999) datasets. These datasets are described in Table 2. We first found for each dataset the minimum even value for minimum-support that produced fewer than 10 000 productive rules when applied with respect to the dataset as a whole with antecedents of size up to six. These values are listed in the minsup column of Table 2. Each treatment was then applied to each dataset six times, once with each maximum limit $X_{\max}$

Table 2 Datasets

Dataset	Records	Items	Minsup	Description
BMS-WebView-1	59 602	497	60	E-commerce clickstream data
Covtype	581 012	125	359 866	Geographic forest vegetation data
IPUMS LA 99	88 443	1883	42 098	Census data
KDDCup98	52 256	4244	43 668	Mailing list profitability data
Letter Recognition	20 000	74	1304	Image recognition data
Mush	8124	127	1018	Biological data
Retail	88 162	16 470	96	Retail market-basket data
Shuttle	58 000	34	878	Space shuttle mission data
Splice Junction	3177	243	244	Gene sequence data
TICDATA 2000	5822	709	5612	Insurance policy holder data

on the size of the antecedent increased from 1 to 6. All runs used the minimum-support specified, except for the holdout treatments which only use half the data for rule discovery and for which the minimum-support was therefore halved. For the sake of computational efficiency, where the number of rules satisfying the minimum-support threshold exceeded 100 000, search was restricted to the 100 000 rules with the highest support.

6.3.1 Results

Table 6 in Appendix 2 presents the number of rules found by each technique for each dataset and setting of $X_{\max}$. The relative numbers of rules discovered for each dataset and $X_{\max}$ by the direct-adjustment significance tests and by each of the holdout evaluation techniques are plotted in Fig. 4.

It is striking how many patterns of dubious quality are discovered when statistical evaluation is not performed. For most treatments the vast majority of non-redundant rules do not even pass an unadjusted significance test on the data from which they are discovered, let alone one which is adjusted to allow for multiple tests. For example, for TICDATA 2000 with antecedent size 4 more than 100 000 non-redundant rules are found, but the most powerful of the statistically sound approaches passes only 78 (less than 0.1%), meaning that more than 99.9% of the non-redundant rules found are of questionable value. While discarding unproductive rules removes many non-redundant patterns that are most probably spurious, in many cases there are still many thousands of rules discovered that do not pass a significance test. In the TICDATA 2000 example, more than 99% of the unproductive rules are not accepted by any of the statistically sound approaches. Even the use of an unadjusted significance test results in very large numbers of discoveries that do not pass an adjusted test. In the TICDATA 2000 case, more than 96% of these rules are not accepted by any of the statistically sound approaches. Given the propensity of an unadjusted test to make false discoveries, as illustrated in Experiment 1, it seems likely that many of these additional discoveries are likewise spurious.

Holdout evaluation with rules that pass an unadjusted significance test usually finds slightly more rules than Holdout evaluation with productive rules, which in turn usually finds slightly more rules that Holdout evaluation with non-redundant rules. This is because the size of the correction for multiple tests that is performed during holdout evaluation is smaller when a stronger filter is applied during pattern discovery. The use of a stronger filter will only result in fewer discoveries if it excludes patterns at the exploratory discovery stage that would be passed at the holdout stage. In only two cases do the use of a weaker filter result in more rules being discovered than the use of an unadjusted significance test, the use of a productive filter for Covtype with $X_{\max} = 1$ and for IPUMS LA 99 with $X_{\max} = 2$. In both cases two additional discoveries are made. In contrast the stronger filter usually results in substantially more discoveries than the weaker filters.

For three datasets, the greatest number of discoveries were made by the direct-adjustment approach. For the remaining seven datasets the greatest number of discoveries were made by holdout evaluation applied after an unadjusted significance test. Overall, the holdout approach found more rules than direct-adjustment significance testing, with the relative performance being more favorable to direct-adjustment when the size of the rule space was smaller (small $X_{\max}$ or fewer items) and more favorable to holdout evaluation as the size of the search space increased. This is because of the extremely small significance levels that are employed with large search spaces. The total number of rules discovered by direct-adjustment often decreased as the search space increased. Such decreases occurred less frequently for holdout evaluation and when they did, the decreases were smaller. Note, however, that these relative results are in part an artifact of the experimental design. If minimum

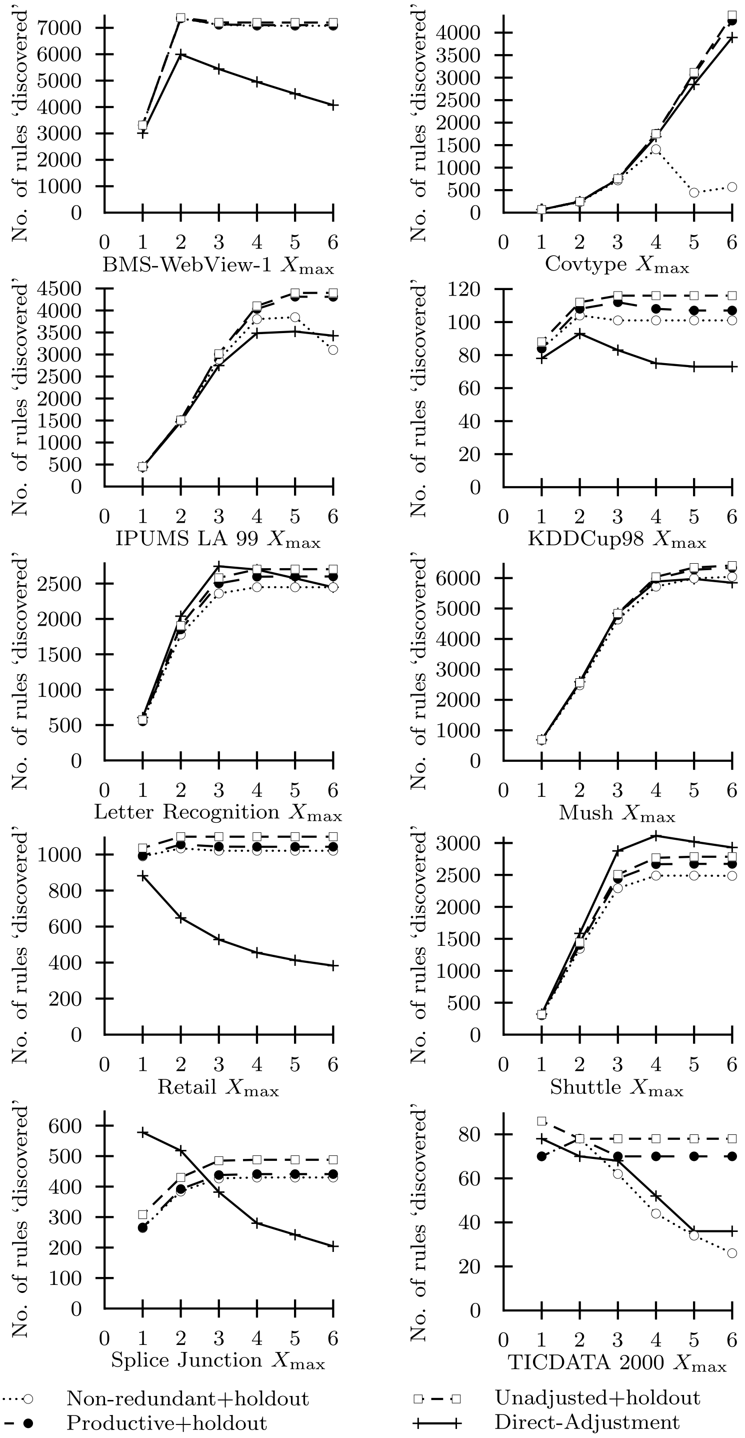

Fig. 4 Numbers of rules 'discovered'

support had been set lower, direct-adjustment would have found more rules, as the rules found under the current setting would not have been affected and further rules with lower support could have potentially passed the adjusted significance test. However, increasing minimum support could have reduced the number of rules found by holdout evaluation as it would have increased the number of candidate rules and hence lowered the adjusted significance level that rules had to pass.

