Errors in the Clinical Laboratory: A Novel Approach to Autoverification

Gregory B. Strylewicz

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2007

Program Authorized to Offer Degree: Medical Education
and Biomedical Informatics

University of Washington
Graduate School

This is to certify that I have examined this copy of a doctoral dissertation by

Gregory B. Strylewicz

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Chair of the Supervisory Committee:

_____
Peter Tarczy-Hornoch

Reading Committee:

_____
Peter Tarczy-Hornoch

_____
Jason Doctor

_____
Michael Astion

Date: _____

In presenting this dissertation in partial fulfillment of the requirements for the doctoral degree at the University of Washington, I agree that the Library shall make its copies freely available for inspection. I further agree that extensive copying of the dissertation is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for copying or reproduction of this dissertation may be referred to ProQuest Information and Learning, 300 North Zeeb Road, Ann Arbor, MI 48106-1346, 1-800-521-0600, or to the author.

Signature _____

Date _____

University of Washington

Abstract

Detecting Errors in the Clinical Laboratory: Detecting the Invisible

Gregory B. Strylewicz

Chair of the Supervisory Committee:
Professor Peter Tarczy-Hornoch
Department of Medical Education

Clinical laboratories provide a critical service to the health care and well-being of the world's population. Estimates suggest that the clinical laboratory influences some 70 percent of health-care decisions, but requires only about 4 percent of the health-care expenditures. Given an estimated 7 billion laboratory tests per year in the United States, about 1% of the results, or 70 million laboratory errors annually, are erroneous with an estimated 6%, of those errors causing harm to the patient. Laboratory errors harm millions of patients each year and laboratory experts spend countless hours reviewing billions of laboratory results each year in the search for these rare errors. Autoverification systems, automated programs used to check laboratory results for errors, can save laboratories countless hours and be more accurate than laboratory experts, but the current generation of rule-based systems is not appropriate for the clinical laboratory domain due to its inherent uncertainty. This research demonstrates that a novel approach using a synthetic error generation system to create training datasets for a conditional Gaussian Bayesian network produces an autoverification system superior to ones trained using standard methods and superior to laboratory experts. Unlike standard approaches that require an expensive and time-consuming expert annotation process to create training datasets, the synthetic error generation method uses results that were reviewed normally.

By creating synthetic datasets, the synthetic error generation process creates customized training datasets, which maximize the Bayesian network's performance in detecting errors. In this dissertation, we review the clinical laboratory process and the many sources of errors in clinical laboratory results, Bayesian networks, and the class imbalance problem. Next, we elucidate the performance characteristics of the synthetic error generation process, which is followed by a comparison between our novel method and standard approaches to the class imbalance problem. Finally, we compare the results of a synthetic error autoverification system against laboratory experts in the identification of errors.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

ANALYTE: A chemical substance, for example glucose and cholesterol, which is measured in a clinical laboratory.

AREA UNDER CURVE (AUC): The area under the receiver operating characteristics (ROC) curve, which provides a metric for the performance of a classifier. An AUC of 0.5 indicates a guessing whereas an AUC of 1.0 indicates a perfect classifier.

AUTOVERIFICATION: The process by which clinical laboratory results are programmatically reviewed to determine if they meet the laboratory's acceptance criteria. Results that pass the laboratory's acceptance criteria may be free for reporting while a laboratory expert reviews failing results.

CLASS IMBALANCE: In a dichotomous classification, this refers to one class being more frequent than the other class.

CLASS OVERLAP: In a dichotomous classification, this refers to the degree to which the classes are dissimilar or disjunct.

DELTA CHECK: A method used to check laboratory results for errors by evaluating the plausibility of the observed change in an analyte's value over time.

DISJUNCT: The difference in attributes between two classes that make an object a member of one class instead of another class.

ERROR: An error is the difference between an analyte's true value and its measured value.

ERROR, ANALYTICAL: An error that occurs during in the clinical laboratory cycle where the specimen is being analyzed.

ERROR, POST-ANALYTICAL: An error that occurs in the clinical laboratory cycle after the result has been released for reporting.

ERROR, PRE-ANALYTICAL: An error that occurs in the clinical laboratory cycle prior to the clinical laboratory analyzing the specimen.

ERROR, SAMPLE-SWITCHING: An error that occurs when a patient's specimen is identified as a different patient.

FAITHFULNESS: A graph and joint probability distribution is said to satisfy the faithfulness condition if they satisfy the Markov condition and the only conditional independencies in the joint probability distribution are those entailed by the Markov condition (Neapolitan 2004). See Markov condition.

GLUCOSE: An analyte in the human body commonly measured to detect and monitor the progression of diabetes.

GLYCOSYLATED HEMOGLOBIN (HBA1C): An analyte in the human body formed by a non-catalytic reaction between hemoglobin and glucose that is measured to monitor the progression of diabetes.

GOLD STANDARD: A definitive test or method to classify an analyte's measured result as an error or not and that is never wrong.

INTERNAL CONSISTENCY: A method to evaluate clinical laboratory results by checking that correlated analytes maintain the expected relationship.

LIMS: Laboratory Information Management System – a database used to manage data in a clinical laboratory.

LIS: Laboratory Information System – see LIMS.

MAJORITY-CLASS: In a dichotomous classification system with a class imbalance, this term refers to the more frequent class.

MARKOV CONDITION: A graph and joint probability distribution is said to satisfy the Markov condition if for every variable in the graph, it is conditionally independent of all its nondescendants given the values for all of its parents (Neapolitan 2004). See faithfulness.

MINORITY-CLASS: In a dichotomous classification system with a class imbalance, this term refers to the less frequent class.

NODE: An element in the graph of a Bayesian network, which generally represents an analyte.

NODE - ANCESTOR: In a directed acyclic graph, an ancestor of a node is a node that is upstream of the first node.

NODE - ADJACENT: In a graph, two nodes with an edge between them are said to be adjacent.

NODE - DESCENDENT: In a directed acyclic graph, a descendent node is a node that is downstream of the first node.

NODE - PARENT: In a directed acyclic graph, a parent of a node is a node that is immediately upstream of the first node. The graph contains a node from the parent node to the child node.

PROBABILITY, POSTERIOR: The probability of some event after observing some evidence.

PROBABILITY, PRIOR: The probability of some event prior to obtaining new evidence.

RECEIVER OPERATING CHARACTERISTIC (ROC) CURVE: A graphical plot of a dichotomous classifier's performance. The vertical axis corresponds to the true positive rate and the horizontal axis corresponds to the false positive rate. Each point on the curve corresponds to a classification threshold.

SENSITIVITY: A measure of how well a dichotomous classifier correctly classifies positive cases (erroneous laboratory results). This measure depends on the classification threshold being used and varies between 0.0 and 1.0.

SINGLE-CLASS CLASSIFIER: A classification system that, instead classifying an object as belonging to one class or another, classifies objects as belonging to the single class. For example, a system in the clinical laboratory that is trained to only recognize the single class of acceptable results.

SPECIFICITY: A measure of how well a dichotomous classifier correctly classifies negative cases (acceptable laboratory results). This measure depends on the classification threshold being used and varies between 0.0 and 1.0.

ACKNOWLEDGEMENTS

I wish to express my appreciation to the following individuals:

To Drs. Peter Tarcy-Hornoch and Michael Astion for their invaluable advice and guidance.

To Dr. Santica Marcovina for her support, patience, and willingness that made this journey possible.

To Katherine Rosecrans for keeping my spirits up on those gloomy days when nothing went right.

To the staff of Northwest Lipid Metabolism and Diabetes Research Laboratories for their priceless patience as I always seemed to be running out the door.

Finally, I wish to expresses my deepest appreciation to Dr. Jason Doctor for his guidance and support in helping take a tiny seed of an idea and growing it into what it is today. He is a true mentor.

DEDICATION

To my friends and family for their never-ending love and support.

Chapter 1: Introduction

Clinical laboratories provide a critical service to the health care and well-being of the world's population. It is estimated that the clinical laboratory influences some 70 percent of health-care decisions, but requires only about 4 percent of the health-care expenditures (Silverstein 2003; Laboratory Corporation of America 2007). Clinical laboratories have been able to provide this enormous value by using highly automated instruments and complex database applications. While the instruments have become increasingly automated and complex, the methods employed by the clinical laboratory to check the results for errors has not changed. Over half of the clinical laboratories do not use an automated program, called an autoverification system, to check for errors (American Association for Clinical Chemistry 2007). Unfortunately, most errors in clinical laboratory results are caused by errors that occur before the sample even reaches the analyzer, so improvements to analytic instruments do little to reduce the number of errors (Bonini, Plebani et al. 2002). Of those laboratories that employ an autoverification program, virtually all are based on rules. However, rules are a poor choice as a decision tool in an environment, such as the clinical laboratory, with rare errors and a high degree of uncertainty. The primary research question is whether we can do better at detecting errors in the clinical laboratory.

As we will discuss, development of an effective autoverification system is stymied by the rarity of laboratory errors as well as characteristics of laboratory

analyses and their errors. This dissertation is the culmination of nearly 8 years of effort to develop an effective laboratory autoverification system that is a radical departure from historical approaches. It describes complex issues involved in the development of autoverification systems and proposes a novel method, synthetic error generation, for addressing those issues. Synthetic-error generation is used to create training datasets, which are then used to train Bayesian networks to detect laboratory errors. After discussing the clinical laboratory process, Bayesian networks, and a major technical impediment (class imbalance), we describe the performance characteristics of the proposed method and compare performance against standard methods as well as laboratory experts.

1.1 Background and Significance

Since the Institute of Medicine's 1999 report that estimated 44,000 to 98,000 deaths each year in the United States due to medical errors, there has been an increasing public awareness of medical errors including a call in the President's State of the Union Address to reduce medical errors using better information technology (Institute of Medicine 1999; Bush 2007). Only more recently has there been an increasing public awareness of the importance of clinical laboratories and the costs due to laboratory errors (Landro 2006). Laboratory errors come from a variety of sources and each has its own implication for patient safety with the overall error rate generally estimated to be between 0.1% and 1.0% (Plebani and Carraro 1997). Given an estimated 7 billion laboratory tests per year in the United States, this equates to upwards of 70 million

laboratory errors annually. Laboratory errors are particularly problematic because they are pivotal in making many medical decisions. Hence, errors in laboratory values may lead to unnecessary further testing and erroneous treatment decisions. It is estimated that approximately 6% of erroneous laboratory results cause some harm to the patient (Astion 2006). Physicians acting on erroneous information can have adverse health consequences for patients and increase the cost of medical care by introducing inefficiency.

Laboratories use either expert review or a combination of automated methods, called autoverification systems, and expert review of flagged results, to review laboratory data and identify errors. Autoverification systems in use today such as LabRespond or VALAB may be purchased as middle-ware applications or may be developed by the local laboratory (Oosterhuis, Ulenkate et al. 2000; Prost and Rogari 2002). These systems virtually always use rules to review laboratory results, checking for such conditions as abnormal values, large changes in values over time, and internal consistency of results. For example, the autoverification system developed at New Cook County Hospital in Chicago, Illinois automatically releases for reporting (autoverifies), any cholesterol value between 80 and 450mg/dl or any glucose between 60 and 325mg/dl and holds for expert's manual review any cholesterol or glucose result outside of that range (Torke, Boral et al. 2005). In contrast to the few simple rules by New Cook County Hospital, VALAB uses over 25,000 rules to review laboratory results (Prost and Rogari 2002). While VALAB with its plethora of rules is able to out-

perform many existing systems, the large number and proprietary nature of rules renders VALAB's logic opaque to inspection (Oosterhuis, Ulenkate et al. 2000).

When developing an automated system to detect laboratory errors, developers need to address the rarity of laboratory errors, estimated at less than 1.0%, in order to balance the system's sensitivity and specificity along with the greater cost of mis-classifying erroneous values as acceptable. Class imbalance is the difference in the percentage of erroneous items compared to the percentage of acceptable items, and is a significant impediment to developing autoverification systems. If not properly addressed, the severe class imbalance in the clinical laboratory domain inhibits many machine-learning algorithms since they can achieve near-perfect performance, as measured by percent correctly classified, by concluding all results are acceptable. Typical methods used to ameliorate the class-imbalance problem include: 1) over-sampling, either directed or random, the minority class; 2) under-sampling, either directed or random, the majority class; 3) adjustment of misclassification costs; 4) single-class classifier (Japkowicz and Stephen 2002). However, due to characteristics of the clinical laboratory, which are discussed next, standard approaches are not expected to produce optimal systems.

In addition to the class imbalance problem, developers must also consider their gold standard for classifying results as erroneous or acceptable. Laboratory analyses are subject to varying degrees of biological variability, instrument imprecision and biases, and treatment affects. For example, cholesterol has a 6.0% within-in subject

coefficient of variation (Ricos, Alvarez et al. 1999). The Centers for Disease Control laboratory standards for an instrument performing cholesterol analysis allows for a bias not to exceed 3.0% and a coefficient of variation not to exceed of 3.0%, for a combined total allowed error not to exceed 8.9% (Centers for Disease Control and Prevention 2004). The combination of biological and instrumentation variability renders the results of laboratory analyses as only estimates of the true values and, therefore, limits the ability of a system to detect accurately deviations from the true value. At a great expense in human capital, human experts are usually capable of identifying most gross laboratory errors. As with other domains in the medical sciences, however, human experts should not be viewed as a "gold standard" because they do not have perfect sensitivity and specificity. However, several human experts evaluating the same data are often able to detect and remove virtually all sizeable naturally occurring errors from a dataset. When working off such a "cleaned" dataset, the only error for which there exists a gold standard are those knowingly introduced by the researcher (i.e., synthetic errors). Synthetic errors are errors deliberately introduced by the researcher via some rule or set of rules to facilitate the study of error identification. The set of rules used to synthesize errors often represents an analogue to naturally occurring error processes. For example, sample-switching errors can be modeled effectively by randomly switching a proportion of samples within a cleaned dataset and transcription errors by randomly changing the digits of single analyte values. This approach yields a criterion-based dataset with a gold standard for error identification as given by the record of the deliberate introduction of errors to the data.

1.2 Contribution of Work

In this dissertation, we describe a novel laboratory autoverification system utilizing Bayesian networks and synthetic error generation that out performs existing systems. We believe that once a predictive, reliable and valid Bayesian belief error detection model is developed and tested, it can be incorporated into autoverification protocols in laboratories or managed care organizations with relative ease and minimal cost. An autoverification system would alert the technician to a potential error, which (s)he could then investigate. The cost-offset of such a system could potentially be enormous. That is, the small cost of implementing a Bayesian autoverification protocol in laboratory settings would be offset by tremendous savings both in the reduction of laboratory errors and in the reduction of time needed to review results. Because primitive autoverification protocols are currently widely used in medical laboratories, there would be potentially a low socio-technical barrier to implementation of improved error detection protocols.

Specifically, this dissertation contributes to the biomedical and health informatics domain by showing that we can detect errors in the clinical laboratory better by:

- Showing that we can better train a classification algorithm. The performance characteristics of the proposed system are elucidated to show the relationship between class imbalance, class distinction, and within-class imbalance. In developing an autoverification system that learns from training data and is then

used to identify errors in a real dataset, the training data's parameters must be carefully configured to optimize performance.

- Demonstrating the superiority of the proposed system to existing methods designed to address the class imbalance problem in the clinical laboratory domain. The clinical laboratory domain is unique and standard approaches may not work as well as one designed specifically for the domain of interest.
- Demonstrating the superiority of the proposed system to laboratory experts. Autoverification systems supplement laboratory experts, who must necessarily trust them to function at least as well as a laboratory expert.

1.3 Outline of this Dissertation

This dissertation describes the development and evaluation of the synthetic error generation method in order to create a laboratory autoverification system that outperforms existing systems and laboratory experts.

The dissertation is organized as follows:

- *Chapter 1: Introduction* – a brief overview of the significance of autoverification systems along with impediments to their successful development.
- *Chapter 2: The Clinical Laboratory: Processes and Errors* – a description of the clinical laboratory process including sources for errors and the methods used to detect laboratory errors. We will discuss pre-analytical, analytical, and post-

analytical errors and how errors are detected using internal consistency checks, delta checks, and extreme values.

- *Chapter 3: Bayesian Networks: Overview and Operation* - a review of Bayesian networks as utilized in this dissertation including structure learning, parameter learning, and inference in both discrete and conditional Gaussian networks. We will guide the reader through parameter learning and making inferences with examples specific to the medical domain.
- *Chapter 4: Class Imbalance: Standard Solutions* - a discussion of class imbalance problem and its impacts on the clinical laboratory autoverification domain as well as standard methods for addressing the class imbalance problem. Class imbalance in the clinical laboratory domain is more extreme than is observed in other domains, and introduces unique problems that inhibit the effective training of autoverification systems.
- *Chapter 5: Synthetic Minority-Class Generation* – a description of the synthetic error generation method with performance characteristics over a wide range of model parameters. Our approach, given the unique and extreme class imbalance problem, is to create part of the training dataset synthetically, thereby better training the autoverification system.
- *Chapter 6: Comparison of Synthetic Error Method Against Standard Methods* – the results of statistical comparisons between the domain-specific synthetic error-generation method and the standard methods for detecting laboratory errors demonstrating that better training results in better performance.

- *Chapter 7: Comparison of Synthetic Error Generation Against Laboratory Experts* – survey results from laboratory experts of their utility for laboratory error detection systems and a statistical comparison against the synthetic error-generation method demonstrating the superiority of a Bayesian network autoverification system training with synthetic errors.
- *Chapter 8: Summary and Conclusions* – a summary of the dissertation with limitations and future work.

1.4 Conventions and Notations

The following words are used frequently within this dissertation and, therefore, it is critical that their use be understood:

- **Error** – An error is the difference between an analyte's true value and its measured value.
- **Dichotomous system** – a system containing two mutually exclusive and exhaustive classes or labels. For example, a clinical laboratory result is either acceptable or not.
- **Minority-class** – the class with small frequency in a dichotomous system.
- **Majority-class** – the class with larger frequency in a dichotomous system.
- **Between-class Imbalance** – when one class is significantly less frequent than the other class.

- **Small disjuncts** – when the class disjunct is small, the overlap between classes is substantial making class differentiation difficult.
- **Within-class Imbalance** – when one class contains subclusters with significantly different frequencies.
- **Class imbalance** – a general term describing between-class imbalances, class disjuncts, and within-class imbalances.
- **Autoverification system** – a system that automatically reviews laboratory results and identifies those results that are possibly in error.

Chapter 2: The Clinical Laboratory: Processes and Errors

The clinical laboratory is a critical part of the health care system, providing services to hospitals and local clinics as well as to researchers investigating novel drugs and treatment regimes. The timely identification and resolution of errors is critical to the laboratory's ability to provide quality results. While, as we will see in section 2.2, most laboratory errors occur outside of the analytical process, the term "laboratory error" encompasses any error within the process described in section 2.1. Within the laboratory process cycle, the laboratory is often best suited to detect indications of an erroneous result, discussed in section 2.3, either via an expert's review or via an automated program. We start with an overview of the clinical laboratory process.

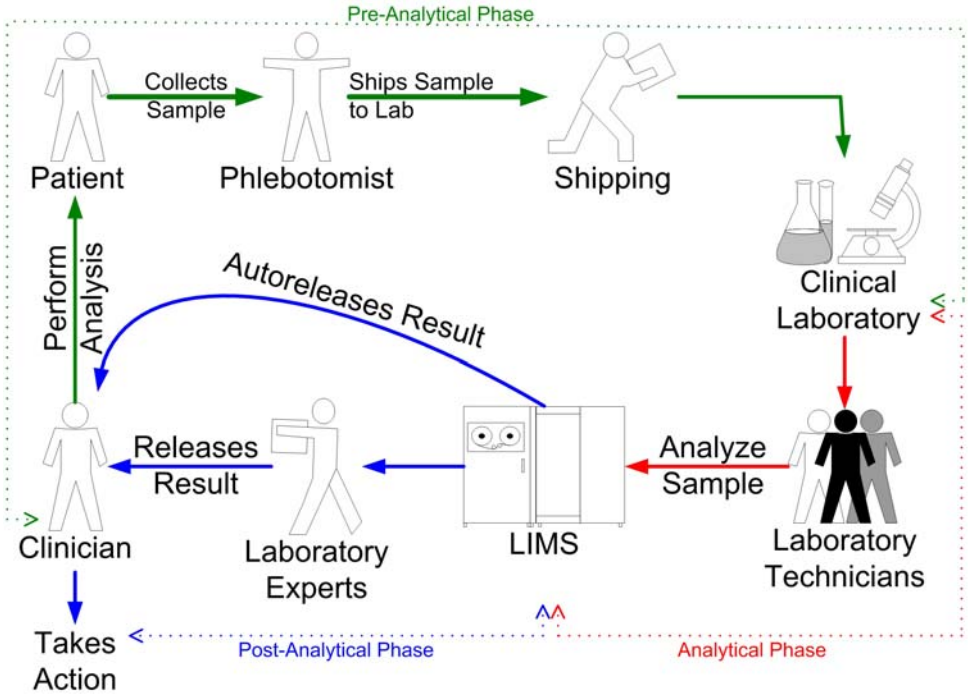

Figure 2.1 Clinical Laboratory Process

2.1. Overview of Laboratory Process

The clinical laboratory process or cycle, Figure 2.1, shows the process of the clinical laboratory that starts and ends with the clinician. The clinician may be in a hospital, a local office, a research environment, or similar, and decides to perform some laboratory analysis on a patient. Depending on the analysis, the patient may need to fast for several hours before the phlebotomist is able to collect the sample. The phlebotomist has procedures that they must follow, which include proper patient identification, assessing analysis-interfering conditions such as non-fasting, collecting the specimen with the proper gauge needle, using a test-appropriate type of container, and labeling the vial (University of Utah 2007). Once the specimen is collected, it generally needs to be processed. This includes letting it sit for a specific period of time, centrifuging, aliquoting the plasma or serum, etc. Once processed, the sample is shipped or delivered to the laboratory where the sample is logged into their Laboratory Information Management System (LIMS), identifying the patient and requested tests. The laboratory may need to re-label the specimen with its own accessioning number, which adds another potential source for error. Laboratory technicians analyze the specimen and results are entered into the LIMS, typically directly from the analyzer. Laboratory experts or an autoverification system reviews the results of the analyses and releases the results for reporting back to clinician, who interprets the results and may make take some action regarding the patient's treatment. The part of the cycle from when the clinician orders the test to when the sample is received by the laboratory technicians is called the pre-analytical phase and is indicated in green. The analytical phase, indicated

in red, covers the portion of the cycle from the laboratory technicians' receipt of the sample to when the results are released for reporting to the ordering clinician. The post-analytical phase, indicated in blue, is the portion of the process occurring after results have been released for reporting. As discussed next, errors may occur at any point along this process.

2.1.1 Scope of Applicability

Errors occur at points within the clinical laboratory process and errors are also caught at points along the process as well. A diligent login staff can catch many sample quality errors, such as low volume, and improperly identified samples or requisitions. Autoverification systems are concerned with detecting errors once a specimen is logged into the Laboratory Information System (LIS) and given to the laboratory technicians for analysis. Laboratory analysis may be performed using automated analyzers or may be performed using a manual procedure. The result of the analysis may be numeric (cholesterol: 220 mg/dl) or non-numeric (Apo E Genotype: E3/E3). The autoverification system described herein is designed only to detect errors in numeric data. Furthermore, in order to hypothesize that a given result is in error, we must assume that the analytical result can be predicted. The autoverification system discussed herein is not meant to be an all-encompassing autoverification solution, but rather, is meant to be a layer in a laboratory's error defense system. The clinical laboratory must continue to use proper quality control procedures and, where appropriate, rule-based autoverification systems (Westgard 2004).

2.1.2 The Canonical Clinical Laboratory

In 1999, there were approximately 170,000 clinical laboratories in the United States with hospital laboratories performing 60% of the testing, commercial laboratories performing 30%, and doctors' offices performing 10% (Pollack 2001). The clinical laboratory process in each of these three setting is different. For example, in a doctor's office, a nurse may collect and process the specimen, perform the analysis, and provide results to the clinician using a hand-written note. A hospital laboratory may have staff perform much of the specimen processing occurring in the pre-analytical phase whereas a commercial laboratory may perform virtually none of the pre-analytical processing. With such a wide range of settings, we define a canonical clinical laboratory as our setting. This canonical clinical laboratory is considered to use the following process:

1. Receives samples from the client along with a requisition form identifying the patient and desired analyses.
2. Logs the samples into their LIS, including available patient demographic data.
3. Labels the samples with a unique bar-coded sample accession number.
4. Provides samples to the technicians, who place the sample on an automated analyzer.
5. The automated analyzer, if capable, reads the bar-code, queries the LIS for the desired analyses, performs the analyses, and reports the result to the LIS via an interface. Less capable analyzers may require the technician to enter results manually or identify the sample accession number.

6. The laboratory technician checks results ensuring quality controls are acceptable and reasonable values. If warranted, the laboratory technician will contact the client with critical values. Once checked, the laboratory technician posts the results to the LIS.
7. Laboratory experts and/or autoverification systems evaluate the results, designating suspect results to be repeated or investigated and releasing the rest for reporting.
8. Results are reported to the client via any combination of email, postal mail, fax, automated interface, phone, remote printers, etc.

The canonical clinical laboratory autoverification system detects errors when the value is transferred from the automated analyzer to the LIS or after the laboratory technician has posted the value.

2.1.3. The Role of Autoverification in the Clinical Laboratory

Clinical laboratories use autoverification systems, such as the one described herein, to screen laboratory results for potentially erroneous results. Laboratory directors are in many cases required, by law and by licensing organization, to ensure that their autoverification systems operate as required (California Assembly 2006; Commission on Laboratory Accreditation 2006). One requirement is for the autoverification system to flag for expert evaluation results that exceed some predefined threshold, which is usually defined by clinical significance. These thresholds are easily set using rule-based autoverification systems. However, detecting possible errors and hypothesizing

as to its origin, is best performed using a probabilistic system, not a rule-based system. In practice, the clinical laboratory will need to implement both a rule-based system to flag results exceeding predefined thresholds and a probabilistic system to detect erroneous results. The system described herein is a probabilistic system and as such, is designed to detect when an analytical result differs significantly from the expected value.

2.2. Sources of Laboratory Errors

Table 2.1 lists the three phases of the clinical laboratory process, along with representative errors common to each phase. The phases are pre-analytical (Wiwanitkit 2001), analytical (Witte, VanNess et al. 1997), and post-analytical (Stroobants, Goldschmidt et al. 2003). A recent review found tremendous variability between laboratories, but estimated that about two-thirds of these errors occur in the pre-analytical stage, one-sixth in the analytical stage, and one-sixth in the post-analytical stage (Bonini, Plebani et al. 2002). The pre-analytical phase contains errors originating from the patient, such as being mis-identified or non-fasting, and errors due to improper sample collection and processing, such as using the wrong type of collection container or poor sample quality. Analytical errors are those originating in the analytical sections of the clinical laboratory and include analyzer errors, sample mis-handling, data entry errors. Post-analytical errors are those errors occurring after the results have been released for reporting back to the clinician and include excessive turn-around-time of results. Autoverification systems generally do not address post-analytical errors except

for turn-around time: an accurate autoverification system can review results quickly, holding only suspect results from reporting, so that the turn-around time is as short as possible.

Table 2.1 Three phases of the laboratory testing process with representative errors

Pre-Analytical	Analytical	Post-Analytical
• Patient misidentified • Incorrect collection vial • Inappropriate specimen quality (clotted, low volume, hemolysis, etc.) • Specimen mishandled • Delay in shipment	• Sample switch • Instrument error • Procedural error • Dilution error • Quality control failure • Data entry error • Delay in analyzing	• Results misidentified • Physician not notified of problem • Results misinterpreted • Delay in reporting

2.2.1 Errors in Context

As is clear from Table 2.1, laboratory errors occur in several contexts. Pre-analytic errors occur in the context of specimen collection, processing, handling, and logging into the clinical laboratory's LIS. Analytic errors occur during the analysis of the specimen. Finally, post-analytic errors occur within the context of communication between laboratory and clinician or within the clinical practice setting. A careful analysis of workflow and processes within these contexts could reveal "root causes" of errors. However, because laboratories, clinics and hospitals represent numerous largely independent entities, what represents a cause of error at one location may not represent a cause at another. For example, neither the measurements of insulin or HbA1c are

standardized, which results in a wide range of analysis protocols and well as a wide range of analytical variability (Hoelzel, Weykamp et al. 2004; Marcovina, Bowsher et al. 2007).

Another approach to incorporating context into the analysis of error would be to build models that predict error around laboratory value contextual annotations (e.g., "sample hemolyzed"). Such an approach would target directly the causes of error within the context of laboratory data collection and analysis. However, such annotations, if available, may not be usable for detecting errors due to their dependence on the analytical method and reagents used in the analysis (Jay and Provasek 1993). Furthermore, some critical types of annotations would never be available. For example, one rarely knows if a sample was switched or an instrument failed. Had such information been known in context, then avoiding an error would have been possible. The "contextual annotations" approach is useful in measuring the affect of an intervention aimed at reducing laboratory errors, but, as we will see in Chapter 4, is not useful in building a predictive system to identify those errors.

We approach the problem of context from a third perspective by evaluating the belief in an analyte's value being in error within the context of results of other analytic values. Unlike annotations or workflow analyses, information on other analytic values has virtually no additional cost in reasoning about errors; they are already available in laboratory databases. Analytic values are of course, indicators of biological function, and errors represent exogenous perturbations of these biological indicators that lead to

unusual data patterns. Hence, examining an analyte's value in the context of other biological indicators can influence one's belief that a value is in error. For example, measuring a high fasting glucose and a low glycosylated hemoglobin should increase our belief in an error since such a combination is unlikely.

2.2.2 Qualitative Error Types

Within this method, we attempt to detect errors that affect a single result such as one of the three major types of errors listed in Table 2.2. Value errors are those errors that affect a single result. Such errors have no affect on our belief that any other result is in error. Value errors may be due to transcription mistakes or instrument failures. A value error is, for example, when a technician enters an erroneous cholesterol result of 250 mg/dl instead of an actual value of 150 mg/dl. Sample processing errors are those errors that affect a known set of more than one analytical result for a collected sample and will do so in a predictable manner. For example, hemolysis in a sample will cause the potassium to be higher and the alkaline phosphatase to be lower, than their true values (Jay and Provasek 1993). Finally, a sample switching error occurs when two patient samples are interchanged. For the purposes of this study, however, it is assumed that one result, glucose, comes from one sample and the other result, glycosylated hemoglobin (HbA1c), comes from a different sample such that sample processing errors and sample switching errors only affect one of the two results. This is an appropriate assumption for glucose and HbA1c since glucose is performed using serum or plasma while HbA1c is performed using whole blood. Laboratory errors, such as the types

listed here, come about from any one of a number of steps and perturb the results leading to unusual data patterns that can be detected.

Value Error	Sample Processing Error	Sample Switch
• Instrument failure • Data-entry error	• Incorrect specimen collection vial • Inappropriate specimen quality • Specimen mishandled	• Vial interchange

Table 2.2 The Three Major Qualitative Error Types

2.3. Indications of Laboratory Errors

The affect of laboratory errors in creating unusual data patterns can be very subtle, obvious, or more commonly somewhere in-between. For example, if two patients selected at random have his or her glucose specimens interchanged, each patient is expected to have a glucose measurement near the mean value and the resulting error is expected to be near zero. However, if measuring potassium and the specimen is grossly hemolyzed, the inappropriate specimen quality error is obvious because the measured potassium value will greatly exceed a value compatible with life and the specimen will be a bright cherry red instead of a normal clear. Further confounding the identification of laboratory errors is biological and instrument variability. For example, cholesterol has a 6.0% within-subject biological variability and is expected to be measured within 8.9% of the true value set by a reference laboratory (Ricos, Alvarez et al. 1999; Centers for Disease Control and Prevention 2004). While many errors can be proven, there is, in general, no definitive way to prove that a given value is in error.

2.3.1 Limited Gold Standards for Error Identification

The qualitative types of errors listed in Table 2.2 reflect both human and instrument failures. Typically, these failures go undetected leading to an error. Thus, naturally occurring errors are often not evident to the observer. At a great expense in human capital, human experts are usually capable of identifying most gross laboratory errors. As with other domains in the medical sciences, however, human experts should not be viewed as a "gold standard" because they do not have sufficient sensitivity or specificity. A critical function experts provide is deciding whether a suspected error is significant and worth investigating. Human experts evaluating a dataset are often able to detect and remove most sizeable errors from a dataset. When working off such a "cleaned" dataset, the only error for which there exists a gold standard are those knowingly introduced by the researcher (i.e., synthetic errors). Synthetic errors are errors that have been deliberately introduced by the researcher via some rule or set of rules to facilitate the study of error identification. The set of rules used to synthesize errors often represents an analogue to the naturally occurring error process. For example, sample-switching errors can be modeled effectively by randomly switching a proportion of samples within a cleaned dataset, and transcription errors can be modeled by randomly changing the digits of single analyte values. This approach yields criterion-based datasets with a gold standard for error identification as given by the record of the deliberate introduction of errors to the data. In our method, we use a synthetic-error generation process as the gold standard when creating the training data set.

2.3.2 Delta Checks

Delta checks compare the patient's current results with their previous results and provide an estimate on the plausibility of the observed changes (Ladenson 1975). They are well established and understood by laboratory experts. By using the assay's coefficient of variability along with the biological coefficient of variability, a laboratory can compute changes, called reference change values, that indicate a statistically significant change (Fraser, Stevenson et al. 2002). This approach may be satisfactory if the patient is seen often, has not changed his or her lifestyle, has had no changes in treatments, has had no changes in any of his or her afflictions, has not aged significantly, and the analytes being measured have low biological variability. As the patient's condition gets further away from this ideal, the usefulness of delta checks as traditionally implemented diminishes. Since we use a conditional Gaussian Bayesian network in our method, the incorporation of nodes representing historical results can be included and the structure and parameter learning algorithms presented in Chapter 3 will identify when changes are expected and when they are not. For example, patients with a high cholesterol value are usually treated and, therefore, the expected change is large and negative.

2.3.3 Internal Consistency

Checking internal consistency involves examination of pathophysiologically related variables using empirically derived rules, intuition, or training (Boran, Given et al. 1996). For example, Rohlfing et al. (2002) researched the relationship between fasting

plasma glucose and glycosylated hemoglobin and concluded that a predictable relationship existed between them. As glycosylated hemoglobin increased, however, the variability in glucose increased substantially. Biological variation, especially in morbid patients, may limit the usefulness of the internal consistency approach (Boran, Given et al. 1996). For example, an analysis of the NHanes data from the 2004 survey indicates the relationship between glucose and glycosylated hemoglobin changes as a patient transitions from non-diabetic to pre-diabetic, to diabetic (Centers for Disease Control and Prevention (CDC) 2004). The method presented here properly handles the uncertainty due to biological variation and can adapt for disease states.

2.3.4 Extreme Values

If a specimen is grossly hemolyzed then the measured potassium value will be extreme and instantly identified as an error. The range of values that, outside of which constitutes an extreme value, depends on the setting as well as one's tolerance. For example, the New Cook County hospital in Chicago uses a range of 80 – 450 mg/dl to review cholesterol results and a range of 60 – 325 mg/dl to review glucose results (Torke, Boral et al. 2005). By checking for extreme values, some errors can be identified, but these are often only the most obvious errors. Conditional Gaussian Bayesian networks easily detect extreme values because extremes are improbable.

2.4 Detecting Laboratory Errors

Methods employed by a typical clinical laboratory to detect errors, such as those listed in Table 2.2, in laboratory results have not changed substantially over the years even though technology in general, has grown rapidly. The method for validating patient results is to evaluate the results for indications of laboratory errors either via a review by seasoned laboratory experts, or by a computer algorithm called an autoverification system. Currently, about 48% of clinical laboratories have a rule-based autoverification system implemented (American Association for Clinical Chemistry 2007). When reviewing data, both the autoverification system and the laboratory experts estimate the believability of results based on the internal consistency of the data and against delta checks (Boran, Given et al. 1996). Below we provide a brief overview of the current approaches to laboratory error detection.

2.4.1 Laboratory Experts

Laboratory experts are effective large error detectors, but invariably get fatigued, are interrupted, or just make a mistake. Studies of rule-based autoverification systems have suggested they can reduce technologist review time by about 40% and achieve greater accuracy (Crolla and Westgard 2003; Torke, Boral et al. 2005). This is influencing a move towards automated systems (Crolla and Westgard 2003). According to a 2007 American Associate for Clinical Chemistry survey, approximately 52% of clinical laboratories use experts to review data prior to release (American Association for Clinical Chemistry 2007). Our preliminary research suggested that laboratory experts

are not as sensitive or specific as a Bayesian network trained using the method presented here and this result, discussed in Chapter 7, was confirmed.

2.4.2 Rule-Based Experts

Current systems that implement artificial intelligence based approaches to detecting laboratory errors are generally based on rules. A commercial rule-based autoverification system, VALAB, was compared to a panel of nine clinical chemists at the St. Elisabeth Hospital in The Netherlands and was shown to be more sensitive, though significantly less specific, than the experts in identifying intentionally altered data (Oosterhuis, Ulenkate et al. 2000). However, VALAB requires tens of thousands of rules and a complex weighting system (Valdiguie, Rogari et al. 1996). In addition, VALAB's underlying rules are proprietary and are not open to inspection (Oosterhuis, Ulenkate et al. 2000). The tremendous number of rules in modern autoverification systems renders these systems brittle and unmanageable and their proprietary nature renders their logic opaque, but this is not the primary objection to using rule-based systems to detect laboratory errors. The primary objection to rule-based systems in the laboratory error context is that they are not able to reason abductively (from evidence of error to belief in hypothesis about error). Rule-based systems have difficulty reasoning bi-directionally. As we will demonstrate in Chapter 3, Bayesian networks by virtue of their dual probabilistic and graphical framework are able to reason from evidence to hypothesis and from hypothesis to evidence (Wright and Ayton 1994).

