OPERATIONS RESEARCH CENTER

Working Paper

A Learning Theory Framework for Sequential Event Prediction and Association Rules

by

Cynthia Rudin
Benjamin Letham
Eugene Kogan & David Madigan

OR 394-12 **August 2012**

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Journal of Machine Learning Research 1 (2000) 1-100 Submitted 4/00; Published 10/00

A Learning Theory Framework for Sequential Event Prediction and Association Rules

Cynthia Rudin RUDIN@MIT.EDU
MIT Sloan School of Management
Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139, USA

Benjamin Letham BLETHAM@MIT.EDU
Operations Research Center
Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139, USA

Eugene Kogan KOGAN.GENE@GMAIL.COM
Sourcetone
1295 5th Avenue Apt 14D
New York, NY 10029, USA

David Madigan MADIGAN@STAT.COLUMBIA.EDU
Department of Statistics
Columbia University
1255 Amsterdam Avenue
New York, NY 10027, USA

Editor:

Abstract

We present a theoretical framework for the problem of "sequential event prediction" where events in a sequence are revealed one by one, and the goal is to determine which event will next be revealed. The training set is a collection of past sequences of events. An example application is to predict which item will next be placed into a customer's online shopping cart, given his/her past purchases. In the context of this problem, algorithms based on association rules have distinct advantages over classical statistical and machine learning methods: they look at correlations based on subsets of co-occurring past events (items a and b imply item c), they can be applied to the sequential event prediction problem in a natural way, they can potentially handle the "cold start" problem where the training set is small, and they yield interpretable predictions. In this work, we present two algorithms that incorporate association rules. These algorithms can be used both for sequential event prediction and for supervised classification, and they are simple enough that they can possibly be understood by users, customers, patients, managers, etc. We provide generalization guarantees on these algorithms based on algorithmic stability analysis from statistical learning theory. We include a discussion of the strict minimum support threshold often used in association rule mining, and introduce an "adjusted confidence" measure that provides a weaker minimum support condition that has advantages over the strict minimum support.

The paper brings together ideas from statistical learning theory, association rule mining and Bayesian analysis.

Keywords: statistical learning theory, algorithmic stability, association rules, sequence prediction, associative classification

1. Introduction

We aim to predict the next event within a current event sequence, given a "sequence database" of past event sequences to learn from. Consider for instance, the data generated by a customer placing items into the virtual basket of an online grocery store such as NYC's Fresh Direct, Peapod by Stop & Shop, or Roche Bros. The customer adds items one by one into the current basket, creating a sequence of events. The customer has identified him- or herself, so that all past orders are known. After each item selection, a confirmation screen contains a small list of recommendations for items that are not already in the basket. If the store can find patterns within the customer's past purchases, it may be able to accurately recommend the next item that the customer will add to the basket. Another example is to predict each next symptom of a sick patient, given the patient's past sequence of symptoms and treatments, and a database of the timelines of symptoms and treatments for other patients. We call the problem of predicting these sequentially revealed events based on past sequences of events "sequential event prediction."

In these examples, a subset of past events (for instance, a set of ingredients for a particular recipe, or a set of symptoms associated with a particular disease) can be useful in predicting the next event. In order to make predictions using subsets of past events, we employ *association rules* (Agrawal et al., 1993). An association rule in this setting is an implication $a \rightarrow b$ (such as *lettuce and carrots* $\rightarrow$ *tomatoes*), where a is a subset of items, and b is a single item. The association rule approach has the distinct advantage in being able to directly model underlying conditional probabilities $P(b|a)$ eschewing the linearity assumptions underlying many classical supervised classification, regression, and ranking methods. Rules also yield predictive models that are interpretable, meaning that for the rule $a \rightarrow b$, it is clear that b was recommended because a is satisfied.

The association rules approach makes predictions from subsets of co-occurring past events. Using subsets may make the estimation problem much easier, because it helps avoid problems with the curse of dimensionality. For instance $P(tomatoes \mid lettuce\ and\ carrots)$ could be much easier to estimate than $P(tomatoes \mid lettuce, carrots, pears, potatoes, ketchup, eggs, bread, \text{etc.})$. This is precisely why learning algorithms created from rules can be helpful for the "cold start" problem in recommender systems, where predictions need to be made when there are not enough data available to accurately compute the full probability of a new item being purchased.

Our main contribution is a framework for sequential event prediction, including a generalization analysis for algorithms based on association rules. An important part of this analysis is how a fundamental property of a rule, namely the "support," is incorporated into the generalization bounds. The "support" of an itemset for the sequential event prediction problem is the number of times that the itemset has appeared in the sequence database. For instance, the support of *lettuce* is the number of times lettuce has been purchased in the past. Typically in association rule mining, a strict minimum support threshold con-

dition is placed on the support of itemsets within a rule, so that rules falling below the minimum support threshold are simply discarded. The idea of a condition on the support is not shared with other types of supervised learning algorithms, since they do not use subsets in the same way as when using rules. Thus a new aspect of generalization is explored in our framework in that it handles predictions created from subsets of data. In classical supervised learning paradigms, bounds scale only with the sample size, and a large sample is necessary to create a generalization guarantee. In the context of association rules, the minimum support threshold forces predictions to be made only when there are enough data. Thus, in the association rules framework, there are now two mechanisms for generalization: first a large sample, and second, a minimum support. These are separate mechanisms, in the sense that it is possible to generalize with a somewhat small sample size and a large minimum support threshold, and it is also possible to generalize with a large sample size and no support threshold. We thus derive two types of bounds: large sample bounds, which scale with the sample size, and small sample bounds, which scale with the minimum support of rules. Using both large and small sample bounds (that is, the minimum of the two bounds) gives a complete picture. The large sample bounds are of order $\mathcal{O}(\sqrt{1/m})$ as in classical analysis of supervised learning, where m denotes the number of event sequences in the database, that is, the number of past baskets ordered by the online grocery store customer.

Most of our bounds are derived using a specific notion of algorithmic stability called "pointwise hypothesis stability." The original notions of algorithmic stability were invented in the 1970's and have been revitalized recently (Devroye and Wagner, 1979; Bousquet and Elisseeff, 2002), the main idea being that algorithms may be better able to generalize if they are insensitive to small changes in the training data such as the removal or change of one training example. The pointwise hypothesis stability specifically considers the average change in loss that will occur at one of the training examples if that example is removed from the training set. Our generalization analysis uses conditions on the minimum support of rules in order to bound the pointwise hypothesis stability.

There are two algorithms considered in this work. At the core of each algorithm is a method for rank-ordering association rules where the list of possible rules is generated using the customer's past purchase history and subsets of items within the current basket. These algorithms build off of the rule mining literature that has been developing since the early 1990's (Agrawal et al., 1993) by using an application-specific rule mining method as a subroutine. Our algorithms are interpretable in two different ways: the predictive model coming out of the algorithm is interpretable, and the whole algorithm for producing the predictive model is interpretable. In other words, the algorithms are straightforward enough that they can be understood by users, customers, patients, managers, etc. Further, the rules within the predictive model can provide a simple reason to the customer why an item might be relevant, or identify that a key ingredient is missing from a particular recipe. The rules provide "IF,THEN,ELSE" conditions, and yield models of the same form as those from the expert systems literature from the early days of artificial intelligence (Jackson, 1998). Many authors have emphasized the importance of interpretability and explanation in predictive modeling (see for example the work of Madigan et al., 1997).

The first of the two algorithms considered in this work uses a fixed minimum support threshold to exclude rules whose itemsets occur rarely. Then the remaining rules are ranked

according to the "confidence," which for rule $a \rightarrow b$ is the empirical probability that b will be in the basket given that a is in the basket. The right-hand sides of the highest ranked rules will be recommended by the algorithm. However, the use of a strict minimum support threshold is problematic for several well-known reasons, for instance it is known that important rules ("nuggets," which are rare but strong rules) are often excluded by a minimum support threshold condition.

The other algorithm introduced in this work provides an alternative to the minimum support threshold, in that rules are ranked by an "adjusted" confidence, which is a simple Bayesian shrinkage estimator of the probability of a rule $P(b|a)$. The right-hand sides of rules with the highest adjusted confidence are recommended by the algorithm. For this algorithm, the generalization guarantee is smoothly controlled by a parameter K, which provides only a weak (less restrictive) minimum support condition. The key benefits of an algorithm based on the adjusted confidence are that: 1) it allows the possibility of choosing very accurate (high confidence) rules that have appeared very few times in the training set (low support), and 2) given two rules with the same or similar prediction accuracy on the training set (confidence), the rule that appears more frequently (higher support) achieves a higher adjusted confidence and is thus preferred over the other rule.

All of the bounds are tied to the measure of quality (the loss function) used for the algorithm. We would like to directly compare the performance of algorithms for various settings of the adjusted confidence's K parameter (and for the minimum support threshold θ). It is problematic to have the loss defined using the same K value as the algorithm, in that case we would be using a different method of evaluation for each setting of K, and we would not be able to directly compare performance across different settings of K. To allow a direct comparison, we select one reference value of the adjusted confidence, called K_r (r for "reference"), and the loss depends on K_r rather than on K. The bounds are written generally in terms of K_r. The special case $K_r = 0$ is where the algorithm is evaluated with respect to the confidence measure. The small sample bounds for the adjusted confidence algorithm have two terms: one that generally decreases with K (as the support increases, there is better generalization) and the other that decreases as K gets closer to K_r (better generalization as the algorithm is closer to the way it is being measured). These two terms are thus agreeing if $K_r > K$ and competing if $K_r < K$. In practice, the choice of K can be determined in several ways: K can be manually determined (for instance by the customer), it can be set using side information by "empirical Bayes" as considered by McCormick et al. (2012), or it can be set via cross-validation on an extra hold-out set.

The novel elements of the paper include: 1) the definition of a new supervised learning framework for the sequential event prediction problem, 2) generalization analysis that incorporates the use of association rules, for both classification and sequential event prediction, 3) the algorithm based on adjusted confidence, where the adjusted confidence is a Bayesian version of the confidence. The work falls in the intersection of several fields that are rarely connected: association rule mining and associative classification, supervised machine learning and generalization bounds from statistical learning theory, and Bayesian analysis.

In terms of applications, the definition of "sequential event prediction" was inspired by, but not restricted to, online grocery stores. Examples are Fresh Direct, Amazon.com grocery, and netgrocer.com. Many supermarket chains with local outlets also offer an

online shop-and-delivery option, such as Peapod (paired with Stop & Shop and Giant). Other online retailers and recommendation engines may benefit from ranking algorithms that are transparent to the user like amazon.com's "customers who purchased this also purchased that" recommender system. The same techniques used to solve the sequential event prediction problem could be used in medical applications to predict, for instance, the winners at each round of a tournament (e.g, the winners of games in a football season), or the next move of a video game player in order to design a more interesting game. The work of McCormick et al. (2012) contains a Bayesian algorithm, based on the framework introduced in this paper, for predicting conditions of medical patients in a clinical trial.

Section 2 describes the two algorithms, one based on a hard thresholding of the support (min support) and the other based on the soft thresholding (adjusted confidence). Section 3 provides the generalization analysis, Section 4 contains proofs, and Section 5 provides experimental validation. Section 6 contains a summary of relevant literature. Appendix A discusses the suitability of regression approaches for solving the sequential event prediction problem. Appendix B provides additional experimental results. Appendix C contains an additional proof.

2. Derivation of Algorithms

We assume an interface similar to that of Fresh Direct, where users add items one by one into the basket. After each selection, a confirmation screen contains a handful of recommendations for items that are not already in the customer's basket. The customer's past orders are known.

The set of items is $\mathcal{X}$, for instance $\mathcal{X}$={*apples, bananas, pears*, etc}; $\mathcal{X}$ is the set of possible events. The customer has a past history of orders S which is a collection of m baskets, $S = \{z_i\}_{i=1,\ldots,m}$, $z_i \subseteq \mathcal{X}$; S is the sequence database. The customer's current basket is usually denoted by $B \subset \mathcal{X}$; B is the current sequence. An algorithm uses B and S to find rules $a \to b$, where a is in the basket and b is not in the basket. For instance, if *salsa* and *guacamole* are in the basket B and also if *salsa, guacamole* and *tortilla chips* were often purchased together in S, then the rule (*salsa* and *guacamole*) $\to$ *tortilla chips* might be used to recommend *tortilla chips*.

The support of a, written $\mathrm{Sup}(a)$ or $\#a$, is the number of times in the past the customer has ordered itemset a,

$$\mathrm{Sup}(a) := \#a := \sum_{i=1}^{m} \mathbb{1}_{[a \subseteq z_i]}.$$

If $a = \varnothing$, meaning a contains no items, then $\#a := \sum_i 1 = m$. The confidence of a rule $a \to b$ is denoted "Conf" or "$f_{S,0}$":

$$\mathrm{Conf}(a \to b) := f_{S,0}(a, b) := \frac{\#(a \cup b)}{\#a},$$

the fraction of times b is purchased given that a is purchased. It is an estimate of the conditional probability of b given a. Ultimately an algorithm should order rules by conditional probability; however, the rules that possess the highest confidence values often have a left-hand side with small support, and their confidence values do not yield good estimates

for the true conditional probabilities. In this work we introduce the "adjusted" confidence as a remedy for this problem: The *adjusted confidence* for rule $a \to b$ is:

$$f_{S,K}(a,b) \;:=\; \frac{\#(a \cup b)}{\#a + K}.$$

The adjusted confidence for $K = 0$ is equivalent to the confidence.

The adjusted confidence is a particular Bayesian estimate of the confidence. Specifically, assuming a beta prior distribution for the confidence, the posterior mean is given by:

$$\hat{p} = \frac{L + \#(a \cup b)}{L + K + \#a},$$

where L and K denote the parameters of the beta prior distribution. The beta distribution is the "conjugate" prior distribution for a binomial likelihood. For the adjusted confidence we choose $L = 0$. This choice yields the benefits of the lower bounds derived in the remainder of this section, and the stability properties described later. The prior for the adjusted confidence tends to bias rules towards the *bottom* of the ranked list. Any rule achieving a high adjusted confidence must overcome this bias.

Other possible choices for L and K are meaningful. For instance we could choose the following:

- Collaborative filtering prior: have $L/(L + K)$ represent the probability of purchasing item b given that item a was purchased, calculated over a subset of other customers. This biases estimates of the target user's behavior towards the "average" user.
- Revenue management prior: choose L and K based on the item's price, so more expensive items are more likely to be recommended.
- Time dependent prior: use only the customer's most recent orders, and choose L and K to summarize the user's behavior before this point.

A rule cannot have a high adjusted confidence unless it has a large enough confidence and also a large enough support on the left-hand side. To see this, consider the case when we take $f_{S,K}(a,b)$ large, meaning for some η, we have $f_{S,K}(a,b) > \eta$, implying:

$$\text{Conf}(a \to b) = f_{S,0}(a,b) > \eta \frac{\#a + K}{\#a} \geq \eta,$$

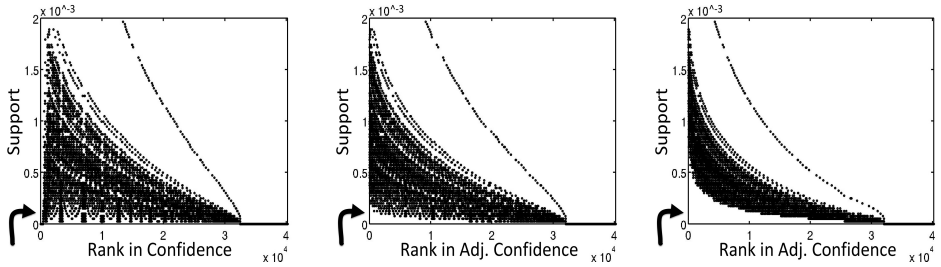
$$\text{Sup}(a) = \#a \geq (\#a + K)\left[\frac{\#(a \cup b)}{\#a + K}\right] > (\#a + K)\eta, \text{ implying } \text{Sup}(a) = \#a > \frac{\eta K}{1 - \eta}. \quad (1)$$

And further, expression (1) implies:

$$\text{Sup}(a \cup b) = \#(a \cup b) > \eta(\#a + K) > \eta K/(1 - \eta).$$

Thus, rules attaining high values of adjusted confidence have a lower bound on confidence, and a lower bound on support of both the right and left-hand sides, which means a better estimate of the conditional probability. The bounds clearly do not provide any advantage when $K = 0$ and the confidence is used.

As K increases, rules with low support are heavily penalized, so they tend not to be at the top of the list. On the other hand, such rules might be chosen when all other rules have low confidence. That is an advantage of having no firm minimum support cutoff: "nuggets" that have fairly low support may filter to the top. Figure 1 illustrates this by showing the support of rules ordered by adjusted confidence, for two values of K, using a transactional dataset "T25I10D10KN200" from the IBM Quest Market-Basket Synthetic Data Generator (Agrawal and Srikant, 1994) which mimics a retail dataset.[1] We use all rules with either one or no items on the left and one item on the right (as produced for instance by *GenRules*, presented in Algorithm 1). On each scatter plot, each of the rules is represented by a point. The rules are ordered on the x-axis by adjusted confidence, and the support of the rule is indicated on the y-axis. As K increases, rules with the highest adjusted confidence are required to achieve a higher support, as can be seen from the gap in the lower left corner of the scatter plot for larger K.



$K = 0$ $K = 10$ $K = 50$

Figure 1: Support vs. rank in adjusted confidence for $K = 0, 10, 50$. Rules with the highest adjusted confidence are on the left.

We now formally state the recommendation algorithms. Both algorithms use a subroutine for mining association rules to generate a set of candidate rules. *GenRules* (Algorithm 1) is one of the simplest such rule mining algorithms, which in practice should be replaced by a rule mining algorithm that retrieves rules tailored to the application. There is a vast literature on such algorithms since the field of association rule mining evolved on their development, *e.g.* Apriori (Agrawal et al., 1993). *GenRules* requires a set A which is the set of allowed left-hand sides of rules.

2.1 Max Confidence, Min Support Algorithm

The max confidence, min support algorithm, shown as Algorithm 2, is based on the idea of eliminating rules whose itemsets occur rarely, which is commonly done in the rule-mining literature. For this algorithm, the rules are ranked by confidence, and rules that do not achieve a predetermined fixed minimum support threshold are completely omitted. The algorithm recommends the right-hand sides from the top ranked rules. Specifically, if c

1. The dataset generated is T25I10D10KN200 that contains 10K transactions, 200 items, and where the average length of transactions is 25 and the average pattern length is 10.

Algorithm 1: *Subroutine GenRules.*

Input: $(S, B, \mathcal{X})$, that is, past orders $S = \{z_i\}_{i=1,\dots,m}$, $z_i \subseteq \mathcal{X}$, current basket $B \subset \mathcal{X}$, set of items $\mathcal{X}$

Output: Set of all rules $\{a_j \to b_j\}_j$ where b_j is a single item that is not in the basket B, and where a_j is either a subset of items in the basket B, or else it is the empty set. Also the left-hand side a_j must be allowed (meaning it is in A). That is, output rules $\{a_j \to b_j\}_j$ such that $b_j \in \mathcal{X} \backslash B$ and $a_j \subseteq B \subset \mathcal{X}$ with $a_j \in A$, or $a_j = \varnothing$.

Algorithm 2: Max Confidence, Min Support Algorithm.

Input: $(\theta, \mathcal{X}, S, B,$ *GenRules*, $c)$, that is, minimum threshold parameter θ, set of items $\mathcal{X}$, past orders $S = \{z_i\}_{i=1,\dots,m}$, $z_i \subseteq \mathcal{X}$, current basket $B \subset \mathcal{X}$, *GenRules* generates candidate rules $\mathit{GenRules}(S, B, \mathcal{X}) = \{a_j \to b_j\}_j$, number of recommendations $c \geq 1$

Output: Recommendation List, which is a subset of c items in $\mathcal{X}$

1 Apply $\mathit{GenRules}(S, B, \mathcal{X})$ to get rules $\{a_j \to b_j\}_j$ where a_j is in the basket B and b_j is not.

2 Compute score for each rule $a_j \to b_j$ as $\bar{f}_{S,\theta}(a_j, b_j) = f_{S,0}(a_j, b_j) = \frac{\#(a_j \cup b_j)}{\#a_j}$ when support $\#a_j \geq \theta$, and $\bar{f}_{S,\theta}(a_j, b_j) = 0$ otherwise.

3 Reorder rules by decreasing score.

4 Find the top c rules with distinct right-hand sides, and let Recommendation List be the right-hand sides of these rules.

items are to be recommended to the user, the algorithm picks the top ranked c distinct items.

It is common that the minimum support threshold is imposed on the right and left side $\text{Sup}(a \cup b) \geq \theta$; however, as long as $\text{Sup}(a)$ is large, we can get a reasonable estimate of $P(b|a)$. In that sense, it is sufficient (and less restrictive) to impose the minimum support threshold on the left side: $\text{Sup}(a) \geq \theta$. Here θ is a number determined beforehand (for instance, the support of the left must be at least 5 items). In this work, we only have a required minimum support on the left side. As a technical note, we might worry about the minimum support threshold being so high that there are no rules that meet the threshold. This is actually not a major concern because of the minimum support being imposed only on the left-hand side: as long as $m \geq \theta$, all rules $\varnothing \to b$ meet the minimum support threshold.

