

# Discovering Frequent Pattern Pairs

Carlos Ordonez, Zhibo Chen  
Department of Computer Science  
University of Houston  
Houston, TX 77204, USA

## Abstract

Cubes and association rules discover frequent patterns in a data set, most of which are not significant. Thus previous research has introduced search constraints and statistical metrics to discover significant patterns and reduce processing time. We introduce cube pairs (comparing cube groups based on a parametric statistical test) and rule pairs (based on two similar association rules), which are pattern pair generalizations of cubes and association rules, respectively. We introduce algorithmic optimizations to discover comparable pattern sets. We carefully study why both techniques agree or disagree on the validity of specific pairs, considering p-value for statistical tests, as well as confidence for association rules. In addition, we analyze the probabilistic distribution of target attributes given confidence thresholds. We also introduce a reliability metric based on cross-validation, which enables an objective comparison between both patterns. We present an extensive experimental evaluation with real data sets to understand significance and reliability of pattern pairs. We show cube pairs generally produce more reliable results than rule pairs.

**Keywords:** association rules, cube, itemset, statistical test, reliability

## 1 Introduction

This work focuses on the generalization of OLAP cubes [14] and association rules [5, 15, 13] to discover frequent pattern pairs. Cubes are generally used to perform exploratory ad-hoc analysis in a multidimensional manner, but they also have the potential to find predictive patterns. Association rules [15], on the other hand, are frequent patterns within a data set found by an exhaustive level-wise algorithm. Constrained association rules [15, 12] exploit search constraints to filter and reduce the number of rules as well as to accelerate processing. The exhaustive search behind association rules and cubes makes it possible to discover specific relationships between predictive attributes (cube dimensions) and target attributes (cube measures). However, based on a specific predictive pattern (cube cell or association rule), it is difficult to determine which specific attribute plays a more important role in the implication (i.e., analogous to variable selection in regression). This fact motivates defining pairs of highly similar patterns isolating such “trigger” attribute.

Our contributions can be separated into two main areas: frequent pattern pairs and efficient algorithms. On the pattern discovery side, we introduce pattern pairs for both cubes and association rules as fundamental new patterns to understand predictive relationships between two subsets of attributes. We carefully study the reliability of such pattern pairs and introduce four complementary propositions to compare pattern pairs with each other. These propositions cover all potential cases when both techniques agree or disagree. Moreover, we provide a guideline to accept or reject pattern pairs based on those cases. Finally, we introduce a reliability metric based on cross-validation, which enables an objective experimental comparison. From an algorithmic perspective, we propose novel query-based techniques for both techniques so that only significant pattern pairs are discovered. We exploit the fact that the level-wise algorithms to explore cubes are highly similar to those used to discover association rules. First, cube statistical tests are extended with search constraints, mainly to mine a similar set of patterns to constrained association rules. With association rules we introduce novel post-processing techniques to match highly similar association rules on the same attributes, where one attribute in the antecedent triggers a major change in the value of the attribute in the consequent (i.e. similar to variable selection).

This paper is organized as follows. Section 2 explains state of the art and closely related work. Definitions and a running example are introduced in Section 3. Section 4 presents efficient algorithms to discover patterns pairs with cubes and association rules. Reliability is studied from a statistical perspective in Section 5. Section 6 presents

experiments with real data sets studying reliability, number of patterns, pattern set overlap and algorithm efficiency. Section 7 provides general conclusions along with directions for future research.

## 2 Related Work

Association rules and cubes are deeply related, both being combinatorial pattern search techniques. However, association rules [1] came before cubes [9, 11]. The cube operator was proposed in the seminal paper [6]. Most cube research has focused either on efficiently building a data cube [18] or on evaluating simple aggregations such as sums on individual subgroups of the cube [2]. More recently, constraints have been applied in cubes to further improve both running time and reduce the number of patterns. We exploit constraints not only to improve performance, but for trimming the number of patterns. Recently, cubes were also used to classify cancers in [23]. In previous research, we proposed an algorithm that embeds statistical tests into OLAP cubes to discover specific trigger dimensions that cause significant changes in one or more of the measure attributes [4]. However, as with the original association rules, we can often find a large number of patterns from small data sets. Therefore, it was necessary to incorporate search constraints into cubes to reduce the number of patterns. On the other hand, in [16] association rules are shown to be better than decision trees to predict multiple target attributes; the main reasons behind are tree overfit, data fragmentation and automated attribute binning. In short, cube pairs represent an advance over association rules and decision trees to understand multiple target attributes and to isolate predictive attributes.

Prediction cubes [3] store on each cube cell summaries based on a predictive model. Each cell stores information such as accuracy that can then be used to predict results when the dimensions are known. In [19], a new way of exploring the data cube is developed. This discovery-driven approach analyzes a cell value compared with the common trends of the neighboring cells to pinpoint exceptions, which can be used by the user as hints of areas requiring further analysis. Our research differs from these two papers in that we are able to pinpoint trigger dimensions by observing pairs of similar cells. Instead of attempting to predict the measure columns when given a set of dimensions, we are looking to find the specific dimensions whose change causes the measure values to change as well. Notice that our goal is not to create a predictive model, but to narrow down the search to those important dimensions that are causing significant changes.

Association rules [1, 15] have been used on many problems including disease prediction [15]. Search constraints were applied on association rules to both decrease the number of rules and to improve the running time of the algorithm [15, 17]. In these works, rules are analyzed individually and conclusions are based on each rule. Filtering out spurious rules is studied in [7], which proposes improved confidence and lift metrics which are more robust to noise in the data set. It is well known there is a tradeoff between rules with high support and rules with high confidence [20]; this work proposes an algorithm that mines the best rules under a Bayesian model. However, no search constraints are considered to find predictive rules. The idea of grouping association rules together to discover information that would otherwise be hidden was explored in [10]. The authors used OLAP cube operations to group association rules together and explored the context information provided by such grouping. Instead of finding global context information, our research approaches rule context from a variable selection viewpoint to find single dimension. Thus, though we also group rules, the method by which this is accomplished is widely different.

## 3 Definitions

In this section, we introduce definitions and a small example, used throughout the paper. Since cube pairs and rule pairs require different input data set and produce different patterns, we must present separate definitions for each. However, we provide a unifying framework for both techniques.

### 3.1 Cube Pairs

In cube pairs, we begin with the raw data set  $X(X_1, X_2, \dots, X_p, Y_1, Y_2, \dots, Y_q)$  with  $n$  records,  $p$  independent attributes, and  $q$  dependent attributes. This data set is transformed into the fact table  $F$  with  $n$  records,  $d$  dimensions, and  $e$  measure attributes [8], to compute multidimensional aggregations. The  $d$  dimensions are discrete while the  $e$  measure attributes are numerical. The mathematical structure of the OLAP cube is represented by a lattice. The lattice contains all possible combinations of dimensions ( $2^d$ ). In this paper, each combination (e.g.  $\{D_1, D_2\}$  or  $\{D_1, D_3\}$ ) is called a node. A node is further divided into groups, with each group representing a specific combination of values of the dimensions (e.g.  $\{D_1 = 1, D_2 = 0\}$ ,  $\{D_1 = 1, D_2 = 1\}$ ). In a cube pair there is a single dimension that has a