It is also interesting to observe that in some cases there was considerable difference in the numbers of rules found during the exploratory stage of holdout evaluation relative to those found from the full dataset. The most extreme examples of this were BMS-Webview-1, for which substantially larger numbers of candidate rules were found relative to discovery from the full dataset and TICDATA 2000 for which substantially fewer rules were found. This illustrates the disadvantage of working with the smaller samples inherent in the holdout approach.

6.4 Computational Results

Table 7 in Appendix 2 presents the User CPU times for each treatment in the experiments on real-world data using an AMD64 3000 939 Linux system. These times should be treated with great caution, as they are the results for a single run only and time can vary substantially from one run to another of the same process.

With this caveat, it appears that the direct-adjustment approach does sometimes substantially increase the compute time in comparison to searching just for non-redundant rules. The most extreme example of this is for Covtype with $\mathbf{X}_{\max} = \mathbf{6}$ for which the direct adjustment approach takes more than 16 times as long, non-redundant rules taking slightly over 6 CPU seconds and direct adjustment taking slightly over 100.

The direct adjustment approach appears to have a very slight computational advantage over the use of a statistical test without adjustment. This presumably reflects an advantage obtained from extra pruning of the search space. On the whole, there is a substantial advantage within the holdout approaches to holdout evaluation with the application of a significance test during search, presumably because the reduced number of tests applied at holdout evaluation time more than compensates for the additional cost of the stricter evaluation during the exploratory search stage. The exception is for Covtype, where the extra cost of applying a statistical test during the search phase outweighs this advantage, at least when comparing against productive rules, which at $\mathbf{X}_{\max} = \mathbf{6}$ takes slightly over 20 CPU seconds.

Neither the direct adjustment nor the holdout approach has a clear advantage over the other, with the relative compute times varying greatly from dataset to dataset. The CPU times for direct adjustment at $\mathbf{X}_{\max} = \mathbf{6}$ vary from 0.12 seconds for TICDATA 2000 through to the 100 seconds already discussed for Covtype. For holdout evaluation with an unadjusted significance test they vary from 0.10 seconds for TICDATA 2000 through to slightly under 62 seconds for Covtype. When direct adjustment is more efficient the difference is not as great as the most extreme advantage enjoyed by holdout evaluation. For example, for BMS-WebView-1 direct adjustment takes 3.70 CPU seconds while holdout evaluation with an unadjusted significance test takes 4.78.

6.5 How Many Highly Significant Rules Are There?

A question that arises naturally is just how many rules are going to reach a sufficient level of significance that they will be accepted by the direct-adjustment approach. Our assumption before embarking on the research reported herein was that the numbers would be so low

Table 3 The number of significant rules for each dataset at $X_{\max} = 4$

Dataset	α'	Significant rules
BMS-WebView-1	4.00×10^{-14}	>1 000 000
Covtype	3.23×10^{-11}	>1 000 000
IPUMS LA 99	1.81×10^{-16}	191 472
KDDCup98	5.71×10^{-20}	>345 000
Letter Recognition	1.87×10^{-09}	57 233
Mush	6.37×10^{-11}	61 901
Retail	9.90×10^{-22}	5088
Shuttle	7.95×10^{-08}	4234
Splice Junction	1.64×10^{-12}	395
TICDATA 2000	9.04×10^{-15}	>155 000

in many applications as to make the direct-adjustment approach infeasible. Experiment 5 sought insight into this issue by running *Magnum Opus* on each of the datasets listed in Table 2 using its default settings except that the maximum number of rules to be discovered was increased to 1 000 000 and the significance level used in the statistical test was adjusted to take account of the search space size.

For two datasets, KDDCup98 and TICDATA 2000, this computational task proved infeasible, and the search was terminated early, establishing a lower bound on the number of significant rules. In the case of TICDATA 2000, search was terminated after more than two CPU weeks of processing on an AMD64 3000 939 Linux system. The reason for this processing inefficiency is the reduction in pruning of the search space that occurs when significance tests with very low critical values are introduced in k-optimal rule discovery. It is only feasible to search the massive search spaces involved in these tasks if most of the search space can be pruned from consideration. *Magnum Opus* has only weak pruning mechanisms relating to its significance tests, and hence gains only moderate increases in pruning from the test. This small increase fails to offset the massive loss in pruning that can occur in k-optimal rule discovery when the significance tests prevent rules from being accepted. This is because the most effective pruning utilizes increases in the minimum bound on leverage that can be derived as additional candidates that pass all other constraints are encountered. This computational inefficiency was not evident in the previous experiments because they did not use the k-optimal rule discovery approach.

As *Magnum Opus*'s default setting allows antecedents of up to size 4, the row for $X_{\max} = 4$ in Table 6 provides the relevant search space size for each dataset. Table 3 lists the resulting adjusted significance level and the number of rules found for each dataset. The column headed “Significant Rules” indicates the number of rules that satisfy a within-search significance test at $X_{\max} = 4$. Where the number is preceded by “>” it indicates a lower bound on the number of such rules. It is apparent that for many datasets there are very large numbers of significant patterns available to be found.

7 Discussion and Future Research

The current work reveals a number of interesting issues that provide potentially productive avenues for future research.

The relative performance of direct-adjustment and holdout evaluation differs substantially from application domain to application domain. While the current research has identified broad factors that influence their relative performance, it would be useful if techniques could be found for predicting which would be more powerful for a specific task.

Similarly, the numbers of discoveries found by the direct-adjustment technique differ substantially as $X_{\max}$ increases. This reflects a trade-off between increasing the number of true patterns contained within the search space and hence available to be discovered and decreasing the critical value employed in the statistical test and hence increasing the strength of association required for a pattern to be discovered. It would be valuable to find approaches for selecting an appropriate trade-off between these two effects for any given application.

The use of a filter on the rules that are passed to holdout evaluation has been demonstrated to be very effective. The strongest filter applied in the current studies was an unadjusted significance test. Further experiments, not presented, have demonstrated that the application of a significance test with a reduced critical value can further increase the number of discoveries, but, as the critical value is decreased, eventually a level will be found at which the number of discoveries starts decreasing. It would be useful to develop techniques for setting appropriate adjustments for the purposes of filtering.

