A MODEL DRIVEN APPROACH TO IMBALANCED DATA LEARNING

YIN HONGLI
B.Comp. (Hons.), NUS

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE

NATIONAL UNIVERSITY OF SINGAPORE

2011

ACKNOWLEDGMENTS

It has never been a solo effort in completing this thesis. I have received tremendous help and support from many people during my PhD study. I would like to take this opportunity to thank the following people who have helped me make this thesis possible, even though many of the names are not possibly listed below:

Firstly, I would like to thank my supervisor Associate Professor Leong Tze-Yun, from School of Computing, National University of Singapore, who has been encouraging, guiding and supporting me all the way from the initial stage to the final stage, and who has never given up on me; without her, this thesis would not be possible.

Professor Lim Tow Keang, from National University Hospital for providing me the Asthma data set, and guiding me in the asthma related research.

Dr. Ivan Ng and Dr. Pang Boon Chuan, both from National Neuron Institute for providing me the mild head injury data set and the severe head injury data set, and whose collaboration and guidance have helped me a lot in the head injury related research.

Dr. Zhu Ai Ling and Dr. Tomi Silander, both from National University of Singapore, and Mr Abdul Latif Bin Mohamed Tahiar's first daughter Mas, who have spent their valuable time in proof reading my thesis.

Associate Professor Poh Kim-Leng and his group from Industrial and System Engineering, National University of Singapore, for their collaboration and guidance in my idea formulation and daily research.

My previous and current colleagues from Medical Computing Lab, Zhu Ai Ling, Li Guo Liang, Rohit Joshi, Chen Qiong Yu, Nguyen Dinh Truong Huy and many others, who have always been helpful in enlightening me and encouraging me during my PhD study.

My special thanks to Zhang Yi who has always encouraged me not to give up, and Zhang Xiu Rong who has constantly given me a lot of support. My dog Tudou who has always been there with me especially during my down time.

Last but not least, I would like to thank my parents who have always been supporting me, especially my father, who has scarified himself for the family and my study, my mother with schizophrenia, who loves me the most, and my grandpas, who passed away, saving all their pennies for my study. I owe my family the most!

TABLE OF CONTENTS

ABSTRACT

Many real life problems, especially in health care and biomedicine, are characterized by imbalanced data. In general, people tend to be more interested in rare events or phenomena. For example, in prognostic predictions, the physicians can take necessary precautions to reduce the risks of the small group of patients who cannot recover in time. Traditional machine learning algorithms often fail to predict the minorities that are of interest. The objective of imbalanced data learning is to correctly identify the rarities without sacrificing prediction of the majorities.

In this thesis, we review the existing approaches to deal with the imbalanced data problem, including data level approaches and algorithm level approaches. Most data sampling approaches are ad-hoc and the exact mechanisms of how they improve prediction performance are not clear. For example, random sampling generates duplicate samples to "fool" the classifier to bias its decision in favor of minorities. Oversampling often leads to data overfitting, and under sampling tends to remove useful information from the original data set. The Synthetic Minority over-Sampling Technique creates synthetic data from the nearest neighbor, but it only makes use of local information and often leads to data over-generalization. On the other hand, most of the algorithmic level approaches have been shown to be equivalent to data sampling approaches. Some other approaches make additional assumptions. For example, a popular approach is cost

sensitive learning which assigns different cost values to different types of misclassifications; but the cost values are usually unknown, and it is hard to discover the right cost value.

We propose a model driven sampling (MDS) approach that can generate new samples based on the global understanding of the entire data set and domain experts' knowledge. This is a first attempt to make use of probabilistic graphical methods to represent the training space and generate synthetic data. Our empirical studies show that in a large class of problems, MDS generally outperforms previous approaches or performs comparably to the best previous approach in the worst case scenario. It performs especially well for extremely imbalanced data without complex connected structures. MDS also works well when domain knowledge is available, as the model created with domain knowledge is better "educated" than that constructed purely from training data and thus, the synthetic data generated are more meaningful. We have also extended MDS to context sensitive MDS and progressive MDS. Context sensitive MDS reduces the problem size by creating more accurate sub models for each individual context. Therefore, the data sampled from context sensitive MDS are more relevant to each context. Instead of assuming the optimal distribution is balanced, progressive MDS iterates over all possible data distributions and selects the best performing data distribution as the optimal distribution. Therefore, progressive MDS improves over MDS by always obtaining the optimal data distribution, as shown by our empirical studies.

LIST OF TABLES

Number *Page*

LIST OF FIGURES

Number *Page*

CHAPTER 1: INTRODUCTION

1. INTRODUCTION

1.1 BACKGROUND

In healthcare, a lot of data have been collected by various institutions and hospitals. These data are valuable resources for outcomes analysis to help doctors to make decisions on disease diagnosis, resource planning, and risk analysis. The definition of outcomes here includes functional outcomes, return to work, quality of life, patient satisfaction, and cost effectiveness. Successful outcomes analysis can help physicians make better decisions about patients' treatments, help in their recovery and cut treatment cost [10, 124].

In health care outcomes analysis, the critical patients normally constitute a very small portion of the whole patient population [137], which leads to the class imbalance problem. For example, this problem was reported in the diagnoses of rare medical conditions such as thyroid diseases [101], asthma control [159], outcomes analysis for severe head injury and mild head injury [158], etc. Besides health care, the class imbalance problem is also widely reported in a lot of other areas with significant environmental, vital or commercial importance [69]. For example, the problem was reported in the detection of oil spills in satellite radar images [83], the detection of

fraudulent telephone calls [46], in-flight helicopter gearbox fault monitoring [67], software defect prediction [162], information retrieval and filtering [86], etc.

Empirical experience shows that traditional data mining algorithms fail to recognize critical patients who are normally the minorities, even though they may have very good prediction accuracy for the majority class. Thus imbalanced data learning – to build a model from the imbalanced data and correctly recognize both majority and minority examples is a very crucial task [87, 159]. Existing approaches mainly include data level approaches [22, 23, 35, 81] and algorithmic level approaches [27, 42, 67, 74, 76, 82, 127]. In this thesis, we mainly focus on data sampling approaches, because empirical studies show that data sampling is more efficient and effective than algorithmic approaches [44, 149]. We have studied the state of the art data sampling approaches – random sampling approach, Synthetic Minority over-Sampling Technique (SMOTE) [23], and progressive sampling [50, 104]. These approaches mainly either duplicate the existing data samples, or create synthetic samples with the nearest neighboring sample. In contrast to the existing approaches, we propose a Model Driven Sampling (MDS) approach to make use of the whole training space and domain knowledge to create synthetic data. To our best knowledge, MDS is the first approach using probabilistic graphical models to model the training space and domain knowledge to generate synthetic data samples.

In this thesis, we compare MDS with existing data sampling approaches on various training data, using different machine learning techniques and evaluation

measures. In particular, Bayesian networks are used to create models in MDS and also used as the data classifier for the evaluation; g-Mean [81] is used as the evaluation metric. MDS is empirically shown to outperform other data sampling approaches in general. It is particularly useful for highly skewed data, and sparse data with domain knowledge. Context sensitive MDS can usually reduce the problem size, and generate more accurate data adapted to each context. Progressive sampling can be combined with MDS to determine the optimal data distribution, instead of using the balanced data distribution that may not be optimal.

1.2 IMBALANCED DATA LEARNING PROBLEM

1.2.1 IMBALANCED DATA DEFINITION

The word "imbalanced" is an antonym for the word "balanced"; Imbalanced dataset refers to the dataset with unbalanced class distribution. Figure 1-1 shows a balanced data distribution – the Singapore population sex distribution with sex as of July 2006 [4]. The number of males and the number of females are roughly equal for each age group. Figure 1-2 illustrates an example of an unbalanced dataset where mild head injury patients greatly outnumber severe head injury patients in a head injury dataset [111].

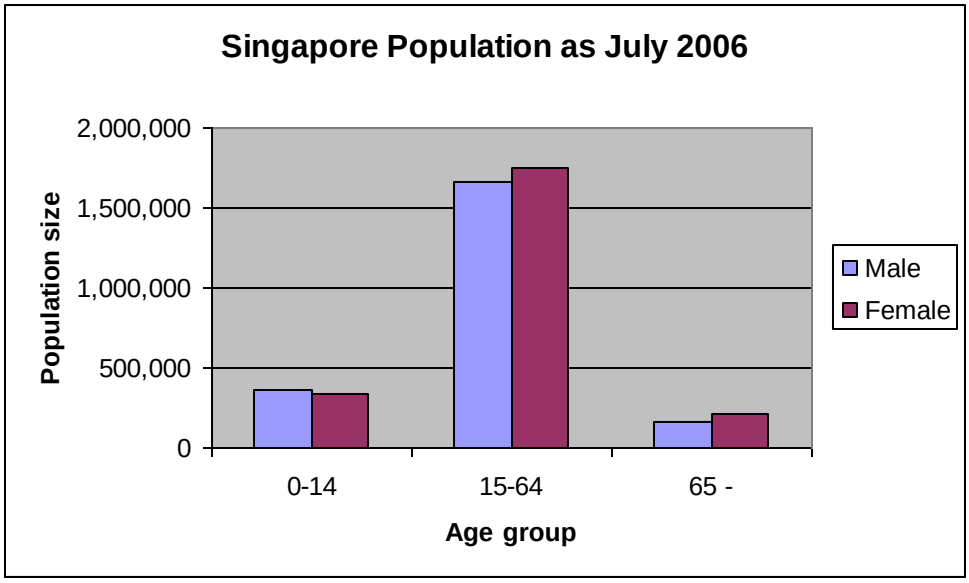


Figure 1-1 a balanced dataset example

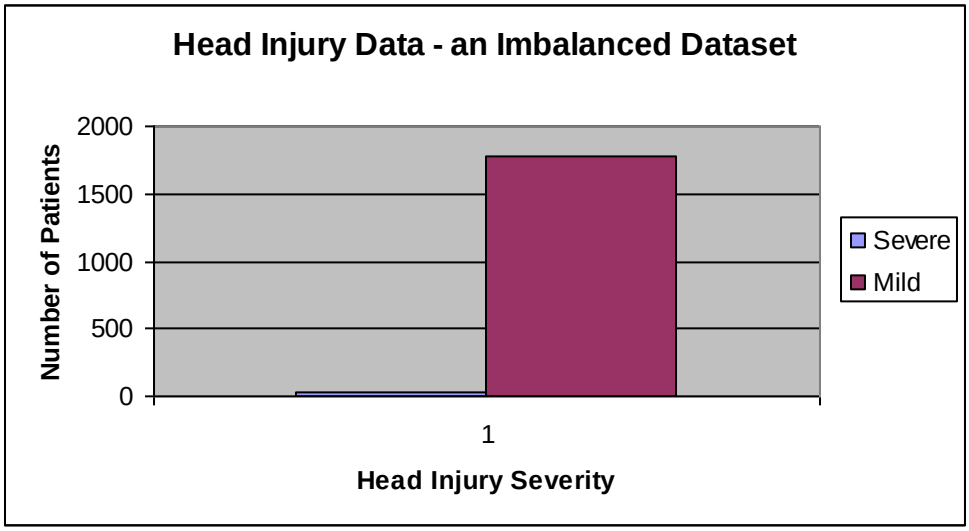


Figure 1-2 an imbalanced dataset example

Class distribution plays an important role in learning. In real life datasets, particularly in medical datasets, class distribution is often uneven, or even highly skewed. For example, in the dataset shown in Figure 1-2, there are only 30 positive (severe) cases

among a total of 1806 head injury patients. There are many more negative examples than positive examples in this dataset, which is therefore imbalanced.

In this work, we focus on imbalanced data learning in the context of biomedical or healthcare outcomes analysis. It is defined as learning from an imbalanced dataset and building a decision model which can correctly recognize the outcomes especially for the minority classes. We assume that the training data are limited, and rare cases and rare classes (discussed in session 4.5.2) exist in the data space.

1.2.2 TYPES OF IMBALANCE

Most of the research on rarity relates to rare classes or more generally, class imbalance. This type of rarity is mainly associated with classification problems. The head injury data set in Figure 1-2 is an example of class imbalance. This type of imbalance is also referred to as "between class" imbalance.

Another type of rarity concerns rare cases. A rare case is normally a sub concept defined within a class that occurs infrequently. For example, in Figure 1-3, the population is a balanced dataset with two classes male and female. However, within each class, age group "0-14" and age group "65-" are rare cases. Unfortunately, it is very hard to detect rare cases in real life, though clustering method may help to identify them. Rare cases, like rare classes, can be considered as a form of data imbalance and it is normally referred to as "within class" imbalance [72].

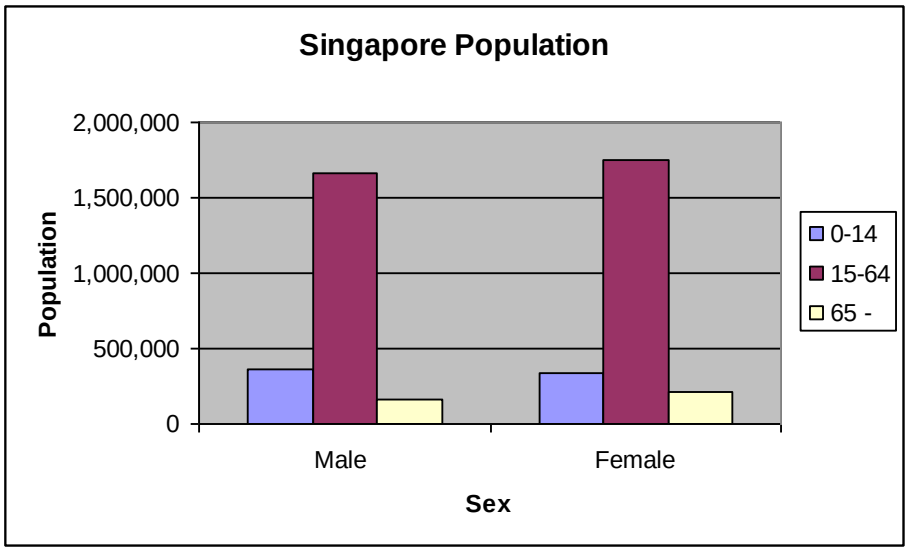


Figure 1-3 an example of within class imbalance

1.2.3 THE PROBLEM OF DATA IMBALANCE

The traditional machine learners assume that the class distribution for the testing data is the same as the training data, and they aim to maximize the overall prediction accuracy on the testing data. These learners usually work well on the balanced data, but often perform poorly on the imbalanced data, misclassifying the minority class, which is normally unacceptable in reality. For example, as shown in the head injury data in Figure 1-2, a trivial classifier can easily achieve 99% accuracy, but it misses all the severe head injury cases. The consequence is very costly – clinicians would miss the best chance to treat those patients who will turn out to be severe.

In order to properly address the imbalanced data problem, the following issues must be considered: a better evaluation metric which is not sensitive to data distribution

should be used; traditional learners should be modified to reduce the bias on minority predictions; or the training space can be re-sampled to form a proper balanced data set, so that existing learners can be applied. We will review all these methods in detail in Chapter 4.

1.2.4 IMBALANCE RATIO

A central concept in imbalanced data learning is the *imbalance ratio*. We define imbalance ratio as the percentage of minority samples among the total sample space. For example in a sample space of 100 examples where 30 are minorities, the imbalance ratio will be 30/100=30% or 0.3.

1.2.5 EXISTING APPROACHES

Existing imbalanced data learning techniques can be generally categorized into two types – algorithm level approaches and data level approaches. Algorithm level approaches either alter the existing machine learning approaches or create new algorithms for addressing the imbalanced data problems. Data level approaches alter the training data distributions by various data sampling techniques. Algorithm level approaches include learning rare class only [67, 82, 100, 127], cost sensitive learning [28, 33, 37, 84, 97, 107, 133, 149], boosting algorithm [27, 45, 76] [75], two phase rule induction [74], kernel modification methods [54, 65, 154, 155], etc. Data level approaches include random oversampling and under-sampling [24, 35, 44, 117], informed under-sampling [93], synthetic sampling with data generation [23], adaptive synthetic sampling [58, 61], sampling with data cleaning techniques [12], cluster based sampling method [73],

progressive sampling [104, 147], generative sampling [91] etc. We will review all these methods in Chapter 4.