SAMPLE DATA SET $X$ .						CUBE STATISTICAL PAIRS INPUT DATA SET $F$ .					ASSOCIATION RULE PAIRS INPUT DATA SET $T$ .					
$i$	$X_1$ Age	$X_2$ BP	$X_3$ HighChol	$X_4$ Smoking	$Y_1$ Blockage	$i$	$D_1$ Age < 60 (Y/N)	$D_2$ BP (H/L)	$D_3$ HighChol (Y/N)	$D_4$ Smoking (Y/N)	$A_1$ Blockage	$AID$	$itemID$	Description	$i$	Items
1	23	120	0	0	0	1	0	0	0	0	0	0	0	AGI < 60	1	Age < 60, BP < 140, HighChol = 0, Smoking = 0, Blockage < 50
2	27	130	1	0	0	2	0	0	0	0	0	0	1	60 ≤ AGE	2	Age < 60, BP < 140, HighChol = 1, Smoking = 0, Blockage < 50
3	31	110	0	0	21	2	0	0	1	0	0	1	10	BP < 140	3	Age < 60, BP < 140, HighChol = 0, Smoking = 0, Blockage < 50
4	12	160	0	1	43	3	0	0	0	0	21	1	11	140 ≤ BP	4	Age < 60, BP ≥ 140, HighChol = 0, Smoking = 1, Blockage < 50
5	46	150	0	0	45	4	0	1	0	1	43	2	20	HighChol = 0	5	Age < 60, BP ≥ 140, HighChol = 0, Smoking = 0, Blockage < 50
6	61	110	1	0	55	5	0	1	0	0	45	2	21	HighChol = 1	6	Age > 60, BP < 140, HighChol = 1, Smoking = 0, Blockage ≥ 50
7	68	100	1	0	57	6	1	0	1	0	55	3	30	Smoking = 0	7	Age > 60, BP < 140, HighChol = 1, Smoking = 0, Blockage ≥ 50
8	71	180	1	1	79	7	1	0	1	0	57	3	31	Smoking = 1	8	Age > 60, BP ≥ 140, HighChol = 1, Smoking = 1, Blockage ≥ 50
9	89	170	0	1	100	8	1	1	1	1	79	4	40	Blockage < 50	9	Age > 60, BP ≥ 140, HighChol = 0, Smoking = 1, Blockage ≥ 50
10	98	170	1	1	100	9	1	1	0	1	100	4	41	50 < Blockage	10	Age > 60, BP ≥ 140, HighChol = 1, Smoking = 1, Blockage > 50
						10	1	1	1	1	100					

Figure 1: Data set  $X$ , cube pairs fact data set  $F$  and rule pairs transaction data set  $T$ .

different value to identify a “trigger” attribute. We do not consider cube pairs differing in two or more dimensions because it makes difficult identifying predictive relationships. To reduce the number of patterns, the cube size constraint  $k$ , determines the maximum number of dimensions that can appear in a cube pair.

### 3.2 Rule Pairs

We now define a pair of association rules, which is simply called a rule pair. In rule pairs, we begin with the raw data set  $X(X_1, X_2, \dots, X_p, Y_1, Y_2, \dots, Y_q)$  with  $n$  records,  $p$  independent attributes, and  $q$  dependent attributes. Then, we transform  $X$  into  $D$  with  $n$  transactions and  $m$  binary dimensions. The binary dimensions data set  $D$  must then be transformed into a transaction format. Thus, pre-processing is required in the form of binning the dimensions into two buckets and pivoting the data set to create the transaction data set  $T$  with  $n$  transactions and  $I$  items.  $T$  would take the form of  $T = \{T_1, T_2, \dots, T_n\}$  with  $I = \{i_1, i_2, \dots, i_m\}$ , where  $T_i \subseteq I$ . Note that since negation is required,  $I$  would contain twice as many items. We call any subset of  $I$  an itemset.

An association rule is a predictive pattern of the form  $A \Rightarrow B$ , where  $A$  and  $B$  are two itemsets such that  $A \subseteq I$ ,  $B \subseteq I$  and  $A \cap B = \emptyset$ . In these rules, the antecedent is  $A$  with  $j$  items and the consequent is  $B$  with  $w$  items. Rules are often limited by a maximum number of items,  $k$ , that can appear. Association rules are evaluated using two metrics: support  $s()$  and confidence  $c()$ . Support  $s(A \Rightarrow B)$  is the fraction of transactions that contain  $A \cup B$ . Confidence is defined as  $c(A \Rightarrow B) = s(A \cup B)/s(A)$ . For an association rule to be considered valid, it must pass two user-defined thresholds: confidence threshold  $\phi$  and support threshold  $\psi$ . A rule pair is the combination of two association rules which differ in one dimension in the antecedent and have items coming from the same attribute on the consequent. We do not consider rules differing in two or dimensions.

### 3.3 Example

Let us now discuss a data set that we will use throughout this paper. Figure 1 shows a medical data set,  $X$  with  $n=10$ ,  $p=4$ , and  $q=1$ , where we show both mathematical notation and attribute names. This data set is in its raw format and needs to be preprocessed. In  $X$ , there are four independent attributes  $X_1, \dots, X_4$  and one dependent attribute  $Y_1$  (Blockage a.k.a. artery disease). In order to use  $X$  in either technique, we need to transform it. The same figure also shows the transformed data set,  $F$ , that is used as a fact input table for cube pairs. Finally, the figure shows the transaction input table  $T$  for rule pairs. Notice how attribute values and ranges were transformed into items in  $T$ .

## 4 Frequent Pattern Pairs

The algorithms we now present extend and combine previous work on cube statistical tests [4] and constrained association rules [15], used to discover frequent patterns. Basically we study how to build pairs of patterns from both techniques: pairs of cube groups and pairs of association rules, as defined in Section 3. To achieve such goal, we first study how to incorporate search constraints into cubes. On the other hand, we study how to build pairs of rules so that patterns from both techniques can be compared. We start by discussing an algorithm to get cube pairs and based on the same framework we then explain how to compute rule pairs. Figure 2 presents an overview of our algorithms.

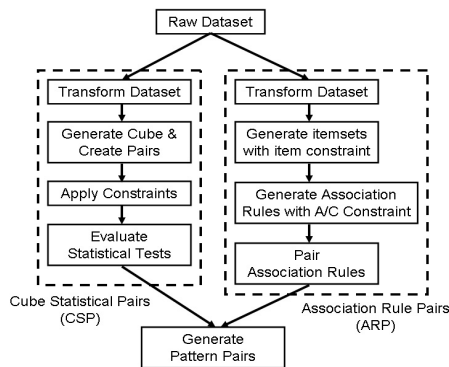


Figure 2: Overview of techniques.

## 4.1 Cube Pairs Algorithm

In previous research, we developed the cube statistical tests algorithm [4] which combines OLAP cubes and parametric statistical tests to discover patterns in data sets. The statistical test is used to compare two similar populations (similar cube dimensions), and return the significance of mean difference in a measure attribute.

Even though a cube algorithm does not return as many patterns as constrained association rules [15], it still can return thousands of patterns for a relatively small data set: the main reason is the lack of search constraints. Thus we first study how to incorporate such search constraints into our algorithm. In this case, we are using search constraints in OLAP cube statistical tests to reduce the set of discovered patterns. We approached this task by first analyzing the three main constraints used in constrained association rules [15]: item filtering, antecedent/consequent (AC), and item grouping. The first designates which items will be used to form the itemsets. We can apply this constraint in cube pairs by removing dimensions that will be filtered. The second constraint determines the location of items in an association rule: either in the antecedent or in the consequent. We emphasize this constraint is not applicable in cube pairs because dimension attributes can only appear on the “independent” side of the patterns, whereas those designated as measures will only appear on the “predicted” side. The third constraint places items into groups and restricts them from appearing in the same itemset. Since the cube pairs algorithm did not have this feature, we added optional input criteria and further changed the dimension grouping step. This optional input informs the algorithm of which dimensions or measures are in the same group. With this knowledge, we altered the dimension grouping step to avoid all nodes that contain dimensions belonging to the same group. We call this new algorithm *cube pairs*, which is the technical term we will use in the remainder of the paper. The main steps of the algorithm are as follows:

1. Aggregate the fact table  $F$  into cube  $C$ , at the finest granularity by aggregating measures based on all  $d$  dimensions.
2. Compute sufficient statistics per cube group for each cuboid up to size  $k \leq d$  (i.e. a subset of all  $2^d$  cuboids). Such sufficient statistics are efficiently computed in one pass.
3. Compute the mean  $\mu$  and standard deviation  $\sigma$  from sufficient statistics per group.
4. Build pairs of cube groups differing in one dimension.
5. For each cube pair evaluate statistical test on all measure attributes to determine significance level p-value.
6. Filter cube pairs that are deemed significant (whose p-value is lower than a given threshold).