The current research has performed holdout evaluation by setting each of the exploratory and holdout data sets to contain 50% of the available data. There is no reason to suppose that this split should be optimal. It could be argued that the exploratory data need only be of sufficient size that the majority of relevant patterns should be revealed and that most of the data should be reserved for holdout evaluation so as to maximize the power of the statistical evaluation. This is a promising issue for further investigation.

It is worth noting that an advantage of the holdout approach relative to direct-adjustment is that it can support controls on the false discovery rate (Benjamini and Hochberg 1995) instead of the experimentwise error rate. If this is to be done, a technique that accommodates correlations between the hypothesis tests should be employed (Benjamini and Yekutieli 2001).

An advantage of the direct-adjustment approach is that it supports k-optimal pattern discovery (Webb 1995; Scheffer and Wrobel 2002; Webb and Zhang 2005). Rather than seeking all patterns that satisfy some set of constraints, as do the frequent pattern approaches, k-optimal (also known as *k-top*) approaches take user-specified metrics of the value of a pattern and k, the number of patterns to be discovered, and find the k patterns that optimize the metric of interest within any other constraints the user might specify. These approaches have proved very popular, as they avoid the need to experimentally determine appropriate settings for minimum support and related constraints. However, the holdout-evaluation approach undermines k-optimal techniques, as it is not possible to determine in advance how many patterns will pass holdout evaluation, and hence not possible to ensure that k discoveries will be made. In contrast, a direct-adjustment significance test can become just another constraint, such that a k-optimal approach finds the k *significant* patterns that optimize the metric of interest.

The current research has used statistical tests for productivity to assess the rules. In practice many other forms of test might be appropriate, such as a simple test for independence between the antecedent and consequent, tests that allow negative rules or tests for a specific minimum level of support or confidence. The holdout and direct-adjustment techniques may be applied to any traditional statistical hypothesis test. An interesting direction for future research is to identify properties of patterns for which it may be desirable to apply such tests.

The current work has considered only patterns in the forms of rules. The generic techniques generalize directly to other forms of pattern such as itemsets or sequential patterns, and it would be valuable to develop appropriate techniques to support each such type of pattern.

The current techniques provide strict control over the risk of type-1 error, the risk of 'discovering' a pattern that is false. The techniques do not provide bounds on the risk of type-2 error, the risk of failing to discover a pattern that is true. This is perhaps inevitable in many real-world data analysis contexts, as assessment of the risk of type-2 error requires knowledge of the size of the effect of the phenomenon being investigated, and this is typically unknown. Nonetheless, it is desirable to minimize the risk of type-2 error, and techniques for better managing the dual risks of type-1 and type-2 error remain an important area for future research.

The large risk of type-2 error inherent in these techniques might lead some practitioners to question their use. After all, do we really want to use pattern discovery systems that fail to discover large numbers of patterns? In many applications the end-users will usually only be prepared to consider a very small number of patterns. While a pattern discovery system may discover millions of patterns, in our experience the numbers of patterns with which an end-user can engage is typically measured in the dozens. If only a small number of patterns will actually be considered, perhaps it is best to limit these to those in which we have very high confidence that the risk of type-1 error is low. As Table 3 demonstrates, for many real-world datasets the numbers of rules in which we can have high confidence will often be large. In many cases there will be more such rules than an end-user will be able consider. In this context it is tempting to ask why we would direct users to consider alternative rules in which our confidence is less high when there are so many in which we can have high confidence?

8 Conclusions

This paper presents two techniques for performing statistically sound pattern discovery. Both provide mechanisms for applying standard statistical hypothesis tests to all patterns discovered in a manner that strictly controls the risk of experimentwise type-1 error. The holdout technique separates the available data into exploratory and holdout sets, discovers rules from the former and evaluates them against the latter, using a Bonferroni or similar adjustment for the number of rules evaluated against the holdout data. It differs from the standard use of holdout evaluation in machine learning which seeks unbiased estimates of the performance of a single model. Rather, it seeks to apply statistical hypothesis tests to accept or reject each of a set of patterns discovered by evaluation of a large space of potential patterns against sample data. The direct-adjustment technique applies to statistical tests that are employed during the pattern discovery process a Bonferroni adjustment for the size of the search space.

These two techniques may be used with any statistical hypothesis test. The current research has used tests for whether a rule is productive. Where the analytic objective is to find rules that represent positive correlations, such tests appear highly desirable.

Experiments demonstrate that application of standard pattern discovery techniques to random data that does not embody any underlying patterns can find numerous spurious patterns, as can the application of a statistical test without adjustment for multiple testing. When applied to real-world data, standard pattern discovery techniques find numerous patterns that do not pass holdout evaluation. By applying well-established statistical principles, the new techniques overcome these serious problems, even when considering search spaces containing in excess of 10^{26} alternative patterns.

For the holdout approach, the application of a statistical filter during rule discovery that discarded rules that were unlikely to pass subsequent holdout evaluation proved very powerful, and resulted in substantial increases in the number of rules that passed holdout evaluation.

Each of the approaches has some advantages and disadvantages compared to the other. The holdout approach can be applied as a simple wrapper to any existing pattern discovery system, allowing those using pattern discovery to essentially keep doing what they have been doing, but add a simple step that has high probability of discarding all spurious patterns that they would otherwise have assumed were real discoveries. In contrast, the direct-adjustment approach may require substantial re-engineering of a system. The holdout approach is less susceptible to decreases in its power as a result of increases in the size of the search space, can utilize more powerful corrections for multiple tests such as the Holm procedure, and can support procedures to control the false discovery rate as well as the experimentwise error rate. Further, the actual number of tests that must be applied will often be orders of magnitude lower than under the direct-adjustment approach, providing a considerable computational advantage when employing computationally demanding statistical tests. On the other hand, the direct-adjustment approach better supports k-optimal pattern discovery and utilizes all available data for both pattern detection and pattern evaluation.

Given the ease with which the holdout technique can be retro-fitted to any existing pattern discovery process, it would appear desirable for anyone seeking to perform such an analysis to consider what statistical tests are appropriate to their specific application and to at least assess the feasibility of applying those tests using either the holdout or the direct-adjustment technique.

This research demonstrates both how computational research can scale conventional statistics to handle massive tasks that appear previously to have been assumed intractable and how the appropriate application of conventional statistics can solve serious problems in machine learning.

Acknowledgements I wish to thank Blue Martini Software for contributing the KDD Cup 2000 data, Tom Brijs and his colleagues for the Retail data, the librarians of and contributors to the UCI machine learning and KDD repositories for contributing the remaining data sets, and Janice Boughton for assistance in compiling the experimental results. I also wish to thank Mike Pazzani, Shane Butler, Janice Boughton, Ying Yang, Michael Hahsler, the action editor Johannes Fürnkranz and anonymous reviewers for helpful feedback and comments on drafts of this paper. This research has been supported by the Australian Research Council under grant DP0450219.