1.2.6 LIMITATIONS OF EXISTING WORK

The existing approaches have major limitations. In cost sensitive learning, classification cost is not always possible to identify, and varies from case to case. One class learning normally has a poor performance in the overall accuracy, because it only learns the rare class. Two phase-rule induction performs better only for complex concepts [74]. Boosting was shown that it cannot guarantee improvements in the classification performance [75], instead, its performance is tied to the choice of base learning algorithm, and it will perform poorly if the base learner performs badly. Kernel-based methods are often biased towards majority class if there is not enough data representing the minority concept or if the training space is non linear separable [7, 125, 153]. Sampling, especially smart sampling was shown to be an effective way in addressing imbalanced data learning problems. However, random sampling either duplicates existing information or may remove useful information. Even smart sampling methods [23] only make use of local information to make new samples, but this can be noise instead of possible useful information. Generative sampling samples data in consideration of the statistical distribution of the training data, but it lacks a concrete backbone model as the clear mechanism for data generation. Progressive sampling, on the other hand, concentrates more on the system efficiency rather than performance effectiveness.

1.3 MOTIVATIONS AND OBJECTIVES

Traditional data mining algorithms tend to predict the minorities inaccurately. Optimized algorithms try to add biases to the minorities, so as to improve the overall performance. The performance gained by simply adding biases to the algorithms is often very limited. A lot of efforts have been spent on data level approaches instead. Random sampling is a simple and effective method in addressing imbalanced data problems. However, random sampling does not add any new knowledge to the data repository, except changing the data size [50, 66, 104]. Essentially, random sampling changes the imbalance ratio of the dataset which makes the classifier biased to the minority. Smart sampling on the other hand can create new knowledge by generating synthetic data, e.g., synthetic minority over sampling technique (SMOTE) [23] can generate synthetic data samples using its nearest neighbors. However most of the existing smart sampling methods generate data using local information, i.e., information from a small subspace of the whole training space. Generative sampling [91], on the other hand, makes use of the total data set to generate samples, but it only uses the statistical data distribution. The training space contains much more useful information besides its statistical distribution. If we can extract such useful information from the whole training space and put it into a model, then intuitively the data generated from such a model should be much more meaningful than those data generated using local information or statistical distribution only. When domain expert knowledge is available, the model can even better approximate the true training space with input from the domain experts.

The ultimate objective is to develop a model driven sampling approach, such that it can effectively and efficiently build machine learning models from the whole training space. Meanwhile, this model should also be easily interpreted and updated by domain experts. We will use this enriched model for synthetic data creation.

1.4 CONTRIBUTIONS

The idea of Model Driven Sampling (MDS) approach is to build a probabilistic graphical model to approximate the relationships among the various attributes both qualitatively and quantitatively. The model allows input from domain experts. In this way, the approximate model is built as close as possible to the true model. Thus the data generated from this model has better quality than the data generated using partial information.

We also extend MDS to progressive MDS and context sensitive MDS. Progressive MDS iteratively tries various data distributions aiming to find a better data distribution for each individual imbalanced data set instead of assuming that balanced distribution is optimal. Context sensitive MDS builds various models adapted to different contexts. Models built in this way are more accurate under a certain context, the generated data contains less noise caused by unrelated contexts, and unnecessary computational costs can be avoided.

We have compared our approach with the current best approaches on various simulated data and real data sets with different size, complexity, and imbalance ratio. We

have shown that our approach generally performs better and in the worst case scenario is comparable to the best performing approach.

1.5 OVERVIEW

In this thesis, we first conduct two real life case studies on head injury patients in Chapter 2 to demonstrate the consequences caused by the imbalanced data, which are the main hurdles for the outcomes analysis model to be built. In chapter 3, we explore the nature of the imbalanced data problem, and the reason that it fails the traditional data learners. We then review the existing approaches to address the data imbalanced problem in Chapter 4, including the algorithmic level approaches and the data level approaches. In chapter 5, we introduce the Model Driven Sampling (MDS) approach, and the basics of Bayesian networks. In Chapter 6, we describe our experimental set ups, the datasets, and also the related experimental results. We present a real life case study on asthma control problems using MDS in chapter 7. Progressive MDS and context sensitive MDS are introduced in chapter 8 and chapter 9 respectively. We then conclude our work with a plan for future work in chapter 10.

CHAPTER 2: REAL LIFE IMBALANCED DATA PROBLEMS

2. REAL LIFE IMBALANCED DATA PROBLEMS

In this chapter, we describe two imbalanced data problems in a real life outcomes analysis project - severe head injury management and mild head injury management. In both problems, we have identified that imbalanced class distribution is the main hurdle for outcome predictions. We describe the two problems in detail, the data sets used, the experiment set ups, the traditional learners used, and we also report the results in different scenarios. We will show that imbalanced data cause a big problem for traditional learners, especially in predicting the minority concept.

2.1 SEVERE HEAD INJURY PROBLEM

Severe head injury management is a very costly and labor-intensive process. We have examined the effectiveness of different outcomes analysis methods on head injury management in a uniform manner. We find that no individual model can always outperform the rest. We have shown that class distribution plays a very important role in prediction accuracy and this problem is indeed a multi-class imbalanced problem. Some of the following results were reported in an earlier paper [111].

2.1.1 INTRODUCTION

Severe head injury is one of the major causes of death and disability worldwide. The process to manage head injury patients is very costly and labor-intensive. In order to optimize head injury management process and resource utilization in hospitals, many efforts have been done in head injury outcomes analysis [30, 34, 59, 109]. For example, Choi et al. [30] achieved an overall prediction rate of 77.7% using a prediction tree for outcome after severe head injury. Nissen et al. [109] used Bayesian Network to get an 84.3% accuracy to predict live (good recovery) and mild disability, 83.6% accuracy to predict death or vegetative survival, and an overall accuracy of 75.8% on a group of 324 patients. Dora et al. [34] designed a decision support system to improve severe head injury treatment procedures. However, we found that inconsistencies in the literature make the comparisons among different results difficult. In particular, one of the most important inconsistencies is that the definitions of class labels for performance evaluation in different papers are inconsistent. Usually, the outcome of a severe head injury patient can be defined as one of the five Glasgow Outcome Scores (GOS 1-5): death, vegetative state, severe disability, moderate disability or good recovery. In head injury outcomes analysis, these five categories can be combined in different ways to build a classification model, e.g., a) death (GOS 1) and live (GOS 2-5) [128], b) death or vegetative state (GOS 1-2), severe disability (GOS 3), and moderate disability or good recovery (GOS 4-5) [109], c) (GOS 1-3) and (GOS 4-5) [9]. Different combinations of GOS scores will affect prediction accuracy significantly, and make results from different work incomparable.

Table 2-1 Description of head injury dataset with list of prognostic factors

	Cases	Min	Max	Mean
1. AGE	706	10	97	45.64
2. Gender	706	1	2	1.22
3. Ethnic Group	706	1	4	1.56
4. Mechanism of injury	706	0	6	2.15
5. Types of motor vehicle accident	706	0	7	1.58
6. Alcohol use	706	0	3	.15
7. Presence of traumatic SAH	706	0	2	1.50
8. Presence of cervical injury	706	1	2	1.92
9. Presence of multiple injuries	706	1	2	1.76
10. Pre-resuscitation GCS	703	3	15	9.00
11. Pre-resuscitation papillary light response	703	0	2	1.67
12. Presence of coagulopathy	689	0	2	1.61
13. Presence of hypoxia	706	1	2	1.89
14. Presence of hypotension	706	1	2	1.88
15. Post-resuscitation GCS	698	3	15	7.79
16. Post-resuscitation papillary light response	691	0	2	1.59
Outcome Glasgow Outcome Scale	706	1	5	3.07

In our experiment, we found that Minimum-Description-Length-based discretization method performs more stably in improving prediction accuracy. We compared evaluation results from both training data and cross validation. We have applied different methods to a data set collected from a local hospital and tried different ways to combine GOS scores as class labels. The results confirmed that different combinations of GOS scores affect prediction results significantly. It suggests that a consistent model has to be able to deal with various GOS combinations, and any fair model comparison should be performed using the same way of GOS combination.

2.1.2 DATA SUMMARY

Our data set contains 706 severe head injury (with Glasgow Outcome Score of 5 or less) patient records, collected in a Singapore hospital from January 1999 to March 2005. Data collected include demographic information, details of injury, presence of coagulopathy, hypoxia (defined as SPO2 <90), hypotension (defined as systolic blood pressure < 90mmhg), pre and post resuscitation Glasgow Coma Score (GCS) and pupillary light response. A single independent scorer (either in outpatient clinic or via telephone contact) determined the outcomes of these patients using the Glasgow Outcome Scale (GOS) at 6 months post injury. In the database, there are more than one hundred attributes in each patient record. Based on domain knowledge and feature selection, sixteen variables measured at admission time were chosen for the experiments. The descriptions of the variables are summarized in Table 2-1. The distribution of GOS scores in our data set is shown in Figure 2-1, from which we know that the data is not equally distributed on different GOS scores: most of the patients are either well recovered or dead. In the data set, there are some missing values. For numeric variables, we filled missing values with the means of the known values, and for categorical variables, the missing values are filled in with the modes of the known values.

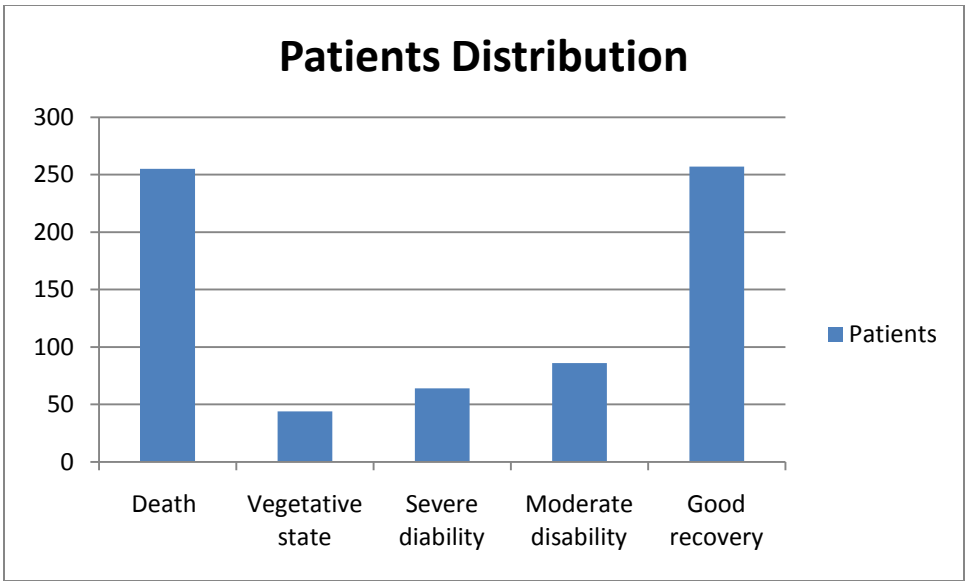


Figure 2-1 Data distribution with GOS score

2.1.3 EVALUATION MEASURES AND DATA DISTRIBUTIONS

We defined prediction accuracy as the total number of correctly predicted samples divided by the number of the total samples. We applied a total of 6 machine learning algorithms (AODE [143], Bayesian Network, Logistic Regression, Support Vector Machine, and Neural Network) to our data set, and we defined the class labels in 5 different ways:

1) 5 class labels. One for each GOS score: [death], [vegetative state], [severe disability], [moderate disability], [good recovery]; the data distribution is shown as in Figure 2-1;
2) 2 class labels: [death, vegetative state] and the rest; the data distribution is 0.424 for death and vegetative state.
3) 2 class labels: [death] and the rest; the relevant is 0.361 for death state;

4) 2 class labels: [good recovery] and the rest; the relevant frequency is 0.364 for good recovery state.
5) 2 class labels: [good recovery, moderate disability], and the rest; the relevant frequency is 0.486 for good recovery and moderate disability.

We conducted 30 experiments in all using six methods on five data sets. In each experiment, we applied 10-fold cross validation. In other words, we performed training and testing for ten rounds. At each round, we randomly split the data into 10 pieces. We then trained our model using 9 pieces of them, and tested it on the 1 remaining piece to get the accuracy. Finally we obtained the overall accuracy by taking the average from 10 rounds of testing results. We also tested our models on the training data in each experiment. All the experiment results are summarized in section 2.1.5. The experiments set up and result analysis are also summarized in our technical report [158].

2.1.4 ABOUT THE TRADITIONAL LEARNERS

2.1.4.1 Bayesian Network

Bayesian Networks model dependencies among a group of variables using directed acyclic graphs. A Bayesian network can be used to infer the states of the unknown variables with prior probabilities and known evidence, and it has an advantage of handling missing data. Besides giving promising performance, a Bayesian Network can also reveal the underlying relationships among the variables or prognostic factors in our case. We used Bayesnet and another Bayesian method AODE [143] from Weka [151]. AODE achieves highly accurate classification by averaging over all of a small space of

alternative naïve-Bayes-like models that have weaker (and hence less detrimental) independence assumptions than naïve Bayes. The resulting algorithm is computationally efficient while delivering highly accurate classification on many learning tasks.

2.1.4.2 Decision Trees

Decision trees [123] represent a supervised approach to classification. A decision tree is a simple structure where non-terminal nodes represent tests on one or more attributes and terminal nodes reflect decision outcome. It can be used to explain why a question is being asked. Decision tree is a map of the reasoning process. Decision trees are excellent tools for helping us choose between several courses of action. They provide a highly effective structure within which we can lay out options and investigate the possible outcomes of choosing those options. They also help us to form a balanced picture of the risks and rewards associated with each possible course of action. The decision tree used in this report is J48 developed by J. Ross Quinlan, the very popular C4.5.

2.1.4.3 Logistic Regression

Logistic regression (LR) is part of a category of statistical models called generalized linear models. Logistic regression allows one to predict discrete outcomes, such as group membership, from a set of variables that may be continuous, discrete, dichotomous, or a mix of any of these. In LR, univariate analyses are first performed to consider the significant risk factors. Then either a backward or forward stepwise method is chosen. In the forward method, one factor is added at a time to increase the prediction performance; in the backward method, one factor is removed at a time to increase (or keep) the

prediction performance. After each addition or removal, a beta coefficient or relative weight for that factor is defined. Odds ratios and risk ratios can then be calculated, which are very helpful for decision making. The LR we used is originally from the paper of le Cessie and van Houwelingen [85].

2.1.4.4 Support Vector Machine

Support vector machines (SVMs) [19] are statistical-learning-based methods for classification and regression. When used for classification, the SVM algorithm creates a hyperplane in a feature space with higher dimension that separates the data into two classes with the maximum-margin. Given training examples labeled either "yes" or "no", a maximum-margin hyperplane is identified which splits the "yes" from the "no" training examples, such that the distance between the hyperplane and the closest examples (the margin) is maximized. The SVM we used implements John Platt's sequential minimal optimization algorithm [118] for training a support vector classifier.

2.1.4.5 Neural Networks

Neural Network or Artificial Neural Network is an information processing technique inspired by the way biological brain system works. A neural network contains a number of interconnected processing nodes (or neurons) working in parallel to solve a particular problem.

Neural networks are powerful in deriving meanings from complex or imprecise data, which can be used to understand or recognize things that are too complex to be noticed by other methodologies. A neural network simulates human brains by learning

expertise from examples, and stored knowledge in interneuron connection strengths known as synaptic weights. In our experiment, we applied multilayer perceptron (MLP) which is the most commonly used neural network architecture. MLP is a supervised network which requires a labeled training data for learning. Back propagation is used to adjust the weights a small amount at a time in a way that reduces the error. The ultimate goal of the training process is to reach an optimal solution based on our performance measurement.

2.1.5 EXPERIMENT ANALYSIS

From our experiments, we have examined the strengths and limitations of different outcomes analysis methods for head injury management in a systematic manner. From the experiments we have found that all the methods can achieve comparable prediction accuracy on the testing data (around 76% ~ 82%) under different assignments of the two GOS classes, though the best performance is not always achieved by a single algorithm. However, the best prediction accuracy on five GOS data set is only 62% as shown in Table 2-2.