2.4.3 Detecting errors via probabilistic methods

Oosterhuis, Ulenkate, & Goldschmit (2000) developed a method call LabRespond by which correlated laboratory tests are examined for patterns. LabRespond compares observed versus expected patterns as an indicator of the likelihood of the observed data. Such an indicator can be used as an error threshold. Oosterhuis (2000) has shown that the error identification rate ranged between 24% and 71% with this method. As with the current dissertation, they too used synthetic errors for the analysis. In their research they demonstrated that LabRespond is comparable to VALAB in its performance (Oosterhuis, Ulenkate et al. 2000). Our preliminary research demonstrated a conditional Gaussian Bayesian network that outperformed LabRespond in detecting laboratory errors.

2.5. Summary

Current approaches used to detect laboratory errors are not effective. Delta checks will flag large changes in a laboratory analysis, but fails to consider when large changes are expected such as a patient starting a new medication. Internal consistency checks will flag results that inconsistent with other observations, but may be limited in morbid patients. Both laboratory experts and rule-based systems are capable of using delta checks and internal-consistency checks to identify laboratory errors, but our preliminary research suggests that Bayesian networks are capable of significantly outperforming current approaches as well as provide hypotheses for the error. An overview of Bayesian networks with a discussion of structure learning, parameter learning, and

making inferences is provided next followed by a critical problem, class imbalance, in the clinical laboratory domain that stymies the development of effective autoverification systems.

Chapter 3: Bayesian Networks: Overview and Operation

The purpose of this chapter is to familiarize the reader with Bayesian networks as utilized later in the dissertation. This is not intended to be a complete discussion of Bayesian networks. For a more thorough explanation, the interested reader is referred to the following books though only Neapolitan and Cowell cover Gaussian networks:

- Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference (Pearl 1988)
- Causality: Models, Reasoning, and Inference (Pearl 2000)
- Learning Bayesian Networks (Neapolitan 2004)
- Probabilistic Networks and Expert Systems (Cowell, Dawid et al. 1999)
- Bayesian Networks and Decision Graphs (Jensen 2001)

Within the domain of biomedical artificial intelligence, researchers have increasingly utilized Bayesian networks because of their strength in properly handling uncertainty. Early researchers, such as Ted Shortliffe with his MYCIN project, attempting to incorporate uncertainty into the omnipresent rule-based systems of the day, were unable to accomplish this feat without imposing significant restrictions on that system due to semantic deficiencies of the language (Pearl 1988). Specifically, rule-based systems, which Pearl refers to as “extensional” systems, are not able to accurately infer bidirectionally, retract evidence, or properly handle correlated evidence (Pearl 1988). Bayesian networks, in contrast, do not have this limitation and offer here a formal

calculus for quantifying belief that a laboratory error has occurred. In this chapter, we start with an overview of Bayesian networks and then examine parameter learning, structure learning, and inference in discrete and conditional Gaussian Bayesian networks.

3.1 Overview of Bayesian networks

A Bayesian network consists of a graph, with some limitations, and joint probability distributions. The graph contains nodes, representing variables such as cholesterol, and directed edges between nodes representing a causal relationship between the variables (Pearl 1988). A limitation of the Bayesian network's graph is that cycles are prohibited. Under the Bayesian network framework, directed acyclic graphs faithfully represent probabilistic relationships and concisely describe probability distributions over the states of variables in the network. A sample Bayesian network, derived from a portion of the NHanes demographic database, is displayed in Figure 3.1 (Centers for Disease Control and Prevention (CDC) 2004). The four nodes in this graph correspond to an individual's gender, race, education level, and income. A square is used to indicate a discrete variable and a circle indicates a continuous variable. Directed edges between the nodes represent statistical correlations and can be defined by subject experts, computer algorithms, or both. The lack of an edge between gender and any other variable indicates that, for the NHanes respondents, gender does not co-vary with education, race, or income. The directed edge from the Race node to the Education node indicates that one's race affects the probability of achieving a level of education.

Note that the use of directed edges enables the system to represent real-world cause and effects: race affects the probability of attaining a level of education; one's education level does not affect one's race. A causal interpretation for the graph, while irrelevant for the prediction process, is important for the laboratory experts, who must evaluate the models during their evaluation process. A model that is not causally sound may not be trusted. Joint probability distributions, called the parameters of a Bayesian network, can similarly be defined by subject experts, learned programmatically, or both. For the Bayesian network defined by Figure 3.1 and Table 3.1 - Table 3.5, a dataset was created from the 2003-2004 NHanes demographic data file DEMO_C.xpt with all missing values removed (Centers for Disease Control and Prevention (CDC) 2004). Using a program called "deal", the structure of the Bayesian network was determined (Bøttcher and Dethlefsen 2003). Finally, a program called "Netica" was used to determine the parameters or joint probability distributions (Norsys Software Corporation 2006). By using a formal calculus, described next, one can reason over the Bayesian network and make inferences, as demonstrated later in this chapter for a discrete and conditional Gaussian Bayesian network.

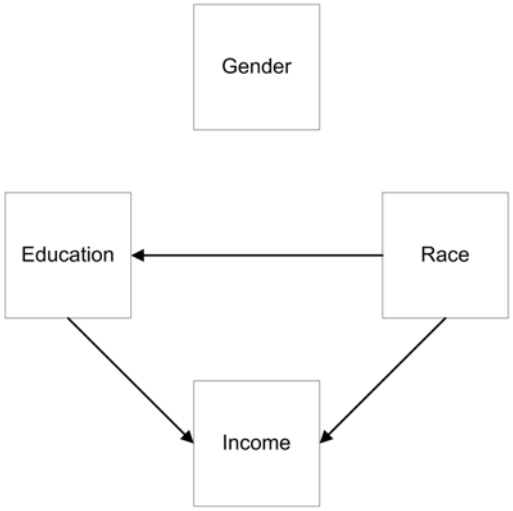

Figure 3.1 Bayesian Network from NHanes Demographics

Table 3.1 Joint Probability Distribution of Gender

Female	Male
51.1%	48.9%

Table 3.2 Joint Probability Distribution of Race

Black	Hispanic	Other	Other-Hispanic	White
26.6%	23.7%	3.1%	3.1%	43.5%

Table 3.3 Joint Probability Distribution of Education Level by Race

	Black	Hispanic	Other	Other-Hispanic	White
Less than High School (HS-)	65.0%	75.4%	47.9%	62.0%	34.6%
High School or equivalent (HS)	13.1%	12.4%	11.7%	14.5%	22.8%
More than High School (HS+)	21.9%	12.2%	40.4%	23.5%	42.6%

Table 3.4 Joint Probability Distribution of Income Level by Race and Education

	Race:	Black			Hispanic			Other		
	Education:	HS-	HS	HS+	HS-	HS	HS+	HS-	HS	HS+
Annual Household Income	0-4,999	3.3%	3.5%	3.3%	3.6%	2.5%	3.8%	1.6%	5.3%	5.6%
	5,000-9,999	9.6%	9.6%	4.8%	6.3%	5.1%	3.0%	1.6%	7.9%	3.7%
	10,000-14,999	12.0%	9.6%	6.5%	13.5%	13.9%	6.4%	9.6%	7.9%	7.5%
	15,000-19,999	9.7%	8.2%	5.4%	8.8%	8.9%	6.0%	12.8%	18.3%	8.4%
	20,000-24,999	9.3%	10.6%	9.1%	10.6%	8.9%	12.0%	12.0%	5.3%	2.8%
	25,000-34,999	14.5%	16.1%	14.1%	18.5%	18.0%	15.8%	16.8%	5.3%	11.3%
	35,000-44,999	9.4%	12.4%	11.3%	13.8%	15.6%	11.5%	8.8%	7.9%	10.3%
	45,000-54,999	8.2%	9.2%	8.0%	10.3%	11.0%	7.7%	9.6%	13.2%	7.5%
	55,000-64,999	4.7%	6.0%	6.3%	4.6%	6.3%	4.7%	4.0%	7.9%	6.5%
	65,000-74,999	3.2%	3.5%	5.4%	1.7%	3.0%	7.3%	2.4%	10.5%	9.3%
	75,000+	16.1%	11.3%	25.8%	8.3%	6.8%	21.8%	20.8%	10.5%	27.1%

Table 3.5 Joint Probability Distribution of Income Level by Race and Education – continued

	Race:	White			Other-Hispanic		
	Education:	HS-	HS	HS+	HS-	HS	HS+
Annual Household Income	0-4,999	1.2%	1.5%	1.7%	1.3%	4.4%	1.5%
	5,000-9,999	4.7%	4.6%	2.9%	7.5%	2.2%	4.5%
	10,000-14,999	10.4%	9.2%	5.3%	13.8%	8.9%	7.5%
	15,000-19,999	9.5%	10.1%	5.3%	10.0%	13.3%	3.0%
	20,000-24,999	9.6%	11.3%	5.3%	10.0%	13.3%	6.0%
	25,000-34,999	9.3%	15.3%	10.6%	16.7%	20.1%	16.4%
	35,000-44,999	8.8%	11.9%	11.3%	15.0%	8.9%	10.4%
	45,000-54,999	9.1%	7.8%	10.0%	6.3%	6.7%	6.0%
	55,000-64,999	7.4%	6.7%	7.4%	7.5%	2.2%	4.5%
	65,000-74,999	5.9%	6.0%	6.9%	2.5%	6.7%	6.0%
	75,000+	24.1%	15.6%	33.3%	9.4%	13.3%	34.2%

3.1.1 Bayes' Theorem and Conditional Probability

In this section we discuss the fundamental probability theory used to reason with a discrete Bayesian network such as the one depicted in Figure 3.1. Bayes' Theorem, named for the Reverend Thomas Bayes (1702 -1761), is used to compute conditional probabilities as well as posterior probabilities. If we observe some fact F, what is the probability of event E? If this probability is difficult to measure, then equation 3.1 provides a means to compute this probability using the probability of observing the event, P(E), the probability of observing the fact, P(F), and the probability of observing the fact given that the event has occurred, P(F|E).

$$P(E \mid F) = \frac{P(F \mid E)P(E)}{P(F)} \tag{3.1}$$

If the event and the facts are independent, then observing fact F has no impact on the probability of observing the event E. This concept is expressed in equation 3.2.

$$P(E \mid F) = P(E) \tag{3.2}$$

If F takes on a set of n mutually exclusive and exhaustive set of states, then equation 3.3 may be used to calculate the probability of event E.

$$P(E) = \sum_{i=1}^{n} P(E \mid F_i)P(F_i) \tag{3.3}$$

These simple formulae are the basis of making inferences with Bayesian networks and enable the development of a Bayesian autoverification system.

3.1.2 d-Separation

An important concept in Bayesian networks is d-separation, which defines how knowledge or the lack of knowledge blocks the flow of information between two nodes. Consider the case of three nodes that are connected to form a path. The four possible arrangements of this graph are depicted in Figure 3.2. For any path, the flow of information is blocked if one of the two following conditions are met (Pearl 2000):

1. The path contains a chain or a fork and the state of the middle node is known.
2. The path contains a collider and the states of the middle node and all its decedents are not known.

If, given some knowledge (or lack of knowledge in the case of a collider) about the state of nodes in the network, all paths between two nodes are blocked, then those nodes are said to be d-separated (Pearl 2000). D-separation is critical in both learning the structure of the Bayesian network and making inferences.

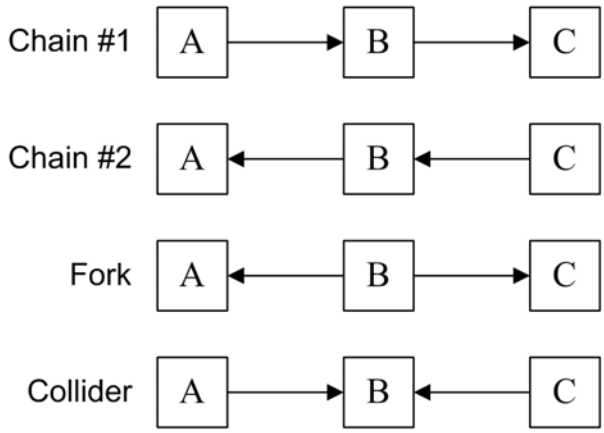

Figure 3.2 Possible Arrangements of Three Singly Connected Nodes

3.1.3 Markov Equivalence

The two chains and fork graphs depicted in Figure 3.2 contain the same conditional independence – given knowledge of node B's state, node A is conditionally independent of node C. These three graphs are clearly not identical, but are indistinguishable statistically. This equivalency is termed Markov Equivalence. When using a heuristic to determine the structure of the graph, one is unable to differentiate between two Markov-equivalent structures. Markov equivalence hinders the ability to consider a learned structure as a causal representation because directionality cannot always be determined. However, a human expert is often able to select the model that best fits reality and can infer causation from directed edges.

3.2 Discrete Bayesian Networks

Discrete Bayesian networks are a common form of Bayesian networks due to their strengths, which include a capability to represent any arbitrary probability distribution within the limits of the number of discrete states. In this section, we will overview the steps required to identify the structure of the directed acyclic graph, learn the parameters or joint probability distributions, and make inferences based on observations. These steps, when applied to a properly created training database, yields a Bayesian network that is able to function as an autoverification system.

3.2.1 Structure Learning

The directed acyclic graph part of the Bayesian network must faithfully represent the conditional independencies contained within the joint probability distributions. In addition, the graph must correspond with reality in that causal connections expressed by the graph need to be temporally sound and model plausible causal connections. For example, the state of a node today cannot affect the state of another node in the past and variables such as race cannot be affected by one's education level. In general, domain experts, an algorithm, or a combination of the two, define the structure of a Bayesian network.

Domain experts may, based upon their expertise, define the Bayesian network's structure by inserting directed edges between nodes to indicate causal actions and their direction of effect. In contrast to the experts, an algorithm does not know a priori the

conditional independencies among the variables. A naïve approach would be to use a training dataset and try all possible models to find the one most likely given the training dataset. However, the number of candidate models grows more than exponentially as the number of nodes increases with a 10 node system having over 4.2 x 10^{18} possible models (Neapolitan 2004). Identifying the most likely model is NP-Hard (Chickering, Geiger et al. 1994). Therefore, a heuristic is used to guide the search through the domain of possible models, scoring each tested model using some criterion, and selecting the best model when some stopping criteria are met. To assist the evaluation of possible models and to ensure a realistic model, experts may define a temporal ordering of the variables as well as identify intrinsic variables, such as gender or race, which cannot be causally influenced by another variable. The selected model will be one of a family of Markov equivalent models and will be a good, but not necessarily the best, model given the training dataset and user-imposed restrictions. The structure of the Bayesian network is a representation of the training dataset and indicates the statistical covariations used to identify laboratory errors. Next, the parameters are learned in order to be able to enable making inferences.

3.2.2 Parameter Learning

As observed in the structure learning phase, the parameters of the Bayesian network may be determined by domain experts, algorithms, or a combination of the two. Algorithmic parameter learning, in contrast to structure learning, in the discrete case is far simpler and computationally tractable using any one of several algorithms. When

there is little missing data in the training dataset and no hidden nodes in the directed acyclic graph, a common approach by virtue of its speed and simplicity is a method based on the Dirichlet distribution, which Netica calls “counting-learning” (Norsys Software Corporation 2006). If there are significant missing data or hidden nodes, then other approaches such Expectation-Maximization or gradient descent are appropriate (Mitchell 1997). The dataset created in earlier Section 3.1 does not contain hidden nodes or any missing data, so the counting-learning method is appropriate for this dataset.

If one were to estimate a joint probability distribution simply from the observed frequency, one would have a biased estimator and would be unable to estimate a confidence range (Mitchell 1997). In addition, an unobserved condition would have an estimated probability of 0.0, which would prevent meaningful inference. Therefore, a Dirichlet distribution, a family of continuous multivariate probability distributions parameterized by a vector of non-negative real numbers, is used to represent the distribution of parameters of each node under the assumption that the parameters of each node are independent (Geiger and Heckerman 1997). The basic process for learning the joint probability distributions is to start with an assumption of the prior probabilities, which are usually initially uniform, to indicate no prior knowledge. As data are observed, the probabilities are then updated to reflect the additional knowledge.

For example, consider the Gender node in the directed acyclic graph of Figure 3.1. Assuming a prior belief that the probability of observing a male is the same as the

probability of observing a female, the expected prior probability of observing a male is simply 0.50. If we now observe new cases, while our estimated posterior probability of observing a male may change after observing the data, our confidence in that probability estimate will change. We use the equation 3.4 to estimate the posterior probability of observing a male given our new data where the variables n_{Male} and n_{Female} represent our prior knowledge (usually set to 1) and n'_{Male} and n'_{Female} simply count the number of males and females, respectfully, observed in the new data. If we had greater prior confidence that the genders were equally probable, then larger, but equal, values for n_{Male} and n_{Female} would be used. Similarly, if the prior probability were other than 1:1, we would use a different ratio.

$$P(Male \mid data) = \frac{n_{Male} + n'_{Male}}{n_{Male} + n'_{Male} + n_{Female} + n'_{Female}} \qquad (3.4)$$

This process is used to learn the parameters for the other nodes in the directed acyclic graph with the difference that the equations are conditioned on the parents. For example, in determining the parameters of the Education node, the above equations would be repeated for each of the five possible values for Race. The parameters of the Bayesian network hold the joint probability distributions and once the parameters of the Bayesian network have been learned, one can make inferences to identify laboratory errors.

3.2.3 Inference

Inference is the process of prediction based on a partial observation of the state of the world. For example, one may use the Bayesian network depicted in Figure 3.1 to estimate a person's household income level based upon observing their race and education level. However, one could also reason as to the likelihood a person with a given household income, completed high school. Inference in a Bayesian network may be accomplished simply by application of Bayes rule and probability theory, though not as efficiently as other algorithms such as Pearl's message-passing algorithm or a Junction tree Algorithm (Lauritzen and Spiegelhalter 1988; Pearl 1988). In general, the computational complexity of inference in a Bayesian network has been shown to be NP-Hard, which means that worst-case performance would very likely require non-polynomial time (Neapolitan 2004). For the NHanes Bayesian network described above, we will demonstrate the process of inference in answering question.

Question: Given a person with a household income between $25,000 - $34,999, what is the posterior probability the person is Hispanic.

Answer: From Table 3.2 we note the prior probability of observing a Hispanic in the NHanes dataset is 23.7%. In order to calculate the probability of observing a Hispanic given that their household income is between $25,000 and $34,999, we use the following steps:

1. The desired probability is P(Race = Hispanic | Income = 25,000 – 34,999), which for simplicity we will write as: $P(R = r \mid I = i)$. In addition, we use the variable E to indicate the education node and e to indicate an individual's education level.
2. Using Bayes' Theorem, we obtain the relationship:

$$P(R = r \mid I = i) = \frac{P(I = i \mid R = r)P(R = r)}{P(I = i)}$$

3. Using Table 3.2, we look up the probability of being Hispanic, $P(R = r)$, as 23.7%.
4. To compute the probability of a household income of between $25,000 and $34,999 given that the person is Hispanic, we use equation 3.3 to sum over the education levels, as demonstrated below, and obtain a result of 18.1%.

$$P(I = i \mid R = r) = \sum_x P(I = i \mid R = r, E = e_x)P(E = e_x \mid R = r)$$

5. To compute the probability of a household income of between $25,000 and $34,999 we again use equation 3.3 but now sum over all combinations of race and education level, as demonstrated below, and obtain a result of 14.0%.

$$P(I = i) = \sum_x \sum_y P(I = i \mid R = r_y, E = e_x)P(E = e_x \mid R = r_y)P(R = r_y)$$

6. Finally, the posterior probability for the probability of observing a Hispanic given that one's household income is between $25,000 and $34,999 is calculated as:

$$P(R = r \mid I = i) = \frac{0.237 \times 0.181}{0.140} = 30.7\%$$

As demonstrated above, one is able to reason over a Bayesian network bidirectionally using Bayes' Theorem and make sound inferences. Given a dataset of discrete variables, one is able to identify a structure for the Bayesian network, learn the parameters of the joint probability distributions, and finally, to make bidirectional inferences. In the clinical laboratory, it is often desirable to leave continuous data as continuous rather than discretizing it. We next consider the case when all of the variables are continuous and can be modeled as a Gaussian distribution.

3.3 Conditional Gaussian Bayesian Networks

The discrete Bayesian networks described in the previous section provide a way to model processes and, once the structure is determined, provide an exact and generally efficient means for making inferences. However, if the underlying process involves continuous variables, the discretization process will result in the loss of some accuracy even when using an optimal discretization algorithm (Friedman and Goldszmidt 1996). While this loss in precision can be compensated, to some degree, by increasing the number of bins that data are discretized into, the increase in computational complexity

limits the effectiveness of this approach (Cowell, Dawid et al. 1999). If the process can be modeled using a conditional Gaussian Bayesian network, then the discretization step is eliminated and the network can still be solved exactly. Figure 3.3 shows a simple conditional Gaussian Bayesian network in three variables that represents the relationship between cholesterol, high-density lipoprotein (HDL) cholesterol, and triglyceride. Each node has a mean value, μ, and a standard deviation, σ, where the variance is the square of the standard deviation. Edges between nodes have a weight, b, which is used in the inference process. After providing an overview of Gaussian systems below, we will overview structure learning, parameter learning, and inference in the conditional Gaussian Bayesian network.

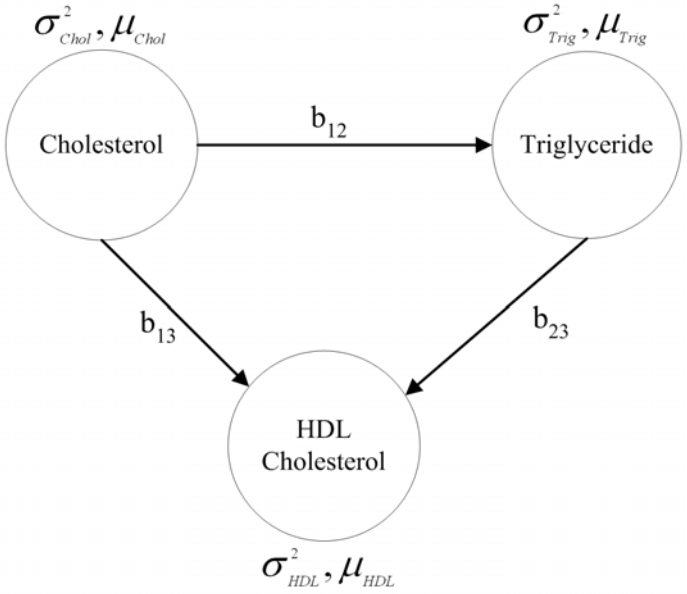

Figure 3.3 Example Conditional Gaussian Bayesian Network

3.3.1 Gaussian Systems

A linear Gaussian model is one where the value of each continuous node depends linearly on the values of the parents (Nachman 2004). The Gaussian Bayesian network, as with the discrete Bayesian network, consists of a directed acyclic graph and joint probability distributions, which are described by a vector of weights for each edge into a node, a vector of variances, and a vector of mean values. Such a representation is equivalent to a multivariate Gaussian system, with a vector of mean values and covariance matrix, which allows for ease of inference and parameter learning as the representations can be converted back and forth (Nachman 2004).

The obvious limitation of conditional Gaussian Bayesian networks is that they are only able to model linear relationships between normal variables but as discussed later, for the purposes of identifying laboratory errors, they often outperform Bayesian networks with discretized data. For example, in a discrete Bayesian network the cholesterol data would need to be discretized into some number, typically between four and eight, of bins. Assuming six bins with equal width and a possible range from 100 to 400 mg/dl, each bin would be 50 mg/dl wide. The Bayesian network would have a limited ability to detect errors smaller than 50 mg/dl since these errors may not cause a change in the discreet value and, therefore, there would be no change in the probability of the observed value.

A conditional Gaussian Bayesian network may have discrete variables, though usually with the restriction that a discrete variable may not have a continuous node as a

parent or ancestor (Lauritzen and Jensen 2001; Lerner, Segal et al. 2001). Figure 3.4 shows such a network where the continuous node HDL has a mean and variance for males and a different mean and variance for females. If gender is not known, then the HDL node's expected value and variance will be a function of the gender-specific values. In order to use the Bayesian network to detect laboratory errors, as with the discrete Bayesian network, we identify its structure, learn the parameters, and then make inferences.

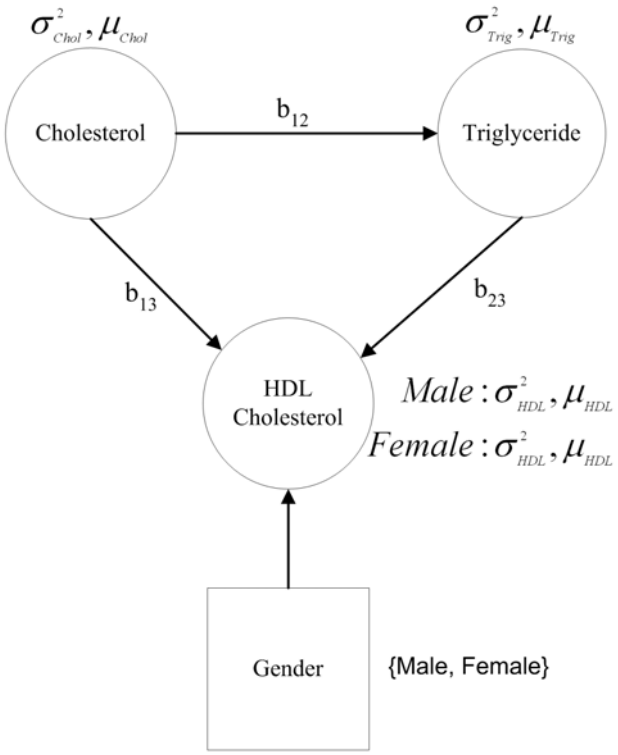

Figure 3.4 Example Mixed Discrete-Gaussian Bayesian Network

3.3.2 Structure Learning

As with discrete Bayesian networks, the directed acyclic graph portion of a Gaussian Bayesian network must represent the conditional independencies contained within the joint probability distributions and model plausible causal connections. In addition, as with the discrete case, the structure may be determined by a domain expert, via an algorithm, or a combination of the both. Since the problem of structure learning is NP-Hard, a heuristic combined with a scoring metric is used to drive the search through the domain of possible models. Once the structure of the Gaussian Bayesian network is identified, the parameters are learned in order to make inferences.

3.3.3 Parameter Learning

Algorithms to learn the parameters of a Gaussian Bayesian network rely on its ability to be transformed into a nonsingular multivariate normal distribution and back to a Gaussian Bayesian network (Geiger and Heckerman 1994). Geiger and Heckerman (1994) first developed the following algorithm, which Neapolitan (2004) subsequently refined, for learning the parameters of the Gaussian Bayesian network. If discrete variables are present, then Gaussian parameters are learned for each combination of values for the discrete parents and the joint probability distributions of discrete nodes are learned as discussed in section 3.2.

Let μ_i be the mean value of node i; σ_i^2 be the variance of the node i; b_{ji} be the multiplication factor in the edge from node j to node i and equals b_{ij}; Σ be the positive-definite covariance matrix, which equals the inverse of the precision matrix, Γ; Γ be the precision matrix and equals the inverse of the covariance matrix, Σ; υ be the size of the hypothetical database from which our prior estimates of the mean values are derived, which usually has an initial value of 0 to indicate the lack of a prior belief in the means; α be the degrees of freedom and is nominally set to υ-1; and let $\beta = \frac{\upsilon(\alpha - n + 1)}{\upsilon + 1}\Gamma^{-1}$, where n is the number of random variables, be a helper term.

Each cell in the covariance matrix is a function of the edge weights, b, and node variances, σ^2, in the directed acyclic graph. This may also be expressed as in equation 3.5, where entries in one cell depend on the values of cells above it, if off the diagonal, and to the left of it, if on the diagonal. For example, the entry in cell (2,1), $b_{12}\Sigma_{11}$, could also be written as $b_{12}\sigma_1^2$, but we choose the former representation due to its compact representation. The basic process in parameter learning is to first compute a covariance matrix from the training dataset and then solve for the edge weights and variances using standard matrix operations.

$$\Sigma = \begin{pmatrix} \sigma_1^2 & -- & -- \\ b_{12}\Sigma_{11} & \sigma_2^2 + b_{12}\Sigma_{12} & -- \\ b_{13}\Sigma_{11} + b_{23}\Sigma_{12} & b_{13}\Sigma_{12} + b_{23}\Sigma_{22} & \sigma_3^2 + b_{13}\Sigma_{13} + b_{23}\Sigma_{23} \end{pmatrix} \quad (3.5)$$

Assuming no prior knowledge concerning the parameters' values, the process to learn the parameters of the conditional Gaussian Bayesian network is as follows:

1. Without prior knowledge, the initial parameter values are:
   a. μ^* = (0,0,…,0)
   b. υ = 0
   c. α = -1
   d. β = 0
2. Calculate the updated parameter values:
   a. μ^* is a vector of unconditional means calculated from the training data.
   b. $\upsilon^* = \upsilon + M$ where M is the size of the training database.
   c. $\alpha^* = \alpha + M$
   d. $s = \sum_{h=1}^{M} (x_h - \bar{x})(x_h - \bar{x})^T$ where x_h is each tuple in the dataset and $\bar{x}$ is the vector of unconditional means.
   e. $\beta^* = \beta + s + \frac{\upsilon M}{\upsilon + M}(\bar{x} - \mu)(\bar{x} - \mu)^T = s$ when no prior knowledge since β =0 and υ =0.

3. Calculate the updated precision matrix, Γ^* via:

$$(\Gamma^*)^{-1} = \Sigma^* = \frac{\upsilon^* + 1}{\upsilon^*(\alpha^* - n + 1)}\beta^*$$

4. Calculate the parameters, using the covariance matrix of the complete graph and the matrix Σ^* to solve for the variances and edge weights. If the original graph is not a complete graph, then the calculation of the variances and edge weights is repeated for each ancestral ordering of the variables.

We now apply this method to a 2004 NHanes dataset created by combining the files L13_c (contains cholesterol and HDL cholesterol), L10_am (contains triglyceride), and demo_c (contains gender) (Centers for Disease Control and Prevention (CDC) 2004). The dataset was limited to the 3,433 tuples with complete data and the natural log was taken of triglyceride value in order to normalize the distribution. We assume no prior knowledge of the parameters of the Gaussian Bayesian network, and use the structure of Figure 3.3 that was determined by deal (Bøttcher and Dethlefsen 2003). The ancestral ordering of this Bayesian network is cholesterol, triglyceride, and HDL cholesterol. We compute the updated values μ^* = (183.8, 4.60, 55.0), υ^* = 3433, α^* = 3432, $\beta^* = S$, and S as:

$$s = \begin{pmatrix} 6,076,807 & 38,182.5 & 409,632.7 \\ 38,182.5 & 1,037.2 & -8,767.1 \\ 409,632.7 & -8,767.1 & 743,455.9 \end{pmatrix} = \beta^*$$

We can then compute the updated covariance matrix as:

$$(\mathrm{T}^{*})^{-1} = \Sigma^{*} = \begin{pmatrix} 1{,}771.7 & 11.13 & 119.4 \\ 11.13 & 0.302 & -2.56 \\ 119.4 & -2.56 & 216.8 \end{pmatrix}$$

Using the updated covariance matrix and solving for the parameters in equation 3.5, we calculate the following:

$$\sigma^{2}_{Chol} = 1{,}771.7,\ \sigma^{2}_{Trig-\ln} = 0.372,\ \sigma^{2}_{HDL} = 179.3$$

$$\mu_{Chol} = 183.8,\ \mu_{Trig-\ln} = 4.60,\ \mu_{HDL} = 55.0$$

$$b_{Chol \to Trig-\ln} = 0.00628,\ b_{Chol \to HDL} = 0.1233,\ b_{Trig-\ln \to HDL} = -8.89$$

Having learned the parameters of the Gaussian Bayesian network, we are now able to perform inferences.

3.3.4 Inference

Inference in a conditional Gaussian Bayesian network, as in the discrete network, is NP-Hard so any exact method may take non-polynomial time to complete the calculations, which can be alleviated in more complex networks by using heuristics (Neapolitan 2004). Within this dissertation, we are only interested in predicting a single node after observing the states of all other variables in the directed acyclic graph. This greatly

simplifies the inference process required to predict a value. In the presence of missing or unobserved data, a more complex inference algorithm, such as the described in detail in Lauritzen and Jensen (2001), is required. Using Figure 3.3, to compute the expected mean and variance of HDL cholesterol given values for cholesterol and triglyceride, we would simply sum the product of each parent node with the appropriate edge weight. The variance, since the state of all parent nodes is observed, is the variance for the HDL cholesterol node calculated in the parameter-learning step.

3.4 Summary

Bayesian networks are a powerful tool for making bi-directional inferences under uncertainty, such as observed in the clinical laboratory. Whether the dataset consists of just discrete variables, just continuous variables, or a combination of the two, as long as certain restrictions are satisfied, a Markov-equivalent structure may be programmatically identified using the algorithms discussed. Once the structure is identified using an algorithm or experts, the parameters, or joint probability distributions, are learned. Finally, by using the formal calculus discussed for making inferences, a system can be built to make inferences and quantify the belief that a laboratory error has occurred. Bayesian networks are an appropriate decision support tool for making inferences in the clinical laboratory, but they, like many other decision support tools, suffer when there is a large disparity between the sizes of the classes.

Chapter 4: Class Imbalance: Standard Solutions

Class imbalance, a condition when one class is more frequent than the other class, is a problem that affects all supervised learning algorithms, negatively affecting their performance. The term "class imbalance" herein will be used to refer to the combined problem of between-class imbalance, small disjuncts, and within-class imbalance. In this chapter we discuss the problem of class imbalance and explore why it is a vexing problem for classification systems. We then discuss current solutions used to ameliorate the class imbalance problem such as minority-class over-sampling and majority-class under-sampling. Finally, we overview a standard method, receiver operating characteristic (ROC) curves, used to measure the performance of a classification system. As we will show, a natural dataset, as opposed to synthetic dataset that we will cover in Chapter 5, of laboratory errors is a very poor choice for using to train an autoverification system. A natural clinical laboratory dataset contains an extreme between-class imbalance, disjuncts that range from very small to huge, and large within-class imbalances.

4.1 Description of Class Imbalance

A dataset is said to be balanced when there are approximately equal percentages of each class. For simplicity, we will assume a dichotomous classification problem such as in the clinical laboratory where each result is classified as error-free or in error. If the percentage of error-free results was approximately equal to the percentage of results in

error, then the classes are said to be balanced. As the percentages of the classes diverge, for example in a clinical laboratory where it is estimated that 99% are error-free and 1% are in error, the dataset is said to become imbalanced (Bonini, Plebani et al. 2002). Standard machine learning approaches typically treat misclassification costs equally and attempt to minimize the overall misclassification error percentage while simultaneously minimizing the complexity of the model (Wang and Japkowicz 2004). Therefore, in the presence of a class imbalance, as we will discuss next, machine learning algorithms such as decision tree algorithms and Bayesian networks will perform poorly. Typical methods used to ameliorate the difficulty in machine learning approaches in the class-imbalance domain include: 1) over-sampling, either directed or random, the minority class; 2) under-sampling, either directed or random, the majority class; 3) adjustment of misclassification costs; 4) single-class classifier (Japkowicz and Stephen 2002). However, recent research has shown that the class imbalance problem, as we will see in the following sections, is more complex than just the relative class balance, but also includes small class disjuncts and within-class imbalance (Japkowicz and Stephen 2002; Japkowicz 2003). Furthermore, these attributes and their affect on classifier performance are not independent factors, but rather interact in a non-linear method.

4.1.1 Between-Class Imbalance

Between-class imbalance refers to the relative proportion of one class compared to the other class. The predominant class is termed to be the major class and the rare class is

termed to by the minor class. Consider a classifier, such as the C4.5 decision tree classifier, that examines a training dataset and develops rules in order to classify entries in the dataset as accurately as possible. The C4.5 algorithm recursively divides the training dataset creating a decision tree where each leaf either consists entirely of one classification or, if the size of the leaf is below a preset limit to avoid overfitting, is assigned the classification of the predominant class within that leaf (Quinlan 1993). Consider the case of the clinical laboratory where the training dataset consists of 99% error-free results and 1% error-containing results. When the C4.5 algorithm reaches its minimum leaf size, assuming equal misclassification costs, that leaf would need to contain at least 50% erroneous results in order for the leaf to be labeled as "erroneous". However, given that the prior probability of observing an error-containing result was only 1%, this requires the leaf node to represent a condition, as determined by the path from the root to the leaf node, where errors are 50 times as likely as the default condition. Anything less than that level will result in the leaf and its contents being labeled as error-free results. While performance of the C4.5 algorithm may be improved by using unequal misclassification costs, in extremely unbalanced datasets the minority class elements may simply not provide a sufficient estimate of the boundary layer between the two classes.

Consider the case depicted in Figure 4.1, which shows a correlated system in two dimensions that is similar to the correlation between glucose and glycosylated hemoglobin (HbA1c). The majority-class, shown as blue circles, was created by randomly selecting an X value from a Gaussian distribution with a set mean and

variance. The Y value was computed as $Y = X + \varepsilon$ where ε is randomly drawn from a Gaussian distribution with mean of 0 and a small variance. The minority-class, shown as red diamonds, was created in exactly the same manner except that the Gaussian noise term ε had a mean of +2 or -2. This training dataset contains 9.1% minority class elements and 90.9% majority class elements, so the classes are clearly not balanced. The classification problem is to learn, in general terms, using glucose and HbA1c those combinations are likely to be errors and which are likely to be non-errors.