Appendix 1 Fisher Exact Test for Productive Rules

Given a rule $x_1 \& x_2 \& \cdots \& x_n \rightarrow y$, where each x_i and y are tests that are individually true or false of each record in the data D, let $X = x_1 \& x_2 \& \cdots \& x_n$ and $X - x_i = x_1 \& \ldots \& x_{i-1} \& x_{i+1} \& \cdots \& x_n$. We wish to test

$$\forall x \in \{1 \ldots n\}, P(y \mid X) > P(y \mid X - x_i). \tag{8}$$

The imposition of (8) ensures that all factors in the antecedent of a rule contribute to its confidence, or in other words, that the confidence is higher than that of any of its immediate specializations. Applying a hypothesis test to test for (8) is similar to the imposition of statistical test on a minimum improvement constraint. It requires that a rule pass a hypothesis test with respect to the null hypothesis that the probability of the consequent given the antecedent in the population from which the sample data are drawn is no greater than that for a generalization, or in other words, that the improvement is greater than zero with respect to the distribution from which the data are drawn.

Chi-square is an obvious test to apply to evaluate (8). This is the test used by Brin et al. (1997), Liu et al. (1999) and Bay and Pazzani (2001) in the context of applying tests during pattern discovery. Brin et al. and Liu et al. use the chi-square test without adjustment for multiple comparisons whereas Bay and Pazzani use it with a partial adjustment for multiple comparisons.

However, there are two reasons why the chi-square test may not always be ideal for this purpose. First, chi-square is an approximate test that is notoriously unreliable for small samples (Johnson 1984). As pattern discovery is often applied to sparse data, the samples relating to a given test may be expected to be small. Second, chi-square is a two-tailed test, and in the current context a one-tailed test appears more appropriate.

The Fisher exact test (Agresti 1992) is appropriate for (8). To evaluate (8) for each x, calculate the probability of observing the observed number or greater of occurrences of $y \,\&\, X$ given the number of observed occurrences of $y \,\&\, X - x_i$ if $P(y \mid X) = P(y \mid X - x_i)$.

The p value for this test can be calculated as follows. Let a, b, c and d be, respectively the frequencies with which X and y co-occur, X occurs without y, y occurs with $X - x_i$ but without x_i, and $X - x_i$ but neither x_i nor y occurs.

$$p = \sum_{i=0}^{\min(b,c)} \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{(a+b+c+d)!(a+i)!(b-i)!(c-i)!(d+i)!}. \tag{9}$$

Here, $n!$ denotes the factorial of n.

The Fisher exact test is exact, and hence reliable even with infrequent data.

The Fisher exact test has a reputation for excessive computational requirements. Certainly, the amount of computation it requires for any non-trivial task makes it infeasible for hand computation. However, it has polynomial time complexity, and in practice the computation it requires is minor in comparison to the other computations required in typical pattern discovery tasks.

Appendix 2 Tables of Experimental Outcomes

Tables 4 and 5 present results for experiments 2 and 3, discussed in Sect. 6.2. For each treatment they list the mean numbers of true and false discoveries per run, together with the number of runs for which any false discoveries occurred.

Table 6 presents the results of the experiments on real-world data (Sect. 6.3). It lists the number of rules found by each technique for each dataset and setting of $X_{\max}$. The meanings of the columns are as follows:

Dataset: The dataset.

X$_{\max}$: The maximum number of items allowed in the antecedent.

NR: The number of non-redundant rules ‘discovered’.

Prod: The number of productive rules ‘discovered’.

0.05: The number of rules ‘discovered’ that passed an unadjusted significance test at the 0.05 level.

Direct-Adjustment Rule Space: The number of rules in the search space. The direct-adjustment technique used a significance level of 0.05 divided by this value.

Direct-Adjustment Disc: The number of rules ‘discovered’ that passed the adjusted significance test. This is abbreviated as WS, below.

Table 4 Results for Experiment 2

Data	Mean true discoveries					Mean false discoveries					Experiment false disc.				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
None 100															
250	29.56	21.19	21.12	21.12	21.12	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
500	33.80	21.77	21.67	21.67	21.67	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
1000	36.24	21.88	21.87	21.87	21.87	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
2000	37.76	22.13	22.13	22.13	22.13	0.08	0.00	0.00	0.00	0.00	2	0	0	0	0
4000	39.56	21.92	21.92	21.92	21.92	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
8000	40.00	22.32	22.32	22.32	22.32	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
16000	40.00	21.84	21.84	21.84	21.84	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
None 1000															
250	27.68	27.48	26.26	26.24	26.24	0.00	0.01	0.00	0.00	0.00	0	1	0	0	0
500	32.00	31.00	28.88	28.82	28.82	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
1000	35.32	33.88	30.20	30.20	30.20	0.00	0.01	0.00	0.00	0.00	0	1	0	0	0
2000	36.72	34.40	30.00	30.32	30.32	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
4000	38.88	34.54	31.32	31.32	31.32	0.00	0.06	0.06	0.06	0.06	0	4	4	4	4
8000	39.96	34.39	31.80	31.80	31.80	0.04	0.07	0.06	0.06	0.06	1	6	5	5	5
16000	40.00	34.39	32.00	32.00	32.00	0.00	0.06	0.03	0.03	0.03	0	5	2	2	2
None 10000															
250	27.56	26.24	26.24	26.16	26.16	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
500	31.48	30.64	30.32	30.16	30.16	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
1000	35.16	34.36	33.76	33.32	33.32	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
2000	36.44	36.24	35.24	34.32	34.32	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
4000	38.68	38.12	35.72	34.52	34.48	0.04	0.02	0.00	0.00	0.00	1	1	0	0	0
8000	39.96	39.92	35.92	34.60	34.56	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
16000	40.00	40.00	36.00	34.92	34.80	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
Sig 100															
250	29.80	29.28	29.06	29.06	29.06	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
500	33.96	32.98	32.31	32.30	32.30	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
1000	36.24	35.75	33.47	33.47	33.47	0.04	0.01	0.01	0.01	0.01	1	1	1	1	1
2000	37.84	37.00	33.27	33.25	33.25	0.08	0.03	0.03	0.03	0.03	2	2	2	2	2
4000	39.56	39.01	32.97	32.96	32.96	0.00	0.04	0.02	0.02	0.02	0	4	2	2	2
8000	40.00	39.81	32.94	32.94	32.94	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
16000	40.00	39.35	32.59	32.59	32.59	0.04	0.05	0.02	0.02	0.02	1	5	2	2	2
Sig 1000															
250	29.80	28.68	27.92	27.64	27.64	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
500	33.96	32.84	32.00	31.92	31.92	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
1000	36.16	35.76	35.28	35.24	35.24	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
2000	37.80	36.88	36.68	36.60	36.60	0.12	0.08	0.04	0.00	0.00	2	1	1	0	0
4000	39.56	39.32	38.88	38.84	38.72	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
8000	40.00	40.00	39.96	39.96	39.76	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
16000	40.00	40.00	40.00	40.00	40.00	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
Sig 10000															
250	29.80	28.68	27.92	27.56	27.52	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
500	33.96	32.84	32.00	31.20	31.04	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
1000	36.16	35.76	35.28	34.84	34.40	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
2000	37.80	36.88	36.52	36.32	36.24	0.12	0.08	0.04	0.00	0.00	2	1	1	0	0
4000	39.56	39.32	38.68	38.40	38.12	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
8000	40.00	40.00	39.96	39.92	39.92	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
16000	40.00	40.00	40.00	40.00	40.00	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
Direct															
250	31.28	29.72	28.76	28.00	27.20	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
500	35.20	33.76	32.68	32.00	31.44	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
1000	36.88	36.12	35.84	35.28	34.76	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
2000	38.56	37.52	36.80	36.56	36.36	0.12	0.00	0.00	0.00	0.00	3	0	0	0	0
4000	39.96	39.84	39.24	39.04	38.76	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
8000	40.00	40.00	40.00	40.00	40.00	0.08	0.00	0.00	0.00	0.00	1	0	0	0	0
16000	40.00	40.00	40.00	40.00	40.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0