Table 2-2 Results for 5 class labels

Methods	Training	Testing
AODE	67.71 %	61.05%
Bayesnet	61.75 %	60.05%
Decision Tree	69.97 %	62.18%
LR	65.86 %	61.47%
SVM	64.73 %	62.46%
Neural Network	89.23 %	52.83%

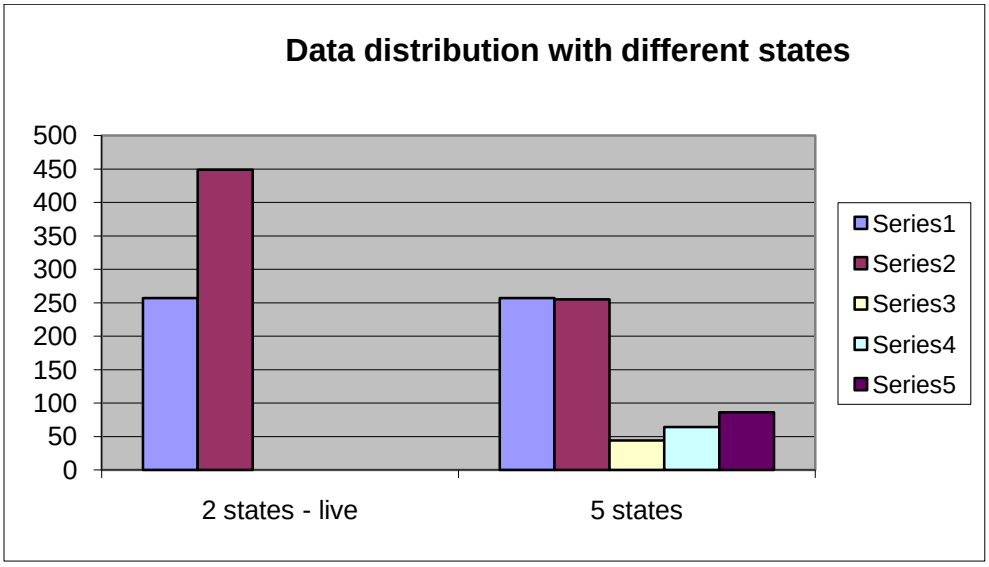


Figure 2-2 Data distribution with different class labels

Table 2-3 Results for 2 class labels
(death vs all others)

Methods	Training	Testing
AODE	82.29 %	80.31 %
Bayesnet	79.46 %	79.32 %
Decision Tree	85.12 %	82.15 %
LR	84.70 %	81.16 %
SVM	83.42 %	81.86 %
Neural Network	96.17 %	77.76 %

Table 2-4 Results for 2 class labels
(death-vegetative vs others)

Methods	Training	Testing
AODE	82.72 %	81.30 %
Bayesnet	79.46 %	79.32 %
Decision Tree	87.54 %	80.59 %
LR	84.42 %	81.58 %
SVM	84.13 %	79.46 %
Neural Network	95.75 %	76.35 %

Table 2-5 Results for 2 class labels
(good recovery & mild-disable vs others)

Methods	Training	Testing
AODE	82.44 %	79.60 %
Bayesnet	80.03 %	79.04 %
Decision Tree	82.86 %	79.75 %
LR	81.87 %	79.89 %
SVM	83.29 %	77.90 %
Neural Network	96.32 %	76.63 %

Table 2-6 Results for 2 class labels
(good recovery vs others)

Methods	Training	Testing
AODE	82.01 %	78.61%
Bayesnet	79.60 %	78.75 %
Decision Tree	83.00 %	80.59 %
LR	81.73 %	79.04 %
SVM	83.29 %	80.31 %
Neural Network	96.46 %	77.76 %

After examining the class distributions with different label assignment as shown in Figure 2-2, we realized that performance drop might not be caused by different class labels. Instead, it is probably caused by class imbalances in five class problem, which is a multi-class imbalanced problem. As shown in Figure 2-2, the imbalance ratio for the two class data is 0.36 (0.5 is the maximum value), while the worst relative imbalance ratio between any two classes in the five class problem is only 0.14.

The assumption of the traditional learners is that the training data and testing data are balanced. The assumption of total accuracy gives equal weight to each class in the data. However, neither assumption is valid anymore in imbalanced data. In an imbalanced data set, both training data and testing data have skewed data distributions; minority concept is often more important, and thus needs more attention. We will explain the reason that prediction accuracy is not a proper evaluation measure for imbalanced data mining in chapter 3.2. We will discuss proper evaluation measures in chapter 4.4.

In this dissertation, we mainly focus on binary imbalanced data learning problems, and most of the techniques should be able to be directly applied to multi-class imbalanced problems. In particular, all data sampling techniques discussed in this dissertation can be applied to multi-class imbalanced problems with minor changes [23].

2.2 MINOR HEAD INJURY PROBLEM – A BINARY CLASS IMBALANCED PROBLEM

In the previous section, we have discussed a multi-class imbalanced data problem. When the data is modified to a binary class problem, the imbalance level is reduced. Thus the performance is improved too. When the class imbalance is not obvious, traditional data mining algorithms can be used to build an outcomes analysis model with reasonable performance. In this section, we describe a highly imbalanced problem in mild head injury management [112]. In this problem, we will show that traditional learners cannot give an acceptable performance, especially in identifying the minority concept; and thus we need research on imbalanced data learning techniques.

2.2.1 BACKGROUND

Clinically, we define minor head injury or mild head injury as a head injury with Glasgow Comma Scale (GCS, the scores on the scale range from 3, indicating no motor or verbal response and no opening of the eyes, to 15, indicating normal motor and verbal responses and normal eye opening.) value on presentation ranging from 13 to 16 [112, 129, 130]. Minor head injury may cause the brain to have trouble working normally for a

short time. Minor head injuries are usually not a serious problem. They are most often caused by a blow to the head. A minor head injury may happen because of a fall, a motor vehicle crash, or a sports injury. Sometimes being forcefully shaken may cause a minor head injury. Every minor head injury is different. Right after the injury, the patient may seem dazed. Other symptoms may show up right away. Some symptoms may not happen for days or weeks after the injury. Symptoms of a minor head injury may last from a few hours to a few weeks. After the injury, one or more of these symptoms may show up:

- Mild to moderate headache.
- Dizziness or loss of balance.
- Nausea (feeling sick) or vomiting (throwing up).
- Change in mood (such as feeling restless or irritable).
- Trouble thinking, remembering things, or concentrating (giving full attention to one thing for a period of time).
- Ringing in the ears.
- Drowsiness or decreased amount of energy.
- Change in normal sleeping pattern (the patient may sleep more than usual or have trouble sleeping).

Normally patients with minor head injury can be well recovered without hospitalization. However, there is also a small group of patients who may have been hurt in other ways when they got their head injury. In this group of patients, minor head injury may mask more serious problems, such as bleeding or a blood clot in the brain, which potentially can develop to severe head injury and lead to death. To correctly detect this

group of patients and prevent mistakenly discharging them is a crucial job for physicians. The use of computed tomography (CT) can effectively detect this group of high risk patients. However, CT scan is very costly, and the majority of patients are in the negative group (normally more than 95%). Therefore, there is a much controversy about the use of CT for patients with minor head injury. Instead of high accuracy, the goal for mild head injury management in the outcomes analysis framework is to achieve higher sensitivity. Normally physicians require a sensitivity of 100% to ensure that all potential severe head injury patients are correctly detected [112]. We then try to improve the specificity to minimize the use of CT in patients with minor head injuries.

2.2.2 DATA SUMMARY

We carried out this cohort study on a dataset containing 1806 patients' records. There are 71 attributes in the dataset all together, of which 43 are selected as the prognostic factors according to the Chi-Square test. The binary factor "talk & deteriorate" is our targeted outcome variable, it means whether the patient can talk ("negative cases") or deteriorate ("positive cases"). The value "yes" corresponding to positive cases which has takes only 1.6 percent, and value "no" corresponding to negative cases which takes 98.4 percent. The data distribution is as shown in Figure 2-3.

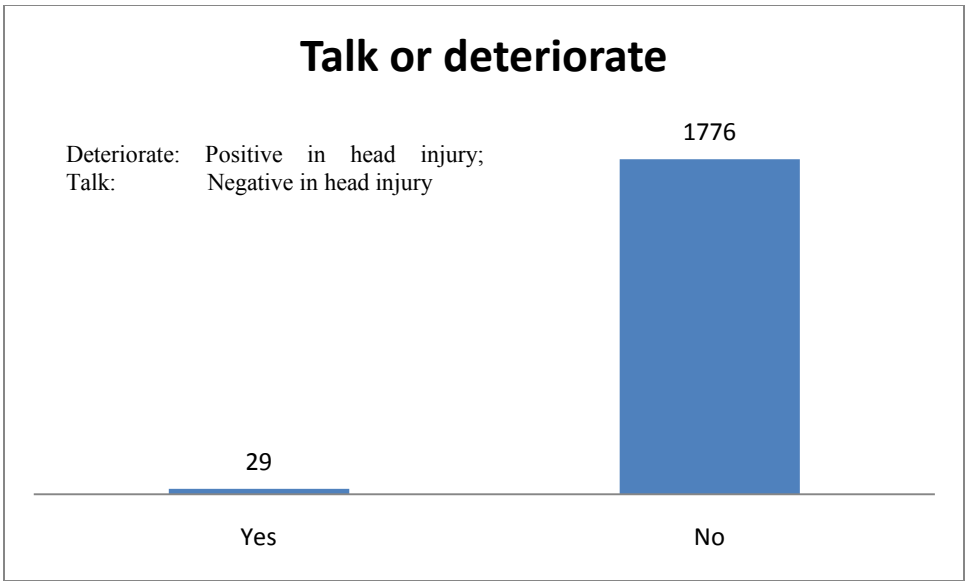


Figure 2-3 Minor head injury outcome distribution

2.2.3 OUTCOME PREDICTION ANALYSIS

We applied five approaches in this experiment: Bayesian methods, Decision Tree, Support Vector Machine, Neural Networks, and Logistic Regression. 10-fold cross validation is used to report the experimental results. Among all the approaches, Bayesian methods have the most promising and stable performance in predicting the positive class, e.g., both Naive Bayes and BayesNet can correctly recognize 24 positive examples out of 29 with a sensitivity of 82.76%; AODE can correctly predict 22 positive examples out of 29 with a sensitivity of 75.86%. Their overall performances are also very good compared to the rest. Three other "state of the art" classification algorithms (Decision Tree, SVM, Neural Networks) and the traditional classification method Logistic Regression give very

good overall performances, but they are very poor in predicting positive examples. (Detailed Running results are shown in Table 2-7)

Table 2-7 Outcome prediction results comparison for mild head injury

	Accuracy	Correctly identified positive instances (out of 29)	Area under ROC curve
Bayes -- AODE	98.5042 %	22	0.9768
Bayes -- Naive Bayes	97.1191 %	24	0.9773
Bayes -- BayesNet	97.0637 %	24	0.9751
Decision Tree -- J48	98.8366 %	14	0.691
Support Vector Machine -- SMO	98.0055 %	9	0.6507
Neural Networks -- MultilayerPerceptron	98.3934 %	14	0.9607
Logistic Regression	97.1191 %	13	0.7783

2.2.4 ROC CURVE ANALYSIS

2.2.4.1 ROC curve analysis for data with 43 attributes

The ROC curve [47, 48] for the above five mentioned methodologies on the dataset with 43 attributes are shown in Figure 2-4. And the area under the curve is shown in Table 2-9. Normally, physicians are not willing to accept the risk of missing a positive case. In a survey of emergency physicians, more than half insisted that a clinical decision system for minor head injury must have a sensitivity of 100 percent [56]. Thus the use of CT scan for minor head injury patients is common, but such screening is expensive. According to one estimate, even a 10 percent reduction in the number of CT scans in patients with minor head injury would save more than $20 million per year. In the ROC

curve, by setting sensitivity to one, we can get the four highest specificities from Naive Bayes, AODE, Neural Networks, and BayesNet. (As shown in Table 2-8)

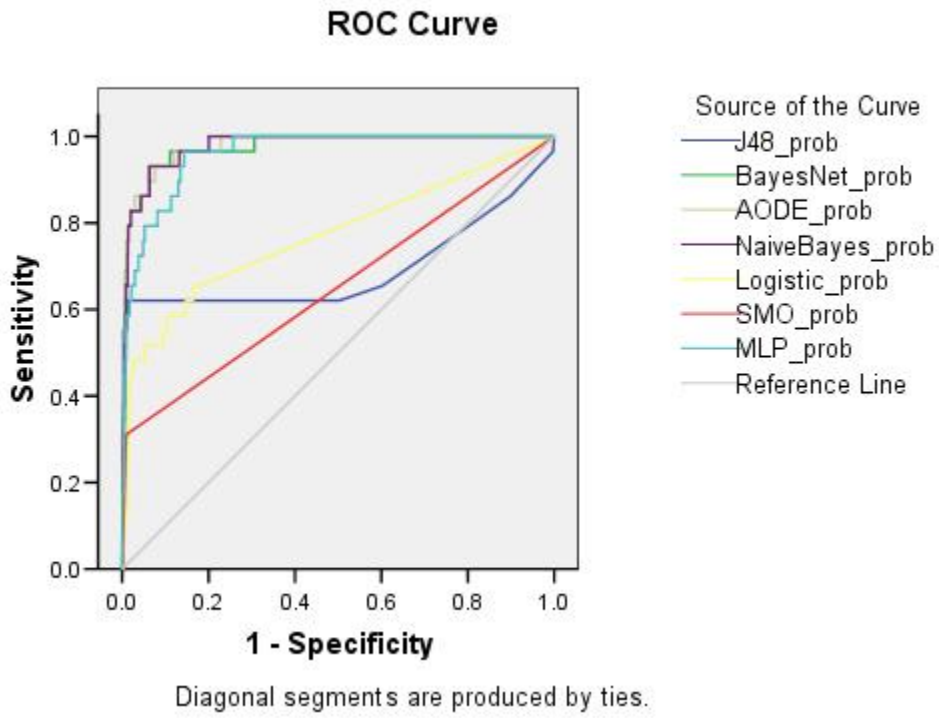


Figure 2-4 ROC curve analysis for mild head injury dataset with 43 attributes

Table 2-8 Sensitivity and specificity analysis for 43 attributes

	Sensitivity	Specificity
Decision Tree J48	1	0
BayesNet	1	0.694
AODE	1	0.772
Na ïve Bayes	1	0.800
Logistics Regression	1	0
Support Vector Machine	1	0
Neural Networks	1	0.743

Table 2-9 Area Under the Curve for 43 attributes

Test Result Variable(s)	Area	Std. Error(a)	Asymptotic Sig.(b)	Asymptotic 95% Confidence Interval	
				Lower Bound	Upper Bound
DecisionTree	.526	.065	.627	.398	.653
BayesNet	.942	.027	.000	.889	.995
AODE	.937	.031	.000	.875	.998
NaiveBayes	.935	.029	.000	.878	.992
Logistic	.834	.044	.000	.748	.921
SVM	.598	.060	.064	.481	.715

a Under the nonparametric assumption b Null hypothesis: true area = 0.5

2.2.4.2 ROC curve analysis for data with 38 attributes

According to doctors' suggestion, five treatment variables (neuroop, comp, tdisch, ctabr, ariance) should be excluded for CT scan prediction. Therefore we obtained the following ROC curve by using the remaining 38 variables (Figure 2-5 and Table 2-11). Compared with the rule based system from [129, 130] which derived a 50% specificity, even our best performance of specificity 42.2% achieved by Bayesian network classifier looks weaker. Although we worked on different datasets, which makes exact comparison unreasonable, the results are still far away from the acceptable specificity 70% without losing the sensitivity recommended by doctors. This shows that existing algorithms cannot address the imbalanced learning problems well. In order to further improve the specificity without affecting the sensitivity, we must look for the approximate imbalanced data learning techniques.

