1998

Data mining and epidemiologic surveillance.

Stephen Edward Brossette
University of Alabama at Birmingham

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

DATA MINING AND EPIDEMIOLOGIC SURVEILLANCE

by

STEPHEN E. BROSSETTE

A DISSERTATION

Submitted to the graduate faculty of The University of Alabama at Birmingham,
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

BIRMINGHAM, ALABAMA

1998

ABSTRACT OF DISSERTATION
GRADUATE SCHOOL, UNIVERSITY OF ALABAMA AT BIRMINGHAM

Degree Ph.D. Program Computer and Information Sciences

Name of Candidate Stephen E. Brossette

Committee Chair Warren T. Jones

Title Data Mining and Epidemiologic Surveillance

Data mining is the partially automated process of finding potentially interesting patterns in data. As a new discipline, it has been driven by the desire to find previously unknown, meaningful patterns in real-world databases. In this dissertation, I explore the use of data mining in epidemiologic surveillance both at the hospital and public health levels. As in many data mining research projects, our work on specific applications has fueled the development of more generally applicable ideas, strategies, and methods.

In this document, we describe our research, including the development of the Data Mining Surveillance System, DMSS. We also give experimental results obtained by using DMSS to analyze clinical laboratory infection control data obtained from University of Alabama at Birmingham Hospital, and invasive *Streptococcus pneumoniae* data obtained from the Centers for Disease Control and Prevention. Analysis of the results show that DMSS can efficiently identify interesting, complex, and previously-unknown patterns in epidemiologic data sets.

We believe that DMSS and systems like it will be indispensable tools in hospital infection control and public health surveillance systems of the future.

ACKNOWLEDGMENTS

I arrived in Birmingham nearly five years ago to become an M.D./Ph.D. student at UAB, and on my third day in town, I met my wife, Lynda. Since that day, my life has been much richer. She has supported me unconditionally in all of my efforts. Thank you. I would also like to thank my mom, dad, granny, and papa for always supporting and encouraging me in my academic endeavors, my sister Ashley for inspiration and the occasional ribbing, and Dr. Patrick Hymel, my best man and good friend.

The journey to computer science as an M.D./Ph.D. student is not a straight one. My decision to pursue both an M.D. and a Ph.D. was inspired by hours of conversation with Dr. Jeremy Jones, a friend whom I met in Dr. Patronis's electronics class in the Department of Physics at Georgia Tech. At the time, it seemed like the right thing for me. I wanted to be a scientist, and my interest in medicine was growing.

A competitive M.D./Ph.D. applicant needs research experience. For this, I had the privilege of working with Dr. Roger Wartell, a biophysicist and Chair of the Department of Biology at Georgia Tech. Roger had developed some computational models of DNA melting, and I used his models to create software that selects DNA fragments to detect mutations by denaturant gel electrophoresis. We co-authored a paper describing this work. Roger also wrote me an extremely nice letter of recommendation.

At UAB, my research career changed when Dr. John Smith gave a lecture on medical informatics sometime during my second year of medical school. By this time,

after almost two grueling years of medical school, I was yearning for an opportunity to return to physical or computational sciences with the hopes of finding a medically bent research project. Luckily, John made his lecture interesting and talked about the future of medical information systems. I approached him after class, and a couple of weeks later, he was taking me to lunch to meet Stephen Moser.

Dr. Stephen Moser, a faculty member in the Department of Pathology, was interested in medical information systems, and in particular clinical laboratory information systems. Steve, serving as one of my co-advisors, has supported my research both financially and intellectually for some time. He and I have collaborated on several posters and a couple of papers. I appreciate his support and advice, and owe him many thanks.

I was lucky enough to have two advisors. I met Dr. Warren Jones, Chair of the Department of Computer and Information Sciences at UAB, to talk to him about getting a Ph.D. in Computer Science. I explained to him my career to that point, and he welcomed me into the department with open arms. Shortly thereafter, he handed me a paper on data mining. I thank him for the opportunity and the support.

Throughout much of my work, I regularly sought the opinions Dr. Alan Sprague, a computer scientist and mathematician at UAB. His generosity, patience, and clarity of thought have guided me throughout. We worked together on the clone algorithm described in Chapter 2.

I would also like to thank committee members Dr. J. Michael Hardin for statistical wisdom at times of confusion, and guru Dr. Robert Hyatt for the use of his machine.

Finally, I would like to thank Dr. Frank Griffin, Director of the MSTP at UAB, for letting me venture off the beaten path.

For the past year, my work has been supported by individual medical informatics fellowship 1 F37 LM00057-01 from the National Library of Medicine.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

Page

CHAPTER

TABLE OF CONTENTS (Continued)

Page

APPENDIX

LIST OF TABLES

LIST OF TABLES (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

CHAPTER 1

INTRODUCTION

Data mining is the partially automated process of finding potentially interesting, previously unknown patterns in data. Like all new academic disciplines, it is at the same time something old and something new.

Since patterns are necessary for the construction of scientific hypotheses and causal models, the identification of meaningful patterns in data has always been an integral part of scientific discovery. Traditionally, however, scientists have relied on intuition and serendipity together with traditional data analysis to bring these patterns to the light of day. Automated pattern discovery was for many years only an elusive goal.

Within the last 15 years, however, computer-based methods for extracting patterns from data have been developed in the fields of statistics, economics, artificial intelligence, decision theory, and astronomy, amongst others. Until the early 1990s, though, the work of researchers in one field was largely unknown to those in others, despite the fact that the strategies they employed were often similar. Recently, the field of *data mining* has served to bring together these researchers to share their experiences and to form a new academic discipline.

Over the last quarter century, our ability to collect and store data has grown significantly faster than our ability to analyze data. According to current estimates, the amount of data stored in the databases of the world doubles every 20 months (Frawley,

Pietetsky-Shapiro, and Matheus 1992). As a result, there is a general consensus amongst experts that significant untapped knowledge lies hidden in many large databases. Therefore, an important factor behind the emergence and development of data mining has been the realization that today's databases, due to their size and complexity, contain knowledge that is not discovered by traditional methods.

Traditional analytical methods, in which the user formulates a query, then statistically compares the results of the query to a prior assumption, are largely confirmatory; they start with a null hypothesis, and they end with the null hypothesis being rejected or confirmed (not rejected). Consequently, if an existing pattern is not suspected (hypothesized), it will likely go undiscovered.

The challenge of data mining is to assist the user in uncovering deposits of interesting and unknown patterns whose discovery would otherwise require large amounts of time, resources, and luck using traditional hypothesis-driven methods. To accomplish this, data mining depends on exploratory analytical methods.

In data mining, as in statistics, exploratory methods search for interesting patterns by testing more than one hypothesis. These tests, called *multiple comparisons*, stray from formal assumptions of statistical inference, but are often useful for discovering new patterns (Tukey 1977). Elder and Pregibon (1996) have noted,

> With increasingly huge and amorphous databases, it is clear that methods for automatically hunting down possible patterns worthy of fuller, interactive attention are required. The existence of such tools can free one up to, for instance, posit a wider range of candidate data features and basis functions (building blocks) than one would wish to deal with, if one were specifying a model structure "by hand". (p. 96)

In data mining, exploratory methods are automated by data mining algorithms. The careless application of these algorithms, however, often results in an abundance of

spurious and uninteresting findings–*pattern glut*. Pattern glut is the product of "data dredging," a derogatory term used to conjure up images of junk from a polluted riverbed (Armitage 1971). While some uninteresting results are inevitable in all data mining projects, too many can make their evaluation cumbersome and unmanageable. The resulting problem, too many patterns, is hardly better than the original problem, too much data.

Well-planned data mining projects, however, successfully deal with pattern glut and yield a manageable number of results that can be reviewed in a timely and efficient manner. Therefore, a difference between data dredging and data mining is that data dredging produces too many uninteresting patterns, i.e., pattern glut, whereas data mining does not. For this reason, data mining is not a canned process that can be successfully applied to any database; it is an iterative and interactive process whose success depends on careful planning and refinement for its success.

In the remainder of this chapter, we outline the motivation of our research, define *data mining*, describe the data mining process, and give an overview of the Data Mining Surveillance System.

1.1 Motivation

Data mining, as a field, has been powered by the desire to find meaningful, new patterns in real-world databases. For example, retail data are mined to determine sales and inventory patterns (Anand and Kahn 1992); credit card data are mined for suspect fraudulent activity (Blanchard 1994); financial market data are mined to aid in the development of stock selection strategies (Hall, Mani, and Barr 1996; John 1997);

molecular sequence data are mined for structural motifs (Holfacker, Huynen, Stadler, and Stolorz 1996); satellite image data are mined for earthquake patterns (Shek, Muntz, Mesrobian, and Ng 1996); and even basketball statistics are mined to help identify key match-ups in upcoming games (IBM Advanced Scout 1995).

In health care, however, there have been relatively few data mining ventures. Matheus, Piatetsky-Shapiro and McNeill (1995) have developed the KEFIR system to identify possible cost saving measures based on deviations in pre-selected health outcomes, and several groups have used statistical and machine learning techniques to generate diagnostic and prognostic rules from clinical data sets (Prather et al. 1997; Tsai et al. 1997; Tsumoto and Tanaka, 1996).

In this dissertation, we address the use of data mining in epidemiologic surveillance. In particular, we focus on the use of data mining in infectious disease surveillance and antibiotic resistance surveillance both at the hospital and public health levels. As in many data mining research projects, our work on these specific applications has fueled the development of more generally applicable ideas, strategies, and methods.

This document describes our research, including the development of the Data Mining Surveillance System, DMSS. In the next section, we introduce the subject of epidemiologic surveillance to put in context the motivation behind much of our work.

1.2 Epidemiologic Surveillance

Epidemiology is the study of the occurrence of disease or other health outcomes (Rothman and Greenland 1997). *Epidemiologic surveillance* is the process by which changes in the occurrence of health outcomes are detected. Although some prefer the

phrase "public health surveillance" to "epidemiologic surveillance" (Thacker 1994), we prefer the latter because surveillance within a hospital, while epidemiologic, is generally not considered public health. We also realize that epidemiologic surveillance entails much more than data analysis (Teutsch and Churchill 1994). However, for the purposes of this document, "epidemiologic surveillance" refers only to the data analysis component of epidemiologic surveillance unless otherwise noted.

Like traditional data analysis, traditional epidemiologic surveillance is largely based on confirmatory, hypothesis-driven methods. Consequently, the systematic discovery of unknown patterns requires extensive time and resources, both of which few epidemiologists have.

Of course, surprise outbreaks of disease are sometimes recognized. Usually, however, these outbreaks are brought to the attention of epidemiologists by astute citizens and local physicians, not by systematic surveillance efforts (Buehler 1997; Kheifets 1993; Smith and Neutra 1993). In fact, state health departments spend a considerable amount of time investigating candidate disease outbreaks reported by concerned citizens (Smith and Neutra 1993).

Active surveillance involves systematically monitoring disease incidence data for interesting patterns. Typically, active surveillance is reserved for identifying specific pre-defined incidence patterns of known diseases. In this kind of analysis, which we call *traditional active surveillance*, the epidemiologist explicitly defines a case or outcome, then monitors a *surveillance group* for changes in the incidence of that outcome over time.

An *outcome* is a specific event or health indicator whose incidence is monitored within one or more surveillance groups. Outcomes can be complex entities. For example, an outcome may be a specific 3-drug resistance pattern in a group of bacterial isolates. If each member of the group is tested against 15 antibiotics, then there are 455 different 3-drug combinations, or outcomes, to which a group member may show resistance.

A *surveillance group* is a population that is monitored for changes in the incidence of an outcome. For example, a surveillance group may be a demographic population such as 18 to 34 year-old, Asian men in San Francisco, or it may be a population of medical cases such as bacterial isolates from ventilated patients in the surgical intensive care unit. Since surveillance groups can be described by a number of attributes (e.g. residence, gender, race, ethnicity, age, occupation, education, and health status for human populations), some of which may themselves contain a number of sub-attributes, all surveillance groups for a given outcome occupy a high-dimensional *group space*.

In traditional active surveillance efforts, epidemiologists simply do not have the time or the resources to search for interesting changes in the incidence of complex outcomes in high-dimensional groups. Often, only simple outcomes over a few low-dimensional groups can be considered. For example, the incidence of a specific disease may be monitored nationally, or by race, or a combination of two attributes such as region and age group, but rarely, however, would the disease be monitored in groups that are described by more than two or three variables. As a result, we claim that important, subtle, and high-dimensional patterns are often missed. Enter data mining.

The need for data mining in public health surveillance has not gone unnoticed. In a chapter on computerized public health surveillance systems in *Principles and Practice of Public Health Surveillance*, Dean, Fagan, and Panter-Connah (1994) describe an ideal public health surveillance system. In doing so, they give a hypothetical example in which the user, an epidemiologist, uses the ideal system to compare data recently collected with similar data collected in the past. Specifying few constraints, the user asks the system to produce a series of maps for all conditions with unusual patterns. Identifying those patterns that are most interesting, the user then employs traditional epidemiologic methods to investigate them further. The system merely suggests potentially interesting patterns. Whether or not they are acted on depends on the judgement of the user.

In addition to the political and administrative barriers that the authors identify as obstacles to practically realizing such a system, they correctly identify the following challenge: "Several kinds of mental shifts, as well as corresponding technical developments, will be necessary before a computerized system can be used to examine automatically a 'time slice' of disease and injury records that originate in clinics and hospitals" (p. 203). It is this challenge that we address.

1.3 Data Mining

Extensive reviews of data mining have been given by Fayyad, Pietetsky-Shapiro and Smyth (1996) and John (1997). In this section, we define some basic terms and briefly review the data mining process.

John (1997) provides a thorough, well-written, and timely overview of data mining. Sharing his sentiment, we consider "data mining" synonymous with "knowledge

discovery in databases," and use "data mining" since it is the simpler of the two phrases and it is the more recognized outside of academic circles.

Fayyad et al. (1996) defines *data mining* as the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data, and John (1997) defines it as the process of discovering advantageous patterns in data. Since the use of automated methods distinguishes data mining from more traditional methods of data analysis, we define *data mining* as the partially automated process of finding potentially interesting, previously unknown patterns in data.

A *pattern*, as defined by John (1997), is a parsimonious statement about a probability distribution. While we generally agree with this definition, a pattern may not always have a probabilistic interpretation (Section 4.1.1). For this reason, we define *pattern* as a description of a configuration of elements that is simpler than the enumeration of those elements. Patterns are useful in constructing causal models and, therefore, play an important role in knowledge discovery.

1.3.1 The Data Mining Process

Data mining is an iterative, interactive, and involved process. While the exact structure of the process differs from one description to another, it consists of the following basic steps (John 1997):

1. Understand the problem.
2. Extract the data.
3. Clean / engineer the data.
4. Engineer a data mining algorithm.

5. Search for potentially interesting patterns by running the data mining algorithm.
6. Evaluate the patterns.

Importantly, results from each step of the process are subject to critical review by domain experts. This review often yields additional insight into the problem that can be incorporated into the data mining process. Consequently, iterating through the data mining process is usually required to make the entire process more efficient and effective. Due to this user-interaction, data mining is only a partially automated process. Furthermore, it seems unlikely that it will ever become fully automated: after all, whether or not a pattern is interesting is ultimately determined by the user.

1.4 The Data Mining Surveillance System (DMSS)

The primary goal of our research is to use data mining techniques to identify new, unexpected, and interesting temporal patterns in epidemiologic surveillance data. To this end, we constructed the Data Mining Surveillance System (DMSS). Unlike traditional surveillance systems, DMSS is not constrained to looking for changes in the incidence of simple outcomes in low-dimensional, pre-specified groups.

For the past 3 years, DMSS has been the test bed for many ideas, some of which are described in this document. Before launching into these ideas, however, we need to define several terms, describe how association rules can be used in epidemiologic surveillance, and introduce the UAB data set.

1.4.1 Definitions

An *itemset* is a subset of the set of all items. The *support* of an itemset x, sup(x), is the number of records in a data set that contain x. If sup(x) $\geq$ T, where T is the frequent set support threshold (FSST), then x is a *frequent set*. The itemset that contains no items, *EmptySet*, has support N, the number of records in the data set.

The following example illustrates the concept of *frequent set*. In a super market database, where each record contains the names of the items in a basket at the checkout, the itemset {bread, milk, cheese} is likely to be to contained in many records from a single day because bread, milk, and cheese are frequently purchased together. If {bread, milk, cheese} is a frequent set, then so too are {bread}, {milk}, {cheese}, {bread, milk}, {milk, cheese}, and {bread, cheese} since the support of each of these itemsets must be at least that of the frequent set {bread, milk, cheese}.

An *association rule*, $A \Rightarrow B$, where A and B are frequent sets and $A \cap B = \varnothing$, is a statement about how often the items of B are found with the items of A. The *incidence proportion* of $A \Rightarrow B$, denoted ip($A \Rightarrow B$), is equal to sup($A \cup B$)/sup(A). The incidence proportion is the numerator and the denominator, whereas the *confidence* is their quotient. For example, in the supermarket setting,

ip({milk, cheese}$\Rightarrow$ {bread}) = sup({milk, cheese, bread})/sup({milk, cheese})

is the incidence of bread in baskets with milk and cheese. The incidence proportion of *EmptySet* $\Rightarrow$ {Bread} is sup({bread})/N, the incidence of bread in all baskets.

The *precondition support* of association rule $A \Rightarrow B$ is sup(A). Association rules that have relatively high precondition support are often more meaningful than rules with relatively low precondition support because the former are statements about the incidence

of *B* in non-trivial groups *A*. The precondition support is often referred to as *denominator data* in epidemiology.

Traditional data mining applications based on association rules focus on discovering high-confidence association rules because these rules describe high-probability events (Piatetsky-Shapiro 1991). For example, a high-confidence rule that says that young men who purchase items a, b, and c also purchase item d 65% of the time, could be used in designing a marketing strategy which places all 4 items contiguously on a shelf. Such a strategy may increase sales of one or more of the items.

While high-confidence rules are useful in epidemiologic surveillance, low-probability rules are often more useful. To understand why this is the case, we first need to describe how association rules can be used in epidemiologic surveillance.

1.4.2 Association Rules in Epidemiologic Surveillance

Association rules are well suited for epidemiologic surveillance because they naturally describe the incidence of an outcome within a group. Let us look at a couple of examples (Table 1). Rule 1 of Table 1 describes the incidence of *Streptococcus pneumoniae* infection in HIV-positive, white, 18-24 year-old, California women. Rule 2 of Table 1 describes the incidence of piperacillin and ticarcillin resistance in nosocomial (hospital acquired), non-*Pseudomonas* gram-negative rod (NPGNR) isolates from the surgical intensive care unit (SICU). In general, the left-hand side (LHS) of an association rule is the surveillance group and the right-hand side (RHS) of an association rule is the outcome.

Table 1: Example Association Rules from Epidemiologic Surveillance.

	LHS	RHS
1.	{HIV+, white, 18-24, California, female}	$\Rightarrow$ {*S. pneumoniae*}
2.	{nosocomial, NP_GNR, SICU}	$\Rightarrow$ {R~piperacillin, R~ticarcillin}

The incidence proportion of an association rule $A \Rightarrow B$ in data partition p_i describes the incidence of the outcome, B, in the group, A, during t_i. For example, in a data set that contains records for hospital bacterial infections from the month January, let us say 24 describe nosocomial, NP_GNR from the SICU. If 4 of the 24 describe isolates that are resistant to piperacillin and ticarcillin, then the incidence proportion of association rule 2 of Table 1 for January is 4/24.

Since an incidence proportion of an association rule $A \Rightarrow B$ in partition p_i describes the incidence of B in A in t_i, a series of incidence proportions for $A \Rightarrow B$ from partitions $p_1, p_2, \ldots, p_n$ describes the incidence of the outcome B in group A from t_1 through t_n. Therefore, by analyzing the time-series of incidence proportions of an association rule $A \Rightarrow B$, it should be possible to detect important shifts or trends in the incidence of B in A over time. In this way, epidemiologic surveillance of B in A is possible.

The reason why low-confidence association rules are often more interesting than high-confidence ones is simple: if B occurs every time A occurs, and A occurs frequently, then the rule $A \Rightarrow B$ is probably known or trivial and, therefore, uninteresting. However, if B occurs infrequently with A and A occurs frequently, then $A \Rightarrow B$ is a low-confidence

rule and changes in ip($A \Rightarrow B$) are likely to go undetected by traditional methods. This is especially true if either the group or the outcome is complex.

1.4.3 The UAB Data Set

To facilitate an overview of DMSS and descriptions of ideas in upcoming chapters, we find it helpful to refer to the infection control / antibiotic resistance data set from the University of Alabama at Birmingham (UAB) Hospital. For short, we call this the *UAB data set*. We describe the UAB data set here, and give a full description of its analysis in Chapter 5.

1.4.3.1 Data Description and Extraction

Fifteen months (September 1996 to November 1997) of bacterial antimicrobial susceptibility results and related patient information were extracted from the UAB clinical laboratory information system. Each record describes a single bacterial isolate and contains items for the following attributes: organism name, gram stain/morphology, date collected, nosocomial status, source of isolate (e.g., sputum, blood, urine), location of patient in hospital (e.g., Surgical Intensive Care Unit, Medical Intensive Care Unit), and test results Resistant (R~), Intermediate resistance (I~), or Susceptible (S~), according to NCCLS criteria (NCCLS 1997), for each member of a set of antimicrobials.

The gram stain/morphology attribute is "GPC" for gram-positive cocci, "NP-_GNR" for non-*Pseudomonas* gram-negative rod, and empty for *Pseudomonas*. The nosocomial status attribute is either "nosocomial" or "community," depending on when the sample was obtained from the patient. If an isolate is from a sample collected on or

after the patient's third day in the hospital, then the isolate was likely acquired in the hospital. Such isolates are classified as "nosocomial." If the isolate is from a sample obtained on the first or second day of the patient's stay, the isolate is classified as "community."

To demonstrate the ability of DMSS to identify outbreaks of resistant organisms, we seeded the data set with records describing a nosocomial outbreak of highly resistant *Acinetobacter baumanii* that occurred in the UAB hospital in 1994. This was done by removing all nosocomial *Acinetobacter baumanii* records from the corresponding months of the 1997 data set, and replacing them with the outbreak *Acinetobacter baumanii* records from the same months in 1994.

1.4.3.2 Data Cleaning and Partitioning

Duplicate records were removed so that the data set contains no more than one record per patient per organism per month. Additionally, for each record, results (S, I, R) for antimicrobials to which the organism historically tests resistant ($\geq 50\%$) are removed. Consequently, of R/S/I~*Antimicrobial* items that remain, most are of the S~*Antimicrobial* type. Then S~*Antimicrobial* and I~*Antimicrobial* items are removed from each record so that only R~*Antimicrobial* items remain. These steps significantly reduce the number of frequent items in the data set. By reducing the number of frequent items, we reduce the computational burden of generating frequent sets.

Finally, all records are split into disjoint monthly partitions and the "date collected" attribute is removed from each.

nosocomial, *Klebsiella pneumoniae*, NP-GNR, SICU, urine, R~piperacillin.
non-nosocomial, *Morganella morganii*, NP-GNR, VNAP, urine, R~cefuroxime.
nosocomial, *Enterobacter cloacae*, NP-GNR, MICU, tracheal aspirate, R~ceftazidime, R~ceftriaxone, R~piperacillin.

Figure 1: Example records from a monthly partition of the UAB data set.

1.4.4 DMSS Data Considerations

DMSS requires that each partition be a text file composed of records where each record is a set of items. Records may contain different numbers of items, but each item must be categorical.

Items may be from a taxonomy, but DMSS currently has no mechanism for explicitly using a taxonomy. Instead, relevant portions of the taxonomy should be expanded in-line. For example, in the UAB data set, *NP_GNR* is a class of bacteria that contains the items *Klebsiella pneumoniae*, *Morganella morganii*, *Enterobacter cloacae*, and others.

To expand the class *NP_GNR* in-line, the item "NP_GNR" was inserted into all records that contain a member of the class *NP_GNR*. The resulting records are like those in Figure 1. Likewise, for records that contain members of the class *GPC*, the item "GPC" is inserted. Of course, portions of the taxonomy to be expanded must be specified by a domain expert. This is yet another example of data mining's dependence on the user.