The thresholded confidence is denoted by $\bar{f}_{S,\theta}$:

$$\bar{f}_{S,\theta}(a, b) := f_{S,0}(a, b) \text{ if } \#a \geq \theta, \text{and } \bar{f}_{S,\theta}(a, b) := 0 \text{ otherwise.}$$

2.2 Adjusted Confidence Algorithm

The adjusted confidence algorithm is shown as Algorithm 3. A chosen value of K is used to compute the adjusted confidence for each rule, and rules are then ranked according to adjusted confidence.

Algorithm 3: Adjusted Confidence Algorithm.

Input: $(K, \mathcal{X}, S, B,$ *GenRules*, $c)$, that is, parameter K, set of items $\mathcal{X}$, past orders $S = \{z_i\}_{i=1,\ldots,m}$, $z_i \subseteq \mathcal{X}$, current basket $B \subset \mathcal{X}$, *GenRules* generates candidate rules $\mathit{GenRules}(S, B, \mathcal{X}) = \{a_j \to b_j\}_j$, number of recommendations $c \geq 1$

Output: Recommendation List, which is a subset of c items in $\mathcal{X}$

1 Apply $\mathit{GenRules}(S, B, \mathcal{X})$ to get rules $\{a_j \to b_j\}_j$ where a_j is in the basket B and b_j is not.

2 Compute adjusted confidence of each rule $a_j \to b_j$ as $f_{S,K}(a_j, b_j) = \frac{\#(a_j \cup b_j)}{\#a_j + K}$.

3 Reorder rules by decreasing adjusted confidence.

4 Find the top c rules with distinct right-hand sides, and let Recommendation List be the right-hand sides of these rules.

The definition of the adjusted confidence makes an implicit assumption that the order in which items were placed into previous baskets is irrelevant. It is easy to include a dependence on the order by defining a "directed" version of the adjusted confidence, and calculations can be adapted accordingly. The numerator of the adjusted confidence becomes the number of past orders where a is placed in the basket *before* b.

$$f_{S,K}^{(\text{directed})}(a, b) \quad = \quad \frac{\#\{(a \cup b) : b \text{ follows } a\}}{\#a + K}.$$

2.3 Rule Selection

In classical supervised machine learning problems, like classification and regression, designing features is one of the main engineering challenges. In association rule modeling, the analogous challenge is designing the allowed sets of items for the left and right sides of rules. For instance, we can choose to capture only positive correlations, as if customers were purchasing items from several independent recipes. The present work considers mainly positive correlations, for the purpose of exposition and to keep things simple. Beyond this, it is easily possible to capture negative correlations between items by creating "negation" items, such as $\neg b$. As an example of using negation rules in the ice cream category, we impose that for *vanilla* to be on the right, both *chocolate* and *strawberry* need to be on the left, in either their usual form or negated. Of these, the rule that is used corresponds to the current basket. In that case, $\neg$*chocolate*, $\neg$*strawberry* $\to$ *vanilla* could have a high score in order to recommend *vanilla* when *chocolate* and *strawberry* are not in the basket, whereas *chocolate*, $\neg$*strawberry* $\to$ *vanilla* might have a low score, conveying that since *chocolate* is already in the basket that *vanilla* should not be recommended. Alternatively, we could create a negation item $\neg$*ice_cream* indicating that the basket contains no ice cream presently, so *sprinkles* + $\neg$*ice_cream* $\to$ *vanilla* could have a high score.

We can also use negation items on the right, where if there is a rule $a \to \neg b$ that receives a higher score (confidence or adjusted confidence) than any other rules recommending b, we can choose not to recommend b. Rules can be designed to capture higher level correlations in specific regimes, for instance the allowed set A can contain up to three items in one product

category, but only two items in another. It is not practical in general to exhaustively enumerate and use all possible rules in a rule modeling algorithm due to problems with computational complexity. The key is to find a small but good set of rules, for instance the set of rules containing exhaustively all subsets of 1, 2, or 3 items on the left; or perhaps use the top rules that come out of the Apriori algorithm (Agrawal et al., 1993). In Section 6 we provide citations to surveys on association rule mining and associative classification that discuss this important issue of rule-construction and rule-engineering.

2.4 Modeling Assumption

The general modeling assumption that we make with the two algorithms above can be written as follows, where current basket B is composed of items $b_1, \ldots b_t$, and X_i is the random variable governing whether item i will be placed into the basket next:

$$\begin{aligned} &\operatorname*{argmax}_{\substack{i=1,\ldots,m \\ i\notin B}} P(X_i = 1 | X_{b_1} = 1, X_{b_2} = 1, \ldots, X_{b_t} = 1) \\ &= \operatorname*{argmax}_{\substack{i=1,\ldots,m \\ i\notin B}} \max_{\substack{a\in A \\ a\subseteq\{b_1,\ldots,b_t\}}} P(X_i = 1 | X_{a_1} = 1, X_{a_2} = 1, \ldots). \end{aligned}$$

This expression states that the most likely item to be added next into the basket can be identified using a subset of items in the basket, denoted a. That subset is restricted to fall into a class A which is chosen based on the application at hand and the ease in which that subset can be searched. The set A determines the hypothesis space for learning, and it would be chosen differently as we move from the small sample regime to the large sample regime, so that the right side of this expression would eventually look just like the left side when the sample is large.

The choice of A can help with the problem of "curse of dimensionality" by allowing us to look at small subsets on the left. A similar example as the one in the introduction is P(machine will break | a particular part is old) could be much easier to estimate accurately than the full probability P(machine will break | part 1 did poorly at last inspection, part 2 is very old, part 3 is new, part 4 is ok,..., part 612 is ok, etc.). The large dimensionality would likely be a problem when estimating the full probability. Further, the approximation also could actually be sufficient to estimate the full probability. We note that there are circumstances in which it is natural to only consider positive correlations. In the example of equipment failure, for instance, individual component failures would always increase the risk of overall failure. More typically, however, consideration of both positive and negative correlations will be important.

Our modeling assumption aligns with sequential event prediction, where only part of a sequence is available to make a prediction at time t. This is a case where standard linear modeling approaches do not naturally apply, since one would need to make a linear combination of terms, some of which are unrealized. We discuss this more in Appendix A.

3. Generalization

Our goal in this section is to provide a foundation for supervised learning with association rules. We will consider several quantities that may be important in the learning process: m,

K or θ, the size of the set of possible itemsets $|A|$, and the probability of the least probable itemsets and items.

We first establish bounds for vanilla supervised binary classification, for "max-score" association rule classifiers. For a given example, a max-score classifier assigns a score to the label +1 and a score to the label -1, and chooses the label corresponding to the higher of the two scores. Max-score association rule classifiers are a special type of "associative classifier" (Liu et al., 1998) and are also called "decision lists" (Rivest, 1987). The first result in this section is a uniform bound based on the VC dimension of the set of max-score classifiers. This bound does not depend explicitly on K, which we hypothesize is an important quantity for the learning process.

In order to understand how K might affect learning, we use algorithmic stability analysis. This approach originated in the 1970's (Rogers and Wagner, 1978; Devroye and Wagner, 1979) and was revitalized by Bousquet and Elisseeff (2002). Stability bounds depend on how the space of functions is searched by the algorithm (rather than the size of the function space), so it often yields more insightful bounds. These bounds are still not often directly useful due to large multiplicative constants (in our case a factor of 6), but they capture more closely the scalability relationship of a particular algorithm with respect to important quantities in the learning process. The calculation required for an algorithmic stability bound is to show that the empirical error will not dramatically change by altering or removing one of the training examples and re-running the algorithm. There are many different ways to measure the stability of an algorithm; most of the bounds presented here use a specific type of algorithmic stability (pointwise hypothesis stability) so that the bounds scale correctly with the number of training examples m.

Section 3.1 presents the uniform VC bound for classification with max-score classifiers. Section 3.2 provides notation. Section 3.3 provides stability bounds for the large sample asymptotic regime (for both sequential event prediction and classification). Then we consider the new small m regime in Section 3.4, starting with stability bounds that formally show that minimum support thresholds can lead to better generalization (for both sequential event prediction and classification). From there, we present small sample bounds for the adjusted confidence algorithm, for classification and (separately) for sequential event prediction.

We note that the space of possible baskets (up to a maximum size) is a combinatorially large, discrete space. Because the space is discrete, all probability estimates converge to the true probabilities, which means that an algorithm that is statistically consistent can be obtained by estimating $p(b|B)$ directly for the current basket B. If m is large, prediction is easy. The difficult part is when we have only enough data to well estimate conditionals that are much smaller, $P(b|a), a \subset B$. That is the problem we are concerned with. Consistency does not imply anything about generalization bounds for the finite sample case.

In sequential event prediction, if any item has a higher score than the next item added, the algorithm incurs an error. (Even if that item is added later on, the algorithm incurs an error at this timestep.) To measure the size of that error, we can use a 0-1 loss, indicating whether or not our algorithm gave the highest score to the next item added. However, the 0-1 loss does not capture how close our algorithm was to correctly predicting the next item, though this information might be useful in determining how well the algorithm will generalize. We approximate the 0-1 loss using a modified loss that decays linearly near the

discontinuity. This modified loss allows us to consider differences in adjusted confidence, not just whether one is larger than another:

$$\begin{aligned}&|(\text{adjusted conf. of highest-scoring-correct rule})\\&\quad-(\text{adjusted conf. of highest-scoring-incorrect rule})|.\end{aligned}$$

However, as discussed in the introduction, if we adjust the loss function's K value to match the adjusted confidence K value, then we cannot fairly compare the algorithm's performance using two different values of K. An illustration of this point is that for large K, all adjusted confidence values are $\ll 1$, and for small K, the adjusted confidence can be ≈ 1; differences in adjusted confidence for small K cannot be directly compared to those for large K. Since we want to directly compare performance as K is adjusted, we fix an evaluation measure that is separate from the choice of K. Specifically, we use the difference in adjusted confidence values with respect to a reference K_r:

$$\begin{aligned}&|(\{\text{adjusted conf.}\}_{K_r}\text{of highest-scoring-correct rule}_K)\\&\quad-(\{\text{adjusted conf.}\}_{K_r}\text{of highest-scoring-incorrect rule}_K)|.\end{aligned}$$

The reference K_r is a parameter of the loss function, whereas K is a parameter of an algorithm. We set $K_r = 0$ to measure loss using the difference in confidence, and $K = 0$ for an algorithm that chooses rules according to the confidence. As K gets farther from K_r, the algorithm is more distant from the way it is being evaluated, which leads to worse generalization. A similar loss will be used in classification, where we incur an error if the adjusted confidence of the incorrect label is higher than that of the correct label.

3.1 Classification with Association Rules: A Uniform Bound

In the classification problem, each basket receives a single label that is one of two possible labels $\{+1, -1\}$. This contrasts with sequential event prediction where there is a sequence of labels, one for each item in the basket as it arrives. For classification, we represent basket x as a binary vector, where entry j is 1 if item j is in the basket. We sample baskets with labels, $z = (x, y)$, where $x \in 2^{\mathcal{X}}$ is a set of items (or, equivalently, a binary feature vector) and $y \in \{-1, 1\}$ is the corresponding label. Each labeled basket z is chosen randomly (iid) from a fixed (but unknown) probability distribution $\mathcal{D}$ over baskets and labels. Given a training set S of m labeled baskets, we wish to construct a classifier that can assign the correct label to new, unlabeled baskets. We begin by defining a scoring function $g : A \times \{-1, 1\} \to \mathbb{R}$ that assigns a score $g(a, y)$ to a rule $a \to y$. The set of left-hand sides A can be any collection of itemsets so long as every $x \in 2^{\mathcal{X}}$ contains at least one $a \in A$. We define a *valid* scoring function as one where $\forall a \in A,\ g(a, 1) \neq g(a, -1)$ and $\forall a_1, a_2 \in A,\ \max_{y\in\{-1,1\}} g(a_1, y) \neq \max_{y\in\{-1,1\}} g(a_2, y)$, *i.e.*, there are no ties. The validity requirement will be discussed in the following paragraph. Define G to be the class of all valid scoring functions. We now define a class of decision functions that use a valid scoring function $g \in G$ to provide a label to a basket x, $f_g : 2^{\mathcal{X}} \to \{-1, 1\}$. The decision function assigns the label corresponding to the highest scoring rule whose left-hand side is contained in x. Specifically,

$$f_g(x) = \operatorname*{argmax}_{y\in\{-1,1\}} \max_{a\in A, a\subseteq x} g(a, y). \tag{2}$$

We call such a classifier a "max-score association rule classifier" (or "decision list") because it uses the association rule with the maximum score to perform the classification. Let $\mathcal{F}_{\text{maxscore}}$ be the class of all max-score association rule classifiers: $\mathcal{F}_{\text{maxscore}} := \{f_g : g \in G\}$. We will bound the VC dimension of class $\mathcal{F}_{\text{maxscore}}$. By definition, the VC dimension is the size of the largest set of baskets to which arbitrary labels can be assigned using some $f_g \in \mathcal{F}_{\text{maxscore}}$; it is the size of the largest set that can be shattered.

The argmax in (2) is unique because g is valid, thus there are no ties. If ties are allowed but broken randomly, arbitrary labels can be realized with some probability, for example by taking $g(a, y) = 0$ for all a and y. In this case the VC dimension can be considered to be infinite, which motivates our definition of a valid scoring function. This problem actually happens with any classification problem where function $f(x) = 0\ \forall x$ is within the hypothesis space, thereby allowing all points to sit on the decision boundary. Our definition of validity is equivalent to one in which ties are allowed but are broken deterministically using a pre-determined ordering on the rules. In practice, ties are generally broken in a deterministic way by the computer, so the inclusion of the function $f = 0$ is not problematic.

The true error of the max-score association rule classifier is the expected misclassification error:

$$\text{TrueErrClass}(f_g) := \mathbb{E}_{(x,y)\sim\mathcal{D}} \mathbb{1}_{[f_g(x)\neq y]}. \tag{3}$$

The empirical error is the average misclassification error over a training set of m baskets:

$$\text{EmpErrClass}(f_g) := \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}_{[f_g(x_i)\neq y_i]}.$$

The main result of this subsection is the following theorem, which indicates that the size of the allowed set of left-hand sides may influence generalization.

Theorem 1 *(VC Dimension for Classification)*
The VC dimension h of the set of max-score classifiers is equal to the size of the allowed set of left hand sides of rules:

$$\text{VCdim}(\mathcal{F}_{\text{maxscore}}) := h := |A|.$$

From this theorem, classical results such as those of Vapnik (1999, equations 20 and 21) can be directly applied to obtain a generalization bound:

Corollary 2 *(Uniform Generalization Bound for Classification)*
With probability at least $1-\delta$ the following holds simultaneously for all $f_g \in \mathcal{F}_{\text{maxscore}}$:

$$\text{TrueErrClass}(f_g) \leq \text{EmpErrClass}(f_g) + \frac{\epsilon}{2}\left(1 + \sqrt{1 + \frac{4\text{EmpErrClass}(f_g)}{\epsilon}}\right),$$

$$\textit{where } \epsilon = 4\frac{|A|\left(\ln\frac{2m}{|A|} + 1\right) - \ln\delta}{m}.$$

Note 1 *(on uniform bounds):* The result of Theorem 1 holds generally, well beyond the simple adjusted confidence or max confidence, min support algorithms. Those two algorithms correspond to specific choices of the scoring function g: the adjusted confidence algorithm takes $g(a,y) = f_{S,K}(a,y)$, and the max confidence, min support algorithm takes $g(a,y) = \bar{f}_{S,\theta}(a,y)$. We could use other strategies to choose g, for example, choosing $f_g \in \mathcal{F}$ to minimize an empirical risk (as we do in Letham et al., 2011).

Note 2 *(on replacing itemsets with general boolean operators):* Although in this paper we restrict our attention to left-hand sides that are sets of items (*e.g.*, "apples and oranges"), association rules can be constructed using the boolean operators AND, OR, and NOT (*e.g.*, "apples or oranges but not bananas"). In this case, the left-hand sides of rules are not contained in x, rather they are *true with respect to* x. By replacing "contained in x" with "true with respect to x" in the first half of the proof of Theorem 1, it can be seen that $h \leq |A|$ even when A contains general boolean association rules. Thus the bound in Corollary 2 extends to boolean operators.

Note 3 *(dependence on $|A|$):* We can use a standard argument involving Hoeffding's inequality and the union bound over elements of $\mathcal{F}_{\text{maxscore}}$ to obtain that with probability at least $1-\delta$, the following holds for all $f \in \mathcal{F}_{\text{maxscore}}$:

$$\text{TrueErrClass}(f_g) \leq \text{EmpErrClass}(f_g) + \sqrt{\frac{1}{2m}\left(\ln(2|\mathcal{F}_{\text{maxscore}}|) + \ln\frac{1}{\delta}\right)}.$$

The value of $|\mathcal{F}_{\text{maxscore}}|$ is at most $2^{|A|}$, since for each a there is a possibility to choose label 1 or -1 using functions from G. The bound then depends on $\sqrt{|A|}$ (as classical VC bounds would also give, using Theorem 1), but not $\log|A|$.

Note 4 *(on reducing $|A|$):* It is possible that many of the possible left-hand sides in $|A|$ are realized with zero probability. (This depends on the unknown probability distribution that the examples are drawn from.) Because of this, if we are willing to redefine A to include only realizable left-hand sides, $|A|$ can be replaced in the bound by $|\mathcal{A}|$, where $\mathcal{A} = \{a \in A : P_z(a \subseteq x) > 0\}$ are the itemsets that have some probability of being chosen.

3.2 Notation for Algorithmic Stability Bounds

We will now introduce the notation that will be used for the algorithmic stability bounds, first for classification and then for sequential event prediction.

3.2.1 Notation for Classification Bounds

Recall that we sample $z = (x,y)$ where $x \in 2^{\mathcal{X}}$ is a set of items and $y \in \{-1,1\}$ is the corresponding label. Each z is sampled randomly (iid) according to a distribution $\mathcal{D}$ over the space $2^{\mathcal{X}} \times \{-1,1\}$. The adjusted confidence algorithm uses the training set S of m iid baskets to compute the adjusted confidences $f_{S,K}$ and find a rule that will be used to label the basket. We use $z = (x,y)$ to refer to a general labeled basket, and $z_i = (x_i, y_i)$ to refer specifically to the i^{th} labeled basket in the training set. We define a *highest-scoring-correct* rule for x as a rule with the highest adjusted confidence that predicts the correct label y.

The left-hand side of a highest-scoring-correct rule obeys:

$$a^+_{SxK} \in \operatorname*{argmax}_{a\subseteq x, a\in A} f_{S,K}(a,y) = \operatorname*{argmax}_{a\subseteq x, a\in A} \frac{\#(a\cup y)}{\#a+K},$$

where $K \geq 0$. If more than one rule is tied for the maximum adjusted confidence, one can now be chosen randomly. If the true label y is not found in the training set, then the confidence of all rules with y on the right-hand side will be 0, and we take $\varnothing \to y$ as the maximizing rule. We define a *highest-scoring incorrect* rule for x as a rule with the highest adjusted confidence that predicts the incorrect label $-y$, so the left-hand side obeys:

$$a^-_{SxK} \in \operatorname*{argmax}_{a\subseteq x, a\in A} f_{S,K}(a,-y) = \operatorname*{argmax}_{a\subseteq x, a\in A} \frac{\#(a\cup -y)}{\#a+K}.$$

Again, if the label $-y$ is not found in the training set, we take $\varnothing \to -y$ as the maximizing rule. Otherwise, ties are broken randomly.

A misclassification error is made for labeled basket z when the highest-scoring-correct rule, $a^+_{SxK} \to y$, has a lower adjusted confidence than the highest-scoring incorrect rule $a^-_{SxK} \to -y$. As discussed earlier, we will measure this difference in adjusted confidence values with respect to a reference K_r in order to allow comparisons with different values of K. We will take $K_r \geq 0$. This leads to the definition of the 0-1 loss for classification:

$$\ell^{\text{class}}_{0-1,K_r}(f_{S,K}, z) := \begin{cases} 1 & \text{if } f_{S,K_r}(a^+_{SxK}, y) - f_{S,K_r}(a^-_{SxK}, -y) \leq 0 \\ 0 & \text{otherwise.} \end{cases}$$

The term $f_{S,K_r}(a^+_{SxK}, y) - f_{S,K_r}(a^-_{SxK}, -y)$ is the "margin" of example z (that is, the gap in score between the predictions for the two classes, see also Shen and Wang, 2007).