Table 2-10 Sensitivity and specificity analysis for 38 attributes data

	Sensitivity	Specificity
Decision Tree J48	1	0
BayesNet	1	0.422
AODE	1	0.262
Na ïve Bayes	1	0.326
Logistics Regression	1	0
Support Vector Machine	1	0
Neural Networks	1	0.197

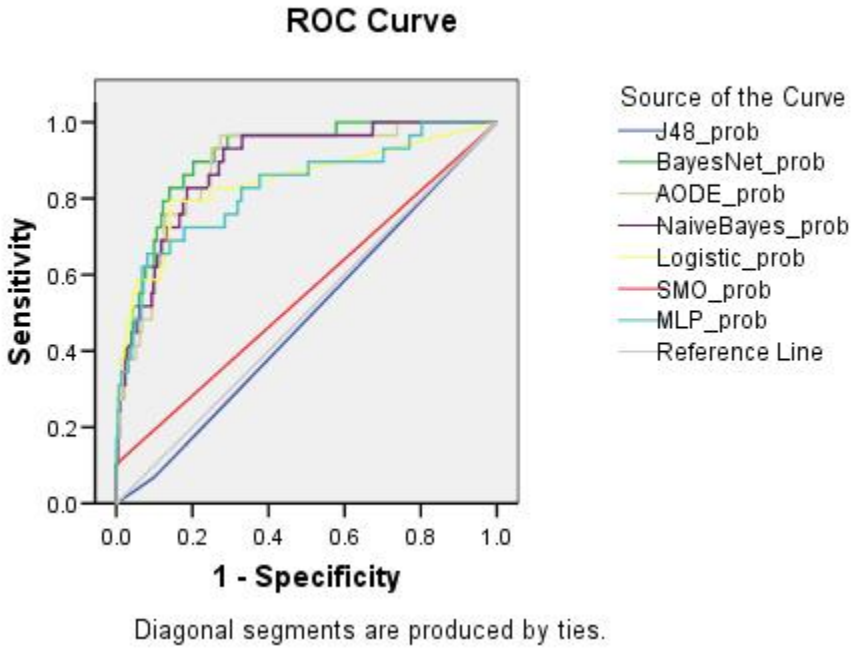


Figure 2-5 ROC curve analysis for mild head injury dataset with 38 attributes

Table 2-11 Area Under the Curve for 38 attributes

Test Result Variable(s)	Area	Std. Error(a)	Asymptotic Sig.(b)	Asymptotic 95% Confidence Interval	
				Lower Bound	Upper Bound
DecisionTree	.484	.053	.764	.381	.587
BayesNet	.844	.032	.000	.781	.907
AODE	.812	.037	.000	.739	.885
NaiveBayes	.810	.036	.000	.739	.881
Logistic	.807	.044	.000	.721	.892
SMO	.500	.054	.996	.394	.606
MLP	.804	.045	.000	.717	.892

a Under the nonparametric assumption
b Null hypothesis: true area = 0.5

2.2.4.3 Experiment analysis

From the above results, we note that some of the state of the art classification algorithms can achieve reasonably good overall accuracy. However, we also realize that they perform badly in predicting the positive cases in case of imbalanced data which is very crucial to the patients and clinicians. By analyzing the ROC curves, we can get a better idea on the performance evaluations. The Bayesian method seems to be minimally affected by the imbalanced data training, and it shows stable performance over different datasets. This experiment has shown the importance of imbalanced learning in critical care, and the right way for performance evaluation. It suggests that choosing appropriate evaluation metrics for imbalanced data learning is crucial. It also shows that accuracy is not a proper evaluation measure for imbalanced data problem, and it is necessary to choose a proper evaluation method for imbalanced data learning.

2.3 SUMMARY

From the above two problems, we have observed that traditional machine learning techniques are not suitable for imbalanced data learning problems. New methodologies need to be proposed for imbalanced data learning. We have discovered that accuracy is not a proper evaluation measure for imbalanced data learning. ROC curve is a good evaluation measure, because it can give a tradeoff between the predictions on majorities and minorities without bias. However, ROC curve is not suitable for large cohort studies and comparisons, as it is infeasible to compare over all the threshold points on a ROC curve; instead, researchers usually choose one typical point such as g-Mean [81] from the ROC curve as the evaluation measure.

In the following chapters, we will survey existing techniques that are targeting to address imbalanced data problem and analyze their limitations, compare different evaluation measures for imbalanced data learning, and propose a novel approach – Model Driven Sampling to address the imbalanced data problem.

CHAPTER 3: NATURE OF THE IMBALANCED DATA PROBLEM

3. NATURE OF THE IMBALANCED DATA PROBLEM

Besides the problems mentioned in the previous chapter, most other existing electronic patient records are characterized by imbalanced learning data, where at least one class is under represented relative to others. The imbalanced data problem also exists in many other critical domains, like intrusion detection [38, 39], satellite oil spill detection [83], disease diagnosis [7] etc. The problem of imbalanced data is often associated with asymmetric costs of misclassifying elements of different classes. In addition, the distribution of the test data may differ from that of the learning samples and the true misclassification costs may be unknown at learning time. There are many other reasons causing imbalanced data to be a problem. Therefore, in order to study the imbalanced data problem, we need to understand the nature of imbalance. There are different types of imbalance existing in the imbalanced problem. Meanwhile, the imbalanced data problem is also affected by various other factors, including the data complexity, the training data size, and the imbalance levels.

In this chapter, we will look at the nature of the imbalanced data problem – absolute rarity, relative rarity, noisy data and data fragmentation. In particular, we will

make use of simulated data to study how the imbalanced data problem behaves when we change the three factors – data complexity, training data size, and the imbalance level.

3.1 NATURE OF DATA IMBALANCE

Although by definition, any unequally distributed data can be considered as an imbalanced data, in this research area only significant imbalances can be considered as an imbalanced data problem. There is no specific quantification about the significance, and the level of significance may vary over data sets or domains. Not surprisingly, we often meet extreme imbalances in the order of 100:1, 1,000:1, or even 10,000:1 and so on [60, 83, 116]. The imbalance caused by unequal data distributions between two different classes is referred to as "between class imbalance". Besides binary imbalanced class problems, there are also multi class imbalanced problems [6, 29, 163, 164]. In this thesis, we are focusing on binary imbalanced data problems. The techniques discussed in this thesis can be easily applied to multi-class imbalance problems with minor changes [23].

In contrast to between class imbalance, there also exists within-class imbalance which is caused by the sub concepts (disjuncts) found inside the minority concept [64]. Within class imbalance is closely related to small disjuncts [70, 73]. Another type of data imbalance is relative imbalance related to absolute rarity. For example in the mild head injury data set, we have 1776 majority patients and 29 minority patients. The poor performance on minority class might be caused by lacking of enough information on minority concept, in addition of the imbalanced distribution.

3.1.1 ABSOLUTE RARITY

The main problem with the imbalanced data is the lack of data. When the minority data is rare, such that in theory no data learners can approximate the true minority concept, we say this type of data imbalanced problem is caused by absolute lack of data or absolute rarity. In this type of problems, it is impossible to find the minority concept in the rare data because the data does not contain sufficient information. As demonstrated in Figure 3-1, the solid rectangle surrounding A is the original region for the rare cases. The dashed rectangle is the estimated region for the rare cases; obviously the one on the right side is a more appropriate estimate of the region because there are more learning examples; while the left side estimation is almost out of the region because the learning samples are too few. So rare cases may be due to lack of data, and the impact of these rare cases has been studied. Weiss et al. [144] studied the effect of rare cases on a set of synthetically generated datasets and showed that rare cases have higher misclassification rate than common class; this is referred to as the problem with rare classes.

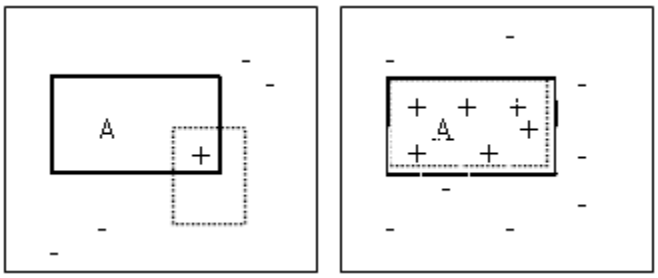


Figure 3-1 the impact of absolute rarity

It is also shown that absolute rarity can cause small disjuncts [144]. Empirical studies showed that small disjuncts can bring more errors than large disjucnts in general

[8, 138, 139, 144, 146], which is the direct result of lack of data. Thus to understand why absolute rarity is a problem, we need to understand why the small disjuncts have so many problems. Instead of representing something meaningful, some small disjuncts may be due to noises. Most of the algorithms used statistical significance test to prevent data over fitting. Disjuncts that cover few samples will normally be rejected; therefore some of the significant small disjuncts may be filtered out at the same time. For example in [64], in a binary balanced data set, a disjunct is 99% significant if and only if it covers at least 7 training examples. These techniques work well for large disjuncts. However they are not reliable for small disjuncts, because the significance cannot be reliably estimated and meaningful small disjuncts might be eliminated instead. Empirical results show that the strategy of eliminating all small disjuncts will increase the overall error rate [64].

3.1.2 RELATIVE RARITY

Comparing to absolute rarity, relative rarity means that one type of object is relatively rare to the other objects. The problem with relative rarity is similar to absolute rarity in that the rare objects are hard to detect. For example in Figure 2-3, even if we have ten times of the minority cases – 270 cases, it is still relatively rare to 1779 majority cases. Rare objects are difficult to detect using greedy search heuristics. Because rare objects may depend on conjunction of many conditions and thus examining any individual condition may not provide much information. For example, we want to mine the association rule “food processor and cooking pan”. Both of these two items are rarely bought in super markets, so even though people buy one of them will normally buy

another one at the same time. This association may not be found because they are rare. In order to find this association, the threshold value will be reduced to be very low which will then cause enormous number of ways of false associations which was referred to as the rare item problem in [92]. Random co-occurrences of events will make the mining of the true associations between rare items difficult, which is one problem of relative rarity.

3.1.3 NOISY DATA

Noisy data has always been a problem in machine learning. However, it has a greater impact on rare data. Consider the case in Figure 3-2, "+" means positive examples, and "-"means negative examples. A is the large disjunct in positive class, and B is the small disjunct in positive class. Dashed rectangles represent the predicted model for A and B. The left side shows the case without noisy data, and both disjunct A and disjunct B are correctly identified. The right side shows the case with noisy data. In this case, the learner cannot distinguish between rare cases and noise and thus misclassify disjunct B. Even if the learner was modified to generalize less to locate minority class B, the noisy data will then be misclassified to be class B to lead data over fitting problem. Unless class B is very important, one should not adjust the bias of the learner to include them. In this case, the learner cannot distinguish the true rare cases and noise [144].

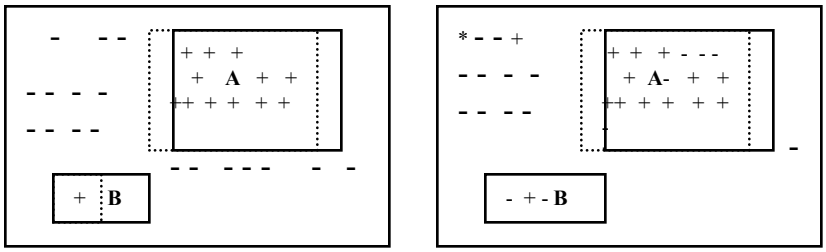


Figure 3-2 the effect of noisy data on rare cases

3.1.4 DATA FRAGMENTATION

Data fragmentation is mainly caused by the "divide and conquer" approach employed in many data learning algorithms. The problem is decomposed into smaller and smaller pieces, and thus the instance space is being partitioned into smaller and smaller pieces. Decision tree is one of the examples which may lead to data fragmentation [51]. Data fragmentation is a problem because regularities can be only found within each individual partition which will contain less data. This is particularly a problem for rare data. Thus all iterative divide and conquer approaches have difficulties in mining rarity class. Therefore, machine learning algorithms that do not employ the divide and conquer approaches are preferred in imbalanced data learning.

3.1.5 INDUCTIVE BIAS

Many machine learning systems make use of a general bias to avoid data overfitting [64]. Most methods that address imbalanced data (small disjuncts or rare cases) try to change the bias of the machine learners. However, most of data learners are biased to the majority class in the priors. For example, in the decision tree method, if there are no

examples covered in a certain branch, it will try to estimate that branch using the majority examples. Therefore the results are biased against the rare class.

3.2 IMPROPER EVALUATION METRICS

Evaluation metrics are used to guide and evaluate machine learning algorithms. Most machine learning algorithms are targeting to optimize their evaluation metrics. Classification accuracy computes the fraction of examples that have been correctly classified. An algorithm using classification accuracy as the evaluation metric will try to maximize the classification accuracy. The flaw with classification accuracy for imbalanced data is – rare class has less impact on accuracy than common classes. For example in the mammography data set – a collection of mammography images for a group of distinct patients [57, 152], there are 10923 healthy patients and 260 cancerous patients. Suppose we have a classifier achieving 100% accuracy on the majority class but only 10% of prediction accuracy on the minority class. This would suggest that 234 cancerous patients are classified as healthy patients, but the overall accuracy is as high as 98%. In the medical domain, the consequence of this diagnosis is very costly even though it achieves very good overall performance accuracy. An empirical study by Weiss and Provost [148] concludes that accuracy leads to a poor performance for minority class samples, it shows that the error rate for minority class is 2-3 times of the majority class. The minority class has much lower precision and recall than the majority class. Many people observe that for extremely unbalanced dataset, the recall for minority class is often 0, there are no classification rules generated for minority class.

There are also problems for the evaluation metrics guiding search algorithms in machine learning. Consider the example in which we build a decision tree, we conduct a goodness test to determine the overall purity values for creating new branches. The metric (e.g. information gain) prefers a test that results in a balanced tree where purity is increased for most of the examples to a test that yields high purity for a relatively small subset of the data but low purity for the rest. The problem with this is that a single high purity branch may identify a useful rare case.

Association rule mining uses the support and confidence metrics to guide search for association rules [113]. Support measures the number of records that contain the association, while confidence measures the percentage of times that the association is found. In general, association rule systems only find rules that have minimum support *minsup*. This allows much of the search space to be pruned. For efficiency reasons, *minsup* cannot be set low enough to identify rare associations.

3.3 IMBALANCE FACTORS

From the previous section, we know that imbalanced data distribution is not the only factor causing data mining difficulties. For example, in mild head injury data, if we increase the minority patients to 290, we get an imbalanced data with a lower imbalance level. However, the predictions on minority might not get improved much by changing the level of imbalance. As shown in [12, 148], the data complexity and training data size also play important roles in imbalanced data learning.

3.3.1 IMBALANCE LEVEL

Imbalance level measures how imbalanced the data is. We define five imbalance levels i = 1, 2, … 5, at each level i, the corresponding imbalance ratio (as defined in section 1.2.4) IB = 1/(1+32/(2^i)). When i = 1, we have most imbalanced data with IB = 1/17; when i = 5, we have a balanced data set with IB =1/2. The smaller the value of i is, the more imbalanced the data is.

3.3.2 DATA COMPLEXITY

Data complexity is a broad term comprising issues like data overlapping, lack of representative data, small disjuncts, number of disjuncts, data noise, missing values, etc. For illustration, we make use of the number of disjuncts or the number of intervals to simulate the complexity of the data sets. The more intervals in the data, the more complex the data is [69]. We use c = 1, 2 … 5 to represent 5 data complexity levels. At complexity level c, there are 2^c regular intervals. As shown in the example in Figure 3-3, the data are generated along the line in the [0, 1] range. There are two classes - class 1 is the majority and class 0 is the minority. The [0, 1] range is divided into a number of intervals according to the complexity of the data. At complexity of level 2, there are 2^2 = 4 regular intervals. Different class values are assigned for adjacent intervals. [0, 0.25) and [0.5, 0.75) are class 1 intervals; [0.25, 0.5) and [0.75, 1] are class 0 intervals. The data is generated randomly from each interval, and the size of the data is determined by the training size and the imbalance level.