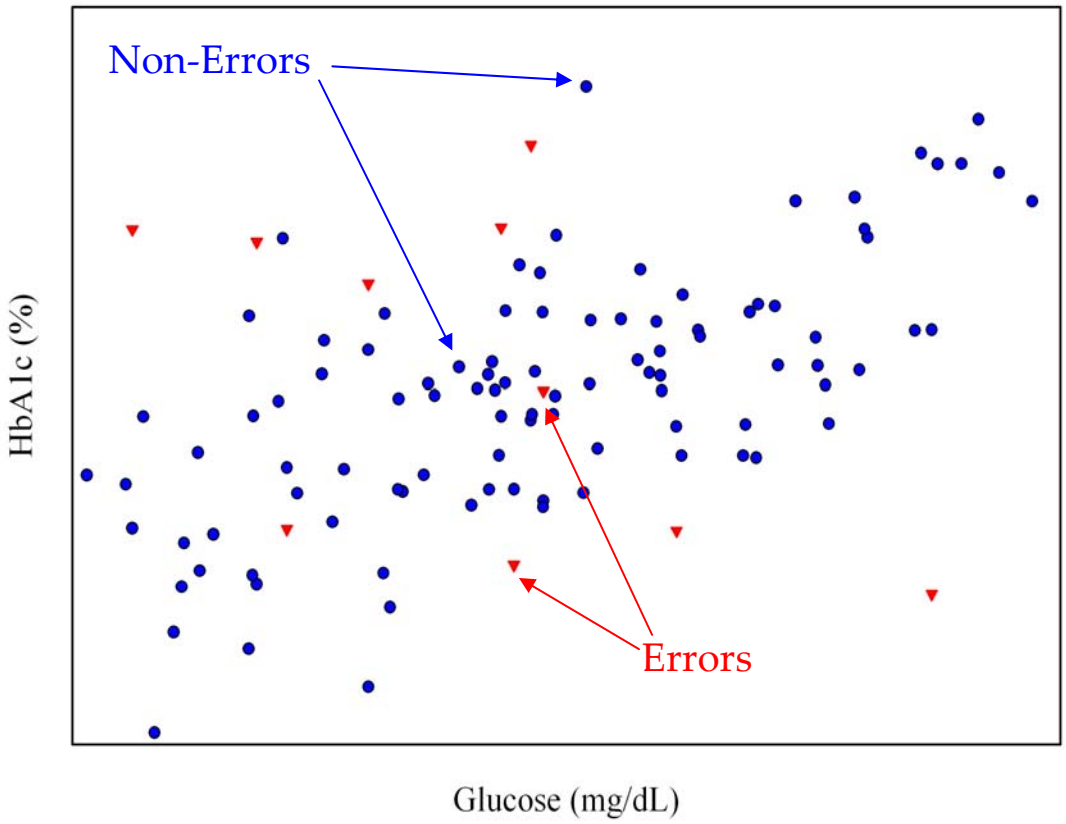

Figure 4.1 Class Imbalance Resulting in Poor Boundary

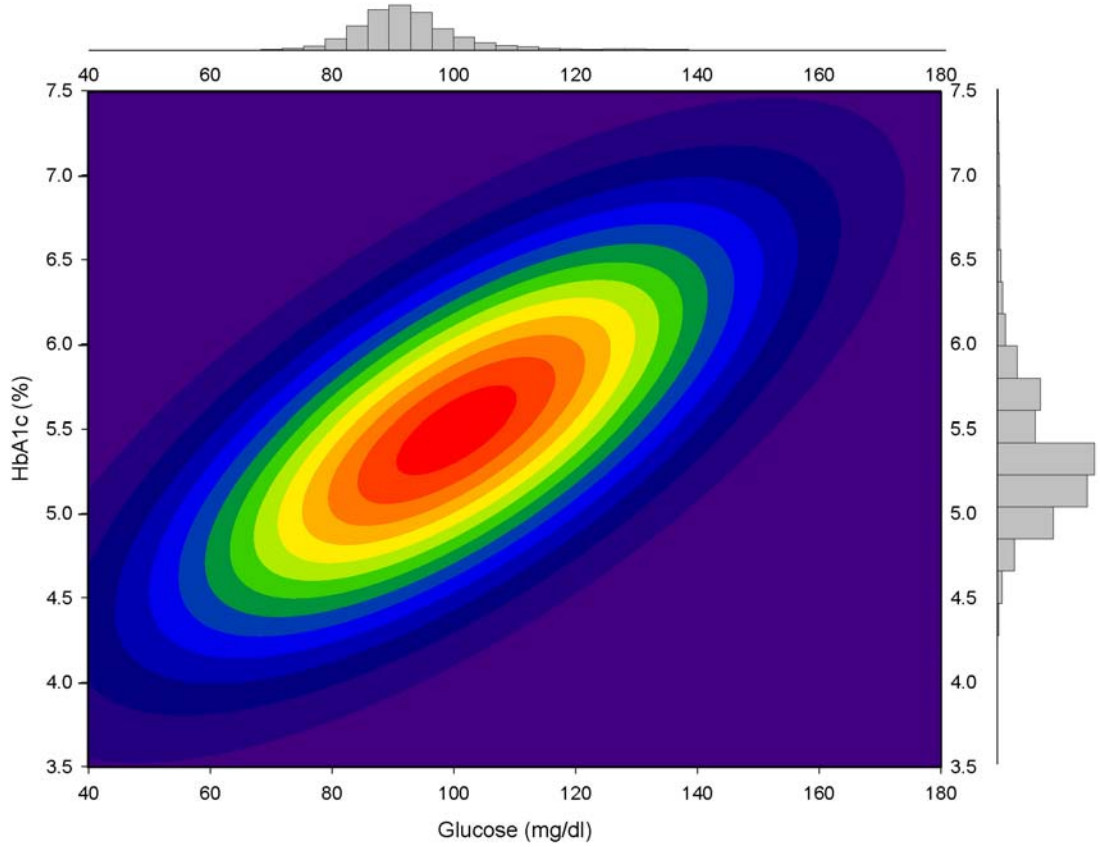

Figure 4.2 NHanes Glucose and HbA1c Correlation

Since the underlying generating function is a simple multivariate Gaussian distribution, the density distribution of the majority class should resemble the ellipse in Figure 4.2 (Centers for Disease Control and Prevention (CDC) 2004). Likewise, the density distributions of the minority-class should also be elliptical with one ellipse above the majority-class ellipse and the other below the majority-class ellipse. However, since there are so few examples of the minority class, the minority-class boundary within the domain space is poorly defined. A classifier is not able to model the underlying

generating function with so few examples, which results in the classifier over-fitting the few examples and yielding a poorly performing system.

4.1.2 Small Disjuncts

Also visible in Figure 4.1 is some overlap between the two classes in that some of the minority-class data elements are well into the center of the majority-class area. Class-overlap, which is more formally called class disjuncts, refers to the degree to which the two classes are differentiated due to differences in their attributes (Jo and Japkowicz 2004). Without loss of generality to higher dimensional problems, consider the one-dimensional classification problem depicted in Figure 4.3, where the area under each curve represents that class's relative proportion and the height of the curve at a given point is proportional to the probability at that point. This figure may represent true glucose measurements, in green and on the right, and erroneous glucose measurements, in red and to the left, due to a significant failure of the instrument that happens 50% of the time and results in an error of constant magnitude. A classification system attempting to differentiate between these two classes could do so trivially by choosing a point along the problem's single dimension and classifying values less than that cutoff one class, for example in error, and values greater than that cutoff the other class, for example error-free.

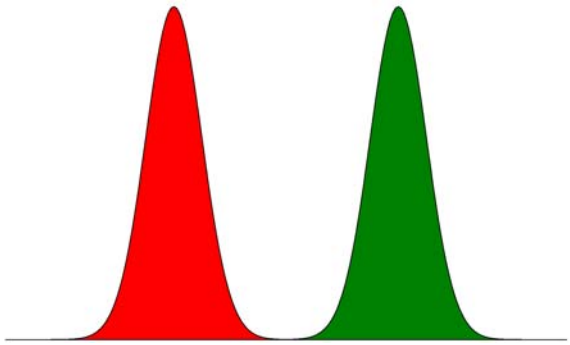

Figure 4.3 Balanced Classes with Large Disjunct

Compare that problem with the one-dimensional problem depicted in Figure 4.4, where erroneous glucose measurements happen only rarely. As with the problem depicted in Figure 4.3, a classifier would be able to easily differentiate the two classes because of the large distinction between the two classes - even though they are significantly imbalanced. As the distinction between the two classes gets smaller and smaller in our simple example, the performance of the classifier will stay relatively constant until the classes begin to overlap, indicating a small disjuncts. When we reach the condition depicted in Figure 4.5, a classifier using equal mis-classification costs will not be able to identify elements of the minority class because for every point along the horizontal dimension, the major class is the more accurate classification. If two classes are disjoint, then a classifier would easily distinguish between the two independently of the class balance. Similarly, if two classes are indistinguishable then the classifier's performance will be poor even when the dataset is balanced. In between those extremes, both class balance and class disjuncts affect performance.

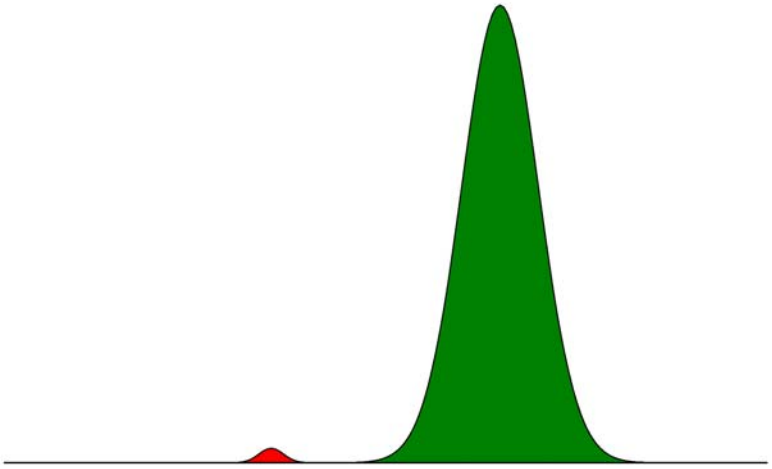

Figure 4.4 Unbalanced Classes with Large Disjunct

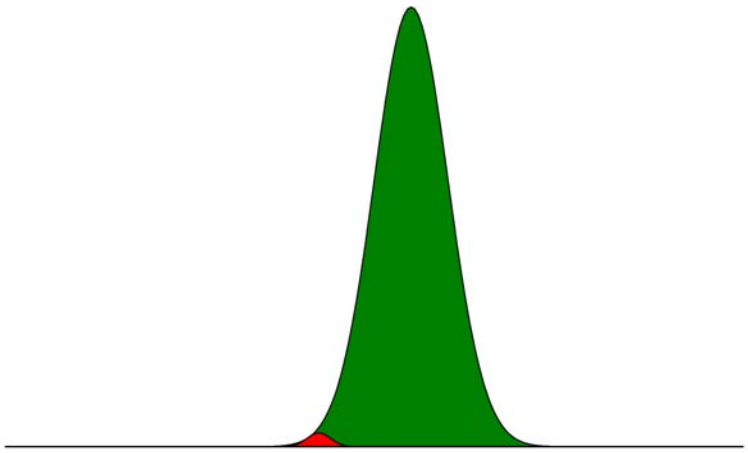

Figure 4.5 Unbalanced Classes with Too Small a Disjunct

4.1.3 Within-Class Imbalances

Within-class imbalance refers to the problem when a class, such as laboratory errors, has subclasses and those subclasses do not contain the same number of examples (Japkowicz 2001). Figure 4.6 depicts a situation when the minor class consists of two

subclasses that are balanced. For example, this may represent glucose with upper erroneous glucose measurements due to the patient not fasting and the lower erroneous glucose measurements due to the patient taking an insulin injection. A classification system would handle each subclass separately and attempt to learn how to identify each separately. However, each subclass is subject to problems due to small disjuncts and/or class imbalances as discussed earlier. As discussed in Chapter 2, the laboratory cycle has many places for errors to occur and many of those errors have a continuum of impacts. For example, a patient who is not fasting will have a measured glucose that is erroneously high. The degree of this error depends, in part, on what the person ate and how long ago. Therefore, the minority class (errors) in a natural laboratory training dataset will consist of many sub-clusters of errors with disjuncts ranging from small to large.

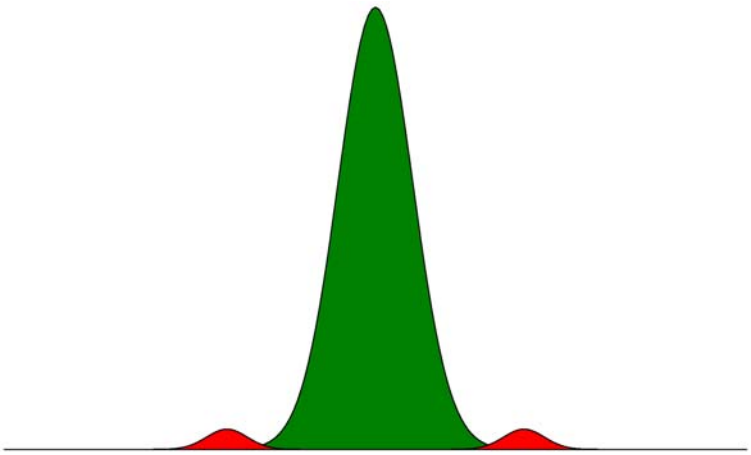

Figure 4.6 Balanced Subclasses

4.2 Solutions to Class Imbalance

Researchers in the data mining field have developed several methods to address the problems caused by class imbalance, small disjuncts, and within-class imbalance, but no clear winner has emerged. Batista (2004) examined eleven approaches on fifteen datasets and found the relative performance of each approach varied over the various datasets. Two significant findings from their work were that random over-sampling, a relatively simple technique, was competitive with more complex approaches and that in the case of an extreme between-class imbalance, approaches that generated synthetic minority-class examples tended to perform best (Batista, Prati et al. 2004). In this section, we review four common approaches to the class imbalance problem and discuss their applicability to the domain of laboratory errors.

4.2.1 Minority-class over-sampling

Over-sampling the minority class involves duplicating minority-class data elements in the training dataset to increase their relative frequency. By over-sampling the minority class, the class imbalance is reduced and performance is typically improved (Weiss and Provost 2001). The minority class examples to be over-sampled may either be selected randomly or via a heuristic. This technique is generally effective in improving the detectability of the minority class but is limited in that it tends to over-fit the training data and increase the computational complexity (Chawla, Japkowicz et al. 2004). An over-fit system is not able to generalize to datasets other than the training dataset. For example, a system might be successfully trained to flag a glucose result of 180mg/dl

when measured with a glycosylated hemoglobin of 4.8%, but then fail to flag a glucose of 190mg/dl when measured with a glycosylated hemoglobin of 4.7% because it did not exactly match the training example. In Figure 4.7 the ten examples of the minority-class have been over-sampled (depicted by their larger size) resulting in balanced classes. In the clinical laboratory domain, there are too few examples of errors, the minority class, which results in an over-fit classifier.

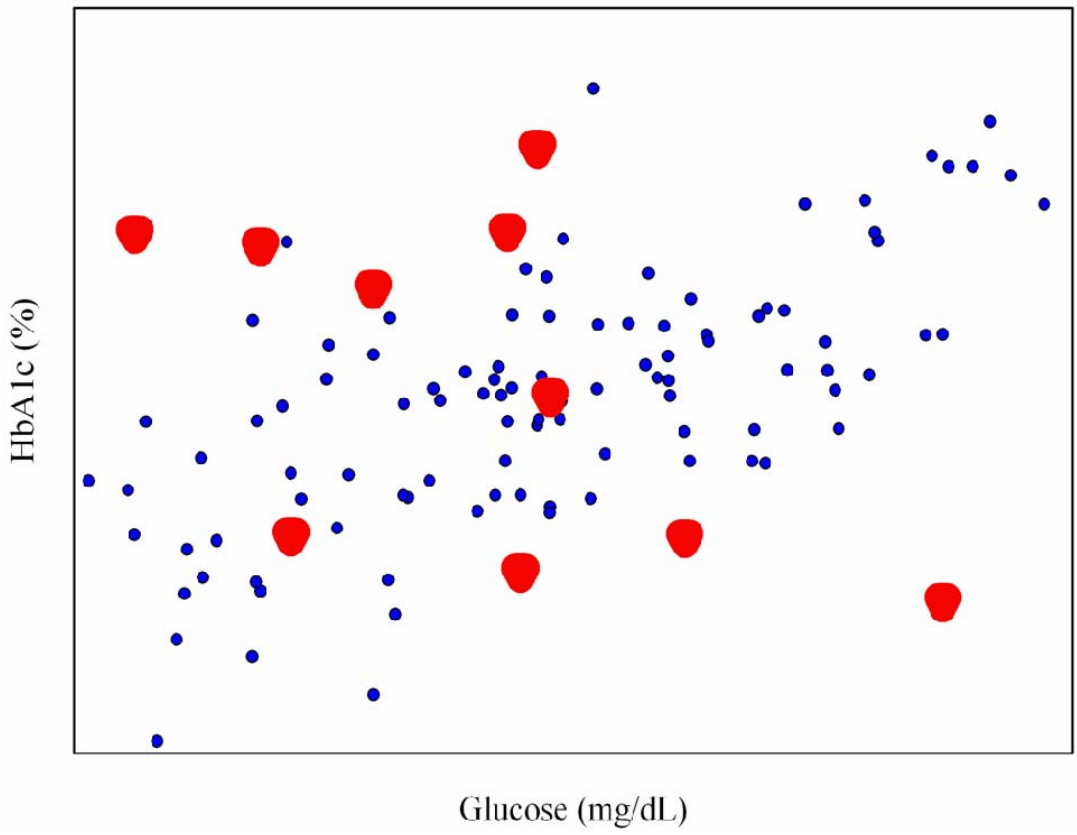

Figure 4.7 Graphical Depiction of Minority-class Over-Sampling

4.2.2 Majority-class under-sampling

In contrast to over-sampling the minority-class, under-sampling the majority class involves the removal of majority-class examples from the training dataset, either randomly or via a heuristic (Chawla, Japkowicz et al. 2004). This process has the potential to remove critical examples resulting in degraded performance (Chawla, Japkowicz et al. 2004). A key advantage of under-sampling the majority-class is that it reduces the computational complexity of the problem, but in the case of the clinical laboratory domain where errors are very rare, this approach removes too many majority-class data elements in balancing the classes, as seen in Figure 4.7.

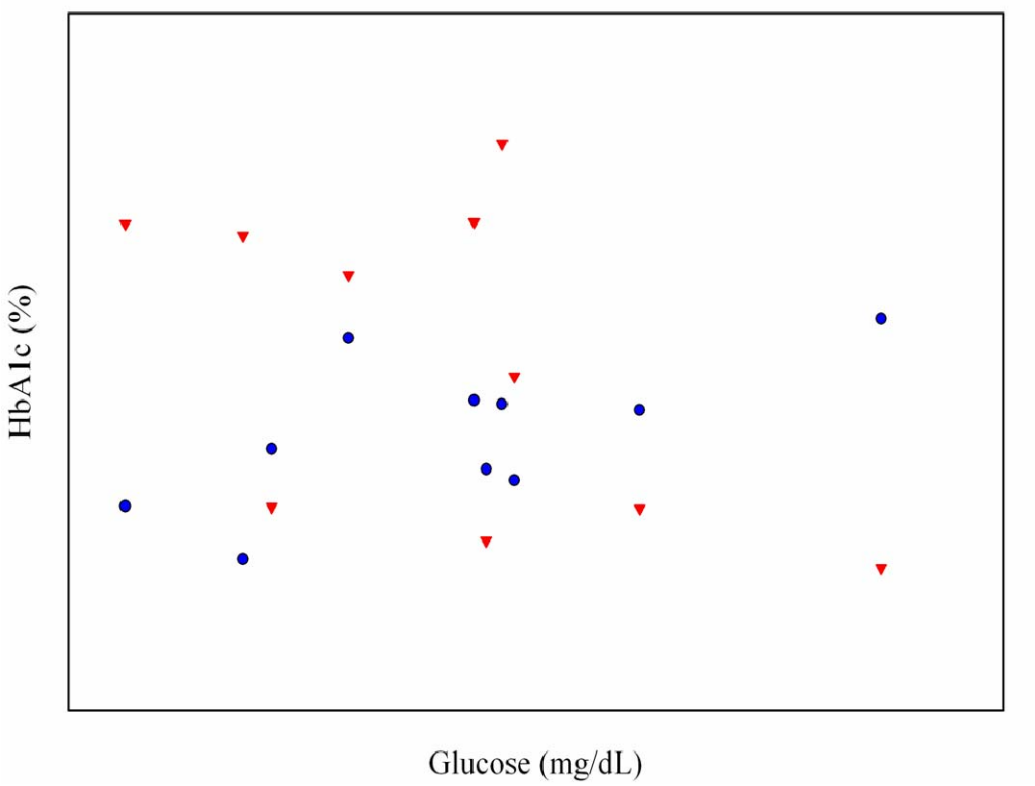

Figure 4.8 Graphical Depiction of Majority-Class Under-Sampling

An approach developed by Nitesh Chawla et al, called Synthetic Minority Over-Sampling Technique or SMOTE, is an attempt to solve the class imbalance problem by combining under-sampling of the majority-class with over-sampling of the minority class via synthetic generation of minority class examples (Chawla, Bowyer et al. 2002). In SMOTE, minority class training examples are created by moving a random distance along a vector from a selected minority class object to one of its nearest minority class neighbors (Chawla, Bowyer et al. 2002). SMOTE, and its derivative algorithms, are not effective when there are too few minority-class data elements or when there are many sub-clusters. In the clinical laboratory, there are very few laboratory errors (minority-class data elements) and a multifactorial source of errors producing many sub-clusters, which impairs the performance of minority-class over-sampling.

4.2.3 Cost Adjustment

Classifiers, in learning how to best classify a training dataset, assign a cost to misclassifying examples, whether they belong to the majority-class or the minority-class, and normally the misclassification cost is equal in both cases. The cost-adjustment process is used to weight the cost of misclassifying a minority-class example to a value greater than that used for a misclassified majority class example. By incurring a higher cost for misclassification of minority-class data elements, the classifier will have a higher sensitivity, but lower specificity, in identifying minority-class objects. As with minority-class over-sampling, cost adjustments are empirically

derived and have the potential to over-fit the training data (Monard and Batista 2002). A comparison of the performance between over/under sampling and cost adjustment under the condition of a class imbalance, indicated no clear winner in general, but that cost-sensitive learning generally outperformed sampling when the number of examples exceeded 10,000 (McCarthy, Zabar et al. 2005).

4.2.4 Single-class classifier

In contrast to the above methods that discriminate between two classes based on learning attributes from a collection of examples and counter-examples, the single-class classifier only learns the attributes of a single class (Japkowicz 1999). Within the context of the clinical laboratory, result sets are compared to the single class, representing acceptable results, and any set not matching the description of that class is classified as not acceptable (erroneous). In the two-variable case, the probability of observing two, correlated variables is determined using a conditional normal distribution to determine the probability of observing a value given the values for the other nodes in the network, $P(Y \mid X)$. In addition, one may elect to use a multivariate normal distribution to calculate the probability of observing all of the variables, $P(X,Y)$. However, a single-class classifier has the limitation that it cannot determine which error is more likely when there is more than a single source of error. This limitation precludes the use of a single-class classifier as an autoverification system.

4.3 Measuring Performance

A central part of the development of any system is in the evaluation of that system. The simple measure of accuracy may seem appropriate when classes are balanced and the misclassification costs are equal, but even under these ideal conditions, accuracy can be misleading. Implicit within the concept of accuracy is a threshold used in the classification, so accuracy only provides a metric of performance at a single classification threshold (Obuchowski 2003). Receiver Operating Characteristic (ROC) curves, which make explicit the classification threshold, are the standard method for displaying and comparing the results of a classification system (Bach, Heckerman et al. 2004).

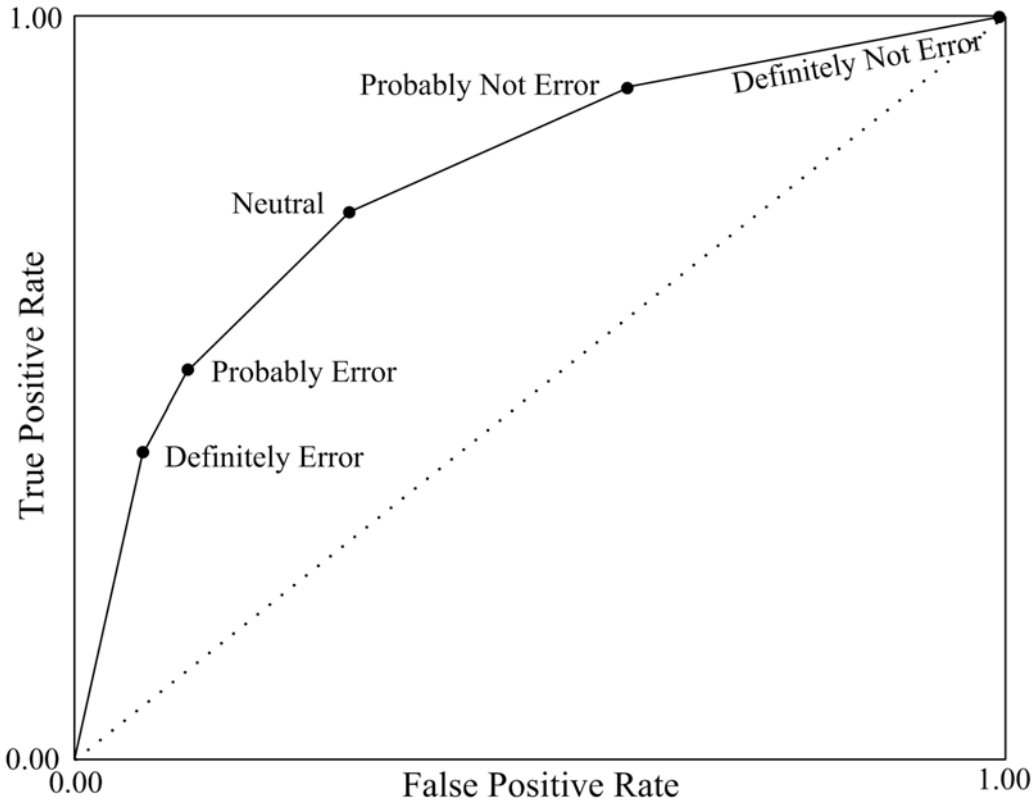

Figure 4.9 Sample Receiver Operating Characteristic (ROC) Curve

4.3.1 Overview of ROC Curves

An ROC curve is a plot of the true positive rate, the fraction of positives (laboratory errors) correctly predicted, versus the false positive rate, the fraction of negatives (acceptable results) erringly predicted as positive, as the classification threshold is varied between all possible points (Flach 2004). An example ROC curve is displayed in Figure 4.9 and models a hypothetical system with five classification thresholds used to classify laboratory results: definitely an error; probably an error; neutral; probably not an error; and definitely not an error. Note that because these thresholds are ordinal, we

can consider each threshold as including lower-ranked thresholds. For example, if one were to use "Neutral" as their decision threshold for classifying a result an error, then any result they were neutral about, thought was probably an error, or thought was definitely an error, would be classified as an error. As the decision threshold is increased, the percentage of true positives and the percentage of false positives can stay the same or increase, but can never decrease. In addition, the curve always starts at (0.00, 0.00) and always ends with (1.00, 1.00) even if no decision threshold yields that point. The area under this curve provides a single metric, AUC for "area under curve", for evaluating and comparing classifier performance. Furthermore, because of the use of true positive and false positive rates, the ROC curve and its AUC are insensitive to class imbalances.

4.3.2 Selecting the Optimal Classification Threshold

The dashed diagonal line in Figure 4.9 represents the performance characteristic of a classifier that is guessing, which corresponds to an area under the curve of 0.50. Classifiers with an AUC of 1.00 are perfect and classifiers with an area greater than 0.50 represent classifiers that perform better then guessing. An ROC curve is generated using the full range of classification thresholds, but in operation a classifier would use only a single classification threshold. To select the optimal point on the curve we return to the concept of accuracy, which is defined as the ratio of the number correctly classified (both positive and negative) to the total number of cases classified (Flach 2004). Iso-accuracy lines are lines of constant accuracy and the slope of each line is

equal to the ratio of negative cases to positive cases (Flach 2004). Since the ROC curves are convex, as depicted in Figure 4.10, the optimal point is selected by moving the iso-accuracy line, with constant slope, down the descending diagonal and finding the location on the ROC curve where the line is first tangential to the curve (Flach 2004). When the classes are balanced and the misclassification costs are equal, the slope of the iso-accuracy line is 45 degrees. As the classes become increasing imbalanced, assuming the percentage of true cases is less than the percentage of negative cases, the slope of the iso-accuracy line gets steeper and steeper and the optimal operating point moves to the left. As the cost of misclassification becomes unequal, assuming the cost of misclassifying a true positive is larger than the cost of misclassifying a true negative, the slope of the curve becomes less and the optimal operating point moves to the right. In the clinical laboratory domain, the classes are very imbalanced and the misclassification costs are very unequal, resulting in a significantly more complex decision in selecting the optimal operating point. In practice, the clinical laboratory may elect to obtain a desired sensitivity or a desired specificity based upon their business model. However, the operating point is determined by the clinical laboratory, they must balance the sensitivity and specificity within the limits of the classification system, as determined by the receiver operating characteristic curve.

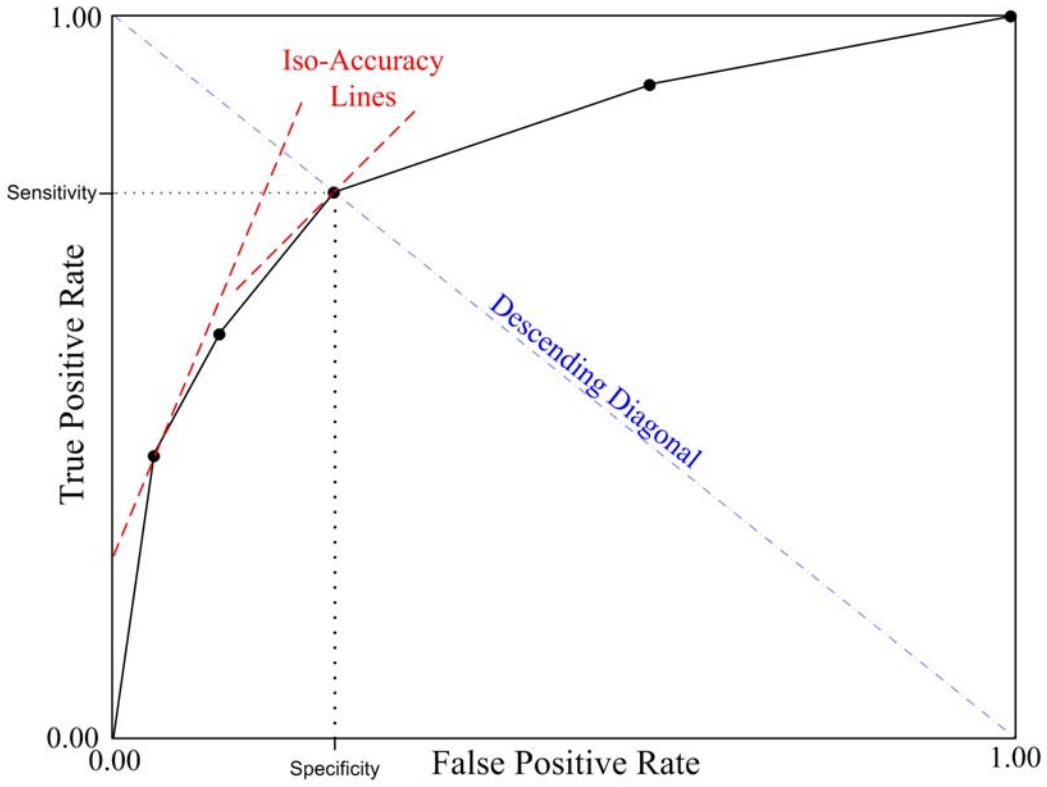

Figure 4.10 Iso-Accuracy Lines

4.3.3 Statistical Comparison of Area Under ROC Curves

To evaluate if one classification system is better than another classification system, one needs to take into account the correlation between the two areas due to the paired nature of the comparison (Hanley and McNeil 1982). The first step is to compute the area under the ROC curve, which may accomplished via several methods. The easiest method is the Mann-Whitney statistic, also called the empirical method, which is a sum of the areas under the curve calculated using a trapezoidal rule (Hanley and McNeil 1983). The empirical method has been shown to systematically underestimate the area

under the curve, especially when the number of positive cases is small and the resulting ROC curve is a step-function, but is an unbiased estimator of the AUC (Hanley and McNeil 1983). Other methods for computing the area under the ROC curve include kernel smoothing to smooth out the ROC curve or assuming normal distribution and using parametric methods (Faraggi and Reiser 2002). When the number of true cases is at least 20, the Mann-Whitney method, while systematically underestimating the true area under the ROC curve, yields a result that is often close to the best of the other methods (Faraggi and Reiser 2002).

To compare two areas under the ROC curve, we use the approach described by Hanley and McNeil (1982, 1983), where the critical ratio, Z, is calculated using equation 4.1. In this equation, A_x is the area, SE_x is the standard error, and r is the correlation coefficient. The standard errors may be estimated using the method described in Hanley and McNeil (1982) and the correlation coefficient may be estimated from Hanley and McNeil (1983). The critical ratio, Z, is evaluated for statistical significance using the normal distribution, which leads to a threshold of 1.96 for a two-sided comparison.

$$z = \frac{A_1 - A_2}{\sqrt{SE_1^2 + SE_2^2 - 2rSE_1SE_2}} \tag{4.1}$$

When comparing the performance of multiple classifiers on multiple test datasets, as we will do in Chapter 6, the standard Friedman non-parametric statistical

test is used to determine if a statistical difference exists between those classifiers (Friedman 1937). When comparing the performance of two classifiers on multiple test datasets, as we will do in Chapter 6, the standard Wilcoxon Signed-Rank test is used.

4.4 Summary

In the clinical laboratory domain, the domain's extreme class imbalance hampers the development of autoverification systems to classify results as acceptable or erroneous. Laboratory errors are estimated to be about 1% of the results, resulting in an extreme between-class imbalance; are multifactorial in their origin, resulting in within-class imbalances; and have a continuum of possible affects, resulting in small disjuncts. By over-sampling the minority-class, under-sampling the majority-class, using unequal misclassification costs, or using a single-class classifier, performance of an autoverification system may be improved. However, as we will see next, a novel synthetic error generation system outperforms existing methods and enables the better detection of errors in the clinical laboratory.

Chapter 5: Synthetic Minority-Class Generation

In order for laboratory experts to trust an autoverification system, it must do more than just flag a value as being possibly erroneous but must also state the criteria used to make that determination. In addition, the inference engine must be able to handle the inherent uncertainty in a clinical laboratory domain. Rule-based systems, which are used in almost all current autoverification systems, are capable of stating their criteria, but cannot properly handle uncertainty. In contrast, Bayesian networks are able to explain their criteria and properly handle uncertainty. In order to classify laboratory results, we must define the structure and parameters of the Bayesian networks, either by eliciting that knowledge from domain experts or using a training dataset to estimate the structure and parameters. Eliciting and maintaining all of the required knowledge from laboratory experts would be a most daunting task, so identification of the Bayesian network's structure and parameters are determined from a training dataset with structural restrictions defined by experts (Mars and Miller 1987).

In the previous chapter, we discussed the trio of issues associated with class imbalance: between-class imbalance, small disjuncts, and within-class imbalance. As discussed in Chapter 2, errors in the clinical laboratory are very rare, have disjuncts ranging from very small to huge, and have a multifactorial etiology that gives rise to within-class imbalances. These factors result in a very poor quality natural training dataset that prevents a Bayesian network from effectively detecting errors. Our solution

is to create minority-class data elements synthetically in a training dataset, which results in the more effective training of the Bayesian network.

5.1. Basis for model

Rather than introducing errors, one could use a single-class classifier to identify acceptable results and, those not meeting the definition of the acceptable class, erroneous results. However, there are two significant limitations to this approach. First, biological variability makes defining acceptability limits problematic. Second, the single-class classifier is not able to hypothesize as to the source of error. Therefore, we will utilize synthetic minority-class data elements. The use of synthetic minority-class data elements is not wholly novel: both the SMOTE and DataBoost-IM methods create some minority-class data elements synthetically (Chawla, Bowyer et al. 2002; Guo and Viktor 2004). However, neither method generated all of the minority-class data elements synthetically, because both examined imbalanced datasets that contained at least some examples of the minority class. As discussed in Chapter 2, the only gold standard for identifying errors in the clinical laboratory are those knowingly introduced by the researcher via some rule or set of rules to facilitate the study of error detection. A natural clinical laboratory dataset does not have a perfect method to accurately identify all laboratory errors. A natural dataset of laboratory results would, having had all sizeable errors removed by the expert's prior review, contain only a single class corresponding to acceptable results, to which we then add synthetic errors.

5.1.1. Creating Synthetic Errors

We start with a natural dataset of real reported laboratory results and assume, since laboratory experts reviewed the data prior to reporting and errors are rare, it is free from sizeable errors. The dataset consists of one target analyte and zero or more covarying analyses, all of which are assumed to have, or to be converted to, a Gaussian distribution. Each analyte is considered one at a time. Without loss of generalizability to higher dimensional distributions, consider the one-dimensional example in Figure 5.1 that shows a Gaussian distribution, with mean μ and standard deviation σ, in blue, representing error-free results for an analyte of interest such as glucose. If we were to randomly copy some percentage, p, of these error-free results, but introduce a bias, m, in the analyte of interest, then we would create a second Gaussian distribution. The second Gaussian distribution, representing erroneous laboratory results, would have a mean value equal to $\mu + m$, a standard deviation σ, and an area under the Gaussian curve that is some portion of the original area, depending on the value for p. Note that the standard deviation of the error-containing Gaussian curve is equal to the standard deviation of the error-free Gaussian curve since we use an additive error. To complete the creation of the new training dataset, synthetic errors of the opposite magnitude would be added to create a minority class with two subclusters that bracket the error-free distribution. This process allows us, by varying the probability and magnitude of error of each minority class subcluster, to create training datasets customized for the domain of interest without the significant class imbalance problems

of a natural dataset. The resulting training dataset, containing the original error-free results and synthetic errors, can now be used to train a Bayesian network to classify laboratory results.