HO-NR Cand: The number of non-redundant candidate rules generated from the exploratory data under the holdout approach.

HO-NR Disc: The number of those candidate rules that passed the subsequent holdout evaluation.

Table 5 Results for Experiment 3

Data	Mean true discoveries					Mean false discoveries					Experiment false disc.				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
None 100															
1000	0.00	0.00	0.00	0.00	0.00	0.16	0.08	0.04	0.04	0.04	4	2	1	1	1
2000	0.20	0.10	0.20	0.40	0.40	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
4000	0.60	0.70	2.00	3.00	3.00	0.16	0.04	0.00	0.00	0.00	4	1	0	0	0
8000	2.90	5.20	9.50	11.70	11.60	0.00	0.01	0.01	0.02	0.02	0	1	1	2	2
16000	9.50	18.70	29.30	32.30	31.30	0.04	0.04	0.06	0.04	0.04	1	4	4	3	3
32000	15.00	36.20	57.20	57.20	54.80	0.00	0.02	0.05	0.04	0.05	0	2	5	4	4
64000	16.00	45.07	70.67	70.27	66.67	0.04	0.07	0.03	0.06	0.09	1	7	3	5	7
None 1000															
1000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
2000	0.00	0.00	0.00	0.20	0.20	0.00	0.01	0.01	0.01	0.01	0	1	1	1	1
4000	0.30	0.40	1.00	2.10	2.10	0.04	0.05	0.04	0.04	0.04	1	2	1	1	1
8000	1.30	2.60	6.20	9.30	9.30	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
16000	6.20	12.40	21.90	28.10	28.10	0.00	0.02	0.02	0.03	0.03	0	2	2	3	3
32000	13.90	28.70	51.80	60.60	60.60	0.04	0.03	0.03	0.03	0.03	1	3	3	3	3
64000	16.00	42.90	70.82	79.84	79.84	0.04	0.02	0.03	0.08	0.07	1	2	3	6	6
None 10000															
1000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
2000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
4000	0.30	0.10	0.30	1.10	1.10	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
8000	1.30	0.90	3.20	5.70	5.70	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
16000	6.20	6.80	13.10	18.60	18.60	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
32000	13.90	21.80	40.60	49.00	49.00	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
64000	16.00	39.42	67.17	76.17	76.17	0.04	0.01	0.01	0.01	0.01	1	1	1	1	1
Sig 100															
1000	0.00	0.00	0.00	0.00	0.00	0.08	0.08	0.05	0.05	0.05	2	2	2	2	2
2000	0.20	0.10	0.20	0.40	0.40	0.00	0.01	0.00	0.00	0.00	0	1	0	0	0
4000	0.80	0.70	2.10	3.10	3.10	0.08	0.05	0.04	0.04	0.04	2	2	1	1	1
8000	3.60	5.40	9.90	12.90	12.80	0.00	0.01	0.02	0.02	0.02	0	1	2	2	2
16000	10.20	19.10	32.80	39.00	38.80	0.04	0.00	0.03	0.03	0.03	1	0	3	3	3
32000	15.20	36.30	63.20	70.50	70.30	0.00	0.01	0.01	0.01	0.01	0	1	1	1	1
64000	16.00	45.06	73.31	81.96	81.91	0.12	0.00	0.00	0.05	0.06	3	0	0	5	6
Sig 1000															
1000	0.00	0.00	0.00	0.00	0.00	0.08	0.04	0.00	0.00	0.00	2	1	0	0	0
2000	0.20	0.10	0.10	0.10	0.10	0.00	0.01	0.01	0.01	0.01	0	1	1	1	1
4000	0.80	0.80	1.30	2.10	2.10	0.08	0.04	0.04	0.04	0.04	2	1	1	1	1
8000	3.60	4.70	6.90	9.20	9.20	0.00	0.01	0.00	0.00	0.00	0	1	0	0	0
16000	10.20	16.10	23.80	27.70	27.70	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
32000	15.20	33.30	53.70	60.20	60.20	0.00	0.00	0.00	0.01	0.01	0	0	0	1	1
64000	16.00	44.54	71.44	79.84	79.84	0.12	0.04	0.00	0.00	0.00	3	1	0	0	0
Sig 10000															
1000	0.00	0.00	0.00	0.00	0.00	0.08	0.04	0.00	0.00	0.00	2	1	0	0	0
2000	0.20	0.10	0.10	0.10	0.10	0.00	0.01	0.01	0.00	0.00	0	1	1	0	0
4000	0.80	0.80	1.30	1.90	1.60	0.08	0.04	0.04	0.04	0.00	2	1	1	1	0
8000	3.60	4.70	6.90	8.30	6.90	0.00	0.01	0.00	0.00	0.00	0	1	0	0	0
16000	10.20	16.10	23.80	25.10	21.80	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
32000	15.20	33.30	53.70	57.60	53.20	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
64000	16.00	44.54	71.44	78.85	77.59	0.12	0.04	0.00	0.00	0.00	3	1	0	0	0
Direct															
1000	0.04	0.00	0.00	0.00	0.00	0.12	0.00	0.00	0.00	0.00	3	0	0	0	0
2000	0.36	0.08	0.11	0.22	0.14	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
4000	1.28	0.96	1.84	2.70	1.92	0.08	0.04	0.00	0.00	0.00	2	1	0	0	0
8000	5.92	6.73	8.76	11.09	9.41	0.00	0.01	0.00	0.00	0.00	0	1	0	0	0
16000	13.96	21.31	31.10	32.29	27.42	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0
32000	16.00	39.22	61.92	66.68	63.02	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0
64000	16.00	45.88	73.60	82.06	81.58	0.04	0.00	0.00	0.00	0.00	1	0	0	0	0

HO-Prod Cand: The number of productive candidate rules generated from the exploratory data under the holdout approach.

HO-Prod Disc: The number of those candidate rules that passed the subsequent holdout evaluation.