3.3.3 TRAINING DATA SIZE

What makes the imbalanced data a problem is the combination of imbalanced data and small training data size [20, 126]. We often encounter real life data sets with high dimensionality and small samples size. Small sample space problem has been studied in pattern recognition extensively in [126]. Dimension reduction methods also are widely available, e.g. principle component analysis (PCA) and its extensions [157]. However, these two problems combined with data imbalance bring us a new challenge. Often, induction rules formed from the small data set are too specific particularly for minority class, which leads to data overfitting. Learning from such a data set is a big challenge, which requires us to have much more sophisticated techniques to address this problem. We use s = 1, 2, … 5 to represent five data size levels, at level s, the total training data size is *round*$((5000/32)*2^{s})$.

3.4 SIMULATED DATA

In order to study the relationships among different data complexities, training sizes, and imbalance levels, we generated a group of data sets varying by complexities, training sizes, and imbalance levels, which is quite similar to the simulated data generated in [68, 69, 83].

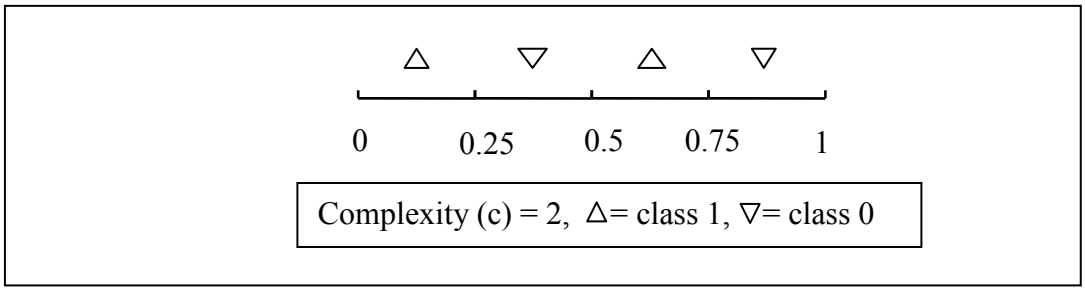


Figure 3-3 A Backbone Model of Complexity 2

Altogether, we generated 125 data sets with various complexities, sizes, and imbalance levels. We considered five different complexity levels (c=1 .. 5), five training sizes (s=1 .. 5), and five imbalance levels (i= 1 .. 5). At training size level s, the training space size will be *round*((5000/32)*2^s). Without considering the imbalance factor, each interval will contain *round*(((5000/32)*2^s)/2^c) data samples. For example, at s=1, c=2, each of the interval has 78 examples. The imbalance level determines the number of samples inside the minority intervals but not affecting the data size in the majority intervals. At imbalance level i, each of the class 0 interval will be represented by *round*((((5000/32)*2^s)/2^c)/(32/2^i)) number of examples. For example, when c=2, s=1, and i=3, intervals [0, 0.25) and [0.5, 0.75) have 78 data samples each; intervals [0.25, 0.5) and [0.75, 1] are represented by 20 examples.

The number of testing examples in each interval is fixed at 50. So the testing space with complexity c =1 has 50 positive samples and 50 negative samples. The testing space with complexity c=2 has 100 positive sample and 100 negative samples.

3.5 RESULTS AND ANALYSIS

The data complexity is increased with value c from 1 to 5, and the training data size is increased with value s from 1 to 5. However, the imbalance level is decreased with the value i from 1 to 5, which means, the data is the most imbalanced at i=1, and is balanced when value i=5. All together, we have 125 training data sets and 125 corresponding testing data sets generated.

We report the results from the Bayesian network classifier as shown in Figure 3-4[1] to Figure 3-8[2]. The evaluation measure is g-Mean [81] which is shown in section 4.4.3 as an effective and efficient method for imbalanced data learning. As shown in Figure 3-4, when the data complexity is low, both training data size and imbalance ratio do not hinder the classifier's performance and the classifier performs well on all cases. From Figure 3-4 to Figure 3-8, when the data is complex, the data imbalance factor plays an important factor when the training data is insufficient. The more imbalanced the data is, the poorer is the performance. However, when there is sufficient training data, as when s=5 for most of the cases, the effect of data imbalance can be neglected. The more complex the data is, the larger training data we need in order to minimize the effect of data imbalance. For example, when complexity = 2, we need at least s = 3 to minimize the data imbalance effect; when complexity = 3, we need at least s = 4 in order to

[1] , [2] The legend in these figures are: s is indicating the training data size from 1 to 5, i is the imbalance level from 1 to 5, c is indicating the data complexity from 1 to 5, and the y axis is the g-Mean value from 0 to 1.

minimize the data imbalance effect. Interestingly, we find that when the data is highly complex at complexity level = 5, and when the data size is small with s = 1, the classifier fails to predict the minorities regardless the imbalance level as shown in Figure 3-8. This is because that there are not enough representative data samples in the highly complex data to support the minority concept.

3.6 DISCUSSION

In this chapter, we have discussed the nature behind the imbalanced data problem. We have looked at different types of imbalances, absolute rarity, relative imbalance, and other factors affecting the imbalanced data problem including data fragmentation, noise and inductive bias. We have also discussed three important factors – data complexity, training data size, and imbalance level. We have shown how they hindered the imbalanced data problem by experimenting on a set of simulated data. Particularly, we have shown that data complexity is another very important factor affecting the imbalanced data problem besides the imbalance level.

We have described the problems brought by the evaluation metrics in imbalanced data learning. In particular, accuracy cannot be used as the performance measure in imbalanced data problems. Therefore, we need to be careful when choosing evaluation metrics in imbalanced data learning problems.

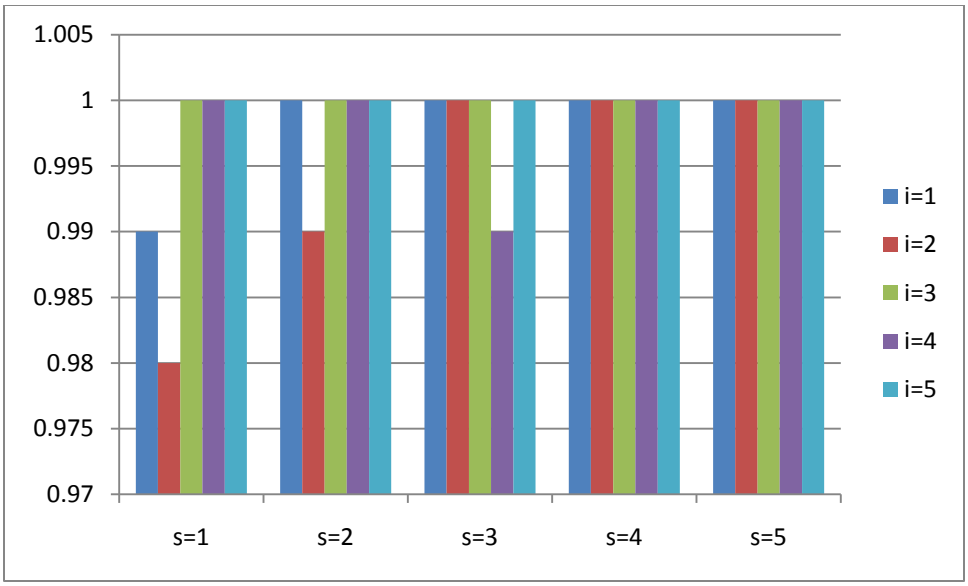


Figure 3-4 Performance of simulated data with complexity level c = 1

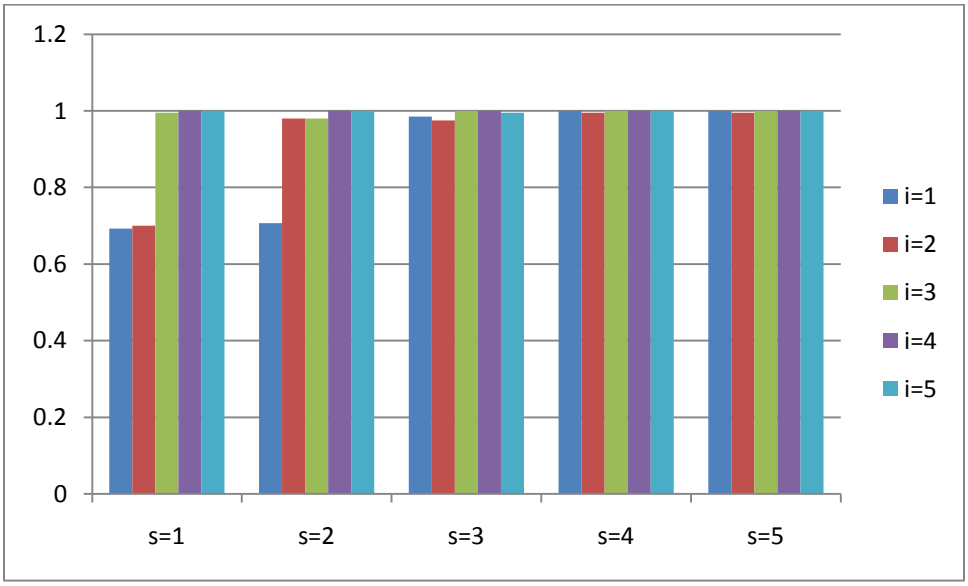


Figure 3-5 Performance of simulated data with complexity level c = 2

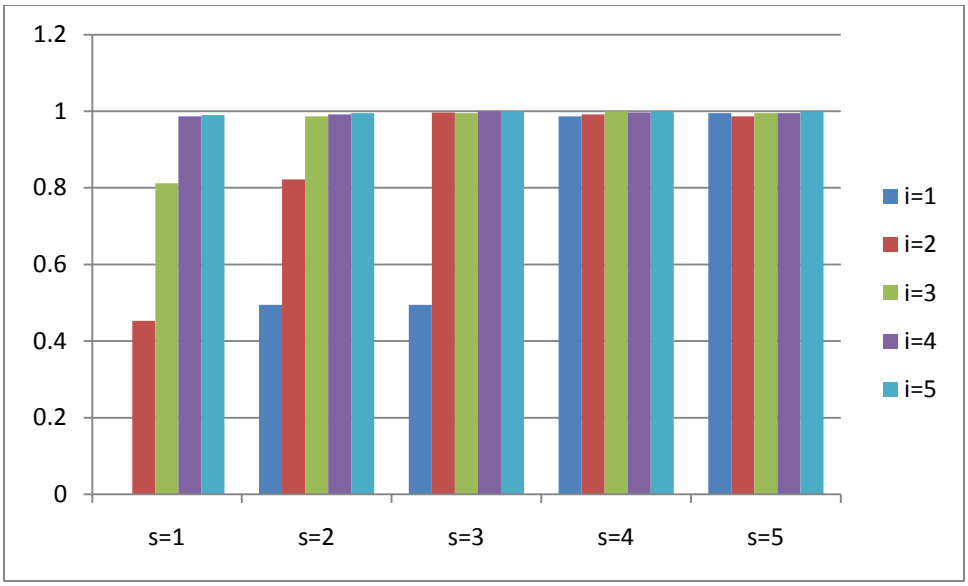


Figure 3-6 Performance of simulated data with complexity level c = 3

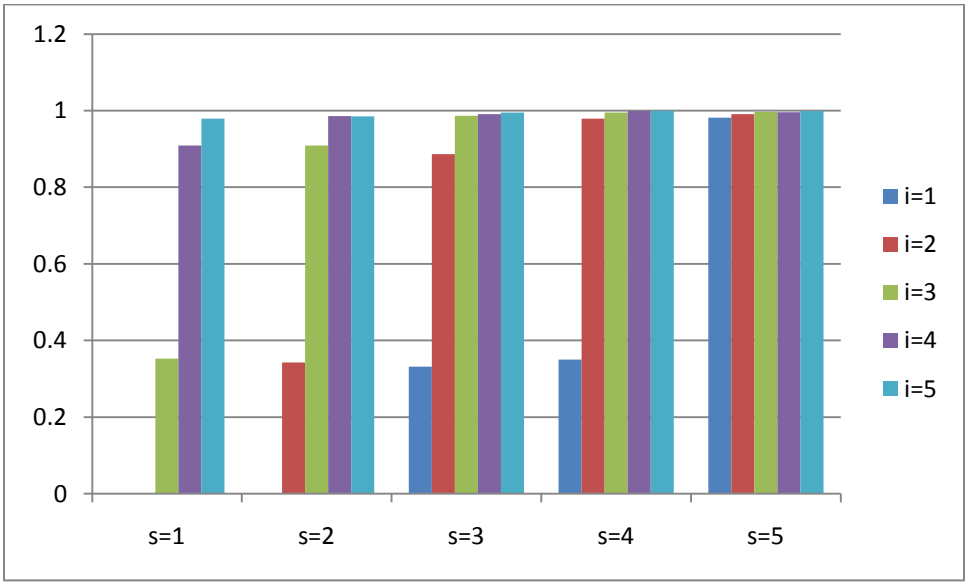


Figure 3-7 Performance of simulated data with complexity level c = 4

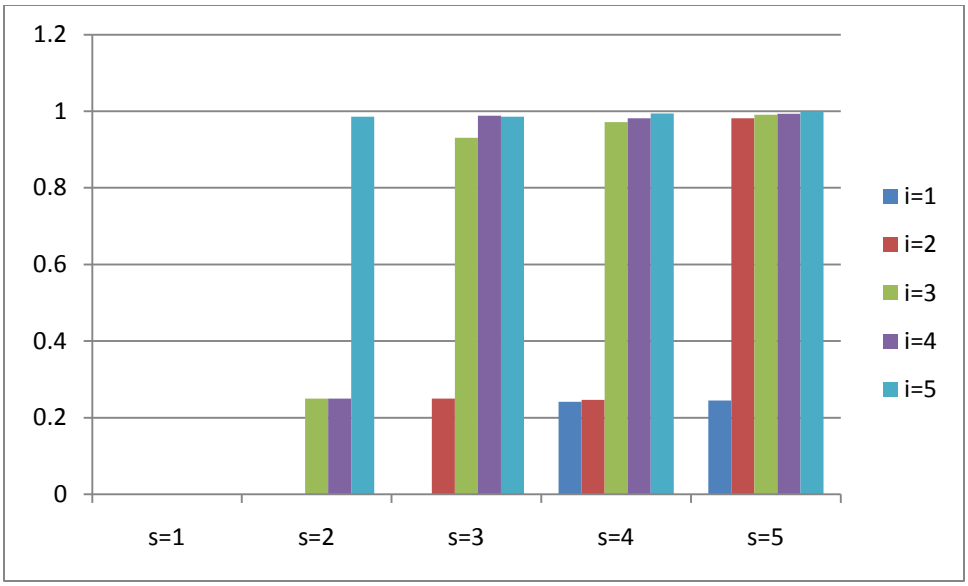


Figure 3-8 Performance of simulated data with complexity level c = 5

CHAPTER 4: LITERATURE REVIEW

4. LITERATURE REVIEW

Generally speaking, there are two major approaches to imbalanced data learning – algorithmic level approaches [27, 42, 67, 74, 76, 82, 127] and data level approaches [22, 23, 35, 81]. Algorithmic level approaches alter the machine learning algorithms to improve the prediction performance in imbalanced data learning. Data level approaches change the training data distributions to achieve performance improvement; they usually refer to the data sampling techniques.

4.1 ALGORITHMIC LEVEL APPROACHES

Algorithmic level approaches [119] [80] include one class learning, cost sensitive learning, adjusting the decision threshold, boosting algorithm, two phase rule induction, and kernel based methods etc.

4.1.1 ONE CLASS LEARNING

If we try to learn classification rules for all classes, the rare classes may be largely ignored. One solution to this problem is to only learn classification rules that predict the rare class. Hippo [67] is one of the systems that utilizes recognition based systems to perform one class learning. Hippo makes use of neural networks and learns only from

positive (rare) examples, rather than to differentiate between positive and negative examples. Hippo employs a two-phase method. In the first phase, a concept is learned from positive examples, and in the second phase, the system learns how to identify positive and negative examples of that concept.