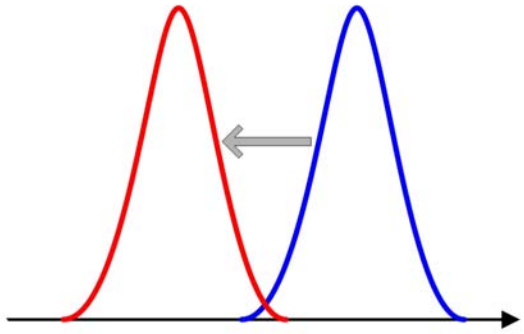

Figure 5.1 Creating Synthetic Errors

5.1.2. Modeling Errors

Synthetic errors, as discussed above, do not model naturally occurring errors in the clinical laboratory, which results in a better training dataset. Consider the hypothetical distribution of naturally occurring errors in glucose, Figure 5.2, which consists of sample switching errors, non-fasting patient, sample hemolysis, and instrument bubble. Sample switching, the most common error in this model since the area under its curve is the largest, is caused by two glucose samples being randomly switched and has a mean error of 0.0. A non-fasting patient, the next most common error in this model, is due to the patient not fasting before blood collection. Depending on what the patient consumed, when they consumed it, and their diabetic state, their glucose measurement will have some positive error. Sample hemolysis interferes with the measurement

process and results in a negative error. Finally, if the instrument draws some air when aspirating the sample for analysis, the measured value will have a negative error proportional to the amount of air drawn in. The overall distribution of laboratory errors will be a function of the individual error distributions, which, in this example, is clearly complex and has an expected error of 0.0.

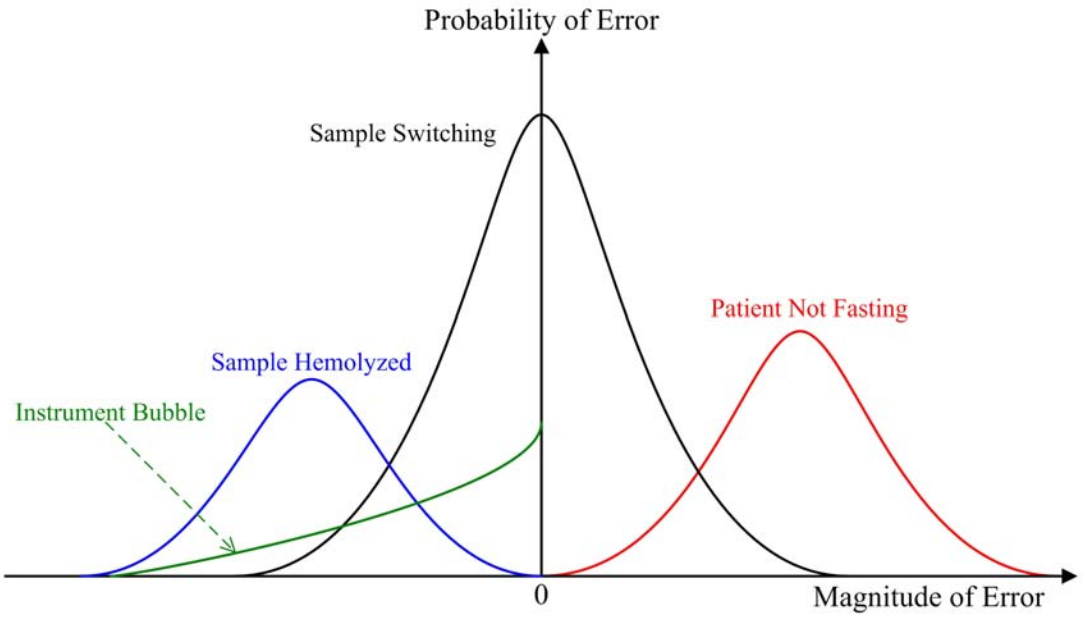

Figure 5.2 Hypothetical Distribution of Natural Laboratory Errors

However, an error of zero or close to zero, even though it is still an error and, if possible, should be addressed to reduce future mistakes, is not going to be detected by an autoverification system or laboratory expert. For every analyte in every clinical lab checked by an autoverification system, the system should have a certain sensitivity and specificity to detect a minimum magnitude of error, which will be a dependent on the detectability of errors in that analyte as well as the operational desires of the clinical

laboratory. For example, a clinical lab may desire a 50% sensitivity to detect a 20 mg/dl error in glucose and a 95% sensitivity to detect a 50 mg/dl error. However, modeling the natural distribution of clinical laboratory errors is not likely to produce an autoverification system sensitive enough to detect errors.

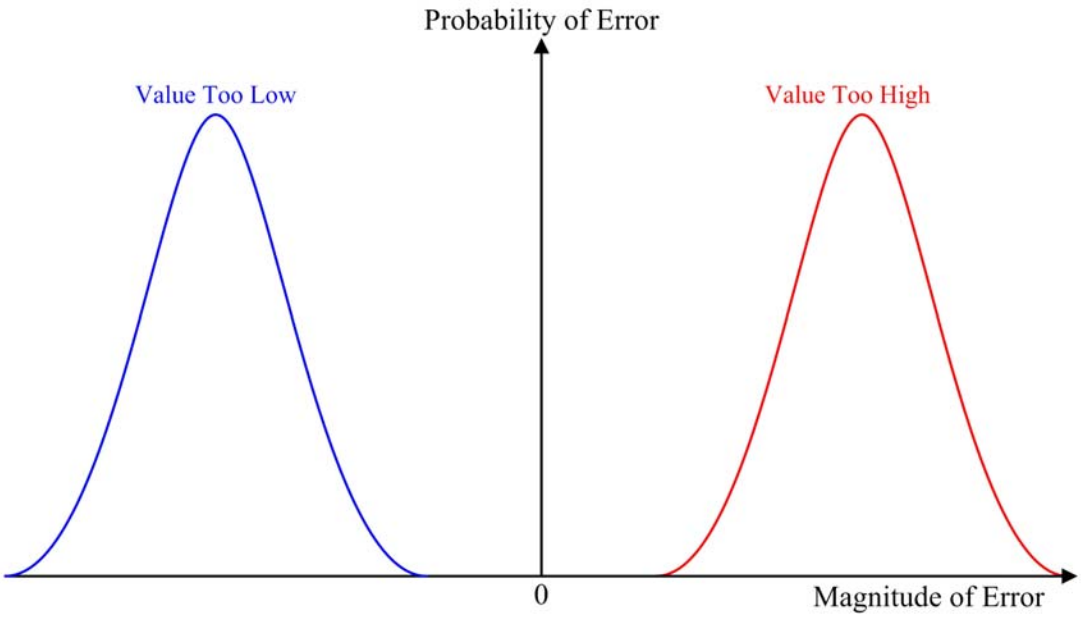

Figure 5.3 Distribution of Synthetic Laboratory Errors

A training dataset with synthetic errors, however, is more likely to produce an autoverification system with the requisite sensitivity. For the purposes of this dissertation, errors are modeled as simply causing the value to be too low or too high. See Chapter 8 for a discussion of work currently underway that will enable hypothesizing as to the probability that a predicted error was due to some specific error such as hemolysis. Into the initial training dataset, we add synthetic errors with a negative magnitude of error to model “value too low” errors and a positive magnitude

of error to model “value too high” errors using the procedure discussed in the previous section. The resulting distribution of synthetic errors is depicted in Figure 5.3. For example, we might train the autoverification system to detect errors of 20 mg/dl in order to detect errors in glucose. In order to create a Bayesian network able to detect errors in clinical laboratory data, we start with a natural dataset presumed free of significant errors, create synthetic errors with the desired minority-class probability and magnitude, learn the parameters of a conditional Gaussian Bayesian network whose structure was identified via structure learn and/or domain experts, and make inferences about the probability of error given the observations.

5.2. Simulation Method

The autoverification system described herein consists of two Bayesian networks. The first Bayesian network, consisting of a glucose node, an HbA1c node, and a directed edge from the glucose node to the HbA1c node , is used to predict the expected HbA1c value based on the value of glucose. The second Bayesian network is used to infer the probability the measured HbA1c value is too low, too high, or acceptable based on the predicted value. A two-stage system is used to enable the use of non-Gaussian algorithms to calculate the predicted value as long as the output is an expected value and an uncertainty term. In addition, a two-stage system reduces the computational complexity.

Figure 5.4 shows the process used to evaluate the performance characteristics of the system and is summarized as follows:

1. Generate initial, error-free training and testing datasets with the desired correlation coefficient and size of 10,000.
2. Add errors to training and testing datasets using the synthetic error method with varying error magnitude and minority-class probability.
3. Learn the parameters of the Bayesian networks to classify data using 10-fold cross-validation.
4. Evaluate the Bayesian network using the testing dataset.
5. Repeat this process 100 times.

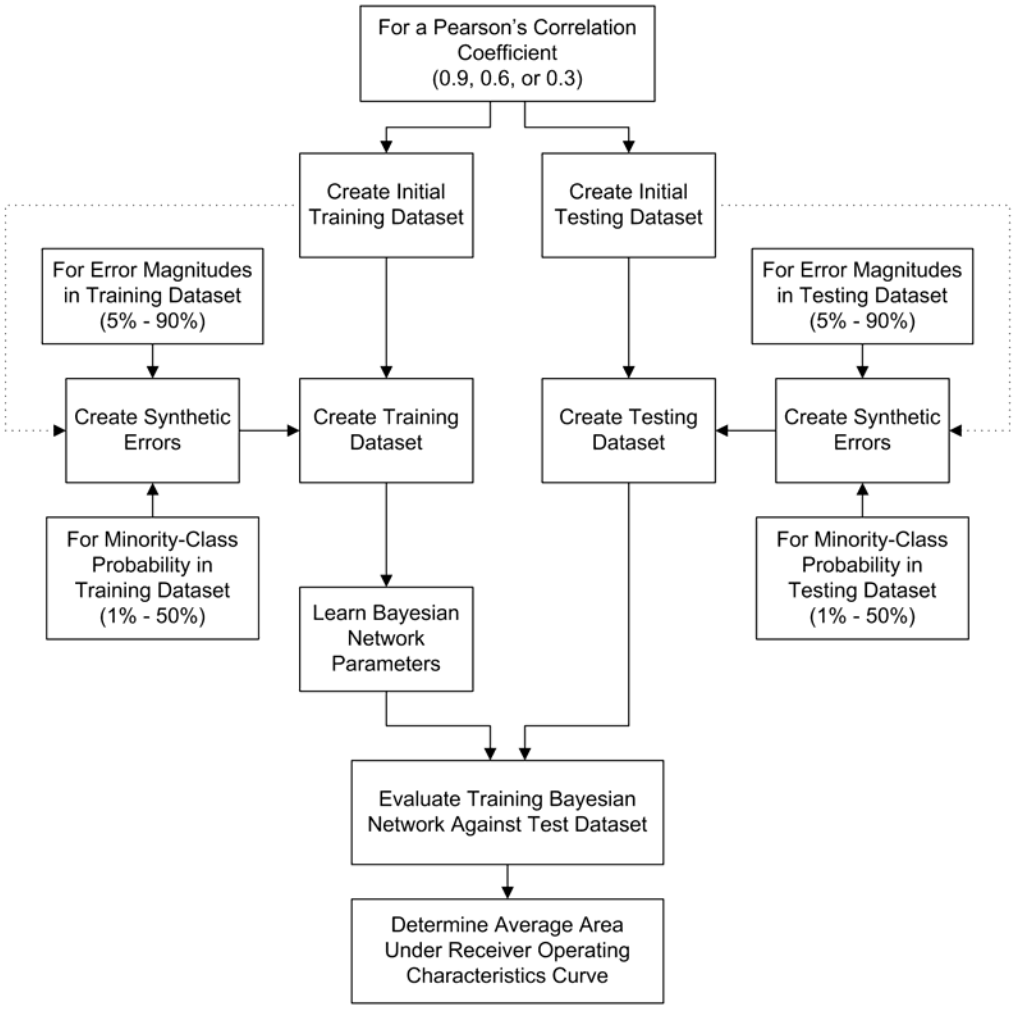

Figure 5.4 Simulation Process

Glucose, derived from analysis of the NHanes dataset, was set to have a Gaussian distribution with a mean value of 95.0 mg/dl and a standard deviation of 13.5 mg/dl. A linear relationship, equation 5.1, between glucose and HbA1c was assumed based on results of the Diabetes Control and Complications Trial and linear regression performed using the NHanes (2004) dataset (Rohlfing, Wiedmeyer et al. 2002) . Equation 5.1 contains a Gaussian noise term, $\mathcal{E}$, with mean 0.0 and standard deviation empirically

derived to produce a Pearson's correlation coefficient of 0.3, 0.6, or 0.9 for a weak, medium, and strongly correlated systems, respectively. The levels of correlation are inspired by the correlation between glucose and HbA1c in non-diabetics, pre-diabetics, and diabetics. It was empirically determined that a standard deviation for the Gaussian noise term of 0.12, 0.33, and 0.75 produced the desired strong, medium, and weak correlations. Both the initial training and testing datasets were created by calculating an HbA1c value using a glucose value randomly sampled from its Gaussian distribution and including some random Gaussian noise.

$$\text{HbA1c} = 3.675 + 0.01765 \times \text{Glucose} + \varepsilon \qquad (5.1)$$

In the training and testing datasets, the minority-class probabilities were varied between 0.1% and 50.0% and the magnitude of HbA1c errors were varied between 0.05 and 2.0. Note that the units of HbA1c are percent so an error of 2% would change a value of 5.0% to 7.0%, but we drop the percentage symbol from the error magnitude term to avoid confusion. For each of the three levels of system correlation and each combination of the four parameters above, we performed 100 simulations computing the average area under the ROC curve. A total of 24,260,400 simulations were performed, requiring approximately four weeks of computer time.

5.3. Performance Characteristics of Basic Model

As discussed in Chapter 4, we expected classifier performance to vary with the degree of the class imbalance. In these simulations, between-class imbalance is modeled as the

minority-class probability. As the minority-class probability increases, the between-class imbalance gets smaller. A within-class imbalance is not modeled at this stage of the simulations as each minority-class subcluster has the same magnitude of error and is equal in size. Class disjuncts are modeled as the magnitude of error and the correlation coefficient. As either the magnitude of error or correlation coefficient increases, the class disjunct is reduced. Magnitudes of error, correlation coefficient, and minority-class probability have direct analogues to the clinical laboratory domain as the size of error, predictability of error due, in part, to biological and instrument variability, and probability of error. For example, errors in cholesterol are more readily identified because it can be measured accurately, has a lower biological variability, and is correlated with other variables (Ricos, Alvarez et al. 1999; Centers for Disease Control and Prevention 2004). In contrast, a diabetic's fasting glucose, while able to be accurately measured, has a high degree of biological variability and, therefore, errors are more difficult to identify.

5.3.1. Performance Varies with Correlation Coefficient

As all other model parameters are kept constant, one would expect the area under the ROC curve to increase as the correlation coefficient increases. Figure 5.5 shows the classifier's average performance at detecting an error in datasets containing 1.0% errors with a 0.50 magnitude as the minority class probability increases for the three different levels of correlation coefficient. As readily observed, for every minority-class probability, errors in the stronger correlated datasets are more detectable than in the

lesser correlated systems. Figure 5.6 shows the classifiers' average performance at detecting an error in datasets containing 1.0% errors as the magnitude of error increases for the three different levels of correlation coefficient. As before, errors in a more-correlated system are more detectable than errors in lesser-correlated system. This pattern is observed for all combinations of minority-class probability and magnitude of error. In a clinical laboratory's autoverification system, the more accurately a value can be predicted, the more accurately errors in that analyte can be detected.

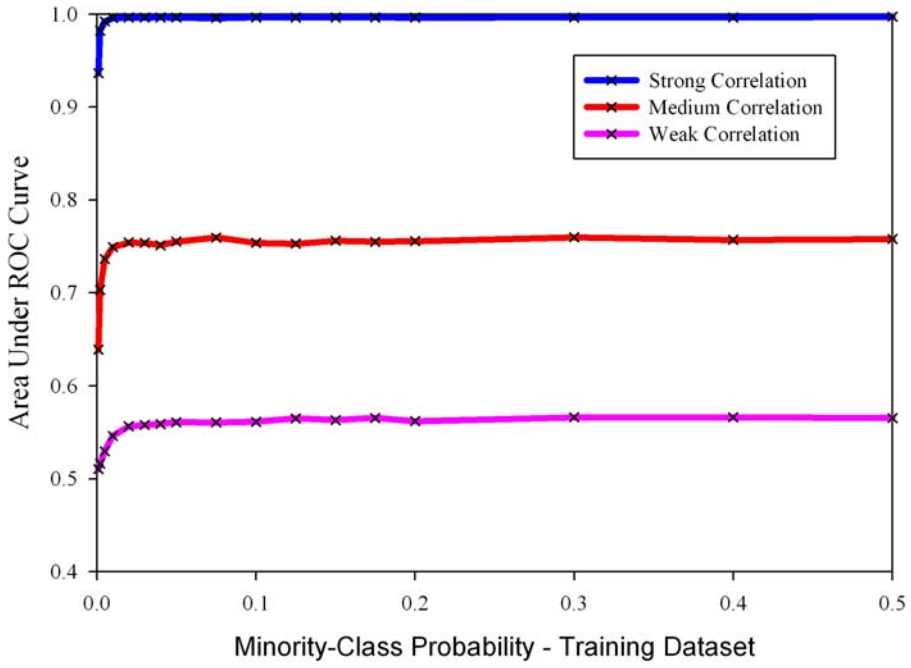

Figure 5.5 Area Under ROC Curve as Training Minority-Class Probability Varies

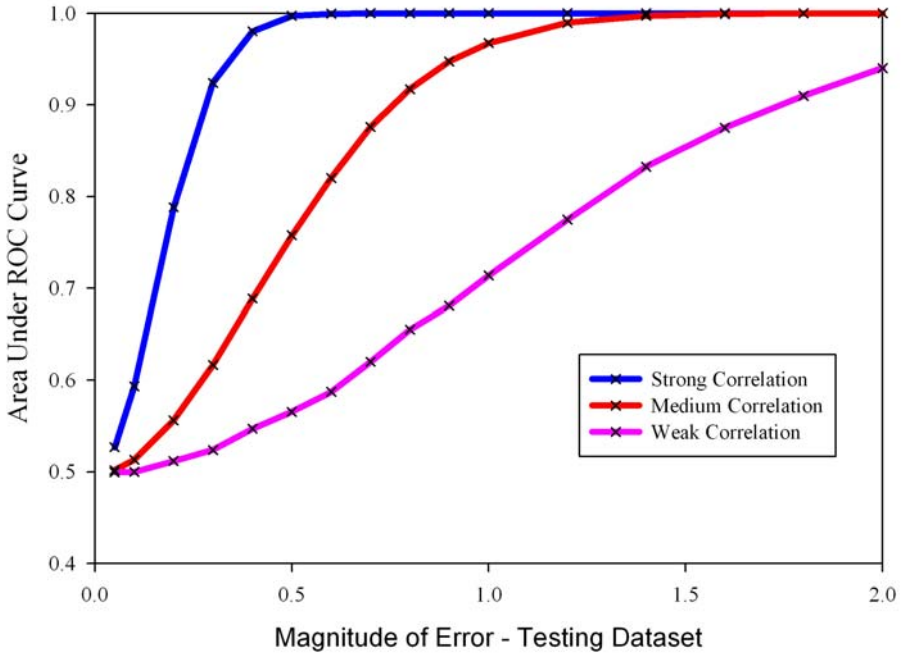

Figure 5.6 Area Under ROC Curve as Error Magnitude Varies

5.3.2. Performance Varies with Minority-Class Probabilities

As discussed earlier, errors in the clinical laboratory are estimated to be roughly between 0.1% and 1%, resulting in a significant between-class imbalance (Bonini, Plebani et al. 2002). Using a representative training dataset is known to produce poor results, but the relationships between minority-class probabilities in the training dataset versus in the testing dataset were not well elucidated. Holding all other variables in the simulation constant, Figure 5.7 shows the average area under the ROC curve as the minority-class probabilities in the training and testing datasets are independently varied between 0.1% and 5%. The relationship between 5% and 50% is not displayed in Figure 5.7, though it follows the same pattern. For a given minority-class probability in

the training dataset, the average performance does not vary with the minority-class probability in the testing dataset. Conversely, for a given minority-class probability in the testing dataset, the average performance increases significantly with the minority-class in the training dataset until it reaches a plateau, as demonstrated in Figure 5.8. The optimal minority-class probability in the training dataset, as indicated in Figure 5.9, also depends on the magnitude of error used in the training dataset. From Figure 5.7 through Figure 5.9, we conclude that using a minority-class probability of 50% in the training dataset produces optimal results for detecting errors in the clinical laboratory domain.

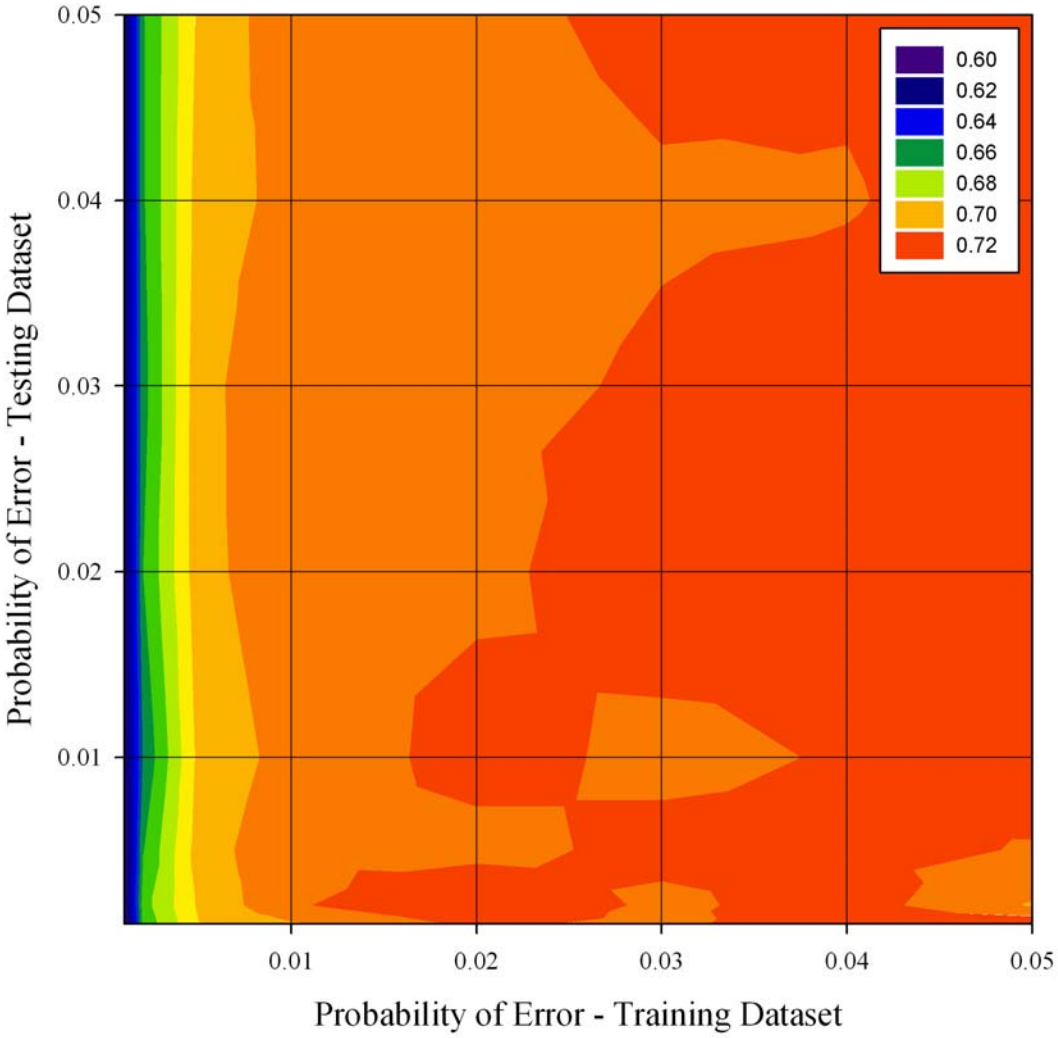

Figure 5.7 Area Under ROC Curve as Minority-Class Probability Varies

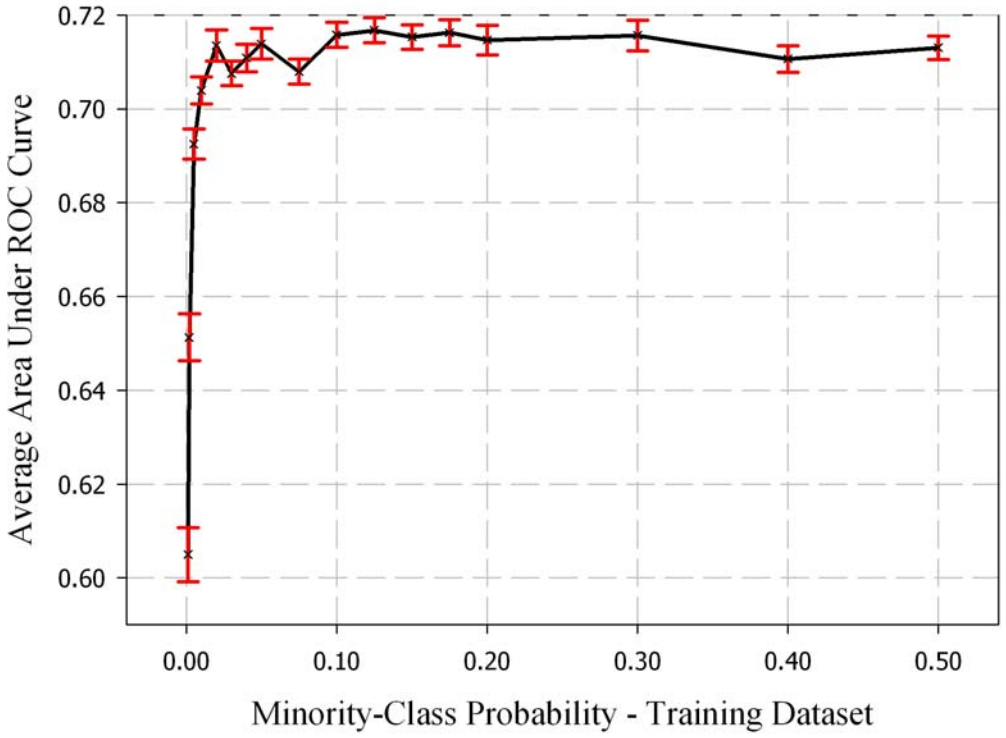

Figure 5.8 Increasing Performance with Non-Representative Minority-Class Probability

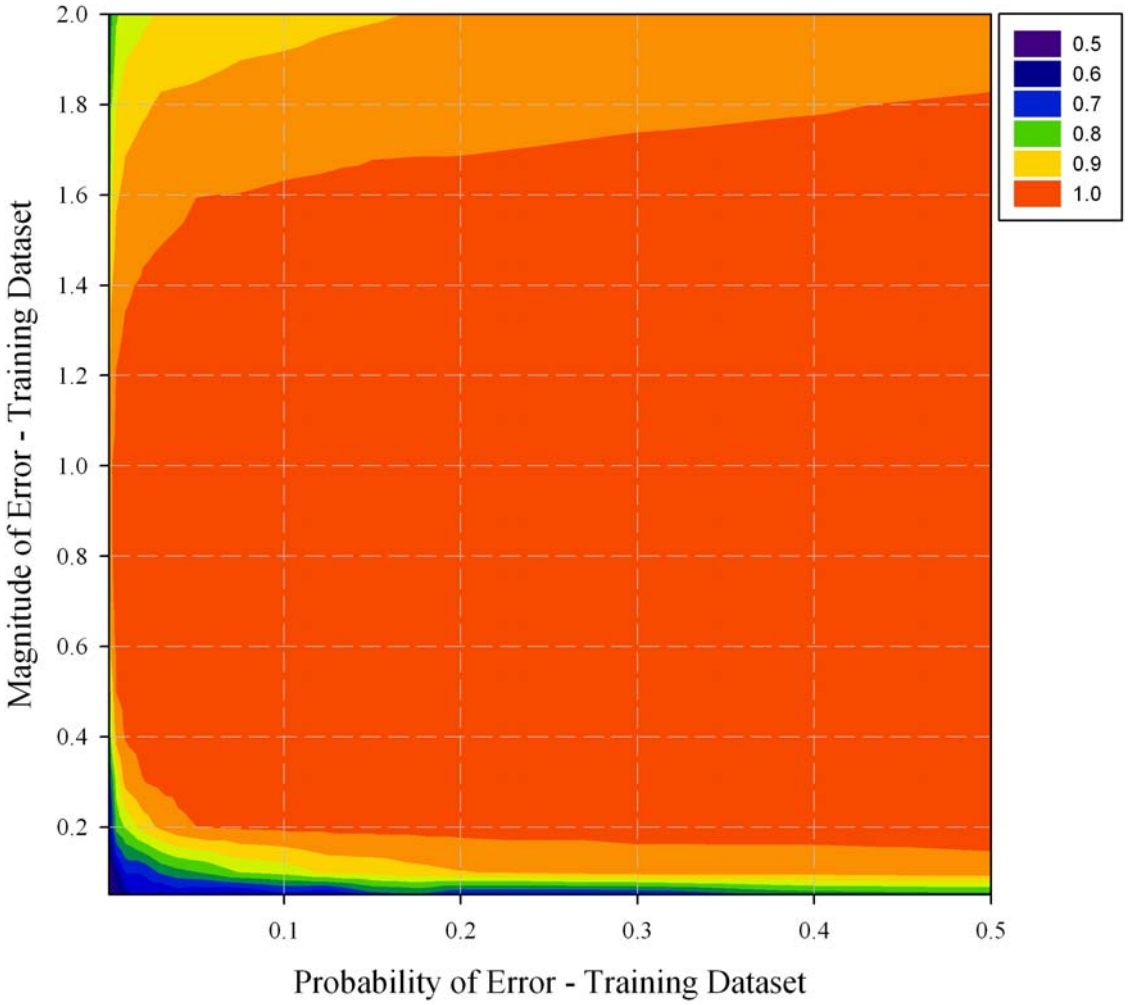

Figure 5.9 Average Performance to Detect Errors of Size 1.0

5.3.3. Performance Varies with Magnitude of Errors

The magnitude of error directly affects the size of the disjunct between classes and larger errors should be more easily detected than smaller errors. Figure 5.10 shows the average classifier performance, as measured by area under the ROC curve, over various magnitudes of error when detecting errors in a system with medium correlation, a minority-class probability of 0.2%, and a representative training dataset. This system has a poor ability to small errors, less than 0.4. In addition, this system over-fits the magnitude of error such that training with a single magnitude of error results in a system that performs poorly for errors that are either larger or smaller. Contrast the system's performance in Figure 5.10 with the system's performance in Figure 5.11, which was conducted in exactly the same manner except that a minority-class probability of 50% was used in the training dataset. Not only is performance for small errors dramatically improved, but also the over-fitting seen in Figure 5.10 is largely gone. Figure 5.12 displays the difference in the area under the ROC curve gained by using a non-representative training dataset. When training an autoverification system, it is possible to minimize the affect of over-fitting the training dataset by using a non-representative minority-class probability such as 50%.

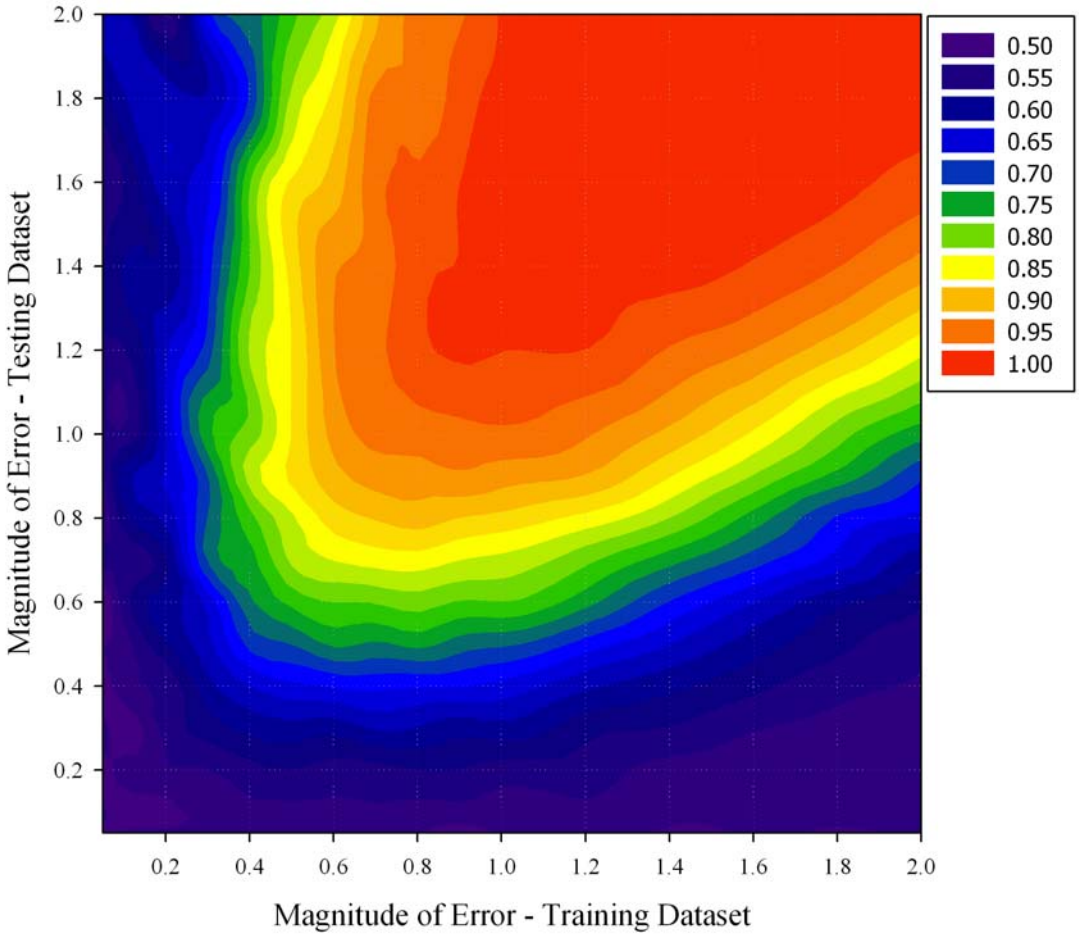

Figure 5.10 Area Under ROC Curve as Error Magnitude Varies with Representative Training Dataset

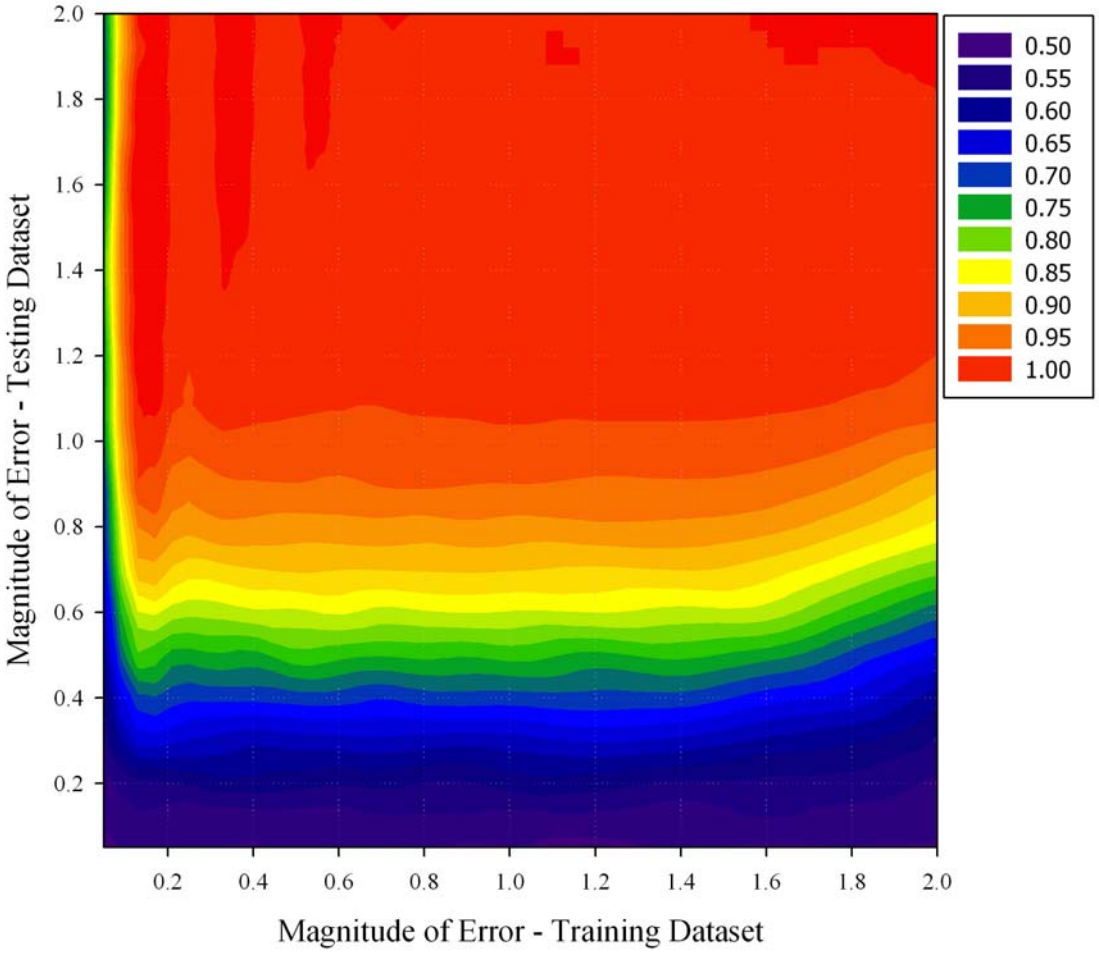

Figure 5.11 Area Under ROC Curve as Error Magnitude Varies with Non-Representative Training Dataset

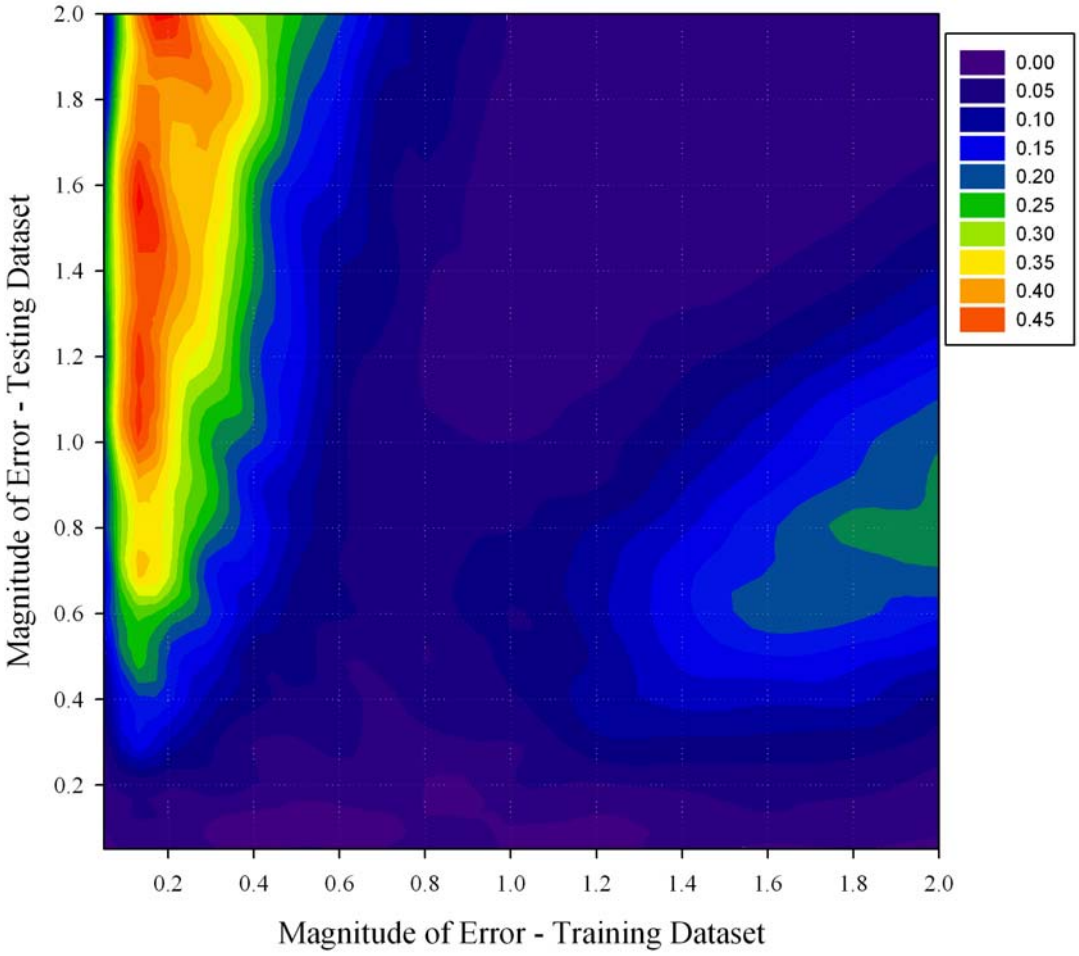

Figure 5.12 Improvement in Area Under ROC Curve Due to Non-Representative Training Dataset

5.4. Performance Characteristics of Advanced Model

The prediction model, **Error! Reference source not found.**, used in the above discussion is a primitive model chosen for its ease of operation and uncomplicated interactions. While primitive, it still addresses a critical need of the clinical laboratory in reviewing tremendous amounts of data. In this section, to estimate the generalizability of the method we examine the performance characteristics of the synthetic error autoverification system using a more complex model and real dataset.