Expanding parts of a taxonomy in-line leads to *redundant frequent sets* (Section 2.2). For example, if every record that contains a member of the class *NP_GNR* is

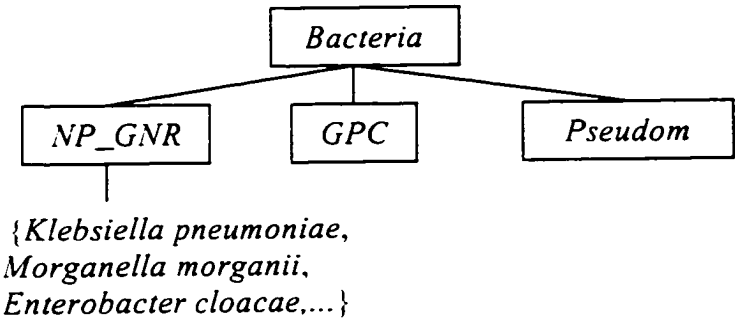


Figure 2: A partial taxonomy on bacteria.

expanded to include the item "NP-GNR," then pairs of frequent sets are generated such that for each pair, both frequent sets have the same support and differ by one item, namely, "NP-GNR." For example, if {*Enterobacter cloacae*, SICU, tracheal aspirate} is a frequent set with support 5, then {*Enterobacter cloacae*, NP_GNR, SICU, tracheal aspirate} is also a frequent set with support 5 since "NP_GNR" always appears with *Enterobacter cloacae*. Redundancies from in-line taxonomy expansions such as this are trivial and can be eliminated by not counting an itemset that contains both an item and its ancestor (Srikant, Vu, and Agrawal 1997). The *clone algorithm* presented in Chapter 2 handles these types of taxonomy redundancies as a simple cases of clonal behavior.

1.4.5 An Overview of DMSS

A general diagram of DMSS is given in Figure 3. As a data mining system, DMSS embodies many of the steps of the data mining process. Data analysis starts by extracting relevant data from a database, cleaning it, modifying it, and dividing it into disjoint partitions. Partitions are then processed one at a time by the procedure outlined in Figure 4. Usually, a search for potentially interesting patterns is conducted after ea

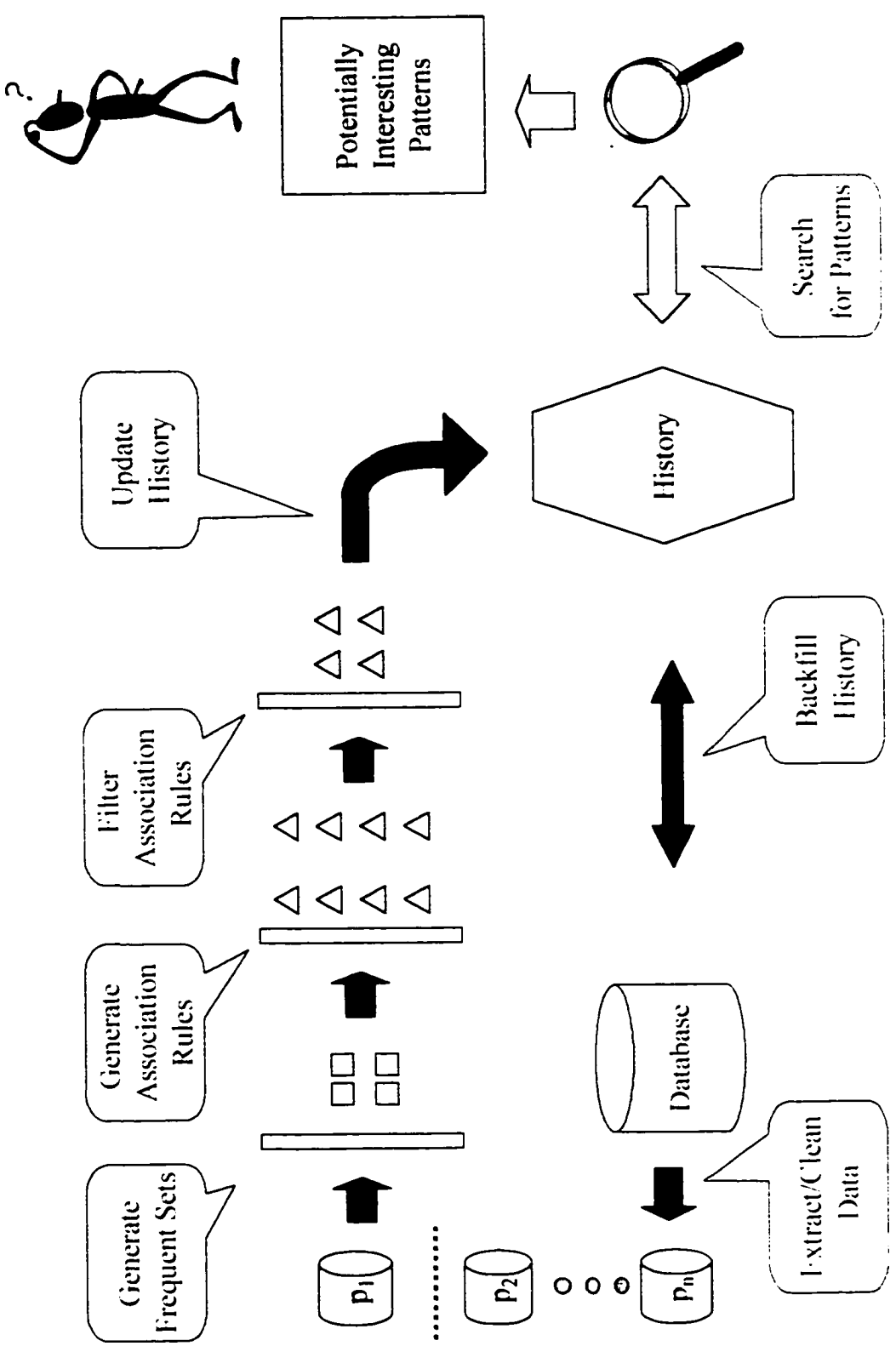


Figure 3: Overview of IDMSS

1) Use the *clone algorithm* /*Chapter 2*/ to generate the set *FS* of all frequent sets in p_i.
2) Use *FS* to generate all association rules that have minimum precondition support. /* Section 3.1 */
3) For each $r \in AR$:
4) if (*r* passes a user-defined set of *association rule templates*) /* Section 3.2 */
5) /* Update the history with *r*. Section 3.3 */
6) If *r* is new to the history, add *r* to the history and query the database to get the incidence proportions of *r* for prior partitions. Update the history with these incidence proportions.
7) Update the history with the incidence proportion of *r* in p_i.

Figure 4: Procedure for processing a data partition.

partition is processed. This simulates the manner in which patterns are generated in real-time surveillance studies where data is collected continuously; as soon as the most recent partition is complete, it is processed and a search for patterns is conducted (Figure 3).

DMSS searches for patterns by looking for changes in the incidence proportion time-series of each association rule in the history. Significant changes in these time-series are called *alerts* (Section 4.2). Due to the hierarchical and sometimes redundant nature of alerts, DMSS uses alerts to construct *events* (Section 4.6.2). Events, in turn, are presented to the user as potentially interesting patterns. The process of constructing events from alerts reduces the number of patterns the user is required to evaluate, thereby reducing pattern glut.

1.5 Related Work

DMSS is an example of the *active data mining* paradigm presented by Agrawal and Psaila (1995). It differs from this paradigm, however, in several important ways. First, it tracks low-confidence association rules instead of high-confidence rules. This introduced new challenges. First, there are many more low-support frequent sets than there are high-support ones. This led us to identifying clonal frequent sets along with a new frequent set discovery algorithm that recognizes them (Chapter 2). Second, there are many more low-confidence association rules than there are high-confidence ones. In order to deal with all of these rules, we use association rule templates (Section 3.2) to classify certain "flavors" of them as interesting and other "flavors" as uninteresting. Rule templates are indispensable in reducing pattern glut. We also employ windowing schedules and statistically based methods (Chapter 4) instead of shape queries for detecting patterns, and develop the concept of event clusters (Section 4.6) to reduce the number of redundant patterns presented to the user.

As far as we know, DMSS is the only example of an active data mining system other than the one briefly described in Agrawal and Psaila (1995). Even there, only an incomplete description of a system was provided and no experimental results were given.

Frequent set discovery algorithms are usually discussed in the context of generating association rules. Association rules were introduced with frequent sets by Agrawal and Swami (Agrawal, Imielinski, and Swami 1993). Since then, frequent set discovery algorithms have been the subjects of many papers in the data mining literature including Agrawal and Srikant (1994), Savasare, Omiecinski and Navathe (1995), and Brin, Motwani, Ullman and Tsur (1997). The *clone algorithm*, presented in Chapter 2, is

a fundamentally new frequent set discovery algorithm that was developed from our experiences with antibiotic resistance data.

Association rule templates were introduced by Klemettinen et al. (1994) as a way to allow the user to describe forms of potentially interesting and uninteresting rules. Rule templates are extremely useful in DMSS for reducing the number of rules stored in the history and for reducing pattern glut (Section 3.2). Without them, data mining in epidemiologic surveillance is extremely cumbersome and practically impossible.

In epidemiology, a number of statistical strategies have been used for detecting disease clusters and outbreaks in surveillance data. These include the scan statistic (Naus 1966; Wallenstein 1980), the MAX statistic (Ederer, Meyers, and Mantel 1964; Grimson 1993), cumulative sums (Hutwagner et al. 1997) and log-linear regression (Farrington, Andrews, and Beale 1996). The MAX statistic was considered for use in DMSS, but was eventually excluded in favor of tests of two-proportions (Section 4.5).

In data mining, several strategies have been proposed for detecting deviations in time-series data. The shape queries of Agrawal and Psaila (1995) and the dynamic programming strategies developed by Berndt and Clifford (1996) are two examples. These methods do not have current implications for DMSS, but are included here for completeness.

1.6 Preview of Upcoming Chapters

In the remainder of the dissertation, we describe in detail specific components of DMSS and present experimental results obtained by using DMSS on real epidemiologic

data sets. Throughout, contributions that have broad implications for data mining and epidemiologic surveillance are emphasized.

Chapter 2 contains a description of the *clone algorithm* for discovering frequent sets and includes results that illustrate its advantage over other frequent set discovery algorithms for certain types of data.

Chapter 3 describes processes for generating association rules from frequent sets and for updating the history. This includes a description of association rule templates and their role in DMSS.

Chapter 4 contains a discussion of the role of statistical significance testing and multiple comparisons in data mining, and a description of how DMSS identifies potentially interesting patterns in time-series data. Events and event clusters are also described here.

Chapter 5 presents the complete analysis of the UAB data and the CDC data sets, and Chapter 6 contains final comments and possibilities for future research.

CHAPTER 2

CLONAL FREQUENT SETS AND THE CLONE ALGORITHM

Clonal frequent sets appear in real-world surveillance data and can severely tax known frequent set discovery algorithms. In this chapter, we define clonal frequent sets, describe real situations in which they arise, present a new frequent set algorithm that recognizes them, and give experimental results of our new algorithm on real data from a medical surveillance application.

2.1 Introduction

Frequent set discovery algorithms are extensively discussed in the data mining literature (Agrawal and Srikant 1994; Brin et al. 1997; Savasare et al. 1995). The problem of discovering frequent sets from data, first defined by Agrawal et al. (1993), is usually framed in the context of generating association rules from market-basket data. Market-basket data analysis, however, is not the only use for association rules (Brin et al. 1997). We employ association rules in analyzing infection control surveillance data (Brossette et al. 1998) which has different characteristics than market-basket data. It is in this surveillance context that clonal frequent sets arise and the inefficiency of traditional frequent set algorithms to handle them becomes apparent.

Clonal frequent sets get their name from our experience with records describing clonal bacterial isolates in infection control surveillance data. In this data, each record

describes a single bacterial isolate from a patient and includes the name of the organism isolated along with the names of the antibiotics that organism tested resistant to in the laboratory. If one bacterial clone is responsible for several isolates, a cluster, each isolate in the cluster will test resistant to the same antibiotics, and each corresponding record will contain identical lists of antibiotic names. If such a clonal cluster is resistant to many antibiotics, an occurrence that is unfortunately becoming more common, then each record in the cluster will contain the same long list of antibiotic names. This antibiotic name list alone can be 10 to 20 items long. If the clonal cluster is limited to a specific location or patient population, the number of identical items in the corresponding records grows even further.

Here is a short illustrative example. The data set shown in Figure 5 contains only 4 records. Records 2 and 3 describe a clonal cluster of Organism1 in Location2 that is

1) Org1, Loc1, Antb1, Antb3.
2) Org1, Loc2, Antb1, Antb2, ..., Antb12.
3) Org1, Loc2, Antb1, Antb2, ..., Antb12.
4) Org2, Loc2, Antb3, Antb4.

Figure 5: Example clonal data set.

resistant to antibiotics one through twelve. Record 1 describes an isolate of Organism1 that is not part of the clonal cluster just described. Record 4 describes an isolate of Organism2. Now we want to generate frequent sets from this data. With an absolute frequent set support threshold of 2, there will be many redundant frequent sets generated. For example, the frequent sets {Org1, Loc2} and {Org1, Loc2, Antb2} are redundant because all records that contain Org1 and Loc2 also contain Antb2. In fact, with a

traditional frequent set algorithm about 2^{11} redundant frequent sets and only 4 non-redundant frequent sets will be generated from this data. Those 4 non-redundant frequent sets are shown in Table 2. Of them, frequent sets 2, 3 and 4 are clonal frequent sets.

The key to not generating redundant frequent sets is to be able to recognize clonal frequent sets during processing. Our new frequent set algorithm does this and, in the case where significant clonal activity exists in a data set, offers significant improvement over traditional algorithms.

Table 2: Non-Redundant Frequent Sets for the Data Set in Figure 5.

frequent set:	support:
1. {Antb3}	4
2. {Loc2, Antb3, Antb4}	3
3. {Org1, Antb1, Antb3}	3
4. {Org1, Loc2, Antb1, Antb2, ... , Antb12}	2

2.2 Definitions

Given the set of all items I, an *itemset* is a subset of I, and a *k-itemset* is an itemset that contains exactly k items. A record t, itself a subset of I, is said to contain an itemset x if x is a subset of t. The *support* of itemset x, sup(x), is the number of records that contain x, or the fraction of records that contain x. If sup(x) $s \geq T$, where T is the frequent set support threshold, then x is a *frequent set.*

Let x be an itemset and $r_1, \ldots, r_n$ be the records containing x. The *closure* of x, closure(x), is the set of all items appearing in each of the records $r_1, \ldots, r_n$. Alternatively, closure(x) is the intersection of records $r_1, \ldots, r_n$. For instance, in the example data set in

Figure 5, closure({Loc2, Antb4}) is {Loc2, Antb3, Antb4} since the intersection of records 2, 3, and 4 is exactly {Loc2, Antb3, Antb4}.

We say that two itemsets are *equivalent* if they have the same closure. Any non-trivial equivalence class, which is an equivalence class that contains more than one itemset, is called a *clone*. A *clonal itemset* is an itemset that belongs to a clone. Each clone contains a maximal itemset, namely the union of all itemsets in the clone (which is the same as the closure of any itemset in the clone). This maximal itemset is called *max-clone*. If a non-trivial equivalence class F contains a k-item max-clone and a j-item itemset(s) but not an i-item itemset, $i < j$, then we call F a *(j,k)-clone*.

For each equivalence class, all itemsets in the class have the same support. An equivalence class is *frequent* if any itemset in the class is frequent. Our intention is to generate for each frequent equivalence class, a representative for that class. The intention of other frequent set algorithms is to generate every member of all frequent equivalence classes. For each non-trivial frequent equivalence class, i.e., clone, we want to generate its max-clone as a representative. We consider all other frequent sets in the clone *redundant*.

A collection S of itemsets *covers* all frequent k-itemsets if for each frequent k-itemset s $\exists s' \in S$ such that $s \subseteq s'$ and closure(s) = closure(s'). A *k-cover* is a collection of frequent sets that covers all k-item frequent sets.

2.3 The Clone Algorithm

The structure of the *clone algorithm* (Figure 6) is similar to that of Apriori (Agrawal and Srikant 1994) and other Apriori-like frequent set algorithms. The clone

1) k = 1.
2) N_1 = all frequent 1-itemsets.
3) $S_1 = \varnothing$.
4) do
5) k = k + 1.
6) Using N_{k-1}, generate k-item frequent sets N_k.
7) Using N_{k-1} and S_{k-1}, generate special frequent sets $S_k^{(1)}$.
8) Using S_{k-1}, generate special frequent sets $S_k^{(2)}$.
9) Construct bipartite graph G on vertex set $(N_{k-1} \cup S_{k-1}) \cup (N_k \cup S_k^{(1)} \cup S_k^{(2)})$.
10) Construct $S_k^{(3)}$ from the nontrivial connected components of G.
11) Modify N_k by removing from it frequent sets belonging to nontrivial connected components of G.
12) $S_k = S_k^{(1)} \cup S_k^{(2)} \cup S_k^{(3)} \cup S_{k-1}$.
13) while $(N_k \cup S_k^{(1)} \cup S_k^{(2)} \neq \varnothing)$
14) Answer = $\bigcup_k (N_k \cup S_k)$.

Algorithm for step 6: Using N_{k-1}, generate k-item frequent sets N_k.
1) $N_k = \varnothing$.
2) For each $s \in N_{k-1}$:
3) For each $t \in N_{k-1}$ having the same (k-2)-item prefix as s:
4) If $s \cup t$ is frequent, adjoin it to N_k.

Algorithm for step 7: Using N_{k-1} and S_{k-1}, generate special frequent sets $S_k^{(1)}$.
1) $S_k^{(1)} = \varnothing$.
2) For each $s \in S_{k-1}$:
3) For each $t \in N_{k-1}$ such that $|t - s| = 1$:
4) If $s \cup t$ is frequent, adjoin it to $S_k^{(1)}$.

Step 8 is similar to steps 6 and 7: for each pair of special frequent sets s and t in S_{k-1}, if $s \cup t$ is frequent, adjoin it to $S_k^{(2)}$.

Figure 6: The clone algorithm.

algorithm is different from existing algorithms, however, because, instead of generating all k-item frequent sets from (k-1)-item frequent sets, the clone algorithm generates a cover for k-item frequent sets from a cover of (k-1)-item frequent sets. Specifically, the k-cover consists of the clones that have already been discovered (called *special* frequent sets) together with k-item frequent sets, called *normal* frequent sets, that are not covered

by the clones. In Figure 6, the special frequent sets generated in iteration k are denoted S_k, and the normal frequent sets generated in iteration k are denoted N_k.

In steps 9-11 of the algorithm, clones are identified by the analysis of a bipartite graph on the cover of all (k-1)-item frequent sets and the cover of k-item frequent sets.

Each frequent set , no matter whether generated in steps 6 or 7 or 8, is generated from a pair of frequent sets from N_{k-1} and S_{k-1}. These two members are called *parents* of the newly generated frequent set.

In graph G, frequent sets $s \in N_{k-1} \cup S_{k-1}$ and $t \in N_k \cup S_k^{(1)} \cup S_k^{(2)}$ are joined by an edge iff:

1. s is a parent of t and,
2. s and t have the same support.

Note that since every item in s is also an item in t, s and t have the same support iff closure(s) = closure(t). A component is called *trivial* if it contains a single member of $N_k \cup S_k^{(1)} \cup S_k^{(2)}$, even if it contains several members of $N_{k-1} \cup S_{k-1}$. It is clear that all frequent sets of a component are equivalent. For each nontrivial component, the union of the itemsets is computed; this is a *special* frequent set and is adjoined to $S_k^{(3)}$. Correspondingly, each frequent set in the nontrivial component is removed from N_k.

We have not been able to demonstrate the necessity of step 8 of the algorithm with either real or concocted data, and suspect that it is not necessary. However, since we have not been able to prove this conjecture, it remains.

The idea behind the clone algorithm is to recognize clonal frequent sets as early as possible in frequent set generation so that most of their sub-clones do not have to be generated. Therefore, N_k, k>1, may be substantially smaller when generated by clone

algorithm than it is when generated by traditional frequent set algorithms. In the next section, we illustrate the efficiency of the algorithm on real data sets.

2.4 Experimental Results

In this section we present experimental results from applying the clone algorithm and our Apriori-like (Alike) algorithm to hospital infection control surveillance data. This data describes bacterial isolates collected from patients. Each record includes the location of the patient, the name of the organism isolated, the source of the isolate, and the names of the antibiotics that the organism tested resistant to in the laboratory. The number of items per record, therefore, depends on the number of antibiotics to which the organism was resistant. Records that describe highly resistant organisms can contain 10-20 antibiotic names.

Results from experiments with five data sets are given in Table 3. All experiments were conducted on an AMD K6-233 with 128MB RAM running Linux 2.0.30. In all experiments, an absolute frequent set support threshold of 3 was used.

Each of the data sets contains substantial clonal behavior as illustrated by the difference between the number of frequent sets generated by Alike and the size of the cover generated by the clone algorithm. In the experiments, Alike generated 15 to 100 times as many frequent sets as the clone algorithm, and was 3 times faster to twice as slow as the clone algorithm.

The largest differences in the performance of the two algorithms can be seen in the results from data sets 3, 4, and 5. Each of these data sets contain records describing

Table 3: Results from Experiments with the Clone Algorithm.

		Apriori-like (Alike)		The clone algorithm	
data set		# frequent sets	time (sec)	# frequent sets in cover	time (sec)
1	834	32,813	48	2,215	122
2	742	20,548	31	1,829	78
3	788	>290,000	>150	2,029	103
4	772	90,548	51	1,947	74
5	728	279,271	117	1,942	65

clonal outbreaks of a highly resistant organism. For example, data set 3 contains a (2,20)-clone, a (3,18)-clone, and a (4,20)-clone amongst others. For these 3 data sets, Alike generated more than 110 times the number of frequent sets than the clone algorithm and was more than 30% slower.

In other experiments, we modified the clone algorithm so that small clones, i.e., (j,k)-clones where k-j < m, were ignored. The results of these experiments for m = 5, 8 are given in Table 4.

With m = 5, the modified clone algorithm generated more than 4 times the number of frequent sets than the clone algorithm in about two-thirds the time. For m=8, the modified clone algorithm generated about nine times the number of frequent sets as the clone algorithm in about two-thirds the time. Due to the large differences in the number of the frequent sets generated, for these data, the clone algorithm is prefered over the modified algorithm.

On data that contains no clones, the clone algorithm runs in about the same amount of time as Apriori.

Table 4: Results from Experiments with the Modified Clone Algorithm.

	$m = 5$		$m = 8$	
data set	# frequent sets in cover	time (sec)	# frequent sets in cover	time (sec)
1	10,786	82	22,769	84
2	6,312	47	13,756	44
3	13,326	68	22,190	55
4	7,020	47	16,387	60
5	7,175	34	12,575	39

2.5 Discussion

The clone algorithm is more efficient than Apriori on data that contains substantial clonal behavior, i.e., on data that contains clones of many frequent sets. In this case, the clone algorithm avoids generating many redundant frequent sets that Apriori is forced to generate. When the data contains clones whose max-clones contain only a few more items than the smallest clonal frequent set(s) in the clone, the space savings offered by the clone algorithm over Apriori will probably be outweighed by the cost incurred by the clone algorithm in crossing N_{k-1} with S_{k-1} and S_{k-1} and S_{k-1}. The exact number and/or size of clones that are necessary for the clone algorithm to outperform Apriori in time is not known. This is an area for further research. We see from the results of experiments on real data, however, that the clone algorithm can be faster than Apriori in cases where Apriori generated about 100 times as many frequent sets.

One significant potential improvement to the clone algorithm would be to have it estimate after the graph analysis step, the time and space savings of constructing each

clonal frequent set in $S_k^{(3)}$ that has connected frequent sets in N_k. The alternative to constructing the frequent set is to not construct it and not remove the components from N_k. This estimate may then be used to help ensure that if an $S_k^{(3)}$ frequent set is generated, the time and/or space savings would justify the effort. Results from experiments with the modified clone algorithm (Table 4) were given to provide some insight into this problem. We see that ignoring small clones, in the case of our data, is not very useful. It will be interesting to see if this result applies to other data.

It should be noted that clonal frequent sets are unlikely to occur in market-basket data because their presence implies the existence of association rules with confidence one. In market-basket data, such rules are extremely rare or non-existent. However, clonal frequent sets appear regularly in infection control surveillance data, and we suspect they also appear in other surveillance contexts.

2.6 Conclusion

We have defined clones and clonal sets and have introduced a new frequent set discovery algorithm, the clone algorithm, that can offer substantial time and space savings over traditional frequent set algorithms when applied to data with clonal behavior.

CHAPTER 3

ASSOCIATION RULES AND THE HISTORY

Association rules were introduced by Agrawal, Imielinski, and Swami (1993), extended by Agrawal at al. (1996), and briefly discussed in Section 1.4.1. An *association rule* is an expression of the form $A \Rightarrow B$, where A and B are frequent sets and $A \cap B = \varnothing$. The association rule $X \Rightarrow Y$ is a statement about how often the items of Y are found with the items of X.

Since generating association rules from frequent sets is less costly than generating frequent sets themselves, efficient frequent set discovery algorithms are necessary for efficient association rule generation. In Chapter 2, we described the clone algorithm for discovering frequent sets and showed that it is more efficient than other frequent set algorithms for certain types of data.

In this chapter, we review a basic algorithm for generating association rules from frequent sets, and describe a modified version of the algorithm that generates only those with high precondition support. Then we define association rule templates, describe their role in DMSS, and show how they are essential in decreasing pattern glut. In the last part of the chapter, we describe the *history* and how it is maintained.