In this algorithm, when comparing a pair of populations, we consider them to be significantly different if the null hypothesis,  $H_0$ , can be rejected. We assume that we are dealing with two independent populations with sizes  $n_1$  and  $n_2$ , means  $\mu_1$  and  $\mu_2$  and standard deviations  $\sigma_1$  and  $\sigma_2$ . In statistical terms, the null hypothesis is  $H_0: \mu_1 = \mu_2$ , complemented by the alternative hypothesis,  $H_1: \mu_1 \neq \mu_2$ . The goal is to find all pairs of data subsets in which we can reject  $H_0$  with high confidence, thus accepting  $H_1$ . In this case, high confidence is equal to  $1 - \text{p-value}$ , where p-value depends on a user-defined threshold. In general, p-value has thresholds of 0.10, 0.05, or 0.01 [21]. The smaller p-value, the higher the confidence toward rejecting  $H_0$ . In this algorithm, the calculation of the p-value comes from the











<u>Cube Statistical Pairs</u> p-value threshold at 0.01		<u>Association Rule Pairs</u> Confidence threshold at 0.7	
[{Age<60} → {Blockage=22} ; {Age≥60} → {Blockage=78} p-value<0.01	 (1)	[[Age<60] ⇒ {Blockage<50} ; conf=1.0, sup=0.5 {Age≥60} ⇒ {Blockage≥50}] conf=1.0, sup=0.5	
[BP<140,HighChol=0} → {Blockage=11} ; {BP<140,HighChol=1} → {Blockage=37} p-value>0.1	 (2)	[[BP<140,HighChol=0} ⇒ {Blockage<50} ; conf=1.0, sup=0.2 {BP<140,HighChol=1} ⇒ {Blockage≥50}] conf=0.7, sup=0.2	
[{BP<140} → {Blockage=27} ; {BP≥140} → {Blockage=73} ] p-value<0.01	 (3)	[[BP<140} ⇒ {Blockage<50} ; conf=0.6, sup=0.5 {BP≥140} ⇒ {Blockage≥50}] conf=0.6, sup=0.5	
[{Smoking=0} → {Blockage=7} ; {Smoking=1} → {Blockage=93} ] p-value<0.01	 (4)	[[Smoking=0} ⇒ {Blockage<50} ; conf=0.67, sup=0.3 {Smoking=1} ⇒ {Blockage≥50}] conf=0.75, sup=0.3	
[{Age<60,BP<140} → {Blockage=7} ; {Age≥60,BP≥140} → {Blockage=93} ] p-value<0.01	 (5)	[[Age<60,BP<140} ⇒ {Blockage<50} ; conf=1.0, sup=0.3 {Age≥60,BP≥140} ⇒ {Blockage≥50}] conf=1.0, sup=0.3	

Figure 3: Cube pairs and rule pairs from example data set.

means comparison parametric test, a statistical test that compares two populations based on their mean and variance. Considering variance is fundamental because it provides discrimination in cases when the distance between  $\mu_1$  and  $\mu_2$  is not enough to discern absence of similarity or distribution overlap.

EXAMPLE: Figure 3 shows several examples of pattern pairs from cube pairs. Pair (1) can be translated to mean there is a significant change in the Blockage attribute value between the populations with age less than 60 and those greater than or equal to 60.

## 4.2 Rule Pairs Algorithm

The constrained association rules algorithm [15] was developed to reduce the large amount of rules and to reduce processing time. The basis behind this algorithm is the use of search constraints to filter rules and items, as explained in Section 4.1. Nevertheless, association rules lack specific information about predictive attributes. We only know the antecedent implies the consequent, but we cannot identify which item(s) cause a change in the consequent.

In order to provide constrained association rules with the ability to isolate predictive attributes, we developed a rule pairs algorithm which builds pairs of similar association rules. In developing this algorithm, we assume that if we discover two rules with similar antecedents, but complementary consequents on the predicted attribute (an attributed binned into two ranges), then we can pinpoint specific “predictive” items. Accomplishing this requires three main steps: (1) Analyzing a pair of rules,  $r_A$  and  $r_B$  and checking  $\phi$  and  $\psi$ . (2) Determining if  $r_A$  is similar to  $r_B$ , as explained below. (3) Storing the rule pair. To determine whether the two rules are similar, we ensure the following properties: (1) both rules have the same number of items in the antecedent; (2) all items in the antecedent, except one, are the same; (3) the total number of differing items in the antecedent is one, and those two different items come from the same attribute. (4) both rules involve the same attribute on the consequent. We exclude generalizations to two or more different items.

EXAMPLE: Figure 3 contains several potential rule pairs. Pairs (1) and (2) illustrate cases when both association rules pass all user thresholds. The final three pairs fail for a variety of reasons. Pair (3) and pair (4) are not accepted because at least one of the association rules has a confidence below the threshold of  $\phi = 0.7$ , while pair (5) differs in two items in the antecedent.

## 4.3 Differences between both Types of Pattern Pairs

Even though cube pairs and rule pairs are similar due to our extensions, there exist two major differences between them. First, there is a fundamental difference in the representation of  $k$ , depth constraint for cube pairs and itemset length for rule pairs. For instance, when cube depth is set to  $k$ , it limits the patterns for cube pairs to a maximum of  $k$  dimensions. On the other hand, when the itemset length is set to  $k$ , it restricts the length of the frequent itemset, not the size of association rules. However, since the itemsets for antecedent and consequent are both obtained from long itemsets, the sum of their lengths cannot be greater than the length of the long itemset. Therefore, the sum of the number of items in the antecedent and consequent cannot be greater than  $k$ . Since each association rule will always have at least one item in the consequent, cube pairs will be able to find longer patterns than rule pairs for the same  $k$ .

The second major difference is found in  $\psi$ , the minimum “support” threshold for cube pairs and the support threshold for rule pairs. For cube pairs, this threshold applies only to dimension combinations, excluding measures. On the other hand, rule pairs applies this threshold on the itemset which is the union of the antecedent and consequent, as defined before. Specifically,  $X \Rightarrow Y$  will pass the  $\psi$  threshold only if there are at least  $\psi n$  transactions containing both  $X$  and  $Y$  (i.e. with support  $\psi$ ). Thus, for the same  $\psi$  on both techniques, we expect cube pairs to find more patterns than rule pairs because the former restricts only to side  $X$  of the pattern as opposed to both sides  $X$  and  $Y$ .

## 5 Reliability Analysis

We start by defining a reliability metric. We then compare the reliability between cube pairs and rule pairs. We first explain cases where both techniques agree. Then we explain cases in which the two techniques disagree. Finally we provide a recommendation guideline to choose one technique as the most reliable.

### 5.1 Measuring Reliability

After carefully studying alternatives, we determined the best way to measure reliability is with the train and validate approach, traditionally used in statistical learning. Specifically, we use a 2-fold cross validation procedure to determine which patterns are found in both the train and validate data sets. The first step is to split the raw data set,  $X$ , into two groups: a train data set,  $X_t$ , and a validate data set,  $X_v$ . These two data sets are later transformed into  $F_t$  and  $F_v$  for cube pairs and  $T_t$  and  $T_v$  for rule pairs, respectively. The respective algorithms are then applied to each data set to obtain the list of final pattern pairs  $\mathcal{P}$ . In the following discussion,  $P(F)$  represents the list of all pattern pairs while  $|PF_t|$  represents the number of pairs found when computing cube pairs on the data set  $F_t$ .

Once we have obtained the pattern pairs from both sets, we compute a reliability metric by computing the ratio of number of patterns found in both the train and validate sets and the number of patterns found only in the training set. In other words, we compute an intersection and then a division to obtain the percentage of pattern pairs appearing in the validation set (the closer to 1 the better). The equation for computing our reliability metric for cube pairs is shown in Equation 1.

$$r(\mathcal{P}, F) = \frac{|(P(F_t) \cap P(F_v))|}{|P(F_t)|} \quad (1)$$

The equation for rule pairs is analogous to cube pairs (basically exchanging patterns and data set), except that  $F_t$  is replaced by  $T_t$  and  $F_v$  is replaced by  $T_v$ . Hence  $r(\mathcal{P}, T) = |(P(T_t) \cap P(T_v))|/|P(T_t)|$ . This is a valuable metric because it is necessary to know if the discovered patterns will actually appear again in a new data set or they are just local patterns. With each validation run, the train and validate data sets will vary slightly. Therefore, if the reliability metric varies a lot with different train and validate sets, then the technique is not robust towards changes in the data set. On the other hand, if the metric shows little variability then we conclude the technique is robust.