We will now define the true error which, when $K = K_r$, is a specific case of TrueErrClass defined in (3). (The function g is chosen using the dataset, and it is $f_{S,K}$.) The true error is an expectation of a loss function with respect to $\mathcal{D}$, and is a random variable since the training set S is random, $S \sim \mathcal{D}^m$.

$$\text{TrueErrClass}(f_{S,K}, K_r) := \mathbb{E}_{z\sim\mathcal{D}} \ell^{\text{class}}_{0-1,K_r}(f_{S,K}, z).$$

We approximate the true error using a different loss $\ell^{\text{class}}_{\gamma,K_r}$ that is a continuous upper bound on the 0-1 loss $\ell^{\text{class}}_{0-1,K_r}$. It is defined with respect to K_r and another real-valued parameter $\gamma > 0$ as follows:

$$\ell^{\text{class}}_{\gamma,K_r}(f_{S,K}, z) := c_\gamma(f_{S,K_r}(a^+_{SxK}, y) - f_{S,K_r}(a^-_{SxK}, -y)),$$

where $c_\gamma : \mathbb{R} \to [0,1]$,

$$c_\gamma(y) = \begin{cases} 1 & \text{for } y \leq 0 \\ 1 - y/\gamma & \text{for } 0 \leq y \leq \gamma \\ 0 & \text{for } y \geq \gamma. \end{cases}$$

As γ approaches 0, loss c_γ approaches the standard 0-1 loss. Also, $\ell^{\text{class}}_{0-1,K_r}(f_{S,K}, z) \leq \ell^{\text{class}}_{\gamma,K_r}(f_{S,K}, z)$. We define $\text{TrueErrClass}_\gamma$ using this loss:

$$\text{TrueErrClass}_\gamma(f_{S,K}, K_r) = \mathbb{E}_{z\sim\mathcal{D}} \ell^{\text{class}}_{\gamma,K_r}(f_{S,K}, z),$$

where TrueErrClass $\leq$ TrueErrClass$_\gamma$. The generalization bounds for classification will bound TrueErrClass by considering the difference between TrueErrClass$_\gamma$ and its empirical counterpart that we will soon define. For training basket x_i, the left-hand side of a highest-scoring-correct rule obeys:

$$a^+_{Sx_iK} \in \operatorname*{argmax}_{a \subseteq x_i, a \in A} f_{S,K}(a, y_i),$$

and the left-hand side of a highest-scoring-incorrect rule obeys:

$$a^-_{Sx_iK} \in \operatorname*{argmax}_{a \subseteq x_i, a \in A} f_{S,K}(a, -y_i).$$

The empirical error is an average of the loss over the baskets:

$$\text{EmpErrClass}_\gamma(f_{S,K}, K_r) := \frac{1}{m} \sum_{i=1}^{m} \ell^{\text{class}}_{\gamma,K_r}(f_{S,K}, z_i).$$

For the max confidence, min support algorithm, we substitute θ where K appears in the notation. For instance, for general labeled basket $z = (x, y)$, we analogously define:

$$\begin{aligned}
a^+_{Sx\theta} &\in \operatorname*{argmax}_{a \subseteq x, a \in A} \bar{f}_{S,\theta}(a, y) \\
a^-_{Sx\theta} &\in \operatorname*{argmax}_{a \subseteq x, a \in A} \bar{f}_{S,\theta}(a, -y) \\
\ell^{\text{class}}_{0-1,K_r}(\bar{f}_{S,\theta}, z) &= \begin{cases} 1 & \text{if } f_{S,K_r}(a^+_{Sx\theta}, y) - f_{S,K_r}(a^-_{Sx\theta}, -y) \leq 0 \\ 0 & \text{otherwise} \end{cases} \\
\ell^{\text{class}}_{\gamma,K_r}(\bar{f}_{S,\theta}, z) &= c_\gamma(f_{S,K_r}(a^+_{Sx\theta}, y) - f_{S,K_r}(a^-_{Sx\theta}, -y))
\end{aligned}$$

and TrueErrClass$(\bar{f}_{S,\theta}, K_r)$ and TrueErrClass$_\gamma(\bar{f}_{S,\theta}, K_r)$ are defined analogously as expectations of the losses, and EmpErrClass$_\gamma(\bar{f}_{S,\theta}, K_r)$ is again an average of the loss over the training baskets.

3.2.2 Notation for Sequential Event Prediction Bounds

The notation and the bounds for sequential event prediction are similar to those of classification, the main differences being an additional index t to denote the different time steps, and a set of possible incorrect recommendations in the place of the single incorrect label $-y$. For simplicity in notation, the algorithm recommends only one item, $c = 1$. A basket z consists of an ordered (permuted) set of items, $z \in 2^{\mathcal{X}} \times \Pi$, where $2^{\mathcal{X}}$ is the set of all subsets of $\mathcal{X}$, and Π is the set of permutations over at most $|\mathcal{X}|$ elements.[2] We have a training set of m baskets $S = \{z_i\}_{1\ldots m}$ that are the customer's past orders. Denote $z \sim \mathcal{D}$ to mean that basket z is drawn randomly (iid) according to distribution $\mathcal{D}$ over the space of possible items in baskets and permutations over those items, $2^{\mathcal{X}} \times \Pi$. The t^{th} item added to the basket is written $z_{\cdot,t}$, where the dot is just a placeholder for the generic basket z. The t^{th} element of the i^{th} basket in the training set is written $z_{i,t}$. We define the number of items in basket z by T_z, *i.e.,* $T_z := |z|$.

2. Even though we define an order for the basket for this discussion of prediction, we are still using the undirected adjusted confidence to make recommendations rather than the directed version introduced in Section 2. The results can be trivially extended to the directed case.

For sequential event prediction, a highest-scoring-correct rule is a highest scoring rule that has the next item $z_{\cdot,t+1}$ on the right. The left-hand side a^+_{SztK} of a highest-scoring-correct rule obeys:

$$a^+_{SztK} \in \operatorname*{argmax}_{a \subseteq \{z_{\cdot,1},\ldots,z_{\cdot,t}\}, a \in A} f_{S,K}(a, z_{\cdot,t+1}).$$

If $z_{\cdot,t+1}$ has never been purchased, the adjusted confidence for all rules $a \to z_{\cdot,t+1}$ is 0, and we choose the maximizing rule to be $\varnothing \to z_{\cdot,t+1}$. Also at time 0 when the basket is empty, the maximizing rule is $\varnothing \to z_{\cdot,t+1}$.

The algorithm incurs an error when it recommends an incorrect item. A highest-scoring-incorrect rule is a highest scoring rule that does not have $z_{\cdot,t+1}$ on the right. It is denoted $a^-_{SztK} \to b^-_{SztK}$, and obeys:

$$[a^-_{SztK}, b^-_{SztK}] \in \operatorname*{argmax}_{\substack{a \subseteq \{z_{\cdot,1},\ldots,z_{\cdot,t}\}, a \in A \\ b \in \mathcal{X} \setminus \{z_{\cdot,1},\ldots,z_{\cdot,t+1}\}}} f_{S,K}(a, b).$$

If there is more than one highest-scoring rule, one is chosen at random (with the exception that all incorrect rules are tied at zero adjusted confidence, in which case the left side is taken as $\varnothing$ and the right side is chosen randomly). At time $t = 0$, the left side is again $\varnothing$. The adjusted confidence algorithm determines a^+_{SztK}, a^-_{SztK}, and b^-_{SztK}, whereas nature chooses $z_{\cdot,t+1}$.

If the adjusted confidence of the rule $a^-_{SztK} \to b^-_{SztK}$ is larger than that of $a^+_{SztK} \to z_{\cdot,t+1}$, it means that the algorithm recommended the wrong item. The loss function below counts the proportion of times this happens for each basket, and is defined with respect to K_r.

$$\ell_{0-1,K_r}(f_{S,K}, z) := \frac{1}{T_z} \sum_{t=0}^{T_z-1} \begin{cases} 1 & \text{if } f_{S,K_r}(a^+_{SztK}, z_{\cdot,t+1}) - f_{S,K_r}(a^-_{SztK}, b^-_{SztK}) \leq 0 \\ 0 & \text{otherwise.} \end{cases}$$

If z contains all items in $\mathcal{X}$, then the recommendation for that last item is deterministic, so we remove it from the basket (that way, it doesn't count towards the loss). The true error for sequential event prediction is an expectation of the loss with respect to $\mathcal{D}$, and is again a random variable since the training set S is random.

$$\text{TrueErr}(f_{S,K}, K_r) := \mathbb{E}_{z \sim \mathcal{D}} \ell_{0-1,K_r}(f_{S,K}, z).$$

We create an upper bound for the true error by using a different loss ℓ_{γ,K_r} that is a continuous upper bound on the 0-1 loss ℓ_{0-1,K_r}. It is defined analogously to classification, with respect to K_r and c_γ:

$$\ell_{\gamma,K_r}(f_{S,K}, z) := \frac{1}{T_z} \sum_{t=0}^{T_z-1} c_\gamma(f_{S,K_r}(a^+_{SztK}, z_{\cdot,t+1}) - f_{S,K_r}(a^-_{SztK}, b^-_{SztK})).$$

It is true that $\ell_{0-1,K_r}(f_{S,K}, z) \leq \ell_{\gamma,K_r}(f_{S,K}, z)$. We define TrueErr_γ:

$$\text{TrueErr}_\gamma(f_{S,K}, K_r) := \mathbb{E}_{z \sim \mathcal{D}} \ell_{\gamma,K_r}(f_{S,K}, z),$$

where $\text{TrueErr} \leq \text{TrueErr}_\gamma$. The first set of results for sequential event prediction below bound TrueErr by considering the difference between TrueErr_γ and its empirical counterpart that we will soon define.

For the specific training basket z_i, the left-hand side $a^+_{Sz_itK}$ of a highest-scoring-correct rule at time t obeys :

$$a^+_{Sz_itK} \in \operatorname*{argmax}_{a\subseteq\{z_{i,1},\ldots,z_{i,t}\},a\in A} f_{S,K}(a, z_{i,t+1}),$$

similarly, a highest-scoring-incorrect rule for z_i at time t has:

$$[a^-_{Sz_itK}, b^-_{Sz_itK}] \in \operatorname*{argmax}_{\substack{a\subseteq\{z_{i,1},\ldots,z_{i,t}\},a\in A\\ b\in\mathcal{X}\setminus\{z_{i,1},\ldots,z_{i,t+1}\}}} f_{S,K}(a,b).$$

The empirical error is defined as:

$$\mathrm{EmpErr}_\gamma(f_{S,K}, K_r) := \frac{1}{m}\sum_{\text{baskets } i=1}^{m} \ell_{\gamma,K_r}(f_{S,K}, z_i).$$

For the max confidence, min support algorithm, we again substitute θ where K appears in the notation. For example, we define:

$$\begin{aligned}
a^+_{Szt\theta} &\in \operatorname*{argmax}_{a\subseteq\{z_{\cdot,1},\ldots,z_{\cdot,t}\},a\in A} \bar{f}_{S,\theta}(a, z_{\cdot,t+1})\\
[a^-_{Szt\theta}, b^-_{Szt\theta}] &\in \operatorname*{argmax}_{\substack{a\subseteq\{z_{\cdot,1},\ldots,z_{\cdot,t}\},a\in A\\ b\in\mathcal{X}\setminus\{z_{\cdot,1},\ldots,z_{\cdot,t+1}\}}} \bar{f}_{S,\theta}(a,b)\\
\ell_{0-1,K_r}(\bar{f}_{S,\theta}, z) &:= \frac{1}{T_z}\sum_{t=0}^{T_z-1}\begin{cases}1 & \text{if } f_{S,K_r}(a^+_{Szt\theta}, z_{\cdot,t+1}) - f_{S,K_r}(a^-_{Szt\theta}, b^-_{Szt\theta}) \le 0\\ 0 & \text{otherwise}\end{cases}\\
\ell_{\gamma,K_r}(\bar{f}_{S,\theta}, z) &:= \frac{1}{T_z}\sum_{t=0}^{T_z-1} c_\gamma(f_{S,K_r}(a^+_{Szt\theta}, z_{\cdot,t+1}) - f_{S,K_r}(a^-_{Szt\theta}, b^-_{Szt\theta})),
\end{aligned}$$

$\mathrm{TrueErr}(\bar{f}_{S,\theta}, K_r)$ and $\mathrm{TrueErr}_\gamma(\bar{f}_{S,\theta}, K_r)$ are expectations of the losses, and $\mathrm{EmpErr}_\gamma(\bar{f}_{S,\theta}, K_r)$ is an average of the loss over the training baskets.

3.3 Generalization Analysis for Large m

The choice of minimum support threshold θ or the choice of parameter K matters mainly in the regime where m is small. For the max confidence, min support algorithm, when m is large, then all (realizable) itemsets have appeared more times than the minimum support threshold with high probability. For the adjusted confidence algorithm, when m is large, prediction ability is guaranteed as follows.

Theorem 3 *(Generalization Bound for Adjusted Confidence Algorithm, Large m) For set of rules A, $K \ge 0$, $K_r \ge 0$, with probability at least $1-\delta$ (with respect to training set $S \sim \mathcal{D}^m$),*

$$\mathrm{TrueErr}(f_{S,K}, K_r) \le \mathrm{EmpErr}_\gamma(f_{S,K}, K_r) + \sqrt{\frac{1}{\delta}\left[\frac{1}{2m} + 6\beta\right]}$$

$$\textit{where } \beta = \frac{2|\mathcal{A}|}{\gamma}\left[\frac{1}{(m-1)p_{\min A} + K} + \frac{|K_r - K|\frac{m}{m+K}}{(m-1)p_{\min A} + K_r}\right] + \mathcal{O}\left(\frac{1}{m^2}\right),$$

and where $\mathcal{A} = \{a \in A : P_z(a \subseteq z) > 0\}$ *are the itemsets that have some probability of being chosen. Out of these, any itemset that is the least likely to be chosen has probability* $p_{\min A}$*:*

$$p_{\min A} := \min_{a \in \mathcal{A}} P_{z \sim \mathcal{D}}(a \subseteq z).$$

As a corollary, the same result holds for classification, replacing $\text{TrueErr}(f_{S,K}, K_r)$ with $\text{TrueErrClass}(f_{S,K}, K_r)$ and $\text{EmpErr}_\gamma(f_{S,K}, K_r)$ with $\text{EmpErrClass}_\gamma(f_{S,K}, K_r)$.

A special case is where $K_r = K = 0$: the algorithm chooses the rule with maximum confidence, and accuracy is then judged by the difference in confidence values between the highest-scoring-incorrect rule and the highest-scoring-correct rule. The bound reduces to:

Corollary 4 *(Generalization Bound for Maximum Confidence Setting, Large* m*)*
With probability at least $1 - \delta$ *(with respect to* $S \sim \mathcal{D}^m$*),*

$$\text{TrueErr}(f_{S,0}, 0) \leq \text{EmpErr}_\gamma(f_{S,0}, 0) + \sqrt{\frac{1}{\delta}\left[\frac{1}{2m} + \frac{12|\mathcal{A}|}{\gamma(m-1)p_{\min A}}\right] + \mathcal{O}\left(\frac{1}{m^2}\right)}.$$

Again the result holds for classification with appropriate substitutions. The use of the pointwise hypothesis stability within this proof is the key to providing a decay of order $\sqrt{(1/m)}$. Now that this bound is established, we move to the small sample case, where the minimum support is the force that provides generalization.

3.4 Generalization Analysis for Small m

The first small sample result is a general bound for the max confidence, min support algorithm, which holds for both classification and sequential event prediction. The max confidence, min support algorithm has "uniform stability," which is a stronger kind of stability than pointwise hypothesis stability. This result strengthens the one in the conference version of this work (Rudin et al., 2011), where we used the bound for pointwise hypothesis stability; uniform stability implies pointwise hypothesis stability, so the result in the conference version follows automatically.

Theorem 5 *(Generalization Bound for Max Confidence, Min Support)*
For $\theta \geq 1$*,* $K_r \geq 0$*, with probability at least* $1 - \delta$ *(with respect to* $S \sim \mathcal{D}^m$*),* $m > \theta$*,*

$$\text{TrueErr}(\bar{f}_{S,\theta}, K_r) \leq \text{EmpErr}_\gamma(\bar{f}_{S,\theta}, K_r) + 2\beta + (4m\beta + 1)\sqrt{\frac{\ln 1/\delta}{2m}}$$
$$\text{where } \beta = \frac{2}{\gamma}\left[\frac{1}{\theta} + K_r\left(\frac{1}{\theta + K_r}\right)\left(1 + \frac{1}{\theta}\right)\right].$$

Note that $|A|$ does not appear in the bound. For classification, $\text{TrueErr}(\bar{f}_{S,\theta}, K_r)$ is replaced by $\text{TrueErrClass}(\bar{f}_{S,\theta}, K_r)$ and $\text{EmpErr}_\gamma(\bar{f}_{S,\theta}, K_r)$ is replaced by $\text{EmpErrClass}_\gamma(\bar{f}_{S,\theta}, K_r)$. Figure 2 shows β as a function of θ for several different values of K_r. The special case of interest is when $K_r = 0$, so that the loss is judged with respect to differences in confidence, as follows:

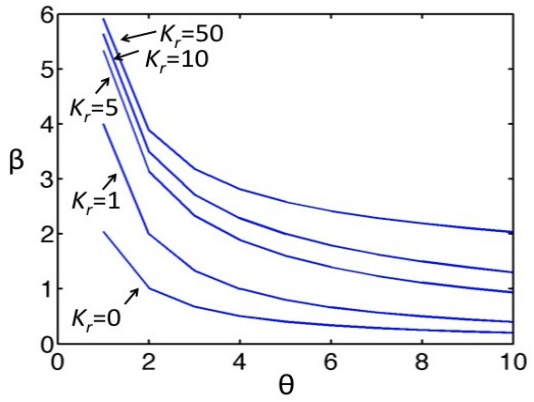


Figure 2: β vs. θ from Theorem 5, with $\gamma = 1$. The different curves are different values of $K_r = 0$, 1, 5, 10, 50 from bottom to top.

Corollary 6 *(Generalization Bound for Max Confidence, Min Support, $K_r = 0$) For $\theta \geq 1$, with probability at least $1 - \delta$ (with respect to $S \sim \mathcal{D}^m$), $m > \theta$,*

$$\text{TrueErr}(\bar{f}_{S,\theta}, 0) \leq \text{EmpErr}_\gamma(\bar{f}_{S,\theta}, 0) + \frac{4}{\gamma\theta} + \left(\frac{8m}{\gamma\theta} + 1\right)\sqrt{\frac{\ln 1/\delta}{2m}}.$$

It is common to use a minimum support threshold that is a fraction of m, for instance, $\theta = 0.1 \times m$. In that case, the bound again scales with $\sqrt{(1/m)}$. Note that there is no generalization guarantee when $\theta = 0$; the minimum support threshold enables generalization in the small m case.

Now we discuss the adjusted confidence algorithm for small m setting. We present separate small sample bounds for classification and sequential event prediction.

Theorem 7 *(Generalization Bound for Adjusted Confidence Algorithm, Small m, For Classification Only) For $K > 0, K_r \geq 0$, with probability at least $1 - \delta$,*

$$\text{TrueErrClass}(f_{S,K}, K_r) \leq \text{EmpErrClass}_\gamma(f_{S,K}, K_r) + \sqrt{\frac{1}{\delta}\left[\frac{1}{2m} + 6\beta\right]} \text{ where}$$

$$\begin{aligned}\beta &= \frac{2}{\gamma}\frac{1}{K}\left(1 - \frac{(m-1)p_{y,\min}}{m+K}\right) \\ &\quad + \frac{2}{\gamma}|K_r - K|\mathbb{E}_{\zeta\sim\text{Bin}(m-1,p_{y,\min})}\left[\frac{1}{K\left(\frac{\zeta}{m+K-\zeta}\right) + K_r}\left(\frac{m}{m+K} + \frac{1}{K}\left(1 - \frac{\zeta}{m+K}\right)\right)\right],\end{aligned}$$

where $p_{y,\min} = \min(P(y=1), P(y=-1))$ is the probability of the less popular label.

Again, $|A|$ does not appear in the bound, and generalization is provided by K, and the difference between K and K_r; the interpretation will be further discussed after we state the small sample bound for sequential event prediction.

In the proof of the following theorem, if we were to use the definitions established in Section 3.2.2, the bound does not simplify beyond a certain point and is difficult to read at an intuitive level. From that bound, it would not be easy to see what are the important quantities for the learning process, and how they scale. In what follows, we redefine the loss function slightly, so that it approximates a 0-1 loss from below instead of from above. This provides a concise and intuitive bound.