Table 6 Number of rules found in the experiments on real-world data

Dataset	$X_{\max}$	NR	Prod	0.05	Direct-Adjustment		HO-NR		HO-Prod		HO-Unadj	
					Space	Disc	Cand	Disc	Cand	Disc	Cand	Disc
BMS-WebView-1	1	3146	3126	3110	2.46×10^{05}	3010	3576	3312	3558	3314	3524	3316
BMS-WebView-1	2	7622	7548	7394	6.12×10^{07}	5995	10245	7357	10159	7363	9886	7386
BMS-WebView-1	3	9710	9511	8844	1.01×10^{10}	5440	14412	7114	14031	7130	12249	7206
BMS-WebView-1	4	10004	9765	8930	1.25×10^{12}	4953	15266	7080	14624	7102	12339	7200
BMS-WebView-1	5	10016	9772	8931	1.23×10^{14}	4503	15315	7079	14632	7102	12339	7200
BMS-WebView-1	6	10016	9772	8931	1.01×10^{16}	4073	15315	7079	14632	7102	12339	7200
Covtype	1	1794	74	74	1.78×10^{04}	68	1794	68	74	70	72	68
Covtype	2	37593	290	286	1.17×10^{06}	245	37590	233	284	245	271	247
Covtype	3	100000	1035	971	4.98×10^{07}	746	100000	717	987	755	919	755
Covtype	4	100000	2829	2477	1.55×10^{09}	1690	100000	1409	2649	1752	2336	1752
Covtype	5	100000	5967	4887	3.80×10^{10}	2848	100000	444	5559	3065	4548	3116
Covtype	6	100000	9995	7778	7.56×10^{11}	3893	100000	570	9351	4267	7135	4390
IPUMS LA 99	1	984	526	472	3.08×10^{06}	440	978	440	526	442	456	452
IPUMS LA 99	2	7822	2152	1782	2.23×10^{09}	1469	7802	1491	2193	1510	1704	1508
IPUMS LA 99	3	33957	5324	3999	9.60×10^{11}	2748	33904	2886	5366	3004	3721	3017
IPUMS LA 99	4	94157	8382	5832	2.77×10^{14}	3483	93912	3803	8337	4034	5301	4103
IPUMS LA 99	5	100000	9763	6440	5.73×10^{16}	3522	100000	3847	9685	4312	5778	4398
IPUMS LA 99	6	100000	9998	6510	8.93×10^{18}	3426	100000	3103	9982	4310	5838	4400
KDDCup98	1	667	402	172	1.50×10^{08}	78	659	84	360	84	116	88
KDDCup98	2	5721	1885	365	4.39×10^{11}	93	5614	104	1579	108	214	112
KDDCup98	3	24681	4638	489	7.49×10^{14}	83	24081	101	3877	112	272	116
KDDCup98	4	63564	7601	574	8.76×10^{17}	75	61758	101	6302	108	303	116
KDDCup98	5	100000	9384	632	7.66×10^{20}	73	100000	101	7757	107	311	116
KDDCup98	6	100000	9988	652	5.28×10^{23}	73	100000	101	8200	107	311	116
Letter Recognition	1	1490	854	744	4.66×10^{03}	606	1520	552	868	556	724	574
Letter Recognition	2	8536	4003	3228	1.34×10^{05}	2039	8448	1778	4093	1852	2994	1905
Letter Recognition	3	19040	7534	5676	2.29×10^{06}	2744	18128	2360	7409	2502	4925	2581
Letter Recognition	4	26697	9443	6762	2.68×10^{07}	2697	24039	2449	8963	2597	5588	2702
Letter Recognition	5	29195	9939	6967	2.27×10^{08}	2574	25495	2448	9238	2600	5667	2703
Letter Recognition	6	29447	9964	6974	1.47×10^{09}	2448	25537	2448	9241	2600	5667	2703
Mush	1	1136	778	748	1.52×10^{04}	686	1144	672	778	684	734	690
Mush	2	7625	3501	3233	8.75×10^{05}	2594	7697	2479	3526	2558	3106	2567
Mush	3	21773	7079	6463	3.12×10^{07}	4844	21873	4639	7205	4781	6202	4838
Mush	4	34866	9229	8351	7.85×10^{08}	5885	34881	5719	9410	5976	7985	6039
Mush	5	41155	9885	8905	1.48×10^{10}	5972	41026	5988	10112	6274	8493	6346
Mush	6	42830	9998	9005	2.16×10^{11}	5845	42584	6041	10245	6335	8590	6412
Retail	1	5908	5250	4142	2.72×10^{08}	882	6056	990	5352	994	3652	1036
Retail	2	10669	8943	6251	2.23×10^{12}	648	10955	1034	9081	1056	5247	1099
Retail	3	12008	9847	6571	1.23×10^{16}	528	12310	1021	9931	1044	5430	1099
Retail	4	12153	9909	6576	5.05×10^{19}	455	12464	1021	9993	1043	5432	1099
Retail	5	12153	9909	6576	1.66×10^{23}	413	12464	1021	9993	1043	5432	1099
Retail	6	12153	9909	6576	4.56×10^{26}	383	12464	1021	9993	1043	5432	1099
Shuttle	1	706	380	354	1.03×10^{03}	322	706	302	382	310	342	316
Shuttle	2	6182	2426	2082	1.46×10^{04}	1585	6172	1345	2440	1404	1952	1446
Shuttle	3	22491	6632	4891	1.19×10^{05}	2876	22455	2292	6635	2442	4426	2507
Shuttle	4	42158	9420	6345	6.30×10^{05}	3113	41701	2489	9354	2668	5554	2768
Shuttle	5	50674	9970	6587	2.28×10^{06}	3019	49884	2489	9885	2673	5717	2785
Shuttle	6	51776	9993	6591	5.83×10^{06}	2930	50907	2486	9900	2673	5719	2785
Splice Junction	1	9172	6846	4004	5.80×10^{04}	578	9786	264	7400	266	3390	308
Splice Junction	2	12574	9111	5152	6.88×10^{06}	518	13055	384	9583	392	4367	430
Splice Junction	3	13553	9697	5484	5.32×10^{08}	382	13932	427	10132	438	4635	485
Splice Junction	4	13675	9743	5514	3.04×10^{10}	280	14051	430	10180	441	4663	488
Splice Junction	5	13683	9744	5514	1.36×10^{12}	242	14059	430	10182	441	4663	488
Splice Junction	6	13683	9744	5514	5.00×10^{13}	204	14059	430	10182	441	4663	488
TICDATA 2000	1	814	454	294	4.68×10^{05}	78	806	70	366	70	206	86
TICDATA 2000	2	9446	2334	1038	1.56×10^{08}	70	9118	78	1310	78	478	78
TICDATA 2000	3	55630	5662	1886	3.42×10^{10}	68	51982	62	2382	70	638	78
TICDATA 2000	4	100000	8670	2270	5.53×10^{12}	52	100000	44	2734	70	670	78
TICDATA 2000	5	100000	9694	2270	7.02×10^{14}	36	100000	34	2734	70	670	78
TICDATA 2000	6	100000	9694	2270	7.32×10^{16}	36	100000	26	2734	70	670	78