The overall goal of our reliability metric is to compare the quality of patterns discovered by both techniques. To obtain an average of this metric, we use a 2-fold cross validation process. The steps are as follows: (1) Partition input data  $F$  into two groups:  $F_t, F_v$ . (2) Compute patterns  $\mathcal{P}$  on both  $F_t$  and  $F_v$  to obtain the result sets  $P(F_t)$  and  $P(F_v)$ , respectively. (3) Compute reliability using Equation 1. (4) Switch  $F_t$  and  $F_v$ , recompute patterns and recompute Equation 1. (5) The final reliability metric is the average of the two metrics found in both iterations. By using the 2-fold cross validation, we are randomizing the train and validate data sets to avoid any unexpected bias that could lead to wrong conclusions. A larger number  $k > 2$  of groups cannot be used, as in  $k$ -fold cross validations, because data subsets become too small (fragmented) for either technique to effectively discover frequent patterns. Under a uniform experimental setup, we will apply 2-fold cross validation on both cube pairs and rule pairs.

### 5.2 Impact of Confidence and p-value on Pattern Pairs

We will now analyze, in a general manner, the effect that confidence and p-value thresholds have on final results. Let us assume that we are observing two cube groups from  $F$  (i.e., populations in statistical terms), as shown in Figure 4. Suppose data subsets  $A$  and  $B$  from  $F$  are being compared based on the measure value  $M$ . Subset  $A$ , can be further divided into two peaks,  $X_{A1}$  and  $X_{A2}$  while  $B$  can also be separated into two peaks,  $X_{B1}$  and  $X_{B2}$ . The x-axis represents  $M$ , while the y-axis indicates the size of each group. For example,  $x_{A1}$  stands for the value of the measure attribute for group  $A_1$  and the value  $y_{A1}$  represents the population that shares the same dimensions and the same measure attribute value. We have also included  $\mu_A, \mu_B, \theta_A$  and  $\theta_B$ , in order to calculate the p-value $_{AB}$ . For further

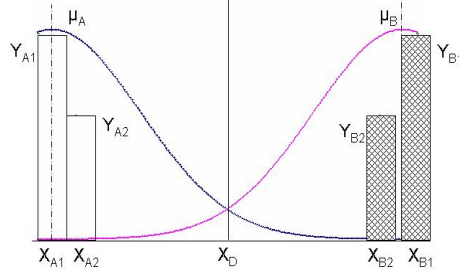


Figure 4: Proposition 1:  $c(r_A) \geq \phi$ ,  $c(r_B) \geq \phi$  (ACCEPT), and  $p\text{-value} \leq \rho$  (ACCEPT).

analysis, assume that the measure value is binned into two sections that are separated at  $S_M$ . We must emphasize the p-value may be sensitive to noise in the data, especially when the support threshold is low.

In terms of rule pairs, we consider  $A$  to contain the set of items,  $I_A$ , while  $B$  contains the item set,  $I_B$ . Figure 4 represents two different rules:  $r_A$  ( $\{I_A\} \Rightarrow \{M < S_M\}$ ) and  $r_B$  ( $\{I_B\} \Rightarrow \{M \geq S_M\}$ ). For the remainder of this section, we consider  $c(r_A)$  and  $s(r_A)$  to be the confidence and support of  $r_A$  and  $c(r_B)$  and  $s(r_B)$  to be the same for  $r_B$ . Notice that we assume  $I_A$  and  $I_B$  differ in one item. As a result, if  $c(r_A) \geq \phi$ ,  $s(r_A) \geq \psi$ ,  $c(r_B) \geq \phi$ , and  $s(r_B) \geq \psi$ , then we can form the rule pair of  $[\{I_A\} \Rightarrow \{M < S_M\}; \{I_B\} \Rightarrow \{M \geq S_M\}]$ .

For cube pairs, we are able to compute z-test and p-values as follows:

$$\text{z-test}_{AB} = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma_A^2}{N_A} + \frac{\sigma_B^2}{N_B}}} \quad (2)$$

$$\begin{aligned} \text{p-value} &= \int_z^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \\ &= \frac{2}{\sqrt{\pi}} \sum_{n=0}^{\infty} \frac{(-1)^n z^{2n+1}}{n!(2n+1)} \\ &\approx \frac{2}{\sqrt{\pi}} \sum_{n=0}^4 \frac{(-1)^n z^{2n+1}}{n!(2n+1)} \end{aligned} \quad (3)$$

As we can observe, the z-test given in Equation (2), can be computed from  $\mu$  and  $\sigma$  of the populations. Equation (3) shows the integration of the normal distribution curve in order to obtain the p-value [22]. The sum can be truncated at  $n = 4$ , giving good precision since this series converges fast, as shown in Equation (3).

### 5.3 Propositions Comparing Reliability of Pattern Pairs

We now introduce four complementary propositions for pattern pairs in terms of confidence  $\phi$  and p-value indicating in parenthesis if the respective pair is accepted or rejected. We also provide examples obtained from the running example given in Section 3.3 to better illustrate our cases. Notice that we use the same  $\psi$  (support threshold) for both techniques to avoid confusion.

For all propositions let  $A \subset F$ ,  $B \subset F$  represent two data subsets behind a cube pair. Let  $r_A$  and  $r_B$  be a rule pair, referring to the same (transformed) data subsets, as defined above.

#### Proposition 1

If  $c(r_A) \geq \phi$  and  $c(r_B) \geq \phi$  (ACCEPT) and  $p\text{-value} \leq \rho$  (ACCEPT), then both techniques agree and we accept the pattern pair. If  $c(r_A) < \phi$  or  $c(r_B) < \phi$  (REJECT) and  $p\text{-value} > \rho$  (REJECT), then both techniques agree and we reject the pattern pair.

*Proof Sketch:* This proposition states that if two techniques agree on whether a pattern pair should be accepted or rejected, then we have a reliable decision. Since both techniques agree, there is no reason to doubt the categorization of that pattern pair. Figure 4 shows a case when both cube pairs and rule pairs agree. In this case, we can see a

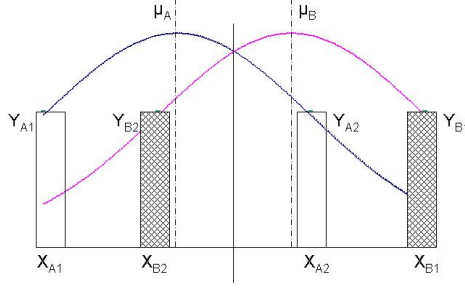


Figure 5: Proposition 1 (AGREE):  $c(r_A) < \phi$ ,  $c(r_B) < \phi$  (REJECT), and  $p\text{-value} > \rho$  (REJECT).

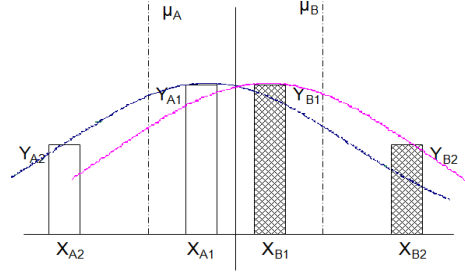


Figure 6: Proposition 2:  $c(r_A) \geq \phi$ ,  $c(r_B) \geq \phi$  (ACCEPT), and  $p\text{-value} > \rho$  (REJECT).

clear separation between  $A$  and  $B$ . It is important to notice that knowing the  $p$ -value would provide a relative size of separation since smaller values would indicate greater differences between the two populations. Similarly, Figure 5 shows a case in which both techniques reject the pattern pair. In this case, we can see that  $A$  and  $B$  are quite mixed, with no detectable separation. Once again, the  $p$ -value would provide a relative scale of similarity. For this proposition, there is little difference in the reliability of both techniques since they agree. However,  $p$ -value does provide additional knowledge of the difference that is lacking for confidence.

EXAMPLE: In the pattern pairs from our example data set, Figure 3, Pair(1) shows the case when both techniques agree. Thus the pair should be accepted.

### Proposition 2

If  $c(r_A) \geq \phi$  and  $c(r_B) \geq \phi$  (ACCEPT) and  $p\text{-value} > \rho$  (REJECT), then we reject the pattern pair.

*Proof Sketch:* This proposition covers the case when rule pairs accepts the pattern pair because both rules have confidences above  $\phi$ , the user-defined confidence threshold, while cube pairs rejects the pattern pair because the  $p$ -value is greater than the user-threshold. This situation is shown in Figure 6. Since a majority of the points for each lies on the "correct" side of the boundary point,  $S_M$ , both  $c(r_A)$  and  $c(r_B)$  are above  $\phi$ . If we analyze the figure, we can see that  $A$  and  $B$  are fairly distributed along  $M$ , not separated as the confidence would lead us to believe. We see that a majority of the points are close to  $S_M$ . In fact, nearly 40% of the data is within 10% of  $S_M$ . In this case, the  $p$ -value appears to be more reliable because it is less susceptible to clustered data.