Define a *highest-scoring* rule $a^*_{SztK} \to b^*_{SztK}$ as a rule that achieves the maximum adjusted confidence, over all of the possible rules. It will either be equal to $a^+_{SztK} \to z_{\cdot,t+1}$ or $a^-_{SztK} \to b^-_{SztK}$, depending on which has the larger adjusted confidence:

$$[a^*_{SztK}, b^*_{SztK}] \in \operatorname*{argmax}_{\substack{a \subseteq \{z_{\cdot,1},\ldots,z_{\cdot,t}\}, a \in A \\ b \in \mathcal{X} \setminus \{z_{\cdot,1},\ldots,z_{\cdot,t}\}}} f_{S,K}(a,b).$$

Note that b^*_{SztK} can be equal to $z_{\cdot,t+1}$ whereas b^-_{SztK} cannot. The notation for $a^*_{Sz_itK}$ and $b^*_{Sz_itK}$ is similar, and the new loss is:

$$\ell^{\text{new}}_{0-1,K_r}(f_{S,K}, z) := \frac{1}{T_z} \sum_{t=0}^{T_z-1} \begin{cases} 1 & \text{if } f_{S,K_r}(a^+_{SztK}, z_{\cdot,t+1}) - f_{S,K_r}(a^*_{SztK}, b^*_{SztK}) < 0 \\ 0 & \text{otherwise.} \end{cases}$$

By definition, the difference $f_{S,K_r}(a^+_{SztK}, z_{\cdot,t+1}) - f_{S,K_r}(a^*_{SztK}, b^*_{SztK})$ can never be strictly positive. The continuous approximation is:

$$\ell^{\text{new}}_{\gamma,K_r}(f_{S,K}, z) := \frac{1}{T_z} \sum_{t=0}^{T_z-1} c^{\text{new}}_\gamma(f_{S,K_r}(a^+_{SztK}, z_{\cdot,t+1}) - f_{S,K_r}(a^*_{SztK}, b^*_{SztK})), \text{ where}$$

$$c^{\text{new}}_\gamma(y) = \begin{cases} 1 & \text{for } y \le -\gamma \\ -y/\gamma & \text{for } -\gamma \le y \le 0 \\ 0 & \text{for } y \ge 0. \end{cases}$$

As γ approaches 0, the c_γ loss approaches the 0-1 loss. We define $\text{TrueErr}^{\text{new}}_\gamma$ and $\text{EmpErr}^{\text{new}}_\gamma$ using this loss: $\text{TrueErr}^{\text{new}}_\gamma(f_{S,K}, K_r) := \mathbb{E}_{z\sim\mathcal{D}}\ell^{\text{new}}_{\gamma,K_r}(f_{S,K}, z)$, and $\text{EmpErr}^{\text{new}}_\gamma(f_{S,K}, K_r) := \frac{1}{m}\sum_{i=1}^m \ell^{\text{new}}_{\gamma,K_r}(f_{S,K}, z_i)$.

The minimum support threshold condition we used in Theorem 5 is replaced by a weaker condition on the support. This weaker condition has the benefit of allowing more rules to be used in order to achieve a better empirical error; however, it is more difficult to get a generalization guarantee. This support condition is derived from the fact that the adjusted confidence of the highest-scoring rule $a^*_{Sz_itK} \to b^*_{Sz_itK}$ exceeds that of the highest-scoring-correct rule $a^+_{Sz_itK} \to z_{i,t+1}$, which exceeds that of the marginal rule $\varnothing \to z_{i,t+1}$:

$$\frac{\#a^*_{Sz_itK}}{\#a^*_{Sz_itK} + K} \ge \frac{\#(a^*_{Sz_itK} \cup b^*_{Sz_itK})}{\#a^*_{Sz_itK} + K} \ge \frac{\#(a^+_{Sz_itK} \cup z_{i,t+1})}{\#a^+_{Sz_itK} + K} \ge \frac{\#z_{i,t+1}}{m + K}. \tag{4}$$

This leads to a lower bound on the support $\#a^*_{Sz_itK}$:

$$\#a^*_{Sz_itK} \ge K\left(\frac{\#z_{i,t+1}}{m + K - \#z_{i,t+1}}\right). \tag{5}$$

This is not a hard minimum support threshold, yet since the support generally increases as K increases, the bound will give a better guarantee for large K. Note that in the original notation, we would replace the condition (4) with $\frac{\#a\tilde{}_{Sz_itK}}{\#a\tilde{}_{Sz_itK}+K} \geq \frac{\#(a\tilde{}_{Sz_itK} \cup b\tilde{}_{Sz_itK})}{\#a\tilde{}_{Sz_itK}+K} \geq \frac{\#b\tilde{}_{Sz_itK}}{m+K}$ and proceed with analogous steps in the proof.

Theorem 8 *(Generalization Bound for Adjusted Confidence Algorithm, Small m) For $K > 0, K_r \geq 0$, with probability at least $1-\delta$,*

$$\text{TrueErr}_\gamma^{\text{new}}(f_{S,K}, K_r) \leq \text{EmpErr}_\gamma^{\text{new}}(f_{S,K}, K_r) + \sqrt{\frac{1}{\delta}\left[\frac{1}{2m} + 6\beta\right]} \text{ where}$$

$$\begin{aligned}\beta &= \frac{2}{\gamma}\frac{1}{K}\left(1 - \frac{(m-1)p_{\min}}{m+K}\right) \\ &\quad + \frac{2}{\gamma}|K_r - K|\mathbb{E}_{\zeta\sim\text{Bin}(m-1,p_{\min})}\frac{1}{K\left(\frac{\zeta}{m+K-\zeta-1}\right)+K_r}\left(\frac{m}{m+K} + \frac{1}{K}\left(1 - \frac{\zeta}{m+K}\right)\right),\end{aligned}$$

and where $Q = \{x \in \mathcal{X} : P_{z\sim D}(x \in z) > 0\}$ are the items that have some probability of being chosen by the customer. Out of these, any item that is the least likely to be chosen has probability $p_{\min} := \min_{x\in Q} P_{z\sim\mathcal{D}}(x \in z)$.

The stability β has two main terms. The first term decreases generally as $1/K$. The second term arises from the error in measuring loss with K_r rather than K. In order to interpret β, consider the following approximation to the expectation in the bound, which assumes that m is large and that $m \gg K \gg 0$, and that $\zeta \approx mp_{\min}$:

$$\beta \approx \frac{2}{\gamma}\frac{1}{K}\left(1 - \frac{(m-1)p_{\min}}{m+K}\right) + \frac{2}{\gamma}|K_r - K|\frac{1}{K\frac{p_{\min}}{1-p_{\min}}+K_r}. \tag{6}$$

Intuitively, if either K is close to K_r or $p_{\min}$ is large (close to 1) then this term becomes small. Figure 3 shows an example plot of β and the approximation using (6), which we denote by β_{Approx}.

One can observe that if $K_r > K$, then both terms tend to improve (decrease) with increasing K. When $K_r < K$, then the two terms can compete as K increases.

3.5 Summary of Bounds

We have provided probabilistic guarantees on performance that show the following: 1) For large m, the association rule-based algorithms have a performance guarantee of the same order as other bounds for supervised learning. 2) For small m, the minimum support threshold guarantees generalization (at the expense of possibly removing important rules). 3) The adjusted confidence provides a weaker support threshold, allowing important rules to be used, while still being able to generalize. 4) All generalization guarantees depend on the way the goodness of the algorithm is measured (the choice of K_r in the loss function). 5) Important quantities in the learning process may include: $|A|$ or $|\mathcal{A}|$, K or θ, $p_{\min A}$ or $p_{\min}$ (or $p_{y,\min}$).

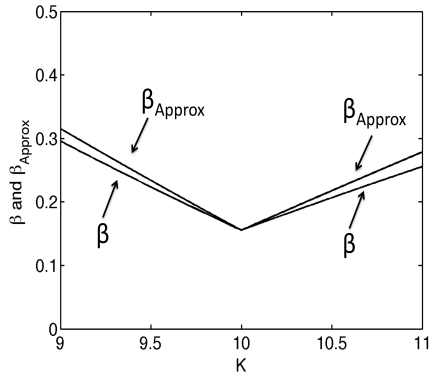


Figure 3: β and β_{Approx} vs. K, where $K_r = 10$, $p_{\min} = 0.3$, $m = 20$, $\gamma = 1$.

4. Proofs

In this section, we prove all results from Section 3.

Proof *(Of Theorem 1)* First we show that $h \leq |A|$. To do this, we must show that for any collection of baskets $x_1, \ldots, x_N$, $N > |A|$, there exists a corresponding set of labels $y_1, \ldots, y_N$ that cannot be realized by any max-score association rule classifier. For each x_i, we introduce a vector $\bar{x}_i$ of length $|A|$, where each element corresponds to an $a \in A$. The element of $\bar{x}_i$ corresponding to a is 1 if $a \subseteq x_i$ and 0 otherwise. Each vector $\bar{x}_i$ is an element of $\mathbb{R}^{|A|}$, so the collection of vectors $\bar{x}_1, \ldots, \bar{x}_N$ must be linearly dependent if $N > |A|$. By linear dependence and the fact that every $\bar{x}_i$ is non-zero and non-negative, there must exist coefficients c_i and disjoint, non-empty sets M_0 and M_1 such that:

$$\sum_{i \in M_0} c_i \bar{x}_i = \sum_{i \in M_1} c_i \bar{x}_i, \; c_i > 0. \tag{7}$$

Define $A_0 = \{a \in A : a \subseteq x_i \text{ for some } i \in M_0\}$ and $A_1 = \{a \in A : a \subseteq x_i \text{ for some } i \in M_1\}$. If $a \subseteq x_i$ for some $i \in M_0$, then the corresponding element of $\bar{x}_i$ will be 1 and the same element in the left part of (7) will be strictly positive. Then, (7) implies that $a \subseteq x_j$ for some $j \in M_1$. Thus, $A_0 \subseteq A_1$, and the reverse argument shows $A_1 \subseteq A_0$, so $A_0 = A_1$. There exists a left-hand side with maximum score, $a^* = \arg\max_{a \in A_0} \max_{y \in \{-1,1\}} g(a, y) = \arg\max_{a \in A_1} \max_{y \in \{-1,1\}} g(a, y)$. The label assigned to x_i, where i is in M_0 or M_1 and x_i contains itemset a^*, is $y^* = \arg\max_{y \in \{-1,1\}} g(a^*, y)$. Thus for at least one $i \in M_0$ and at least one $j \in M_1$, $f_g(x_i) = y^* = f_g(x_j)$. Set $y_i = -1$ for all $i \in M_0$ and $y_i = 1$ for all $i \in M_1$ and this set of labels cannot be realized, which shows that $h \leq |A|$.

We now show that this upper bound can be achieved by providing a set of $|A|$ baskets and finding elements of $\mathcal{F}_{\text{maxscore}}$ that can assign them arbitrary labels. Specifically, we list the elements of A as $a_1, \ldots, a_{|A|}$ and take $x_i = a_i$, for $i = 1, \ldots, |A|$. Thus each basket is one of the left-hand sides from the allowed set. The elements of A are not all the same size, and some elements of A may contain other elements; this could cause problems when we are constructing a max-score classifier that uniquely assigns a given label to each basket. To get around this, we will place the elements of A in order of increasing size. The possible sizes of elements of A are denoted $l_1, \ldots, l_L$, so that $l_1 < l_2 < \ldots < l_L$. We arrange the elements of A into sets based on their sizes: $S_k = \{i : |a_i| = l_k\}$, $k = 1, 2, \ldots, L$. We are

now ready to construct a classifier f_g so that, given an arbitrary set of labels $\{y_i\}_i$, it can label the x_i's according to the y_i's. For all $i \in S_1$, we set $g(a_i, y_i) = c_1$, any positive number, and $g(a_i, -y_i) = 0$. Thus, for the corresponding x_i, $f_g(x_i) = y_i$. Similarly, for all $i \in S_2$, we set $g(a_i, y_i) = c_2$, $c_2 > c_1$, and $g(a_i, -y_i) = 0$. For any $i \in S_2$, it may be that there exists some $j \in S_1$ such that $a_j \subset x_i$. However, because $c_2 > c_1$, the rule with the maximum score will be "$a_i \rightarrow y_i$" and x_i is labeled as desired. In general, for any $i \in S_k$, we set $g(a_i, y_i) = c_k$, where $c_{k-1} < c_k < c_{k+1}$ and $g(a_i, -y_i) = 0$ to get $f_g(x_i) = y_i$. Because this set of $|A|$ examples can be arbitrarily labeled using elements of $\mathcal{F}_{\text{maxscore}}$, we have $h \geq |A|$, which combined with the previous result shows that $h = |A|$. ■

The remaining theorems are based on the algorithmic stability bounds of Bousquet and Elisseeff (2002) (B&E). Many of the proofs that we provide for classification are essentially identical to those for sequential event prediction. In these cases, the proofs are given for sequential event prediction, and afterwards the translation to classification is outlined. The proofs follow this outline: first, we show how differences in adjusted confidence values with respect to K_r can be translated into differences with respect to K (Lemma 11). Then we bound the difference in adjusted confidence values (Lemma 12) in terms of the support. Various lower bounds on the support are used to obtain stability for each of the separate cases: large m (Theorem 3), small m for the max confidence, min support algorithm (Theorem 5, which uses uniform stability), small m for classification with the adjusted confidence algorithm (Theorem 7), and small m for sequential event prediction with the adjusted confidence algorithm (Theorem 8).

Following notation of Bousquet and Elisseeff (2002), the input space and output space are X and Y. Their training set is $S \in \bar{Z}^m$, $S = \{\bar{z}_1 = (x_1, y_1), \ldots, \bar{z}_m = (x_m, y_m)\}$. An algorithm is a function A from $\bar{Z}^m$ into $\mathcal{F} \subset Y^X$ which maps a learning set S onto a function A_S from X to Y. The loss is $\ell(f, \bar{z}) = c(f(x), y)$, where $c : Y \times Y \rightarrow \mathbb{R}_+$. $S^{/i}$ means to exclude the i^{th} example $\bar{z}_i$. B&E assume that $Y \subset \mathbb{R}$ but we believe this assumption is unnecessary. In any case, Y is empty for sequential event prediction. An algorithm A has *pointwise hypothesis stability* β with respect to the loss function ℓ if the following holds:

$$\forall\, i \in \{1, \ldots, m\},\ \mathbb{E}_{S \sim \mathcal{D}^m}[|\ell(A_S, \bar{z}_i) - \ell(A_{S/i}, \bar{z}_i)|] \leq \beta.$$

An algorithm A has *uniform stability* β with respect to the loss function ℓ if the following holds:

$$\forall\, S \in \bar{Z}^m, \forall\, i \in \{1, \ldots, m\},\ ||\ell(A_S, \cdot) - \ell(A_{S/i}, \cdot)||_\infty \leq \beta.$$

The empirical error is defined by:

$$R_{emp}(A, S) := \frac{1}{m} \sum_{i=1}^{m} \ell(A_S, \bar{z}_i)$$

and the true error is:

$$R(A, S) := \mathbb{E}_{\bar{z}}[\ell(A_S, \bar{z})].$$

We will use the following results that are based on ideas of Devroye and Wagner (1979).

Theorem 9 *(B&E Pointwise Hypothesis Stability Bound)(Bousquet and Elisseeff, 2002, Theorem 11, first part)*
For any learning algorithm A with pointwise hypothesis stability β with respect to a loss function ℓ, such that the value of ℓ is at most M, we have with probability $1-\delta$,

$$R(A,S) \leq R_{emp}(A,S) + \sqrt{\frac{M^2 + 12Mm\beta}{2m\delta}}.$$

Theorem 10 *(B&E Uniform Stability Bound)(Bousquet and Elisseeff, 2002, Theorem 12, first part)*
For any learning algorithm A with uniform stability β with respect to a loss function ℓ, such that the value of ℓ is at most M, we have with probability $1-\delta$ over a random draw of S,

$$R \leq R_{emp} + 2\beta + (4m\beta + M)\sqrt{\frac{\ln 1/\delta}{2m}}.$$

Translating B&E's notation to the adjusted confidence setting for sequential event prediction, $\bar{z}_i = x_i = z_i$, with $z_i \in 2^{\mathcal{X}} \times \Pi$. For our problem, $f(x_i)$ is the value of the loss and the y_i's are not defined. In other words, $\ell(A_S, \bar{z}_i) = c(f(x_i), y_i) = f(x_i)$ which in our notation is equal to $\ell_{\gamma,K_r}(f_{S,K}, z_i)$. For the max confidence, min support setting, $\ell(A_S, \bar{z}_i)$ translates to $\ell_{\gamma,K_r}(\bar{f}_{S,\theta}, z_i)$. The adjusted confidence is bounded by 1 so $M = 1$.

The following lemma allows us to convert differences in adjusted confidence with respect to K_r into differences with respect to K.

Lemma 11 *(Conversion of Adjusted Confidence) For $K \geq 0$, $K_r \geq 0$, $0 \leq s_1 \leq S_1$, $0 \leq s_2 \leq S_2$*

$$\left|\frac{s_1}{S_1+K_r} - \frac{s_2}{S_2+K_r}\right| \leq \left|\frac{s_1}{S_1+K} - \frac{s_2}{S_2+K}\right|\left(1 + \frac{|K_r - K|}{S_1+K_r}\right) + \left(\frac{|K_r-K|}{\tilde{S}+K_r}\right)\left(\frac{s_2}{S_2+K}\right)$$

where $\tilde{S} = \min(S_1, S_2)$.

Proof

$$\begin{aligned}
&\left|\frac{s_1}{S_1+K_r} - \frac{s_2}{S_2+K_r}\right| \\
&= \left|\frac{s_1}{S_1+K} - \frac{s_2}{S_2+K} + (-K_r+K)\left[\frac{s_1}{S_1+K}\left(\frac{1}{S_1+K_r}\right) - \frac{s_2}{S_2+K}\left(\frac{1}{S_2+K_r}\right)\right]\right| \\
&\leq \left|\frac{s_1}{S_1+K} - \frac{s_2}{S_2+K}\right| + |K_r-K|\left|\frac{s_1}{S_1+K}\left(\frac{1}{S_1+K_r}\right) - \frac{s_2}{S_2+K}\left(\frac{1}{S_2+K_r}\right)\right|. \quad (8)
\end{aligned}$$

Taking just the second absolute value term:

$$\begin{aligned}
&\left|\frac{s_1}{S_1+K}\left(\frac{1}{S_1+K_r}\right) - \frac{s_2}{S_2+K}\left(\frac{1}{S_2+K_r}\right)\right| \\
&= \left|\frac{s_1}{S_1+K}\left(\frac{1}{S_1+K_r}\right) - \frac{s_2}{S_2+K}\left(\frac{1}{S_1+K_r}\right) + \frac{s_2}{S_2+K}\left(\frac{1}{S_1+K_r}\right) - \frac{s_2}{S_2+K}\left(\frac{1}{S_2+K_r}\right)\right| \\
&\leq \left|\frac{s_1}{S_1+K} - \frac{s_2}{S_2+K}\right|\left|\frac{1}{S_1+K_r}\right| + \frac{s_2}{S_2+K}\left|\frac{1}{S_1+K_r} - \frac{1}{S_2+K_r}\right| \\
&\leq \left|\frac{s_1}{S_1+K} - \frac{s_2}{S_2+K}\right|\left|\frac{1}{S_1+K_r}\right| + \frac{s_2}{S_2+K}\left|\frac{1}{\tilde{S}+K_r}\right|.
\end{aligned}$$

Putting this back into (8) yields the statement. ■

The next results bound the difference in the highest adjusted confidence values when the basket z_i is removed from S. We require some additional notation in order to exclude basket i. Denote $\#^{/i}a$ to be the number of times a has appeared in $S^{/i}$, that is, $\#^{/i}a = \sum_{i' \neq i} \mathbb{1}_{[a \in z_{i'}]}$. For sequential event prediction, the left-hand side of a highest-scoring-correct rule for a general basket z on $S^{/i}$ obeys:

$$a^+_{S^{/i}ztK} \in \operatorname*{argmax}_{a \subseteq \{z_{\cdot,1},\ldots,z_{\cdot,t}\}, a \in A} f_{S^{/i},K}(a, z_{\cdot,t+1}) = \operatorname*{argmax}_{a \subseteq \{z_{\cdot,1},\ldots,z_{\cdot,t}\}, a \in A} \frac{\#^{/i}(a \cup z_{\cdot,t+1})}{\#^{/i}a + K}.$$

A highest-scoring-incorrect rule for basket z on $S^{/i}$ obeys:

$$[a^-_{S^{/i}ztK}, b^-_{S^{/i}ztK}] \in \operatorname*{argmax}_{\substack{a \subseteq \{z_{\cdot,1},\ldots,z_{\cdot,t}\}, a \in A \\ b \in \mathcal{X} \setminus \{z_{\cdot,1},\ldots,z_{\cdot,t+1}\}}} f_{S^{/i},K}(a, b) = \operatorname*{argmax}_{\substack{a \subseteq \{z_{\cdot,1},\ldots,z_{\cdot,t}\}, a \in A \\ b \in \mathcal{X} \setminus \{z_{\cdot,1},\ldots,z_{\cdot,t+1}\}}} \frac{\#^{/i}(a \cup b)}{\#^{/i}a + K}.$$

In Lemma 12 below, we bound the difference in adjusted confidence of a general basket z when z_i is removed from the training set, in the sequential event prediction setting.