5.4.1. NHanes Chemistry Panel

As part of the NHanes laboratory assessments, subjects provided samples for analysis of diabetes factors (glucose, insulin, c-peptide, HbA1c), lipids (cholesterol, HDL cholesterol, triglyceride), biochemistry (serum albumin, ALT, AST, ALP, BUN, calcium, bicarbonate, GGT, iron, LDH, phosphorous, total bilirubin, total protein, uric acid, serum creatinine, sodium, potassium, chloride, globulin) and assorted other analyses (National Center for Health Statistics 2004). From these components, we selected a common chemistry component, AST (aspartate aminotransferase), which is performed to check liver function, for evaluation.

To predict an AST value, we first determined the structure of the directed acyclic graph using deal and R (Bøttcher and Dethlefsen 2003; The R Project for Statistical Computing 2007). We combined the biochemistry data with the patient's age, race, and gender from the NHanes demographics dataset. Persons who identified

their race as “Other” or “Other-Hispanic” constituted only 4.6% and 3.4% , respectfully, of the dataset and were eliminated from further analysis due to their low percentage and non-specific race. Age was discretized into three levels: child (less than 18 years of age), aged (greater then or equal to 65 years of age), and adult. We removed patients with missing results or missing demographics from further analysis and split the dataset equally into training and testing datasets. Using the program deal, which runs under the statistical program R, we determined the structure of the directed acyclic graph. The resulting structure consisted of three distinct sub-graphs anchored by the patient demographics age, race, and gender. One sub-graph consisted of age, race, gender, BUN, potassium, serum creatinine, and uric acid. Another sub-graph consisted of age, race, gender, serum albumin, globulin, total protein, calcium, sodium, chloride, and bicarbonate. The final sub-graph, containing the analyte of interest AST, Figure 5.13, consisted of the remaining analytes. Even though the panel consists of twenty analytes, the value of any given analyte is only influenced by the patient’s demographics and generally two other analytes.

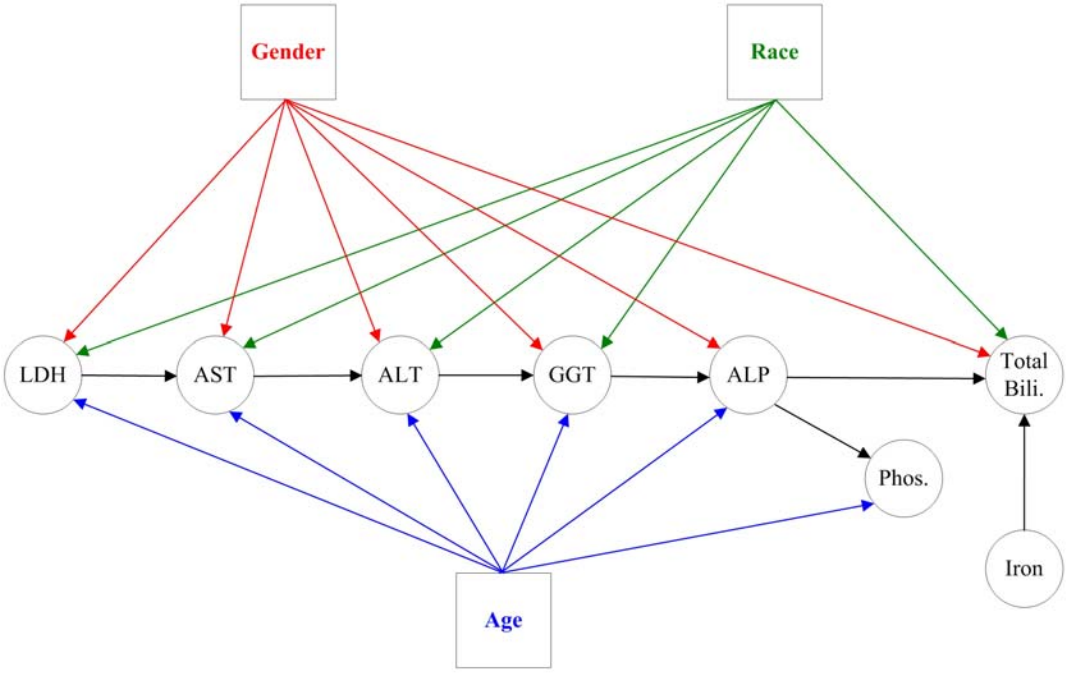

Figure 5.13 Directed Acyclic Sub-Graph from NHanes Biochemistry Panel

The AST-containing sub-graph, Figure 5.13, could be used to predict the value of AST, given values for the other nodes in the network. However, the form of this directed acyclic graph is not convenient. Since the training and testing data do not have missing data, we can use the concept of d-separation to remove unnecessary nodes. In addition, explaining the prediction model to laboratory experts, who rarely have experience with making inferences in directed acyclic graphs, is easier if the node of interest does not have decedents. To accomplish this task, we can either use arc-reversal, which must maintain conditional dependencies, on the larger graph or impose restrictions on the structure learning and repeat the structure-learning step with a reduced data set. We choose the latter method as it represents the more likely method

to be employed by the client. In addition, we removed the top 0.5% AST values as outliers. The bottom 0.5% of the AST values were not outliers. The resulting directed acyclic graph, Figure 5.14, is not Markov equivalent to the sub-graph in Figure 5.13 due to the heuristic nature of structure learning. We are now able to use this model to predict AST values and, therefore, detect errors in AST.

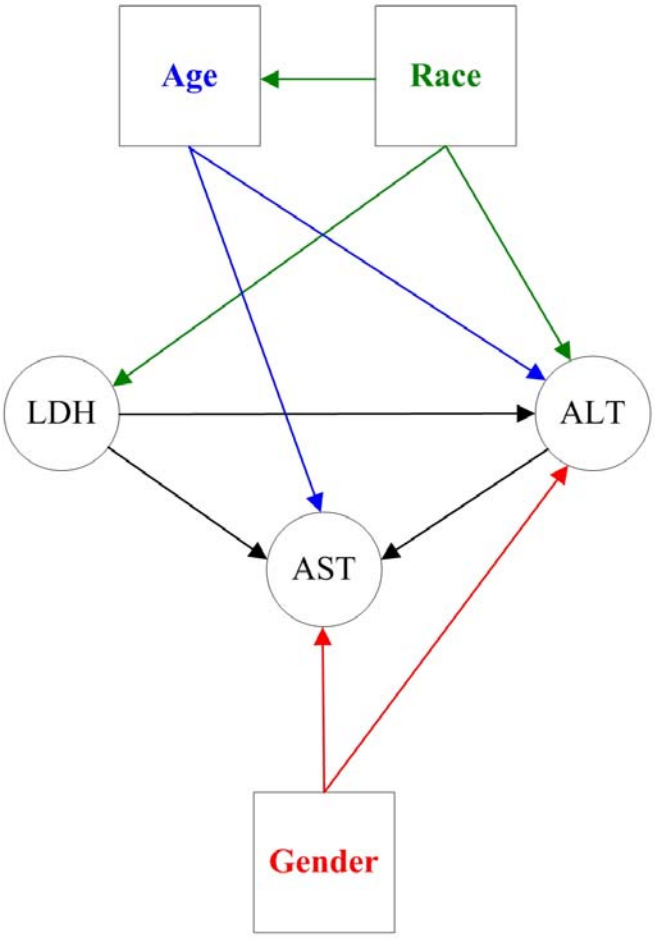

Figure 5.14 AST Prediction Model

5.4.2. Predicting AST

From the directed acyclic graph in Figure 5.14, we are able to predict a value for AST along with a standard deviation. The smaller the standard deviation, the more confident we are in the value and the more readily errors can be detected. When values are known for all nodes other than AST, the standard deviation ranges from 3.3 U/L for a female child to 6.3 U/L for an older male. In contrast, in the absence of any knowledge, the standard deviation in our AST estimate is 8.8 U/L. In the laboratory performing the analyses, the reference range for AST is 13 – 33 U/L for adults and 13 – 63 U/L for children (National Center for Health Statistics 2004).

To estimate the minimum error magnitude we want our system to detect, we compute a reference change value (RCV) for AST. An RCV is the change in an analyte, assuming steady state conditions and previous results, that should be detectable in an autoverification system (Fraser, Stevenson et al. 2002). In the NHanes dataset, we do not know a patient's previous AST result, so the RCV will provide a conservative estimate. An analyte's RCV is calculated is equation 5.2, where CV_A is the analytical coefficient of variation, CV_I is mean within-subject biological coefficient of variation, and Z is the desired standard deviate (Fraser, Stevenson et al. 2002).

$$\mathrm{RCV} = \sqrt{2} \times \mathrm{Z} \times \sqrt{\mathrm{CV}_{\mathrm{A}}^{2} + \mathrm{CV}_{\mathrm{I}}^{2}} \qquad (5.2)$$

For AST, we use a within-subject coefficient of variation of 11.9%, an analytical coefficient of variation of 15.2%, and a standard deviate of 1.96 to obtain an RCV of

53.5% (Ricos, Alvarez et al. 1999). Assuming a mean AST value of 25 U/L, the synthetic error autoverification system should detect errors of 13.4 U/L or larger.

5.4.3. Detectability of Errors in AST

We trained the synthetic error autoverification system using a training dataset containing 50% synthetic errors. The magnitudes of the synthetic errors in the training dataset were varied between 3 and 26 in increments of three. We created the testing datasets by varying the magnitude of error between 3 and 60 and varying the testing minority-class probability between 5.0% and 50.0%. Using Figure 5.14, we predict the expected AST value given gender, age, LDH, and ALT and then compute the probability that the AST value is in error. Only one iteration was done at each combination of evaluation parameters due to the limited amount of testing data.

As expected, small errors in AST are very difficult to detect while larger errors are more readily detected. Figure 5.15 shows the relationship between the training error magnitude and the testing error magnitude, which indicates that training the system to detect errors in AST with a moderate sized error of 25 U/L has good performance for smaller errors and larger errors. This relationship is similar to the one observed in Figure 5.11 for the simple model. Performance did not significantly vary with the minority-class probabilities in the testing dataset, Figure 5.16, though statistical analysis at the low end of the range, minority-class probabilities less than 5.0%, were not possible due to the small size of the NHanes dataset. In the beginning of Section 5.4.2, we stated that AST's standard deviation in the training dataset ranged from 3.3 U/L for

a female child to 6.3 U/L for an older male. In Figure 5.17 we see small AST errors for a female child are readily detected whereas only larger errors are readily detected for aged males.

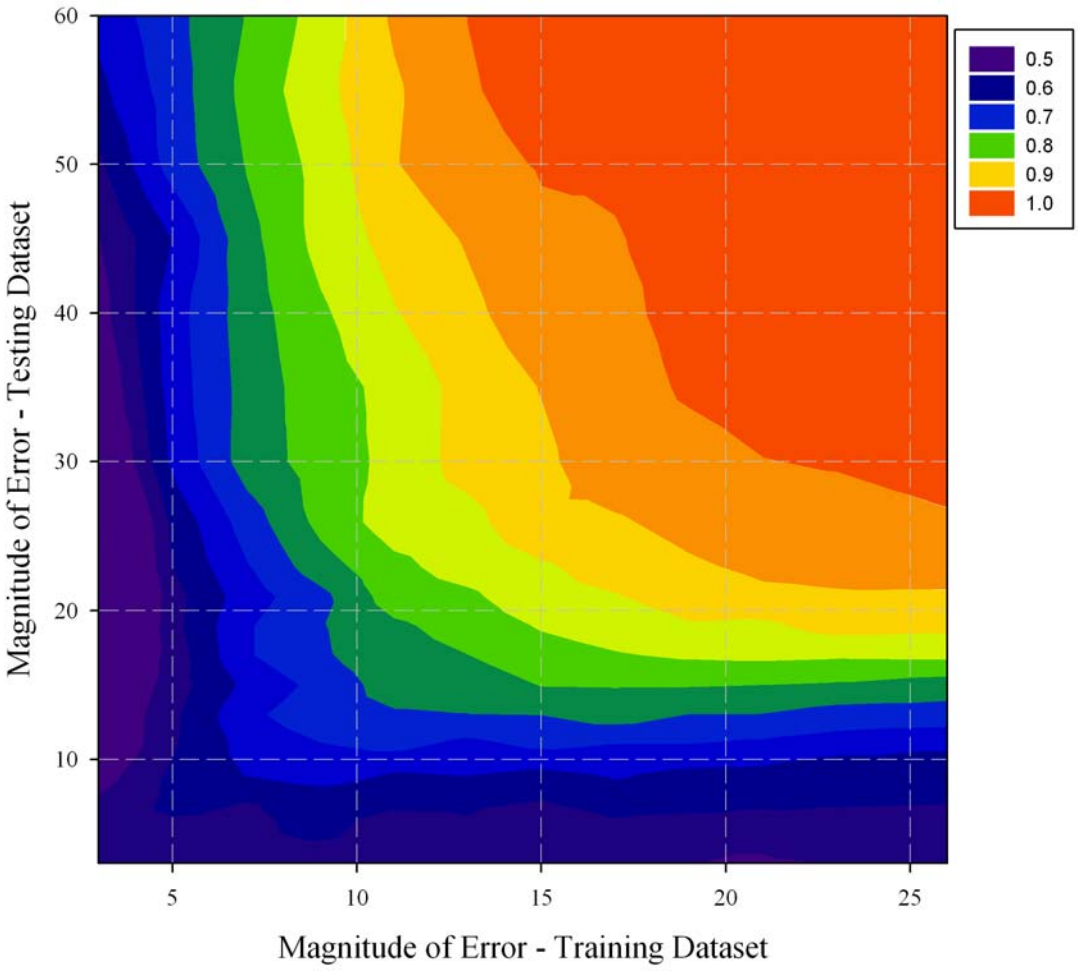

Figure 5.15 Detectability of Errors in AST as Error Magnitude Varies

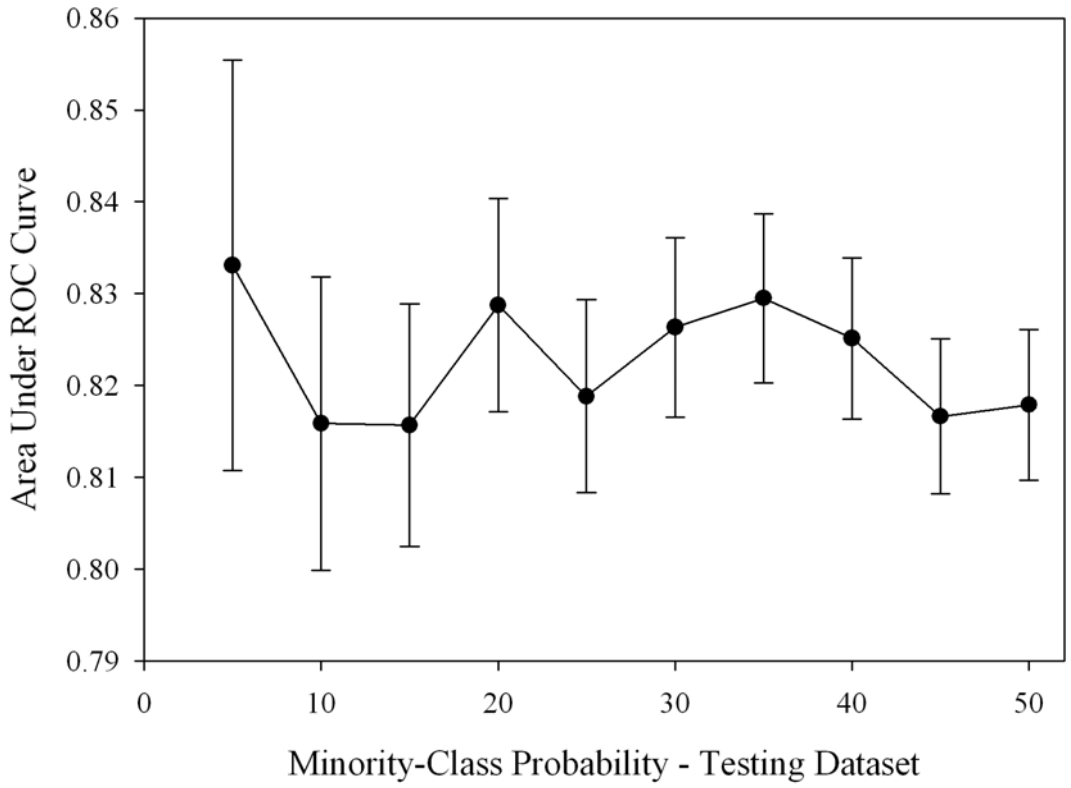

Figure 5.16 Detectability of Errors in AST as Minority-Class Probability in Testing Dataset Varies

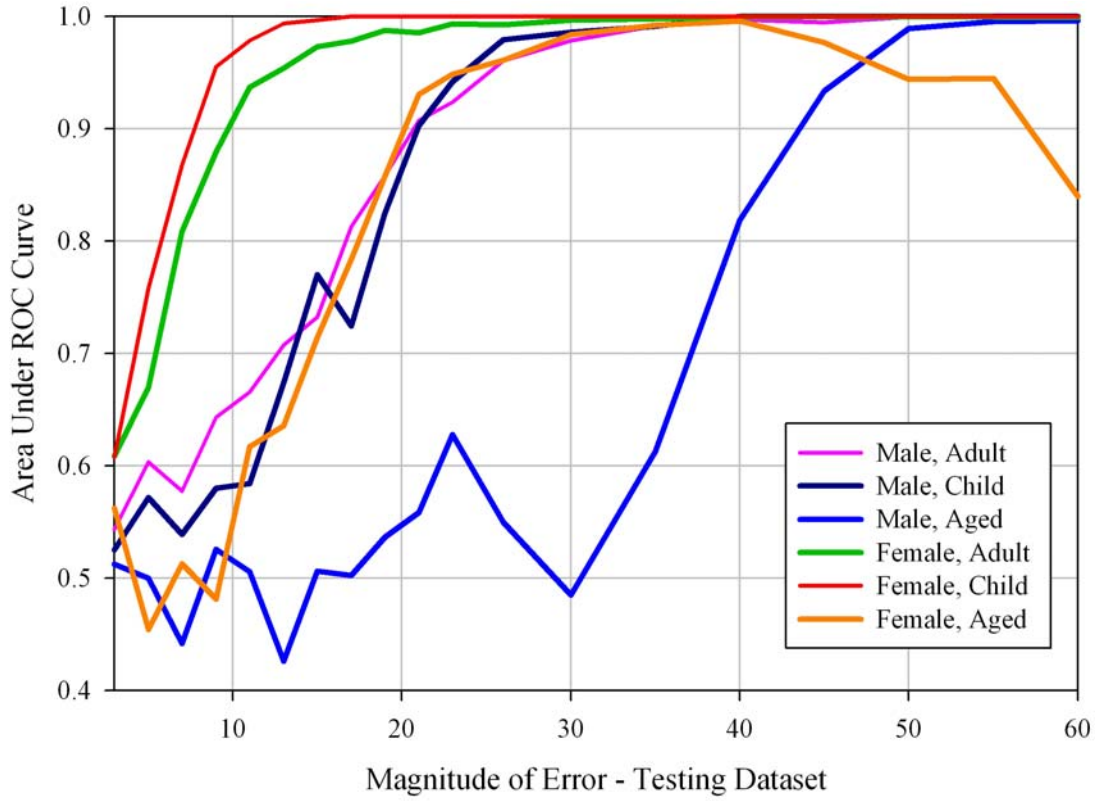

Figure 5.17 Detectability of Errors in AST by Age and Gender

5.5. Summary

The implications for the clinical laboratory due to the results presented in this chapter are profound: Bayesian-based autoverification systems can be developed without an expensive annotated database of laboratory errors. The use of synthetic errors mitigates the between-class imbalance impact and allows for the selection of the most appropriate error magnitude to minimize the affect of small disjuncts. The Bayesian networks used are small and easily constructed from a statistical analysis of a dataset, their parameters are readily determined and exact inference efficient. The synthetic error approach enables the better training of a Bayesian autoverification system. We next compare the

performance of the synthetic error generation method against standard class imbalance methods.

Chapter 6: Comparison of Synthetic Error Method Against Standard Methods

The previous chapter introduced the concept of synthetic error generation as a means to create better training datasets for a Bayesian network-based autoverification system. Results indicate such a system is effective in learning how to identify errors in the presence of an extreme class imbalance. In this chapter, we compare performance of the synthetic error generation method to standard methods for handling class imbalance: minority-class over-sampling and majority-class under-sampling. By comparing performance across a wide range of system parameters, we determine where the synthetic error generation method is statistically superior to standard methods, where it is statistically inferior, and where it is statistically indifferent.

6.1. Model Definitions

In comparing the synthetic error generation method against standard methods, we utilize a different error model, one based on a common laboratory error, to create the testing dataset to enable an unbiased comparison. In addition, this error model is used to generate the initial training datasets for the minority-class over-sampling and majority-class under-sampling. The Bayesian network used for the synthetic error generation, minority-class over-sampling, and majority-class under-sampling is the same as used in Chapter 5.

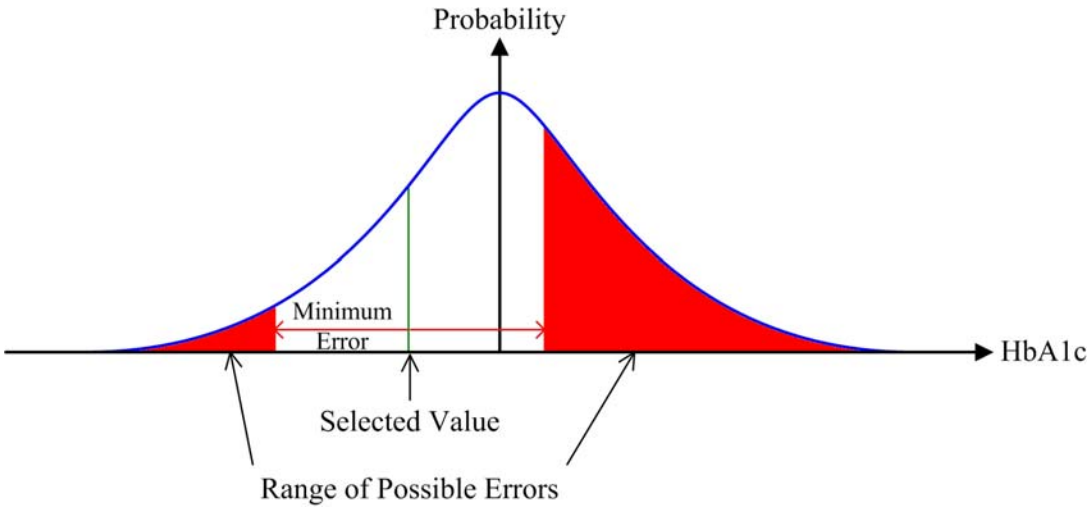

Figure 6.1 Creating a Sample-Switching Error

6.1.1. Error Model

Unlike the method used in Chapter 5, datasets used for training the minority-class over-sampling and majority-class under-sampling Bayesian networks use an error-model based on sample-switching errors, a common laboratory error. The testing dataset is also created using a sample-switching error model in order to provide a realistic and unbiased dataset. Sample-switching errors, caused by randomly switching two HbA1c values, are expected to have a mean error of zero, which is not realistic or meaningful to flag as an error. Therefore, for the purposes of this dissertation, we are only interested in errors that exceed some minimum value. For example, assume that HbA1c has the probability distribution in Figure 6.1 and assume a randomly selected value, as indicated. We use the minimum detectable error to define the range of possible errors, indicated in red, and then randomly select the replacement value from this range.

6.1.2. Bayesian Network Model

The Bayesian network model used for the synthetic error generation, minority-class over-sampling, and majority-class under-sampling methods is identical to the one used in the previous chapter, section 5.2 (page 79). This model uses a conditional Gaussian Bayesian network to predict, based on the observed glucose value, the HbA1c value and a second mixed conditional Gaussian Bayesian network to infer if the observed HbA1c value is in error.

6.2. Simulation Process

The simulation process, Figure 6.2, employed for the comparison is similar to the one used in Chapter 5. One significant difference, however, is that the minority-class probability and error magnitude in the training dataset are the same as in the testing dataset. This restriction, significantly reduces the number of simulations, and represents a more realistic operating condition where the training dataset is designed to maximize the detection of errors in the testing dataset. As before, the relationship between fasting glucose and HbA1c is determined by equation 6.1.

$$\text{HbA1c} = 3.675 + 0.01765 \times \text{Glucose} + \varepsilon \qquad (6.1)$$

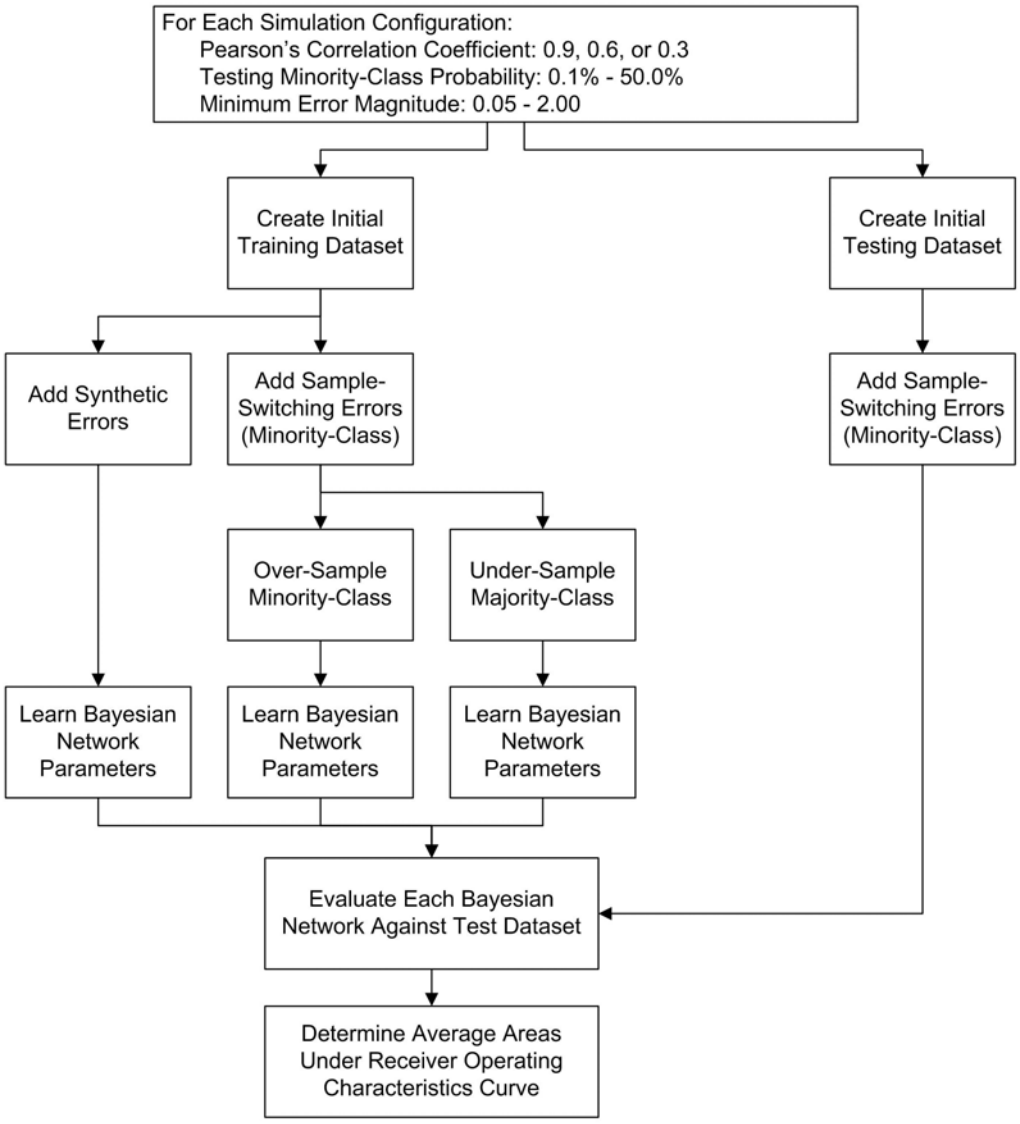

Figure 6.2 Simulation Process

The simulation process is as follows for each combination of correlation coefficient, minority-class probability (ρ), and magnitude of error (m):

1. Generate initial error-free training and testing sets with the desired correlation coefficient and size of 10,000.

2. For the synthetic error generation training dataset, add synthetic errors with a magnitude m. Based on the results from Chapter 5, a minimum error magnitude of 0.4 and a minority-class probability is 50% was used to train the Bayesian network.
3. For the minority-class over-sampling and majority-class under-sampling training dataset, create sample-switching errors with minimum error m such that their initial probability is ρ.
    a. For minority-class over-sampling, randomly duplicate minority-class examples until their percentage equals the percentage of the majority class (50%).
    b. For majority-class under-sampling, randomly remove majority-class examples until their percentage equals the percentage of the minority-class (ρ).
4. Learn the Bayesian network parameters, using 10-fold cross-validation, for synthetic error generation, minority-class over-sampling, and majority-class under-sampling.
5. For the testing dataset, create sample-switching errors with minimum error m such that their initial probability is ρ.
6. Evaluate the performance of synthetic error generation, minority-class over-sampling, and majority-class under-sampling against the testing dataset.
7. Repeat 100 times and calculate average area under ROC curve.

6.3. Statistical Analysis

In our initial statistical analysis, we assumed a null hypothesis that all three algorithms performed consistently throughout each combination of correlation coefficient, magnitude of error, and minority-class probability. Using the Freidman non-parametric statistical test, we determined with at least 95% confidence that there exists a statistical difference between the three algorithms at most of the 816 parameter combinations evaluated except where the area under the ROC curve was very high or very low (Friedman 1937). Unfortunately, the Friedman test does not indicate which algorithm is best for a given set of parameters, just that the three algorithms yield statistically different results.

6.3.1. Variations in Performance

For each of the 816 parameter combinations, one of the three algorithms performed best as indicated by having the highest average area under the ROC curve over the 100 repetitions. While this comparison, Figure 6.3, does not confer statistical significance, it does enable the observation of key patterns. Under-sampling the majority-class was useful when the minority-class probability was higher and when the correlation was stronger. Over-sampling the minority-class was productive at low, but not the lowest, levels of the minority-class probability. The synthetic error generation system appears to be the most useful when the minority-class probability is very low, as observed in the clinical laboratory.

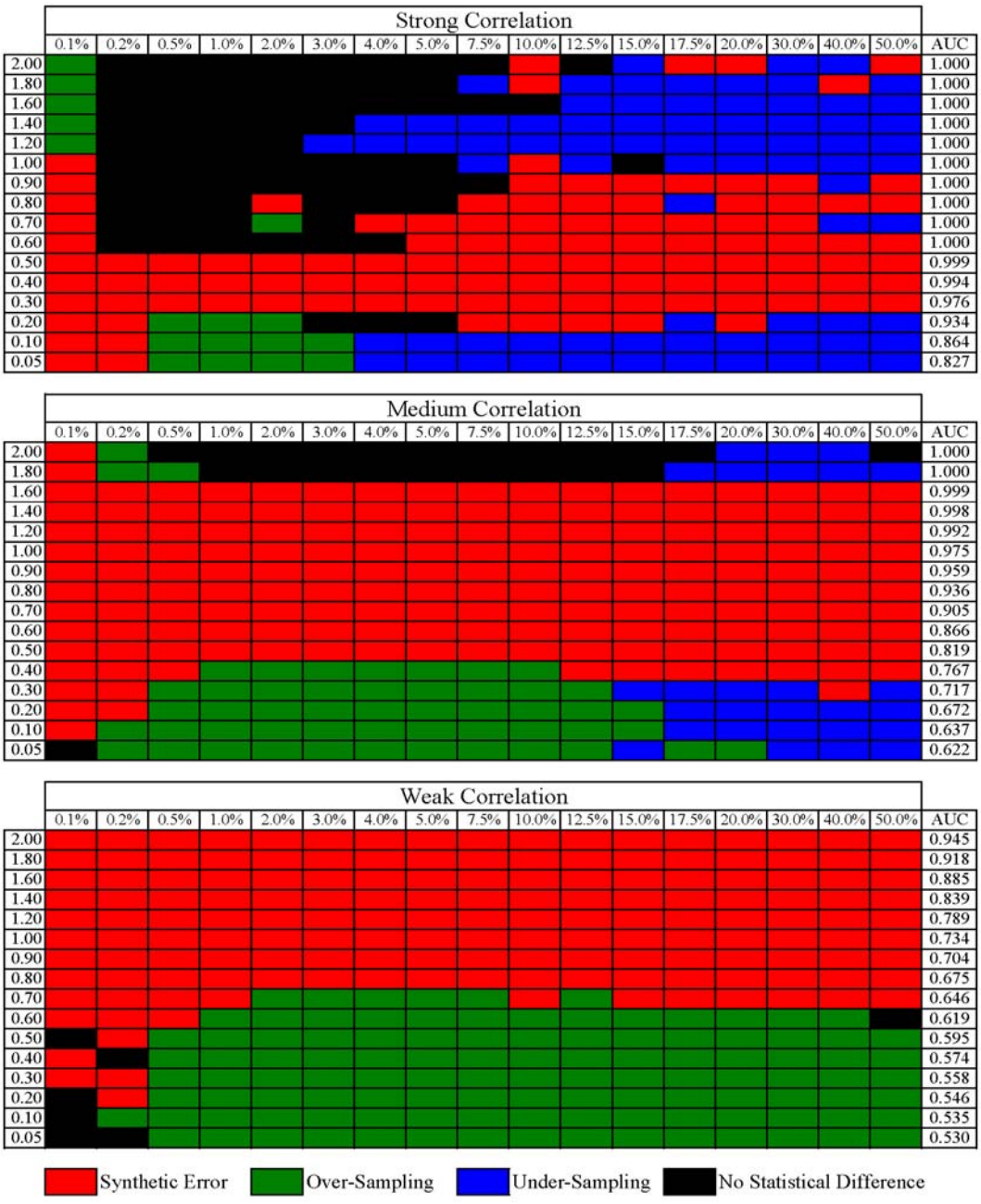

Figure 6.3 Preferred Algorithms over Varying Degrees of Correlation, Size of Error, and Probability of Error

6.3.2. Over-Sampling Compared to Under-Sampling

Figure 6.3 suggests under-sampling the majority class is only effective when the difference in class balance is small and when the data are more strongly correlated. As the correlation became weaker or the between-class imbalance became smaller, conditions more likely to be observed in the clinical laboratory, over-sampling the minority-class appears superior to under-sampling the majority-class. To test this hypothesis, we performed a two-sided Wilcoxon Rank-Sum test between minority-class over-sampling and majority-class under-sampling to determine if a statistical difference existed between these two algorithms and the conditions where one method was preferred to the other (Rosner 2000).