EXAMPLE: Our example data set contains an example of this case in Figure 3(2). A valid rule pair has been found, but the  $p$ -value is too high.

### Proposition 3

If both  $c(r_A) < \phi$  and  $c(r_B) < \phi$  (REJECT) and  $p\text{-value} \leq \rho$  (ACCEPT), then we accept the pattern pair.

*Proof Sketch:* For this proposition, we consider the case when rule pairs reject a pattern pair based on low confidence in both rules while cube pairs accept the pattern pair based on a low  $p$ -value. Figure 7 shows such a scenario of when the  $p$ -value disagrees with rule confidence. Once again, we must analyze the probabilistic distribution of the actual data. We can see that a most data points are close to the  $S_M$  boundary. This means that a minor change of the location of the  $S_M$  boundary can change the value of the two confidences. It should be clear that  $A$  and  $B$  are quite



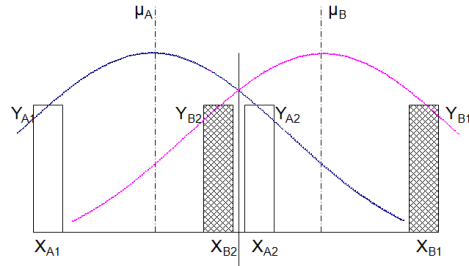


Figure 7: Proposition 3:  $c(r_A) < \phi$ ,  $c(r_B) < \phi$  (REJECT), and  $p\text{-value} \leq \rho$  (ACCEPT).

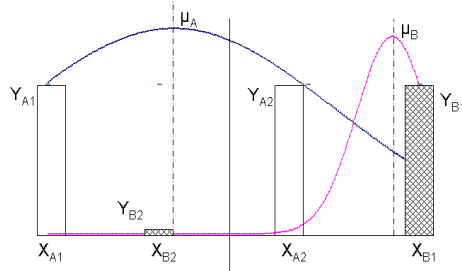


Figure 8: Proposition 4:  $c(r_A) < \phi$ ,  $c(r_B) \geq \phi$  (REJECT), and  $p\text{-value} \leq \rho$  (ACCEPT).  $p\text{-value}$  is correct.

separated from each another. The lack of a sense of probabilistic distance in confidence calculations prevents rules from catching this issue. On the other hand, the  $p\text{-value}$  used in cube pairs does consider the distance and distribution of the populations. Therefore, when both association rules in a pair have low confidences, then we must rely more on  $p\text{-value}$ .

**EXAMPLE:** Our example data set contains examples of this scenario as seen in Figure 3. Pair (3) illustrates the case when rule pairs rejects a pair due to low confidence, which is set at a threshold of 0.7. However, we can clearly see that cube pairs finds the  $p\text{-value}$  to be low enough to warrant acceptance. Further analysis revealed that cube pairs was indeed correct in accepting the pair because the distribution of the values placed some points very close to the threshold point of attribute *Blockage*.

#### Proposition 4

If  $c(r_A) \geq \phi$  and  $c(r_B) < \phi$  (REJECT) or  $c(r_A) < \phi$  and  $c(r_B) \geq \phi$  (REJECT) and  $p\text{-value} \leq \rho$  (ACCEPT), then we are unable to determine if the pattern pair should be accepted or rejected.

*Proof Sketch:* In this proposition, we are comparing the case when rule pairs rejects the pattern pair based on one of the rules having a low confidence while cube pairs accepts the pattern pair based on a low  $p\text{-value}$ . When this occurs,

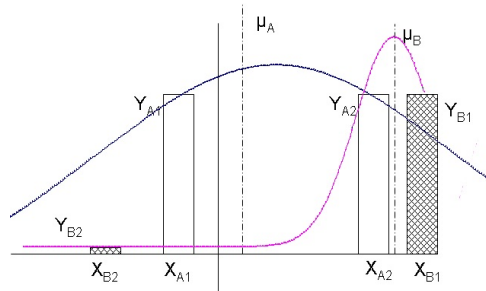


Figure 9: Proposition 4:  $c(r_A) < \phi$ ,  $c(r_B) \geq \phi$  (REJECT), and  $p\text{-value} \leq \rho$  (ACCEPT).  $p\text{-value}$  is incorrect.

Table 1: Comparing Pattern Pairs.

Rule 1 Conf	Rule 2 Conf	Rule pair	Cube pair	Decision
High	High	Yes	Accept $H_1$	Accept
High	High	Yes	Reject $H_1$	Reject
Low	High	No	Accept $H_1$	Inconclusive
Low	High	No	Reject $H_1$	Reject
High	Low	No	Accept $H_1$	Inconclusive
High	Low	No	Reject $H_1$	Reject
Low	Low	No	Accept $H_1$	Accept
Low	Low	No	Reject $H_1$	Reject

we are unable to determine which technique is more reliable based on available information. Figures 8 and 9 shows two situations where cube pairs fluctuates between accepting the pair and rejecting it. In the first case, we can see that the pattern does appear valid.  $A$  and  $B$  are quite different from one another. On the other hand, the second figure shows how p-value could be mistaken because even though the pair is marked as valid, the only difference between the two figures is a shifting of the values for  $A$  towards  $B$ . What we can deduce from these two examples is that p-value is only susceptible when  $A$  and  $B$  have very different distributions. When this occurs, the Gaussians of  $A$  and  $B$  have odd intersections. The problem is that with one narrow and one skewed Gaussian, the overlap may not be understood as much as it should be. Thus, we are unable to make a determination on the reliability of the techniques when the rules in a pair have conflicting confidences.

EXAMPLE: Our example data set also includes an example of this proposition. In this case, Figure 3 contains pair (4), which contains the situation where rule pairs rejects this pair because one of the rules has a confidence, 0.67, below the threshold of 0.7. However, the same pair is seen to contain a very low p-value and it is accepted by cube pairs. In this case, the Gaussians were close to one another, but different, and cube pairs arrived at the correct result.

## 5.4 Combining cube pairs and rule pairs Patterns

Based on the propositions introduced above, we developed the following rules to accept or reject patterns. Table 1 shows all possible combinations of both techniques. The first two columns cover low and high confidences of both association rules, while the third column shows the corresponding cube pair result. The last column shows a recommendation whether to accept or reject the pattern pair. In most cases, when the two techniques disagree, cube pairs is considered to be a more reliable technique. The cases when a cube pair is rejected occurs when the confidence of one association rule is high while the other one is low. That is, the pair is “incomplete”. On the other hand, if the cube pair is accepted, but a rule pair has low and high confidences then this is an indication the confidence threshold must be lowered or increased so that both rules agree.

## 6 Experiments

We now present an experimental validation with real data sets aiming to understand the quality and quantity of patterns as well as their reliability. Our experiments were performed on a DBMS server running Microsoft SQL Server with a 3.2GHz CPU, 4GB of RAM and 1TB of storage space. All our algorithms were programmed in Java and SQL. We begin by presenting the data sets. Then, we analyze reliability and experimentally verify our propositions. Finally, we evaluate algorithmic optimizations and time complexity.

### 6.1 Data Sets

For our experiments, we used three real data sets. The first data set is a financial data set ( $n=3000$ ,  $p=8$ ,  $q=5$ ) that deals with the costs before and after the opening of a new health center. Of the eight independent attributes, only one was not categorical and thus had to be pre-processed. In this case, the age of the patient was binned at 60. All five of the dependent attributes are numerical and were binned for use with association rules.

We analyzed two data sets from the UCI data set repository. The second data set ( $n=5984$ ,  $p=12$ ,  $q=3$ ) comes from the UCI repository and contains thyroid disease data. We used twelve of the independent attributes and binned the

Table 2: Rule pairs summary for financial data set.

Confidence	Support	Assoc. rules	Rule pairs
$\geq 0.5$	10%	3793	53
$\geq 0.5$	5%	5655	140
$\geq 0.5$	1%	11598	669
$\geq 0.7$	10%	2732	0
$\geq 0.7$	5%	4051	5
$\geq 0.7$	1%	7913	167

Table 3: Cube pairs summary for financial data set.