Lemma 12 *(Difference in Adjusted Confidence)*
Define $\tilde{a}_z := \min(\#a^-_{SztK}, \#^{/i}a^-_{S^{/i}ztK})$ *and* $\hat{a}_z := \min(\#a^+_{SztK}, \#^{/i}a^+_{S^{/i}ztK})$. *Then,*

$$\textit{(I)} \quad |f_{S,K}(a^-_{SztK}, b^-_{SztK}) - f_{S^{/i},K}(a^-_{S^{/i}ztK}, b^-_{S^{/i}ztK})| \leq \frac{1}{\tilde{a}_z + K}, \text{ and}$$
$$\textit{(II)} \quad |f_{S,K}(a^+_{SztK}, z_{\cdot,t+1}) - f_{S^{/i},K}(a^+_{S^{/i}ztK}, z_{\cdot,t+1})| \leq \frac{1}{\hat{a}_z + K}.$$

Proof Any itemset a is either in z_i or not, thus $\#^{/i}a \geq \#a - 1$ and $\#^{/i}a \leq \#a$. Also the number of times we see $a \cup b$ is less than or equal to the number of times we see a. These observations lead to the following inequalities that will be used throughout the proof:

$$\begin{aligned}
\#^{/i}(a^-_{SztK} \cup b^-_{SztK}) &\geq \#(a^-_{SztK} \cup b^-_{SztK}) - 1 && (9)\\
\#^{/i}a^-_{SztK} &\leq \#a^-_{SztK} && (10)\\
\#(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK}) &\geq \#^{/i}(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK}) && (11)\\
\#a^-_{S^{/i}ztK} &\leq \#^{/i}a^-_{S^{/i}ztK} + 1 && (12)\\
\#^{/i}(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK}) &\leq \#^{/i}a^-_{S^{/i}ztK} && (13)\\
\#^{/i}(a^+_{SztK} \cup z_{\cdot,t+1}) &\geq \#(a^+_{SztK} \cup z_{\cdot,t+1}) - 1 && (14)\\
\#^{/i}a^+_{SztK} &\leq \#a^+_{SztK} && (15)\\
\#(a^+_{S^{/i}ztK} \cup z_{\cdot,t+1}) &\geq \#^{/i}(a^+_{S^{/i}ztK} \cup z_{\cdot,t+1}) && (16)\\
\#a^+_{S^{/i}ztK} &\leq \#^{/i}a^+_{S^{/i}ztK} + 1 && (17)\\
\#^{/i}(a^+_{S^{/i}ztK} \cup z_{\cdot,t+1}) &\leq \#^{/i}a^+_{S^{/i}ztK}. && (18)
\end{aligned}$$

To prove (I) we provide upper bounds for both $f_{S,K}(a^-_{SztK}, b^-_{SztK}) - f_{S^{/i},K}(a^-_{S^{/i}ztK}, b^-_{S^{/i}ztK})$ and $f_{S^{/i},K}(a^-_{S^{/i}ztK}, b^-_{S^{/i}ztK}) - f_{S,K}(a^-_{SztK}, b^-_{SztK})$. Using that for basket z the adjusted confidence of the highest-scoring-incorrect rule on $S^{/i}$, $a^-_{S^{/i}ztK} \to b^-_{S^{/i}ztK}$, exceeds that of another incorrect rule $a^-_{SztK} \to b^-_{SztK}$, and using inequalities (9) and (10),

$$\frac{\#^{/i}(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK})}{\#^{/i}a^-_{S^{/i}ztK} + K} \geq \frac{\#^{/i}(a^-_{SztK} \cup b^-_{SztK})}{\#^{/i}a^-_{SztK} + K} \geq \frac{\#(a^-_{SztK} \cup b^-_{SztK}) - 1}{\#a^-_{SztK} + K}.$$

Using the inequality above:

$$\begin{aligned} & f_{S,K}(a^-_{SztK}, b^-_{SztK}) - f_{S^{/i},K}(a^-_{S^{/i}ztK}, b^-_{S^{/i}ztK}) \\ = \;& \frac{\#(a^-_{SztK} \cup b^-_{SztK})}{\#a^-_{SztK} + K} - \frac{\#^{/i}(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK})}{\#^{/i}a^-_{S^{/i}ztK} + K} \\ \leq \;& \frac{\#(a^-_{SztK} \cup b^-_{SztK})}{\#a^-_{SztK} + K} - \frac{\#(a^-_{SztK} \cup b^-_{SztK}) - 1}{\#a^-_{SztK} + K} = \frac{1}{\#a^-_{SztK} + K}. \end{aligned} \tag{19}$$

Considering the other direction, using that the highest-scoring-incorrect rule under S has higher adjusted confidence than the rule $a^-_{S^{/i}ztK} \to b^-_{S^{/i}ztK}$ and inequalities (11) and (12):

$$\frac{\#(a^-_{SztK} \cup b^-_{SztK})}{\#a^-_{SztK} + K} \geq \frac{\#(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK})}{\#a^-_{S^{/i}ztK} + K} \geq \frac{\#^{/i}(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK})}{\#^{/i}a^-_{S^{/i}ztK} + 1 + K}.$$

Using this, and inequality (13),

$$\begin{aligned} & f_{S^{/i},K}(a^-_{S^{/i}ztK}, b^-_{S^{/i}ztK}) - f_{S,K}(a^-_{SztK}, b^-_{SztK}) = \frac{\#^{/i}(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK})}{\#^{/i}a^-_{S^{/i}ztK} + K} - \frac{\#(a^-_{SztK} \cup b^-_{SztK})}{\#a^-_{SztK} + K} \\ \leq \;& \frac{\#^{/i}(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK})}{\#^{/i}a^-_{S^{/i}ztK} + K} - \frac{\#^{/i}(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK})}{\#^{/i}a^-_{S^{/i}ztK} + 1 + K} = \frac{\#^{/i}(a^-_{S^{/i}ztK} \cup b^-_{S^{/i}ztK})}{(\#^{/i}a^-_{S^{/i}ztK} + K)(\#^{/i}a^-_{S^{/i}ztK} + 1 + K)} \\ \leq \;& \frac{\#^{/i}a^-_{S^{/i}ztK}}{(\#^{/i}a^-_{S^{/i}ztK} + K)(\#^{/i}a^-_{S^{/i}ztK} + 1 + K)} \leq \frac{1}{\#^{/i}a^-_{S^{/i}ztK} + K}. \end{aligned}$$

Together with (19) this proves (I). The proof of part (II) is identical, using a^+_{SztK} and $a^+_{S^{/i}ztK}$ in the place of a^-_{SztK} and $a^-_{S^{/i}ztK}$, $z_{\cdot,t+1}$ in the place of b^-_{SztK} and $b^-_{S^{/i}ztK}$, and inequalities (14)-(18). ■

The following lemma is the backbone for our stability computations. The upper bound in this lemma depends only on the supports of the relevant rules. Recall that $\tilde{a}_z := \min(\#a^-_{SztK}, \#^{/i}a^-_{S^{/i}ztK})$ and $\hat{a}_z := \min(\#a^+_{SztK}, \#^{/i}a^+_{S^{/i}ztK})$.

Lemma 13 *(Large Support Implies Stability)*

$$\begin{aligned} & |\ell_{\gamma,K_r}(f_{S,K}, z) - \ell_{\gamma,K_r}(f_{S^{/i},K}, z)| \\ \leq \;& \frac{1}{\gamma}\frac{1}{T_z}\sum_{t=0}^{T_z-1}\left[\frac{1}{\tilde{a}_z + K} + |K_r - K|\left[\frac{1}{\tilde{a}_z + K_r}\left(\frac{m}{m+K} + \frac{1}{\tilde{a}_z + K}\right)\right]\right. \\ & \left. + \frac{1}{\hat{a}_z + K} + |K_r - K|\left[\frac{1}{\hat{a}_z + K_r}\left(\frac{m}{m+K} + \frac{1}{\hat{a}_z + K}\right)\right]\right]. \end{aligned}$$

Proof

$$
\begin{aligned}
&|\ell_{\gamma,K_r}(f_{S,K},z)-\ell_{\gamma,K_r}(f_{S^{/i},K},z)| \\
&= \left|\frac{1}{T_z}\sum_{t=0}^{T_z-1} c_\gamma\left(f_{S,K_r}(a^+_{SztK},z_{\cdot,t+1})-f_{S,K_r}(a^-_{SztK},b^-_{SztK})\right)\right. \\
&\quad \left. -c_\gamma\left(f_{S^{/i},K_r}(a^+_{S^{/i}ztK},z_{\cdot,t+1})-f_{S^{/i},K_r}(a^-_{S^{/i}ztK},b^-_{S^{/i}ztK})\right)\right| \\
&\leq \frac{1}{\gamma}\frac{1}{T_z}\sum_{t=0}^{T_z-1}|f_{S,K_r}(a^+_{SztK},z_{\cdot,t+1})-f_{S,K_r}(a^-_{SztK},b^-_{SztK}) \\
&\quad -f_{S^{/i},K_r}(a^+_{S^{/i}ztK},z_{\cdot,t+1})+f_{S^{/i},K_r}(a^-_{S^{/i}ztK},b^-_{S^{/i}ztK})| \\
&\leq \frac{1}{\gamma}\frac{1}{T_z}\sum_{t=0}^{T_z-1}|f_{S,K_r}(a^-_{SztK},b^-_{SztK})-f_{S^{/i},K_r}(a^-_{S^{/i}ztK},b^-_{S^{/i}ztK})| \\
&\quad +|f_{S,K_r}(a^+_{SztK},z_{\cdot,t+1})-f_{S^{/i},K_r}(a^+_{S^{/i}ztK},z_{\cdot,t+1})| \\
&=: \frac{1}{\gamma}\frac{1}{T_z}\sum_{t=0}^{T_z-1}\mathrm{term}_1+\mathrm{term}_2.
\end{aligned}
$$

The first inequality above used that c_γ is $1/\gamma$-Lipschitz. Consider an upper bound for term_1 as follows from Lemma 11:

$$
\begin{aligned}
\mathrm{term}_1 &= |f_{S,K_r}(a^-_{SztK},b^-_{SztK})-f_{S^{/i},K_r}(a^-_{S^{/i}ztK},b^-_{S^{/i}ztK})| \\
&\leq |f_{S,K}(a^-_{SztK},b^-_{SztK})-f_{S^{/i},K}(a^-_{S^{/i}ztK},b^-_{S^{/i}ztK})|\left(1+\frac{|K_r-K|}{\#a^-_{SztK}+K_r}\right) \\
&\quad +\frac{|K_r-K|}{\min(\#a^-_{SztK},\#^{/i}a^-_{S^{/i}ztK})+K_r}\frac{\#^{/i}(a^-_{S^{/i}ztK}\cup b^-_{S^{/i}ztK})}{\#^{/i}a^-_{S^{/i}ztK}+K} \\
&\leq |f_{S,K}(a^-_{SztK},b^-_{SztK})-f_{S^{/i},K}(a^-_{S^{/i}ztK},b^-_{S^{/i}ztK})|\left(1+\frac{|K_r-K|}{\tilde{a}_z+K_r}\right) \\
&\quad +\frac{|K_r-K|}{\tilde{a}_z+K_r}\frac{\#^{/i}(a^-_{S^{/i}ztK}\cup b^-_{S^{/i}ztK})}{\#^{/i}a^-_{S^{/i}ztK}+K}.
\end{aligned}
$$

Now incorporating Lemma 12 and that $\frac{\#^{/i}(a^-_{S^{/i}ztK}\cup b^-_{S^{/i}ztK})}{\#^{/i}a^-_{S^{/i}ztK}+K}\leq\frac{m-1}{m-1+K}\leq\frac{m}{m+K}$,

$$
\begin{aligned}
\mathrm{term}_1 &\leq \frac{1}{\tilde{a}_z+K}\left(1+\frac{|K_r-K|}{\tilde{a}_z+K_r}\right)+\frac{|K_r-K|}{\tilde{a}_z+K_r}\frac{m}{m+K} \\
&= \frac{1}{\tilde{a}_z+K}+|K_r-K|\left[\frac{1}{\tilde{a}_z+K_r}\left(\frac{m}{m+K}+\frac{1}{\tilde{a}_z+K}\right)\right].
\end{aligned}
$$

The same steps can be followed exactly for term_2. ■

The following lemma is used for the proof for the large sample bound.

Lemma 14 *(Asymptotic Expectation of $1/(\#a + K)$) For any itemset $a \in A$ and any $K \geq 0$,*

$$\mathbb{E}_{S\sim\mathcal{D}} \frac{1}{\#a + K} \leq \frac{1}{mp_a + K} + \mathcal{O}\left(\frac{1}{m^2}\right),$$

where p_a is the probability that a random basket contains a, that is, $p_a = P_{z\sim\mathcal{D}}(a \subseteq z)$.

Since $\#a$ is binomially distributed, $\#a \sim \text{Binomial}(m, p_a)$, the proof of this lemma can be found by directly applying Lemma 17 in Appendix C.

We now give the proof of pointwise hypothesis stability for the large sample bound. We are interested in the change in adjusted confidence of specific basket z_i when that same basket is removed from the training set, that is on $S^{/i}$. Because Lemma 13 holds for any z, it also holds for z_i, where $\tilde{a}_{z_i} := \min(\#a^-_{Sz_itK}, \#^{/i}a^-_{S^{/i}z_itK})$ and $\hat{a}_{z_i} := \min(\#a^+_{Sz_itK}, \#^{/i}a^+_{S^{/i}z_itK})$.

Proof *(Of Theorem 3)* First, note that:

$$\begin{aligned}\frac{1}{\tilde{a}_{z_i} + K_r} &= \frac{1}{\min(\#a^-_{Sz_itK}, \#^{/i}a^-_{S^{/i}z_itK}) + K_r} \\ &\leq \frac{1}{\min(\#^{/i}a^-_{Sz_itK}, \#^{/i}a^-_{S^{/i}z_itK}) + K_r} \leq \sum_{a\in\mathcal{A}} \frac{1}{\#^{/i}a + K_r}.\end{aligned}$$

By the same reasoning, similar upper bounds hold for $1/(\tilde{a}_{z_i} + K)$, $1/(\hat{a}_{z_i} + K_r)$, and $1/(\hat{a}_{z_i} + K)$. Starting from Lemma 13 using specific basket z_i and incorporating these bounds on each fraction,

$$\begin{aligned}&|\ell_{\gamma,K_r}(f_{S,K}, z_i) - \ell_{\gamma,K_r}(f_{S^{/i},K}, z_i)| \\ &\leq \frac{2}{\gamma}\frac{1}{T_{z_i}} \sum_{t=0}^{T_{z_i}-1} \left[\sum_{a\in\mathcal{A}} \frac{1}{\#^{/i}a + K} + |K_r - K| \left[\left(\sum_{a\in\mathcal{A}} \frac{1}{\#^{/i}a + K_r}\right)\left(\frac{m}{m+K} + \sum_{a\in\mathcal{A}} \frac{1}{\#^{/i}a + K}\right)\right]\right].\end{aligned} \tag{20}$$

We have also that for any K_r, using that $p_{\min A} \leq p_a$ for all $a \in \mathcal{A}$, and Lemma 14:

$$\mathbb{E}_{S^{/i}\sim\mathcal{D}^{m-1}} \sum_{a\in\mathcal{A}} \frac{1}{\#^{/i}a + K_r} \leq \frac{|\mathcal{A}|}{(m-1)p_{\min A} + K_r} + \mathcal{O}\left(\frac{1}{m^2}\right). \tag{21}$$

Thus from (20) and (21), for any $1 \leq i \leq m$,

$$\begin{aligned}
&\mathbb{E}_{S\sim\mathcal{D}^m}|\ell_{\gamma,K_r}(f_{S,K},z_i)-\ell_{\gamma,K_r}(f_{S^{/i},K},z_i)| \\
&\leq \frac{2}{\gamma}\mathbb{E}_{z_i\sim\mathcal{D}}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\mathbb{E}_{S^{/i}\sim\mathcal{D}^{m-1}}\left[\sum_{a\in\mathcal{A}}\frac{1}{\#^{/i}a+K}\right. \\
&+ \left.|K_r-K|\left[\left(\sum_{a\in\mathcal{A}}\frac{1}{\#^{/i}a+K_r}\right)\left(\frac{m}{m+K}+\sum_{a\in\mathcal{A}}\frac{1}{\#^{/i}a+K}\right)\right]\right] \\
&\leq \frac{2}{\gamma}\mathbb{E}_{z_i\sim\mathcal{D}}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\frac{|\mathcal{A}|}{(m-1)p_{\min A}+K}+\mathcal{O}\left(\frac{1}{m^2}\right) \\
&\quad +|K_r-K|\left[\left(\frac{|\mathcal{A}|}{(m-1)p_{\min A}+K_r}\right)\left(\frac{m}{m+K}\right)+\mathcal{O}\left(\frac{1}{m^2}\right)\right] \\
&= \frac{2}{\gamma}\frac{|\mathcal{A}|}{(m-1)p_{\min A}+K}+|K_r-K|\frac{2}{\gamma}\left(\frac{|\mathcal{A}|}{(m-1)p_{\min A}+K_r}\right)\left(\frac{m}{m+K}\right)+\mathcal{O}\left(\frac{1}{m^2}\right)=:\beta,
\end{aligned}$$

where in the second inequality, we moved the $(\sum_{a\in\mathcal{A}}1/(\#^{/i}a+K_r))(\sum_{a\in\mathcal{A}}1/(\#^{/i}a+K))$ terms into the $\mathcal{O}\left(\frac{1}{m^2}\right)$. To see this, one can take a Taylor expansion around the mean for all of the terms similar to $\frac{1}{\#a+K}$ as follows:

$$\frac{1}{\#a+K}\approx\frac{1}{mp_a+K}-\frac{(\#a-mp_a)}{(mp_a+K)^2}+\frac{(\#a-mp_a)^2}{(mp_a+K)^3}+\ldots.$$

When these terms are multiplied together, the result is always $\mathcal{O}\left(\frac{1}{m^2}\right)$. Thus, the algorithm has pointwise hypothesis stability β. Using β within the B&E theorem yields the result. ■

Proof *(Of Theorem 5)*
Starting from Lemma 13, we will use the minimum support threshold to provide the upper bound for the reciprocal of the support of rules. All of the steps used to derive Lemma 13 are valid for the max confidence, min support setting, only the notation needs to be changed. We define $\tilde{a}_{z,\theta}:=\min(\#a^{-}_{Szt\theta},\#^{/i}a^{-}_{S^{/i}zt\theta})$, and now define also $\hat{a}_{z,\theta}:=\min(\#a^{+}_{Szt\theta},\#^{/i}a^{+}_{S^{/i}zt\theta})$. Lemma 13 provides for $\bar{f}_{S,\theta}$ and using $K=0$:

$$\begin{aligned}
&|\ell_{\gamma,K_r}(\bar{f}_{S,\theta},z)-\ell_{\gamma,K_r}(\bar{f}_{S^{/i},\theta},z)| \\
&\leq \frac{1}{\gamma}\frac{1}{T_z}\sum_{t=0}^{T_z-1}\left[\frac{1}{\tilde{a}_{z,\theta}}+K_r\left[\frac{1}{\tilde{a}_{z,\theta}+K_r}\left(1+\frac{1}{\tilde{a}_{z,\theta}}\right)\right]+\frac{1}{\hat{a}_{z,\theta}}+K_r\left[\frac{1}{\hat{a}_{z,\theta}+K_r}\left(1+\frac{1}{\hat{a}_{z,\theta}}\right)\right]\right].
\end{aligned}$$

The requirement of a minimum support threshold ensures that for any particular item b, the highest scoring rule with b on the right must have support at least θ, that is: $\underset{a\subseteq\{z_{\cdot,1},\ldots,z_{\cdot,t}\},a\in A}{\operatorname{argmax}}\bar{f}_{S,\theta}(a,b)$ includes only itemsets with support at least θ. If b has never been ordered, $\max_a\bar{f}_{S,\theta}(a,b)=0$ and we choose the maximizing rule to be $\varnothing\rightarrow b$, with support $m>m-1\geq\theta$. By this reasoning, all of the rules we use have support at least

θ: $\#a^-_{Szt\theta} \geq \theta$, $\#^{/i}a^-_{S^{/i}zt\theta} \geq \theta$, $\#a^+_{Szt\theta} \geq \theta$, and $\#^{/i}a^+_{S^{/i}zt\theta} \geq \theta$. Thus, $\tilde{a}_{z,\theta} \geq \theta$ and also $\hat{a}_{z,\theta} \geq \theta$. Using this in the previous expression:

$$
\begin{aligned}
&|\ell_{\gamma,K_r}(\bar{f}_{S,\theta}, z) - \ell_{\gamma,K_r}(\bar{f}_{S^{/i},\theta}, z)| \\
&\quad \leq \quad \frac{2}{\gamma}\frac{1}{T_z}\sum_{t=0}^{T_z-1}\left[\frac{1}{\theta} + K_r\left[\left(\frac{1}{\theta+K_r}\right)\left(1+\frac{1}{\theta}\right)\right]\right] = \frac{2}{\gamma}\left[\frac{1}{\theta} + K_r\left(\frac{1}{\theta+K_r}\right)\left(1+\frac{1}{\theta}\right)\right] =: \beta.
\end{aligned}
$$

This expression holds for all S and for all z. It is thus an upper bound on the uniform stability. Using β within the B&E theorem yields the result. ■

The proofs of Theorems 3 and 5 for classification are essentially identical to those provided above for sequential event prediction. The left-hand side of a highest-scoring-correct rule for general basket x on $S^{/i}$ obeys:

$$
a^+_{S^{/i}xK} \in \operatorname*{argmax}_{a\subseteq x, a\in A} f_{S^{/i},K}(a, y) = \operatorname*{argmax}_{a\subseteq x, a\in A} \frac{\#^{/i}(a \cup y)}{\#^{/i}a + K}.
$$

And the left-hand side of a highest-scoring-incorrect rule for x on $S^{/i}$ obeys:

$$
a^-_{S^{/i}xK} \in \operatorname*{argmax}_{a\subseteq x, a\in A} f_{S^{/i},K}(a, -y) = \operatorname*{argmax}_{a\subseteq x, a\in A} \frac{\#^{/i}(a \cup -y)}{\#^{/i}a + K}.
$$

We further define $\tilde{a}_x = \min(\#a^-_{SxK}, \#^{/i}a^-_{S^{/i}xK})$ and $\hat{a}_x := \min(\#a^+_{SxK}, \#^{/i}a^+_{S^{/i}xK})$, and $\tilde{a}_{x_i}$ and $\hat{a}_{x_i}$ as the analogous quantities for specific basket x_i. Lemma 12, Lemma 13, and the proof of Theorem 3 all hold for classification by making the following substitutions in notation: $\tilde{a}_x$ and $\tilde{a}_{x_i}$ for $\tilde{a}_z$ and $\tilde{a}_{z_i}$; $\hat{a}_x$ and $\hat{a}_{x_i}$ for $\hat{a}_z$ and $\hat{a}_{z_i}$; a^-_{SxK} and $-y$ for a^-_{SztK} and b^-_{SztK}; a^+_{SxK} and y for a^+_{SztK} and $z_{\cdot,t+1}$; $a^-_{S^{/i}xK}$ and $-y$ for $a^-_{S^{/i}ztK}$ and $b^-_{S^{/i}ztK}$; $a^+_{S^{/i}xK}$ for $a^+_{S^{/i}ztK}$; $\ell^{\text{class}}_{\gamma,K_r}$ for ℓ_{γ,K_r}; and removing entirely $\frac{1}{T_z}\sum_{t=0}^{T_z-1}$. For Theorem 5, we again replace K with θ in the notation to define $\tilde{a}_{x,\theta} = \min(\#a^-_{Sx\theta}, \#^{/i}a^-_{S^{/i}x\theta})$ and $\hat{a}_{x,\theta} := \min(\#a^+_{Sx\theta}, \#^{/i}a^+_{S^{/i}x\theta})$, and then substitute $\tilde{a}_{x,\theta}$ and $\hat{a}_{x,\theta}$ for $\tilde{a}_{z,\theta}$ and $\hat{a}_{z,\theta}$ in the proof of the theorem.