When the system is highly correlated, Figure 6.4, minority-class over-sampling is statistically superior to majority-class under-sampling when the minority-class probability is less than about 3%. When the minority-class probability reaches about 8%, majority-class under-sampling is statistically superior. Under-sampling the majority-class removes examples of the major class, which results in the Bayesian network fitting a Gaussian model using fewer data points. As more and more majority-class examples are removed, the uncertainty in the Bayesian network's parameters increases and performance declines as it over-fits the remaining majority-class data elements. Hence, under-sampling the majority-class is not expected to be the best algorithm when the minority-class probability is very low.

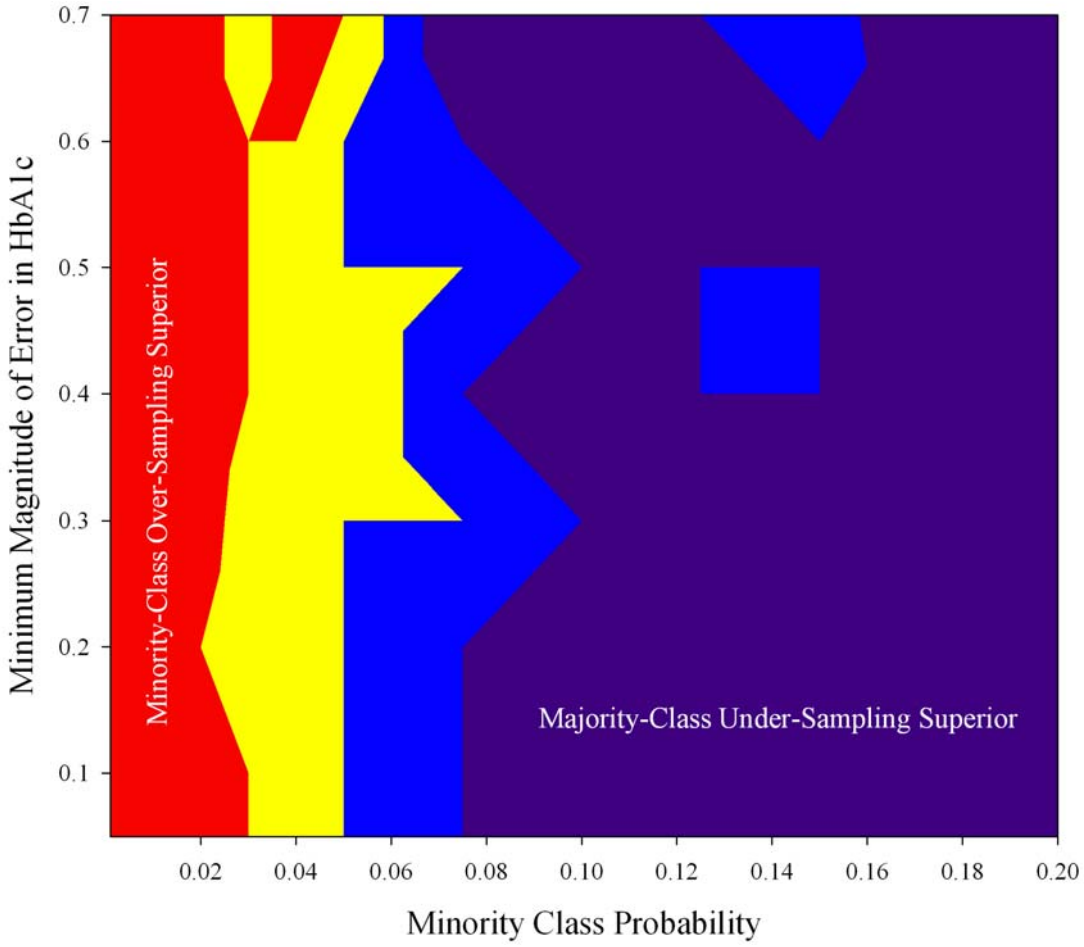

Figure 6.4 Statistical Difference in Highly Correlated System

Over-sampling the minority-class duplicates examples of the minority-class, but does not create novel examples. The between-class imbalance is reduced in the training dataset and the Bayesian network parameters are largely unaffected. For example, if one takes 100 random numbers and duplicates each number n times, the mean and variance will not change as n changes. The Bayesian network will, however, also over-fit the minority-class examples when the minority-class probability is very low. Hence,

over-sampling the minority-class is also not expected to be the best algorithm when the minority-class probability is very low.

In a weakly correlated system, Figure 6.5, over-sampling the minority-class is statistically superior to under-sampling the majority-class at virtually all parameter combinations. This would suggest that discarding training examples in a weakly correlated system results in over-fitting the remaining training examples, resulting in reduced performance.

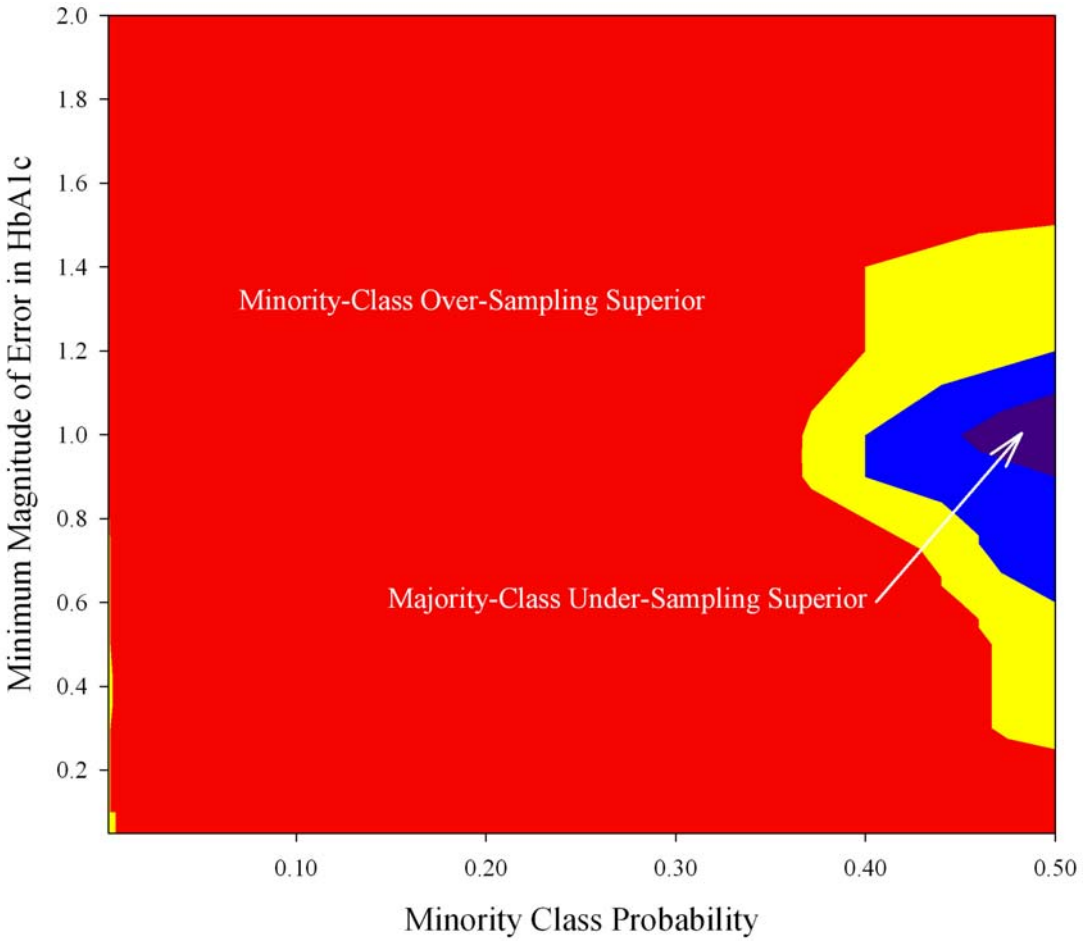

Figure 6.5 Statistical Difference in Weakly Correlated System

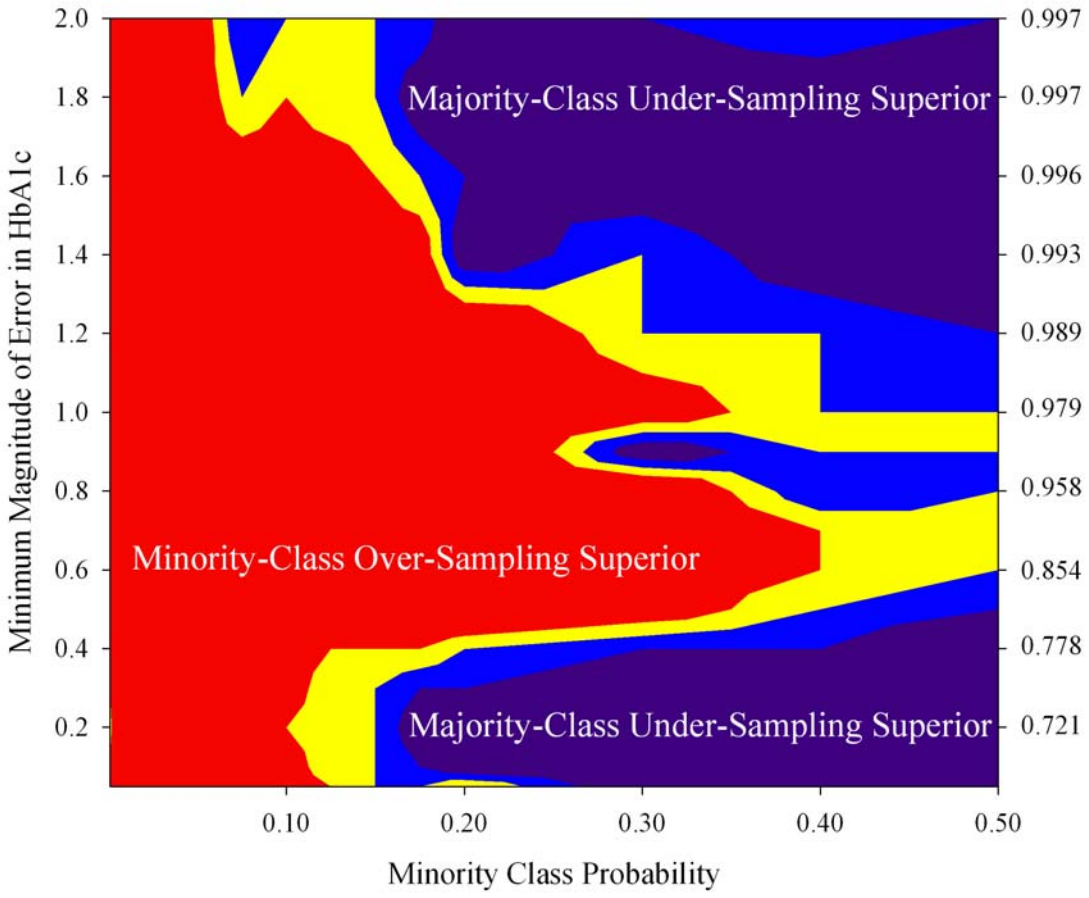

Figure 6.6 Statistical Difference in Moderately Correlated System

The moderately correlated system, Figure 6.6, shows a significantly more complex relationship between minority-class over-sampling and majority-class under-sampling than observed in the previous two systems. When the size of the disjunct is relatively small, corresponding to poor performance, majority-class under-sampling is statistically superior as long as the minority-class probability is relatively high, at least 17%. When the disjunct size is large, majority-class under-sampling is again statistically superior as long as the minority-class probability is at least 17%. This

suggests that under-sampling is only effective when the minority class probability is above a threshold that depends on the system's correlation and the disjunct size. From this complex relationship, we conclude that both approaches must be evaluated in the domain of interest. In the clinical laboratory domain, a domain with very low minority class probabilities and generally poorer correlation, over-sampling the minority-class is expected to be statistically superior to under-sampling the majority-class. Therefore, we only compare the performance of synthetic error generation to minority-class over-sampling.

6.3.3. Superiority of Synthetic Error Generation

Errors in the clinical laboratory are rare, generally estimated at about 1%, so we limit our comparison to where the minority-class probability is between 0.1% and 2.0%. With this assumption, as discussed in the previous section, majority-class under-sampling is always expected to be statistically inferior to minority-class over-sampling. Therefore, we limit our evaluation of the synthetic error generation method to just the minority-class over-sampling method.

In a strongly correlated system, Figure 6.7, the difference between the two methods is generally statistically insignificant. At very low minority-class probabilities, synthetic error generation is superior whereas minority-class over-sampling is superior at larger minority-class probabilities, if the magnitude of error is small. To estimate the performance difference between these two systems, we calculated the difference in the average areas under the ROC curves, Figure 6.8. When minority-class over-sampling is

superior, the difference in area in the ROC curve is only about 0.01 whereas the synthetic error generation method, when statistically superior, improves performance between 0.01 and 0.04. Since the performance difference between the minority-class over-sampling method and the synthetic error generation method is very small when the system is highly correlated and the minority-class probability is less then 2.0%, either method can be expected to produce good results in training an clinical laboratory autoverification system.

In a moderately correlated system, Figure 6.9, the performance gains of the synthetic error generation method are readily apparent. For all regions when the area under the ROC curve is expected to be between about 0.750 and 0.999, synthetic error generation is statistically superior and can be expected to improve performance, Figure 6.10, by 0.01 to 0.08. In a weakly correlated system, Figure 6.11, synthetic error generation is statistically superior when the area under the ROC curve exceeds 0.675 and adds, Figure 6.12, between 0.01 and 0.14 to the area under the ROC curve.

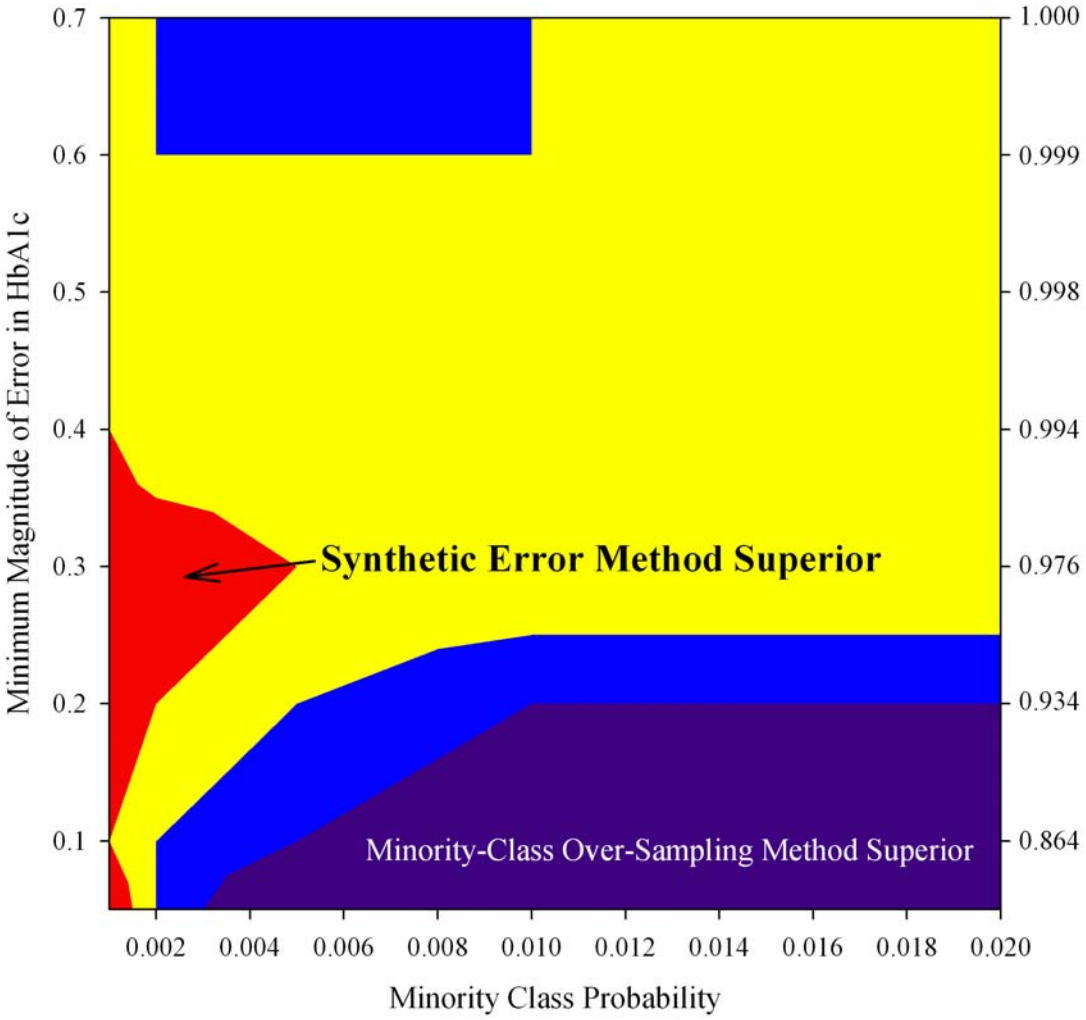

Figure 6.7 Statistically Significant Differences between Synthetic Error and Minority-Class Over-Sampling in Strongly Correlated System

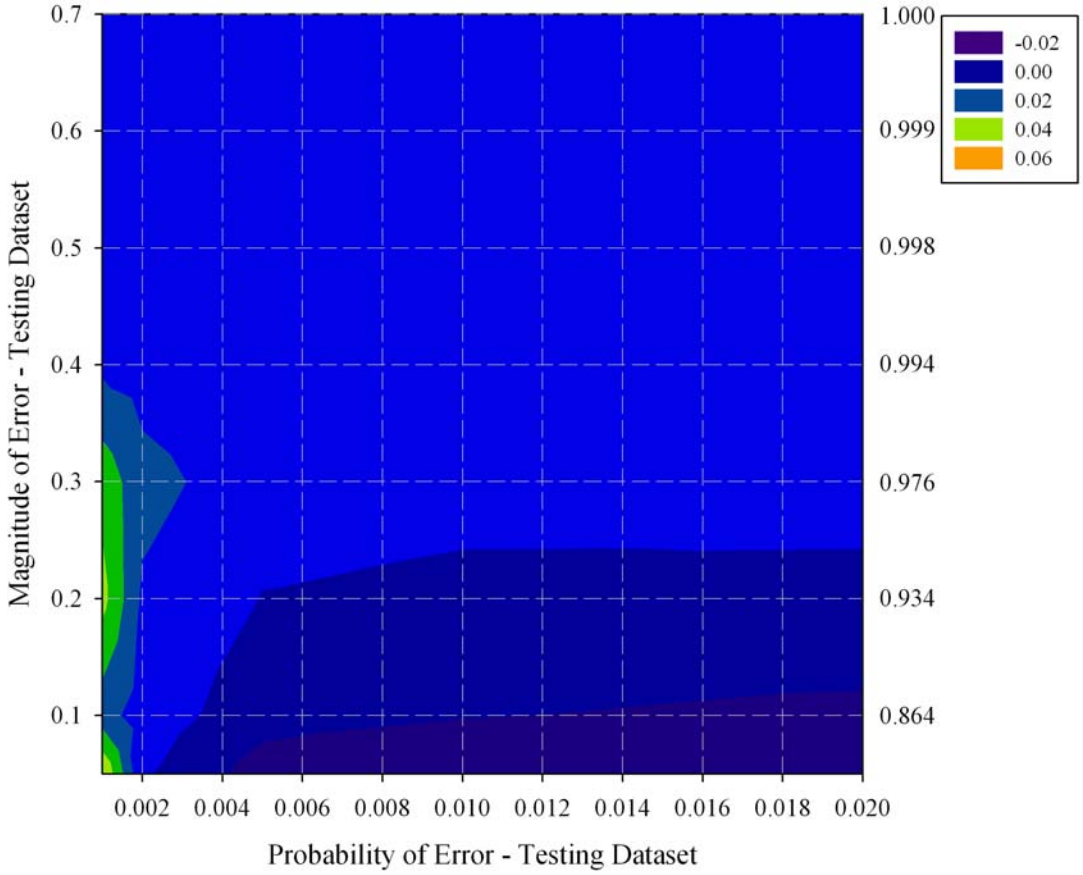

Figure 6.8 Difference in Area under ROC Curves between Synthetic Error and Minority-Class Over-Sampling in Strongly Correlated System

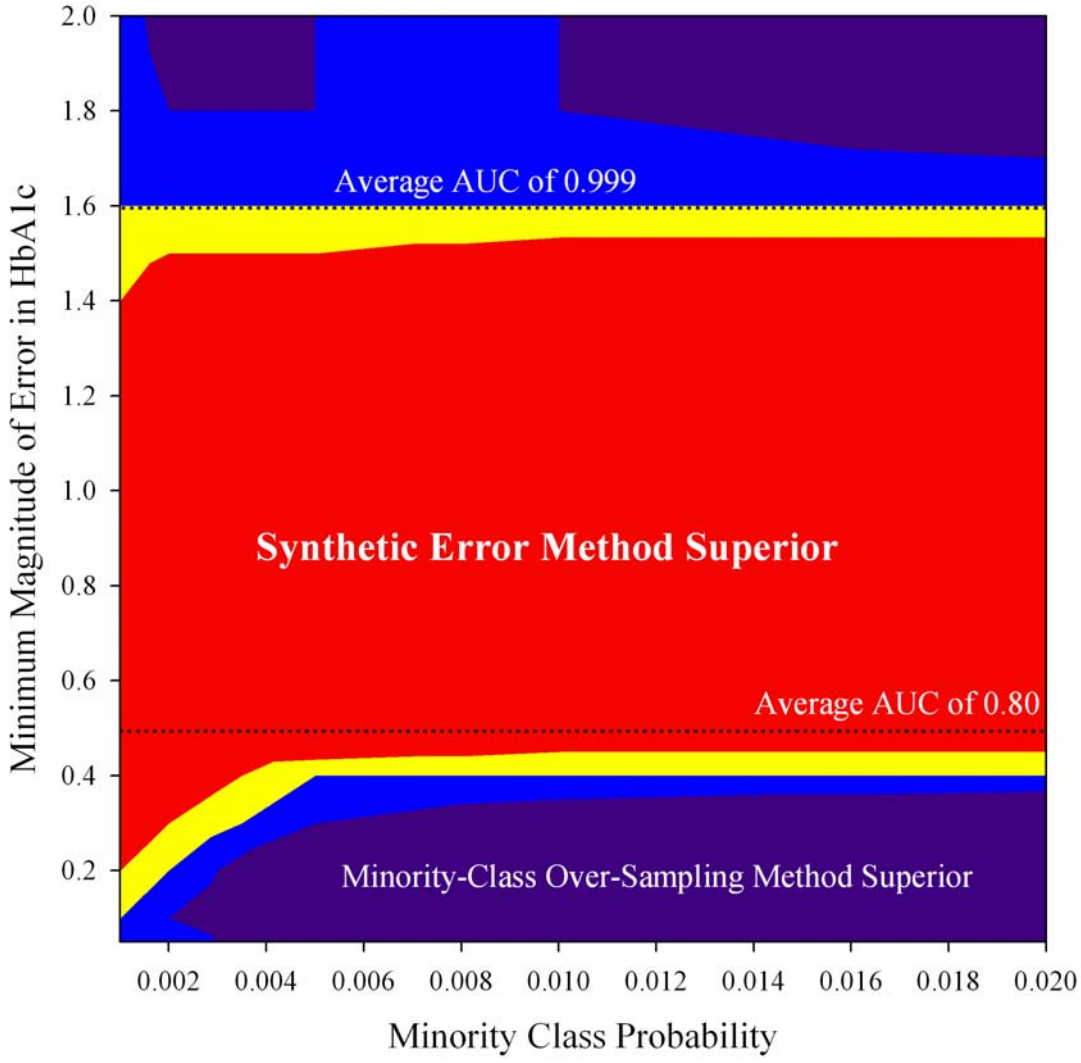

Figure 6.9 Statistically Significant Differences between Synthetic Error and Minority-Class Over-Sampling in Moderately Correlated System

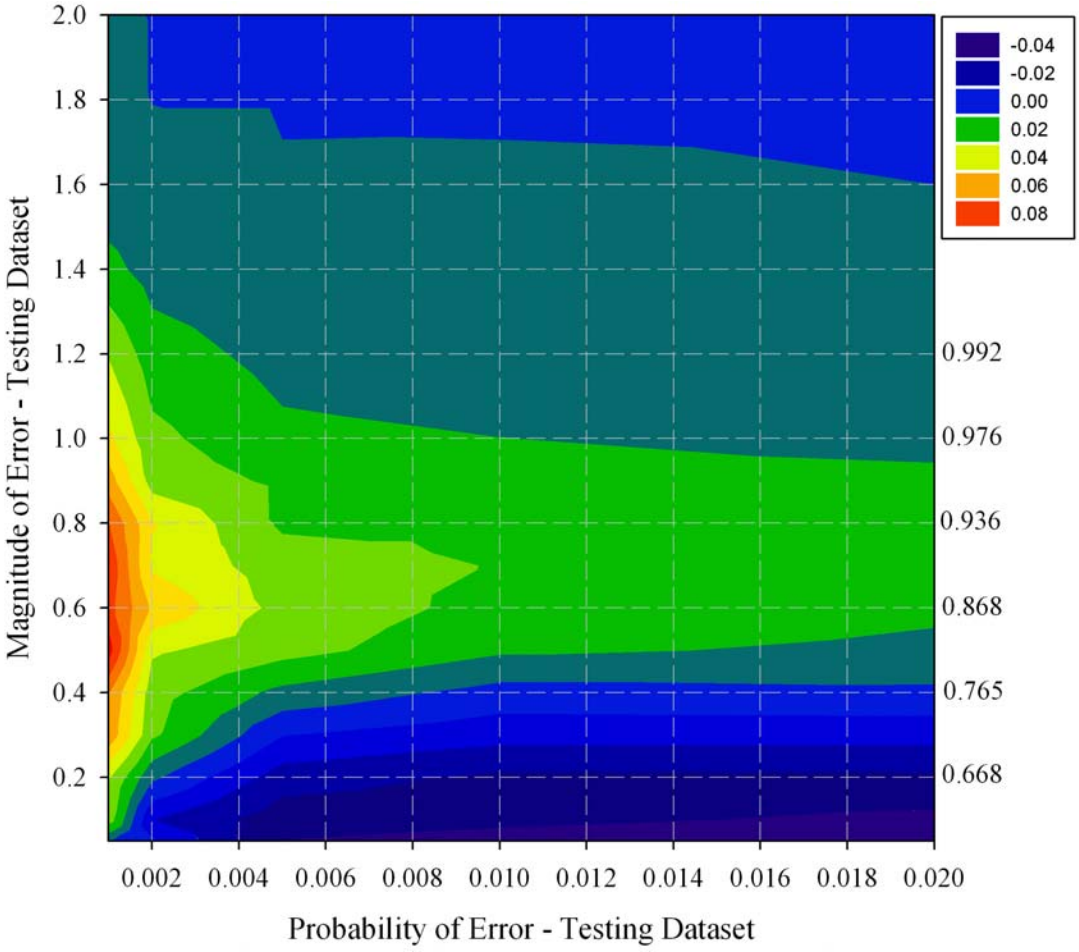

Figure 6.10 Difference in Area under ROC Curves between Synthetic Error and Minority-Class Over-Sampling in Moderately Correlated System

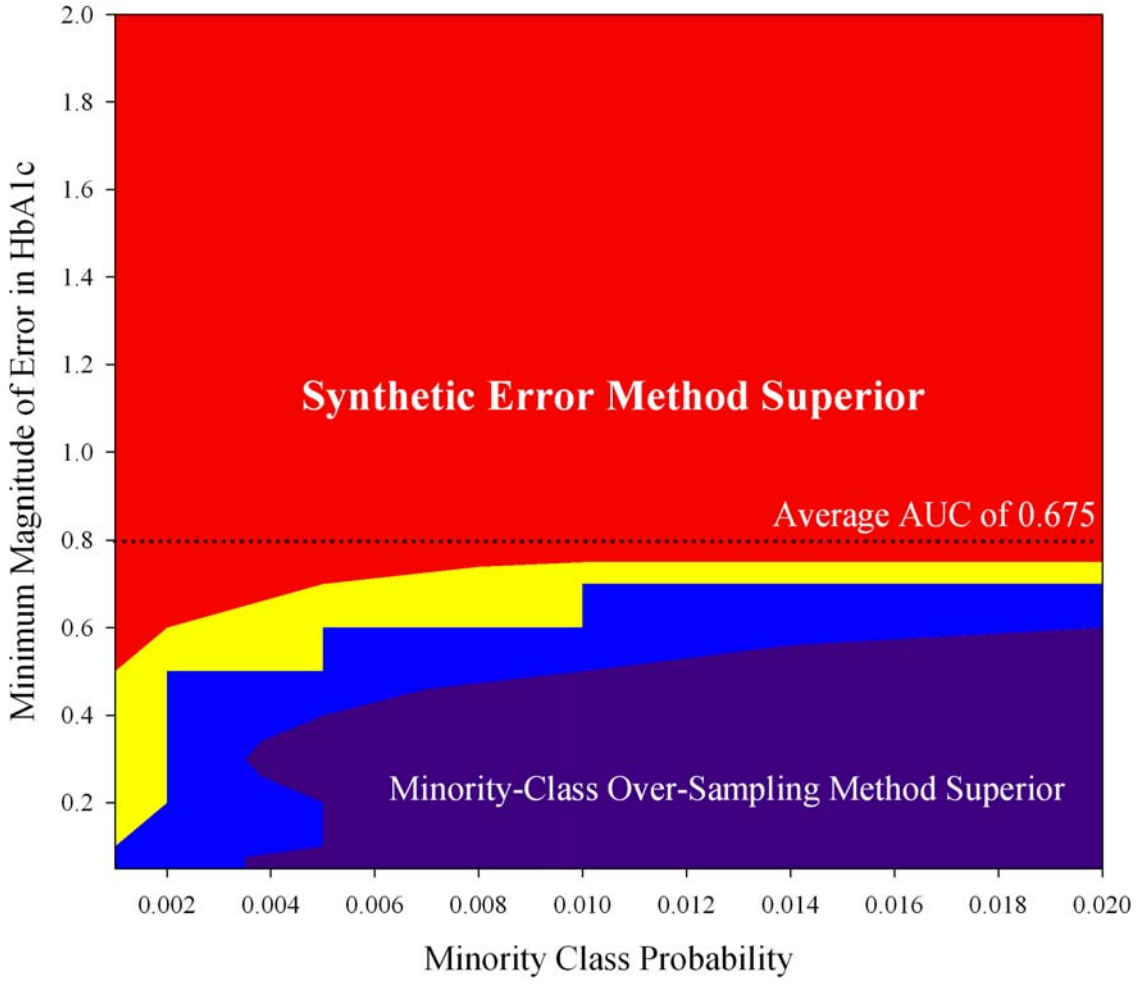

Figure 6.11 Statistically Significant Differences between Synthetic Error and Minority-Class Over-Sampling in Weakly Correlated System

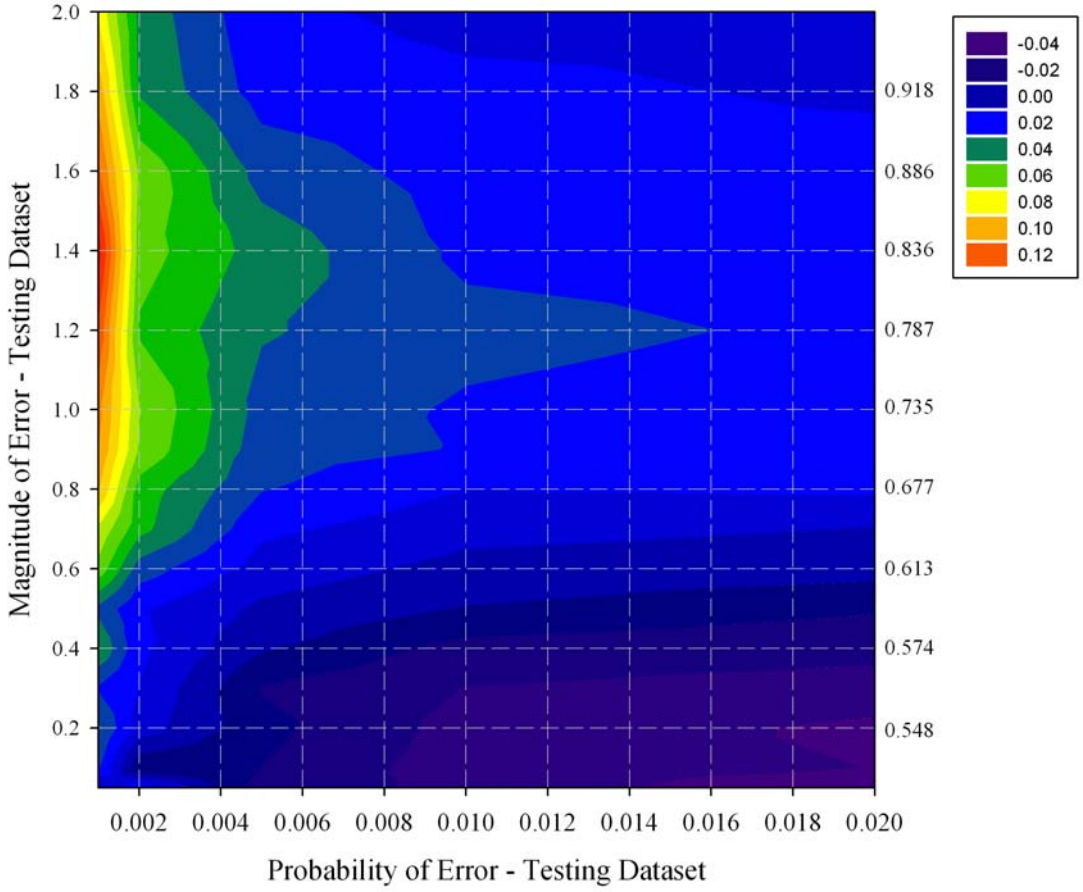

Figure 6.12 Difference in Area under ROC Curves between Synthetic Error and Minority-Class Over-Sampling in Weakly Correlated System

6.4. Summary

In this chapter, we have shown that the synthetic error generation method is generally statistically superior, though with only a modest gain in performance, to the standard minority-class over-sampling and majority-class under-sampling methods in detecting errors under a broad range of conditions likely to be observed in the clinical laboratory. Under conditions when the synthetic error generation method is not the best method, its performance is only slightly below the best method. Errors in the clinical laboratory are very rare and have a complex multifactorial etiology that makes their detection very difficult. In addition, as discussed in Chapter 2, there is no gold standard that can be applied to an existing dataset to identify clinical laboratory errors. Even if a gold standard existed and, at a great cost, a realistic training dataset could be created, the results of this chapter indicate this is neither necessary nor beneficial. Synthetic error generation is statistically superior to existing methods in training a Bayesian network to identify errors in the clinical laboratory.

Chapter 7: Comparison of Synthetic Error Generation Against Laboratory Experts

In Chapter 2, we discussed that only about half of the clinical laboratories use an autoverification system, which is virtually always based on rules, and that rule-based systems are not capable of handling the inherent uncertainty within the domain. In Chapter 6, we saw the superiority of the synthetic error generation method compared to standard methods for training a Bayesian network to identify errors in the clinical laboratory. In this chapter, we survey laboratory experts to estimate the desired performance characteristics for an autoverification system and evaluate their performance against the synthetic error generation method using simulated results. For a clinical laboratory using an autoverification system to be accredited, their autoverification system must be evaluated and authorized by their laboratory director prior to its use (Commission on Laboratory Accreditation 2006). The core requirement is that it check if a result meets "*laboratory-defined acceptance parameters*" and require expert review of results failing to meet its acceptance parameters (Commission on Laboratory Accreditation 2006).

7.1. Laboratory-Defined Acceptance Parameters

By definition, laboratory-defined acceptance parameters are specific to a given laboratory. Each laboratory has its own desired sensitivity and specificity for detecting errors in the laboratory results. However, if we assume that each laboratory expert has the same capability to detect errors, then their receiver operating characteristic (ROC)

curves will be the same and their "laboratory-defined acceptance parameters" will be seen as operating at different points along the same curve. For example, Laboratory A may desire a more conservative autoverification system than Laboratory B, Figure 7.1, and choose an operating point on the ROC curve with a higher sensitivity (true positive rate), but with a correspondingly lower specificity (1 – false positive rate). This would result in Laboratory A's system detecting more errors, but also results in fewer results being autoverified.