$p$ -value	Population	cube pair
$\leq 0.05$	10%	1062
$\leq 0.05$	5%	2013
$\leq 0.05$	2%	6531
$\leq 0.01$	10%	938
$\leq 0.01$	5%	1761
$\leq 0.01$	2%	5512

age attribute at 60. These attributes include medical information, such as whether the patient is pregnant or currently on thyroid medication. Then the three dependent attributes represent measurements found through blood tests. The third data set ( $n=6495$ ,  $p=11$ ,  $q=2$ ) contains data on various wines. The eleven independent attributes include various amounts of chemicals present in the wine as well as other specific chemical information such as pH and sulfates. All these attributes were numerical and were binned. On the other hand, the two dependent attributes represent the alcohol content and quality of each wine.

## 6.2 Matching Rule Pairs and Cube Pairs

The existing association rules algorithm discovers patterns involving cube dimensions and target attributes. By adding our new pairing procedure, we are able to pair two rules together to generate a new unified rule that can specifically point to one dimension as a probable cause of the rule consequent change. We now present interesting rule pairs using constrained association rules.

We present a summary of all the rules and rule pairs that were found in the financial data set in Table 2. We can see that as the various thresholds increase, the number of rules and rule pairs drastically drop. For the confidence threshold, the drop in the number of rules is not as dramatic as with the support threshold. This is in line with previous work [15].

The overall purpose of cube pairs is to discover specific dimensions that cause a difference in some disease measure attributes. Like other data mining techniques, a more specific goal is to find results that are surprising to the domain expert. Recall there are two main thresholds for cube pairs: the population threshold and the  $p$ -value threshold. We must emphasize the  $p$ -value may be sensitive to noise in the data, especially when the support threshold is low. Table 3 shows a breakdown of the number of results found with this technique when both the population and the  $p$ -value are varied. Notice that the population threshold has the most important influence on the final number of results while the  $p$ -value threshold has a much lesser effect.

## 6.3 Understanding Coverage of Patterns

For rule pairs, both patterns in the pair must pass all thresholds to be valid. However, it is often the case that one pattern passes all thresholds, while its partner pattern does not. On the other hand, cube pairs internally compare population sets to obtain the same type of patterns. We mainly used  $p$ -value and population fraction (i.e. support) as the two thresholds. In the next sections, we will analyze the amount of coverage (overlap) between the two techniques and break down those discovered patterns into groups according to the propositions given in Section 5.2. We consider the coverage between the patterns found by rule pairs and cube pairs to be the set of results that appears in both techniques.

Table 4: Summary of coverage.  $sup \geq 5\%$ ,  $lift \geq 1.2$ .

conf	<i>p</i> - value							
	cube pairs cover rule pairs				rule pairs cover cube pairs			
	0.10	0.05	0.01	0.001	0.10	0.05	0.01	0.001
0.4	97	95	86	81	11	11	11	12
0.5	98	97	92	89	6	7	8	8
0.6	100	100	100	100	2	3	3	4
0.7	100	100	100	100	1	1	3	1

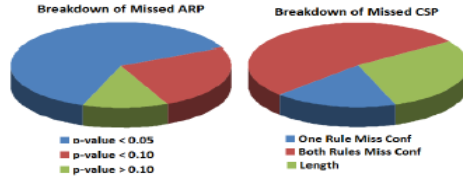


Figure 10: Breakdown of Missed Pattern Pairs.

In order for a pattern to be considered covered, both techniques must find the same item (rules) or dimension patterns (cubes) as well as have the same discriminating dimension.

In addition to the number of covered patterns, in order to have a complete analysis, we will also be analyzing the percentage of missed patterns. We calculate the percentage of missed patterns for technique A as a ratio of the number of non-covered patterns found by technique B to the total number of patterns found by technique B. In other words, the missed percentage will allow us to judge how well one technique covers the patterns discovered by the other technique. For example, if technique A has a missed percentage of 60% to technique B, then it means that technique A was unable to find 60% of the patterns discovered by technique B.

Table 4 shows coverage between the two techniques. The first number represents the number of rule pairs results that were found by cube pairs, while the second number represents the number of cube pairs results that were found by rule pairs. For both coverage values, there are two general trends: (1) with a constant confidence threshold, the percent covered decreases as the p-value threshold decreases and (2) with a constant p-value threshold, the percent covered increases as the confidence threshold increases.

We will now analyze the reasons why such high coverage occur. First, we will focus on the patterns that were discovered by rule pairs but were discarded by cube pairs. Results are shown in Figure 10. As explained above, the only reason for cube pairs to discard a pattern is due to the p-value being higher than the user-defined threshold. In our experiments, we found a total of 16 patterns discovered by rule pairs that were missed by cube pairs. To provide more detail, we have broken down the p-value threshold to three different levels:  $<0.05$ ,  $<0.10$ , and  $\geq 0.10$ . We can see that the first level contains the largest number of patterns while the last level contains the smallest. In most cases, when the p-value is very high, the confidence will be correspondingly low and would thus the pattern is discarded. Similarly, if a pattern has very low p-value, then the confidence will often be quite high.

We know analyze the reasons why rule pairs was unable to find the patterns discovered by Cube Statistical Pairs. Let us assume that the confidence threshold is set at 0.5, the support threshold is set at 5%, and the p-value threshold is set at 0.01. Figure 10 shows a breakdown of the reasons why some cube pairs patterns were filtered out by rule pairs. Recall that in order for rule pairs to detect a pattern, both rules that make up the pattern must pass all user-defined thresholds. There are three cases where a pattern will be discarded by this technique: (1) at least one rule has low confidence; (2) at least one of the rules has a low support; (3) the number of dimensions in the pattern exceeds the maximum length of association rules. The first two cases are fairly straightforward and involve rules that do not meet the required thresholds. The third case involves the difference in meaning between rule length and the lattice depth in cube pairs, as described in Section 4.3.

From our experiments, we detected 2895 patterns discovered by cube pairs were rejected by rule pairs. We can see that case one comprises a large majority of the reasons, which is precisely what we expected because of the large influence that confidence has on deciding whether a pattern can be filtered. We can further see that when a rule is missed due to confidence, it is twice as likely to have both rules fail as opposed to just one. We also found that for all

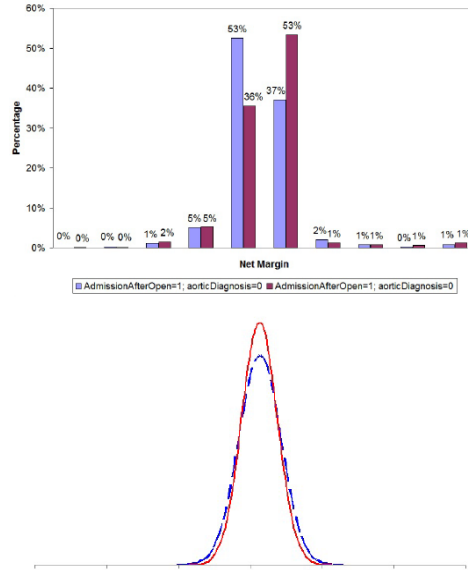


Figure 11: Proposition 2: Population Distribution and Gaussian.

the cases where support is too low, the confidence is below the threshold. As a result, we categorized those rules as missing the confidence threshold. In fact, nearly 65% of all missed rules has both a support and confidence that was below the user-defined thresholds.

## 6.4 Comparing Reliability Cases

We have now seen both similar and different patterns, discovered by the two techniques. In this section, we will separate patterns into the groups defined by the propositions introduced in Section 5.2. Since Proposition 1 is about the trivial case of when both techniques agree, we will focus on the remaining propositions where there is disagreement.

### Proposition 2

Proposition 2 stated that when both association rules that comprised a rule pair have confidence  $\geq \phi$ , then the p-value is more reliable. In our experiments, we found a total of 603 patterns with confidence  $\geq \phi$  for both rules. Of those patterns, cube pairs confirmed 587 patterns while rejecting 16 patterns.