3.1 Generating Association Rules

The association rule generation algorithm described by Agrawal et al. (1996) is shown in Figure 7. This algorithm generates high-confidence ($\geq$ *minconf*) association

1) for each k-item frequent set $L = \{a_1, \ldots, a_k\}$: // L is ordered s.t. $a_i < a_j$ for all i<j
2) $H_1 = \{ L\text{-}a_1 \Rightarrow a_1, L\text{-}a_2 \Rightarrow a_2, \ldots, L\text{-}a_k \Rightarrow a_k\}$ // all rules w/ 1 item on RHS
3) for (i=1 to k-1):
4) for each $h \in H_i$:
5) if (ip(h) < *minconf*)
6) Remove h from H_i
7) Use H_i to generate H_{i+1}
8) *Answer* = *Answer* $\cup \bigcup_{i=1}^{k-1} H_i$

Algorithm for step 7:
a) for each $h \in H_i$:
b) for each $l \in H_i$, $l \neq h$:
c) if (h.RHS and l.RHS share the first i-1 items)
d) *newRule*.LHS = h.LHS $\cap$ l.LHS
e) *newRule*.RHS = h.RHS $\cup$ l.RHS
f) Add *newRule* to H_{i+1}

Figure 7: An algorithm for generating high-confidence association rules.

rules from frequent sets. The algorithm utilizes the fact that if an association rule {a, b}⇒{c, d} has high-confidence, then so do the rules {a, b, c}⇒{d} and {a, b, d}⇒{c}. In other words, if either {a, b, c}⇒{d} or {a, b, d}⇒{c} is low confidence, then so is {a, b}⇒{c, d} because sup({a, b, c}) ≤ sup({a, b}) and sup({a, b, d}) ≤ sup({a, b}). As described in Section 1.4.1, the confidence, or incidence proportion of association rule $A \Rightarrow B$, is ip($A \Rightarrow B$) = sup($A \cup B$)/sup(A).

For epidemiologic surveillance, we want to generate rules that have high precondition support regardless confidence (Section 1.4.2). To do this, we need to slightly modify algorithm in Figure 7. The modified algorithm (Figure 8) uses the fact that if {d}⇒{a, b, c} is low support, then so is {c, d} ⇒ {a, b} because $sup(\{c,d\}) \leq sup(\{d\})$. In the modified algorithm, only lines 2, 5, C, D, and E are different from the algorithm in Figure 7. DMSS uses the modified algorithm to generate all high support association rules from

1) for each k-item frequent set $L = \{a_1, \ldots, a_k\}$: // L is ordered s.t. $a_i < a_j$ for all i<j
2) $H_1 = \{ a_1 \Rightarrow L\text{-}a_1, a_2 \Rightarrow L\text{-}a_2, \ldots, a_k \Rightarrow L\text{-}a_k \}$ // all rules w/ 1 item on LHS
3) for (i=1 to k-1):
4) for each $h \in H_i$:
5) if ($sup(h.\text{LHS}) < minsup$)
6) Remove h from H_i
7) Use H_i to generate H_{i+1}
1) $Answer = Answer \cup \bigcup_{i=1}^{k-1} H_i$

Algorithm for step 7:
a) for each $h \in H_i$:
b) for each $l \in H_i$, $l \neq h$:
c) if (h.LHS and l.LHS share the first i-1 items)
d) $newRule.\text{LHS} = h.\text{LHS} \cup l.\text{LHS}$
e) $newRule.\text{RHS} = h.\text{RHS} \cap l.\text{RHS}$
f) Add *newRule* to H_{i+1}

Figure 8: An algorithm for generating high-precondition support association rules.

frequent sets discovered by the *clone algorithm*. For each k-item frequent set, a maximum of 2^k+1 high-support association rules are possible. With so many high support association rules, many are inevitably uninteresting.

3.2 Association Rule Templates

Association rule templates, introduced by Klemettinen et al. (1994), can be used to describe "flavors" of interesting and uninteresting rules. As such, they can be used to discard inherently uninteresting association rules.

Asscciation rule templates are constructs of the form $be_1 \Rightarrow be_2$ where be_1 and be_2 are Boolean expressions over items and attributes. An association rule $A \Rightarrow B$ *satisfies* rule template $be_1 \Rightarrow be_2$ if A satisfies *be1* and B satisfies *be2*.

Two types of association rule templates are used: *inclusive templates* and *restrictive templates*. Inclusive templates describe types of rules that are *tentatively useful* by specifying necessary group and outcome items. A tentatively useful rule becomes *useful* only if it does not satisfy a restrictive template. Alternatively, an association rule $A \Rightarrow B$ *passes* a set of rule templates if $A \Rightarrow B$ satisfies at least one inclusive template in the set and does not satisfy any restrictive template in the set.

In DMSS, the user can specify a set of rule templates that contains any number of inclusive and restrictive templates. Once a rule passes this set, it is included in the history.

Since rule templates contain domain knowledge, they must be handcrafted by a domain expert. In general, an expert usually has an idea of some types of rules that are interesting, or may know of some types that are never interesting. Even if this is not known initially, iterations through the data mining process often provide insight.

We have found the following strategy effective for creating a set of association rule templates. First, a trial iteration through the DMSS process is performed using as few restrictions as possible on the types of rules generated. Then, the results should be carefully reviewed by a domain expert. The results may look ridiculous. Not to worry, these are only the first steps. By reviewing them, the expert will begin to recognize "flavors" of rules that are useful and others that are not. In the process, he will also recognize types of rules that were unanticipated but are quite useful. With each iteration through the process, rule templates are created, deleted, and modified. After 5 or 6 iterations, the results look good; their number is manageable and some are interesting. It is then time to run a full analysis.

An association rule is useful if it describes the incidence of a meaningful outcome in a meaningful group. For example in the UAB data set, the rule:

{nosocomial, *E. coli*, MICU} ⇒ {R~piperacillin} (3.1)

is useful because it describes the incidence of piperacillin resistance in nosocomial, *E. coli* isolates from the MICU. Significant changes in the incidence proportion of (3.1) over time are interesting to MICU physicians, hospital pharmacists, and infection control officers. On the other hand, the rule

{R~piperacillin} ⇒ {nosocomial, *E. coli*, MICU} (3.2)

describes the proportion of piperacillin resistant isolates that are *E. coli* from the MICU. Since we normally consider piperacillin resistance an outcome, not a group and {*E. coli*, MICU} a group, not an outcome, (3.2) is awkward and inherently uninteresting; therefore, it is not useful.

In general, association rules that have a location item, e.g., MICU, are useful only if the location is on the left-hand side of the rule. This condition can be specified by the following set of association rule templates:

include: Location* ⇒ **

restrict: ** ⇒ Location*

where ** is any item and Location* is any item of the Location attribute.

The rule templates used in the analysis of the UAB data set are shown in Figure 9. These templates were obtained over 5 or 6 iterations of trial analysis, examination of results, and template modification.

Template 1 of Figure 9 specifies that a rule of the form EmptySet ⇒ *B* is tentatively useful only if *B* contains an organism item, gram stain/morphology item, or a

1) include: EmptySet $\Rightarrow$ Organism* $\vee$ GrMp* $\vee$ R~*
2) include: Organism* $\vee$ GrMp* $\vee$ Location* $\Rightarrow$ Organism* $\vee$ GrMp* $\vee$ R~*
3) restrict: R~* $\Rightarrow$ **
4) restrict: ** $\Rightarrow$ Source*
5) restrict: NS* $\vee$ Organism* $\vee$ GrMp* $\Rightarrow$ NS* $\vee$ Organism* $\vee$ GrMp*
6) restrict: $\neg$ EmptySet $\Rightarrow$ Location*
7) restrict: $\neg$(Location* $\vee$ EmptySet) $\Rightarrow$ Organism* $\vee$ GrMp* $\vee$ NS*
8) 8. restrict: Location* $\Rightarrow$ (Organism* $\vee$ GrMp*) $\wedge$ R~*

Figure 9: Association rule templates used in the analysis of the UAB data set. GrMp = gram stain/morphology (NP-GNR, GPC). R~ = resistant antibiotic. NS = nosocomial status (nosocomial, non-nosocomial).

resistant antibiotic item. Template 2 of Figure 9 says that a rule that contains an organism, gram stain/morphology, or location item on the LHS (left-hand side) and an organism item, gram stain/morphology item, or a resistant antibiotic item on the RHS is also tentatively useful.

Of the tentatively useful rules, i.e., those that satisfy an inclusive template, only some are useful -- the others are excluded by restrictive templates. For example, template 3 of Figure 9 excludes all rules with a resistant antibiotic item on the LHS. This makes sense because the LHS of a rule is reserved for groups, and resistant antibiotic items are components of outcomes. Template 4 of Figure 9 excludes tentatively interesting rules that have a source item on the RHS because source outcomes are not useful. However, certain outcomes from source-specific groups are useful. For example, a significant change in the incidence of cephalothin resistance in nosocomial, NP_GNR, urine isolates is interesting. Therefore, we want to include the rule {NP-GNR, urine, nosocomial} $\Rightarrow$

{R~cephalothin}, but we do not want to include the rule {NP-GNR, nosocomial} ⇒ {R~cephalothin, urine}. Templates 2 and 4 accomplish this. Templates 5 through 8 of Figure 9 are more complex. Template 5 excludes tentatively useful rules that have NS, organism, or GrMp items on both the RHS and LHS. For example, template 5 excludes the rule {*P. aeruginosa*, NP-GNR} ⇒ {community-acquired, R~ticarcillin} but does not exclude the rule {*P. aeruginosa*, NP-GNR, community-acquired } ⇒ {R~ticarcillin}. This is desirable because the first rule is awkward; it describes the incidence of non-nosocomial, ticarcillin resistance in *P. aeruginosa* isolates. The second rule is more intuitive; it describes the incidence of ticarcillin resistance in community-acquired, *P. aeruginosa* isolates. Likewise, rules like {*P. aeruginosa*} ⇒ {NP-GNR, R~ticarcillin} and {NP-GNR} ⇒ {*P. aeruginosa*, R~ticarcillin} are excluded by template 5 because the organism item, *P. aeruginosa*, and the GrMp item, NP-GNR, are on opposite sides of the rules. Since every record that contains "*P. aeruginosa*" also contains "NP-GNR" (Section 1.4.4) and the clone algorithm generates max-clones (Section 2.3), every frequent set that contains "*P. aeruginosa*" also contains "NP-GNR." Therefore, template 5 is required to exclude nonsense rules such as {*P. aeruginosa*} ⇒ {NP-GNR, R~ticarcillin} and uninteresting rules such as {NP-GNR} ⇒ {*P. aeruginosa*, R~ticarcillin}. Templates 6 through 8 also exclude tentatively useful association rules by specifying combinations of group and outcome items that are inherently uninteresting.

In the analysis of the UAB data set, an average of 36,129 association rules were generated for each of the 15 data partitions of which 1,820 passed the rule templates of Figure 9. Therefore, about 5% of all rules generated for each data partition made it into

the history. If the number of findings is proportional to the number of rules in the history (a decent approximation), the rule templates significantly reduce pattern glut.

We end this section with a cautionary note. Specifying rule templates requires considerable care; templates that pass too many rules clutter the history and lead to pattern glut, while those that exclude too many rules lead to no new findings. Striking a balance between these two extremes is important to the success of a DMSS analysis. We have found the iterative strategy described above useful in arriving at an appropriate set of association rule templates.

3.3 Updating the History

The *history* H is a database that holds association rules and their incidence proportions for different data partitions. Only association rules that pass the rule templates are included in the history.

From the current data partition p_c, let R_c be the set of high-support association rules that pass the user-defined association rule templates. Using R_c, DMSS updates the history by the procedure outlined in Figure 10. The incidence proportion of rule r in partition p_i is denoted $ip(r, p_i)$.

The history can be conceptualized as a table that contains a row for each association rule and a column for each data partition processed (Table 5). A cell (r_i,p_j) contains the incidence proportion of association rule r_i in partition p_j, $ip(r, p_i)$.

In step 2 of Figure 10, if r is a new rule, i.e., one not already stored in the history, then the incidence proportions of r in prior partitions are computed and stored in the history. These prior incidence proportions constitute a *baseline* for r so that an extreme

1) For each association rule $r \in R_c$:
2) If r is new to H, add r to H and query the database to get the incidence proportions of r for previous partitions. Update H with these incidence proportions.
3) Update H with ip(r, p_c).
4) For each association rule $r \in$ H such that $r \notin R_c$:
5) Query the database to get ip(r, p_c).
6) Update H with ip(r, p_c).

Figure 10: Procedure for updating the history.

Table 5: A Conceptual Structure of the History.

	p_1	p_2	...	p_{c-4}	p_{c-3}	p_{c-2}	p_{c-1}	p_c
r_1	ip(r_1,p_1)	ip(r_1,p_2)		ip(r_1,p_{c-4})	ip(r_1,p_{c-3})	ip(r_1,p_{c-2})	ip(r_1,p_{c-1})	ip(r_1,p_c)
r_2	ip(r_2,p_1)	ip(r_2,p_2)		ip(r_2,p_{c-4})	ip(r_2,p_{c-3})	ip(r_2,p_{c-2})	ip(r_2,p_{c-1})	ip(r_2,p_c)
...								
r_n	ip(r_n,p_1)	ip(r_n,p_2)		ip(r_n,p_{c-4})	ip(r_n,p_{c-3})	ip(r_n,p_{c-2})	ip(r_n,p_{c-1})	ip(r_n,p_c)

deviation in incidence proportion of r from the past to the present can be detected. For example, in the analysis of the UAB data set, a new association rule {NP_GNR, nosocomial} $\Rightarrow$ {R~cefotetan, R~cefuroxime, R~ciprofloxacin} with incidence proportion 5/121 was generated for the January 1997 partition. This means that in January 1997, of the 121 hospital-born, non-*Pseudomonas* gram-negative rod isolates, 5 were resistant to cefotetan, cefuroxime, and ciprofloxacin. To establish a baseline for the new association rule, the incidence proportions of the 3 previous partitions are obtained and stored in H (Table 6). Since the frequent set support threshold was 3 and the precondition support threshold was 8, we know that in all previously processed partitions

sup({NP_GNR, nosocomial, R~cefotetan, R~cefuroxime, R~ciprofloxacin}) < 3 and/or sup({NP_GNR, nosocomial} < 8. In this case, it was the numerator that was less than 3 (Table 6).

Table 6: A Baseline of Incidence Proportions for an Association Rule.

			p_{c-3} Oct96	p_{c-2} Nov96	p_{c-1} Dec96	p_c Jan97
{NP_GNR, nosocomial}	⇒	{R~cefotetan, R~cefuroxime, R~ciprofloxacin}	0/124	0/130	1/100	5/121

Once a rule is stored in the history, it is updated for each new partition regardless of whether or not it is generated in the partition. Therefore, for every association rule in the history, the history contains an up-to-date time-series of incidence proportions. As a result, each row of the history is guaranteed to have incidence proportions for the most recent n partitions including p_c, but may not have incidence proportions for partitions before p_{c-n}. Therefore, the conceptual structure of the history in Table 5 is generally not correct.

The number of previous partitions, n, to be baselined is domain specific and depends on the frequent set support threshold and the windowing schedule used to search for alerts (Section 4.2.1). For the UAB data set, n = 3 is appropriate.

Before launching headlong into the next chapter, let us summarize how the current partition p_c is processed. First, the clone algorithm is used to discover frequent

sets in p_c (Chapter 2). Then, high support association rules are generated using the algorithm in Table 10. Those rules are then filtered by user-defined association rule templates (Section 3.2), and the useful rules are used to update the history. Once the history is updated by the procedure in Figure 10, DMSS is finished processing the current partition. At this point, it can either process another partition, or search the history for interesting patterns. The next chapter is devoted to the search for patterns.

CHAPTER 4

SEARCHING FOR PATTERNS

In this chapter, we describe the process by which DMSS generates potentially interesting patterns. This process includes generating alerts (Section 4.2), eliminating redundant alerts (Section 4.3), and generating event sets (Section 4.6). Since DMSS uses statistical methods to help identify interesting patterns, this chapter also includes a discussion about the role of statistics in data mining and epidemiology. We visit this subject first.

4.1 Statistical Considerations

A couple of statistical issues need to be addressed with respect to the work described in this dissertation. The first is the use (or abuse) of significance tests in epidemiology. The second is the issue of multiple comparisons or "data dredging" as it relates to epidemiological investigations in particular and to data mining in general.

4.1.1 Significance Testing and P-Values

Given a population and a sample from it, classical inferential statistics is concerned with making probabilistic statements about properties of the population based on the corresponding properties of the sample. Specifically, given a population P that has some measurable property k, we form a null hypothesis H_0: k = m, where m is the

suspected value of k in P. Assuming the sampling distribution of a statistic S based on k is known, we can compute a p-value for a sample value of S which is the probability that the observed sample or one more extreme than it is from P under H_0. If this p-value is sufficiently small, $p \leq \alpha$, then under the paradigm of significance testing, one of two conclusions can be made. First, although the chance of drawing this sample from P under H_0 is small, we conclude that the population value of k = m is consistent with the sample. Second, since the probability of drawing this sample from P under H_0 is so small, that we reject H_0 and, with probability α that we are wrong, conclude that in P, $k \neq m$. Either way, we have made an inference about the population P. Such inferences, however, require known sampling distributions. These distributions are almost always based on random sampling.

Classical inferential statistics is firmly based on the idea of random sampling. Randomization, or random sampling, provides known sampling distributions for test statistics under H_0. Consequently, without random samples, sampling distributions of test statistics are unknown and no p-value can be computed for a sample. As a result, no probabilistic inferential statement can be made about the null hypothesis.

In the world of epidemiology, one rarely obtains true random samples. In a review of randomization and causal inference in epidemiology, Greenland (1990) discussed the Framingham study of heart disease and addressed the effect of randomization on the study's interpretation.

In the Framingham heart study, a strong association was noted between cigarette smoking and heart disease--an important result that has often been applied to the population at large. A strict interpretation of the result, however, limits its interpretation

to the population from which the Framingham cohort was taken. The Framingham study cohort, though, was composed mostly of Anglo-English white males born after 1900. Therefore, any claim that the cohort was a random sample of U.S. white males is incorrect, for it fails to account for ethnic diversity of the U.S. white male population at the time. Additionally, of those studied, full compliance and follow-up was not achieved (Gordon, Moore, and Shurtleff 1959). Consequently, the cohort was not a random sample of any population, and the association between cigarette smoking and heart disease has no formal statistical interpretation outside of the study group itself. This does not mean, however, that the result of the study is meaningless. Greenland (1990) stated: "The point is that the study was informative despite the fact that the study statistics bore no randomization interpretation, and that any defensible descriptive interpretation would have been trivial in character." It seems, therefore, that statistical tests devised under strict assumptions of randomization are useful even when those assumptions are violated in real-world studies. So how should one interpret p-values in the absence of randomization?

If a significance test cannot be given a probabilistic interpretation due to the lack of randomization, it can still be used for data description (Greenland 1990) or for decision-making (Fleiss 1986). For example, if one takes 2 samples with no assurance of randomness and wants to compare the proportion of the first sample that is defective, θ_1, to the proportion of the second sample that is defective, θ_2, a statistical significance test of 2 proportions under H_0: $\theta_1=\theta_2$ could be performed. The resulting p-value may indicate that the difference between the proportions is extreme, but because the samples are not guaranteed random, it does not give the probability of getting a more extreme

result under H_0. Therefore, no statistical inference about H_0 can be made based on the result of the test. A careful consideration of the result by one who has an understanding of the processes responsible for generating the samples, however, may still lead to interesting and meaningful conclusions. Therefore, as in the case of the Framingham study, useful information can be extracted from non-probabilistic interpretations of statistical significance tests. All that is needed is critical, expert evaluation of the result.

In the remainder of this document, a result from a significance test on non-random data is called "statistically extreme" or "extreme" if that result would have been statistically significant had the data been randomized. No probabilistic meaning is given to "statistically extreme." It is merely suggests that the result may be interesting.

In some observational studies, the entire group, not a sample, is monitored for the presence of a characteristic *B*. If the monitoring consists of observing the group at specific points in time, is there a way to detect whether or not *B* has changed over time? Specifically, if some proportion p_1 of group *A* at time 1 has characteristic *B*, and some proportion p_2 of group *A* at time 2 has characteristic *B*, what is the probability that the processes that generated characteristic *B* in group *A* at time 1 are the same as the processes that generated characteristic *B* in group *A* at time 2? This is a fundamental question in observational epidemiological studies. Walker (1986) described the problem as follows: "In an observational study, we hypothesize that unmeasured determinants are distributed between comparison groups as if by chance and we apply techniques proper to the analysis of truly probabilistic phenomena to access the possible contribution of 'chance' to a study's finding."

If the processes that generate characteristic *B* in group *A* are the same at time 1 and time 2, we can envision a Population Ψ of which some proportion has characteristic *B*. From Ψ, group *A* at time 1 and group *A* at time 2 are randomly selected. In other words, group *A* at each time is a random sample from Ψ. Under the null hypothesis H_0: $p_1 = p_2$, i.e., the 2 samples were both randomly drawn from Ψ, we can compute a test statistic using p_1 and p_2 whose sampling distribution is known. Using this test statistic, we can then compute a p-value for H_0. This p-value is the probability that the 2 samples or 2 samples whose proportions are more extreme were drawn from Ψ. As we know, however, probabilistic statements of this type depend on a random distribution of determinants among all possible samples in Ψ. Therefore, a probabilistic statement about H_0 depends on the knowledge that the determinants of *B* are randomly distributed in *A* over time. This proposition, however, as Walker (1986) noted, is not testable. Therefore, the best that we can do is to describe the plausibility of the proposition before giving any probabilistic interpretation to the result of a significance test.

The rampant abuse of significance testing in epidemiology is well-described (Fleiss 1986; Walker 1986; Poole 1987; Thompson 1987; Greenland 1990; Rothman and Greenland 1997). The key to the rational use of such tests is to make explicit their purpose and possible interpretations while at the same time considering the limitations of the study designs. Specifically, if the samples are not random, probabilistic interpretations of null hypotheses should be avoided. Additionally, no attempt to state the scientific importance of a result should be based on a p-value alone. With that said, let us move to the second statistical issue that needs to be addressed.

4.1.2 Multiple Comparisons

In data mining, the search for unexpected, potentially interesting patterns requires that many significance tests or comparisons be performed on the data -- one for each candidate pattern. Armitage (1971) refered to this process as "data dredging." Data dredging has been criticized by some statisticians for the following reason: if many tests are performed on random samples under null hypotheses that differences between factors of interest are caused by chance alone, then some tests will return a significant result by chance alone. Specifically, if the significance level of each test is α, and N tests of N null hypotheses are performed, one can expect α *N significant results with the probability $(1-(1-\alpha)^N)$ of getting at least one significant result assuming all N null hypotheses are true. Therefore, in some sense, the null hypotheses are rejected too often and too many false positive results are generated. False positive results, some claim, are the problem of multiple comparisons.

A common solution to reduce the number of false positives generated is to "correct" α by decreasing it in some way that depends on the number of tests to be performed. Rothman (1990) wrote a nice exposition on the philosophical implications of the multiple comparisons problem and argued that the presumptions that underlie "corrections" for multiple comparisons are wrong. Since data mining depends on multiple comparisons, this subject deserves some attention here. The remainder of this section is based largely on the arguments presented by Rothman (1990).

"Corrections" for multiple comparisons assume that all null hypotheses, one per test, are true. This assumption, known as the *universal null hypothesis*, means that no

association exists between any pair of variables. If this is the case, then chance alone is the cause of every unusual finding, and all unusual findings are false positives.

To reduce the number of false positives in real-world studies, one usually reduces α based on the number of tests performed. Such strategies, however, assume that the universal null hypothesis is true--an assumption that Rothman argues is logically inconsistent with our notions of causality.