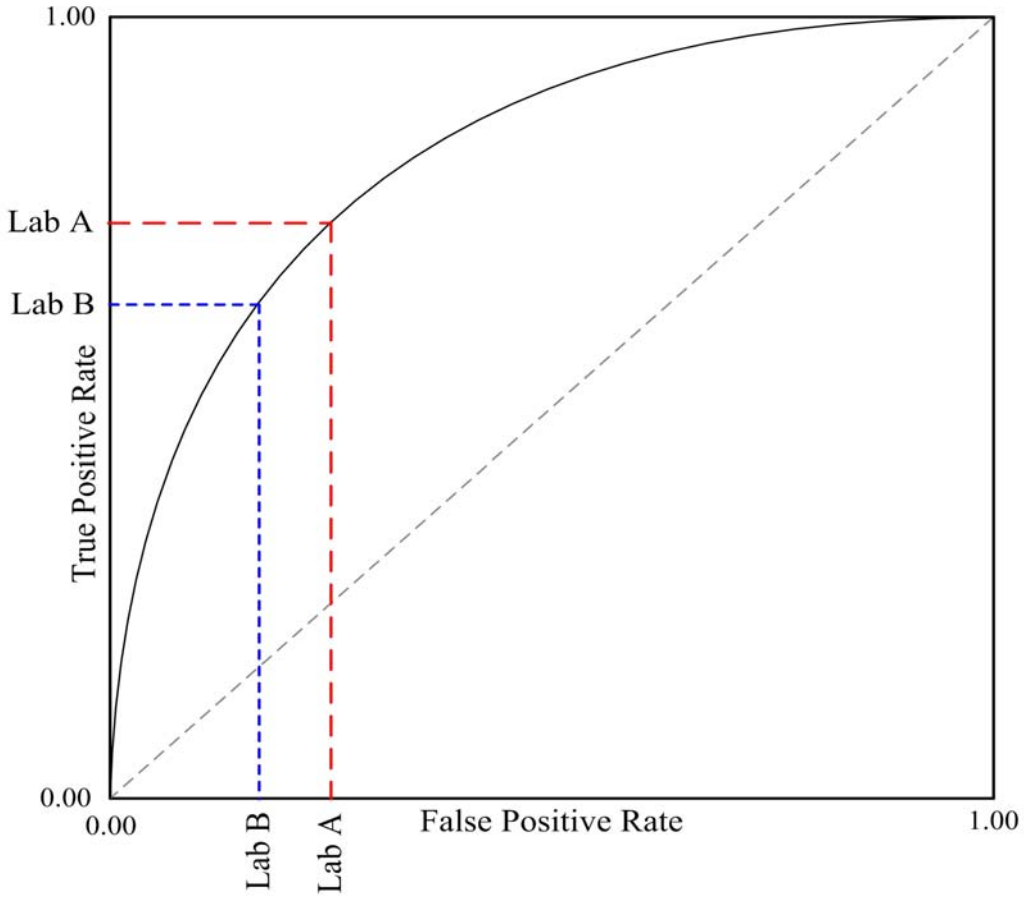

Figure 7.1 Choosing an Operating Point on the ROC Curve

We conducted a survey of potential laboratory experts to estimate the range of desired operating characteristics for an autoverification system. The survey, Appendix II, was advertised via email to 280 people with email addresses currently listed in the American Association for Clinical Chemistry's Laboratory Information Systems and Medical Informatics Division. Twenty-eight (10%) of the respondents completed at least part of the survey with one respondent not answering eight questions and another respondent not answering one question. The remaining twenty-six respondents answered all questions posed to them.

7.1.1. Cost and Frequency of Laboratory Errors

All twenty-eight respondents indicated their laboratory's size and four (14%) were from small laboratories, thirteen (46%) from medium laboratories, and eleven (40%) were from large laboratories. In line with estimates, twenty-three (85%) of the twenty-seven answering respondents felt laboratory errors rates are less then 0.5%, and only one respondent felt laboratory error rates exceed 1.5%. However, sixteen (59%) of the twenty-seven responding laboratory experts appeared to underestimate the percentage of errors causing harm, estimated at about 6%, at less than 1% (Astion, Krueger-Nielsen S et al. 2004).

Table 7.1 Percentage of Errors Resulting in Patient Harm by Laboratory Size (n = 24 Respondents)

	< 1%	5%	10%	> 10%
Small	1	0	1	1
Medium	9	1	2	1
Large	4	1	3	0
Total	14	2	6	2

Only eleven (39%) of the twenty-eight respondents indicated they currently use an autoverification system and larger laboratories were more likely than smaller laboratories to use one. When reviewing data via an autoverification system or expert review, twenty-six (93%) of the twenty-eight respondents indicated they have access to previous clinical results performed in their laboratory. Access to previous clinical laboratory results increases the laboratory's ability to detect errors. Of laboratories with access to historical results, twelve (46%) of the twenty-six also have access to the patient's electronic medical record (EMR). Again, larger laboratories tend to have greater access to the patient's data. Access to a patient's EMR further enhances the clinical laboratory's ability to predict values and, therefore, detect errors.

All respondents utilizing an autoverification system were satisfied or very satisfied with its performance. We then asked them to explain the reason for their satisfaction. A sampling of comments from six of the nine answering respondents:

- “Works … and it’s FAST”
- “Reduces human errors and increases consistency”
- “Autoverification … works reasonably well where we use it and for what it is capable”
- “While I am satisfied with the performance of the Autoverification system in use, I feel it is rather limited when it comes to setting up the "Rules" used to evaluate whether a result can be autoverified.”
- “Although none of these systems (is) perfect, they have improved the quality of review and alleviated much of the drudgery associated with test verification.”
- “Handles release criteria as specified well but certain criteria are not easily modeled.”

Laboratory experts appear to be satisfied with their autoverification systems because they are fast and have “alleviated much of the drudgery” of reviewing results, but several respondents commented on the difficulty in establishing effective rules. From respondents’ comments, we hypothesize that users would be satisfied with a synthetic error generation-based Bayesian autoverification system because its speed is on par with a rule based system and its ability to learn from data will simplify setup.

When asked what percentage of results they expect to be automatically released by an autoverification system, half of the respondents indicated they would be satisfied if approximately 75% of the results were autoverified. This autoverification rate corresponds to a false positive rate of about 0.75, which indicate that clinical

laboratories highly value the detection of laboratory errors. Using the optimal classification threshold discussed in Section 4.3.2 (page 68), the rarity of laboratory errors moves the operating point to the left along the false positive rate dimension and a misclassification cost imbalance moves the operating point to the right. Since the operating point appears far to the right of where it would appear assuming equal misclassification costs, the misclassification cost associated with laboratory errors must be much greater than the misclassification cost associated with acceptable results.

Twenty-five (89%) of the twenty-eight respondents indicated that it was somewhat or very important for the autoverification to explain why a result was flagged as potentially in error. Furthermore, eighteen (67%) of the twenty-seven believe it somewhat or very important to provide possible causes for the perceived error. Current rule-based autoverification systems are able to state the criteria that caused a result to be flagged, but are not able to hypothesize as to the cause of the error. Finally, twenty-two (81%) of the twenty-seven respondents indicated that an autoverification system should be able to detect a 10 – 20% error in an analyte such as cholesterol. In analytes with higher biological variability, such as triglyceride, twenty-four (89%) of the twenty-seven respondents indicated a 10 – 30% error should be detected at least 50% of time. For larger errors, the autoverification system should detect errors more readily. Based on the results of the survey, we estimate that an effective autoverification system should operate with a specificity of approximately 0.75 and a sensitivity of approximately 0.50 for an appropriate minimum magnitude of error for the analyte.

7.2. Experimental Design

To evaluate the synthetic error generation method's performance against laboratory experts in detecting errors in clinical laboratory data, we asked laboratory experts who completed the survey if they were able to review glucose and glycosylated hemoglobin (HbA1c) results to identify errors. Since the survey and evaluation were required to be anonymous, we were not able to ensure those who completed the evaluation where indeed qualified. Subject volunteers completed one of two randomly selected comparisons, which are listed in Appendix III and Appendix IV. Each of the two surveys consisted of 60 questions split over two sections. Each of the four sections started with the question "*Consider a pre-diabetic population where the average glucose is 103 mg/dl (standard deviation 11mg/dl) and the average glycosylated hemoglobin (HbA1c) is 5.9 (standard deviation 0.2). For each of the 30 sets below, what is your belief that the HbA1c value is in error given the fasting glucose value?"*. For each of the 60 questions, respondents selected one of *Definitely Not an Error, Probably Not an Error, Neutral, Probably an Error, Definitely an Error*. As discussed later, we are confident all eleven experts who completed the evaluation understood the questions and were qualified to participate, since they performed reasonably well.

The training and testing datasets were artificially created using a model of a pre-diabetic population in order to provide a clean dataset known to be free from errors and one with sufficient variability for a meaningful evaluation. We again, based on the results of the Diabetes Control and Complications Trial (DCCT) results, assumed a

linear relationship, equation 7.1, between glucose and glycosylated hemoglobin in the region of interest (Rohlfing, Wiedmeyer et al. 2002). The parameters of the linear relationship were determined from an analysis of glucose and glycosylated hemoglobin (HbA1c) results in a pre-diabetic population and are similar to published data for a pre-diabetic group (The Diabetes Prevention Program Research Group 2000). As before, we randomly selected glucose values from a Gaussian distribution and used equation 7.1 to derive an HbA1c result. Table 7.2 lists the parameters of the original, training, and testing datasets.

$$\text{HbA1c} = 4.22 + 0.01604 \times \text{Glucose} + \varepsilon \qquad (7.1)$$

Table 7.2 Parameters of Target and Artificial Dataset

	ORIGINAL	TRAINING	TESTING
Glucose Average	103.4	103.3	101.4
Glucose Standard Deviation	11.3	11.3	9.5
HbA1c Average	5.87	5.88	5.86
HbA1c Standard Deviation	0.22	0.26	0.38
Pearson's Correlation Coefficient	0.361	0.629	0.521
Number of Results	> 3,000	10,000	120

The procedure used to evaluate the performance of the synthetic generation method against laboratory experts, Figure 7.2, is similar to the methods used in previous chapters. We continue to use the Bayesian network described in the previous chapter (section 5.2, page 79). To train the Bayesian network, we used the following procedure:

1. Create an initial error-free dataset by randomly selecting from the glucose distribution and using equation 7.1 to calculate an HbA1c value.
2. Add synthetic errors with an error magnitude of 0.50 and a frequency of 50%.
3. Learn the parameters of the Bayesian network using 10-fold cross-validation.

We created the testing dataset using the following procedure:

1. Create an initial error-free dataset by randomly selecting from the glucose distribution and using equation 7.1 to calculate an HbA1c value.
2. Add sample switching errors with a minimum error magnitude of 0.50 and a frequency of 40%.

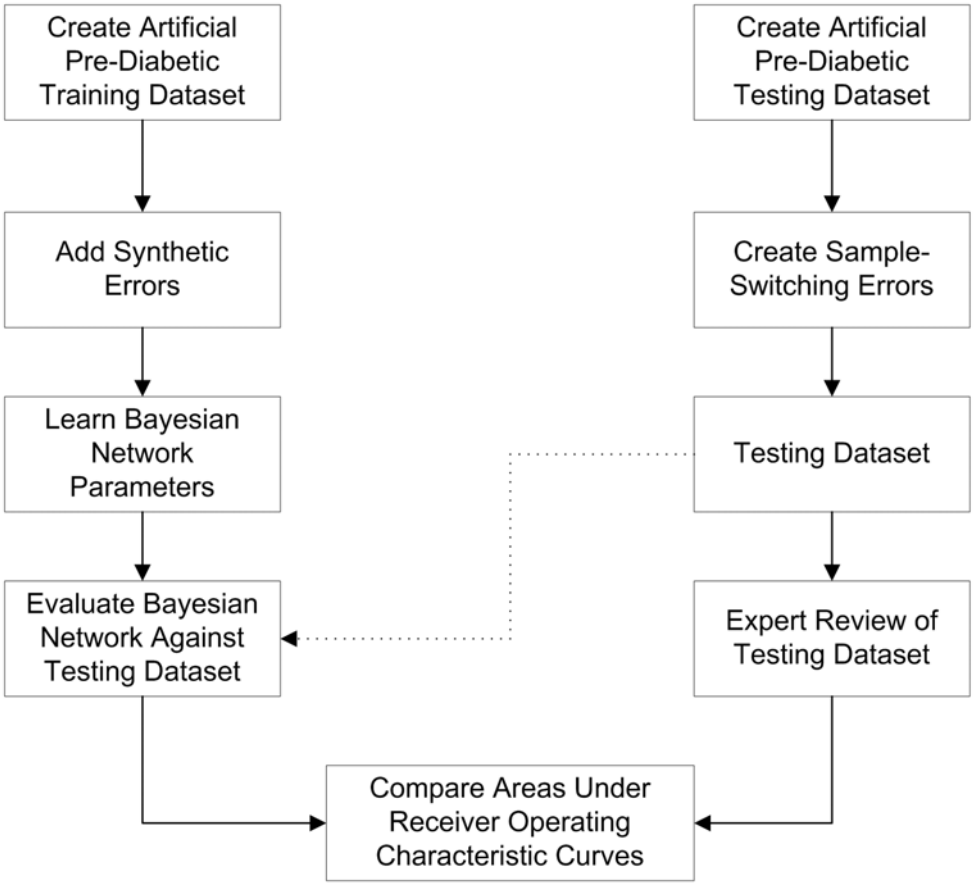

Figure 7.2 Evaluation Procedure

We used a high minority-class probability in the testing dataset to provide sufficient power for analysis and since, from the results in Chapter 5, the performance of the synthetic error generation method would be unaffected. Had we selected a lower minority-class probability for the testing dataset, the experts would have needed to review significantly more results to obtain sufficient power to observe a difference in performance. We do not know the extent to which an unrealistic minority-class probability affected the performance of the laboratory experts, or the extent to which the

format of the question mitigated this affect. We split the testing dataset in half to present a reasonably sized (sixty questions) comparison to the experts. For the sixty pairs of glucose and HbA1c results presented, the subjects annotated each HbA1c result as definitely an error, probably an error, neutral, probably not an error, and definitely not an error. Four respondents answered the same for each question and were excluded from further analysis. Five respondents satisfactorily completed comparison #1 and six respondents completed comparison #2. Two respondents did not answer a total of three questions for unknown reasons. When respondents did not answer a question, we assumed they were neutral in their belief of an error. The synthetic-error generation trained Bayesian network evaluated both comparison sets, producing the probability that a value is in error.

7.3. Statistical Analysis

By varying the classification threshold between 0% and 100%, we produce an ROC curve for the Bayesian network's performance for each of the two comparisons. The laboratory experts, however, did not provide a probability for use in creating an ROC curve. As discussed in Section 4.3.1 (page 67), since the rating system is ordinal, we created an ROC curve for each expert by computing their sensitivity (true positive rate) and specificity (1 – false positive rate) as the classification threshold is varied from "definitely an error" to "definitely not an error". For example, using the classification threshold of "neutral", we classified as an error all entries the experts labeled as "definitely an error", "probably an error", or "neutral" and computed the resulting true

positive rate and false positive rate. We computed an average ROC curve by averaging each expert's linearly interpolated true positive rate at each false positive rate point between 0.0 and 1.0. The experts' results for comparison #1 are displayed in Figure 7.3 and Table 7.3. Comparison #2 results are displayed in Figure 7.4 and Table 7.4.

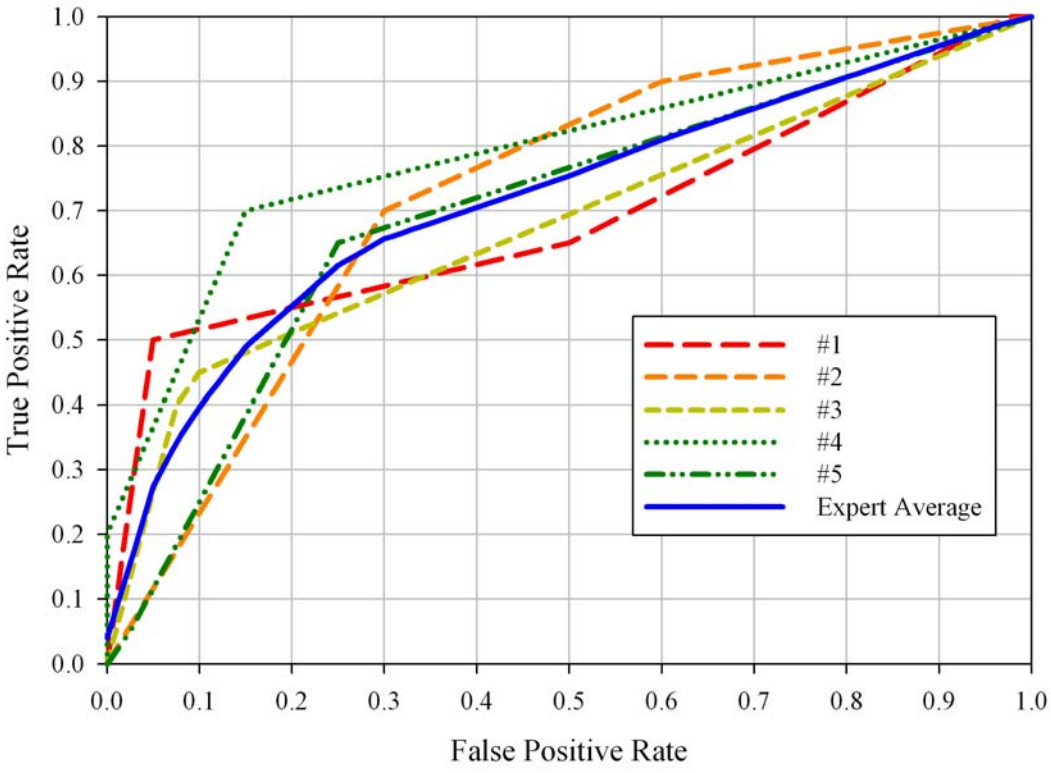

Figure 7.3 Comparison #1: Expert Performance

Table 7.3 Comparison #1: Experts' Area Under the ROC Curve

#1	#2	#3	#4	#5	AVERAGE
0.6875	0.7250	0.6781	0.7900	0.6981	0.7158

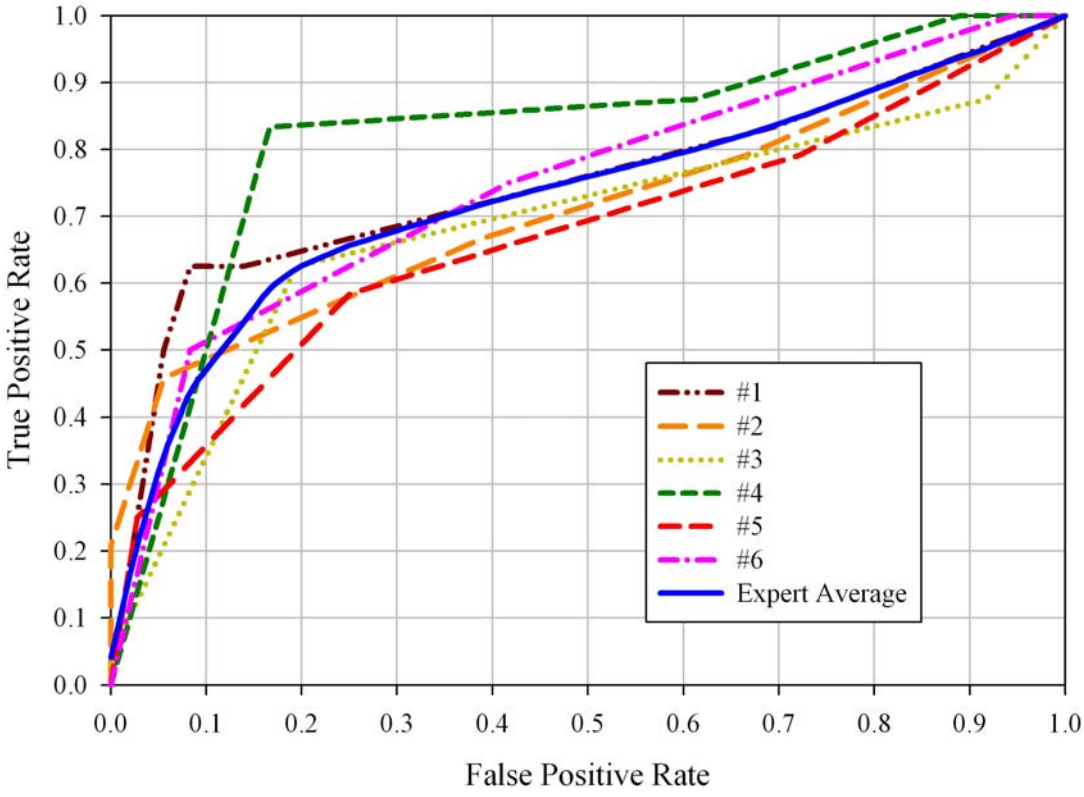

Figure 7.4 Comparison #2: Expert Performance

Table 7.4 Comparison #2: Experts' Area Under the ROC Curve

#1	#2	#3	#4	#5	#6	AVERAGE
0.7494	0.7072	0.6846	0.8206	0.6696	0.7465	0.7296

From Table 7.3, we compute a standard error of 0.020 in the average area under the ROC curve for comparison #1 and from Table 7.4, a standard error of 0.022 for comparison #2. The average expert's performance against the synthetic error autoverification system is displayed in Figure 7.5 for comparison #1 and Figure 7.6 for comparison #2. In both comparisons, the synthetic error autoverification system outperformed the experts.

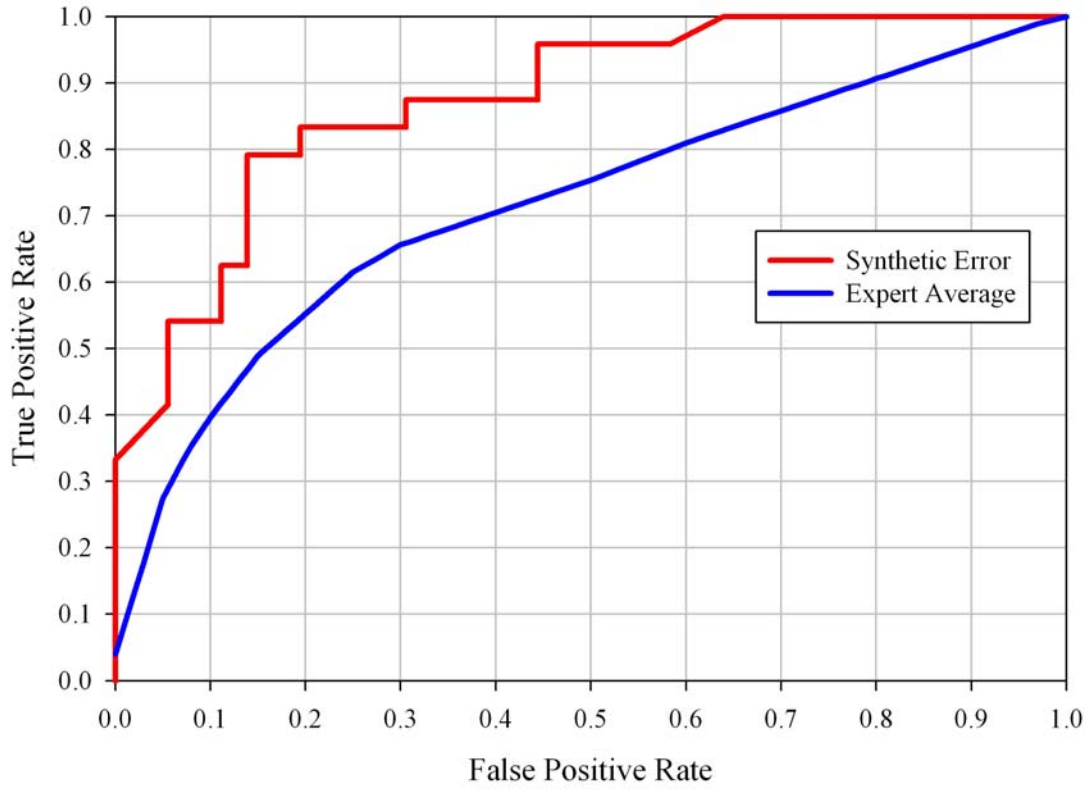

Figure 7.5 Comparison #1 Between Synthetic Error and Laboratory Experts

For comparison #1, the synthetic error autoverification system produced an area under the ROC curve of 0.8750 compared to the expert's 0.7158. From Hanley and

McNeil (1983), which is summarized in Appendix I, we estimate the standard error in the synthetic error autoverification system at 0.05. The critical ratio z is calculated as

$$z = \frac{A_1 - A_2}{\sqrt{SE_1^2 + SE_2^2 - 2rSE_1SE_2}} \tag{7.2}$$

where A_x is the area under the ROC curve, SE_x is the standard error, and r is the estimated correlation between the two areas (Hanley and McNeil 1983). Using a conservative assumption that the correlation is 0.0, equation 7.2 produces a critical ratio of 2.91, which indicates a statistically significant difference between the two with $p <$ 0.002. At a specificity of 75%, the average expert had a sensitivity of 62% whereas the synthetic error autoverification system had a sensitivity of 84%, a statistically significant increase in performance.

For comparison #2, the synthetic error autoverification system produced an area under the ROC curve of 0.9531 compared to the average expert's 0.7296. The testing dataset used for comparison #2 was created by randomly splitting the original testing dataset and, by chance, did not contain errors near the expected value, as can be observed in Figure 7.7. This difference resulted in the synthetic error autoverification system performing better than in comparison #1. Using equation 7.2 with a standard error estimate of 0.05 for the synthetic error autoverification system and conservatively assuming a correlation of 0.0, the critical ratio z is calculated at 4.15. This again

indicates a statistically significant difference between the two systems with $p <$ 0.00002. At a specificity of 75%, the laboratory experts achieved a sensitivity of 65% whereas the synthetic error autoverification system achieved a sensitivity of 100%.

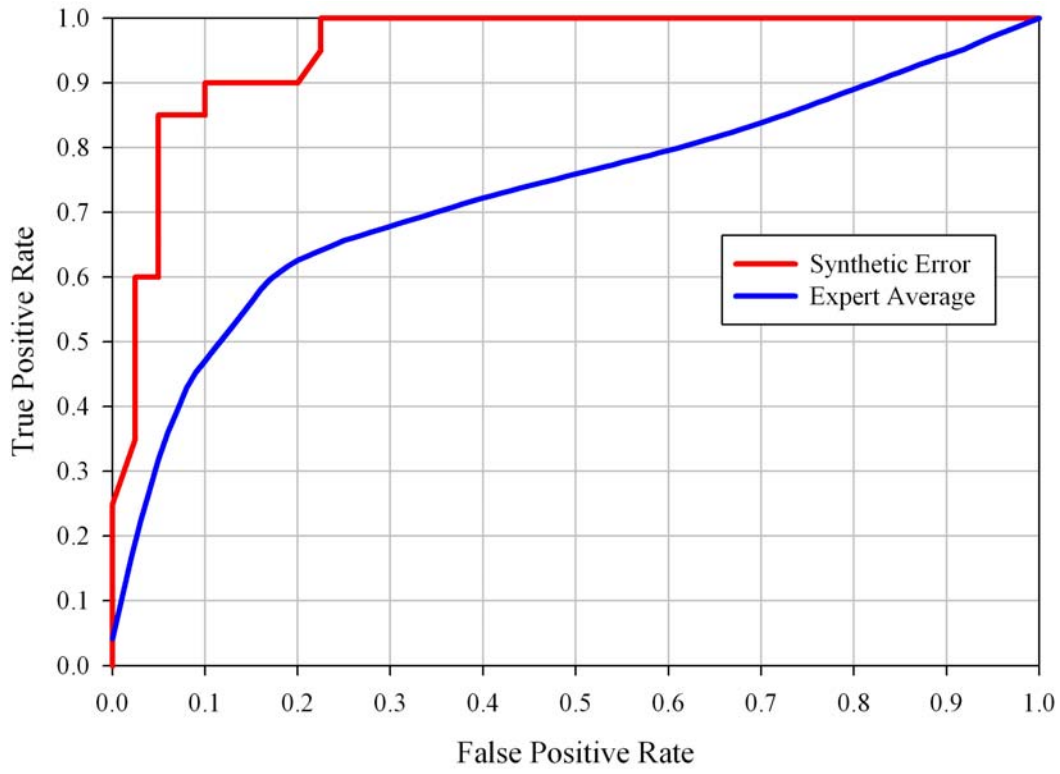

Figure 7.6 Comparison #2 Between Synthetic Error and Laboratory Experts

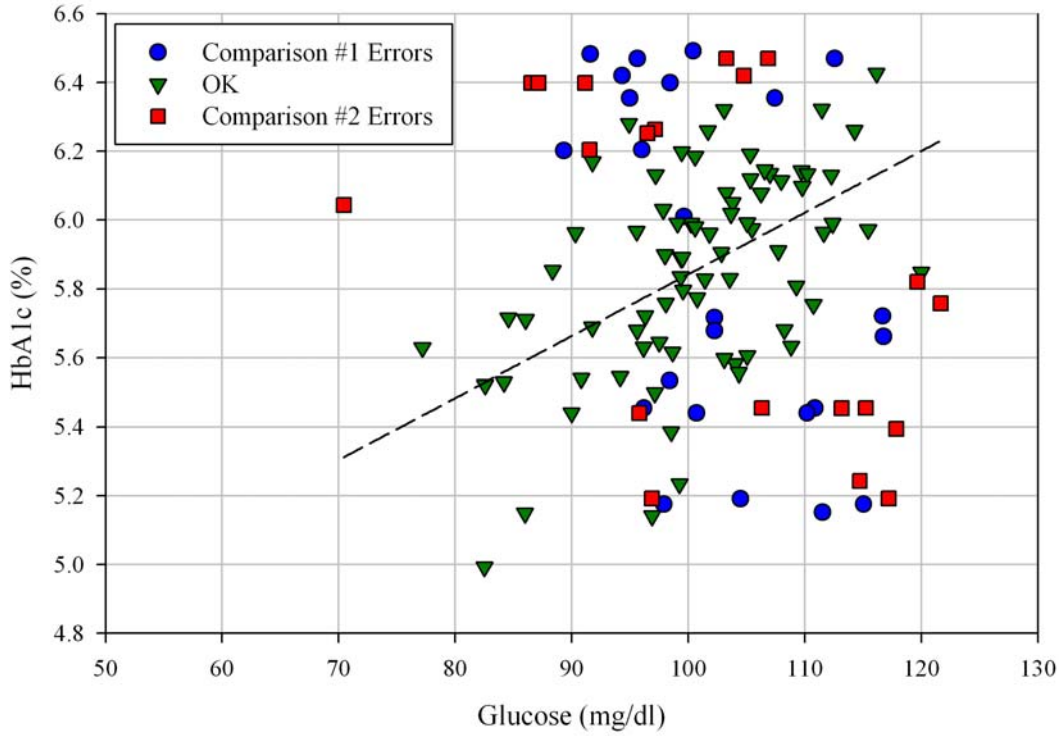

Figure 7.7 Scatter Plot of Testing Datasets #1 and #2

7.4. Summary

Previous chapters have shown the statistically significant improvement in performance of the synthetic error generation method over standard methods for training a Bayesian network to detect rare events. In this chapter, we showed that a Bayesian network trained by the synthetic error generation method produces an autoverification system that, for the analyte considered, is statistically superior to laboratory experts. Most clinical laboratories do not use autoverification systems, but those that do are satisfied with their performance. Laboratory experts indicate they would be satisfied with an autoverification system that autoverified 75% of the results, leaving the rest for

laboratory experts to review. At that level, laboratories with the synthetic error autoverification system can expect approximately a 35% increase in sensitivity over the performance of laboratory experts. The synthetic error autoverification system has shown that it is statistically superior to laboratory experts in detecting errors in the clinical laboratory.

Chapter 8: Summary and Conclusions

In this final chapter, we summarize the results presented herein showing the utility of a novel approach to autoverification in the clinical laboratory, describe the contribution of this dissertation to the Biomedical and Health Informatics domain, discuss limitations in our research, and highlight future work to address those limitations and further expand this work.

8.1. Summary of Results

This dissertation started with the simple question of whether we could develop an autoverification system that better detecte errors in the clinical laboratory than currently possible. Creating an effective autoverification system for a clinical laboratory is a very challenging task. Rule-based systems, virtually the only inference engine used in the current generation of autoverification systems, are not capable of handling the inherent uncertainty in the domain. This limitation relegates rule-base autoverification systems to doing what they are capable of doing: alerting when pre-defined criteria, such as an extreme value, are exceeded. Such a system performs poorly when attempting to identify errors, especially false-normal results. Bayesian networks, however, are a powerful tool for making bi-directional inferences under uncertainty and are more appropriate for the domain.

In order to use a Bayesian network as an autoverification system, the structure of the network is first identified using an algorithm or expert specification and then a

training dataset is used to learn how to identify potentially erroneous results. The clinical laboratory, unfortunately, lacks a gold standard that can be used to create training datasets. Furthermore, should a gold standard exist, the performance of the Bayesian network would be very poor due to the extreme class imbalance and small disjuncts that arise from the complex multifactorial etiology of laboratory errors.

A synthetic error generation method, used to create a synthetic training dataset, yields an autoverification system statistically superior to minority-class over-sampling and majority-class under-sampling under the broad range of conditions likely to be present in the clinical laboratory. Unlike standard approaches that require an expensive and time-consuming expert annotation process to create training datasets, the synthetic error generation method uses datasets of results that were reviewed normally. By creating synthetic datasets, the synthetic error generation process creates customized datasets, which maximize the Bayesian network's performance in detecting errors.

Errors are very rare in the clinical laboratory and in this domain, the synthetic error autoverification excels. Compared to minority-class over-sampling, the only standard class imbalance method that approached the performance levels of the synthetic error method, a Bayesian network trained using the synthetic error generation method is superior. The synthetic error generation method enables the better training of an autoverification system, which results in better performance when detecting errors.

Laboratories that use an autoverification system are satisfied with them, but fewer than half use one. Laboratory experts want an autoverification system to

autoverify approximately 75% of the results, leaving the rest to be reviewed by experts. They also want to know why a result was not autoverified and the possible causes of the perceived error. However, the current generation of autoverification systems is not capable of hypothesizing as to the source of the error, whereas Bayesian networks can. When asked to review laboratory results mimicking a pre-diabetic population, the synthetic error autoverification system significantly outperformed laboratory experts. At their desired specificity of 75%, laboratory experts completing one of two comparisons achieved an average sensitivity of 64% while the synthetic autoverification system achieved an average sensitivity of 92%. Using the synthetic error generation method to create training datasets for a Bayesian network produces a superior autoverification system compared to systems trained using standard methods and to laboratory experts.

8.2. Contributions

In this dissertation, we describe a novel approach to clinical laboratory autoverification systems that utilizes a synthetic error generation method to create training datasets, which are then used to train a Bayesian network to detect errors in clinical laboratory results. Our novel approach produced an autoverification system that is superior to laboratory experts and standard class-imbalance methods. Thus, this dissertation contributes to the biomedical and health informatics domain by:

- Demonstrating the performance characteristics of the synthetic error generation autoverification system to show that we can better train our system.

- Demonstrating the superiority of the synthetic error generation autoverification system compared to the standard methods of addressing the class imbalance problem in the clinical laboratory domain to show that better training results in better performance.
- Demonstrating the superiority of the synthetic error generation autoverification system compared to laboratory experts.

An implicit question in our research was whether a training dataset needed to represent accurately the target testing dataset and, if not, how non-representative training datasets affected performance. In Chapter 5, we demonstrated that superior performance is obtained when the training dataset is very unrealistic. When the minority-class probability is very low in the testing dataset, maximum performance is obtained by training with a minority-class probability of 50%. Furthermore, for a given minority-class probability in the training dataset, performance is not affected by the minority-class probability in the testing dataset. The synthetic error generation system will tend to over-fit the magnitude of the error in the training dataset, but a large training minority-class probability minimizes this affect. By demonstrating the performance characteristics of the synthetic error generation system, we have contributed to the biomedical and health informatics domain by adding a new tool for researchers to use when addressing supervised learning under an extreme class imbalance.

The class imbalance problem is not unique to the clinical laboratory domain and experts have developed established methods to address this problem. However, it was not known if these standard methods were appropriate or best in the clinical laboratory domain. In Chapter 6, we demonstrated that minority-class over-sampling is the best standard method for the clinical laboratory domain, but that the synthetic error generation method is even better. In addition to the superior performance of the synthetic error generation method, it does not require a costly expert-annotated database, which serves to expand greatly the usefulness of the method. We contribute to the biomedical and health informatics domain by showing that clinical laboratories, with precious little resources and ever-tightening budgets, can more readily implement autoverification specific to their client population with synthetic error generation.

Prior to their use, the laboratory director must evaluate and authorize the autoverification system. In Chapter 7, we evaluated the performance of the synthetic error autoverification against laboratory experts in the detection of errors in a dataset mimicking a pre-diabetic population. The results of the comparison were clear: in such a dataset, a synthetic error autoverification system significantly outperforms laboratory experts. We contribute to the biomedical and health informatics domain by demonstrating the effectiveness of autoverification systems against laboratory experts.

8.3. Limitations

The method of using a synthetic error generation system to create training datasets for a conditional Gaussian Bayesian network has some clear limitations, primarily due to the choice of a conditional Gaussian Bayesian network for predicting the expected value. These limitations may be inherent limitations due to the way data are modeled in a Gaussian Bayesian network, or they may be due to the way Gaussian Bayesian networks are developed. In addition, the comparison with laboratory experts has limitations due to its necessarily limited depth and breadth.

8.3.1. Model Limitations

First, we must assume that laboratory data can be modeled by a multivariate Gaussian distribution, either directly or after some transformation of the data. For example, triglyceride has a skewed distribution that, in part, can be corrected by taking the natural log of the data. Still, highly skewed data such as autoantibody indices may not be sufficiently modeled by a Gaussian representation. Conditional Gaussian Bayesian networks require that each node be a linear weighted sum of its parents where the weights are constant. However, the relationship between glucose and glycosylated hemoglobin is known to vary depending on treatment and stage of diabetes (Kilpatrick, Rigby et al. 2007). It is not known if a single Bayesian network is able to effectively model the relationship over the entire range of treatments and stages of diabetes, or if a collection of models is needed.

The structure of the Bayesian network must be identified either by using an algorithm or through expert specification. Either way, the relationships entailed by the Bayesian network will be based on statistical co-variation, rather than causation. As such, there is a risk that both the structure and parameters of the Bayesian network are specific to the data source utilized in its construction. The synthetic error autoverification's performance on a disparate dataset may result in reduced performance.