We are mostly interested in understanding the 16 patterns where rule pairs and cube pairs disagree. We now show of these sixteen pattern pairs as follows:  $\{AdmissionAfterOpen = 0, DischargeDisposition = 0\} \rightarrow \{NetMargin < 0\}$ ;  $\{AdmissionAfterOpen = 0, DischargeDisposition = 1\} \rightarrow \{NetMargin \geq 0\}$ . This pattern was accepted by rule pairs, but was filtered out by cube pairs. Let us take a closer look at the breakdown of the populations to analyze why these two techniques disagreed. Figure 11 shows both the distribution of Net Margin within both of the population sets as well as the Gaussian from the populations. We can see that the majority of points are located near the middle, which is also where the boundary point lies for rule pairs. We can see from this setup that both association rules will pass the confidence threshold of 50% because each has a majority of data on one side of the boundary. As a result, rule pairs accept this pattern. On the other hand, if we use cube pairs to find the mean and standard deviations of the two population sets, we can see that the two populations are actually quite similar to one another. Thus this pattern is rejected.

### Proposition 3

Proposition 3 is the opposite of Proposition 2. It states that when both rules have confidences  $< \phi$ , then p-value is more reliable and should be treated as the final decision maker. For the financial data set, there are a total of 502 patterns in which both rules have confidences  $< \phi$ . In those, cube pairs confirmed the rejection of 153 patterns while the remaining 349 patterns were accepted. Figure 12 shows a breakdown of one of the 349 patterns in which cube pairs accepted the pattern and rule pairs rejected the pattern. This pattern is identified as:  $\{DischargeDisposition =$

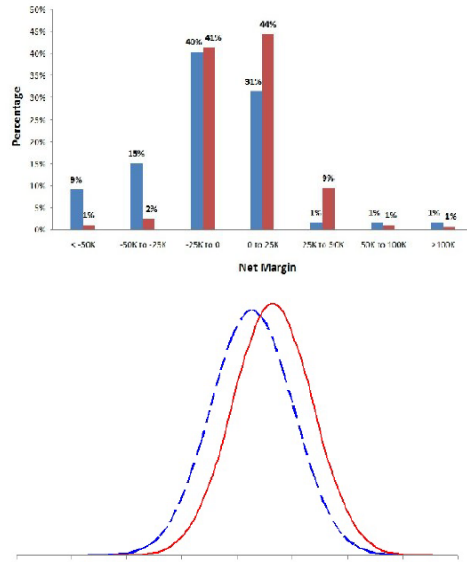


Figure 12: Proposition 3: Population distribution and Gaussian.

$0\} \rightarrow \{NetMargin < 0\}; \{DischargeDisposition = 1\} \rightarrow \{NetMargin \geq 0\}$ . Once again, the majority of the distribution is spread around the boundary point,  $S_M$ , of a Net Margin of 0. In this case, both association rules obtained confidences below the 50% threshold. However, since so many points are close to  $S_M$ , we cannot be sure that the populations are similar or not. If we look at the Gaussian breakdown of these two populations, also shown in Figure 12, we can see that they form two distinctly separate Gaussians. With such separation, cube pairs would accept this pattern.

#### Proposition 4

This proposition covers the remaining rule pairs patterns, which are the cases when the two rules have disagreeing confidences with regard to  $\phi$ . In this case, neither of the two techniques can be considered more reliable. We found a total of 490 patterns of which cube pairs accepted 339 patterns while it rejected 151 patterns.

In this case, there is no clear cut winner between the two techniques. Figures 13 and 14 show two population breakdowns in which we have conflicting results. The two patterns are as follows:  $\{\{OldAge = 0\} \rightarrow \{TotalCost < 0\}; \{OldAge = 1\} \rightarrow \{TotalCost \geq 0\}\}$  and  $\{\{OldAge = 1, InPatient = 0\} \rightarrow \{TotalCost < 0\}; \{OldAge = 1, InPatient = 1\} \rightarrow \{TotalCost \geq 0\}\}$ . In both examples, rule pairs results in two conflicting rules. If we look at the Gaussian distributions of these two cases, then we can observe two different results. For example, Figure 13 shows the two Gaussian curves for the first example. We can clearly see that there are two distinct "hills". However, the second example yields a different result, as shown in Figure 14. Here, we see two different Gaussian, but the amplitudes are widely different. As such, we would accept the first example while rejecting the second. Such conflicting results means that we cannot determine which patterns are more reliable.

## 6.5 Reliability Metric

Tables 5 and 6 show the percentage of validated patterns for both cube pairs and rule pairs on the three data sets at various confidence and p-value levels. These experiments show that in most cases, cube pairs retains more patterns between the train and validate sets. We conducted five separate 2-fold cross validation runs for each of the data sets. Each of the runs also includes different train and validate sets, each of which is randomly extracted from the full data set. We can observe that cube pairs consistently retains as much, if not more, of the training set patterns as rule pairs. This experiment shows that even with varying records within the training and validate sets, cube pairs is still able to maintain a better train-validate ratio. It is also important to note that the percent retained for cube pairs is more stable than for rule pairs. We see that for the two publicly available data sets (Thyroid and Wine), there are times when rule pairs has no pairs validated between the train and validate sets. After further analysis, we found that potential pairs

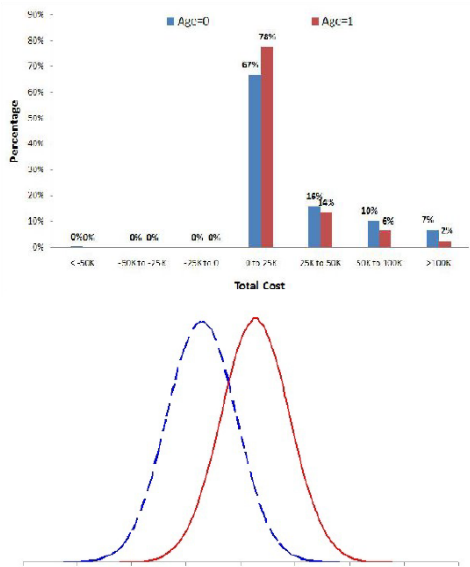


Figure 13: Proposition 4: Population distribution and Gaussian (cube pairs correct).

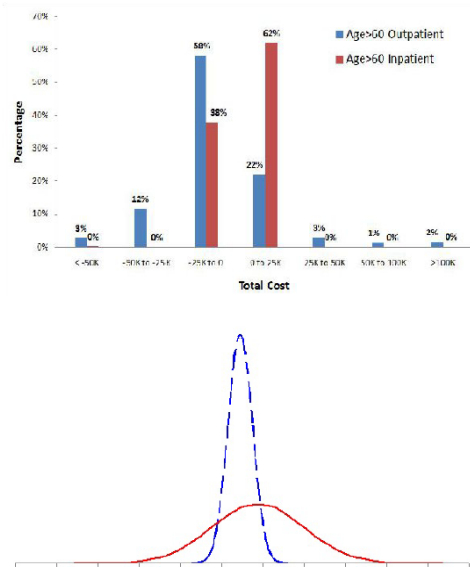


Figure 14: Proposition 4: Population distribution and Gaussian (cube pair incorrect).

Table 5: Reliability Metrics for cube pairs and rule pairs with  $p\text{-value} \leq 0.05$  and  $\text{confidence} \geq 0.5$ .

Data Set	Cube pairs		Rule pairs	
	Avg Num Patterns	% pass validation	Avg Num Patterns	% pass validation
Financial	2945	71	1618	87
Thyroid	6352	90	73	2
Wine	2818	86	50	1

Table 6: Reliability metrics for cube pairs and rule pairs with p-value  $\leq 0.01$  and confidence  $\geq 0.7$ .

Data Set	Cube pairs		Rule pairs	
	Avg Num Patterns	% pass validation	Avg Num Patterns	% pass validation
Financial	2515	70	56	42
Thyroid	5803	89	25	0
Wine	2561	82	0	0

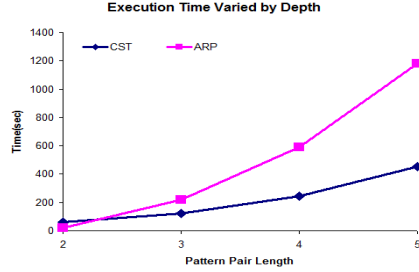


Figure 15: Times for cube pairs and rule pairs varying  $k$ .

were being discarded mainly due to one of the rules having a confidence that was slightly below the threshold of 0.5. The financial data set is also interesting in the sense that rule pairs actually beats cube pairs for the less restrictive thresholds of confidence at 0.5 and p-value at 0.05. This result appears to be because cube pairs found some erroneous results due to the “loose” high p-value limit. However, when we raised thresholds to confidence of 0.7 and p-value at 0.01, cube pairs again beats rule pairs by a wide margin. We would like to emphasize that these stringent thresholds are more commonly used than the looser thresholds.