The next lemma is specific to classification and is used for the small sample bound for the adjusted confidence algorithm.

Lemma 15 *(Support Thresholds for Adjusted Confidence, Classification)*
For specific basket x_i, it is true that:

$$
\begin{aligned}
\frac{1}{\tilde{a}_{x_i} + K_r} &\leq \tilde{\alpha}_{K_r}, \text{ where } \tilde{\alpha}_{K_r} = \frac{m + K - \#^{/i}(-y_i)}{K(\#^{/i}(-y_i)) + K_r(m + K - \#^{/i}(-y_i))}; \\
\frac{1}{\tilde{a}_{x_i} + K} &\leq \tilde{\alpha}_K, \text{ where } \tilde{\alpha}_K = \frac{1}{K}\left(1 - \frac{\#^{/i}(-y_i)}{m+K}\right); \\
\frac{1}{\hat{a}_{x_i} + K_r} &\leq \hat{\alpha}_{K_r}, \text{ where } \hat{\alpha}_{K_r} = \frac{m + K - \#^{/i}y_i}{K(\#^{/i}y_i) + K_r(m + K - \#^{/i}y_i)}; \text{ and,} \\
\frac{1}{\hat{a}_{x_i} + K} &\leq \hat{\alpha}_K, \text{ where } \hat{\alpha}_K = \frac{1}{K}\left(1 - \frac{\#^{/i}y_i}{m+K}\right).
\end{aligned}
$$

Proof First we use the fact that on S, the adjusted confidence of the highest-scoring-incorrect rule for x_i, $a^-_{Sx_iK} \to -y_i$, exceeds that of the rule $\varnothing \to -y_i$:

$$\frac{\#a^-_{Sx_iK}}{\#a^-_{Sx_iK}+K} \;\geq\; \frac{\#(a^-_{Sx_iK}\cup -y_i)}{\#a^-_{Sx_iK}+K} \geq \frac{\#(-y_i)}{m+K} = \frac{\#^{/i}(-y_i)}{m+K},$$

where in the last step we used that basket x_i does not have label $-y_i$. Rearranging,

$$\#a^-_{Sx_iK} \geq \tilde{\sigma} \text{ where } \tilde{\sigma} := K\left(\frac{\#^{/i}(-y_i)}{m+K-\#^{/i}(-y_i)}\right).$$

Similarly, the adjusted confidence of the highest-scoring-incorrect rule for x_i with dataset $S^{/i}$, $a^-_{S^{/i}x_iK} \to -y_i$, exceeds that of the rule $\varnothing \to -y_i$, thus:

$$\frac{\#^{/i}a^-_{S^{/i}x_iK}}{\#^{/i}a^-_{S^{/i}x_iK}+K} \;\geq\; \frac{\#^{/i}(a^-_{S^{/i}x_iK}\cup -y_i)}{\#^{/i}a^-_{S^{/i}x_iK}+K} \geq \frac{\#^{/i}(-y_i)}{m-1+K} \geq \frac{\#^{/i}(-y_i)}{m+K}.$$

Rearranging, we find that $\#^{/i}a^-_{S^{/i}x_iK} \geq \tilde{\sigma}$. Thus, $\tilde{a}_{x_i} = \min(\#a^-_{Sx_iK}, \#^{/i}a^-_{S^{/i}x_iK}) \geq \tilde{\sigma}$. We can derive a similar bound for $\hat{a}_{x_i}$, beginning with $\#a^+_{Sx_iK}$:

$$\frac{\#a^+_{Sx_iK}}{\#a^+_{Sx_iK}+K} \;\geq\; \frac{\#(a^+_{Sx_iK}\cup y_i)}{\#a^+_{Sx_iK}+K} \geq \frac{\#y_i}{m+K} = \frac{\#^{/i}y_i+1}{m+K} > \frac{\#^{/i}y_i}{m+K}.$$

The first equality uses that basket x_i has label y_i. Rearranging,

$$\#a^+_{Sx_iK} > \hat{\sigma} \text{ where } \hat{\sigma} := K\left(\frac{\#^{/i}y_i}{m+K-\#^{/i}y_i}\right).$$

Similarly for $\#^{/i}a^+_{S^{/i}x_iK}$:

$$\frac{\#^{/i}a^+_{S^{/i}x_iK}}{\#^{/i}a^+_{S^{/i}x_iK}+K} \;\geq\; \frac{\#^{/i}(a^+_{S^{/i}x_iK}\cup y_i)}{\#^{/i}a^+_{S^{/i}x_iK}+K} \geq \frac{\#^{/i}y_i}{m-1+K} \geq \frac{\#^{/i}y_i}{m+K}.$$

Rearranging, we find $\#^{/i}a^-_{S^{/i}x_iK} \geq \hat{\sigma}$. Thus $\hat{a}_{x_i} = \min(\#a^+_{Sx_iK}, \#^{/i}a^+_{S^{/i}x_iK}) \geq \hat{\sigma}$. These lower bounds on the supports are now used to create upper bounds for the reciprocals:

$$\frac{1}{\tilde{a}_{x_i}+K_r} \;\leq\; \frac{1}{\tilde{\sigma}+K_r} = \tilde{\alpha}_{K_r} \quad \text{and} \quad \frac{1}{\tilde{a}_{x_i}+K} \leq \frac{1}{\tilde{\sigma}+K} = \tilde{\alpha}_K.$$

The bounds for $\frac{1}{\hat{a}_{x_i}+K_r}$ and $\frac{1}{\hat{a}_{x_i}+K}$ are obtained in a similar way using $\hat{\sigma}$. ■

The proof of the small sample bound for classification follows directly from this lemma.

Proof *(Of Theorem 7)*
From Lemma 13, adapted for classification,

$$\begin{aligned}
&|\ell^{\text{class}}_{\gamma,K_r}(f_{S,K},z_i) - \ell^{\text{class}}_{\gamma,K_r}(f_{S^{/i},K},z_i)| \\
&\quad\leq \frac{1}{\gamma}\left[\frac{1}{\tilde{a}_{x_i}+K} + |K_r-K|\left[\frac{1}{\tilde{a}_{x_i}+K_r}\left(\frac{m}{m+K}+\frac{1}{\tilde{a}_{x_i}+K}\right)\right]\right. \\
&\qquad\left. + \frac{1}{\hat{a}_{x_i}+K} + |K_r-K|\left[\frac{1}{\hat{a}_{x_i}+K_r}\left(\frac{m}{m+K}+\frac{1}{\hat{a}_{x_i}+K}\right)\right]\right].
\end{aligned}$$

Combining this and Lemma 15, we have:

$$\begin{aligned}
&|\ell^{\text{class}}_{\gamma,K_r}(f_{S,K},z_i) - \ell^{\text{class}}_{\gamma,K_r}(f_{S^{/i},K},z_i)| \\
&\quad\leq \frac{1}{\gamma}\left[\tilde{\alpha}_K + |K_r-K|\tilde{\alpha}_{K_r}\left(\frac{m}{m+K}+\tilde{\alpha}_K\right) + \hat{\alpha}_K + |K_r-K|\hat{\alpha}_{K_r}\left(\frac{m}{m+K}+\hat{\alpha}_K\right)\right] \\
&\quad= \frac{1}{\gamma}\left(\tilde{\alpha}_K+\hat{\alpha}_K\right) + \frac{1}{\gamma}|K_r-K|\left[\tilde{\alpha}_{K_r}\left(\frac{m}{m+K}+\tilde{\alpha}_K\right) + \hat{\alpha}_{K_r}\left(\frac{m}{m+K}+\hat{\alpha}_K\right)\right].
\end{aligned}$$

We now provide an upper bound on the expectation of this quantity, beginning with the first term:

$$\begin{aligned}
\mathbb{E}_{z_1,\ldots,z_m}\frac{1}{\gamma}\left(\tilde{\alpha}_K+\hat{\alpha}_K\right) &= \mathbb{E}_{z_1,\ldots,z_m}\frac{1}{\gamma}\frac{1}{K}\left[\left(1-\frac{\#^{/i}(-y_i)}{m+K}\right)+\left(1-\frac{\#^{/i}y_i}{m+K}\right)\right] \\
&= \frac{1}{\gamma}\frac{1}{K}\left(2-\frac{(m-1)p_{-y_i}}{m+K}-\frac{(m-1)p_{y_i}}{m+K}\right) \\
&\leq \frac{2}{\gamma}\frac{1}{K}\left(1-\frac{(m-1)p_{y,\min}}{m+K}\right).
\end{aligned}$$

Here we used the fact that the mean of the binomial distribution $\text{Bin}(m-1,p_{y_i})$ is $(m-1)p_{y_i}$, and we use a lower bound for p_{y_i} and p_{-y_i}, namely $p_{y,\min} = \min(P(y=1), P(y=-1))$ the minimum probability of a randomly chosen basket having any particular label. For

the second term,

$$\begin{aligned}
&\mathbb{E}_{z_1,\ldots,z_m}\frac{1}{\gamma}|K_r-K|\left[\tilde{\alpha}_{K_r}\left(\frac{m}{m+K}+\tilde{\alpha}_K\right)+\hat{\alpha}_{K_r}\left(\frac{m}{m+K}+\hat{\alpha}_K\right)\right]\\
&= \frac{1}{\gamma}|K_r-K|\mathbb{E}_{z_1,\ldots,z_m}\left[\frac{1}{K\left(\frac{\#^{/i}(-y_i)}{m+K-\#^{/i}(-y_i)}\right)+K_r}\left(\frac{m}{m+K}+\frac{1}{K}\left(1-\frac{\#^{/i}(-y_i)}{m+K}\right)\right)\right]\\
&\quad +\frac{1}{\gamma}|K_r-K|\mathbb{E}_{z_1,\ldots,z_m}\left[\frac{1}{K\left(\frac{\#^{/i}y_i}{m+K-\#^{/i}y_i}\right)+K_r}\left(\frac{m}{m+K}+\frac{1}{K}\left(1-\frac{\#^{/i}y_i}{m+K}\right)\right)\right]\\
&= \frac{1}{\gamma}|K_r-K|\mathbb{E}_{\tilde{\zeta}\sim\mathrm{Bin}(m-1,p_{-y_i})}\left[\frac{1}{K\left(\frac{\tilde{\zeta}}{m+K-\tilde{\zeta}}\right)+K_r}\left(\frac{m}{m+K}+\frac{1}{K}\left(1-\frac{\tilde{\zeta}}{m+K}\right)\right)\right]\\
&\quad +\frac{1}{\gamma}|K_r-K|\mathbb{E}_{\hat{\zeta}\sim\mathrm{Bin}(m-1,p_{y_i})}\left[\frac{1}{K\left(\frac{\hat{\zeta}}{m+K-\hat{\zeta}}\right)+K_r}\left(\frac{m}{m+K}+\frac{1}{K}\left(1-\frac{\hat{\zeta}}{m+K}\right)\right)\right]\\
&=: \frac{1}{\gamma}|K_r-K|\mathbb{E}_{\tilde{\zeta}\sim\mathrm{Bin}(m-1,p_{-y_i})}F(\tilde{\zeta})+\frac{1}{\gamma}|K_r-K|\mathbb{E}_{\hat{\zeta}\sim\mathrm{Bin}(m-1,p_{y_i})}F(\hat{\zeta}).
\end{aligned}$$

Since the function $F(\zeta)$ is decreasing as ζ increases, then an upper bound is produced by using the distribution $\mathrm{Bin}(m-1,p_{y,\min})$:

$$\begin{aligned}
&\mathbb{E}_{z_1,\ldots,z_m}\frac{1}{\gamma}|K_r-K|\left[\tilde{\alpha}_{K_r}\left(\frac{m}{m+K}+\tilde{\alpha}_K\right)+\hat{\alpha}_{K_r}\left(\frac{m}{m+K}+\hat{\alpha}_K\right)\right]\\
&\leq \frac{2}{\gamma}|K_r-K|\mathbb{E}_{\zeta\sim\mathrm{Bin}(m-1,p_{y,\min})}F(\zeta)\\
&= \frac{2}{\gamma}|K_r-K|\mathbb{E}_{\zeta\sim\mathrm{Bin}(m-1,p_{y,\min})}\left[\frac{1}{K\left(\frac{\zeta}{m+K-\zeta}\right)+K_r}\left(\frac{m}{m+K}+\frac{1}{K}\left(1-\frac{\zeta}{m+K}\right)\right)\right].
\end{aligned}$$

■

The following lemma is similar to the previous lemma, but specific to sequential event prediction. It uses the support guarantee for the adjusted confidence algorithm (5) in order to bound the terms of Lemma 13, which holds with the same proof when the loss ℓ_{γ,K_r} is changed to the new loss $\ell^{\mathrm{new}}_{\gamma,K_r}$ and superscript "-" is replaced by "*". We define the analogy to $\tilde{a}_{z_i}$ as $\tilde{a}^*_{z_i} := \min(\#a^*_{Sz_itK}, \#^{/i}a^*_{S^{/i}z_itK})$. The result below will immediately yield a proof of Theorem 8.

Lemma 16 *(Support Thresholds for Adjusted Confidence, Sequential Event Prediction) For specific basket z_i, define:*

$$\alpha_{K_r} := \frac{m+K-\#z_{i,t+1}}{K(\#z_{i,t+1}-1)+K_r(m+K-\#z_{i,t+1})} \quad \textit{and} \quad \alpha_K := \frac{1}{K}\left(1-\frac{\#z_{i,t+1}-1}{m+K}\right).$$

It is true that:

$$\frac{1}{\tilde{a}^*_{z_i}+K_r} \le \alpha_{K_r}, \quad \frac{1}{\tilde{a}^*_{z_i}+K} \le \alpha_K, \quad \frac{1}{\hat{a}_{z_i}+K_r} \le \alpha_{K_r}, \text{ and } \frac{1}{\hat{a}_{z_i}+K} \le \alpha_K.$$

Proof Starting with (4), we know that $a^*_{Sz_itK} > \sigma$, where

$$\sigma := K\left(\frac{\#z_{i,t+1}-1}{m+K-\#z_{i,t+1}}\right).$$

We use the same type of argument as in (4), incorporating the fact that on $S^{/i}$, the adjusted confidence of the highest scoring rule $a^*_{S^{/i}z_itK} \to b^*_{S^{/i}z_itK}$ exceeds that of the highest-scoring-correct rule $a^+_{S^{/i}z_itK} \to z_{i,t+1}$, which exceeds that of the rule $\varnothing \to z_{i,t+1}$,

$$\begin{aligned}\frac{\#^{/i}a^*_{S^{/i}z_itK}}{\#^{/i}a^*_{S^{/i}z_itK}+K} &\ge \frac{\#^{/i}(a^*_{S^{/i}z_itK}\cup b^*_{S^{/i}z_itK})}{\#^{/i}a^*_{S^{/i}z_itK}+K} \ge \frac{\#^{/i}(a^+_{S^{/i}z_itK}\cup z_{i,t+1})}{\#^{/i}a^+_{S^{/i}z_itK}+K} \\ &\ge \frac{\#^{/i}z_{i,t+1}}{m-1+K} = \frac{\#z_{i,t+1}-1}{m-1+K}. \end{aligned} \tag{22}$$

Rearranging, we find that $\#^{/i}a^*_{S^{/i}z_itK} > \sigma$. Similarly for $\#a^+_{Sz_itK}$,

$$\frac{\#a^+_{Sz_itK}}{\#a^+_{Sz_itK}+K} \ge \frac{(\#a^+_{Sz_itK}\cup z_{i,t+1})}{\#a^+_{Sz_itK}+K} \ge \frac{\#z_{i,t+1}}{m+K}$$

so $\#a^+_{Sz_itK} \ge K\left(\frac{\#z_{i,t+1}}{m+K-\#z_{i,t+1}}\right) > \sigma$. And again for $\#^{/i}a^+_{S^{/i}z_itK}$ using (22),

$$\frac{\#^{/i}a^+_{S^{/i}z_itK}}{\#^{/i}a^+_{S^{/i}z_itK}+K} \ge \frac{\#^{/i}(a^+_{S^{/i}z_itK}\cup z_{i,t+1})}{\#^{/i}a^+_{S^{/i}z_itK}+K} \ge \frac{\#z_{i,t+1}-1}{m-1+K}.$$

so $\#^{/i}a^+_{S^{/i}z_itK} \ge \sigma$. We now have $\tilde{a}^*_{z_i} = \min(\#a^*_{Sz_itK}, \#^{/i}a^*_{S^{/i}z_itK}) \ge \sigma$, and also $\hat{a}_{z_i} = \min(\#a^+_{Sz_itK}, \#^{/i}a^+_{S^{/i}z_itK}) \ge \sigma$. Since σ is a lower bound on all the supports, it can be used to create an upper bound for the reciprocals, as follows, using $\tilde{a}^*_{z_i}$ as an example:

$$\frac{1}{\tilde{a}^*_{z_i}+K_r} \le \frac{1}{\sigma+K_r} = \alpha_{K_r} \quad \text{and} \quad \frac{1}{\tilde{a}^*_{z_i}+K} \le \frac{1}{\sigma+K} = \alpha_K.$$

■

Proof *(Of Theorem 8)* First, all of the steps in the proof of Lemma 13 hold when we replace the loss ℓ_{γ,K_r} with the new loss $\ell^{\text{new}}_{\gamma,K_r}$, replace c_γ with c^{new}_γ, and $\tilde{a}_{z_i}$ by $\tilde{a}^*_{z_i}$, so we obtain:

$$\begin{aligned}&|\ell^{\text{new}}_{\gamma,K_r}(f_{S,K},z_i) - \ell^{\text{new}}_{\gamma,K_r}(f_{S^{/i},K},z_i)| \\ &\le \frac{1}{\gamma}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\left[\frac{1}{\tilde{a}^*_{z_i}+K} + |K_r-K|\left[\frac{1}{\tilde{a}^*_{z_i}+K_r}\left(\frac{m}{m+K}+\frac{1}{\tilde{a}^*_{z_i}+K}\right)\right]\right. \\ &\quad \left. + \frac{1}{\hat{a}_{z_i}+K} + |K_r-K|\left[\frac{1}{\hat{a}_{z_i}+K_r}\left(\frac{m}{m+K}+\frac{1}{\hat{a}_{z_i}+K}\right)\right]\right].\end{aligned}$$

Combining this and Lemma 16, we have:

$$|\ell^{\text{new}}_{\gamma,K_r}(f_{S,K},z_i)-\ell^{\text{new}}_{\gamma,K_r}(f_{S^{/i},K},z_i)|\leq\frac{2}{\gamma}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\alpha_K+|K_r-K|\alpha_{K_r}\left(\frac{m}{m+K}+\alpha_K\right).$$

To calculate the stability, we need an upper bound on the expectation of this quantity. Let us first create an upper bound for the expectation of the first term, $\frac{2}{\gamma}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\alpha_K$:

$$\begin{aligned}
\mathbb{E}_{z_1,\ldots,z_m}\frac{2}{\gamma}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\alpha_K &= \mathbb{E}_{z_1,\ldots,z_m}\frac{2}{\gamma}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\frac{1}{K}\left(1-\frac{\#z_{i,t+1}-1}{m+K}\right)\\
&= \mathbb{E}_{z_i}\frac{2}{\gamma}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\frac{1}{K}\left(1-\frac{\mathbb{E}_{z_1,\ldots,z_{i-1},z_{i+1},\ldots,z_m}\#z_{i,t+1}-1}{m+K}\right)\\
&= \mathbb{E}_{z_i}\frac{2}{\gamma}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\frac{1}{K}\left(1-\frac{(m-1)p_{z_{i,t+1}}}{m+K}\right)\\
&\leq \mathbb{E}_{z_i}\frac{2}{\gamma}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\frac{1}{K}\left(1-\frac{(m-1)p_{\min}}{m+K}\right)=\frac{2}{\gamma}\frac{1}{K}\left(1-\frac{(m-1)p_{\min}}{m+K}\right).
\end{aligned}$$

The first line above uses the definition of α_K, the second line uses the fact that each basket is chosen independently, the third line uses that $z_{i,t+1}$ is always contained in z_i and also uses the fact that the mean of the binomial distribution $\text{Bin}(m-1,p_{z_{i,t+1}})$ is $(m-1)p_{z_{i,t+1}}$. The fourth line uses that $p_{z_{i,t+1}}$ has the lower bound $p_{\min}$, which no longer depends on z_i.