In truly random systems, there can only be false-positives in the form of chance unusual findings. In the real world, however, we naturally search for causal explanations to observed events. Rothman (1990) noted,

> No empiricist could comfortably presume that randomness underlies the variability of all observations... In a body of data replete with associations, it may be that some are explained by what we call "chance," but there is no empirical justification for a hypothesis that all associations are unpredictable manifestations of random processes... Without a firm basis for posing a universal null hypothesis, the adjustments based on it are counterproductive. Instead, it is always reasonable to consider each association on its own for the information it conveys. (p 45)

Walker (1986) argued against corrections for multiple comparisons when he asked, "Should I discount an interesting finding because the investigator tested some hypothesis which I consider to be absurd?" (p. 558). This question embodies the logical conundrum of mechanistic corrections for multiple comparisons, namely, that the interestingness of a finding depends on the number of tests performed. Thompson (1987) agreed with Walker when he wrote, "Large numbers of comparisons do greatly increase the likelihood of inappropriately excluding the null value for at least one of the total set of associations examined, but the result of a particular association depends in no way on what else has been examined" (p. 193)

The conclusion seems clear. The sensitivity of a significance test should not be adjusted based on the number of comparisons made. Doing so compromises our ability to reject the null hypothesis when the null hypothesis should be rejected. As a result, patterns or associations that deserve further consideration may go undetected. Rothman (1990) boldly concluded,

> To the extent that adjustment for multiple comparisons shields some observed associations from more intensive scrutiny by labeling them as chance findings, it defeats the purpose of scientists... Since an empirical scientist presumes that nature follows regular laws, the scientist confronted with an extreme observation or association should grasp at every opportunity to understand it rather than ignore it. Being impressed by an extreme result should not be considered a mistake in a universe brimming with interrelated phenomena. The possibility that we may be misled is inherent to the trial-and-error process of science; we might avoid all such errors by eschewing science completely, but then we learn nothing. (p. 46)

According to these arguments, data mining should not be constrained in contrived ways allowing the user should have access to all potentially interesting findings. Clearly, however, this must be done in a reasonable fashion, for if the user is overwhelmed with potentially interesting finding, i.e., pattern glut, little is accomplished. Therefore, the delicate balance between ignoring potentially interesting findings and overwhelming the user with too many of them is important to the success of any data mining effort. Therefore, each step taken to reduce pattern glut allows us to maximize the sensitivity of the system. This is desirable, for in the spirit of the Rothman's argument, data mining should present as many potentially interesting findings to the user as possible so that each finding can be evaluated on its own merit.

In the remainder of the chapter, we describe how DMSS generates potentially interesting findings.

4.2 Alerts

DMSS discovers low-level patterns, then clusters them into higher level patterns. The higher level patterns, called *events*, are presented to the user as potentially interesting findings. The low-level patterns, called *alerts*, are not seen by the user, but are used to construct events. Therefore, the search for potentially interesting findings begins with alerts.

DMSS generates alerts by analyzing information stored in the history. An *alert* describes an extreme change in the incidence of an outcome B in a group A over time.

Table 7: A Possible Outbreak of Bacterial Infection.

Association Rule			p_{c-5}	p_{c-4}	p_{c-3}	p_{c-2}	p_{c-1}	p_c
{nosocomial, SICU, trach aspirate}	==>	{*Acinetobacter baumannii*}	0/11	0/10	0/9	0/13	2/9	3/9
			w_p				w_c	

For example, Table 7 describes the incidence of *Acinetobacer baumannii* in nosocomial, tracheal aspirate, surgical intensive care unit (SICU) isolates over the past 6 partitions. Clearly, a shift in incidence occurs between the first 4 months and the most recent 2 months of the series. If we call the first, second, third, and fourth months the *past window*, w_p, and the fifth and sixth the *current window*, w_c, we can ask if there is an extreme change in the incidence between w_p and w_c. To find out, we compute the cumulative incidence proportion for w_p and the cumulative incidence proportion for w_c,

compare the two by a statistical test of 2 proportions, and compute their relative difference. If the difference between the 2 proportions is statistically extreme and their relative difference exceeds a user-defined threshold, we say there is an *extreme difference* in the incidence proportions of w_p and w_c.

This is exactly how DMSS generates an alert for an association rule *r*. First, it constructs a current window and a past window on the time-series of incidence proportions of *r* (Section 4.2.1). Second, it computes the cumulative incidence proportion for each window (Section 4.2.2). Third, it compares the two cumulative incidence proportions by a test of 2 proportions (Section 4.2.3), then computes their relative difference (Section 4.2.4). Finally, if the difference between the proportions is statistically extreme, and if the relative difference exceeds a user-defined threshold, it generates an alert. If an alert is not generated, then a different pair of current and past windows is formed and their cumulative incidence proportions are compared. This continues for the same association rule until an alert is generated or no more current and past window pairs remain to be formed. DMSS generates all alerts by executing the procedure just described on every association rule in the history. In the following sections, we describe this procedure in detail.

1) The difference between the two cumulative incidence proportions is statistically extreme.
2) The relative difference between the two cumulative incidence proportions exceeds a user-defined threshold.

Figure 11: Criteria for an extreme difference between two cumulative incidence proportions.

1) Compute cumulative incidence proportions for w_c, w_p.
2) If there is an extreme difference between the two cumulative incidence proportions, generate a type-one-alert.

Figure 12: Process for generating an alert given w_p and w_c.

4.2.1 Time Windows and Windowing Schedules

Generating an alert for an association rule $r = A \Rightarrow B$ starts with constructing a current window, w_c, and a past window, w_p, on the time-series of incidence proportions of r. Each time window corresponds to a set of contiguous data partitions. For example, in Table 7, w_c corresponds to partitions from the two most recent partitions: p_c and p_{c-1}, and w_p to the 4 previous partitions: p_{c-2}, p_{c-3}, p_{c-4}, and p_{c-5}. The example in Table 7 also illustrates several properties of current and past windows. These properties are listed in Figure 13.

Given a time-series of incidence proportions for an association rule r, a *windowing schedule* specifies a series of past window and current window pairs for the time-series. Each entry in the schedule is of the form $(|w_p|, |w_c|)$ where $|w_p|$ is the number of partitions in w_p and $|w_c|$ is the number of partitions in w_c.

1) Each is composed of one or more contiguous partitions, p_i.
2) w_c contains p_c, the current partition.
3) w_c and w_p are disjoint and contiguous.

Figure 13: Properties of current and past time windows.

The windowing schedule used in the analysis of the UAB data set (Figure 14) specifies a series of window pairs for each rule $A \Rightarrow B$ in the history such that changes in the incidence of B in A over several time scales can be detected.

1)	(3, 1)
2)	(6, 2)
3)	(9, 3)

Figure 14: Windowing schedule for the analysis of the UAB data set.

For example, when this schedule (Figure 14) is used to generate window pairs for the time-series of incidence proportions in Table 8, the two window pairs of Table 9 are created. Since window pair 3 of the schedule requires more than 8 partitions in the time-series, it could not be created.

Each window pair forms the basis for a comparison between the cumulative incidence proportion of the past window and the cumulative incidence proportion of the current window. Therefore, window pair 1 of Table 9 allows the comparison of the incidence proportion of current partition to the cumulative incidence proportion from the previous 3 partitions, and window pair 2 allows for the comparison of the cumulative incidence proportion from the most recent 2 partitions to that from the previous 6 partitions.

When a difference between cumulative incidence proportions is extreme by the criteria in Figure 11, DMSS generates an alert. When this happens, no additional window pairs for that rule are considered. For example, if an extreme difference is found between the proportions of the window pair specified by the first entry of a windowing

Table 8: A Series of 8 Incidence Proportions.
{SICU} ⇒ {NP_GNR, nosocomial, *Enterobacter cloacae*}

p_{c-7}	p_{c-6}	p_{c-5}	p_{c-4}	p_{c-3}	p_{c-2}	p_{c-1}	p_c
3/38	1/26	2/32	3/33	1/26	3/54	7/50	9/46

Table 9: Window Pairs Generated when the Windowing Schedule of Figure 14 is Applied to the Incidence Proportions of Table 8.

window pair 1					[]1/26	[]3/54	[]7/50	*9/46
window pair 2	[]3/38	[]1/26	[]2/32	[]3/33	[]1/26	[]3/54	*7/50	*9/46

[] => in w_p * => in w_c

schedule, then window pairs specified by the following entries in the schedule are not generated, and the next rule in the history is considered.

In the UAB data set, detecting emerging problems and outbreaks is a primary concern. Therefore, the windowing schedule (Figure 14) is designed so that window pairs more sensitive to recent changes in incidence are generated before those less sensitive to recent changes.

4.2.2 Cumulative Incidence Proportion

A *cumulative incidence proportion* is itself an incidence proportion computed by "summing" one or more incidence proportions in a time window. The cumulative incidence proportion of a rule $r = A \Rightarrow B$ in a time window w is given by:

$$cip(r, w) = \frac{\sum_{p_i \in w} sup(A \cup B, p_i)}{\sum_{p_i \in w} sup(A, p_i)}$$

Simply stated, the numerator of the cumulative incidence proportion is the sum of the numerators of all incidence proportions in w; the denominator is the sum of the denominators of all incidence proportions in w. For example, for window pair 1 in Table 9, the cumulative incidence proportion of w_p, cip(r, w_p), is 11/130, and the cumulative incidence proportion of w_c, cip(r,w_c), is 9/46. For window pair 2 in Table 9, cip(r,w_p) is 13/209 and cip(r,w_c) is 16/96.

With 2 cumulative incidence proportions in hand, it is time to compare them to see if they are extremely different.

4.2.3 Statistical Tests of 2 Proportions

The comparison of 2 cumulative incidence proportions for extreme difference is a 2 step process (Figure 11) that starts with a statistical test to see if the difference between the two proportions is statistically extreme. To see how this is accomplished, it helps to summarize the proportions in a 2 x 2 contingency table.

The 2 x 2 contingency table, sometimes called the fourfold table, is commonly used for summarizing statistical data to detect associations between two independent binomial random variables. In general, the 2 x 2 contingency table looks like the one in Table 10.

Typically, significance tests on data in 2 x 2 contingency tables test for the independence of the 2 variables A and B. The test hypothesis is that the presence of characteristic A in the population is independent of the presence of characteristic B in the

Table 10: A General 2 x 2 Contingency Table.

Characteristic A	Characteristic B		Total
	Present	Absent	
Present	n_{11}	n_{12}	$n_{1.}$
Absent	n_{21}	n_{22}	$n_{2.}$
Total	$n_{.1}$	$n_{.2}$	$n_{..}$

same population. Generally, this null hypothesis can be tested by computing a chi-squared statistic or a Fisher's exact statistic and comparing it to an arbitrary cut-off value, α, in the respective distribution. Both statistics are based on the cell values in the observed table versus the cell values in the table that would be expected if characteristic B was independent of characteristic A.

The 2 x 2 contingency table is discussed in most introductory statistics texts. A thorough basic treatment of 2 x 2 contingency tables is given by Rosner (1990). Subtle issues about creating 2 x 2 contingency tables and assessing their significance are discussed by Fleiss (1973).

The 2 x 2 contingency table can also be used to summarize 2 cumulative incidence proportions. In Table 11, the incidence of outcome B in group A during time window one is $h_1 = n_{11}/n_{1.}$, and the incidence of B in A during time window two is $h_2 = n_{2}/n_{2.}$. To compare the two incidence proportions, a p-value for $H_0:h_1=h_2$ is computed. If the expected values of n_{11}, n_{12}, n_{21}, n_{22} are each greater than five under H_0, then a chi-squared test statistic is computed. If any of the 4 expected values is less than 5, then an exact p-value given by Fisher's exact test is computed. From here on, we refer to a test of two proportions h_1 and h_2 that returns a p-value under $H_0:h_1 = h_2$ as $ttp(h_1,h_2)$.

Table 11: A General 2 x 2 Contingency Table for the Comparison of two Incidence Proportions.

	Outcome B		Group A
	Present	Absent	Total
Time window one	n_{11}	n_{12}	$n_{1.}$
Time window two	n_{21}	n_{22}	$n_{2.}$
Total	$n_{.1}$	$n_{.2}$	$n_{..}$

Whether $ttp(h_1,h_2)$ is returned by the chi-squared or Fisher's exact test depends on the expected frequencies of the cell data in the corresponding 2x2 contingency table. A justification for using the chi-squared statistic to compare two binomial proportions is given by Brownlee (1965). General descriptions of the chi-squared test for the equality of two binomial proportions and of Fisher's exact test for the same purpose are given by Rosner (1990).

DMSS employs tests of two proportions for classifying pairs of cumulative incidence proportions as statistically extreme or not. "Statistically extreme" as discussed in Section 4.1.1, carries no probabilistic interpretation. When a significance test of two cumulative incidence proportions returns a p-value less than α (e.g. 0.01), the two proportions are classified as statistically extreme. For example, the contingency table for the two cumulative proportions computed from window pair 1 of Table 9 is given in Table 12. The cell values are adjusted according to Yate's continuity correction (Rosner 1990). Since the expected value of each cell is greater than 5, the following statistic is computed.

$$X^2 = \frac{176[(11.5)(37.5) - (118.5)(8.5)]^2}{(130)(46)(20)(156)} = 3.13$$

Table 12: A 2 x 2 Contingency Table for the Comparison of Two Cumulative Incidence Proportions from Table 9.

	{*Enterobacter cloacae*, NP_GNR, nosocomial}		{SICU}
	Present	Absent	Total
w_p	11.5	118.5	130
w_c	8.5	37.5	46
Total	20	156	176

The sampling distribution of this statistic under H_0 is the chi-squared distribution with one degree of freedom (DOF). Since $X^2_{0.01,1} = 6.635$ is the value of chi-squared with one DOF that corresponds to a p-value of 0.01, $X^2 = 3.13$ leads us to conclude that the difference between the two cumulative incidence proportions is not statistically extreme.

For the two cumulative incidence proportions computed from window pair 2 of Table 9, $X^2 = 7.56$ which is greater than $X^2_{0.01,1} = 6.635$. This allows us to conclude that the difference between the two incidence proportions computed from window pair 2 of Table 9 is statistically extreme.

The choice of α for classifying the results of significance tests is rather arbitrary, and indeed, this is a general criticism of significance testing (Rothman and Greenland 1997). However, since significance testing is used in DMSS for exploratory purposes only, the rather arbitrary choice of α is not troublesome. In our experience, an alpha of 0.01 is preferable to an alpha of 0.05 because the latter tends to classify too many patterns as statistically extreme thereby increasing the number of potentially interesting findings that the user is required to evaluate.

The first step in comparing two cumulative incidence proportions is complete. Given two proportions, DMSS classifies them as statistically extreme or not. The second step in comparing two cumulative incidence proportions is simply to evaluate their relative difference.

4.2.4 The Relative Difference between 2 Incidence Proportions

At first thought, evaluating the relative difference between two incidence proportions seems redundant. After all, the proportions have already been compared statistically and their difference should have already been considered. The problem with comparing two observed proportions, however, is that the populations giving rise to the observed samples will inevitably differ to some extent (Fleiss 1973). In our model, this means that the processes responsible for the cumulative incidence proportion of a past window will almost always differ, at least by some miniscule amount, from the processes responsible for the cumulative incidence proportion of the current window. Consequently, according to Fleiss (1973), a type-I error under the null hypothesis that the processes are identical probably never occurs in practice because the null hypothesis is usually false. Therefore, given sufficiently large samples, even small differences in observed proportions are statistically significant (or in our case statistically extreme), thereby leading to the rejection of H_0.

The ability of statistical significance tests to identify small differences between proportions as statistically significant when large samples are obtained has practical implications for DMSS. Namely, the user is generally not interested in small changes in incidence over time. Therefore, if alerts are generated only on the condition that two

proportions are statistically extreme, the user would inevitably be required to review findings that are uninteresting because they describe incidence changes that are small in magnitude. To alleviate this problem, DMSS evaluates the relative difference between the two incidence proportions and compares it to a user-defined threshold. For example, if $cip(R,w_p)$ is 15/100 and the $cip(R,w_c)$ is 30/100, then the relative difference between the two is (30-15)/15 = 1. In general, the relative difference between two proportions is

$$rd(p1, p2) = \frac{p2 - p1}{p1}$$

where $p1 = cip(R,w_p)$ and $p2 = cip(R,w_c)$.

If the user provides an *relative difference threshold* of 1, then for two cumulative incidence proportions $p1$ and $p2$, $p2$ must be at least twice $p1$ for $rd(p1,p2) \geq 1$.

In our experiments, a relative difference threshold of 1 is appropriate. This threshold reduces the number of alerts generated in the analysis of the UAB and CDC data sets by about 15%. Importantly, review of those alerts excluded by this criterion reveals that no potentially interesting alerts were rejected by it. This type of domain expert review and interaction is crucial to real-world data mining exercises, and in this case, it guided the selection of the relative difference threshold.

4.3 Redundant Alerts

Since DMSS is designed to be a real-time surveillance system, it searches for interesting patterns after each new data partition is processed. Because some window pairs of the windowing schedule usually extend the current window back in time to include partitions before the current partition, it is possible for DMSS to generate alerts

that are essentially repeats of alerts generated in the past. For example, let us consider the series of incidence proportions in Table 13.

Table 13: A Series of Incidence Proportions that Contains a Redundant Alert.

p_{c-7}	p_{c-6}	p_{c-5}	p_{c-4}	p_{c-3}	p_{c-2}	p_{c-1}	p_c
0/20	1/18	0/25	0/18	0/22	1/19	11/20	5/19

With the windowing schedule in Figure 14, the first window pair generated for the proportions in Table 13 is $w_c = \{p_c\}$ and $w_p = \{p_{c-1}, p_{c-2}, p_{c-3}\}$. For this pair, there is no statistically extreme difference between the two corresponding cumulative incidence proportions. The next window pair generated by the schedule is $w_c = \{p_c, p_{c-1}\}$ and $w_p = \{p_{c-2}, p_{c-3}, p_{c-4}, p_{c-5}, p_{c-6}, p_{c-7}\}$. For this pair, the difference between the corresponding cumulative incidence proportions is statistically extreme and since $rd(cip(R,w_p), cip(R,w_c)) = 25$, an extreme difference between the two proportions is noted and an alert is generated. This is called the *current alert*.

A casual look at the series of proportions, however, indicates that the incidence actually peaks at p_{c-1}. This suggests that a more alarming alert may have been generated immediately after p_{c-1} was processed and the user would have already been notified of this problem. If so, the current alert is redundant.

To determine if the current alert is redundant, we return to the windowing schedule (Figure 14) and again use entry 1, only this time to generate $w_c = \{p_{c-1}\}$ and $w_p = \{p_{c-2}, p_{c-3}, p_{c-4}\}$. In order not to confuse this window pair with the one that was used to identify the current alert, we call this pair (w_c', w_p'). Therefore, $(w_p', w_c') = (\{p_{c-2}, p_{c-3}, p_{c-4}\}, \{p_{c-1}\})$ and $(w_p, w_c) = (\{p_{c-2}, p_{c-3}, p_{c-4}, p_{c-5}, p_{c-6}, p_{c-7}\}, \{p_c, p_{c-1}\})$. Now, we repeat

the test of two proportions on cip(R, w_p') = 1/59 and cip(R, w_c') = 11/20 and see if their difference is statistically extreme. It is. Next, we compute rd(cip(R, w_p'), cip(R, w_c')) = 32 and compare it rd(cip(R, w_p), cip(R, w_c)) = 25. Since 32 is greater than 25, an alert was indeed generated with window pair (w_p', w_c'). Moreover, the current alert is likely not as alarming as the one identified with (w_p', w_c') since rd(cip(R, w_p), cip(R, w_c)) smaller than rd(cip(R, w_p'), cip(R, w_c')) indicates that the change in incidence was more pronounced in the previous alert than in the current one. Therefore, the problem, while it still may exist, is less in magnitude than it was in the previous alert. Since the current alert is less alarming and the user is already aware of the problem from the previous alert, the current alert is redundant.

The process used in the example is generalized in the procedure outlined in Table 26. DMSS uses this procedure to identify redundant alerts. Line 6 of the procedure (Figure 15) specifies that w_c' is the least current ($|w_c| - d_c$) partitions of w_c and line 7 specifies that w_p' is the most current ($|w_p| - d_p$) partitions of w_p. For example, if (w_p, w_c) for the current alert is ({p_{c-5}, p_{c-4}, p_{c-3}}, {p_{c-2}, p_{c-1}, p_c}) and the windowing schedule contains entries (3,1), (3,2), and (2,2), then the procedure will test (w_p', w_c') = ({p_{c-5}, p_{c-4}, p_{c-3}}, {p_{c-2},}) for schedule entry (3,1), (w_p', w_c') = ({p_{c-5}, p_{c-4}, p_{c-3}}, {p_{c-2}, p_{c-1}}) for entry (3,2), and (w_p', w_c') = ({p_{c-4}, p_{c-3}}, {p_{c-2}, p_{c-1}}) for entry (2,2) as needed.

Removing duplicate alerts is yet another example of an attempt to reduce pattern glut. Our experiments on the UAB data set and the CDC data set (Section 5.3) show that by eliminating redundant alerts, the number of alerts generated in a search for interesting patterns is reduced by as little as 1% to as much as 25%. For the entire analysis of the

1) (w_p, w_c) = window pair for *current alert*.
2) for each pair (x, y) $\in$ windowing schedule:
3) if ($x \leq |w_p|$ & $y < |w_c|$)
4) $d_c = |w_c| - y$.
5) $d_p = |w_p| - x$.
6) $w_c' = w_c$ without the d_c most current partitions.
7) $w_p' = w_p$ without the d_p least current partitions.
8) if ((cip(R, w_c') - cip(R, w_p') is statistically extreme)
9) & (rd(cip(R, w_p'), cip(R, w_c')) $\geq$ rd(cip(R, w_p), cip(R, w_c)))
10) *current alert* is redundant.
11) break.

Figure 15: Procedure to identify redundant alerts.

UAB data set, eliminating redundant alerts decreases the total number of alerts by about 10%.

4.4 Alerts: The Big Picture

Let *gen_type1_alert*(*r*,wp,*a*) be a function that takes an association rule *r* and a window pair wp. *gen_type1_alert* utilizes the entire process for generating an alert, including ignoring duplicate alerts, and returns *true* and an alert in *a*, if an alert is generated, and *false* if an alert is not generated.

Until now, we have described how to identify an alert and how to generate them, but have not specified exactly what an alert contains. An alert simply contains the information that was used to identify it: the association rule, a description of the window

```
1) alerts = ∅.
2) For each rule ∈ history:
3)     for each (x,y) ∈ windowing schedule:
4)         Generate the current window pair wp using (x,y).
5)         if (gen_type1_alert(rule, wp, a))
6)             alerts = alerts ∪ a.
7)             break. // for each (x,y)
```

Figure 16: Algorithm for generating all alerts.

pair, the result of the test of the two cumulative incidence proportions, and the relative difference between the two cumulative incidence proportions.

4.5 An Alternate Method for Identifying Alerts

In this section, we examine the use of cell occupancy models and exact probability distributions to identify alerts. Cell occupancy models are useful in epidemiologic investigations (Ederer et al. 1964; Grimson 1993; Wallenstein 1980). In particular, the *MAX* statistic and the scan statistic are used in epidemiologic cluster investigations (Ederer et al. 1964; Grimson 1993). In this section we briefly describe the MAX statistic and why it is not used in DMSS.

An *ordinary cell occupancy model* consists of L cells and N items randomly distributed among them. MAX(N,L) is the sampling distribution of the maximum number of items, *MAX*, found in any cell under the null hypothesis that N items are randomly assigned to L cells. pMAX(n,N,L) is the probability of finding at least n items in a single cell under H_0. Consider the event frequency data for the 12 disjoint partitions in Table 14.

Table 14: An Apparent Cluster of Disease.

# cases	1	1	1	2	1	0	5	1	0	1	1	1
partition	1	2	3	4	5	6	7	8	9	10	11	12

Looking over this data, we may suspect that the 5 events of partition 7 comprise a cluster of events. Under the null hypothesis that the 15 events are randomly distributed over the 12 partitions, pMAX(5,15,12) = 0.07 is the probability of observing 5 or more events in any one partition. After a critical examination of data with respect to H_0, we may or may not conclude that the 5 events of partition seven comprise a cluster. Again, the interpretation of a p-value depends on a critical examination of the data and the assumptions of the model distribution.

For the use of *MAX* in epidemiologic studies, the following 2 conditions must be met:

1. The population at risk of an event must be constant over all cells.
2. The risk of an event for each member of the population must be the same within and between cells.

While condition 2 is also a condition of the statistical tests of 2 proportions described in Section 4.2.3, condition 1 is not a condition of those tests. Therefore, *MAX* has an additional constraint that needs to be considered.

In observational studies, condition 2 is rarely, if ever, satisfied. Disease processes that generate health events are not static over time. Moreover, this condition is not testable. This was discussed in Section 4.1.1. With little thought, it is clear that condition 1 is rarely satisfied. As a result, probabilistic interpretations of p-values from observational studies are not allowed. This is consistent with our statistically extreme

interpretation of p-values from tests of 2 proportions in Section 4.2.3. A similar interpretation could be given to values of pMAX.

While it is not valid to make statistical inferences from significance tests in observational studies, we nevertheless want to violate as few assumptions of the underlying statistical models as possible so that the test result is in some sense as accurate as possible. This presents a significant problem for the use of the MAX statistic in DMSS because for an association rule $A \Rightarrow B$, the size of the population at risk, i.e., sup(A), usually changes from one partition to another. Consequently, condition 1 for using the MAX statistic is blatantly and consistently violated. Therefore, to use the MAX statistic when the population or group size changes requires attempts to normalize its size for each partition. This extra normalizing step is not required in the tests of 2 proportions.

The MAX statistic also requires that a database of p-values be generated for certain values of n, N, and L so that a p-value does not have to be computed or simulated each time it is required. In addition, since L must be an integer, the MAX statistic does not allow one to compare say the incidence proportion over the last 2 months to that of the previous three months. The tests of 2 proportions do not have this restriction.