## 6.6 Algorithmic Optimizations and Time Complexity

We now analyze the impact of our algorithmic optimizations on running time. There were three main performance optimizations. For cube pairs, the main optimization was exploiting attribute grouping (simply called group) constraints, which will reduce both the number of results and running time. On the other hand, there were two optimizations for rule pairs: filtering and strategic numbering of the items.

For cube pairs, three constraints were embedded into the algorithm: item filtering, antecedent/consequent (AC), and item grouping. Because the first two constraints were already embedded within the original OLAP cube test algorithm, we looked at the effect of the group constraint on performance time as shown in Table 7. The number of items grouped means that those items cannot appear in the same combination set. We can see that there is drastic improvement as we group more items together. This is expected because the more grouped items, the more nodes that can be excluded from the dimensions lattice.

For the filtering optimization, we observed that while filtering only marginally improves performance at high confidence thresholds (due to smaller number of rules), the algorithm becomes significantly faster as we decrease the confidence threshold. In fact, at high confidence thresholds, such as 0.7, this optimization improves performance by

Table 7: Time improvement due to group constraint in cube pairs.

Grouped Items	Running time	Improvement
1 of 8	452	0%
2 of 8	292	35%
3 of 8	203	55%
4 of 8	128	72%
5 of 8	80	82%



only 5%, but at a confidence threshold of 0.5, the improvement increases to 25%. These results confirm we are able to filter out many more rules.

For time complexity analysis, we varied  $k$ , the cube lattice depth constraint, to show how much restricting the length of the pattern pairs impacts the final performance as shown in Figure 15. Our experiments prove that the cube pairs algorithm has better performance than rule pairs. As  $k$  increases, cube pairs exhibits an almost linear increase while rule pairs exponentially increases with each  $k$ . The slowest step for cube pairs, taking over 35% of the execution time, is the actual grouping of the dimensions for the dimensional lattice. For rule pairs, the slowest step, taking nearly 70% of the execution time, was rule pair generation.

## 7 Conclusions

In this paper, we extended cubes and association rules to produce pattern pairs. On one hand, we used parametric statistical tests on cubes to compare highly similar cubes, to isolate one dimension triggering a significant change in some cube measure. On the other hand, we paired highly similar association rules differing on one item in the antecedent implying opposite ranges on an attribute binned into two intervals. We carefully compared their respective predictive metrics: p-value and confidence. We introduced four propositions to decide acceptance or rejection of pattern pairs based on when techniques agree or disagree. Also, we introduced a reliability metric based on two-fold cross validation, allowing a neutral and objective comparison. Basically, this metric compares the percentage of patterns discovered on the training data set that remain true on the validation data set. We introduced algorithmic optimizations to reduce the number of patterns and running time as well as to produce comparable sets of pattern pairs. We altered the cube algorithm with search constraints to reduce the number of patterns returned. On the other hand, we added a post-processing matching step to constrained association rules to obtain specific rule pairs. We then experimentally studied the reliability of the pattern pairs discovered by both techniques with our new reliability metric. We discovered that for almost all p-value and confidence threshold levels and data sets, cube pairs produced a higher number of patterns than rule pairs that passed thresholds on the validation set. We conclude that, based on our metric, cube pairs are more reliable than rule pairs.

Generalizing patterns into pairs introduces important research issues. We want to derive a single metric that summarizes support and confidence for a rule pair. We would like to understand how to isolate two predictive attributes, instead of one, giving them different weight depending on their impact on the predicted attribute. We want to understand if other different statistical tests can be combined with cubes. It is important to identify which search constraints work well with both kinds of patterns. Our proposed reliability metric based on cross-validation deserves further study, considering tradeoffs between p-value and confidence. Even though both cubes and association rules work in discrete space it is necessary to compare their predictive attribute identification with traditional variable selection from statistical learning models like regression or Bayesian classification.

## Acknowledgments

This research work was partially supported by National Science Foundation grant IIS 0914861.

## References

- [1] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *ACM SIGMOD Conference*, pages 207–216, 1993.
- [2] S. Chaudhuri and U. Dayal. An overview of data warehousing and OLAP technology. *ACM SIGMOD Record*, 26(1):65–74, 1997.
- [3] B. Chen, L. Chen, Y. Lin, and R. Ramakrishnan. Prediction cubes. In *Proc. of VLDB Conference*, pages 982–993, 2005.
- [4] Z. Chen, C. Ordonez, and K. Zhao. Comparing reliability of association rules and OLAP statistical tests. In *IEEE Reliability Issues in Knowledge Discovery Workshop (RIKD, ICDM Workshop)*, 2008.
- [5] M. Delgado, D. Sanchez, M.J. Martin-Bautista, and M.A. Vila. Mining association rules with improved semantics in medical databases. *Artificial Intelligence in Medicine*, 21(1-3):241–5, 2001.
- [6] J. Gray, A. Bosworth, A. Layman, and H. Pirahesh. Data cube: A relational aggregation operator generalizing group-by, cross-tab and sub-total. In *ICDE Conference*, pages 152–159, 1996.
- [7] M. Hahsler and K. Hornik. New probabilistic interest measures for association rules. *Intelligent Data Analysis*, 11(5):437–455, 2007.
- [8] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, San Francisco, 1st edition, 2001.
- [9] V. Harinarayan, A. Rajaraman, and J.D. Ullman. Implementing data cubes efficiently. In *ACM SIGMOD Conference*, pages 205–216, 1996.

- [10] B. Liu, K. Zhao, J. Benkler, and W. Xiao. Rule interestingness analysis using olap operations. In *ACM KDD*, pages 297–306, 2006.
- [11] A. Netz, S. Chaudhuri, J. Berhardt, and U. Fayyad. Integration of data mining with database technology. In *VLDB Conference*, 2000.
- [12] R. Ng, Laks Lakshmanan, and J. Han. Exploratory mining and pruning optimizations of constrained association rules. In *Proc. ACM SIGMOD Conference*, pages 13–24, 1998.
- [13] C. Ordonez. Association rule discovery with the train and test approach for heart disease prediction. *IEEE Transactions on Information Technology in Biomedicine (TITB)*, 10(2):334–343, 2006.
- [14] C. Ordonez and Z. Chen. Evaluating statistical tests on OLAP cubes to compare degree of disease. *IEEE Transactions on Information Technology in Biomedicine (TITB)*, 13(5):756–765, 2009.
- [15] C. Ordonez, N. Ezquerra, and C.A. Santana. Constraining and summarizing association rules in medical data. *Knowledge and Information Systems (KAIS)*, 9(3):259–283, 2006.
- [16] C. Ordonez and K. Zhao. A Comparison between Association Rules and Decision Trees to Predict Multiple Target Attributes. *Intelligent Data Analysis (IDA)*, 18(8), 2011.
- [17] J. Pei and J. Han. Constraints in data mining: Constrained frequent pattern mining: a pattern-growth view. *SIGKDD Explorations*, 4(1):31–39, 2002.
- [18] N. Roussopoulos, Y. Kotidis, and M. Roussopoulos. Cubetree: organization of and bulk incremental updates on the data cube. *ACM SIGMOD Record*, 26:89–99, June 1997.
- [19] S. Sarawagi, R. Agrawal, and N. Megiddo. Discovery-driven exploration of OLAP data cubes. In *EDBT*, pages 168–182. Springer-Verlag, 1998.
- [20] T. Scheffer. Finding association rules that trade support optimally against confidence. *Intelligent Data Analysis*, 9(4):381–395, 2005.
- [21] M. Triola. *Essentials of Statistics*. Addison Wesley, 2nd edition, 2005.
- [22] J. Tukey. *Exploratory Data Analysis*. Addison Wesley, 1977.
- [23] J. Wei, S. Wang, and X. Yuan. Ensemble rough hypercuboid approach for classifying cancers. *IEEE Transactions on Knowledge and Data Engineering*, 22(3):381–391, 2010.