Besides training from one class only, it is also meaningful to train systems with examples belonging to other classes. Brute [64], Shrink [82] and Ripper [100] are three such machine learning systems. The Brute system is used for detecting flaws in the Boeing manufacturing process. Brute focuses only on the rules that predict failures. The advantage of Brute system is that by measuring the performance only for the positive predicting rules; Brute is not influenced by the majority negative examples that are not covered by the positive predicting rules. Shrink uses a similar approach to detect rare oil spills from satellite radar images. Shrink labels regions containing both positive and negative examples with positive class. The task then is to search for the best positive regions which have the highest ratio of positive to negative examples. Ripper is a rule induction system that generates rules for each class from the rarest class to the most common class. Therefore, it is quite straightforward to only learn rules for the minority class.

Only a subset of classification rules of the above systems can be used to choose m of total learned n rules, m <= n. By varying the value m, we then can generate a precision/recall curve, and then a desired solution can be selected based on the requirements of the problem.

Raskutti et al. [125] showed that one class learning is particularly useful for extremely imbalanced data with high dimensional feature space and comparing to feature selection methods, it is less expensive.

4.1.2 COST-SENSITIVE LEARNING

In medicine (or other critical fields), it is often that the minority classes are of primary interest. However, most existing classification algorithms assume that the input features and outcomes have no costs, and the goal is to minimize the total misclassification errors. For example, in medical diagnosis, different outcomes have different costs. The cost of a false medical diagnosis is an unnecessary treatment, but the cost for a false negative diagnosis may cause the death of the patient. So a cost sensitive learning algorithm should prefer to make less costly errors, e.g., false negative diagnosis is preferred in this case. Another example is about Intensive Care Units (ICU) equipments, which are supposed to give an alarm if the patient is in a critical condition. A false alarm is a waste of man power, but a missed alarm may cost a patient's life. So cost sensitive learning algorithms are important in such situations [37] .

Assigning greater cost to false negatives than to false positives will improve learning performance with respect to the positive class. For example, if the misclassification cost ratio is 3:1, then the region which has 10 negative examples and 4 positive examples will be labeled as positive. Most cost sensitive learning approaches incorporate costs into machine learning by defining fixed and unequal costs to different classes [33]. However, the problem with these approaches is that the cost information is

normally hard to determine. This is mainly because the costs often depend on multiple factors that are not easily comparable [25]. For example, the cost of a false positive of the diagnosis leading patients to death or patients' well being is usually hard to be estimated.

Other approaches change the cost indirectly. Cost sensitivity is obtained by changing the ratio of positive and negative samples in the training space, or by adjusting the decision threshold in the assignment of class labels [37].

4.1.3 BOOSTING ALGORITHM

Boosting algorithms are iterative algorithms that place different weights on the training distribution at each iteration. After each iteration, boosting increases the weights associated with incorrectly classified examples and decreases the weights associated with correctly classified examples. This forces the system to focus on the rare items. Thus it is reasonable to believe that boosting may improve rare class prediction because overall it will increase the weights assigned to rare classes.

A cost sensitive version of AdaBoost – Adacost [45], has been empirically shown to produce higher classification rate than AdaBoost [114]. There is even a special AdaBoost algorithm addressing rarity – RareBoost [76]. Rare-Boost scales false positive examples in proportion to how well they are distinguished from true positive examples and scales false negative examples in proportion to how well they are distinguished from true negative examples. Another algorithm that uses boosting to address the problems with rare class is SMOTEBoost [27]. SMOTEBoost addresses the problem of data over fitting, because boosting weight rare examples more heavily by duplicating rare

examples. SMOTEBoost alters the distribution by adding new synthetic examples using the SMOTE algorithm. Empirical results indicate that SMOTEBoost achieve higher F-value than Adacost. Joshi et al. [75] showed that the performance improvement from boosting is strongly related to the base learning algorithm. Boosting will perform poorly if the base learner always achieves poor precision or recall; however, if it can effectively trade-off precision and recall, then boosting can significantly improve the performance of the base learner.

4.1.4 TWO PHASE RULE INDUCTION

In order to achieve balanced prediction accuracy and not to bias to any class, induction techniques that deal with rare classes must try to maximize both precision and recall. Most induction techniques try to optimize both of them which are shown to be too difficult to accomplish for complex problems. Joshi et al. [74] used two-phase rule induction to focus on each measure separately. The first phase focuses on recall, and then in the second phase, precision is optimized which is accomplished by learning to identify false positives within the rules from phase 1. If we use the needle in the hay analogy, this approach identifies regions likely containing needles in the first phase, and then learns to discard the strands of hay within these regions in the second phase. The presence of the second phase permits the first phase to be sensitive to the problem of small disjuncts while the second phase allows the false positives to be grouped together, addressing the problem of data fragmentation. Experimental results indicate that it performs competitively well to other learners, especially when many rare cases are introduced.

4.1.5 KERNEL BASED METHODS

Kernel based methods are widely used in many applications with success. They are also being applied in imbalanced data learning. The principles of kernel based methods are statistical learning and Vapnik-Chervonenkis (VC) dimensions [141]. Support vector machines (SVMs) is a typical kernel based learning method that can provide robust classification results for imbalanced data [69]. Since SVMs try to minimize total error, they are biased towards the majority class. In a binary class, the support vectors for the minority "concept" might be far away from the ideal separation line, and thus contribute less to the final hypothesis [7, 125, 153]. The same thing happens in nonlinear separable spaces. In this case, a kernel function is used to map the non-separable spaces into high dimensional separable spaces. However, doing this can often cause the optimal hyperplane to be biased towards the majority class.

One type of kernel based method integrates kernel methods with sampling methods. Some of the examples include SMOTE with Different Costs methods [7] and the ensembles of sampled SVMs [78, 142]. The Granular Support Vector Machines – Repetitive Undersampling algorithm (GSVM – RU) was proposed by Tang et al. [136] to integrate SVM learning with undersampling methods. These methods develop an ensemble system by modifying the data distributions without modifying the underlying SVM classifier.

Another type of kernel based methods is kernel modification methods which focus on the SVM mechanism itself. One example is the algorithm proposed by Hong et

al. [65] which is based on orthogonal forward selection (OFS) and the regularized orthogonal weighted least squares (ROWLSs) estimator. The algorithm contains two components in dealing with imbalanced data. The first component integrates the leave-one-out cross validation (LOO) and the area under the curve (AUC) evaluation metric to develop an LOO-AUC objective function as a selection mechanism of the most optimal kernel model. The second component makes use of cost sensitivity to assign greater weight to the minority class.

Other kernel based methods mainly focus on adjusting the class boundaries. Some of the methods include, for example, the boundary movement (BM) approach proposed in [153], the kernel-boundary alignment (KBA) approach in [155], an integrated approach – the total margin-based adaptive fuzzy SVM kernel method (TAF-SVM) proposed in [95] and [94], the support cluster machines (SCMS) [161] for large scale imbalanced data learning, and the kernel neural gas (KNG) [121] algorithm for imbalanced clustering etc.

4.1.6 ACTIVE LEARNING

Active learning is mainly used in unsupervised learning problems. Recent approaches have integrated active learning with SVM approaches [40, 41, 120] or data sampling techniques [5, 165] in imbalanced data learning.

Ertekin et al. [40, 41] proposed an efficient SVM-based active learning method. It first trains an SVM on the given training data, and then generates the most informative

training data to retrain the SVM with all unseen training data using LASVM [11] to facilitate the active learning procedure.

Zhu [165] combined active learning with the random sampling method (both under sampling and over sampling) for the word sense disambiguation (WSD) imbalanced learning problem, in which entropy is used as a metric to measure the uncertainty of the training instances.

4.2 DATA LEVEL APPROACHES

Data level approaches include many forms of re-sampling techniques generally categorized into basic data sampling approaches and advanced data sampling approaches or data segmentation. Basic data sampling techniques include random oversampling with replacement, random undersampling, directed oversampling, directed undersampling; Advanced sampling generates synthetic data either using local data (local sampling) or global data (global sampling).

4.2.1 DATA SEGMENTATION

Data segmentation is to carefully partition the original data into different parts, to reduce extreme imbalance in sub data sets. For example, some rare targets constitute 1% in the original data set. By segmenting the data, the rare events occupy 30% in one data set A, and 0.1% in another data set B. We can then mine the rare cases easily from data set A, though it becomes more difficult for data set B. It is acceptable, because most of the rare cases are in data set A. Considering the example in Figure 2-1, where severe head injury

patients are a minority in all head injury patients. In real life, most of the severe head injury patients are elderly. So if we divide the data set into two parts – elderly and non-elderly, we can expect that severe head injury patients will take up a much larger percentage in the elderly group. Thus the problem is simplified. Segmentation can be viewed as an example of how knowledge can be used to address rarity.

4.2.2 BASIC DATA SAMPLING

The most common technique used in dealing with imbalanced data is sampling. The idea of sampling is to artificially adjust the data distribution to reduce the imbalance of the data set.

Random under sampling and random over sampling are two basic sampling methods. Under sampling eliminates majority class examples while over sampling replicates minority class examples. Both of them can reduce the class imbalance and therefore improve the prediction accuracy for imbalanced data. However, there are also drawbacks in the sampling methods. Under sampling can possibly remove useful information from the majority data set. On the other hand, over sampling produces more training data, and thus make the system inefficient. Since over sampling normally replicates exact copies of the minority training cases, it is very easy to lead to data over fitting [26, 35]. Over sampling does not produce new training data, so some of the research shows that it is not as effective as under sampling [35].

4.2.3 ADVANCED SAMPLING

Unlike basic data sampling, advanced sampling uses intelligence to make smarter decisions; such sampling methods can combine over sampling and under sampling or they can generate new synthetic data samples.

4.2.3.1 Local sampling

Local sampling refers to data sampling based on the data sample itself or based on a limited local region near the data sample in the training space. As shown in Figure 4-1, Local sampling algorithms make use of a limited amount of information to generate data samples, thus they are very efficient. The drawback is that local sampling may lead to local maxima or even generate false positive samples because of insufficient and limited knowledge that can be learned from the local neighborhood. For instance, in Figure 4-1, the decision of local sampling for instance A will be based on instance A itself or A's nearest neighbor like instance B. Instance A may get duplicated in over sampling, or instance A may be removed in under sampling, or synthetic samples may be generated along the line between A and B in Synthetic Minority Over-sampling Technique (SMOTE) [23].

Most of the existing sampling approaches belong to the local sampling group, including one sided selection [81], Synthetic Minority Over-sampling Technique (SMOTE) [23], sampling according to a designed distribution [22, 148], and a mixture of experts method which combines different sampling approaches [43].

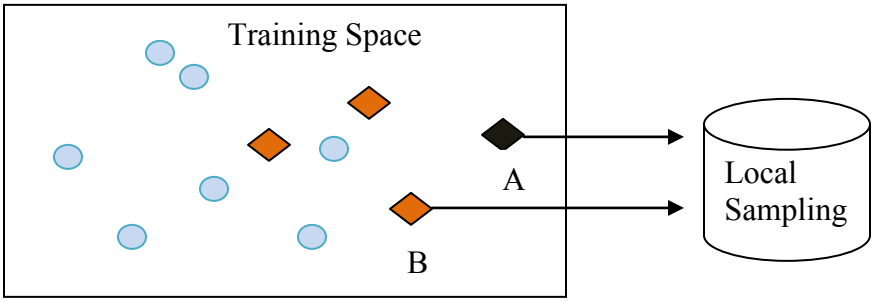


Figure 4-1 Local sampling with instance A

4.2.3.1.1 One sided selection

One sided selection [81] is an under sampling strategy which only removes majority examples that are duplicates of existing examples, or border regions that may be noises.

4.2.3.1.2 SMOTE sampling

SMOTE [23] does not duplicate existing examples, instead it creates new examples. It creates new samples by random sampling from the segments which join the *k* nearest neighbors from the minority class example. This may cause over generalization problem instead of specialization by simply replicating existing examples.

SMOTE operates in "feature space" rather than "data space". The minority class is over-sampled by taking each minority class sample and introducing synthetic examples along the line segments joining any/all of the *k* nearest neighbors. Depending on the amount of over-sampling required, neighbors from the *k* nearest neighbors are randomly chosen. For instance, if the amount of over-sampling needed is 200%, only two neighbors from the five nearest neighbors are chosen and one sample is generated in the direction of each. Synthetic samples are generated in the following way: Take the difference between the

feature vector (sample) under consideration and its nearest neighbor. Multiply this difference by a random number between 0 and 1, and add it to the feature vector under consideration. This causes the selection of a random point along the line segment between two specific features (as shown in Figure 4-2). This approach effectively forces the decision region of the minority class to become more general.

Recent developments on SMOTE approach include SMOTEBoost [27] as mentioned in section 4.1.3 and Borderline-SMOTE [58] in which only the minority examples near the borderline are over sampled. However, SMOTE's procedure is inherently "dangerous" since it blindly generalizes the minority area without regard to the majority class. For example as shown in Figure 4-3, if there is a majority example lying between the two nearest neighbors, the synthetic minority sample generated might coincide with the majority sample and cause noises.

This strategy is particularly problematic in the case of highly skewed class distributions, since in such cases the minority class is very sparse with respect to the majority class, thus resulting in a greater chance of class mixture as shown in Figure 4-4.