8.3.2. Expert Comparison Limitations

The survey response rate of 10% is too low for statistical analysis, so opinions expressed by the respondents may not be generalizable to the population in general. However, a key result of the survey was an estimate of the desired specificity, 75%, and this value is comparable to other published values. In Chapter 7, we compared the performance of the synthetic error autoverification against laboratory experts. Due to the anonymous nature of this comparison, we do not know the qualifications of each expert and if their qualifications extended to the review of glycosylated hemoglobin data in a pre-diabetic population. In addition, a real database could not be used for the comparison so a database was created that mimicked the target population. It is possible that an expert unqualified to perform the comparison did so or that the artificial dataset was unrealistic, reducing the average performance of the experts. In addition, due to time constraints, we were not able to evaluate the synthetic error autoverification system against laboratory experts for analyses other than glycosylated hemoglobin in a

single population nor were we able to use a realistic testing dataset containing approximately one percent errors. The low response rate, however, does not affect the conclusion that the synthetic error autoverification system is statistically superior to laboratory experts since the low response rate is accounted for in the larger standard error estimate.

8.4. Future work

The results presented in this dissertation suggest that the synthetic error autoverification system is an effective tool for the clinical laboratory to identify errors. To enhance further this work we will continue the development and evaluation of the system. The system can be thought of as two parts: 1) The first part predicts a value and compares that predicted value to the measured value to estimate the error; 2) The second part uses the error estimates to hypothesize as to the source of error. The prediction model used in this system is not required to be a Bayesian network, but we have not evaluated different models. In this dissertation, we did not hypothesize as to the source of error. By modeling a variety of errors and their affects on correlated analyses, we can train the system to detect a wide range of errors. Preliminary work with the 2004 NHanes dataset has already demonstrated the significant potential of this approach.

Comparing an autoverification system against laboratory experts is very time consuming. In a realistic comparison, experts would need to review thousands of results. However, some researchers have culled annotated datasets containing laboratory errors. While laboratory experts are not perfect in their identification of

errors and, therefore, these datasets would contain mis-labeled results, those datasets may still be useful in demonstrating the performance of the synthetic error autoverification system.

8.5. Concluding Remarks

Estimates of the harm caused by laboratory errors vary significantly depending on the true error rate, the observed error rate, definition of harm, and percentage of errors causing harm. Laboratory errors may harm millions of patients each year or they may only harm thousands. Either way, laboratory experts spend countless hours reviewing billions of laboratory results each year in the search for these errors. Autoverification systems can save countless hours and be more accurate than laboratory experts, but the current generation of rule-based systems is not appropriate for the clinical laboratory domain with its inherent uncertainty. This research has demonstrated that a novel approach using a synthetic error generation system to create training datasets for a conditional Gaussian Bayesian network produces an autoverification system that is superior to ones trained using standard methods and to laboratory experts.

Bibliography

Acklam, P. J. (2005). "An algorithm for computing the inverse normal cumulative distribution function." Retrieved November 19, 2006, from http://home.online.no/~pjacklam/notes/invnorm/index.html.

American Association for Clinical Chemistry (2007). Third Middleware Initiative Meeting. Enterprise Analysis Corporation. Las Vegas.

Astion, M. (2006). Putting Power into Patient Safety Interventions in the Clinical Laboratory. Canadian Society of Clinical Chemists. Victoria, BC.

Astion, M., Krueger-Nielsen S, et al. (2004). "Errors and patient outcomes associated with problems in stat chemistry testing." Clin Chem **50**(Supplement 6): A114-115.

Bach, F. R., D. Heckerman, et al. (2004). On the Path to an Ideal ROC Curve: Considering Cost Asymmetry in Learning Classifiers, Microsoft Research.

Batista, G. E., R. C. Prati, et al. (2004). "A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data." SIGKDD Explorations **6**(1): 20-29.

Black, R., P. Woolman, et al. (2004). "Variation in the transcription of laboratory data in an intensive care unit." Anaesthesia **59**(8): 767-9.

Bolann, B. J. and B. Stolsnes (1999). "[Analytical uncertainty--how wrong can a laboratory result be?]." Tidsskr Nor Laegeforen **119**(30): 4472-5.

Bonini, P., M. Plebani, et al. (2002). "Errors in laboratory medicine." Clin Chem **48**(5): 691-8.

Boran, G., P. Given, et al. (1996). "Patient result validation services." Comput Methods Programs Biomed **50**(2): 161-8.

Bøttcher, S. G. and C. Dethlefsen (2003). DEAL: A Package for Learning Bayesian Networks.

Boulle, M. (2004). "Khiops: A Statistical Discretization Method of Continuous Attributes." Machine Learning **55**(1): 53.

Brown, L., I. Tsamardinos, et al. (2004). A Novel Algorithm for Scalable and Accurate Bayesian Network Learning. MedInfo. M. F. e. al. San Francisco, CA.

Bush, G. W. (2007). State of the Union Address.

California Assembly (2006). Business and Professions Code 1209.5.

Centers for Disease Control and Prevention (2004). Cholesterol Reference Method Laboratory Network - Total Cholesterol Recertification Protocol for Manufacturers.

Centers for Disease Control and Prevention (CDC) (2004). National Health and Nutrition Examination Survey Data. Hyattsville, MD, U.S. Department of Health and Human Services, Centers for Disease Control and Prevention.

Chawla, N. V., K. W. Bowyer, et al. (2002). "SMOTE: Synthetic Minority Over-sampling TEchnique." Journal of Artificial Intelligence Research **16**: 341-378.

Chawla, N. V., N. Japkowicz, et al. (2004). "Editorial: special issue on learning from imbalanced data sets." SIGKDD Explor. Newsl. **6**(1): 1-6.

Chickering, D., D. Geiger, et al. (1994). Learning Bayesian Networks is NP-Hard. Redmond, Microsoft Corporation.

Chickering, D. M. (2002). The WinMine Toolkit. Redmond, WA, Microsoft.

Commission on Laboratory Accreditation (2006). Laboratory General Checklist. C. o. A. Pathologists.

Cooper, G. F., C. F. Aliferis, et al. (1997). "An evaluation of machine-learning methods for predicting pneumonia mortality." Artif Intell Med **9**(2): 107-38.

Cowell, R. G., A. P. Dawid, et al. (1999). Probabilistic networks and expert systems. New York, Springer.

Crolla, L. J. and J. O. Westgard (2003). "Evaluation of rule-based autoverification protocols." Clin Leadersh Manag Rev **17**(5): 268-72.

Diabetes Prevention Program (2001). Diabetes Prevention Program Protocol (May 18, 2001)**:** 200.

Doshi, P., L. Greenwald, et al. (2002). Towards Effective Structure Learning for Large Bayesian Networks. AAAI Workshop on Probabilistic Approaches in Search, Edmonton, Canada.

Dougherty, J., R. Kohavi, et al. (1995). Supervised and Unsupervised Discretization of Continuous Features. Machine Learning: Proceeding of the Twelfth International Conference, San Francisco, CA.

Dzik, W. H., M. F. Murphy, et al. (2003). "An international study of the performance of sample collection from patients." Vox Sang **85**(1): 40-7.

Faraggi, D. and B. Reiser (2002). "Estimation of the area under the ROC curve." Stat Med **21**(20): 3093-106.

Flach, P. (2004). The Many Faces of ROC Analysis in Machine Learning. The Twenty-First International Conference on Machine Learning. Banff, Canada.

Fraser, C. G., H. P. Stevenson, et al. (2002). "Biological variation data are necessary prerequisites for objective autoverification of clinical laboratory data." Accreditation and Quality Assurance: Journal for Quality, Comparability and Reliability in Chemical Measurement **Volume 7**(11): 455-460.

Friedman, M. (1937). "The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance." Journal of the American Statistical Association **32**(200): 675-701.

Friedman, N. (1998). The Bayesian Structural EM Algorithm. Fourteenth Conf. on Uncertainty in Artificial Intelligence.

Friedman, N. and M. Goldszmidt, Eds. (1996). Discretizing Continuous Attributes While Learning Bayesian Networks. Machine Learning: Proceedings of the Thirteenth International Conference, Morgan Kaufmann Publishers.

Friedman, N., I. Nachman, et al. (1999). Learning bayesian network structures from massive datasets: The sparse candidate algorithm. Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, San Francisco, CA, Morgan Kaufmann.

Geiger, D. and D. Heckerman (1994). Learning Gaussian Networks. Redmond, WA, Microsoft.

Geiger, D. and D. Heckerman (1997). "A Characterization of the Dirichlet Distribution through Global and Local Parameter Independence." The Annals of Statistics **25**(3): 1344-1369.

Glick, M. R., K. W. Ryder, et al. (1986). "Graphical comparisons of interferences in clinical chemistry instrumentation." Clin Chem **32**(3): 470-5.

Goldenberg, A. and A. Goldenberg (2004). Tractable Learning of Large Bayes Net Structures from Sparse Data. The Twenty-First International Conference on Machine Learning. Banff, Alberta, Canada

Goldschmidt, H. and R. Lent (1995). "Gross errors and work flow analysis in the clinical laboratory." Klin Biochem Metab **3**: 131-40.

Guo, H. and H. L. Viktor (2004). "Learning from imbalanced data sets with boosting and data generation: the DataBoost-IM approach." SIGKDD Explorations **6**(1): 30-39.

Hanley, J. A. and B. J. McNeil (1982). "The meaning and use of the area under a receiver operating characteristic (ROC) curve." Radiology **143**(1): 29-36.

Hanley, J. A. and B. J. McNeil (1983). "A method of comparing the areas under receiver operating characteristic curves derived from the same cases." Radiology **148**(3): 839-43.

Heckerman, D., D. Geiger, et al. (1995). "Learning Bayesian Networks: The Combination of Knowledge and Statistical Data." Machine Learning **20**(3): 197.

Hippisley-Cox, J., R. Cater, et al. (2003). "Cross sectional survey of effectiveness of lipid lowering drugs in reducing serum cholesterol concentration in patients in 17 general practices." Bmj **326**(7391): 689.

Hoelzel, W., C. Weykamp, et al. (2004). IFCC Reference System for Measurement of Hemoglobin A1c in Human Blood and the National Standardization Schemes in the United States, Japan, and Sweden: A Method-Comparison Study. **50:** 166-174.

Hoeting, J., D. Madigan, et al. (1998). Bayesian Model Averaging. Seattle, University of Washington.

Hollensead, S. C., W. B. Lockwood, et al. (2004). "Errors in pathology and laboratory medicine: consequences and prevention." J Surg Oncol **88**(3): 161-81.

Hripcsak, G., S. Bakken, et al. (2003). "Mining complex clinical data for patient safety research: a framework for event discovery." Journal of Biomedical Informatics **36**(1-2): 120.

Hripcsak, G. and D. F. Heitjan (2002). "Measuring agreement in medical informatics reliability studies." J Biomed Inform **35**(2): 99-110.

Institute of Medicine (1999). To Err is Human : Building a Safer Health System. Washington, D.C., National Academy Press.

Ito, C., R. Maeda, et al. (2000). "Correlation among fasting plasma glucose, two-hour plasma glucose levels in OGTT and HbA1c." Diabetes Research and Clinical Practice **50**(3): 225-230.

Japkowicz, N. (1999). Concept-Learning in the Absence of Counter-Examples: An Autoassociation-Based Approach to Classification, Rutgers University. **PhD thesis**.

Japkowicz, N. (2001). Concept-Learning in the Presence of Between-Class and Within-Class Imbalances. Proceedings of the Fourteenth Conference of the Canadian Society for Computational Studies of Intelligence (AI'2001).

Japkowicz, N. (2003). Class Imbalances: Are we Focusing on the Right Issue? Workshop on Learning from Imbalanced Datasets II. Washington DC, ICML.

Japkowicz, N. and S. Stephen (2002). "The Class Imbalance Problem: A Systematic Study." Intelligent Data Analysis **6**(5): 429-450.

Jay, D. W. and D. Provasek (1993). "Characterization and mathematical correction of hemolysis interference in selected Hitachi 717 assays." Clin Chem **39**(9): 1804-10.

Jensen, F. V. (2001). Bayesian networks and decision graphs. New York, Springer.

Jo, T. and N. Japkowicz (2004). "Class Imbalances versus Small Disjuncts." SIGKDD Explorations **6**(1): 40-49.

Jordan, M. I. (1999). Learning in graphical models. Cambridge, Mass., MIT Press.

Kazmierczak, S. C. (2003). "Laboratory quality control: using patient data to assess analytical performance." Clin Chem Lab Med **41**(5): 617-27.

Khoury, M., L. Burnett, et al. (1996). "Error rates in Australian chemical pathology laboratories." Med J Aust **165**(3): 128-30.

Kilpatrick, E. S., A. S. Rigby, et al. (2007). Variability in the Relationship between Mean Plasma Glucose and HbA1c: Implications for the Assessment of Glycemic Control. **53:** 897-901.

Kohavi, R. (1995). A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. International Joint Conference on Artificial Intelligence.

Kroll, M. H. and R. J. Elin (1994). "Interference with clinical laboratory analyses." Clin Chem **40**(11 Pt 1): 1996-2005.

Laboratory Corporation of America (2007). 2006 Annual Report.

Ladenson, J. H. (1975). "Patients as their own controls: use of the computer to identify "laboratory error"." Clin Chem **21**(11): 1648-53.

Landro, L. (2006). The Informed Patient: Hospitals Move to Cut Dangerous Lab Errors. The Wall Street Journal. New York**:** D1.

Lauritzen, S. L. and F. Jensen (2001). "Stable local computation with conditional Gaussian distributions." Statistics and Computing **11**(2): 191-203.

Lauritzen, S. L. and D. J. Spiegelhalter (1988). "Local Computations with Probabilities on Graphical Structures and Their Application to Expert Systems." Journal of the Royal Statistical Society. Series B (Methodological) **50**(2): 157-224.

Lemmens, A. and C. Croux (2006). "Bagging and boosting classification trees to predict churn." Journal of Marketing Research **43**(2): 276-286.

Lerner, U., E. Segal, et al. (2001). Exact Inference in Networks with Discrete Children of Continuous Parents Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc.

Mani, S. and G. F. Cooper (2004). Causal Discovery Using A Bayesian Local Causal Discovery Algorithm. MedInfo. San Francisco, CA, Fieschi, M. et al.

Marchand, M., J. Guibourdenche, et al. (1997). "Real time validation of paediatric biochemical reports using the Valab-Biochem system." Ann Clin Biochem **34 (Pt 4)**: 389-95.

Marcovina, S., R. R. Bowsher, et al. (2007). Standardization of Insulin Immunoassays: Report of the American Diabetes Association Workgroup. **53:** 711-716.

Mars, N. J. and P. L. Miller (1987). "Knowledge acquisition and verification tools for medical expert systems." Med Decis Making **7**(1): 6-11.

McCarthy, K., B. Zabar, et al. (2005). Does cost-sensitive learning beat sampling for classifying rare classes? Proceedings of the 1st International Workshop on Utility-based Data Mining. Chicago, Illinois ACM Press.

McDowell, J. (2007). "Managed Care Contract Shakeups for Lab Testing." Clinical Laboratory News **33**(5).

McNair, P., J. Brender, et al. (1998). "Computer-aided test selection and result validation-opportunities and pitfalls." Clin Chim Acta **278**(2): 243-55.

Menendez, G. R. and C. O. Cabrera (2000). "Selection of an optimal combination of decision criteria for the internal quality control in an automatic analyzer." Clin Chim Acta **292**(1-2): 139-47.

Mitchell, T. M. (1997). Machine Learning. New York, McGraw-Hill.

Monard, M. C. and G. E. A. P. A. Batista (2002). Learning with Skewed Class Distributions. Advances in Logic, Artificial Intelligence and Robotics. J. M. Abe and J. I. d. S. Filho, IOS Press**:** 173-180.

Murff, H. J., V. L. Patel, et al. (2003). "Detecting adverse events for patient safety research: a review of current methodologies." Journal of Biomedical Informatics **36**(1-2): 131.

Nachman, I. (2004). Probabilistic Modeling of Gene Regulatory Networks from Data, Hebrew University. **Doctor of Philosophy:** 146.

Narayanan, S. (2000). "The preanalytic phase. An important component of laboratory medicine." Am J Clin Pathol **113**(3): 429-52.

National Center for Health Statistics. (2004). "Documentation, Codebook, and Frequencies MEC Laboratory Component: Biochemistry Profile." Retrieved May 25, 2007, from http://www.cdc.gov/nchs/data/nhanes/nhanes_03_04/l43_c.pdf.

National Center for Health Statistics. (2004). "Laboratory Procedure Manual: Aspartate Aminotransferase (AST)." Retrieved May 25, 2007, from http://www.cdc.gov/nchs/data/nhanes/nhanes_03_04/l40_c_met_aspartate_aminotransferase.pdf.

Neapolitan, R. E. (2004). Learning Bayesian networks. Upper Saddle River, N.J., Pearson Prentice Hall.

Norsys Software Corporation (2006). Netica.

Nutting, P. A., D. S. Main, et al. (1996). "Toward optimal laboratory use. Problems in laboratory testing in primary care." Jama **275**(8): 635-9.

Obuchowski, N. A. (2003). Receiver Operating Characteristic Curves and Their Use in Radiology. **229:** 3-8.

Ollerton, R. L., R. Playle, et al. (1999). "Day-to-day variability of fasting plasma glucose in newly diagnosed type 2 diabetic subjects." Diabetes Care **22**(3): 394-8.

Oosterhuis, W. P., H. J. Ulenkate, et al. (2000). "Evaluation of LabRespond, a new automated validation system for clinical laboratory test results." Clin Chem **46**(11): 1811-7.

Pearl, J. (1988). Probabilistic reasoning in intelligent systems : networks of plausible inference. San Mateo, Calif., Morgan Kaufmann Publishers.

Pearl, J. (2000). Causality : models, reasoning, and inference. Cambridge, U.K. ; New York, Cambridge University Press.

Pearl, J. and S. Russell (2000). Bayesian Networks.

Pearlman, E. S., L. Bilello, et al. (2002). "Implications of autoverification for the clinical laboratory." Clin Leadersh Manag Rev **16**(4): 237-9.

Pednault, E. P. D., B. K. Rosen, et al. (2000). Handling Imbalanced Data Sets in Insurance Risk Modeling. New York, IBM Research.

Plebani, M. and P. Carraro (1997). "Mistakes in a stat laboratory: types and frequency." Clin Chem **43**(8 Pt 1): 1348-51.

Pollack, A. (2001). A Positive Culture For Making Profits; Buoyed by Mergers, Medical Labs Await Era of Gene Testing. The New York Times. New York.

Poon, E. G., S. J. Wang, et al. (2003). "Design and implementation of a comprehensive outpatient Results Manager." Journal of Biomedical Informatics **36**(1-2): 80.

Prosser, L. A., A. A. Stinnett, et al. (2000). "Cost-Effectiveness of Cholesterol-Lowering Therapies according to Selected Patient Characteristics." Ann Intern Med **132**(10): 769-779.

Prost, L. and E. Rogari (2002). "How autoverification through the expert system VALAB can make your laboratory more efficient." Accreditation and Quality Assurance: Journal for Quality, Comparability and Reliability in Chemical Measurement **7**(11): 480-487.

Quality Interagency Coordination Task Force. (2000). "Doing what counts for patient safety federal actions to reduce medical errors and their impact ; report of the Quality Interagency Coordination Task Force (QuIC) to the President, February 2000." from http://www.quic.gov/report/.

Quinlan, J. R. (1993). C4.5 : programs for machine learning. San Mateo, Calif., Morgan Kaufmann Publishers.

Regeniter, A., W. H. Siede, et al. (1996). "Computer assisted interpretation of laboratory test data with 'MDI-LabLink'." Clin Chim Acta **248**(1): 107-18.

Ricos, C., V. Alvarez, et al. (1999). "Current databases on biological variation: pros, cons and progress." Scand J Clin Lab Invest **59**(7): 491-500.

Rohlfing, C. L., H.-M. Wiedmeyer, et al. (2002). "Defining the Relationship Between Plasma Glucose and HbA1c: Analysis of glucose profiles and HbA1c in the Diabetes Control and Complications Trial." Diabetes Care **25**(2): 275-278.

Rosner, B. (2000). Fundamentals of biostatistics. Pacific Grove, CA, Duxbury.

Rumenjak, V., S. Milardovic, et al. (2003). "The study of some possible measurement errors in clinical blood electrolyte potentiometric (ISE) analysers." Clin Chim Acta **335**(1-2): 75-81.

Shahangian, S., R. D. Cohn, et al. (1999). "System to monitor a portion of the total testing process in medical clinics and laboratories: evaluation of a split-specimen design." Clin Chem **45**(2): 269-80.

Shojania, K. G., E. C. Burton, et al. (2003). "Changes in rates of autopsy-detected diagnostic errors over time: a systematic review." Jama **289**(21): 2849-56.

Shortliffe, E. H. (1991). "Medical informatics and clinical decision making: the science and the pragmatics." Med Decis Making **11**(4 Suppl): S2-14.

Silverstein, M. D. (2003). An Approach to Medical Errors and Patient Safety in Laboratory Services. Quality Institute Meeting: Making the Laboratory a Partner in Patient Safety. Atlanta.

Snyder, J. A., M. W. Rogers, et al. (2004). "The impact of hemolysis on Ortho-Clinical Diagnostic's ECi and Roche's elecsys immunoassay systems." Clin Chim Acta **348**(1-2): 181-7.

Spirtes, P., C. N. Glymour, et al. (2000). Causation, prediction, and search. Cambridge, Mass., MIT Press.

Stroobants, A. K., H. M. Goldschmidt, et al. (2003). "Error budget calculations in laboratory medicine: linking the concepts of biological variation and allowable medical errors." Clin Chim Acta **333**(2): 169-76.

Tahara, Y. and K. Shima (1995). "Kinetics of HbA1c, glycated albumin, and fructosamine and analysis of their weight functions against preceding plasma glucose level." Diabetes Care **18**(4): 440-7.

The Advisory Board Company (1999). Prescription for change: toward a higher standard in medication management. Washington, DC, The Advisory Board Company.

The Diabetes Prevention Program Research Group (2000). "The Diabetes Prevention Program: baseline characteristics of the randomized cohort." Diabetes Care **23**(11): 1619-1629.

The R Project for Statistical Computing (2007). R.

Torke, N., L. Boral, et al. (2005). "Process Improvement and Operational Efficiency through Test Result Autoverification." Clin Chem **51**(12): 2406-2408.

Twomey, P. J., A. S. Wierzbicki, et al. (2003). "Issues to consider when attempting to achieve the American Diabetes Association clinical quality requirement for haemoglobin A1c." Curr Med Res Opin **19**(8): 719-23.

University of Utah. (2007). "Blood Collection: Routine Venipuncture and Specimen Handling." Phlebotomy Retrieved April 15, 2007, from http://library.med.utah.edu/WebPath/TUTORIAL/PHLEB/PHLEB.html.

Valdiguie, P. M., E. Rogari, et al. (1996). "The performance of the knowledge-based system VALAB revisited: an evaluation after five years." Eur J Clin Chem Clin Biochem **34**(4): 371-6.

Valdiguie, P. M., E. Rogari, et al. (1992). "VALAB: expert system for validation of biochemical data." Clin Chem **38**(1): 83-7.

Wang, B. X. and N. Japkowicz (2004). Dealing with Class Imbalances with Synthetic Examples. IRIS Machine Learning Workshop, Ottawa, Canada.

Wang, S. and V. Ho (2004). "Corrections of clinical chemistry test results in a laboratory information system." Arch Pathol Lab Med **128**(8): 890-2.

Weiss, G. and F. Provost (2001). The effect of class distribution on classifier learning. Technical Report ML-TR 43, Department of Computer Science, Rutgers University.

Weiss, G. M. (1995). "Learning with Rare Cases and Small Disjuncts." International Conference on Machine Learning: 558-565.

Westgard, J. O. (2004). "Taking Autoverification to the Next Level – Is that up or down?" Retrieved May 11, 2007, from http://www.westgard.com/essay57.htm.

Westgard, J. O. and T. Darcy (2004). "The truth about quality: medical usefulness and analytical reliability of laboratory tests." Clin Chim Acta **346**(1): 3-11.

Witte, D. L., S. A. VanNess, et al. (1997). "Errors, mistakes, blunders, outliers, or unacceptable results: how many?" Clin Chem **43**(8 Pt 1): 1352-6.

Wiwanitkit, V. (2001). "Types and frequency of preanalytical mistakes in the first Thai ISO 9002:1994 certified clinical laboratory, a 6 - month monitoring." BMC Clin Pathol **1**(1): 5.

Wright, G. and P. Ayton (1994). Subjective probability. Chichester ; New York, Wiley.

Yu, J., V. A. Smith, et al. (2004). "Advances to Bayesian network inference for generating causal networks from observational biological data." Bioinformatics **20**(18): 3594-603.

Appendix I – Power Calculations

Power calculations, detailed below, are from Hanley and McNeil (1983; 1982). All comparisons of average area under the receiver operating characteristic curves are pair-wise. The following equation is used to estimate the standard error in the area under the receiver operating characteristic curve for a single simulation.

$$SE(\theta) = \sqrt{\frac{\theta(1-\theta) + (n_A - 1)(Q_1 - \theta^2) + (n_N - 1)(Q_2 - \theta^2)}{n_A n_N}}$$

Where:

θ = the area under the receiver operating characteristic curve for one simulation

n_A = the number of minority-class elements in the simulation

n_N = the number of majority-class elements in the simulation

$$Q_1 = \frac{\theta}{2-\theta}$$

$$Q_2 = \frac{2\theta^2}{1+\theta}$$

We make the following assumptions:

- The area under the receiver operating characteristic curve is 0.60.
- The minority-class probability is 1.0%.

- The sample size is 10,000.

With these assumptions and using the equation above, the standard error for a single simulation is 0.0030. Each simulation is repeated 100 times so the standard error in the mean area under the receiver operating characteristic curve is 0.0003. We then use this estimate for the standard error to derive the minimum difference that we have 80% power to detect. Hanley and McNeil in (1983) provide the following equation:

$$z = \frac{A_1 - A_2}{\sqrt{SE_1^2 + SE_2^2 - 2rSE_1SE_2}}$$

Where:

A_x is the area under the receiver operating characteristic curve for x.

SE_x is the standard error estimate for x.

r is the correlation coefficient between areas.

z is the critical ratio

We make the following assumptions:

- An 80% power to detect an effect.

- A Type I error rate of 5%. For a two-sided comparison, this corresponds to a value of Z equal to 1.96. For a one-sided comparison, this corresponds to a value of 1.65.
- The correlation coefficient between areas, r, is conservatively estimated at 0.0.
- The standard errors are equal.

With these assumptions, we will have an 80% chance of detecting a difference of 0.008 between the two areas under the receiver operating characteristic curves. Our power to detect an effect increases as the areas under the receiver operating characteristic curves increase and as the minority-class probability increases.

Appendix II - Survey Questions

1. How would you describe the size of your laboratory?
    a. Small
    b. Medium
    c. Large
2. What is your opinion as to the percentage of overall errors in your laboratory reports?
    a. < 0.01%
    b. 0.01 – 0.5%
    c. 0.5% - 1.5%
    d. > 1.5%
3. What is your opinion as to the percentage of all errors from above that result in some harm to the patient? Harm is defined as including delayed treatment or additional testing.
    a. < 1%
    b. 5%
    c. 10%
    d. > 10%
4. Do you generally have access to a patient's previous laboratory results performed in your laboratory for use in reviewing current results?
    a. Yes
    b. No
5. Do you generally have electronic access to a patient's medical records for use in reviewing results?
    a. Yes
    b. No
6. Do you use an autoverification system to review results?
    a. Yes
    b. No (go to Question #10)

7. What is the approximate percentage of results that are automatically released?
    a. < 50%
    b. 67%
    c. 75%
    d. > 95%
8. What is your satisfaction level with your autoverification system?
    a. Very dissatisfied
    b. Dissatisfied
    c. Satisfied
    d. Very satisfied
9. Please explain.
    a. (Open text)
10. For you to be satisfied with an autoverification system, what percentage of results should be automatically released?
    a. 50%
    b. 67%
    c. 75%
    d. >95%
11. How important is it that an autoverification system be able to explain why a result was flagged? For example: “Fasting glucose > 150mg/dl and HbA1c < 5.0%”
    a. Very important
    b. Somewhat important
    c. Somewhat unimportant
    d. Very unimportant

12. How important is it that an autoverification system rank possible sources of error in a flagged result? For example: “Patient not fasting (30%), samples switched (20%),…”
    a. Very important
    b. Somewhat important
    c. Somewhat unimportant
    d. Very unimportant
13. For a test like cholesterol, what magnitude of error would you want detected at least 50% of the time?
    a. 10%
    b. 20%
    c. 30%
    d. 40%
    e. >50%
14. For a test like cholesterol, what magnitude of error would you want detected at least 95% of the time?
    a. 10%
    b. 20%
    c. 30%
    d. 40%
    e. >50%
15. For a test like triglyceride, what magnitude of error would you want detected at least 50% of the time?
    a. 10%
    b. 20%
    c. 30%
    d. 40%
    e. >50%

16. For a test like triglyceride, what magnitude of error would you want detected at least 95% of the time?
    a. 10%
    b. 20%
    c. 30%
    d. 40%
    e. >50%

Appendix III – Comparison #1 Questions

Subjects were asked the following questions, with answers in parentheses, and asked to rate their belief using one of: Definitely not an error, probably not an error, neutral, probably an error, definitely an error:

> Consider a pre-diabetic population where the average glucose is 103 mg/dl (standard deviation 11mg/dl) and the average glycosylated hemoglobin (HbA1c) is 5.9 (standard deviation 0.2).
>
> For each of the 30 sets below, what is your belief that the HbA1c value is in error given the fasting glucose value?

Section #1

1. Glucose: 96, HbA1c: 5.4% (error)
2. Glucose: 97, HbA1c: 5.2% (error)
3. Glucose: 105, HbA1c: 6.4% (error)
4. Glucose: 104, HbA1c: 5.8% (OK)
5. Glucose: 103, HbA1c: 6.5% (error)
6. Glucose: 105, HbA1c: 6.0% (OK)
7. Glucose: 96, HbA1c: 5.7% (OK)
8. Glucose: 120, HbA1c: 5.8% (OK)
9. Glucose: 103, HbA1c: 5.9% (OK)
10. Glucose: 100, HbA1c: 5.8% (OK)
11. Glucose: 87, HbA1c: 6.4% (error)
12. Glucose: 110, HbA1c: 6.1% (OK)
13. Glucose: 101, HbA1c: 5.8% (OK)

14. Glucose: 90, HbA1c: 5.4% (OK)
15. Glucose: 87, HbA1c: 6.4% (error)
16. Glucose: 118, HbA1c: 5.4% (error)
17. Glucose: 106, HbA1c: 5.5% (error)
18. Glucose: 107, HbA1c: 6.5% (error)
19. Glucose: 98, HbA1c: 5.8% (OK)
20. Glucose: 111, HbA1c: 5.8% (OK)
21. Glucose: 99, HbA1c: 6.2% (OK)
22. Glucose: 99, HbA1c: 5.6% (OK)
23. Glucose: 70, HbA1c: 6.0% (error)
24. Glucose: 96, HbA1c: 5.6% (OK)
25. Glucose: 97, HbA1c: 6.3% (error)
26. Glucose: 99, HbA1c: 5.8% (OK)
27. Glucose: 97, HbA1c: 5.1% (OK)
28. Glucose: 91, HbA1c: 6.4% (error)
29. Glucose: 77, HbA1c: 5.6% (OK)
30. Glucose: 115, HbA1c: 5.5% (error)

Section #2

1. Glucose: 108, HbA1c: 6.1% (OK)
2. Glucose: 114, HbA1c: 6.3% (OK)
3. Glucose: 97, HbA1c: 5.5% (OK)
4. Glucose: 122, HbA1c: 5.8% (error)
5. Glucose: 99, HbA1c: 6.0% (OK)
6. Glucose: 92, HbA1c: 6.2% (error)
7. Glucose: 88, HbA1c: 5.9% (OK)
8. Glucose: 101, HbA1c: 6.0% (OK)
9. Glucose: 116, HbA1c: 6.4% (OK)

10. Glucose: 98, HbA1c: 5.6% (OK)
11. Glucose: 111, HbA1c: 6.3% (OK)
12. Glucose: 113, HbA1c: 5.5% (error)
13. Glucose: 110, HbA1c: 6.1% (OK)
14. Glucose: 103, HbA1c: 6.3% (OK)
15. Glucose: 109, HbA1c: 5.6% (OK)
16. Glucose: 105, HbA1c: 5.6% (OK)
17. Glucose: 108, HbA1c: 5.9% (OK)
18. Glucose: 97, HbA1c: 6.3% (error)
19. Glucose: 85, HbA1c: 5.7% (OK)
20. Glucose: 115, HbA1c: 5.2% (error)
21. Glucose: 96, HbA1c: 5.7% (OK)
22. Glucose: 108, HbA1c: 5.7% (OK)
23. Glucose: 102, HbA1c: 6.0% (OK)
24. Glucose: 117, HbA1c: 5.2% (error)
25. Glucose: 102, HbA1c: 6.3% (OK)
26. Glucose: 120, HbA1c: 5.8% (error)
27. Glucose: 94, HbA1c: 5.5% (OK)
28. Glucose: 97, HbA1c: 6.1% (OK)
29. Glucose: 96, HbA1c: 6.0% (OK)
30. Glucose: 95, HbA1c: 6.3% (OK)

Appendix IV – Comparison #2 Questions

Subjects were asked the following questions, with answers in parentheses, and asked to rate their belief using one of: Definitely not an error, probably not an error, neutral, probably an error, definitely an error:

> Consider a pre-diabetic population where the average glucose is 103 mg/dl (standard deviation 11mg/dl) and the average glycosylated hemoglobin (HbA1c) is 5.9 (standard deviation 0.2).
>
> For each of the 30 sets below, what is your belief that the HbA1c value is in error given the fasting glucose value?

Section #1

1. Glucose: 96, HbA1c: 6.5% (error)
2. Glucose: 112, HbA1c: 6.0% (OK)
3. Glucose: 91, HbA1c: 5.5% (OK)
4. Glucose: 104, HbA1c: 5.6% (OK)
5. Glucose: 103, HbA1c: 6.1% (OK)
6. Glucose: 96, HbA1c: 5.5% (error)
7. Glucose: 98, HbA1c: 5.2% (error)
8. Glucose: 92, HbA1c: 5.7% (OK)
9. Glucose: 92, HbA1c: 6.2% (OK)
10. Glucose: 102, HbA1c: 5.7% (error)
11. Glucose: 107, HbA1c: 6.1% (OK)
12. Glucose: 111, HbA1c: 5.5% (error)
13. Glucose: 104, HbA1c: 6.1% (OK)

14. Glucose: 112, HbA1c: 5.2% (error)
15. Glucose: 99, HbA1c: 5.9% (OK)
16. Glucose: 92, HbA1c: 6.5% (error)
17. Glucose: 105, HbA1c: 6.0% (OK)
18. Glucose: 117, HbA1c: 5.7% (error)
19. Glucose: 104, HbA1c: 5.6% (OK)
20. Glucose: 104, HbA1c: 6.0% (OK)
21. Glucose: 98, HbA1c: 6.0% (OK)
22. Glucose: 83, HbA1c: 5.5% (OK)
23. Glucose: 106, HbA1c: 6.1% (OK)
24. Glucose: 107, HbA1c: 6.1% (OK)
25. Glucose: 101, HbA1c: 6.2% (OK)
26. Glucose: 98, HbA1c: 6.4% (error)
27. Glucose: 112, HbA1c: 6.0% (OK)
28. Glucose: 117, HbA1c: 5.7% (error)
29. Glucose: 112, HbA1c: 6.1% (OK)
30. Glucose: 113, HbA1c: 6.5% (error)

Section #2

1. Glucose: 90, HbA1c: 6.0% (OK)
2. Glucose: 94, HbA1c: 6.4% (error)
3. Glucose: 86, HbA1c: 5.1% (OK)
4. Glucose: 105, HbA1c: 6.2% (OK)
5. Glucose: 99, HbA1c: 5.2% (OK)
6. Glucose: 101, HbA1c: 5.4% (error)
7. Glucose: 107, HbA1c: 6.4% (error)
8. Glucose: 99, HbA1c: 5.4% (OK)
9. Glucose: 109, HbA1c: 5.8% (OK)

10. Glucose: 82, HbA1c: 5.0% (OK)
11. Glucose: 101, HbA1c: 5.8% (OK)
12. Glucose: 103, HbA1c: 5.6% (OK)
13. Glucose: 110, HbA1c: 6.1% (OK)
14. Glucose: 100, HbA1c: 6.0% (OK)
15. Glucose: 98, HbA1c: 5.9% (OK)
16. Glucose: 100, HbA1c: 6.5% (error)
17. Glucose: 99, HbA1c: 5.9% (OK)
18. Glucose: 95, HbA1c: 6.4% (error)
19. Glucose: 110, HbA1c: 5.4% (error)
20. Glucose: 86, HbA1c: 5.7% (OK)
21. Glucose: 102, HbA1c: 5.7% (error)
22. Glucose: 98, HbA1c: 5.5% (error)
23. Glucose: 105, HbA1c: 6.1% (OK)
24. Glucose: 115, HbA1c: 5.2% (error)
25. Glucose: 115, HbA1c: 6.0% (OK)
26. Glucose: 89, HbA1c: 6.2% (error)
27. Glucose: 104, HbA1c: 5.2% (error)
28. Glucose: 100, HbA1c: 6.0% (error)
29. Glucose: 84, HbA1c: 5.5% (OK)
30. Glucose: 96, HbA1c: 6.2% (error)

VITA

Greg Strylewicz was born in Portland, Oregon. He has lived in many parts of country and has traveled much of the world at a depth of about 400 feet below sea level while in the Navy. He earned his Bachelor of Science degree in Physics from Rose-Hulman Institute of Technology in Terre Haute, Indiana in 1990. He earned his Master of Science in Computer Science from the University of Washington in 2002. In 2007, he earned a Doctor of Philosophy at the University of Washington in Biomedical and Health Informatics.