Table 7 User CPU time in seconds under each treatment for the experiments on real-world data

Dataset	$x_{\max}$	NR	Prod	0.05	DA	HO-NR	HO-Prod	HO-Unadj
BMS-WebView-1	1	3.27	3.32	3.39	3.38	4.64	3.10	3.12
BMS-WebView-1	2	3.48	3.77	3.91	3.75	3.36	3.80	3.88
BMS-WebView-1	3	3.57	4.14	4.27	3.81	3.57	4.90	4.67
BMS-WebView-1	4	3.58	4.19	4.32	3.77	3.64	5.17	4.77
BMS-WebView-1	5	3.59	4.21	4.31	3.69	3.64	5.16	4.74
BMS-WebView-1	6	3.59	4.18	4.32	3.70	3.62	5.16	4.78
Covtype	1	4.02	4.01	4.14	4.14	4.09	3.99	4.06
Covtype	2	5.08	4.17	4.82	4.76	10.70	4.36	4.55
Covtype	3	5.57	4.63	8.18	7.89	51.68	5.88	7.07
Covtype	4	5.87	6.15	19.41	18.48	199.98	10.58	14.81
Covtype	5	6.24	10.40	48.09	45.79	247.86	20.78	32.07
Covtype	6	6.23	20.37	104.33	100.26	254.14	36.78	61.93
IPUMS LA 99	1	1.58	1.57	2.22	2.21	1.80	1.77	2.06
IPUMS LA 99	2	1.70	1.65	4.15	3.96	4.78	3.10	3.99
IPUMS LA 99	3	1.86	2.02	7.63	7.21	16.33	5.67	7.24
IPUMS LA 99	4	1.95	3.09	12.86	12.05	39.00	8.44	10.84
IPUMS LA 99	5	1.98	4.42	21.32	20.44	28.73	10.10	15.03
IPUMS LA 99	6	1.98	5.10	27.67	26.75	26.40	10.64	17.86
KDDCup98	1	27.68	27.66	27.72	27.80	31.66	31.79	31.67
KDDCup98	2	27.76	27.69	27.82	27.83	31.86	31.75	31.68
KDDCup98	3	27.88	28.00	28.10	28.02	33.22	32.02	31.91
KDDCup98	4	27.90	28.91	28.63	28.55	37.45	32.47	32.09
KDDCup98	5	28.02	30.25	29.08	28.83	38.85	32.94	32.36
KDDCup98	6	28.04	30.93	29.76	29.37	39.50	33.36	32.56
Letter Recognition	1	0.07	0.06	0.18	0.16	0.11	0.09	0.12
Letter Recognition	2	0.16	0.19	0.47	0.37	0.50	0.32	0.36
Letter Recognition	3	0.24	0.59	0.92	0.62	1.11	0.71	0.66
Letter Recognition	4	0.28	1.16	1.27	0.74	1.51	1.08	0.82
Letter Recognition	5	0.29	1.36	1.40	0.77	1.61	1.17	0.86
Letter Recognition	6	0.30	1.39	1.40	0.77	1.60	1.17	0.86
Mush	1	0.03	0.03	0.06	0.06	0.05	0.04	0.04
Mush	2	0.10	0.10	0.20	0.18	0.23	0.14	0.16
Mush	3	0.20	0.33	0.51	0.40	0.78	0.39	0.41
Mush	4	0.24	0.66	0.88	0.61	1.60	0.70	0.69
Mush	5	0.25	0.82	1.10	0.72	2.13	0.87	0.80
Mush	6	0.26	0.87	1.14	0.73	2.33	0.88	0.82
Retail	1	21.01	21.37	21.32	23.68	12.61	12.80	12.51
Retail	2	22.64	23.76	25.41	25.80	16.77	17.14	15.80
Retail	3	23.19	24.55	26.95	26.94	18.71	18.73	16.69
Retail	4	23.26	24.68	27.08	27.69	18.98	18.89	16.78
Retail	5	23.30	24.66	27.12	28.94	18.98	18.86	16.82
Retail	6	23.29	24.69	27.12	29.09	18.95	18.82	16.83
Shuttle	1	0.09	0.09	0.37	0.36	0.21	0.16	0.24
Shuttle	2	0.18	0.19	0.90	0.81	0.97	0.51	0.70
Shuttle	3	0.37	0.70	1.84	1.55	2.75	1.25	1.39
Shuttle	4	0.44	1.84	2.80	2.14	4.79	2.37	1.97
Shuttle	5	0.46	2.28	3.19	2.37	5.70	2.78	2.17
Shuttle	6	0.45	2.34	3.23	2.35	5.64	2.79	2.17
Splice Junction	1	0.14	0.25	0.19	0.06	0.24	0.31	0.15
Splice Junction	2	0.25	0.61	0.37	0.15	0.37	0.60	0.32
Splice Junction	3	0.26	0.72	0.40	0.15	0.40	0.70	0.34
Splice Junction	4	0.26	0.73	0.41	0.15	0.40	0.71	0.35
Splice Junction	5	0.26	0.74	0.42	0.15	0.40	0.70	0.34
Splice Junction	6	0.26	0.74	0.42	0.14	0.40	0.70	0.34
TICDATA 2000	1	0.08	0.08	0.08	0.08	0.09	0.08	0.08
TICDATA 2000	2	0.16	0.11	0.09	0.08	0.22	0.10	0.09
TICDATA 2000	3	0.26	0.26	0.16	0.09	2.27	0.13	0.10
TICDATA 2000	4	0.33	0.63	0.22	0.10	6.88	0.13	0.10
TICDATA 2000	5	0.37	0.87	0.25	0.11	12.20	0.14	0.10
TICDATA 2000	6	0.38	0.88	0.25	0.12	17.97	0.14	0.10

HO-Unadj Cand: The number of candidate rules that passed an unadjusted significance test generated from the exploratory data under the holdout approach.

HO-Unadj Disc: The number of those candidate rules that passed the subsequent holdout evaluation.

Table 7 presents the User CPU times for each treatment in the experiments on real-world data using an AMD64 3000 939 Linux system. These results are discussed in Sect. 6.4.

References

Agrawal, R., & Srikant, R. (1995). Mining sequential patterns. In *Eleventh international conference on data engineering* (pp. 3–14). Taipei, Taiwan.

Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining associations between sets of items in massive databases. In *Proceedings of the 1993 ACM-SIGMOD international conference on management of data* (pp. 207–216). Washington, DC.

Agresti, A. (1992). A survey of exact inference for contingency tables. *Statistical Science*, *7*(1), 131–153.

Aumann, Y., & Lindell, Y. (1999). A statistical theory for quantitative association rules. In *Proceedings of the fifth ACM SIGKDD international conference on knowledge discovery and data mining (KDD-99)* (pp. 261–270).