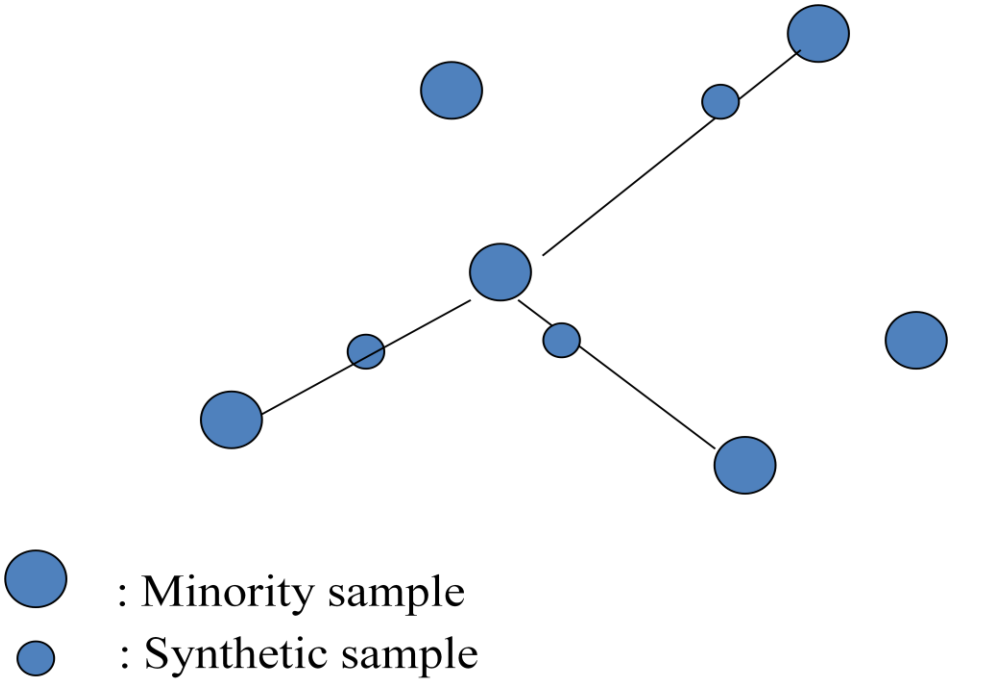


Figure 4-2 Synthetic samples generated by SMOTE

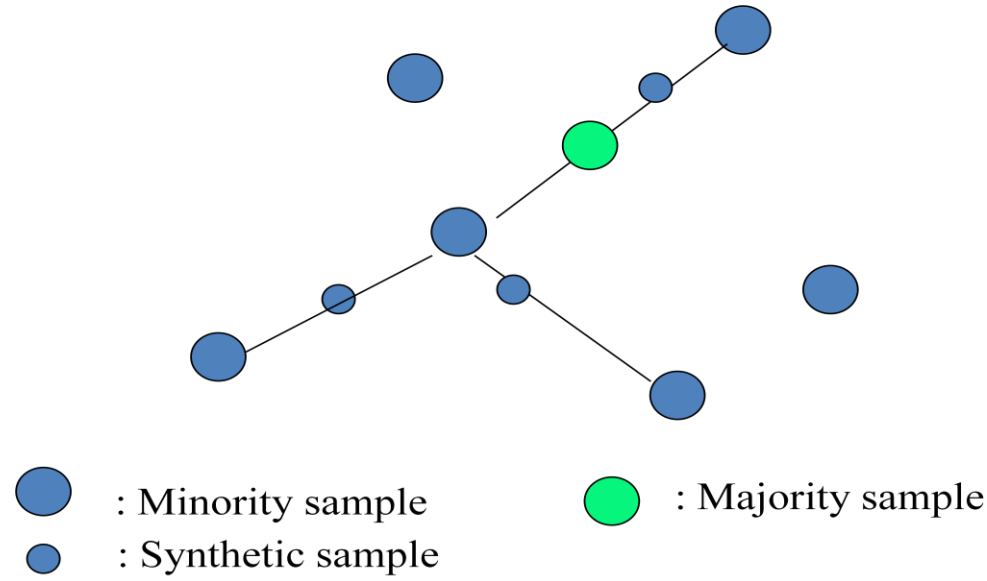


Figure 4-3 Over generalization caused by SMOTE

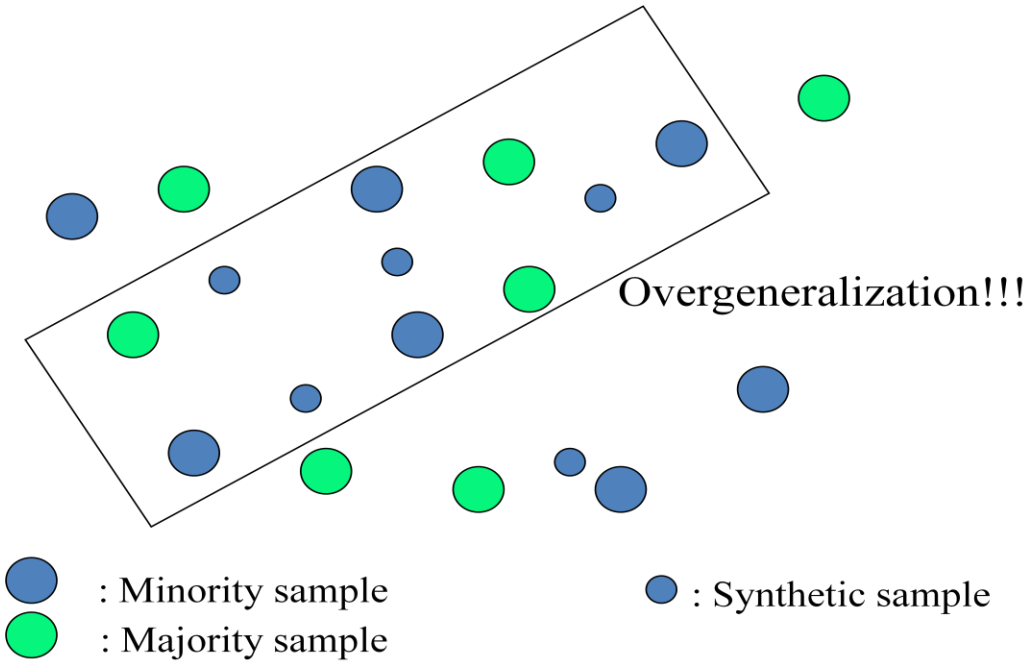


Figure 4-4 Data over-generalization caused by SMOTE

4.2.3.1.3 Class distribution based methods

Another approach is to identify a good class distribution first, and then generate dataset with that distribution. Chan et al [22] identified a good class distribution by testing on a set of preliminary experiments, and then generate a group of training sets with the desired distributions. Each training set includes all the minority examples and a subset of the majority examples, each majority example is guaranteed to appear in one of the training sets. The learning algorithm is applied for each of the training sets, and then a composite learner is formed from the resulted classifiers. This approach can be used for any learning algorithms. This ensemble approach is empirically shown to be effective for dealing with rare classes. Yan et al [156] showed that a resulting SVM ensemble outperforms both under sampling and over sampling. A similar approach is proposed by Weiss & Provost

[148]. It employs a progressive-sampling algorithm to build larger and larger training sets, where the ratio of positive examples to negative examples added is based on the best performance distribution in previous iteration. Experimental results show that it generally converges to a nearly optimal value for learning. This approach is based on the assumption that not all examples are immediately available for learning, rather there is cost associated with procuring each example. This is contrasting with other sampling algorithms which assume that there are already a collection of training examples without cost.

4.2.3.1.4 A mixture of experts method

A mixture of experts [43] method has been used to combine the results of many classifiers, each induced after over sampling or under sampling the data with different rates. This approach is based on an assumption that – we are not clear which sampling strategy is better, and we do not know which sampling rate should be applied to our sampling method or dataset. Generally, the mixture-of-experts method performs well, and does especially well in rare examples.

4.2.3.1.5 Summary

Local sampling algorithms make use of a limited amount of information to generate data samples, thus they are very efficient. The drawback is that local sampling may lead to local maxima or even generate false positive samples because of insufficient and limited knowledge that can be learned from the local neighborhood.

4.2.3.2 Global sampling

Global sampling refers to data sampling based on the whole training space. Comparing to local sampling, global sampling for instance A is based on all instances in the training space, instead of only A's neighbors. This scenario is shown in Figure 4-5. Global sampling is a relatively new in imbalanced data learning. One of the most representative work is generative oversampling proposed by Liu et al. [91].

Generative oversampling creates completely new, artificial data points via a chosen probability distribution. Generative oversampling can be used in any domain where exists a probability distribution of the data set. It works as following: Firstly, a probability distribution is chosen to model the minority class; then, based on the training data, parameters for the probability distribution are learned; finally, artificial data points are generated from the learned probability distribution until the desired data balance is achieved.

4.2.3.3 Progressive sampling

Progressive sampling was first proposed and thoroughly described by Foster et al. [50]. Its original objective was to maximize the system performance with minimal training data. Progressive sampling was later used in [104, 147] for imbalanced data learning. Since it can be used in both local sampling and global sampling, we categorize it into a third type of sampling approach.

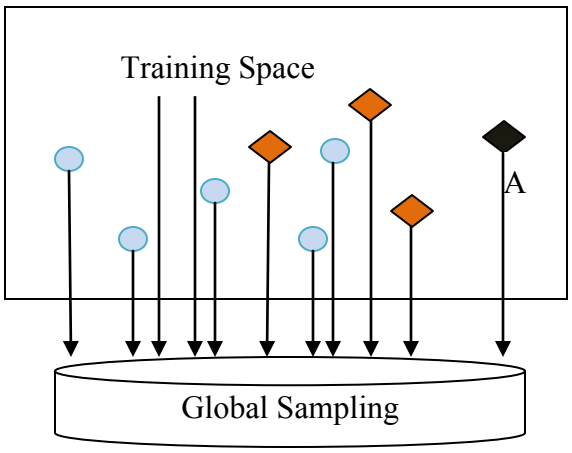


Figure 4-5 Global sampling with all data samples

The idea behind progressive sampling is simple. It starts with a small sample and uses progressively larger ones until the model built from the data cannot improve the overall accuracy any more. The central component is the sampling schedule S = [n_0, n_1, n_2... n_k] where each n_i is an integer that specifies the size of a sample to be provided to an induction algorithm. For $i<j$, $n_i < n_j$. If the data set contains N instances in total, $n_i \leq N$ for all i. The commonly used schedule is geometric sampling schedule $S_g = a^i \cdot n_0 = [n_0, a \cdot n_0, a^2 \cdot n_0 ..., a^k \cdot n_0]$. For example, when a=2, n_0=100, S_g= [100, 200, 400, 800 …]. Geometric sampling is an asymptotic optimal schedule [50].

Weiss [147] proposed a budget sensitive progressive sampling strategy for imbalanced data learning. Budget sensitive sampling assumes that the cost associated with forming the training set may be limited by budget B and the cost of executing the algorithm is negligible compared to the cost of procuring examples. It begins with a small amount of training data and progressively adds training examples using a geometric sampling schedule. The important step is to find the distribution that gives the best

performance, and Weiss analyzed the effect of class distribution to the algorithm performance. The amount of minority or majority data added is determined by the current distribution that performs the best. All examples from the current iteration will be used in the next iteration. Budget sensitive progressive sampling may not always give the best performance, but it can generally give a near optimal performance.

Willie [104] proposed a progressive sampling with over sampling (PSOS) approach. PSOS always maintains a balanced class distribution throughout the sampling schedule. In PSOS, training examples are sampled separately from minority and majority examples, and random replication will start when minority exhausts. Willie shows that PSOS outperforms progressive sampling.

One important assumption in progressive sampling is that the available training data is potentially large. However, this assumption is not true in this thesis. We are addressing the imbalanced data problems which are not only imbalanced but also with limited training data. Many problems are suffering from the lack of training data particularly the lack of minority data in reality.

4.3 OTHER APPROACHES

Besides algorithmic level approaches and data level approaches, there are also other approaches that are not well categorized into either of the approaches. We discuss them here to end the review of the existing approaches.

4.3.1.1 Place rare cases into separate classes

Rare cases in the imbalanced data set make machine learning difficult because there is often very little in common among them and it is hard to assign the same class label to various rare cases. Japkowicz [71] proposed an approach that viewed rare cases as separate classes. Firstly, each class is separated into subclasses using clustering method; and then the training examples are re-labeled based on the clusters from the first step; lastly, the model is re-learned from the revised training data. The performance of this approach is promising, but further research is needed.

4.3.1.2 Using domain knowledge

Correct domain knowledge is always helpful in improving the machine learning performance, and this is especially true for the rare data. Domain knowledge can provide better understanding of the training data, for instance, domain knowledge can provide a more meaningful feature set or a valid model structure in Bayesian network.

Machine learning is an interactive process, and the domain experts' opinion is very important. This is especially true for the mining of rare data, because domain knowledge can help in the searching process. This is supported by the quote "only in rare cases will users wish to see patterns with miniscule support. In those cases it is more likely that users will start the mining on the small filtered sample (which may be the result of a previous drill-down operation)." [79].

4.3.1.3 Additional methods

Non-greedy search techniques such as genetic algorithms can be used for imbalanced data learning. Weiss [145] makes use of genetic algorithm to predict very rare events while Carvalho et. al [21] uses genetic algorithm to discover "small disjuncts rules".

The Mahalanobis-Taguchi System (MTS) has also been used for imbalanced data learning [131]. Since learning in MTS is performed by developing a continuous measurement scale using single class example instead of the whole training space, it is less influenced by the data imbalance and provides robust classification performance. It was shown in [131] that MTS outperforms the rest of the approaches such as decision trees and SVM etc.

Another approach is to combine the imbalanced data and small disjunct problem [20]. Rank metrics are used as the evaluation metrics for model selection instead of accuracy. Rank metrics emphasize in distinguishing classes instead of the data internal structure such as feature space conjunctions. Therefore, it can help the learning from imbalanced data and small disjuncts with high dimensions. The other approach proposed in [20] is based on multi task learning methodology. A shared representation of the data is used to train the extra task model related to the main task. Therefore, learning of the minority data is amplified by adding extra information to the data.

Besides the above existing approaches on binary class imbalanced data problem, there are also approaches on multi-class imbalanced data problem. For example, Sun et al. [132] proposed a cost sensitive boosting algorithm AdaC2.M1 to tackle the

imbalanced data problem with multiple classes. Chen et al. [29] proposed a min-max modular network to decompose the multiclass imbalanced problem into multiple binary class subproblems. Other approaches include the rescaling approach for multiclass cost sensitive neural networks [163, 164], the ensemble knowledge for imbalance sample sets (eKISS) method [135] and others.

4.4 PERFORMANCE EVALUATION MEASURES

As it is already shown in the literature and chapter 2, accuracy and a lot of other evaluation metrics are not suitable for imbalanced data learning. Therefore, proper evaluation metrics need to be selected for imbalanced data learning. We review the major evaluation metrics and list the characteristics for each of them.

We use the following definitions and abbreviations to ease the descriptions in this section. As shown in Table 4-1, True Positive (TP) is the number of true samples that are correctly classified to be positive; false positive (FP) is the number of false samples that are incorrectly classified to be positive; false negative (FN) is the number of true examples that are incorrectly classified as negative samples; true negative (TN) is the number of samples that correctly classified to be negative. Sensitivity = TP/(TP+FN), it measures the ability of a classifier that can identify true samples correctly, and in information retrieval, this value is named as “recall”; Specificity=TN/(TN+FP), it measures the ability of a classifier that can correctly identify true negative samples.

Table 4-1 Performance Evaluation Metrics

		Condition (e.g., disease) As determined by *Gold* standard		
		True	**False**	
Test outcome	**Positive**	**True positive**	**False positive**	→ Positive predictive value
	Negative	**False negative**	**True negative**	→ Negative predictive value
		↓ Sensitivity	↓ Specificity	Accuracy

4.4.1 ACCURACY

Accuracy is commonly used in machine learning research. It is defined as the percentage of samples that are correctly predicted among the total samples in the training space as shown in Equation 4-1. However, accuracy is not suitable to be used as the performance evaluation measure for imbalanced data learning. Considering the example with 98 false samples and 2 true samples, a default classifier that classifies everything as negative can achieve a high accuracy of 98 percent. But this accuracy is seriously biased to the majority class, it totally misses the positive samples. So in imbalanced data learning, accuracy is not a proper evaluation metric [27, 57, 97, 149].

$$Accuracy = (TP+TN)/(TP+TN+FP+FN)$$

Equation 4-1

4.4.2 F-MEASURE

Other evaluation metrics which are frequently used in machine learning community and suitable for imbalanced data learning are *precision, recall* [110], and *F-measure* [140] as defined in Equation 4-2. Precision is a measure of exactness, which is equal to the

percentage of correctly labeled positive examples among those examples being labeled as positive. Recall measures the completeness, which is equal to the percentage of correctly labeled positive examples among actually positive examples. When used properly, precision and recall combined (For instance, *F-measure*) can be used to evaluate imbalanced data learners. However, *F-measure* remains sensitive to data distributions.

$$Precision = TP/(TP+FP);\ Recall = TP/(TP+FN)$$

$$F\text{-}Measure = 2 \times Precision \times Recall\ /(Precision + Recall)$$

Equation 4-2

4.4.3 G-MEAN

The G-Mean metric evaluates the degree of inductive bias using the square root of true positive rate and true negative rate. Kubat et al [81] uses the geometric mean of the accuracies measured separately on each class as shown in Equation 4-3. a^+ is true positive rate which is equal to TP/(TP+FN) (sensitivity); a^- is true negative rate which is defined as TN/(TN+FP) (specificity).

The basic idea behind this measure is to maximize the accuracy on both classes. In this study the geometric mean will be used as a check to see how balanced the combination scheme is. For example, if we consider an imbalanced data set that has 240 positive examples and 6000 negative examples and stubbornly classify each example as negative, we could see, as in many imbalanced domains, a very high accuracy (acc =

96%). Using the geometric mean, however, would quickly show that this line of thinking is flawed. It would be calculated as g=sqrt(0 × 1) = 0.

$$a^+ = TP/(TP+FN);\ a^- = TN/(TN+FP)$$

$$g = \sqrt{a^+ \times a^-}$$

Equation 4-3 g-Mean

4.4.4 ROC CURVES

G-Mean is an effective evaluation metric for imbalanced data learning for a certain threshold that evaluates the best performance. On the other hand, *ROC curves* (Receiving Operator Characteristic) [47, 48] provide a visual representation of the tradeoff between true positives and false positives. They are plots of true positive rate or the percentage of correctly classified positive examples a^+ or sensitivity with respect to false positive rate or the percentage of incorrectly classified negative examples 1-a^- or 1-specificity. ROC curves can give the comparisons among different classifiers over a set of continuous threshold points.

As shown in Figure 4-6, the point (0, 0) along a curve would represent a classifier that by default classifies all examples as being negative, whereas a point (0, 100) represents a classifier that correctly classifies all examples.

Many learning algorithms allow induced classifiers to move along the curve by varying their learning parameters. For example, decision tree learning algorithms provide

options allowing induced classifiers to move along the curve by way of pruning parameters. Stiell et al. [134] proposed that classifiers' performances can be compared by calculating the area under the curves generated by the algorithms on identical data sets. In Figure 4-6, the learner associated with Series 1 would be considered superior to the algorithm that generated Series 2.