Even with these disadvantages, we tried the MAX statistic in DMSS. This effort included generating a sizable database of p-values and employing a normalizing scheme for group sizes across partitions. After much work, we concluded that tests based on the MAX statistic work about as well as the tests of 2 proportions in identifying statistically extreme changes between the incidence proportions of past and current windows. Since the methods that employ tests of 2 proportions are conceptually cleaner and more

efficient than the methods that use the MAX statistic for identifying alerts, we currently use the former DMSS.

4.6 Event Sets and Events

Consider the set of alerts in Figure 17. This set has several characteristics. First, the left-hand sides of the association rules for alerts 1 through 5 are each a subset of the left-hand side of the association rule for alert 6. Likewise, the right-hand sides of the rules for alerts 1 through 5 are each a subset of the right-hand side of alert 6. Second, the current window, $w_c = \{4\}$, is the same for all alerts. The past window, $w_p = \{1, 2, 3\}$, is also the same for all alerts. Finally, changes in the cumulative support of {SICU, NP_GNR, nosocomial, *K. pneumonia*, sputum, R~A1, R~A2} between w_p and w_c, which is the change in the numerator of the cumulative incidence proportion between w_c and w_p in alert 6, account for most of the changes in the numerators of the cumulative incidence proportions between w_p and w_c in alerts 1 through 5. From these characteristics, our intuition tells us that alert 6 is responsible for alerts 1 through 5. If so, alert 6 contains the pertinent information about all alerts in the set and is, therefore, the only alert from the set that need be presented to the user. Indeed, for this set, alert 6 is the *event* and the entire set is the *event set* of alert 6.

4.6.1 Alert capture

Association rule $A1 \Rightarrow B1$ is called a *descendent* of association rule $A2 \Rightarrow B2$ if $A2$ is contained in $A1$ and $B2 \subseteq B1$. $A2 \Rightarrow B2$ is called an *ancestor* of $A1 \Rightarrow B1$ if the same conditions hold.

For any two alerts x and y, x is said to *capture* y if the association rule of x, $A_x \Rightarrow B_x$, is a descendent of the association rule of y, $A_y \Rightarrow B_y$, (w_p, w_c) of x is equal to (w_p, w_c) of y, and

$$\text{ttp}\left(\frac{\sup(Ay \cup By, w_p) - \sup(Ax \cup Bx, w_p)}{\sup(Ay, w_p)}, \frac{\sup(Ay \cup By, w_c) - \sup(Ax \cup Bx, w_c)}{\sup(Ay, w_c)}\right) \quad (4.1)$$

is greater than 0.01. Intuitively, alert x captures alert y if when x is "removed" from y, y is no longer an alert.

1) {EmptySet}⇒{R~A2}
[]1: 1/950 | []2: 1/812 | []3: 2/768 | *4: 8/780

2) {SICU}⇒{R~A2}
[]1: 0/57 | []2: 1/60 | []3: 2/52 | *4: 7 65

3) {SICU, NP_GNR}⇒{R~A1, R~A2}
[]1: 0/23 | []2: 0/20 | []3: 2/18 | *4: 7 21

4) {SICU, NP_GNR, nosocomial}⇒{R~A1, R~A2}
[]1: 0/11 | []2: 0/10 | []3: 1/12 | *4: 5 13

5) {SICU, NP_GNR, nosocomial, *K. pneumonia*}⇒{R~A1, R~A2}
[]1: 0/5 | []2: 0/5 | []3: 1/6 | *4: 5/7

6) {SICU, NP_GNR, nosocomial, *K. pneumonia*, sputum}⇒{R~A1, R~A2}
[]1: 0/5 | []2: 0/4 | []3: 1/4 | *4: 5/6

Figure 17: A set of related alerts. Bracketed partitions (e.g. []1) are in w_p. Starred partitions (*4) are in w_c.

For example, let us test alerts 3 and 6 from Figure 17 to see if alert 6 captures alert 3, as we suspect. First, we must check to see if the rule of alert 6 is a descendant of the rule of alert 3. It is. Next, we want to check if (w_p, w_c) of x is equal to (w_p, w_c) of y. It is. Now, we want "remove" alert 6 from alert 3, then test the altered alert 3 to see if it

is still statistically extreme. Substituing the appropriate values into expression 4.1, we get $ttp\left(\frac{2-1}{61}, \frac{7-5}{21}\right) = 0.32$ which is greater than 0.01. Therefore, alert 6 does capture alert 3. By the same process, alert 6 also captures alerts 1, 2, 4, and 5.

4.6.2 Event Sets

An *event set x'* is the alert *x* together with all alerts that *x* captures. Event sets are created by the algorithm in Figure 18.

```
1)  A is the set of all alerts.
2)  while A ≠ ∅:
3)      for each a ∈ A:
4)          if (A does not contain a descendant of a)
5)              Create a new event set a'.
6)              Add a to a' as the event.
7)              Remove a from A.
8)              for each b ∈ A:
9)                  if (a captures b)
10)                 Add b to a'.
11)                 Remove b from A.
```

Figure 18: Algorithm for generating event sets.

4.6.3 Events and Pattern Glut

After executing the algorithm in Figure 18 on a set of alerts, each alert is a member of an event set. Some event sets may contain only one alert -- the event. Others, however, will contain many alerts, only one of which is the event.

Since the event of an event set contains all pertinent information for the entire set, only events are shown to the user as potentially interesting patterns; all other alerts are redundant.

Figure 19 contains an actual event set generated during the analysis of the UAB data set. The event is the first alert listed in the table.

In the analysis of a data set, a search for potentially interesting patterns is usually conducted after each partition is processed. For the UAB data set, each search generated an average of 119 alerts and 27 events. The Jan-97 search yielded 413 alerts and 28 events, and the Aug-97 search generated 462 alerts and 49 events. Table 15 contains other related summary statistics.

One purpose of eliminating redundant alerts (Section 4.3) and generating event sets is to reduce pattern glut. Pattern glut is a subjective measure that depends in part on the time and resources that the user can commit to evaluating potentially interesting patterns. With fewer patterns to consider, the user will feel less taxed by the data mining process. Consequently, less time is spent analyzing misleading results, and more time is spent doing productive investigations and taking corrective action. After all, data mining is useless unless its results can be evaluated and acted upon in an efficient manner. In our experiences with the UAB and CDC data sets, DMSS generates a manageable number of patterns. Without association rule templates (Section 3.2) and the methods described in this chapter, however, the number of patterns generated for each data set would be completely unmanageable. After all, the task of looking over 27 patterns is much less daunting than that of looking over 119, much less 462! Therefore, DMSS is

event* {*Klebsiella pneumoniae*, NP_GNR} ==> {R~Cefotetan, R~Ceftazidime, R~Cefuroxime, R~Cefazolin, R~Cephalothin, R~Ceftriaxone}
[]9609: 0/79 | []9610: 1/79 | []9611: 4/85 | *9612: 8/76 |
-- {*Klebsiella pneumoniae*, NP_GNR} ==> {R~Cephalothin}
[]9609: 7/79 | []9610: 5/79 | []9611: 7/85 | *9612: 14/76 |
-- {NP_GNR} ==> {R~Cefotetan, R~Ceftriaxone}
[]9609: 0/563 | []9610: 7/504 | []9611: 9/439 | *9612: 13/392 |
-- {NP_GNR} ==> {R~Ceftazidime, R~Cefazolin}
[]9609: 8/563 | []9610: 11/504 | []9611: 8/439 | *9612: 16/392 |
-- {NP_GNR} ==> {R~Ceftazidime, R~Cephalothin}
[]9609: 5/563 | []9610: 6/504 | []9611: 7/439 | *9612: 13/392 |
-- {NP_GNR} ==> {R~Ceftazidime, R~Ceftriaxone}
[]9609: 2/563 | []9610: 15/504 | []9611: 19/439 | *9612: 19/392 |
-- {NP_GNR} ==> {R~Cefuroxime, R~Cephalothin}
[]9609: 11/563 | []9610: 9/504 | []9611: 8/439 | *9612: 18/392 |
-- {NP_GNR} ==> {R~Cefazolin, R~Cephalothin}
[]9609: 18/563 | []9610: 14/504 | []9611: 18/439 | *9612: 27/392 |
-- {*Klebsiella pneumoniae*, NP_GNR} ==> {R~Cefuroxime, R~Cephalothin}
[]9609: 5/79 | []9610: 3/79 | []9611: 5/85 | *9612: 11/76 |
-- {*Klebsiella pneumoniae*, NP_GNR} ==> {R~Cefazolin, R~Cephalothin}
[]9609: 2/79 | []9610: 2/79 | []9611: 5/85 | *9612: 10/76 |
-- {NP_GNR} ==> {R~Cefotetan, R~Ceftazidime, R~Ceftriaxone}
[]9609: 0/563 | []9610: 7/504 | []9611: 9/439 | *9612: 12/392 |
-- {NP_GNR} ==> {R~Ceftazidime, R~Cefuroxime, R~Cephalothin}
[]9609: 5/563 | []9610: 6/504 | []9611: 7/439 | *9612: 12/392 |
-- {NP_GNR} ==> {R~Ceftazidime, R~Cefuroxime, R~Ceftriaxone}
[]9609: 0/563 | []9610: 8/504 | []9611: 11/439 | *9612: 15/392 |
-- {NP_GNR} ==> {R~Ceftazidime, R~Cefazolin, R~Cephalothin}
[]9609: 5/563 | []9610: 6/504 | []9611: 5/439 | *9612: 12/392 |
-- {NP_GNR} ==> {R~Cefuroxime, R~Cefazolin, R~Cephalothin}
[]9609: 7/563 | []9610: 8/504 | []9611: 6/439 | *9612: 15/392 |
-- {NP_GNR} ==> {R~Cefuroxime, R~Cefazolin, R~Ceftriaxone}
[]9609: 1/563 | []9610: 9/504 | []9611: 8/439 | *9612: 13/392 |
-- {*Klebsiella pneumoniae*, NP_GNR} ==> {R~Ceftazidime, R~Cefazolin, R~Cephalothin}
[]9609: 1/79 | []9610: 1/79 | []9611: 4/85 | *9612: 9/76 |
-- { NP_GNR} ==> {R~Cefotetan, R~Ceftazidime, R~Cefuroxime, R~Ceftriaxone}
[]9609: 0/563 | []9610: 6/504 | []9611: 8/439 | *9612: 12/392 |
-- {NP_GNR} ==> {R~Ceftazidime, R~Cefuroxime, R~Cefazolin, R~Cephalothin}
[]9609: 5/563 | []9610: 6/504 | []9611: 5/439 | *9612: 11/392 |
-- {NP_GNR} ==> {R~Ceftazidime, R~Cefuroxime, R~Cefazolin, R~Ceftriaxone}
[]9609: 0/563 | []9610: 8/504 | []9611: 8/439 | *9612: 13/392 |
-- {NP_GNR} ==> {R~Ceftazidime, R~Cefuroxime, R~Cephalothin, R~Ceftriaxone}
[]9609: 0/563 | []9610: 3/504 | []9611: 7/439 | *9612: 10/392 |
-- { NP_GNR} ==> {R~Cefotetan, R~Ceftazidime, R~Cefuroxime, R~Cephalothin R~Ceftriaxone}
[]9609: 0/563 | []9610: 1/504 | []9611: 5/439 | *9612: 9/392 |
-- {NP_GNR} ==> {R~Ceftazidime, R~Cefuroxime, R~Cefazolin, R~Cephalothin R~Ceftriaxone}
[]9609: 0/563 | []9610: 3/504 | []9611: 5/439 | *9612: 9/392 |

Figure 19: An event set from the analysis of the UAB data set.

Table 15: Summary Statistics for Alerts and Events from the UAB Data Set.

	alerts	events
total	1423	322
mean	118.6	26.8
st. dev.	155.1	12.4
max	462	52
min	12	10

effective in reducing pattern glut while preserving potentially interesting findings for expert evaluation.

4.6.4 Event Sets and Descriptive Specificity

Event sets ensure that the events presented to the user are as specific as possible according to the data processed. For example, let us consider the following real event (4.2) from the analysis of the UAB data set.

{NP_GNR, nosocomial} ⇒ {R~Piperacillin, R~Ceftazidime, R~Gentamicin} (4.2)
[]9609: 0/180 | []9610: 2/124 | []9611: 1/130 | *9612: 6/100 |

How do we know that the increase in piperacillin, ceftazidime, and gentamicin resistance did not occur only amongst nosocomial *Klebsiella pneumoniae* isolates? After all, these are nosocomial, non-*Pseudomonas* gram-negative rods. Simply put, if nosocomial *Klebsiella pneumoniae* isolates were responsible for the increase in piperacillin, ceftazidime, and gentamicin resistance in NP_GNRs, then (4.2) would have been captured by the alert with association rule {*Klebsiella pneumoniae*, NP_GNR, nosocomial} ⇒ {R~Piperacillin, R~Ceftazidime, R~Gentamicin}, and this alert, not (4.2), would have been presented to the user. Therefore, since an event with rule

{*Klebsiella pneumoniae*, NP_GNR, nosocomial} ⇒ {R~Piperacillin, R~Ceftazidime, R~Gentamicin} was not presented, we can be confident that *Klebsiella pneumoniae* isolates are not responsible for (4.2). An easy way to verify this is to look at the original data. In this case, from partition 9612 of the UAB data set, of the 6 NP_GNR, nosocomial isolates, resistant to piperacillin, ceftazidime, and gentamicin, 3 are *K. pneumoniae*, one is *E. coli*, one is *E. cloacae*, and 1 is *M. morgannii*. By similar logic, we can also be certain that the isolates of (4.2) are not all from the same location. Again, checking the data to verify this, we find that 2 are from the NICU, 1 from J10, 1 from J5, 1 from W9NW, and 1 from BMT. Whether the isolates are related in other ways, e.g., the service to which the patient was assigned, can not be known from looking at the original data set because these attributes were not available at the time of analysis. However, if they had been, then DMSS could have used them to construct events that are even more specific. Although not done in this case, (4.2) could be investigated by traditional means in more detail to determine if other possible associations exist.

In the next chapter, we present the culmination of everything presented up to this point–experimental results from 2 real-world epidemiologic data sets.

CHAPTER 5

EXPERIMENTAL RESULTS

The value of any data mining project is ultimately determined by whether or not new and interesting patterns are discovered. In this chapter, we present the fruits of our data mining labors – the patterns generated by DMSS in the analysis of the UAB data set, and the CDC *Streptococcus pneumoniae* data set. Notably different from other data mining analyses are the sizes of these data sets. Whereas traditional data mining focuses on very large data sets that are megabytes to terabytes in size, our data sets are substantially smaller. As we will show, however, they contain a wealth of previously unknown, complex, and interesting patterns. One could call this *data mining in the small*, but small, in this case, is interesting and important.

5.1 Introduction

The impact of nosocomial (hospital-acquired) infections on health care can hardly be overstated. Each year in the United States, nosocomial infections affect 2 million patients, cost more than $4.5 billion, and account for half of all major hospital complications (Centers for Disease Control 1992). Even more alarming is that amongst nosocomial infections, the number of drug-resistant infections has reached unprecedented levels (Goldman et al. 1996). Vancomycin-resistant enterococci, extended beta-lactamase producing gram-negative rods, and multi-drug resistant tuberculosis are but a few examples of emerging, highly-resistant bacteria that now cause significant morbidity

and mortality. While the biology and epidemiology of bacterial drug resistance is complex, the overuse of antimicrobials by both hospital and community physicians is largely responsible for the problems that exist today (Schlaes et al. 1997; Gold and Moellering 1996).

Bacterial resistance is a global problem, but like most large-scale problems, its origins are local. Microenvironments, especially hospitals and hospital intensive care units, are usually where resistant organisms originate and propagate, only to spread to larger environments as opportunity provides (Jones 1992; Koontz 1992; Neu et al. 1991; Schlaes et al. 1997). Therefore, early recognition of emerging problems requires proactive surveillance at the hospital and sub-hospital levels (Jones 1992; Koontz 1992; Neu et al. 1991; Schlaes et al. 1997). Unfortunately, most hospital surveillance efforts are passive; if no one suspects a problem, it goes undetected. Additionally, active surveillance for trends in bacterial resistance usually consists of yearly, hospital-wide summaries that are compiled in a table of susceptibility results with one entry per organism/drug combination. These summaries are not timely and often mask emerging, complex problems within the hospital (Neu et al. 1992). Consequently, it has been widely recognized that sophisticated, active, and timely intra-hospital surveillance is needed (Neu et al. 1991; Schlaes et al. 1997).

One source of surveillance data within the hospital is the antibiotic susceptibility data from the clinical microbiology laboratory (Schlaes et al. 1997). Such data, if carefully analyzed, can be used to identify local outbreaks. Extensive analysis of these data sets, however, requires considerable time and resources, both of which few hospital

epidemiologists have. Consequently, these data sets are underutilized and the patterns they contain go undiscovered. DMSS can be used to address this problem.

5.2 The UAB Data Set

The UAB data set, first described in Section 1.4.3, is 15 monthly partitions (September 1996 through November 1997) of data obtained from the clinical microbiology laboratory information system at UAB hospital. Each record of the data set describes a single bacterial isolate and contains items for the attributes listed in Section 1.4.3.1.

DMSS analysis of the data set was conducted with frequent set support threshold (FSST) 3, rule support threshold (RST) 8, the association rule templates in Figure 9, and the windowing schedule in Figure 14. The partitions were processed sequentially and a search for potentially interesting patterns was conducted after each of the December 1996 through November 1997 partitions was processed. These searches resulted in 12 sets of events, one for each search. The sequential nature in which these searches were performed simulates real-time surveillance in which emerging patterns are searched for at the end of each month. Each event in each set was then evaluated by a domain expert for interestingness.

As described in Section 1.4.3.1, the data set was seeded with records describing a nosocomial outbreak of *Acinetobacter baumanni* that occurred in 1994.

Figures 20, 21, 22, and 23 summarize some characteristics of the UAB data set and its analysis. Importantly, no more than 52 events (April 1997) were generated by any

search (Figure 22). Consequently, the reviewer was easily able to inspect each set of events in less than a half-hour.

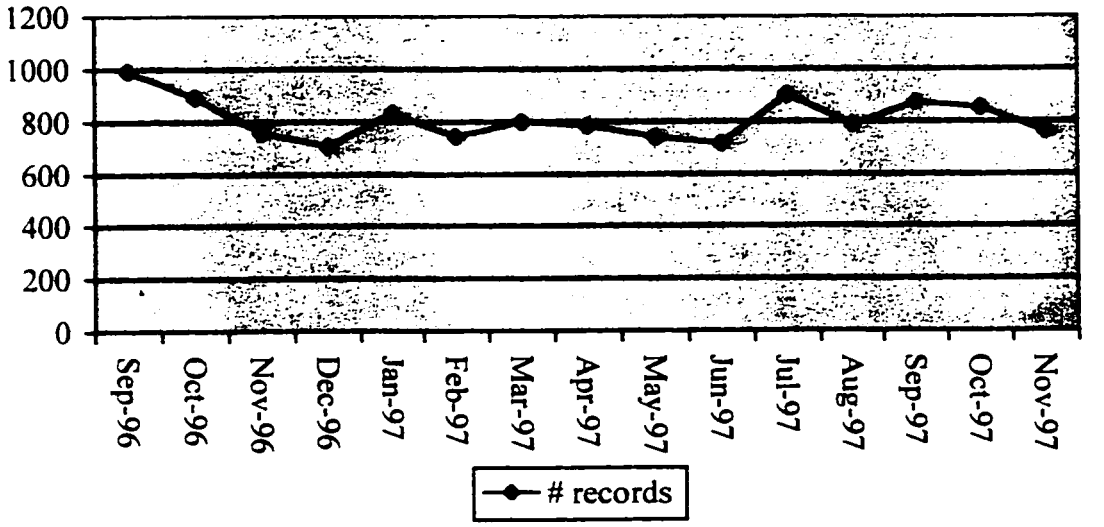


Figure 20: Sizes of the UAB partitions.

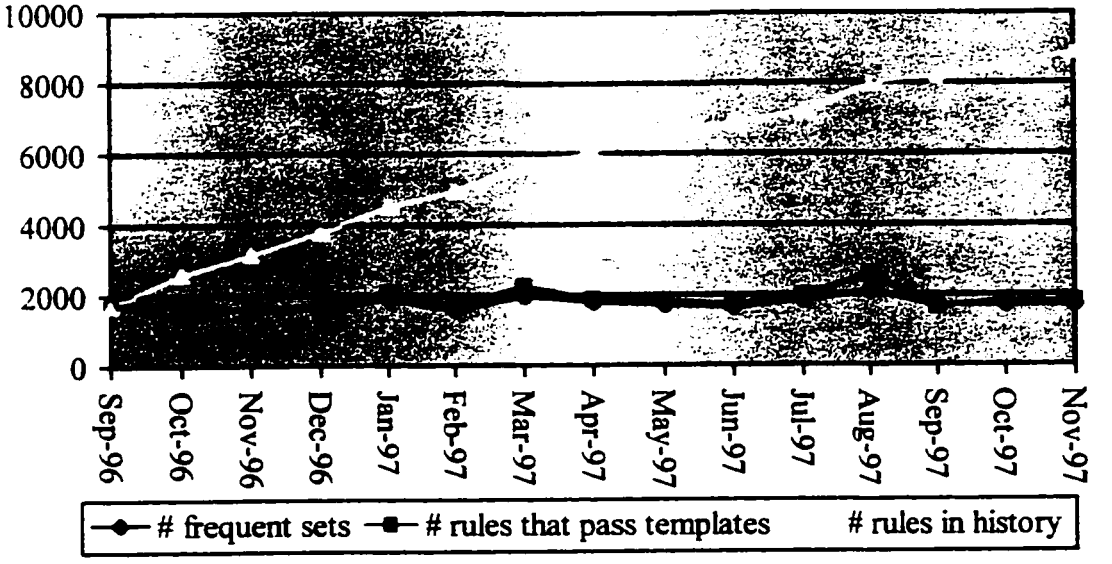


Figure 21: Numbers of frequent sets and rules generated for the UAB data set.

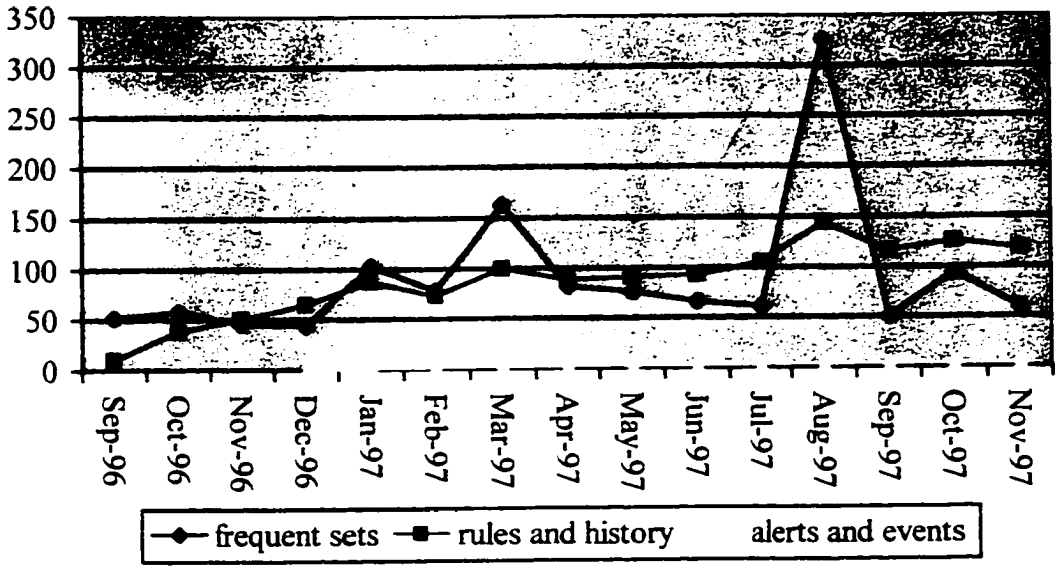


Figure 22: DMSS running times in seconds for the UAB data set.