We repeat this outline for the second term:

$$\begin{aligned}
&\mathbb{E}_{z_1,\ldots,z_m}\frac{2}{\gamma}|K_r-K|\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\alpha_{K_r}\left(\frac{m}{m+K}+\alpha_K\right)\\
&= \mathbb{E}_{z_1,\ldots,z_m}\frac{2}{\gamma}|K_r-K|\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\frac{1}{K\left(\frac{\#z_{i,t+1}-1}{m+K-\#z_{i,t+1}}\right)+K_r}\left(\frac{m}{m+K}+\frac{1}{K}\left(1-\frac{\#z_{i,t+1}-1}{m+K}\right)\right)\\
&= \frac{2}{\gamma}|K_r-K|\mathbb{E}_{z_i}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\mathbb{E}_{z_1,\ldots,z_{i-1},z_{i+1},\ldots,z_m}\left[\frac{1}{K\left(\frac{\#z_{i,t+1}-1}{m+K-\#z_{i,t+1}}\right)+K_r}\times\right.\\
&\quad\left.\left(\frac{m}{m+K}+\frac{1}{K}\left(1-\frac{\#z_{i,t+1}-1}{m+K}\right)\right)\right]\\
&= \frac{2}{\gamma}|K_r-K|\mathbb{E}_{z_i}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\mathbb{E}_{\zeta\sim\text{Bin}(m-1,p_{z_{i,t+1}})}\frac{1}{K\left(\frac{\zeta+1-1}{m+K-\zeta-1}\right)+K_r}\left(\frac{m}{m+K}+\frac{1}{K}\left(1-\frac{\zeta+1-1}{m+K}\right)\right)\\
&=: \frac{2}{\gamma}|K_r-K|\mathbb{E}_{z_i}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\mathbb{E}_{\zeta\sim\text{Bin}(m-1,p_{z_{i,t+1}})}F(\zeta).
\end{aligned}$$

Since the function F is decreasing as ζ increases, then an upper bound is produced by using the distribution $\text{Bin}(m-1, p_{min})$. Namely,

$$\begin{aligned}
&\mathbb{E}_{z_1,\ldots,z_m}\frac{2}{\gamma}|K_r - K|\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\alpha_{K_r}\left(\frac{m}{m+K}+\alpha_K\right)\\
&\leq \frac{2}{\gamma}|K_r - K|\mathbb{E}_{z_i}\frac{1}{T_{z_i}}\sum_{t=0}^{T_{z_i}-1}\mathbb{E}_{\zeta\sim\text{Bin}(m-1,p_{\min})}F(\zeta)\\
&= \frac{2}{\gamma}|K_r - K|\mathbb{E}_{\zeta\sim\text{Bin}(m-1,p_{\min})}\frac{1}{K\left(\frac{\zeta}{m+K-\zeta-1}\right)+K_r}\left(\frac{m}{m+K}+\frac{1}{K}\left(1-\frac{\zeta}{m+K}\right)\right).
\end{aligned}$$

■

In all of the theorems and proofs, the empirical loss and true loss are defined only for the case where the algorithm only recommends one item ($c = 1$). It is possible to use a vector norm to generalize to larger c.

5. Experiments

All datasets chosen for these experiments are publicly available from the UCI machine learning repository (Frank and Asuncion, 2010), and from the IBM Quest Market-Basket Synthetic Data Generator (Agrawal and Srikant, 1994). To obtain formatted market-basket data, categorical data were converted into binary features (one feature per category). Each feature represents an item, and each example represents a basket. The feature value (0 or 1) indicates the presence of an item. Training baskets and test baskets were chosen randomly without replacement from the full dataset. Since these data do not come naturally with a time ordering, items in the basket were randomly permuted to attain an order. At each iteration, rules were formed from one item or the empty item on the left, and one item on the right (See *GenRules* in Figure 4). Recommendations of one item were made using the following 15 algorithms: highest support, highest confidence, highest adjusted confidence for eight K levels, max confidence, min support algorithm for five support threshold levels θ. All 15 algorithms were evaluated by the average fraction of correct recommendations (AvgCorrect) per basket. As recommendations were made, it was common to have ties where multiple items were equally good to recommend, in which case the tie was broken at random; AvgCorrect is similar to $\ell_{0-1,K}$ except for this way of dealing with ties.

The parameters of the experiment are: number of training baskets (20 in all cases), number of test baskets (100 in all cases), values of K for the adjusted confidence algorithm (0.0001, 0.001, 0.01, 0.1, 1, 5, 10, 15), and values of θ for the max confidence, min support algorithm (1, 2, 3, 5, 10). Note that two of these algorithms are the same: the max confidence algorithm is the same as the max confidence, min support algorithm for θ=1. Datasets are: Car Evaluation (25 items, 1728 baskets), Chess King-Rook vs. King-Pawn, (75 items, 3196 baskets), MONK's problems (19 items, 1711 baskets) Mushroom (119 items, 8124 baskets), Nursery (32 items, 12960 baskets), Plants (70 items, 34781 baskets), T20I18D10KN22CR50 (22 items, 10000 baskets).

Algorithm 4: *Subroutine GenRules*, simplest version that considers only "marginal" rules.

Input: $(S, B, \mathcal{X})$, that is, past orders $S = \{z_i\}_{i=1,\ldots,m}$, $z_i \subseteq \mathcal{X}$, current basket $B \subset \mathcal{X}$, set of items $\mathcal{X}$

Output: Set of all rules where a_j is an item in the basket B (or the empty set) and b_j is not in B. That is, rules $\{a_j \to b_j\}_j$ such that $b_j \in \mathcal{X} \backslash B$ and either $a_j \in B$ or $a_j = \varnothing$.

Each experiment (training, test, evaluation for all 15 algorithms) was performed 100 times, (totaling 100×100×15 = 150,000 test basket evaluations per dataset, for each of 7 datasets). In Figures 4 through 7, the distribution of AvgCorrect values for datasets Chess and Monk are shown via boxplot, along with the mean and standard deviation of AvgCorrect values. Bold indicates that the mean is not significantly different from that of the algorithm with the largest mean value; that is, bold indicates the highest scores. The boxplots and means for the other datasets are shown in Figures 9 through 18 in Appendix B.

Figure 8 summarizes the results of all of the experiments by totaling the number of datasets for which each algorithm achieved one of the highest scores. The best performing algorithms were $K = 0.01$ and $K = 0.1$, both algorithms achieving one of the top scores for 6 out of 7 of the datasets. The single dataset for which these algorithms did not achieve one the best scores was the very dense dataset T20I18D10KN22CR50, where the algorithms requiring a higher support (the max support algorithm, and also the adjusted confidence algorithm for $K = 5, 10$, and 15) achieved the highest AvgCorrect score. In that case, the $K = 0.01$ and $K = 0.1$ algorithms still performed better than the max confidence, min support algorithms for the parameters we tried.

The adjusted confidence algorithm with a very small K is similar to using the max confidence algorithm, except that whenever there is a tie, the tie is broken in favor of the rule with largest support. It seems that in most of the datasets we chose, this type of algorithm performed the best, which indicates two things. First, that for some datasets, increasing K too much can have the same effect as a too-large minimum support threshold, in that large values of K could potentially remove the best rules, leading to too much bias, and where the algorithm cannot explain enough of the variance in the data. Second, when comparing rules, it is important not to break ties at random as in the max confidence, min support algorithm, but instead to use the support of the rules. Another observation is that the performance levels of the adjusted confidence algorithm vary less than those of the max confidence, min support algorithm. In other words, our experiments indicate that a less-than-perfect choice of K for the adjusted confidence algorithm is likely to perform better than a less-than-perfect choice of θ for the max confidence, min support algorithm.

6. Related Works

We provide background on related works within several fields: association rule mining and associative classification, decision lists, recommender systems, and Bayesian analysis. There is also a body of literature on pattern mining in sequences, but not in the sequential

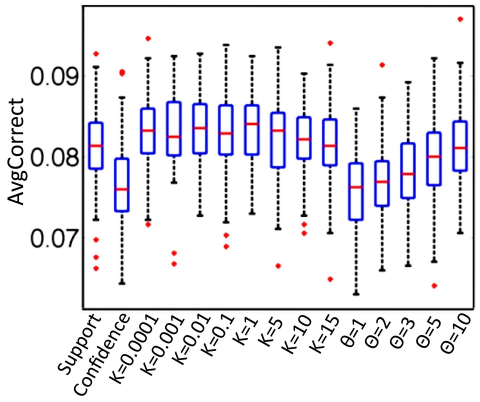


Figure 4: Boxplots of AvgCorrect values for Chess dataset.

Algorithm	mean ± standard dev.
Support	0.0813 ± 0.0046
Confidence	0.0764 ± 0.0053
K=0.0001	**0.0831 ± 0.0045**
K=0.001	**0.0832 ± 0.0048**
K=0.01	**0.0835 ± 0.0041**
K=0.1	**0.0831 ± 0.0049**
K=1	**0.0835 ± 0.0043**
K=5	0.0821 ± 0.0049
K=10	0.0821 ± 0.004
K=15	0.0816 ± 0.0049
θ=1	0.0759 ± 0.0049
θ=2	0.0767 ± 0.0045
θ=3	0.078 ± 0.0049
θ=5	0.0794 ± 0.0052
θ=10	0.0813 ± 0.0046

Figure 5: Means and standard deviations for Chess dataset.

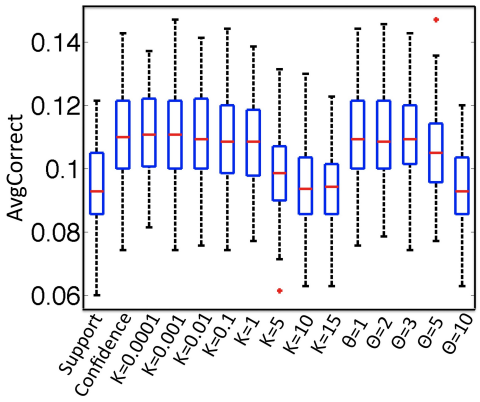


Figure 6: Boxplots of AvgCorrect values for MONK's problems dataset.

Algorithm	mean ± standard dev.
Support	0.0943 ± 0.0126
Confidence	**0.1103 ± 0.0145**
K=0.0001	**0.1108 ± 0.0137**
K=0.001	**0.1109 ± 0.0147**
K=0.01	**0.1104 ± 0.0149**
K=0.1	**0.11 ± 0.0151**
K=1	0.1081 ± 0.0148
K=5	0.0992 ± 0.0138
K=10	0.0947 ± 0.0133
K=15	0.0948 ± 0.012
θ=1	**0.1098 ± 0.0138**
θ=2	**0.1095 ± 0.0146**
θ=3	**0.1092 ± 0.0146**
θ=5	0.1054 ± 0.0143
θ=10	0.0944 ± 0.0129

Figure 7: Means and standard deviations for MONK's problems dataset.

event prediction setting defined here. This type of work generally considers the order in which items are added, and often uses a Markov assumption (see, for instance, Ayres et al., 2002; Berchtold and Raftery, 2002), whereas in our work, subsets of items are used to predict the next item, possibly without regard to the order in which they occurred, and a Markov assumption can be false. There is also work relating statistics to pattern mining and sequence mining, (*e.g.*, Chernoff bounds for the confidence, Jacquemont et al., 2009). Our work also relates to multi-class classification, since there is a multi-class classification step at each point in time t of each sequence. For a recent work on generalization bounds in multi-class classification see Shen and Wang (2007). Remember that in multi-class classification, each example is a feature vector, whereas in sequential event prediction, each example is

Algorithm	Number of datasets
Support	1
Confidence	1
K=0.0001	4
K=0.001	5
K=0.01	6
K=0.1	6
K=1	2
K=5	2
K=10	2
K=15	2
θ=1	1
θ=2	1
θ=3	1
θ=5	0
θ=10	1

Figure 8: Summary of experiments: For each algorithm, the number of datasets where it performed comparably with the best algorithm.

an event sequence. Related work on generalization bounds includes those on algorithmic stability (Devroye and Wagner, 1979; Bousquet and Elisseeff, 2002).

Mining Association Rules

Association rule mining has proven successful for many applications, including market basket analysis (cross selling, product placement, affinity promotion, see also Kohavi et al., 2004), mining gene expression data (Jiang and Gruenwald, 2005), and weblog analysis (Huang et al., 2002). The majority of literature on association rule mining concerns the design of efficient algorithms to address the time-and-memory-consuming task of mining rules within very large databases. Discovering rules is usually a two-step process. First, itemsets are mined that meet a predetermined minimum support threshold. Then using this set, rules are formed and the strength of the rules is assessed using "interestingness" measures, such as the confidence. Many "interestingness" measures have been proposed in the literature (see Tan et al., 2002; Geng and Hamilton, 2007). It is clearly possible to use the adjusted confidence as an interestingness measure for database exploration. In that setting, the adjusted confidence would provide a ranking of rules in terms of their ability to predict, including both "common sense rules" and "nuggets."

Although association rule mining has proven successful for many applications, it is well-known that the usefulness of association rules and their impact on even a wider range of practical applications remains limited due to problems arising from the minimum support threshold: first, the large number of rules mined can be intractable to domain experts who analyze rules and act on them, unless the minimum support threshold is set to a large value; second, the heuristic choice of the minimum support threshold tends to over-prune the search space of association rules, disregarding "nuggets" which can be very useful in many applications. Most prior work relies on the strong requirement of the minimum support threshold; some exceptions include the works of Li et al. (1999); Koh (2008) and DuMouchel and Pregibon (2001). Some recent work (Cohen et al., 2001; Wang et al., 2001)

attempts to avoid the support measure altogether. In our work, the use of the adjusted confidence eliminates the need for the minimum support threshold.

When a set of rules is used to form a classifier, this is called "associative classification" (see, for instance, Liu et al., 1998).

Decision Lists

A decision list is an ordered set of association rules that forms a classifier (Rivest, 1987). Usually decision lists are formed the same way as decision trees are formed, which is by greedily splitting on each nodes to form the tree, and then pruning. However, it is possible to mine a set of rules, and order them to produce a classifier, as in the associative classification literature. The work of Bertsimas et al. (2011) uses training data to learn the ordering of rules to form a decision list for multi-class classification.

The work of Anthony (2004) contains a generalization bound for decision lists, but each rule in the list requires a linear combination, which is problematic in the sequential setting by the reasoning in Appendix A. (Similarly, there are many papers using a set of pre-computed rules as features for supervised learning, where a linear combination of rules is constructed, rather than a decision list; one recent example is by Friedman and Popescu 2008.)

Recommender Systems

Association rule mining has proven to be particularly useful for finding "goes with" relationships between items purchased simultaneously. Lin et al. (2002) also construct a recommender system using rules, having a minimum confidence threshold and then an adjustable minimum support threshold. Their scoring system is essentially based on support $\times$ confidence, which is not an estimate of $P(b|a)$ for rule $a \to b$. Lawrence et al. (2001) provide a recommender system for a grocery store, but the setting differs entirely from ours in that they always recommend items that have never been previously purchased.

In other work, we designed a Bayesian framework that estimates K for the adjusted confidence by "borrowing strength" across both users and items (McCormick et al., 2012). We are also looking at different approaches to the sequential event prediction problem, where we allow the predictions to alter the sequence in which items are placed into the basket (Letham et al., 2011). This work uses a supervised learning framework for sequential event prediction.

Often, item-based collaborative filtering is used for problems that are actually sequential event prediction problems. There are several problems in applying standard item-based collaborative filtering techniques in sequential event prediction, the first one being that standard item-based collaborative filtering requires us to compute a similarity measure between all "co-rated" items. The similarity measure is often symmetric between two items, there is no distinguishing between $P(a|b)$ and $P(b|a)$. Even if item b is *always* found when a is found, $P(b|a) = 1$, is it possible for b not to be recommended when a is present, even with more than sufficient data to see the pattern. Further, for an incomplete basket, we do not have the ratings for all "co-rated" items, since there is no natural way to differentiate between items that have not yet been purchased in this transaction, and items that will not be purchased in this transaction, as both have a "rating" of 0 at time t. Thus, the only ratings that are available are ratings of "1" indicating that an item is in the basket. In other words, where the association rule approach we present here is intrinsically sequential, it is

unnatural to force item-based collaborative filtering into a sequential framework. In general, item-based collaborative filtering is not based in a machine learning framework, in that it is not based on either loss minimization or probabilistic modeling (as the association rule approach is). The work of Letham et al. (2011) also shows experimentally that item-based collaborative filtering can be worse than the max-confidence association rule approach.

Bayesian Analysis

DuMouchel and Pregibon (2001, "D&P") present a Bayesian approach to the identification of interesting itemsets. While not a rule mining algorithm per se, the approach could be extended to produce rules. D&P consider the ratio of observed itemset frequencies to baseline frequencies computed under a particular independence model. A prior distribution over the collection of such ratios results in shrinkage estimates for the true ratios. The amount of shrinkage depends on the observed frequency and tends to be more pronounced for less frequent itemsets. Our approach differs from D&P in several key regards. Most importantly we focus directly on Bayesian estimation for rules rather than itemsets. Second, D&P use an empirical Bayes approach to choose the prior hyperparameters. Since our approach requires just a single hyperparameter, K, we instead let the user choose an appropriate value (the value might be determined by cross validation or empirical Bayes). Finally, D&P perform a stratified analysis; one interesting future direction for our proposed approach would be to incorporate stratification.

Breese et al. (1998) present a number of different algorithms for collaborative filtering, including two Bayesian approaches. One of their Bayesian approaches clusters users while the other constructs a Bayesian network. Condliff et al. (1999) present a hierarchical Bayesian approach to collaborative filtering that "borrows strength" across users. Neither Breese et al. nor Condliff et al. focus on repeated purchases but both present ideas and techniques that may have relevance to future versions of our approach, especially the borrowing strength ideas.

7. Conclusion

This work synthesizes tools from several fields to analyze the use of association rules in a new supervised learning framework. This analysis is necessarily different from that of classical supervised learning analysis; as we have discussed, association rules provide two mechanisms for generalization: first a large sample, and second, a minimum support of rules. We considered two simple algorithms based on association rules: a max confidence, min support algorithm, and the Bayesian adjusted confidence algorithm. Both algorithms have a parameter that creates a bound on the support, regulating a tradeoff between accuracy on the training set and generalization ability. We have also demonstrated that the adjusted confidence introduced here has some advantages over the minimum support threshold that is commonly considered in association rule mining: it allows rare rules to be used while still encouraging generalization, and among rules with similar confidence, it prefers those with larger support.

Acknowledgments

C. Rudin is also at the Center for Computational Learning Systems, Columbia University. This work was performed partly while E. Kogan was at Fresh Direct. We would like to acknowledge support for this project from the National Science Foundation under grant IIS-1053407.

References

Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules. In *Proc. 20th Int'l Conf. Very Large Data Bases, (VLDB)*, pages 487–499. Morgan Kaufmann, 1994.

Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. In *Proc. ACM SIGMOD Int'l Conference on Management of Data*, pages 207–216, 1993.

Martin Anthony. Generalization error bounds for threshold decision lists. *Journal of Machine Learning Research*, 5:189–217, 2004.

Jay Ayres, J. E. Gehrke, Tomi Yiu, and Jason Flannick. Sequential PAttern Mining using bitmaps. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002.

André Berchtold and Adrian E. Raftery. The mixture transition distribution model for high-order markov chains and non-gaussian time series. *Statistical Science*, 17(3):328–356, 2002.

Dimitris Bertsimas, Allison Chang, and Cynthia Rudin. Ordered rules for classification: A discrete optimization approach to associative classification. Operations Research Center Working Paper Series OR 386-11, MIT, 2011.

Olivier Bousquet and André Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2:499–526, 2002.

John S. Breese, David Heckerman, and Carl Myers Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proc. Uncertainty in Artificial Intelligence (UAI)*, pages 43–52, 1998.

Edith Cohen, Mayur Datar, Shinji Fujiwara, Aristides Gionis, Piotr Indyk, Rajeev Motwani, Jeffrey D. Ullman, and Cheng Yang. Finding interesting associations without support pruning. *IEEE Trans. Knowl. Data Eng.*, 13(1):64–78, 2001.