Bastide, Y., Pasquier, N., Taouil, R., Stumme, G., & Lakhal, L. (2000). Mining minimal non-redundant association rules using frequent closed itemsets. In *First international conference on computational logic—CL 2000* (pp. 972–986). Berlin: Springer.

Bay, S. D., & Pazzani, M. J. (2001). Detecting group differences: Mining contrast sets. *Data Mining and Knowledge Discovery*, *5*(3), 213–246.

Bayardo, R. J., Jr., Agrawal, R., & Gunopulos, D. (2000). Constraint-based rule mining in large, dense databases. *Data Mining and Knowledge Discovery*, *4*(2/3), 217–240.

Benjamini, Y., & Hochberg, Y. (1995) Controlling the false discovery rate: A new and powerful approach to multiple testing. *Journal of the Royal Statistical Society Series B*, *57*, 289–300.

Benjamini, Y., & Yekutieli, D. (2001) The control of the false discovery rate in multiple testing under dependency. *The Annals of Statistics*, *29*(4), 1165–1188.

Brijs, T., Swinnen, G., Vanhoof, K., & Wets, G. (1999). Using association rules for product assortment decisions: A case study. In *Knowledge discovery and data mining* (pp. 254–260).

Brin, S., Motwani, R. & Silverstein, C. (1997). Beyond market baskets: Generalizing association rules to correlations. In J. Peckham (Ed.), *SIGMOD 1997, proceedings ACM SIGMOD international conference on management of data* (pp. 265–276). New York: ACM.

Calders, T., & Goethals, B. (2002). Mining all non-derivable frequent itemsets. In *Proceedings of the 6th European conference on principles and practice of knowledge discovery in databases, PKDD 2002* (pp. 74–85). Berlin: Springer.

Dong, G., & Li, J. (1999). Efficient mining of emerging patterns: Discovering trends and differences. In *Proceedings of the fifth ACM SIGKDD international conference on knowledge discovery and data mining (KDD-99)* (pp. 15–18). New York: ACM.

DuMouchel, W., & Pregibon, D. (2001). Empirical Bayes screening for multi-item associations. In *KDD-2001: Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 76–76). New York: ACM.

Hettich, S., & Bay, S. D. (2006). *The UCI KDD archive.* From http://kdd.ics.uci.edu. Irvine, CA: University of California, Department of Information and Computer Science.

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, *6*, 65–70.

International Business Machines. (1996). *IBM intelligent miner user's guide, version 1, release 1.*

Jaroszewicz, S., & Simovici, D. A. (2004). Interestingness of frequent itemsets using Bayesian networks as background knowledge. In R. Kohavi, J. Gehrke, & J. Ghosh (Eds.), *KDD-2004: Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 178–186). New York: ACM.

Jensen, D. D., & Cohen, P. R. (2000) Multiple comparisons in induction algorithms. *Machine Learning 38*(3), 309–338.

Johnson, R., (1984). *Elementary statistics*. Boston: Duxbury.

Klösgen, W. (1996). Explora: A multipattern and multistrategy discovery assistant. In U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (Eds.), *Advances in knowledge discovery and data mining* (pp. 249–271). Menlo Park: AAAI.

Kuramochi, M., & Karypis, G. (2001). Frequent subgraph discovery. In *Proceedings of the 2001 IEEE international conference on data mining (ICDM-01)* (pp. 313–320).

Liu, B., Hsu, W., & Ma, Y. (1999). Pruning and summarizing the discovered associations. In *Proceedings of the fifth ACM SIGKDD international conference on knowledge discovery and data mining (KDD-99)* (pp. 125–134). New York: AAAI.

Megiddo, N., & Srikant, R. (1998). Discovering predictive association rules. In *Proceedings of the fourth international conference on knowledge discovery and data mining (KDD-98)* (pp. 27–78). Menlo Park: AAAI.

Michalski, R. S. (1983). A theory and methodology of inductive learning. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (pp. 83–129). Berlin: Springer.

Newman, D. J., Hettich, S., Blake, C., & Merz, C. J. (2006). *UCI repository of machine learning databases* [Machine-readable data repository]. University of California, Department of Information and Computer Science, Irvine, CA.

Piatetsky-Shapiro, G. (1991). Discovery, analysis, and presentation of strong rules. In G. Piatetsky-Shapiro, J. Frawley (Eds.), *Knowledge discovery in databases* (pp. 229–248). Menlo Park: AAAI/MIT Press.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. San Mateo: Kaufmann.

Quinlan, J. R., & Cameron-Jones, R. M. (1995). Oversearching and layered search in empirical learning. In *IJCAI'95* (pp. 1019–1024). Los Altos: Kaufmann.

Scheffer, T. (1995). Finding association rules that trade support optimally against confidence. *Intelligent Data Analysis*, *9*(4), 381–395.

Scheffer, T., & Wrobel, S. (2002). Finding the most interesting patterns in a database quickly by using sequential sampling. *Journal of Machine Learning Research*, *3*, 833–862.

Shaffer, J. P. (1995). Multiple hypothesis testing. *Annual Review of Psychology*, *46*, 561–584.

Turney, P. D. (2000). Types of cost in inductive concept learning. In *Workshop on cost-sensitive learning at the seventeenth international conference on machine learning* (pp. 15–21). Stanford University, CA.

Webb, G. I. (1995). OPUS: An efficient admissible algorithm for unordered search. *Journal of Artificial Intelligence Research*, *3*, 431–465.

Webb, G. I. (2001). Discovering associations with numeric variables. In *Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining (KDD-2001)* (pp. 383–388). New York: The Association for Computing Machinery.

Webb, G. I. (2002). *Magnum Opus Version 1.3.* Software, G.I. Webb & Associates, Melbourne, Australia.

Webb, G. I. (2003). Preliminary investigations into statistically valid exploratory rule discovery. In *Proceedings of the Australasian data mining workshop (AusDM03)* (pp. 1–9). University of Technology, Sydney.

Webb, G. I. (2005). *Magnum Opus Version 3.0.1.* Software, G.I. Webb & Associates, Melbourne, Australia.

Webb, G. I. (2006). Discovering significant rules. In *Proceedings of the twelfth ACM SIGKDD international conference on knowledge discovery and data mining, KDD-2006.* (pp. 434–443). New York: ACM.

Webb, G. I., & Zhang, S. (2005). K-optimal rule discovery. *Data Mining and Knowledge Discovery*, *10*(1), 39–79.

Zaki, M. J. (2000). Generating non-redundant association rules. In *Proceedings of the sixth ACM SIGKDD international conference on knowledge discovery and data mining (KDD-2000)* (pp. 34–43). New York: ACM.

Zhang, H., Padmanabhan, B., & Tuzhilin, A. (2004). On the discovery of significant statistical quantitative rules. In *Proceedings of the tenth international conference on knowledge discovery and data mining (KDD-2004)* (pp. 374–383). New York: ACM.

Zheng, Z., Kohavi, R., & Mason, L. (2001). Real world performance of association rule algorithms. In *Proceedings of the seventh international conference on knowledge discovery and data mining (KDD-2001)* (pp. 401–406). New York: ACM.