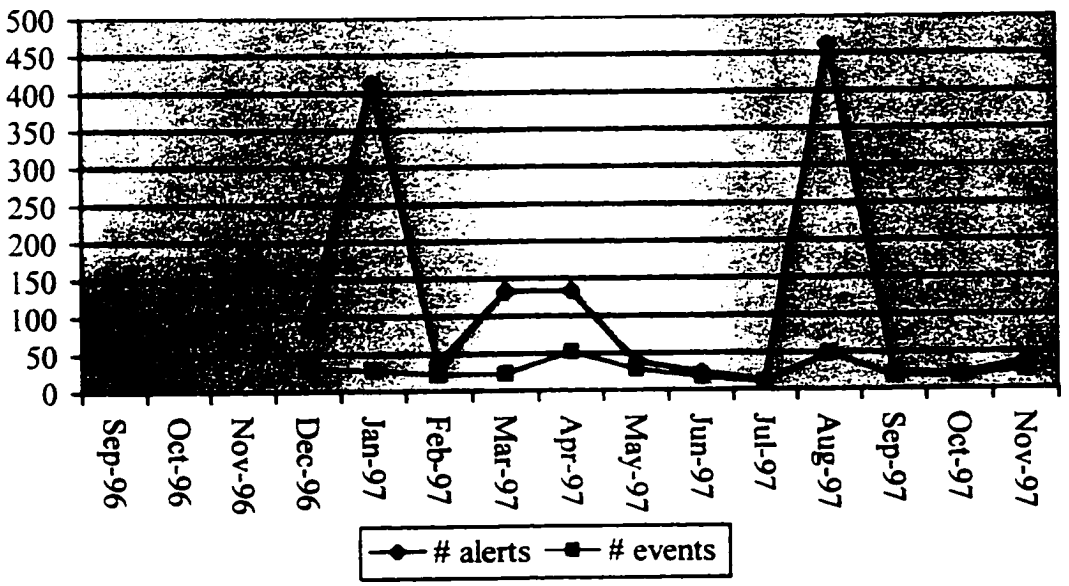


Figure 23: Numbers of alerts and events generated for the UAB data set.

5.2.1 Interesting Events

An event can be interesting for a number of reasons. Some properties of an interesting event are:

1. It is unexpected.
2. The outcome is nosocomial or is within a nosocomial group.
3. It is about a problematic organism.
4. The outcome or group is specific to a hospital unit or location.
5. The outcome contains antimicrobial resistance items, especially those for broad-spectrum antibiotics.

Criteria 2-5 are domain specific. Criterion 1, however, is a general criterion for interestingness; it is also usually a necessary one.

All events presented from this point on have the following format:

$A \Rightarrow B$
[]yymm: n/m | []yymm: n/m | ... | []yymm: n/m | *yymm: n/m | ... | *yymm: n/m |
p = xx rel diff = xx

where "$A \Rightarrow B$" is the association rule, "[]yymm" is the year and month of a partition in w_p, "*yymm" is the year and month of a partition in w_c, "n/m" is an incidence proportion, "p" is the p-value from the test of two proportions, and "rel diff" is the relative difference between the cumulative incidence proportions computed from w_p and w_c. For example, event (5.1):

{NP_GNR, nosocomial} ⇒ {R~Piperacillin, R~Ceftazidime, R~Gentamicin}
[]9609: 0/180 | []9610: 2/124 | []9611: 1/130 | *9612: 6/100 | (5.1)
p = 0.004 rel diff = 8.68

This event describes an increase in the incidence of piperacillin, ceftazidime, and gentamicin resistance in nosocomial, non-*Pseudomonas* gram-negative rod isolates from

September, October, November, and November of 1996 to December of 1996. A p-value of 0.004 indicates that the difference between the corresponding cumulative incidence proportions is statistically extreme, and a relative difference of 8.68 tells us that there was almost a nine-fold increase in the cumulative incidence proportion from w_p to w_c.

Let us take this moment to briefly review how DMSS arrives at an event by using (5.1) as an example. In processing partition 9612, DMSS generated the frequent set {NP_GNR, nosocomial, R~Piperacillin, R~Ceftazidime, R~Gentamicin} with support 6. Then, it created all high-support association rules for that frequent set. Of those association rules, {NP_GNR, nosocomial} $\Rightarrow$ {R~Piperacillin, R~Ceftazidime, R~Gentamicin} passed the rule templates in Figure 9 and therefore was used to update the history. Since the rule was not already in the history, DMSS added the rule to the history, then queried the original database to get the incidence proportions of the rule in the previous 3 partitions. It then added these prior incidence proportions along with current incidence proportion to the history. In the search for interesting patterns performed after partition 9612 was processed, DMSS identified an extreme change in the incidence proportion of {NP_GNR, nosocomial} $\Rightarrow$ {R~Piperacillin, R~Ceftazidime, R~Gentamicin} between w_p = {9609, 9610, 9611} and w_c = {9612} and so generated an alert identical to (5.1). Then, in constructing event sets for all alerts discovered in the search, DMSS identified this alert as the event of an event set. As such, it was presented to the user as a potentially interesting finding.

From the 322 events generated in the entire analysis, Appendix B contains the 41 that were interesting to the domain expert. Each event is accompanied by a short description of why it was selected.

Interesting events, such as those in Appendix A, can be investigated in a number of ways. Any in-depth investigation should begin by carefully looking over the original data that corresponds to the event. For example, (5.1) which is also event 1 in Appendix A, indicates an extreme increase in the incidence in piperacillin, ceftazidime, and gentamicin resistance amongst nosocomial non-*Pseudomonas* gram-negative rods. Inspection of the data reveal that three *K. pneumoniae*, one *E. coli*, one *E. cloacae*, and one *M. morgannii* comprise the 6 isolates from partition 9612. One *K. pneumoniae* and the *M. morgannii* were from the NICU, and the other isolates were from J10, J5, W9NW, and BMT. In this case, nothing else seems suspicious about the organisms or locations.

Highlights from the events in Appendix A include a number of potential nosocomial outbreaks of specific organisms, including the seeded *Acinetobacter* outbreak from 1994 (Table 16) and trends in nosocomial, multiple-organism antimicrobial resistance (Table 17).

5.3 The CDC Data Set

DMSS is also useful in the analysis of public health surveillance data. In this section, we present results from a preliminary analysis of 15 months (January 1995 – March 1996) of *Streptococcus pneumoniae* data received from the Centers for Disease Control and Prevention (CDC).

Since the late 1980's, drug-resistant *Streptococcus pneumoniae* (DRSP) has been emerging problem in the United States (Cetron et al. 1997; Gold and Moellering 1996).

The goal of DRSP surveillance is to monitor the prevalence and geographic distribution of DRSP and to rapidly recognize outbreaks and new patterns of resistance

Table 16: An Index of UAB Events that Describe Possible Nosocomial Outbreaks of Disease.

Organism	Index of event in Appendix A	Comment
Acinetobacter baumannii	14	known outbreak from 1994 (seeded)
	31	
Citrobacter freundii	9	
Enterobacter aerogenes	32	
Enterobacter cloacae	24	
	39	SICU
Klebsiella oxytoca	15	
Klebsiella pneumoniae	5	
	30	
Proteus mirabilis	21	CCU
Serratia marcescens	35	NICU
Staph aureus	4	MRSA, W9NW
	25	MRSA, S9SW
Streptococcus pneumoniae	13	

Table 17: An Index of UAB Events that Describe Changes in Nosocomial Antimicrobial Susceptibilities.

Antimicrobials	Index of event in Appendix A	Comment
piperacillin, ceftazidime, gentamicin	1	
piperacillin	11	W7NW
cefazolin	16	MICU
	19	
ceftriaxone	37	SICU
piperacillin & pan-cephalosporin	8	SICU

(Cetron et al. 1997). To this end, the CDC collects data on invasive pneumococcal isolates from approximately 15 hospital laboratories around the United States. The data set contains demographic attributes such as county, hospital, patient's zip code, race, and

ethnicity, as well as attributes describing infection outcome, serotype, and antibiogram, amongst others. CDC generously sent us 15 months of data for analysis. All data received is coded to keep anonymous the identity of states, counties, hosptials, and zip codes.

In general, this data set is not as clean or as timely as the UAB data set. Some records have missing data that should not be missing. For example, the bacteremia and meningitis fields are binary and one or both should be true for an infection to be invasive and, therefore, get included in the data set; some records have missing values for both. Additionally, all DRSP reporting to the CDC is currently voluntary. This delays the arrival of some data and perhaps makes for an incomplete and biased data set. Nationwide, consistent, timely, and mandatory reporting of invasive isolates is clearly desirable (Cetron et al. 1997). Until then, however, we make due with what we have.

Also limiting our analysis was our lack of communication with the CDC. As a result, we were not able to compare our findings to theirs, an exercise that would have been valuable. In any case, we believe that our results, while "raw," demonstrate DMSS's ability to find interesting, emerging patterns in public health surveillance data.

5.3.1 Analysis

DMSS analysis was performed with a frequent set support threshold of 3, a rule support threshold of 8, a p-value threshold of 0.01, a relative difference threshold of 1, the association rule templates in Figure 24, and the windowing schedule in Figure 25. The 15 partitions were processed sequentially and a search for potentially interesting patterns was conducted after each of the April 1995 through March 1998 partitions was

processed. These searches resulted in 12 sets of events, one for each search. The sequential nature in which these searches were executed simulates real-time surveillance in which emerging patterns are searched for at the end of each month. Although this data set was not collected in a timely enough manner for monthly searches, future data sets assembled by more timely and sophisticated surveillance systems could be analyzed monthly. Figures 26, 27, 28, and 29 summarize characteristics of the CDC data set and its analysis. On average, 76,741 association rules were generated for each of the 15 partitions, 1,793 of which passed the rule templates in Figure 24. Therefore, only 2% of

1. include: EmptySet $\Rightarrow$ State* $\vee$ R~*
2. include: $\neg$ EmptySet $\Rightarrow$ **
3. exclude: R~* $\vee$ I~* $\vee$ Antbgram* $\Rightarrow$ **
4. exclude: Outcome* $\Rightarrow$ **
5. exclude: ** $\Rightarrow$ Outcome1
6. exclude: Bacteremia $\vee$ Pneumonia $\Rightarrow$ **
7. exclude: Hiv $\vee$ Aids $\Rightarrow$ Hiv $\vee$ Aids
8. exclude: Race* $\vee$ Ethnic* $\Rightarrow$ Race* $\vee$ Ethnic*
9. exclude: $\neg$ EmptySet $\Rightarrow$ State* $\vee$ Race* $\vee$ Sex* $\vee$ AgeGrp*

Figure 24: Association rule templates used in the analysis of the CDC data set.

1. (3, 1)
2. (6, 2)
3. (9, 3)

Figure 25: The windowing schedule for the analysis of the CDC data set.

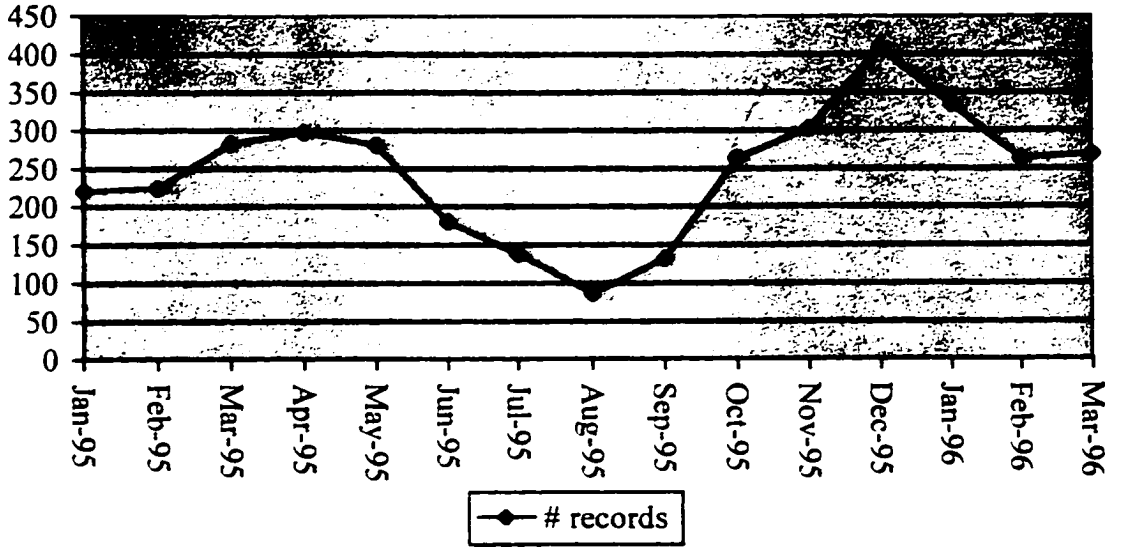


Figure 26: Sizes of the CDC partitions.

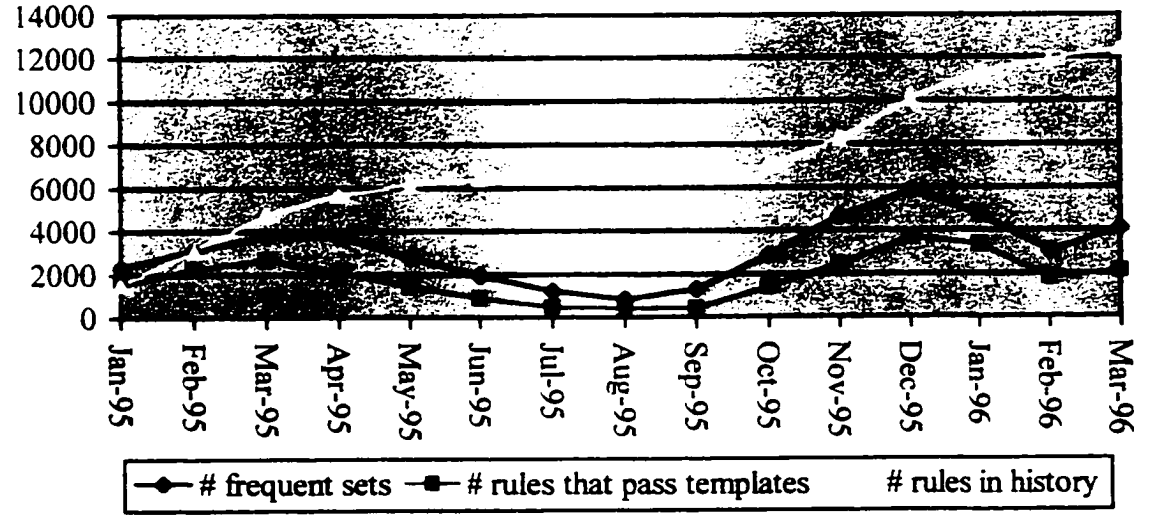


Figure 27: Numbers of frequent sets and rules generated for the CDC data set.

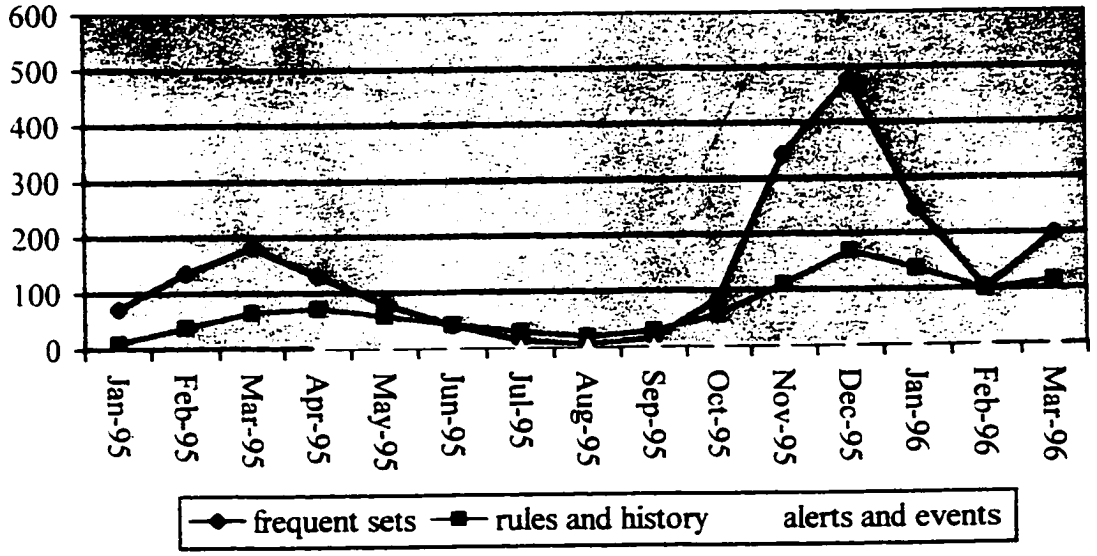


Figure 28: DMSS running times in seconds for the CDC data set.

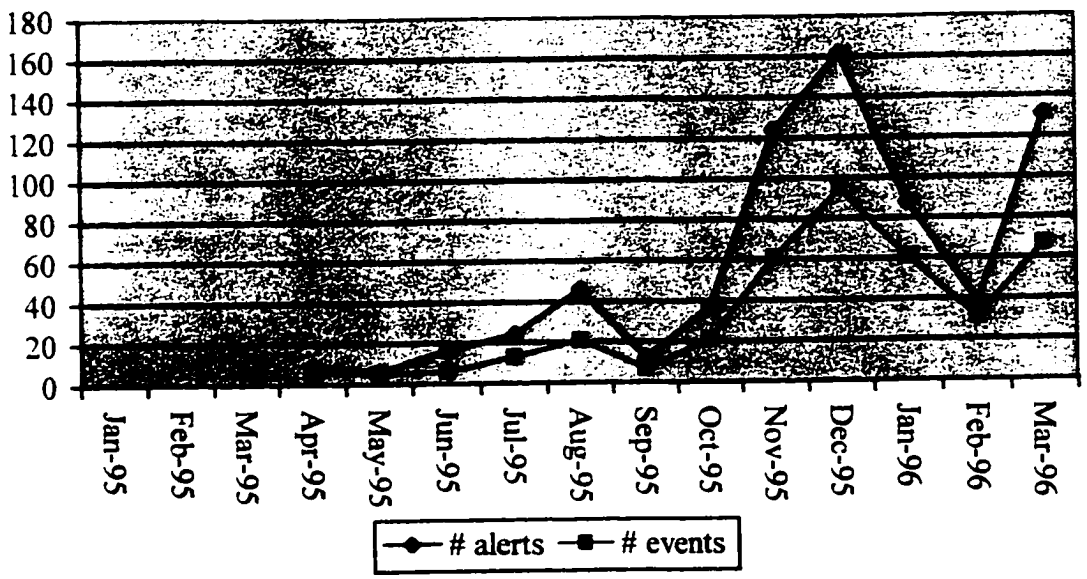


Figure 29: Numbers of alerts and events generated for the CDC data set.

the rules generated for each data partition passed the rule templates to get included in the history. If, as a result, the history was 2% the size it would have been if all rules were included, and the number of events generated by a given search is proportional to the size of the history at the time of the search, then the templates in Figure 24 reduce pattern glut in the CDC analysis by about 98%. This may seem extreme, but in our experience, most of the rule space is truly uninteresting.

Again, as discussed in Section 3.2, specifying rule templates requires considerable care because templates that include too many rules in the history will lead to pattern glut while those that exclude too many rules will lead to no interesting findings. In the analysis of the CDC data set, as in the analysis of the UAB data set, we have found the iterative strategy described in Section 3.2 useful in arriving at an appropriate set of association rule templates.

5.3.2 Interesting Events

Forty-nine interesting events from the analysis are given in Appendix B. Most of the interesting events are location-specific. They describe possible state, county, hospital, and zip-code outbreaks of invasive disease. Of these, some describe outbreaks amongst specific ethnic, age, and race groups, some describe possible serotype-specific outbreaks, and others describe possible outbreaks of DRSP. In all, Appendix B contains some very interesting events. For example, events 4 and 7 of Appendix B describe a possible outbreak of invasive pneumococci amongst infants in hospital AA015. An index to some events in Appendix B is given in Table 18.

5.4 Inter-Seasonal Analysis

Unlike the UAB data set, the CDC data set has a clear seasonal component. The number of invasive *Streptococcus pneumoniae* isolates is relatively high in the early

Table 18: An Index of CDC Events.

	Indeces in Appendix B
zip code specific events	25, 28, 29, 31, 39, 43
hospital specific events	4, 7, 8, 9, 13, 26, 32, 34, 39, 40, 42, 46, 48
serotype specific events	11, 15, 18, 20, 21, 24, 35, 41
DRSP events	11, 16, 17, 30, 45, 47, 49

winter months and relatively low in the summer months (Figure 23). Typically with seasonal data, it is customary to analyze trends between like seasons of successive years so that inter-seasonal trends can be identified.

Inter-seasonal analysis can be accomplished by DMSS. In such an analysis, each partition should contain data from an entire season in a given year, and successive partitions should contain data from the same season in successive years. Then user-defined parameters should be set appropriately. To do an inter-seasonal analysis requires at least several years of data. Since this much data was not available to us for either the UAB or CDC data sets, we were unable to do an inter-seasonal analysis for either domain.

5.5 Processing Larger Data Sets

In the CDC data set, the attribute of largest geographic location is "state" (e.g. Louisiana), and the attributes of smallest geographic location are "hospital" and "ZIP code." If we require that all interesting events contain location information, then we need to estimate the incidence of such events that would constitute an outbreak at the smallest and largest locations in the data set. In the analysis of the CDC data set, we require that an event occur at least 3 times in a hospital or ZIP code in a single partition to be included in the history and monitored for interesting changes in time. Consequently, to detect an outbreak of events of at least 3 cases in a calendar month in a hospital or ZIP code, we used an FSST of 3 for processing each partition.

For a state-wide outbreak, i.e., one that is state-specific but not specific to smaller geographic areas, we may want to see at least 12 cases of an event in a single partition. The frequent set support threshold for such an analysis could therefore be set to 12. This means that in the analysis of the CDC data set, we could have looked for state-wide events by rerunning the entire analysis with a higher FSST. If an outbreak of 12 cases occured in one county or ZIP code, then DMSS would still generate a county or ZIP code specific event. It would not, however, detect an outbreak of 6 cases within a county or ZIP code.

An estimation of outbreak size is relevant because for data sets larger than the CDC data set, generating low-support frequent sets becomes prohibitive. For such data sets, several analyses should be done with different subsets of the data and different FSSTs. For example, if the CDC data set contained just 5 times the number of records per partition, DMSS would not be able to efficiently generate frequent sets with an FSST

of 3. This does not mean, however, that we could not do an analysis. Since the attribute of largest geographic location is "state," we could run an analysis of the entire data set with FSST = 12 to detect state-wide outbreaks. If 12 was still too low, then we could break the data set up into pieces, one piece per region (several states) and run the analysis with FSST=12 on a time series of partitions from each region. We would still detect all state-wide outbreaks of events that occurred 12 times per partition, but we would have to run a separate analysis for each region. This way, we distribute the work by running a separate analysis for each of region. Assuming an FSST = 12 is reasonable for analyzing the entire data set, we could then break the data set up by "state" and run a more detailed analysis (i.e., lower FSST) for each state. In each of these state analyses, we want to find county, ZIP code, and hospital outbreaks. Therefore, we could try the analysis with FSST = 3. If this were too small, then we could analyze the state data with an FSST of say 6 to search for county outbreaks, then break up a county's data into smaller subsets to look for zip-code and hospital-specific outbreaks.

In summary, we believe that a strategy of recursively splitting a data set and analyzing each smaller piece with a lower FSST would be effective for analyzing larger data sets.

CHAPTER 6

THE FUTURE OF DATA MINING AND EPIDEMIOLOGIC SURVEILLANCE

As data-intensive health information systems of the future are created, epidemiologists will need new tools to help them more efficiently utilize the data that are collected. In this dissertation, we have constructed a link between data mining and epidemiologic surveillance through the development, implementation, and application of the Data Mining Surveillance System (DMSS). Using DMSS to analyze the UAB data set, we demonstrated that DMSS could efficiently find complex and emerging patterns of nosocomial disease by examining clinical laboratory data. With the CDC data set, we demonstrated the potential for using DMSS in the analysis of public health surveillance data.

New and sophisticated analytical tools are needed in both public health and hospital epidemiology surveillance. As described by Dean et al. (1994), the ideal public health surveillance system of the future will include analysis tools that automatically identify, on different time and geographical scales, unusual and interesting patterns from time-slices of raw data. DMSS is a representative of the first generation of these tools. Likewise, in hospital epidemiology, the infection control systems of the future will require efficient and timely recognition of trends in nosocomial infection and antimicrobial resistance (Schlaes et al. 1997). Systems like DMSS, therefore, will be needed.

We hope that the work presented here is only the beginning of the use of data mining in epidemiologic surveillance. While we believe it is a good start, many issues need more research. For example, data mining is clearly a human-centered process. With this in mind, how can we improve the interaction between DMSS and the user or a group of users? A more efficient interaction would be especially valuable for defining association rule templates. Once templates are satisfactory, i.e., useful results are generated, can they be made to adapt to the changing perceptions of users over time?

Other areas for research include better, perhaps automated, ways to analyze large data sets (Section 5.5), more intuitive information presentation, developing a distributed version of the clone algorithm, and increasing the expressiveness of association rule templates and windowing schedules.

Other strategies for detecting trends and outbreaks in time-series data could be investigated. These include cumulative sums, log-linear regression, the scan statistic, and shape templates. Would any of these be more effective than tests of two proportions for detecting certain trends? Could DMSS use one or more of these at a time?