Michelle Keim Condliff, David D. Lewis, David Madigan, and Christian Posse. Bayesian mixed-effects models for recommender systems. In *ACM SIGIR Workshop on Recommender Systems: Algorithms and Evaluation*, 1999.

Luc Devroye and T. J. Wagner. Distribution-free performance bounds for potential function rules. *IEEE Transactions on Information Theory*, 25(5):601–604, 1979.

William DuMouchel and Daryl Pregibon. Empirical bayes screening for multi-item associations. In *Proc. ACM SIGKDD Int'l Conf. on Knowl. Discovery and Data Mining*, pages 67–76, 2001.

A. Frank and A. Asuncion. UCI machine learning repository, 2010. URL `http://archive.ics.uci.edu/ml`.

Jerome H. Friedman and Bogdan E. Popescu. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2(3):916–954, 2008.

Liqiang Geng and Howard J. Hamilton. Choosing the right lens: Finding what is interesting in data mining. In *Quality Measures in Data Mining*, pages 3–24. Springer, 2007.

Xiangji Huang, Aijun An, Nick Cercone, and Gary Promhouse. Discovery of interesting association rules from Livelink web log data. In *Proc. IEEE Int'l Conf. on Data Mining (ICDM)*, 2002.

Peter Jackson. *Introduction to Expert Systems (Third Edition)*. Addison-Wesley, 1998.

Stéphanie Jacquemont, François Jacquenet, and Marc Sebban. A lower bound on the sample size needed to perform a significant frequent pattern mining task. *Pattern Recogn. Lett.*, 30:960–967, August 2009.

Xiang-Rong Jiang and Le Gruenwald. Microarray gene expression data association rules mining based on BSC-tree and FIS-tree. *Data & Knowl. Eng.*, 53(1):3–29, 2005.

Norman Lloyd Johnson, Adrienne W. Kemp, and Samuel Kotz. *Univariate Discrete Distributions*. John Wiley & Sons, August 2005.

Yun Sing Koh. Mining non-coincidental rules without a user defined support threshold. In *Advances in Knowl. Discovery and Data Mining, 12th Pacific-Asia Conf., (PAKDD)*, pages 910–915, 2008.

Ron Kohavi, Llew Mason, Rajesh Parekh, and Zijian Zheng. Lessons and challenges from mining retail e-commerce data. *Machine Learning*, 57(1-2):83–113, 2004.

R.D. Lawrence, G.S. Almasi, V. Kotlyar, M.S. Viveros, and S.S. Duri. Personalization of supermarket product recommendations. *Data Mining and Knowledge Discovery*, 5(1-2): 11–32, 2001.

Ben Letham, Cynthia Rudin, and David Madigan. A supervised ranking approach to sequential event prediction. In Preparation, 2011.

Benjamin Letham, Cynthia Rudin, and Katherine Heller. Growing a list. Operations Research Center Working Paper Series OR 393-12, MIT, 2012.

Jinyan Li, Xiuzhen Zhang, Guozho Dong, Kotagiri Ramamohanarao, and Qun Sun. Efficient mining of high confidence association rules without support thresholds. In *Proc. Principles of Data Mining and Knowledge Discovery (PKDD)*, pages 406–411, 1999.

Weiyang Lin, Sergio A. Alvarez, and Carolina Ruiz. Efficient adaptive-support association rule mining for recommender systems. *Data Mining and Knowledge Discovery*, 6(1): 83–105, 2002.

Bing Liu, Wynne Hsu, and Yiming Ma. Integrating classification and association rule mining. In *Proceedings of the 4^{th} International Conference on Knowledge Discovery and Data Mining (KDD)*, 1998.

David Madigan, Krzysztof Mosurski, and Russell G Almond. Graphical explanation in belief networks. *Journal of Computational and Graphical Statistics*, 6:160–181, 1997.

Tyler H. McCormick, Cynthia Rudin, and David Madigan. Bayesian hierarchical modeling for predicting medical conditions. *The Annals of Applied Statistics*, 6(2):652–668, 2012.

Grzegorz A. Rempala. Asymptotic factorial powers expansions for binomial and negative binomial reciprocals. In *Proceedings of the American Mathematical Society*, volume 132, pages 261–272, 2003.

Ronald L. Rivest. Learning decision lists. *Machine Learning*, 2(3):229–246, 1987.

W. H. Rogers and T. J. Wagner. A finite sample distribution-free performance bound for local discrimination rules. *Annals of Statistics*, 6(3):506–514, 1978.

V. Romanovsky. Note on the moments of a binomial $(p+q)^n$ about its mean. *Biometrika*, 15:410–412, 1923.

Cynthia Rudin, Benjamin Letham, Ansaf Salleb-Aouissi, Eugene Kogan, and David Madigan. A framework for supervised learning with association rules. In *Proceedings of the 24th Annual Conference on Learning Theory (COLT)*, 2011.

Xiaotong Shen and Lifeng Wang. Generalization error for multi-class margin classification. *Electronic Journal of Statistics*, pages 307–330, 2007.

Pang N. Tan, Vipin Kumar, and Jaideep Srivastava. Selecting the right interestingness measure for association patterns. In *Proc. ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining*, 2002.

Vladimir Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, September 1999.

Ke Wang, Yu He, David Wai-Lok Cheung, and Francis Y. L. Chin. Mining confident rules without support requirement. In *Proc. Conference on Information and Knowledge Management (CIKM)*, pages 89–96, 2001.

Appendix A. Regression and the Sequential Event Prediction Problem

By using association rules to model conditional probabilities for the sequential event prediction problem, we make a general assumption about the Markov chains governing our application, namely that a subset of knowledge about the current state can be used to predict the most likely future state. In this section we will address the suitability of two natural regression approaches that do not make this assumption. Let X_i be an indicator variable that is 1 if item i is in the current basket and 0 otherwise.

First regression method: Apply regression (*e.g.*, logistic regression) to create a model for each item separately. Consider the model for the last item (item m), where the predictor variables will be X_i for $i \in \{1, \ldots, m-1\}$, and X_m will be the response variable. This model would provide:

$$P(X_m = 1 | X_1 = x_1, \ldots, X_{m-1} = x_{m-1}) = \frac{1}{1 + \exp(f)},$$

where $f = \sum_{i=1}^{m-1} \lambda_i x_i + \lambda_{0,m}$, with each $x_i \in \{0, 1\}$.

Because the data are being revealed sequentially, the correct application of this technique is not straightforward. Only a *partial* basket is available when predictions need to be made. It is incorrect to substitute the current state of the basket directly into the formula above. For instance, if the current basket contains items 1 and 2, so $X_1 = 1$ and $X_2 = 1$, it is incorrect to write $P(X_m | X_1 = 1, X_2 = 1) = \frac{1}{1+\exp(f)}$, where $f = \lambda_1 + \lambda_2 + \lambda_{0,m}$. This statement would be equivalent to the expression:

$$P(X_m = 1 | X_1 = 1, X_2 = 1) = P(X_m = 1 | X_1 = 1, X_2 = 1, X_3 = 0, \ldots, X_{m-1} = 0),$$

which is clearly false in general. It is not that, for instance, $X_3 = 0$, it is simply that X_3 is not yet realized.

On the other hand, it is possible to integrate in order to obtain conditional probability estimates:

$$P(X_m = 1 | X_1 = 1, X_2 = 1) = \sum_{x_3 = \{0,1\}, \ldots, x_{m-1} = \{0,1\}} P(X_m = 1 | X_1 = 1, X_2 = 1, X_3 = x_3 \ldots, X_{m-1} = x_m) \times P(X_3 = x_3, \ldots, X_{m-1} = x_m),$$

where estimates of $P(X_3 = x_3, \ldots, X_{m-1} = x_m)$ would need to be made also for every one of the 2^{m-3} combinations of $x_3, \ldots x_{m-1}$. Thus, this approach would rely on a large number of uncertain estimates (given limited data, and even moderately large m), each introducing errors into the final estimate. This is in contrast to the association rule approaches where a class of conditional probabilities are directly estimated. Further, the regression method provided above would not be able to be explained easily to customers or managers. In most circumstances, it would also require a large amount of computation between recommendations. Finally, it is not clear how to incorporate the order in which items are placed into the basket within this type of model, whereas it is trivial to incorporate this into the association rule techniques as discussed in Section 2.2.

Second regression method: Apply regression methods (*e.g.*, logistic regression) for each item and at each timestep, in total $m \times T$ regression models, where T is the size of the largest

possible basket. This would give a direct way to incorporate time into the predictions. If the current basket contains t items, one would use only the models constructed using the first t items in each basket to predict the next item to be added. However, this would be making an entirely different assumption than the one given by the rule-mining approach. The rule-mining approach uses time only implicitly, and purchase patterns are counted the same regardless of the exact time within the transaction when the pattern occurred. In contrast, this regression approach would ignore all items added after time t in previous baskets. If apples were always followed by oranges, but in the past apples and oranges were always added after timestep t, then this approach would fail to recommend oranges when apples are added before timestep t. Further, the models for each timestep t must be constructed from baskets at least as large as t. This means that for very large baskets, there would only be a few past baskets that could be used to construct the models. Further, if the current basket is larger than any of the past baskets, the models would be trivial, since none of the past baskets can be used to construct them.

It may indeed be possible to use regression approaches for the sequential event prediction problem, but given the discussion above, it is not clear how this should be accomplished. We explore other ways to solve the sequential event prediction problem using supervised ranking techniques in another work (Letham et al., 2011).

Appendix B. Additional Experimental Results

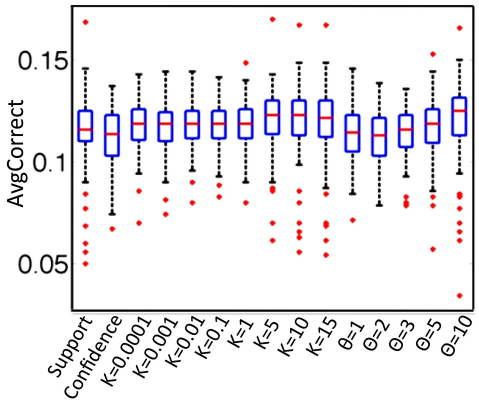


Figure 9: Boxplots of AvgCorrect values for Cars dataset.

Algorithm	mean ± standard dev.
Support	0.115 ± 0.0176
Conf.	0.1125 ± 0.0143
K=0.0001	0.1173 ± 0.0127
K=0.001	0.1163 ± 0.0122
K=0.01	**0.1176 ± 0.0117**
K=0.1	**0.1177 ± 0.0109**
K=1	0.1176 ± 0.0116
K=5	**0.1204 ± 0.015**
K=10	**0.1199 ± 0.0172**
K=15	**0.1192 ± 0.0174**
θ=1	0.1133 ± 0.0134
θ=2	0.1119 ± 0.0131
θ=3	0.114 ± 0.0118
θ=5	0.1161 ± 0.0143
θ=10	**0.1205 ± 0.0191**

Figure 10: Means and standard deviations for Cars dataset.

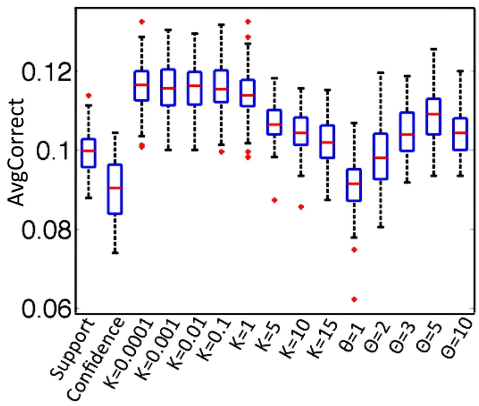


Figure 11: Boxplots of AvgCorrect values for Mushroom dataset.

Algorithm	mean ± standard dev.
Support	0.0996 ± 0.0051
Confidence	0.0902 ± 0.0075
K=0.0001	**0.1164 ± 0.0061**
K=0.001	**0.1158 ± 0.0062**
K=0.01	**0.1161 ± 0.0061**
K=0.1	**0.116 ± 0.0058**
K=1	0.1142 ± 0.0062
K=5	0.1069 ± 0.0052
K=10	0.1044 ± 0.0054
K=15	0.1024 ± 0.0053
θ=1	0.0909 ± 0.007
θ=2	0.0986 ± 0.0077
θ=3	0.1048 ± 0.0064
θ=5	0.1088 ± 0.0069
θ=10	0.1042 ± 0.0057

Figure 12: Means and standard deviations for Mushroom dataset.

Appendix C.

Lemma 17 *For $t \sim Binomial(m, p)$ and $K \geq 0$,*

$$\mathbb{E}\left[\frac{1}{K+t}\right] = \frac{1}{K+mp} + O\left(\frac{1}{m^2}\right). \tag{23}$$

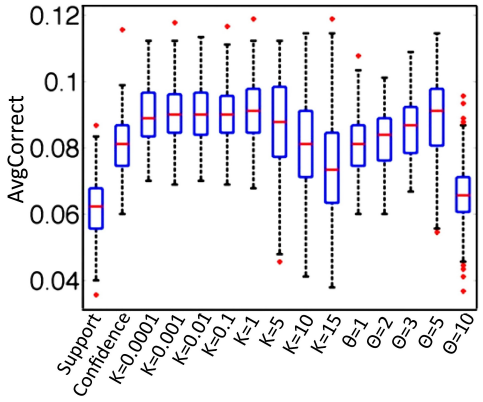


Figure 13: Boxplots of AvgCorrect values for Nursery dataset.

Algorithm	mean ± standard dev.
Support	0.0619 ± 0.0098
Confidence	0.081 ± 0.0094
K=0.0001	0.0898 ± 0.0091
K=0.001	**0.0902 ± 0.0093**
K=0.01	**0.0902 ± 0.0085**
K=0.1	**0.0903 ± 0.0095**
K=1	**0.0909 ± 0.0096**
K=5	0.0869 ± 0.0139
K=10	0.0804 ± 0.0154
K=15	0.0747 ± 0.0154
θ=1	0.0811 ± 0.0088
θ=2	0.0819 ± 0.0094
θ=3	0.0858 ± 0.0095
θ=5	0.0883 ± 0.0137
θ=10	0.0654 ± 0.0111

Figure 14: Means and standard deviations for Nursery dataset.

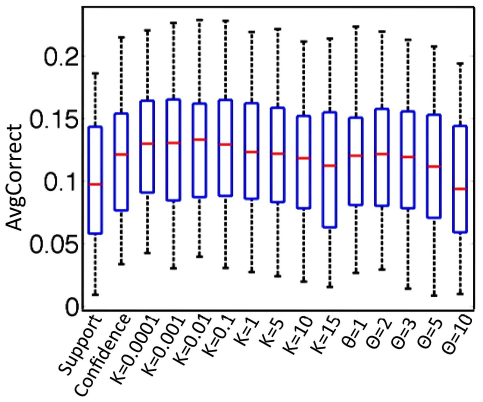


Figure 15: Boxplots of AvgCorrect values for Plants dataset.

Algorithm	mean ± standard dev.
Algorithm	mean pm standard dev.
Support	0.0983 ± 0.0494
Confidence	0.1187 ± 0.0465
K=0.0001	**0.1271 ± 0.0448**
K=0.001	**0.1251 ± 0.0454**
K=0.01	**0.1255 ± 0.0446**
K=0.1	**0.1251 ± 0.0464**
K=1	0.1235 ± 0.0454
K=5	0.1205 ± 0.0466
K=10	0.1141 ± 0.0464
K=15	0.1093 ± 0.0498
θ=1	0.1182 ± 0.0457
θ=2	0.1182 ± 0.0466
θ=3	0.118 ± 0.047
θ=5	0.11 ± 0.0511
θ=10	0.0981 ± 0.0496

Figure 16: Means and standard deviations for Plants dataset.

The proof of this lemma for $K = 0$ is provided by Rempala (2003). The proof of this lemma for $K > 0$ comes from (Letham et al., 2012), which we provide here for completeness. The proof of the lemma uses the following result.

Lemma 18 *Let* $X \sim \text{Binomial}(m, p)$ *and let* $\mu_k = \mathbb{E}\left[(X - \mathbb{E}[X])^k\right]$ *be the* k^{th} *central moment. For integer* $k \geq 1$, μ_{2k} *and* μ_{2k+1} *are* $O\left(m^k\right)$.

Proof We will use induction. For $k = 1$, the central moments are well known (*e.g.*, Johnson et al., 2005): $\mu_2 = mp(1-p)$ and $\mu_3 = mp(1-p)(1-2p)$, which are both $O(m)$. We rely

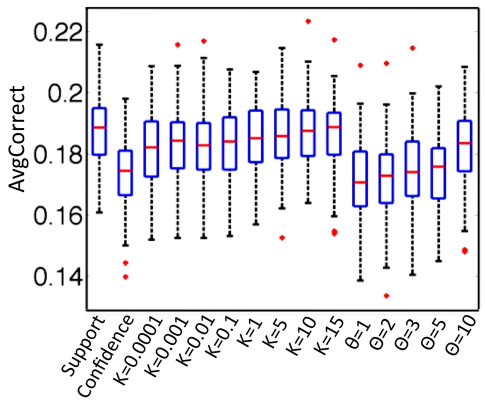


Figure 17: Boxplots of AvgCorrect values for T20I18D10KN22CR50 dataset.

Algorithm	mean ± standard dev.
Support	**0.1874 ± 0.0115**
Confidence	0.1728 ± 0.0118
K=0.0001	0.1817 ± 0.012
K=0.001	0.1827 ± 0.0121
K=0.01	0.1821 ± 0.0124
K=0.1	0.183 ± 0.0125
K=1	0.1843 ± 0.0117
K=5	**0.1857 ± 0.0119**
K=10	**0.1871 ± 0.0115**
K=15	**0.1867 ± 0.0116**
θ=1	0.1722 ± 0.0126
θ=2	0.1716 ± 0.0128
θ=3	0.1748 ± 0.0131
θ=5	0.1742 ± 0.0125
θ=10	0.182 ± 0.0125

Figure 18: Means and standard deviations for T20I18D10KN22CR50 dataset.

on the following recursion formula (Johnson et al., 2005; Romanovsky, 1923):

$$\mu_{s+1} = p(1-p)\left(\frac{d\mu_s}{dp} + ms\mu_{s-1}\right). \tag{24}$$

Because μ_2 and μ_3 are polynomials in p, their derivatives will also be polynomials in p. This recursion makes it clear that for all s, μ_s is a polynomial in p whose coefficients include terms involving m.

For the inductive step, suppose that the result holds for $k = s$. That is, μ_{2s} and μ_{2s+1} are $O(m^s)$. Then, by (24),

$$\mu_{2(s+1)} = p(1-p)\left(\frac{d\mu_{2s+1}}{dp} + (2s+1)m\mu_{2s}\right). \tag{25}$$

Differentiating μ_{2s+1} with respect to p yields a term that is $O(m^s)$. The term $(2s+1)m\mu_{2s}$ is $O(m^{s+1})$, and thus $\mu_{2(s+1)}$ is $O(m^{s+1})$. Also,

$$\mu_{2(s+1)+1} = p(1-p)\left(\frac{d\mu_{2(s+1)}}{dp} + 2(s+1)m\mu_{2s+1}\right). \tag{26}$$

Here $\frac{d\mu_{2(s+1)}}{dp}$ is $O(m^{s+1})$ and $2(s+1)m\mu_{2s+1}$ is $O(m^{s+1})$, and thus $\mu_{2(s+1)+1}$ is $O(m^{s+1})$.

This shows that if the result holds for $k = s$ then it must also hold for $k = s+1$ which completes the proof. ■

We can now prove Lemma 17.

Proof (Of Lemma 17) We expand $\frac{1}{K+X}$ at $X = mp$:

$$\begin{aligned}\mathbb{E}\left[\frac{1}{K+X}\right] &= \mathbb{E}\left[\sum_{i=0}^{\infty}(-1)^i\frac{(X-mp)^i}{(K+mp)^{i+1}}\right] \\ &= \sum_{i=0}^{\infty}(-1)^i\frac{\mathbb{E}\left[(X-mp)^i\right]}{(K+mp)^{i+1}} \\ &= \frac{1}{K+mp} + \sum_{i=2}^{\infty}(-1)^i\frac{\mu_i}{(K+mp)^{i+1}} \end{aligned} \tag{27}$$

where μ_i is the i^{th} central moment and we recognize that $\mu_1 = 0$. By Lemma 18,

$$\frac{\mu_i}{(K+mp)^{i+1}} = \frac{O\left(m^{\lfloor\frac{i}{2}\rfloor}\right)}{O\left(m^{i+1}\right)} = O\left(m^{\lfloor\frac{i}{2}\rfloor-i-1}\right). \tag{28}$$

The alternating sum in (27) can be split into two sums:

$$\sum_{i=2}^{\infty}(-1)^i\frac{\mu_i}{(K+mp)^{i+1}} = \sum_{i=2}^{\infty}O\left(m^{\lfloor\frac{i}{2}\rfloor-i-1}\right) = \sum_{i=2}^{\infty}O\left(\frac{1}{m^i}\right) + \sum_{i=3}^{\infty}O\left(\frac{1}{m^i}\right). \tag{29}$$

These are, for m large enough, bounded by a geometric series that converges to $O\left(\frac{1}{m^2}\right)$. ■