A fertile area of research is the use of maps or graphs for detecting geographic outbreaks of disease. While DMSS can currently identify geographic patterns, it can only do so to the extent that the geography can be represented in a taxonomy. A significant improvement would allow DMSS to utilize maps (e.g., a floor plan of a hospital, or a ZIP code graph) to identify outbreaks that cover contiguous or related geographic units.

Utilization research issues are plentiful. For example, comprehensive hospital and public health surveillance will need to utilize automated analysis systems such as DMSS. To do so, what data are needed for the types of surveillance desired, and how do

we collect it and clean it in a timely fashion? How will the results be used to make proactive policy decisions, and how are those decisions going to be implemented to change current practice? The answers to these questions will require new paradigms for both public health and hospital epidemiology.

We hope that the work described in this dissertation is a start of a new relationship between data mining and epidemiologic surveillance.

LIST OF REFERENCES

Agrawal, R.; Imielinski, T.; and Swami, A. 1993. Mining Association Rules Between Sets of Items in Large Databases. In Proceedings of the ACM SIGMOD Conference on Management of Data, 207-216. Reading, MA: ACM Press.

Agrawal, R.; and Psaila, G. 1995. Active Data Mining. In Proceedings of the First International Conference on Knowledge Discovery and Data Mining, 3-8. Cambridge: MIT Press.

Agrawal, R.; and Srikant, R. 1994. Fast Algorithms for Mining Association Rules. In Proceedings of the 20th International Conference on Very Large Data Bases, 487-499. New York: Morgan Kaufmann.

Anand, T.; and Kahn, G. 1992. SPOTLIGHT: A Data Explanation System. In Proceedings of the Eigth IEEE Conference on Aplied AI, 2-8. Piscataway, NJ: IEEE Press.

Armitage, P. 1971. *Statistical Methods in Medical Research*. New York: John Wiley and Sons.

Berndt, D. J.; and Clifford, J. 1996. Finding Patterns in Time-Series: A Dynamic Programming Approach. In *Advances in Knowledge Discovery and Data Mining*, eds. U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 229-248. Menlo Park: AAAI Press.

Blanchard, D. 1994. News Watch. *AI Expert* 7: 3.

Brin, S.; Motwani, R.; Ullman J.D.; and Tsur, S. 1997. Dynamic itemset counting and implication rules for market-basket data. In Proceedings of the ACM SIGMOD Conference on Management of Data, 255-263. Reading, MA: ACM Press.

Brossette, S. E.; Sprague, A. P.; Hardin, J. M.; Waites, K. B.; Jones, W. T.; and Moser, S. A. 1998. Association rules and data mining in hospital infection control and public health surveillance. *Journal of the American Medical Informatics Association*. Forthcoming.

Brownlee, K. A. 1965. *Statistical Theory and Methodology* (2nd ed.). Malabar, FL: Robert E. Kriegler Publishing Co., Inc.

Buehler, J. W. 1997. Surveillance. In *Modern Epidemiology* (2nd ed.), eds. K. J. Rothman and S. Greenland, 435-457. Philidelphia: Lippincott-Raven.

Centers for Disease Control. 1992. Public health focus surveillance: prevention and control of nosocomial infections. *Morbidity and Mortality Weekly Report* 41: 783-787.

Cetron, M. C.; Butler, J.; Jernigan, D.; Alexander, M.; Roush, S.; and Brieman, R. 1997. Pneumococcal disease. In *Manual for the Surveillance of Vaccine-Preventable Diseases*, eds. M. Wharton and S. Roush, 91-97. Atlanta: Centers for Disease Control and Prevention.

Dean, A. G.; Fagan, R. F.; and Panter-Connah, A. H. 1994. Computerizing Public Health Surveillance Systems. In *Principles and Practice of Public Health Surveillance*, eds. S. M. Teutsch and R. E. Churchill, 200-217. New York: Oxford University Press.

Ederer, F.; Myers, M. H.; and Mantel, N. 1964. A Statistical Problem in Space and Time: Do Leukemia Cases Come in Clusters? *Biometrics*, September, 626-638.

Elder, J.; and Pregibon, D. 1996. A Statistical Perspective on Knowledge Discovery in Databases. In *Advances in Knowledge Discovery and Data Mining*, eds. U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 83-113. Menlo Park: AAAI Press.

Farrington, C. P.; Andrews, N. J.; and Beale, A. D. 1996. A Statistical Algorithm for the Early Detection of Outbreaks of Infectious Disease. *Journal of the Royal Statistical Society A* 159: 547-563.

Fayyad, U. M.; Piatetsky-Shapiro, G.; and Smyth, P. 1996. From Data Mining to Knowledge Discovery: An Overview. In *Advances in Knowledge Discovery and Data Mining*, eds. U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 1-34. Menlo Park: AAAI Press.

Fleiss, J. L. 1973. *Statistical Methods for Rates and Proportions*. New York: John-Wiley and Sons.

Fleiss, J. L. 1986. Significance Tests Have a Role in Epidemiologic Research: Reactions to A. M. Walker. *American Journal of Public Health* 76: 559-560.

Frawley, W. J.; Pietetsky-Shapiro, G.; and Matheus, C. J. 1992. Knowledge Discovery in Databases: An overview. *AI Magazine* 13(3): 57-70.

Gold, H. S.; and Moellering, R. C. 1996. Antimicrobial-Drug Resistance. *The New England Journal of Medicine* 335: 1445-1453.

Goldman, D. A.; Weinstein, R. A.; and Wenzel, R. P.; et al. 1996. Strategies to Prevent and Control the Emergence and Spread of Antimicrobial-Resistant Microorganisms in Hospitals. *Journal of the American Medical Association* 275: 234-240.

Gordon, T.; Moore, F.E.; and Shurtleff, D. 1959. Some Methodologic Problems from the Long-Term Study of Cardiovascular Disease: Observations on the Framingham Study. *Journal of Chronic Diseases* 10: 186-206.

Greenland, S. 1990. Randomization, Statistics, and Causal Inference. *Epidemiology* 1: 421-429.

Grimson, R. C. 1993. Disease Clusters, Exact Distributions of Maxima, and p-Values. *Statistics in Medicine* 12: 1773-1794.

Hall, J.; Mani, G.; and Barr, D. 1996. Applying Computational Intelligence to the Investment Process. In Proceeding of CIFER-96: Computational Intelligence in Financial Engineering, 10-18. Piscataway, NJ: IEEE Press.

Holfacker I. L.; Huynen M. A.; Stadler, P. F; and Stolorz, P. E. 1996. Knowledge Discovery in RNA Sequence Families of HIV Using Scalable Computers. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 20-25. Menlo Park: AAAI Press.

Hutwagner, C.; Maloney, E. K.; Bean, N. H.; Slutsker, L.; and Martin, S. 1997. Using Laboratory-Based Surveillance Data for Prevention: An Algorithm for Detecting Salmonella Outbreaks. *Emerging Infect Diseases* 3: 395-400.

IBM Advanced Scout (1995). Data Mining: Advanced Scout [ONLINE]. Available: http://www.research.ibm.com/scout/ [1997, April 26].

John, G. H. 1997. *Enhancements to the Data Mining Process*. Ann Arbor: UMI.

Jones, R. N. 1992. The Current and Future Impact of Antimicrobial Resistance Among Nosocomial Bacterial Pathogens. *Diagnostic Microbiology and Infectious Disease* 15: 3s-10s.

Kheifets I. K. 1993. Cluster Analysis: A Perspective. Statistics in Medicine 12: 1755-1756.

Klemettinen, M.; Mannila, H.; Ronkainen, P.; Toivonen, H.; and Verkamo, I. 1994. Finding Interesting Rules from Large Sets of Association Rules. In Proceedings of the Third International Conference on Information and Knowledge Management, 401-407. New York: ACM Press.

Koontz, F. P. 1992. A Review of Traditional Resistance Surveillance Methodologies and Infection Control. *Diagnostic Microbiology and Infectious Disease* 15: 43s-47s.

Matheus, C. J.; Piatetsky-Shapiro, G.; and McNeill, D. 1996. Selecting and Reporting what is Interesting: The KEFIR Application to Healthcare Data. In *Advances in Knowledge Discovery and Data Mining*, eds. U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 495-516. Menlo Park: AAAI Press.

Naus, J. 1966. Some Probabilities, Expectations, and Variances for the Size of the Smallest Intervals and the Largest Clusters. *Journal of the American Statistical Association* 61: 1191-1199.

NCCLS. 1997. *Methods for Dilution Antimicrobial Susceptibility Tests for Bacteria that Grow Aerobically* (4th ed.). Wayne, PA: NCCLS.

Neu, H. C.; Duma, R. J.; Jones, R. N.; et al. 1992. Antibiotic Resistance: Epidemiology and Therapeutics. *Diagnostic Microbiology and Infectious Disease* 15: 53s-60s.

Piatetsky-Shapiro, G. 1991. Discovery, Analysis, and Presentation of Strong Rules. In *Knowledge Discovery in Databases*, eds. G. Piatetsky-Shapiro, and W. Frawley, 229-248. Menlo Park: AAAI Press.

Poole, C. 1987. Beyond the Confidence Interval. *American Journal of Public Health* 77: 195-199.

Prather, J. C.; Lobach, D. F.; Goodwin, L. K.; Hales, J. W.; Hage, M. L.; and Hammond, W. E. 1997. Medical Data Mining: Knowledge Discovery in a Clinical Data Warehouse. In Proceedings of the 1997 American Medical Informatics Association Annual Fall Symposium, 101-105. Bethesda: AMIA.

Rosner, B. 1990. *Fundamentals of Boistatistics* (3rd ed.). Belmont: Duxbury Press.

Rothman, K. J. 1990. No Adjustments are Needed for Multiple Comparisons. *Epidemiology* 1: 43-46.

Rothman, K. J.; and Greenland, S. 1997. *Modern Epidemiology* (2nd ed.). Philadelphia: Lippincott-Raven.

Savasare, A.; Omiecinski, E.; and Navathe S. 1995. An Efficient Algorithm for Mining Association Rules in Large Databases. In Proceedings of the 21st Very Large Data Base Conference, 432-444. New York: Morgan Kaufmann.

Schlaes, D. M.; Gerding, D. N.; and John, J. F.; et al. 1997. Society for Healthcare Epidemiology of America and Infectious Diseases Society of America Joint Committee on the Prevention of Antimicrobial Resistance: Guidelines for the Prevention of Antimicrobial Resistance in Hospitals. *Clinical Infectious Diseases* 25: 584-598.

Shek, E. C.; Muntz, R. R.; Mesrobian, E.; and Ng, K. 1996. Scalable Exploratory Data Mining of Distributed Geoscientific Data. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 32-37. Menlo Park: AAAI Press.

Smith, D.; and Neutra, R. 1993. Approaches to Disease Cluster Investigations in a State Health Department. *Statistics in Medicine* 12: 1757-1762.

Srikant, R.; Vu, Q.; and Agrawal, R. 1997. Mining Association Rules with Item Constraints. In Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, 67-73. Menlo Park: AAAI Press.

Thacker, S. B. 1994. Historical Development. In *Principles and Practice of Public Health Surveillance*, eds. S. M. Teutsch and R. E. Churchill, 3-17. New York: Oxford University Press.

Thompson, W. D. 1987. Statistical Criteria in the Interpretation of Epidemiologic Data. *American Journal of Public Health* 77: 191-194.

Teutsch, S. M.; and Churchill, R. E., eds. 1994. *Principles and Practice of Public Health Surveillance*. New York: Oxford University Press.

Tsai, Y.; King, P. H.; Higgins, M. S.; Pierce, D.; and Patel, N. P. 1997. An Expert-Guided Decision Tree Construction Strategy: An Application in Knowledge Discovery with Medical Databases. In Proceeding of the 1997 American Medical Informatics Association Annual Fall Symposium, 208-212. Bethesda: AMIA.

Tsumoto, S.; and Tanaka, H. 1996. Incremental Learning of Probabilistic Rules from Clinical Databases Based on Rough Set Theory. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 198-202. Menlo Park: AAAI Press.

Tukey, J. W. 1977. *Exploratory Data Analysis*. New York: Addison-Wesley.

Walker, A. M. 1986. Reporting the results of epidemiologic studies. *American Journal of Public Health* 76: 556-558.

Wallenstein, S. 1980. A test for detection of clustering over time. *American Journal of Epidemiology* 111: 367-372.

APPENDIX A

SELECTED EVENTS FROM THE ANALYSIS OF THE UAB DATA SET

EVENTS FROM THE 9612 SEARCH

1. {NP_GNR, nosocomial} ⇒ {R~Piperacillin R~Ceftazidime R~Gentamicin}
   - []9609: 0/180 | []9610: 2/124 | []9611: 1/130 | *9612: 6/100 |
   - p = 0.004 rel diff = 8.68
   - Increased nosocomial NP_GNR resistance to three important gram-negative drugs. Not location or organism specific.

2. {NP_GNR, nosocomial} ⇒ {R~Piperacillin, R~Ceftazidime, R~Cefuroxime, R~Cefazolin R~Ceftriaxone}
   - []9609: 0/180 | []9610: 1/124 | []9611: 4/130 | *9612: 7/100 |
   - p = 0.005 rel diff = 6.076
   - Piperacillin and cephalosporin resistance up in nosocomial NP_GNR. Not location or organism specific.

3. {*Klebsiella pneumoniae*, NP_GNR} ⇒ {R~Cefotetan, R~Ceftazidime, R~Cefuroxime R~Cefazolin, R~Cephalothin, R~Ceftriaxone}
   - []9609: 0/79 | []9610: 1/79 | []9611: 4/85 | *9612: 8/76 |
   - p = 0.007 rel diff = 5.11579
   - Pan-cephalosporin resistance up in *Klebsiella pneumoniae.*

EVENTS FROM THE 9701 SEARCH

4. EmptySet ⇒ {LocW9NW, GPC, *Staphylococcus aureus*, R~AmoxicillinClK, R~Cefazolin, R~ErythromycinEst R~Cephalothin R~Clindamycin R~Oxacillin}
   - []9610: 0/894 | []9611: 0/760 | []9612: 0/709 | *9701: 4/834 |
   - p = 0.009 rel diff = Inf
   - Possible MRSA outbreak in W9NW.

5. {*Klebsiella pneumoniae*, NP_GNR, nosocomial} ⇒ {R~Piperacillin, R~Cefotetan, R~Ceftazidime, R~Cefuroxime, R~Cefazolin, R~Cotrimazole, R~Cephalothin, R~Gentamicin, R~Ceftriaxone}
   - []9610: 1/12 | []9611: 1/25 | []9612: 1/23 | *9701: 10/26 |
   - p = 0 rel diff = 7.69231
   - Possible clonal, nosocomial outbreak of highly-resistant *Klebsiella pneumoniae.*

EVENTS FROM THE 9702 SEARCH

6. {LocM8} ⇒ {NP_GNR, nosocomial}
   - []9611: 3/17 | []9612: 2/12 | []9701: 5/15 | *9702: 10/15 |
   - p = 0.002 rel diff = 2.9333
   - Incidence of NP_GNR isolation up in M8.

7. {LocHTIC} ⇒ {nosocomial, GPC, *Staphylococcus aureus*}
   - []9611: 0/16 | []9612: 0/16 | []9701: 0/12 | *9702: 4/15 |
   - p = 0.006 rel diff = Inf
   - Possible outbreak of nosocomial *S. aureus* in the HTIC.

8. {NP_GNR, nosocomial, LocSICU} ⇒ {R~Piperacillin, R~Cefotetan, R~Ceftazidime, R~Cefuroxime, R~Cefazolin, R~Cephalothin}
   - []9611: 0/21 | []9612: 1/15 | []9701: 1/24 | *9702: 5/15 |
   - p = 0.006 rel diff = 10
   - Cephelosporin and piperacillin resistance up in nosocomial, non-psuedomonas gram-negative rods in the SICU.

9. EmptySet ⇒ {NP_GNR, *Citrobacter freundii*, nosocomial, R~Piperacillin, R~Cefotetan, R~Ceftazidime, R~Cefuroxime, R~Cefazolin, R~Cephalothin, R~Ceftriaxone}
   - []9611: 0/760 | []9612: 0/709 | []9701: 1/836 | *9702: 5/742 |
   - p = 0.008 rel diff = 15.5323
   - Possible clonal outbreak of nosocomial *Citrobacter freundii.*

EVENTS FROM THE 9703 SEARCH

10. EmptySet ⇒ {NP_GNR, nosocomial, LocURO}
    - []9612: 0/709 | []9701: 1/836 | []9702: 2/744 | *9703: 7/798 |
    - p = 0.009 rel diff = 6.69298
    - Possible infection control breach in URO.

11. EmptySet ⇒ {nosocomial, R~Piperacillin, LocW7NW}
    - []9612: 0/709 | []9701: 0/836 | []9702: 0/744 | *9703: 4/798 |
    - p = 0.009 rel diff = Inf
    - Nosocomial piperacillin resistance up in W7NW.

12. {non-nosocomial, *Pseudomonas aeruginosa*} ⇒ {R~TicarcillinClavK}
    - []9612: 1/43 | []9701: 2/55 | []9702: 1/46 | *9703: 10/52 |
    - p = 0.001 rel diff = 6.92308
    - Ticarcillin resistance amongst community-acquired *P. aeruginosa.*

13. EmptySet ⇒ {nosocomial, *Streptococcus pneumoniae*, GPC, R~ErythromycinEst }
    - []9612: 0/709 | []9701: 0/836 | []9702: 1/744 | *9703: 6/798 |
    - p = 0.003 rel diff = 17.2105
    - Possible nosocomial outbreak of *Strep. pneumoniae.*

14. {NP_GNR, nosocomial} ⇒ {*Acinetobacter baumannii*, R~Piperacillin, R~Cefotetan, R~Ceftazidime, R~Cefuroxime, R~Cefazolin, R~Ampicillin, R~Cephalothin, R~Gentamicin, R~Tobramycin, R~Ciprofloxacin, R~Amikacin, R~Aztreonam, R~Mezlocillin, R~TicarcillinClav, R~Ofloxacin, R~Cefotaxime, R~Trimethoprim}
    - []9612: 0/100 | []9701: 0/121 | []9702: 0/108 | *9703: 5/106 |
    - p = 0.002 rel diff = Inf
    - !! An apparent nosocomial outbreak of highly resistant *Acinetobacter baumannii*. This is the seeded outbreak from 1994.

EVENTS FROM THE 9704 SEARCH

15. EmptySet ⇒ { NP_GNR, nosocomial, *Klebsiella oxytoca*}
    - []9701: 2/836 | []9702: 5/742 | []9703: 1/800 | *9704: 10/784 |
    - p = 0.01 rel diff = 3.79145
    - Possible outbreak of nosocomial *Klebsiella oxytoca*.

16. {LocMICU} ⇒ {nosocomial, R~Cefazolin}
    - []9609: 0/26 | []9610: 3/15 | []9611: 0/23 | []9612: 0/16 | []9701: 1/15 | []9702: 1/23 | *9703: 5/24 | *9704: 4/17 |
    - p = 0.003 rel diff = 5.18049
    - Nosocomial resistance to cefazolin is up in the MICU.

17. {*Pseudomonas aeruginosa*} ⇒ {R~TicarcillinClavK, R~Ciprofloxacin}
    - []9609: 0/98 | []9610: 3/100 | []9611: 1/79 | []9612: 1/73 | []9701: 2/83 | []9702: 0/66 | *9703: 5/85 |*9704: 6/69 |
    - p = 0.001 rel diff = 5.09184
    - *P. aeruginosa* is becoming increasingly resistant to two important anti-pseudomonas drugs.

18. {NP_GNR, nosocomial} ⇒ {R~Ceftazidime, R~Cefuroxime, R~Ciprofloxacin}
    - []9609: 0/180 | []9610: 1/124 | []9611: 0/130 | []9612: 3/100 | []9701: 4/121 | []9702: 6/108 | *9703: 8/106 | *9704: 5/122 |
    - p = 0.002 rel diff = 3.10746
    - Cephalosporin and fluoroquinolone resistance is on the rise amongst non-*Pseudomonas* gram-negative rods.

EVENTS FROM THE 9705 SEARCH

19. {nosocomial, LocMICU} ⇒ {R~Cefazolin}
    - []9610: 3/8 | []9611: 0/14 | []9612: 0/11 | []9701: 1/7 | []9702: 1/13 | []9703: 5/20 | *9704: 4/11 | *9705: 7/17 |
    - p = 0.005 rel diff = 2.86786
    - Cefazolin resistance remains up amongst nosocomial MICU isolates.

20. {*Pseudomonas aeruginosa*} ⇒ {R~TicarcillinClavK, R~Ciprofloxacin}
    - []9610: 3/100 | []9611: 1/79 | []9612: 1/73 | []9701: 2/83 | []9702: 0/66 | []9703: 5/85 | *9704: 6/69 | *9705: 6/74 |
    - p = 0.001 rel diff = 3.3986
    - *P. aeruginosa* resistance to ticarcillin and ciprofloxacin remains high.

21. {LocCCU} ⇒ {NP_GNR, *Proteus mirabilis*, nosocomial}
    - []9702: 0/20 | []9703: 0/23 | []9704: 0/21 | *9705: 3/13 |
    - p = 0.008 rel diff = Inf
    - Nosocomial *P. mirabilis* makes an appearance in the CCU.

22. {*Pseudomonas aeruginosa*} ⇒ {R~Piperacillin, R~Tobramycin, R~TicarcillinClavK}
    - []9702: 0/66 | []9703: 0/85 | []9704: 0/69 | *9705: 4/74 |
    - p = 0.008 rel diff = Inf
    - A new combination of *P. aeruginosa* resistance comes on the scene.

EVENTS FROM THE 9706 SEARCH

23. {LocP7} ⇒ {NP_GNR, nosocomial}
    - []9611: 4/21 | []9612: 0/13 | []9701: 1/29 | []9702: 0/24 | []9703: 0/21 | []9704: 1/15 | *9705: 4/18 | *9706: 5/21 |
    - p = 0.004 rel diff = 4.73077
    - Possible infection control breach in P7.

24. {NICU} ⇒ {NP_GNR, nosocomial, *Enterobacter cloacae*}
    - []9703: 1/32 | []9704: 0/33 | []9705: 0/24 | *9706: 3/12 |
    - p = 0.01 rel diff = 22.25
    - Nosocomial *Enterobacter cloacae* up in the NICU.

EVENT FROM THE 9707 SEARCH

25. EmptySet ⇒ {GPC, *Staphylococcus aureus*, LocS9SW, R~Cefazolin, R~AmoxicillinClK, R~Cephalothin, R~Oxacillin}
    - []9704: 0/786 | []9705: 0/742 | []9706: 0/717 | *9707: 5/901 |
    - p = 0.004 rel diff = Inf
    - A possible outbreak of MRSA in S9SW.

EVENTS FROM THE 9708 SEARCH

26. EmptySet ⇒ {NP_GNR, Nosoocomial, LocIM}
    - []9705: 0/742 | []9706: 0/717 | []9707: 0/903 | *9708: 4/787 |
    - p = 0.008 rel diff = Inf
    - Possible infection control breach in IM.

27. {NP_GNR, nosocomial} ⇒ {R~TicarcillinClavK}
    - []9705: 1/118 | []9706: 1/120 | []9707: 2/141 | *9708: 12/127 |
    - p = 0 rel diff = 8.95276
    - Ticarcillin resistance on the rise amongst nosocomial NP_GNR.

28. {nosocomial, LocRNIC} ⇒ {R~Ceftazidime}
    - []9705: 1/10 | []9706: 0/8 | []9707: 1/13 | *9708: 6/13 |
    - p = 0.01 rel diff = 7.15385
    - Ceftazidime resistance up in nosocomial isolates from the RNIC.

29. EmptySet ⇒ {nosocomial, GPC, LocNICU, *Staphylococcus aureus*}
    - []9701: 3/834 | []9702: 3/742 | []9703: 2/800 | []9704: 6/784 | []9705: 5/740 | []9706: 3/715 | *9707: 9/901 | *9708: 9/787 |
    - p = 0.009 rel diff = 2.23691
    - Higher incidence of nosocomial *Staphylococcus aureus* from the NICU.

30. {*Klebsiella pneumoniae*, NP_GNR, nosocomial} ⇒ {R~Piperacillin, R~Cefotetan, R~Ceftazidime, R~Cefuroxime, R~Cefazolin, R~Cotrimazole, R~Cephalothin, R~Tobramycin, R~TicarcillinClavK, R~Ciprofloxacin, R~Ceftriaxone}
    - []9705: 0/28 | []9706: 0/28 | []9707: 0/21 | *9708: 4/26 |
    - p = 0.007 rel diff = Inf
    - Possible clonal outbreak of highly-resistant *Klebsiella pneumoniae.*

31. {NP_GNR, nosocomial} ⇒ {*Acinetobacter baumannii*, R~Piperacillin, R~Cefotetan, R~Cefuroxime, R~Cefazolin R~Cephalothin R~Gentamicin R~Tobramycin R~Ciprofloxacin R~Amikacin R~Aztreonam R~Mezlocillin R~Ofloxacin R~Cefotaxime R~Trimethoprim}
    - []9705: 3/118 | []9706: 0/120 | []9707: 0/141 | *9708: 7/127 |
    - p = 0.007 rel diff = 6.96325
    - Vicious strain of nosocomial *Acinetobacter baumannii* seen in the 9703 search makes a return appearance.

EVENTS FROM THE 9709 SEARCH

32. EmptySet ⇒ {NP_GNR, nosocomial, *Enterobacter aerogenes*}
    - []9706: 1/717 | []9707: 5/901 | []9708: 7/787 | *9709: 13/872 |
    - p = 0.007 rel diff = 2.75803
    - Nosocomial *Enterobacter aerogenes* up in each of last four months.

33. {LocIM} ⇒ {NP_GNR, nosocomial}
    - []9702: 0/40 | []9703: 0/28 | []9704: 0/28 | []9705: 0/35 | []9706: 0/20 | []9707: 0/41 | *9708: 4/39 | *9709: 2/34 |
    - p = 0.001 rel diff = Inf
    - Rare nosocomial NP_GNR isolates from the IM. Possible infection control breach.

34. EmptySet ⇒ {*Serratia marcescens*, NP_GNR, nosocomial, R~Ceftazidime}
    - []9610: 1/894 | []9611: 1/760 | []9612: 0/709 | []9701: 3/834 | []9702: 0/744 | []9703: 0/800 | []9704: 2/786 | []9705: 1/742 | []9706: 1/717 | *9707: 2/903 | *9708: 3/787 | *9709: 6/872 |
    - p = 0.004 rel diff = 3.33273
    - Nosocomial, ceftazidime resistant *Serratia marcescens* up in last three months.

35. EmptySet ⇒ {NP_GNR, nosocomial, LocNICU, *Serratia marcescens*}
    - []9706: 0/717 | []9707: 0/903 | []9708: 0/789 | *9709: 4/872 |
    - p = 0.01 rel diff = Inf
    - Possible outbreak of nosocomial *Serratia marcescens* in the NICU.

EVENTS FROM THE 9710 SEARCH

36. EmptySet ⇒ {NP_GNR, nosocomial, LocS6SW}
    - []9707: 0/903 | []9708: 1/789 | []9709: 0/874 | *9710: 5/854 |
    - p = 0.009 rel diff = 15.0234
    - Possible infection control breach in S6SW.

EVENTS FROM THE 9711 SEARCH

37. EmptySet ⇒ {nosocomial, LocSICU, R~Ceftriaxone}
    - []9704: 0/786 | []9705: 1/742 | []9706: 1/717 | []9707: 2/903 | []9708: 0/789 | []9709: 1/874 | *9710: 5/856 | *9711: 4/764 |
    - p = 0.005 rel diff = 5.34556
    - Nosocomial, SICU, Ceftriaxone resistance up.

38. EmptySet ⇒ {non-nosocomial, GPC, *Streptococcus pneumoniae*}
    - []9708: 1/789 | []9709: 4/872 | []9710: 4/856 | *9711: 10/762 |
    - p = 0.01 rel diff = 3.67017
    - Community acquired *Streptococcus pneumoniae* up. (expected seasonal)

39. {LocSICU} ⇒ {NP_GNR, nosocomial, *Enterobacter cloacae*}
    - []9704: 3/38 | []9705: 1/26 | []9706: 2/32 | []9707: 3/33 | []9708: 1/26 | []9709: 3/54 | *9710: 7/50 | *9711: 9/46 |
    - p = 0.004 rel diff = 2.67949
    - Nosocomial *Enterobacter cloacae* up in the SICU.

40. {LocRNIC} ⇒ {nosocomial, *Staphylococcus epidermidis*, GPC}
    - []9708: 5/13 | []9709: 6/13 | []9710: 5/12 | *9711: 10/11 |
    - p = 0.004 rel diff = 2.15909
    - Investigate infection control practices in RNIC

41. EmptySet ⇒ {nosocomial, *Staphylococcus epidermidis*, GPC, R~AmoxicillinClK, R~ErythromycinEst, R~Clindamycin, LocRNIC}
    - []9612: 1/709 | []9701: 2/836 | []9702: 2/744 | []9703: 4/798 | []9704: 1/786 | []9705: 2/742 | []9706: 4/715 | []9707: 3/901 | []9708: 3/787 | *9709: 5/874 | *9710: 5/854 | *9711: 8/762 |
    - p = 0.007 rel diff = 2.30602
    - Related to event 40. Suggests increase incidence of a specific strain of *Staphylococcus epidermidis* in the RNIC.

APPENDIX B

SELECTED EVENTS FROM THE ANALYSIS OF THE CDC DATA SET

EVENTS FROM THE 9504 SEARCH

1. {StateEE} ⇒ {Ethnic9}
   - []9501: 1/39 | []9502: 4/38 | []9503: 11/51 | *9504: 19/46 |
   - p = 0 rel diff = 3.30435

2. {Sex2, StateBB} ⇒ {Bacteremia, Ethnic2, CountyBB03}
   - []9501: 0/4 | []9502: 0/9 | []9503: 1/13 | *9504: 6/13 |
   - p = 0.006 rel diff = 12

EVENTS FROM THE 9505 SEARCH

3. {Sex1, StateDD} ⇒ {CountyDD20}
   - []9502: 0/0 | []9503: 0/0 | []9504: 2/17 | *9505: 10/18 |
   - p = 0.006 rel diff = 4.72222

4. {StateAA, AgeGrp2YRS} ⇒ {Ethnic9, Bacteremia, HospIDAA015}
   - []9502: 0/16 | []9503: 0/14 | []9504: 0/22 | *9505: 3/11 |
   - p = 0.008 rel diff = Inf

EVENTS FROM THE 9507 SEARCH

5. {StateCC, CountyCC02} ⇒ {Ethnic9}
   - []9504: 2/16 | []9505: 2/16 | []9506: 2/8 | *9507: 8/8 |
   - p = 0 rel diff = 6.66667

6. {RaceWHITE, StateHH, CountyHH03} ⇒ {Bacteremia}
   - []9504: 1/18 | []9505: 1/8 | []9506: 1/8 | *9507: 4/6 |
   - p = 0.01 rel diff = 7.55556

7. {StateAA, Ethnic9, AgeGrp2YRS} ⇒ {Bacteremia, HospIDAA015}
   - []9504: 0/16 | []9505: 3/10 | []9506: 1/9 | *9507: 7/13 |
   - p = 0.009 rel diff = 4.71154

8. {RaceBLACK, Sex1, Ethnic2, StateEE} ⇒ {HospIDA25}
   - []9504: 1/10 | []9505: 2/10 | []9506: 2/7 | *9507: 7/8 |
   - p = 0.002 rel diff = 4.725

9. {Sex1, StateCC} ⇒ {Ethnic9, CountyCC02, HospIDCC002}
   - []9504: 0/32 | []9505: 0/24 | []9506: 0/14 | *9507: 4/18 |
   - p = 0.003 rel diff = Inf

EVENTS FROM THE 9508 SEARCH

10. {Ethnic9, StateCC} ⇒ {CountyCC02}
    - []9501: 0/0 | []9502: 0/0 | []9503: 3/23 | []9504: 2/18 | []9505: 2/15 | []9506: 2/9 | *9507: 8/20 | *9508: 4/8 |
    - p = 0.002 rel diff = 3.09524

11. {StateAA} ⇒ {Bacteremia, R~CotSIR2, R~TaxSIR, Serotype23F}
    - []9505: 0/50 | []9506: 1/37 | []9507: 1/34 | *9508: 3/11 |
    - p = 0.008 rel diff = 16.5

EVENTS FROM THE 9510 SEARCH

12. {StateCC} ⇒ {Ethnic9, CountyCC02}
    - []9503: 3/60 | []9504: 2/59 | []9505: 2/50 | []9506: 2/39 | []9507: 8/29 | []9508: 4/16 | *9509: 4/25 | *9510: 15/53 |
    - p = 0 rel diff = 2.93468

13. {Ethnic9, StateCC, CountyCC02} ⇒ {HospIDCC021}
    - []9507: 0/8 | []9508: 0/4 | []9509: 0/4 | *9510: 7/15 |
    - p = 0.005 rel diff = Inf

14. {StateBB} ⇒ {Bacteremia, Pneum1, Ethnic2, CountyBB05}
    - []9507: 0/7 | []9508: 1/6 | []9509: 2/10 | *9510: 9/17 |
    - p = 0.006 rel diff = 4.05882

EVENTS FROM THE 9511 SEARCH

15. {StateAA, Ethnic9} ⇒ {Serotype23F}
    - []9504: 3/50 | []9505: 1/45 | []9506: 3/34 | []9507: 3/32 | []9508: 1/9 | []9509: 3/21 | *9510: 6/35 | *9511: 10/52 |
    - p = 0.006 rel diff = 2.50903

16. {RaceWHITE, StateCC} ⇒ {R~CotSIR2, R~PenSIR2}
    - []9504: 3/41 | []9505: 3/38 | []9506: 1/25 | []9507: 0/16 | []9508: 0/9 | []9509: 1/15 | *9510: 3/35 | *9511: 10/42 |
    - p = 0.006 rel diff = 3.03896

17. EmptySet ⇒ {R~CotSIR2, R~PenSIR2, Ethnic2, AntbgramRISRSSSSI}
    - []9508: 0/89 | []9509: 0/134 | []9510: 0/266 | *9511: 6/305 |
    - p = 0.006 rel diff = Inf

18. {StateAA, Sex2, Ethnic9} ⇒ {Bacteremia, Serotype014}
    - []9504: 1/23 | []9505: 0/21 | []9506: 3/14 | []9507: 0/15 | []9508: 0/2 | []9509: 1/9 | *9510: 3/16 | *9511: 8/28 |
    - p = 0.002 rel diff = 4.2

19. {Sex2, StateBB} ⇒ {Bacteremia, Ethnic2, CountyBB05}
    - []9504: 2/13 | []9505: 2/9 | []9506: 1/4 | []9507: 0/2 | []9508: 1/4 | []9509: 2/5 | *9510: 6/7 | *9511: 7/12 |
    - p = 0.001 rel diff = 3.16447

EVENTS FROM THE 9512 SEARCH

20. {StateAA, Ethnic9} ⇒ {Serotype004}
    - []9509: 1/21 | []9510: 1/35 | []9511: 1/52 | *9512: 13/81 |
    - p = 0.001 rel diff = 5.77778

21. {StateAA, Ethnic9} ⇒ {Serotype06B}
    - []9501: 4/56 | []9502: 3/64 | []9503: 2/55 | []9504: 3/50 | []9505: 3/45 | []9506: 3/34 | []9507: 2/32 | []9508: 0/9 | []9509: 1/21 | *9510: 5/35 | *9511: 6/52 | *9512: 10/81 |
    - p = 0.007 rel diff = 2.17857

22. {StateCC} ⇒ {Ethnic9, CountyCC02}
    - []9501: 0/0 | []9502: 0/0 | []9503: 3/60 | []9504: 2/59 | []9505: 2/50 | []9506: 2/39 | []9507: 8/29 | []9508: 4/16 | []9509: 4/25 | *9510: 15/53 | *9511: 13/66 | *9512: 21/89 |
    - p = 0 rel diff = 2.61962

23. {Ethnic2, StateDD} ⇒ {CountyDD20}
    - []9509: 2/7 | []9510: 0/8 | []9511: 2/10 | *9512: 9/14 |
    - p = 0.007 rel diff = 4.01786

24. {StateAA, AgeGrp1864YRS} ⇒ {Ethnic9, Serotype004}
    - []9509: 1/8 | []9510: 1/18 | []9511: 0/16 | *9512: 10/40 |
    - p = 0.01 rel diff = 5.25

25. {StateAA, CountyAA05} ⇒ {Ethnic9, ZipAA03Z}
    - []9509: 0/5 | []9510: 0/6 | []9511: 0/11 | *9512: 7/21 |
    - p = 0.007 rel diff = Inf

26. {StateCC} ⇒ {Ethnic9, CountyCC02, HospIDCC002}
    - []9509: 0/25 | []9510: 0/53 | []9511: 0/66 | *9512: 8/89 |
    - p = 0.001 rel diff = Inf

27. {AgeGrp1864YRS, StateDD} ⇒ {Pneum1, CountyDD20}
    - []9509: 0/2 | []9510: 0/7 | []9511: 1/10 | *9512: 12/27 |
    - p = 0.004 rel diff = 8.44444

28. {StateCC, CountyCC07} ⇒ {Pneum1, ZipCC51D}
    - []9505: 0/16 | []9506: 1/18 | []9507: 1/7 | []9508: 0/0 | []9509: 0/10 | []9510: 2/12 | *9511: 3/19 | *9512: 8/25 |
    - p = 0.006 rel diff = 3.9375

29. {StateAA} ⇒ {Ethnic9, Bacteremia, CountyAA05, ZipAA03Z}
    - []9509: 0/24 | []9510: 0/39 | []9511: 0/65 | *9512: 6/90 |
    - p = 0.009 rel diff = Inf

30. {Sex2, StateBB} ⇒ {Bacteremia, R~CotSIR2, R~PenSIR2}
    - []9505: 0/9 | []9506: 1/4 | []9507: 0/2 | []9508: 0/4 | []9509: 1/5 | []9510: 0/7 | *9511: 5/12 | *9512: 5/16 |
    - p = 0.005 rel diff = 5.53571

31. {RaceBLACK , StateCC} ⇒ {Pneum1, CountyCC07, ZipCC51D}
    - []9505: 0/8 | []9506: 1/11 | []9507: 1/9 | []9508: 0/5 | []9509: 0/7 | []9510: 1/12 | *9511: 3/14 | *9512: 7/25 |
    - p = 0.007 rel diff = 4.44444

32. {StateHH , CountyHH03} ⇒ {Pneum1, Ethnic2, HospID00002}
    - []9509: 0/5 | []9510: 0/10 | []9511: 2/22 | *9512: 7/15 |
    - p = 0.002 rel diff = 8.63333

EVENTS FROM THE 9601 SEARCH

33. {StateEE} ⇒ {CountyEE08}
    - []9510: 1/38 | []9511: 2/50 | []9512: 4/80 | *9601: 10/61 |
    - p = 0.007 rel diff = 3.93443

34. {StateDD} ⇒ {Ethnic9, HospIDDD524}
    - []9510: 0/42 | []9511: 1/38 | []9512: 0/64 | *9601: 5/35 |
    - p = 0.002 rel diff = 20.5714

35. {StateAA, Ethnic9, AgeGrp1864YRS} ⇒ {Serotype004}
    - []9506: 0/13 | []9507: 0/10 | []9508: 1/2 | []9509: 1/5 | []9510: 1/17 | []9511: 0/13 | *9512: 10/37 | *9601: 5/26 |
    - p = 0.003 rel diff = 4.7619

36. {Sex2, StateDD} ⇒ {Ethnic9, CountyDD14}
    - []9510: 0/16 | []9511: 0/12 | []9512: 1/27 | *9601: 5/14 |
    - p = 0.002 rel diff = 19.6429

37. {Sex2, Ethnic9, StateDD} ⇒ {CountyDD14}
    - []9510: 0/13 | []9511: 0/7 | []9512: 1/18 | *9601: 5/11 |
    - p = 0.003 rel diff = 17.2727

38. {Sex2, Ethnic2, StateCC} ⇒ {Bacteremia}
    - []9506: 0/18 | []9507: 0/2 | []9508: 0/2 | []9509: 0/2 | []9510: 1/6 | []9511: 0/8 | *9512: 4/14 | *9601: 2/4 |
    - p = 0.006 rel diff = 12.6667

39. {Sex1, StateCC, CountyCC02} ⇒ {Ethnic9, ZipCC90Z, HospIDCC028}
    - []9510: 0/10 | []9511: 0/9 | []9512: 0/20 | *9601: 3/8 |
    - p = 0.007 rel diff = Inf

40. {Sex1, AgeGrp1864YRS, StateCA, CountySANFRANCI} ⇒ {Outcome9, HospIDSF051}
    - []9510: 0/7 | []9511: 0/11 | []9512: 0/10 | *9601: 5/16 |
    - p = 0.008 rel diff = Inf

EVENTS FROM THE 9602 SEARCH

41. {StateAA} ⇒ {Ethnic9, Bacteremia, Serotype004}
    - []9503: 1/72 | []9504: 2/66 | []9505: 3/50 | []9506: 0/37 | []9507: 2/34 | []9508: 0/11 | []9509: 1/24 | []9510: 1/39 | []9511: 1/65 | *9512: 7/90 | *9601: 5/64 | *9602: 2/19 |
    - p = 0.004 rel diff = 2.92801

42. {StateHH, CountyHH03} ⇒ {Ethnic2, HospID00006}
    - []9507: 0/7 | []9508: 0/10 | []9509: 0/5 | []9510: 0/10 | []9511: 1/22 | []9512: 0/15 | *9601: 3/21 | *9602: 8/35 |
    - p = 0.001 rel diff = 13.5536

43. {StateAA} ⇒ {CountyAA07, Ethnic9, Bacteremia, ZipAA31B}
    - []9511: 2/65 | []9512: 2/90 | []9601: 0/64 | *9602: 4/19 |
    - p = 0.003 rel diff = 11.5263

44. {StateAA, Ethnic9, RaceWHITE, AgeGrp65YRS} ⇒ {Pneum1, CountyAA08}
    - []9503: 0/5 | []9504: 0/7 | []9505: 1/6 | []9506: 0/8 | []9507: 0/1 | []9508: 0/1 | []9509: 0/1 | []9510: 0/5 | []9511: 0/7 | *9512: 3/10 | *9601: 2/6 | *9602: 1/2 |
    - p = 0.005 rel diff = 13.6667

EVENTS FROM THE 9603 SEARCH

45. {StateAA} ⇒ {Ethnic9, R~CotSIR2, R~PenSIR2}
    - []9512: 7/90 | []9601: 6/64 | []9602: 1/19 | *9603: 11/44 |
    - p = 0.002 rel diff = 3.08929

46. {StateHH, CountyHH03} ⇒ {Ethnic2, HospID00006}
    - []9508: 0/10 | []9509: 0/5 | []9510: 0/10 | []9511: 1/22 | []9512: 0/15 | []9601: 3/21 | *9602: 8/35 | *9603: 5/26 |
    - p = 0.002 rel diff = 4.42213

47. {StateAA, Sex2} ⇒ {Ethnic9, Bacteremia, R~TetSIR2}
    - []9512: 1/41 | []9601: 2/25 | []9602: 0/10 | *9603: 5/14 |
    - p = 0.004 rel diff = 9.04762

48. {Sex2, StateHH, CountyHH03} ⇒ {Ethnic2, HospID00006}
    - []9504: 0/8 | []9505: 0/7 | []9506: 0/6 | []9507: 0/4 | []9508: 0/7 | []9509: 0/1 | []9510: 0/4 | []9511: 1/10 | []9512: 0/6 | *9601: 1/7 | *9602: 4/18 | *9603: 4/15 |
    - p = 0.004 rel diff = 11.925

49. {StateAA, Sex2} ⇒ {Bacteremia, R~CotSIR2, R~PenSIR2, R~TaxSIR, R~ErySIR2, R~TetSIR2}
    - []9512: 0/41 | []9601: 0/25 | []9602: 0/10 | *9603: 3/14 |
    - p = 0.006 rel diff = Inf

GRADUATE SCHOOL
UNIVERSITY OF ALABAMA AT BIRMINGHAM
DISSERTATION APPROVAL FORM
DOCTOR OF PHILOSOPHY

Name of Candidate Stephen Brossette

Major Subject Computer and Information Sciences

Title of Dissertation Data Mining and Epidemiologic Surveillance

I certify that I have read this document and examined the student regarding its content. In my opinion, this dissertation conforms to acceptable standards of scholarly presentation and is adequate in scope and quality, and the attainments of this student are such that he may be recommended for the degree of Doctor of Philosophy.

Dissertation Committee:

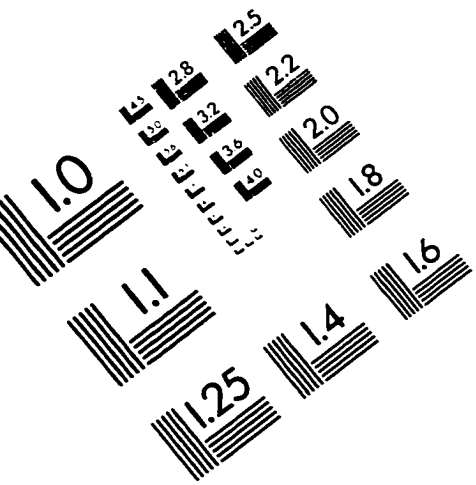
Name		Signature
Dr. Warren Jones	**, Chair**	Warren Jones
Dr. Stephen A. Moser	**, Co-Chair**	Stephen A Moser
Dr. J. Michael Hardin		J. Michael Hardin
Dr. Robert Hyatt		Robert M Hyatt
Dr. Alan Sprague		Alan P Sprague

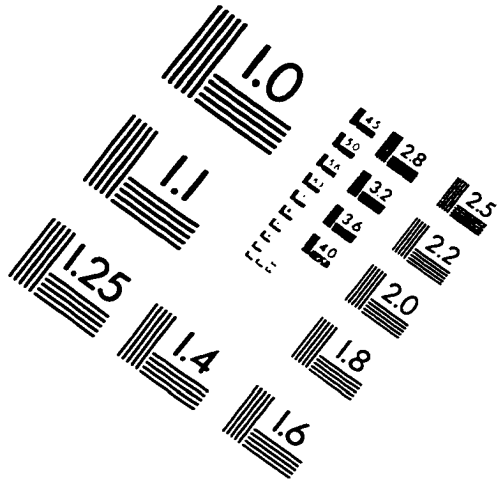
Director of Graduate Program Warren Jones

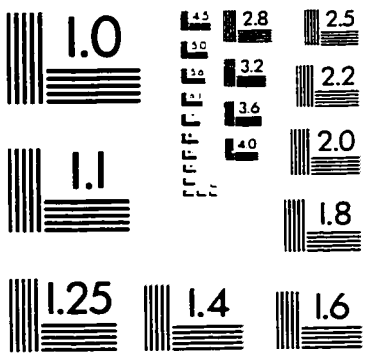
Dean, UAB Graduate School James L Lorden

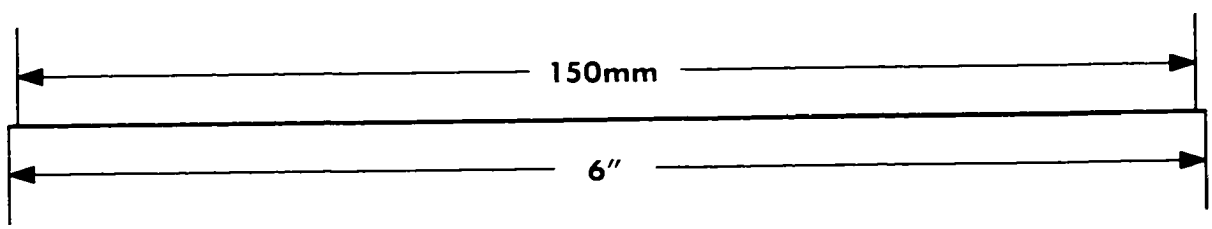
Date 7/6/98

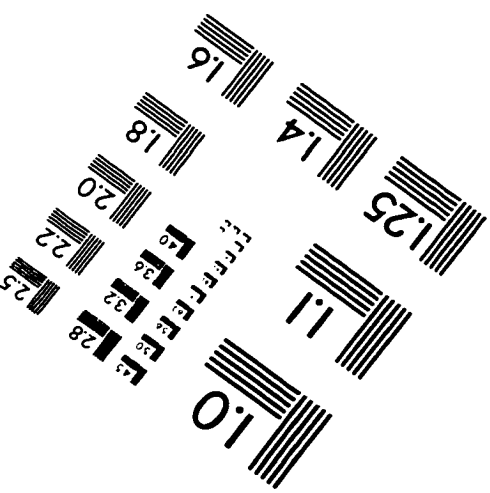
IMAGE EVALUATION
TEST TARGET (QA-3)

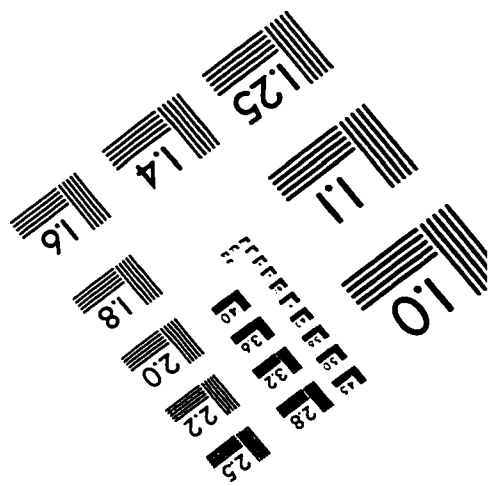












APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989