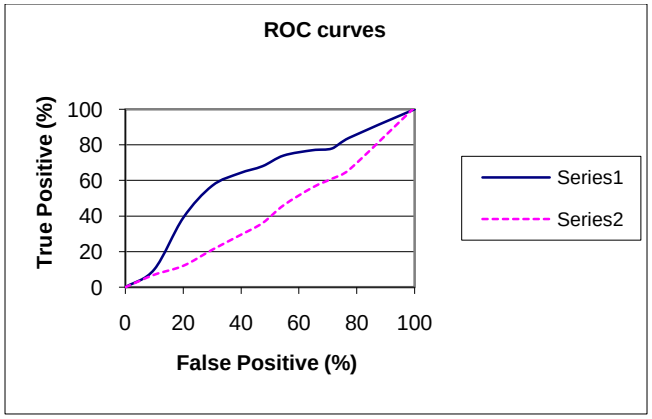


Figure 4-6 an example of ROC curves

4.5 DISCUSSION AND ANALYSIS

4.5.1 MAPPING OF IMBALANCED PROBLEMS TO SOLUTIONS

In Table 4-2, we have summarized the solutions to respective imbalanced data learning problems. For each problem, there are multiple solutions available; we then provide the most direct solutions.

There is no specific order for the methods listed in each cell, they are somewhat arbitrary. In particular, for problems with absolute rarity, there are also problems with relative rarity. Sampling is the only method that directly addresses absolute rarity by

duplicating rare examples, synthetically generating new rare examples, or procuring new rare examples. For the relative rarity, sampling is instead used to rebalance the data distribution to reduce the between-class and within-class imbalances.

We have described many methods so far for dealing with imbalanced data. One important question is which method has the most promising result in dealing with imbalanced data learning. There is no empirical study on comparing all the above methods yet. Most research compared their methods to the base learning that has no special modification for handling imbalanced dataset. Sampling techniques are used in most of research algorithms, but yet the conclusions induced are not consistent. We can discuss their drawbacks, advantages or even some misconceptions in some of these methods.

Table 4-2 Mapping of imbalanced problems to solutions

Imbalanced data problem	**Methods to address the problem**
Improper evaluation metrics	More appropriate evaluation metrics
Absolute rarity	Over-sampling The others are chosen from the cell below
Relative rarity	1. Segmenting the data 2. Boosting 3. Cost sensitive learning 4. Two phase rule induction 5. More appropriate evaluation metrics.
Data fragmentation	1. Non-greedy search techniques 2. Learn only the rare cases
Noise	Advanced Sampling

4.5.2 RARE CASES VS RARE CLASSES

Both rare cases (within class imbalance) and rare classes (between class imbalance) are problematic for machine learning. This section begins by describing the connection between rare cases and rare classes and then shows that both forms of rarity cause similar problems for data mining. An empirical study [148] showed that within 18 dataset with class distribution 2:1, only in two cases does the majority class have a smaller average disjunct size than the minority class. So in general, rare classes tend to have a higher proportion of rare cases than common cases and between class and within class imbalances are linked. We expect that when between-class imbalance is reduced, then within-class imbalance will also be reduced.

Both rare classes and rare cases are similar phenomena, and affect data mining in a similar way. Thus they share the same set of solutions. Among the problems listed in section 3.1 and summarized in Table 4-2, all apply equally to rare classes and rare cases. For example, data fragmentation can be a problem for rare classes, because the examples belonging to rare classes can become separated, or examples belong to rare cases can be separated. Thus both rare class and rare cases are the same fundamental problems. This is not surprising, since a rare case can be viewed as a rare class, as shown by a method proposed by [71] which places rare cases into separate classes.

Next we show the methods for addressing rarity. Many methods (e.g. changing evaluation metrics, non-greedy search techniques, sampling, two phase rule induction etc)

can also be equally applied to both rare cases and rare classes. Data segmentation can also be equally applied to rare cases and rare classes, though sometimes it is harder to segment rare cases. Cost sensitive learning is mostly used in rare classes, because misclassification costs are normally assigned based on the characteristics of examples that are not easily identified for rare cases.

The discussion shows that both rare classes and rare cases suffer from the similar problems, and share most of the solutions. Although some of algorithms are mostly used in rare classes, they could be applied to rare cases if the rare cases can be easily identified. However, we are mainly focusing on rare class problems in this dissertation.

4.6 LIMITATIONS OF THE EXISTING WORK

Many of the methods for addressing rarity are still in the research stage or are not widely implemented (for example, two-phase rule induction) or they are widely available but the advantage for addressing rarity is not proven yet (e.g. boosting algorithms). Some of the algorithms are domain specific (e.g. data segmentation) and thus cannot be universally applied. In this section, we will specially discuss the limitations on sampling and cost sensitive learning which are the two most commonly used techniques for learning with imbalanced data.

4.6.1 SAMPLING AND OTHER METHODS

Breiman [16] showed that sampling is equivalent to other methods in dealing with imbalanced data. For example, one can make false negative twice as costly as false

positive by using cost sensitive learning or by increasing the positive training example size to a factor of two. But in practice, this is not true. One reason is that imbalanced data learning algorithms are usually task and method specific as shown by many empirical studies. Another reason is that it is impossible to have the complete freedom to vary all kinds of quantities. For example, suppose we have a training set with total 1000 cases and a class distribution of 10:1, so there are only 100 positive examples. If we use cost sensitive learning method, we can impose a cost for false negative which is 10 times of false positive. In theory, this is equivalent to using a balanced data set. However, it is generally impossible to generate a perfectly balanced dataset using sampling method. In practice sampling can discard majority class examples (under sampling) or duplicate minority examples (over sampling), or use some combination of both. As discussed in section 4.2.1, such sampling methods bring problems. They may discard useful information or lead to data over fitting.

Another issue is that the effect of sampling on rare class is not fully understood. Sampling normally will cause bias in favor of rare class prediction. The intent in sampling is to create more data for rare class, not to bias machine learning algorithm towards them. The bias is normally caused by duplicated data generated by over sampling methods. A good sampling technique should be able to generate useful new information and approximate the true data distribution.

4.6.2 SAMPLING AND CLASS DISTRIBUTION

Based on the previous discussion, one should use all available data to avoid information loss. If the cost information is known, then cost-sensitive learning algorithm should be used. But normally cost information is not known, one option is to use cost sensitive learning and vary the cost values to improve the performance on rare class at the expense of majority class. The performance of this model is then really dependent on how important the minority class is.

If the training data size is limited because of tractability issues or the training data is costly, then sampling must be used. Ideally, the relative sampling rate between classes should be chosen so that the generated distribution provides the best results. Unfortunately, as shown in [148], there is no general answer to which class distribution will perform the best, and the answer is surely domain and method dependent. A better approach is to determine the class distribution once the method and the domain are given.

4.7 SUMMARY

In this chapter, we have reviewed the imbalanced data learning techniques and evaluation measures. We have discussed two types of imbalanced data learning techniques – algorithmic level approaches and data level approaches. In algorithmic level approaches, we have discussed one class learning, cost sensitive learning, two phase rule induction, boosting algorithm, kernel based methods and active learning. In data level approaches,

we focus on data sampling techniques. We discussed random sampling, local sampling (SMOTE sampling), global sampling and progressive sampling methods. We also analyzed the advantages and limitations of existing well known approaches.

We will propose a new approach – Model Driven Sampling approach (MDS) in the next chapter. We are going to evaluate the typical approaches - random sampling, synthetic minority over sampling technique (SMOTE) and our approach - model driven sampling (MDS) on artificial data and real life data sets, and demonstrate the advantages and the limitations of various techniques in the following chapters. We prefer to use g-Mean as the evaluation metric in this thesis, because g-Mean is efficient and effective for conducting large cohort comparisons. G-Mean is a special threshold point on ROC curve which maximizes both true positive rate and true negative rate. Though other meaningful threshold points can be chosen or Area Under the Curve can be used as possible evaluation metrics.

We will not include SMOTE extensions such as SMOTEBoost [27] and Borderline-SMOTE [58] for comparison, because they do not show obvious advantages over the SMOTE algorithm in general, instead they have only been proven to work in special scenarios or datasets, and they are not proven to outperform SMOTE approach.

CHAPTER 5: A MODEL DRIVEN SAMPLING APPROACH

5. A MODEL DRIVEN SAMPLING APPROACH

In many biomedical data sets, minority data is sparse, and local sampling for minorities often can lead to local maxima or incur data noise. There are two ways that can address this problem – one way is to use global sampling to prevent local maxima, and the other way is to use domain knowledge to guide data sampling. Model driven sampling (MDS) is an approach that combines the above two ways by learning from the whole data set and the domain knowledge to form a concrete model to generate new data samples for imbalanced data sampling.

5.1 MOTIVATION

Consider the example in Figure 5-1, the data samples are in the two dimensional space. From the data samples in the left part of Figure a, existing sampling approaches will often lead to a smooth curve model as shown in the right part of Figure a. However if we have the knowledge of the gradient at each data sample as shown in bottom left, we may derive the correct model as shown in the right part of Figure b, which is quite different from the model derived by local data sampling. It is obvious that the gradient knowledge

cannot be derived from the three local data samples. However, this knowledge can be obtained by learning from the global data samples, or this information can be obtained from domain knowledge. Model driven sampling is a new data sampling approach that can generate data from a model built from global data and domain knowledge.

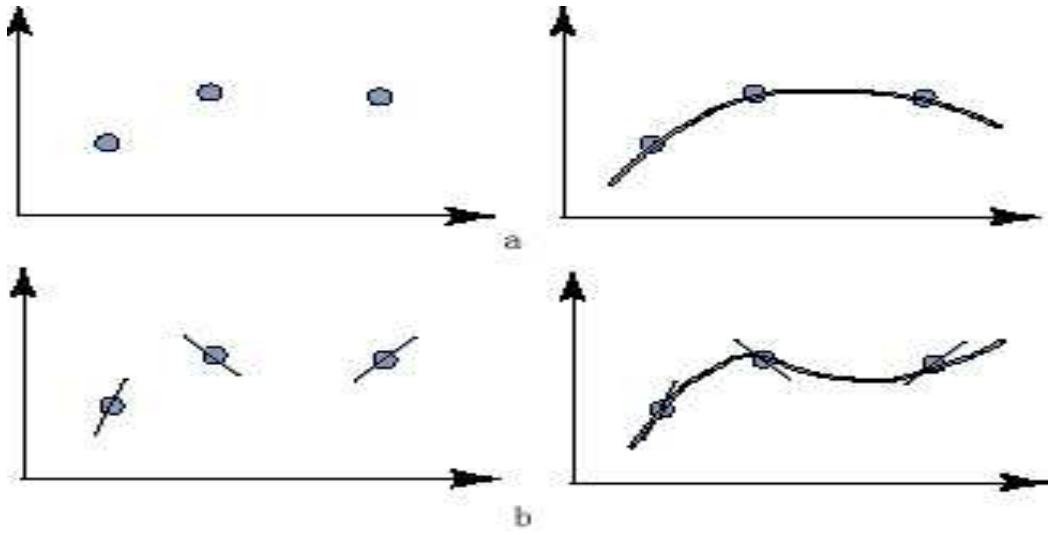


Figure 5-1 Domain knowledge in building a model

In model driven sampling, we can use any probability distribution to model the training data. We choose to use Bayesian network – a probabilistic graphical network, to model the training data set. Bayesian network is an effective methodology in machine leaning, and more importantly it can easily combine expert knowledge into the learned model. Bayesian network uses probabilistic graphical network to model the training data and domain knowledge, therefore, the data sampled have a stronger knowledge base and are more meaningful than data sampled from other sampling approaches.

5.2 ABOUT BAYESIAN NETWORK

A Bayesian Network (BN) is a directed acyclic graph, with its nodes representing uncertain variables, and edges representing conditional probabilistic dependencies between its two connected variables. (Details about Bayesian networks including structure learning, parameter learning, context sensitiveness, and sampling methods are summarized in Appendix C.)

5.2.1 BASICS ABOUT BAYESIAN NETWORK

BN was introduced to Artificial Intelligence more than 20 years ago [96, 115]. It is based on probability theories, with a strong ability in modeling uncertainties in real world problems. From 1990s, scientist began to apply the BN formalism to medical domains, and gradually BN researchers formed a separate community in medical computing, generating quite a number of new ways and new ideas in addressing complex medical problems. Bayesian Network is a factored representation of a probability distribution, representing the probabilistic relationships among a set of random variables as shown in Equation 5-1. The joint probability density function can be written as a product of the individual density functions, conditional on their parent variables, where pa(v) is the set of parents of v (i.e. those vertices pointing directly to v via a single edge).

As shown in the commonly cited Asia network example in Figure 5-2, a Bayesian network consists of the following elements:

(i) A network structure, consisting of nodes and links which represent mostly causal relationships among the nodes. This forms the qualitative layer of Bayesian network.

(ii) The conditional probability table at each node which captures the probabilities of the outcome values conditional on different configurations of the node's parent variables. This forms the quantitative layer of Bayesian network.

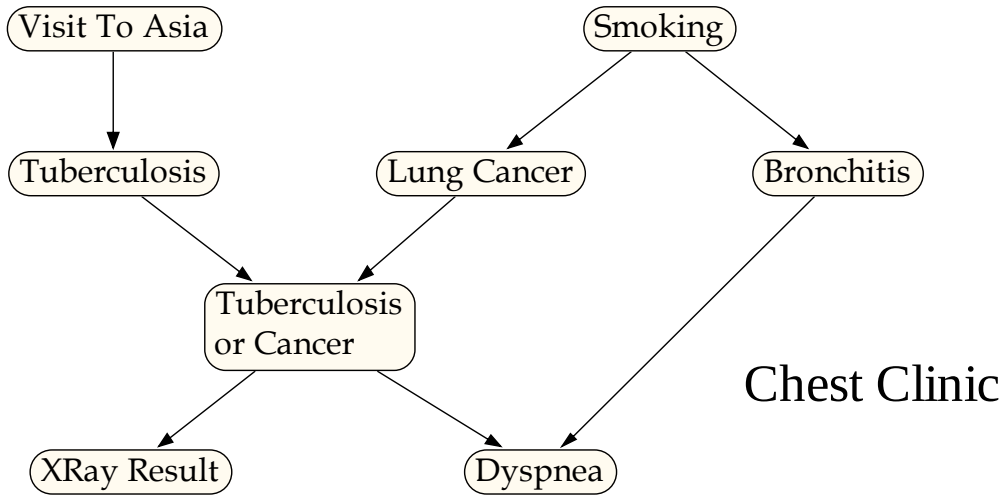


Figure 5-2 The visit-to-Asia Bayesian Network

$$p(x) = \prod_{v \in V} p(x_v | x_{pa(v)})$$

Equation 5-1 Factorization equation

A critical feature of Bayesian Network is that all the uncertainties in the network structure can be represented by conditional probabilities. In any real world problems, there are a lot of uncertain factors. In medical decision making, for example, there are different sources of uncertainty that need to be understood and quantified. This may

include various sampling in experimentation, imperfect expert knowledge, and different results across different studies. Statistically, all these uncertainties can be modeled by probabilities.

The Bayesian network can involve relationships and influences among nodes which can allow researchers to manually specify dependences and independences of variables into the network structure. Bayesian network itself is based on probability theory, and thus it can easily combine domain knowledge and machine learning together.

5.2.2 ADVANTAGES OF BAYESIAN NETWORK

There are many advantages for using Bayesian networks. As a Bayesian network models the probability distributions for a certain problem domain, it can be used to predict the probability distribution for the outcome given a set of evidences. An extension of Bayesian networks – Influence Diagrams use decision theory for risk analysis to choose the solution that can maximize the expected utility. It can be shown that in a very natural sense, this is the optimal procedure for making decisions. Some other very important properties are summarized in the following paragraphs.

Consistency Bayesian network is consistent in processing uncertainties. Probability theories provide a consistent calculus in uncertainty inferencing. Given the same input, a Bayesian network can produce exactly the same answer with different mechanisms in theory.

Smoothness Bayesian network is robust. The performance will not be affected much by small alterations. Therefore, maintaining and updating of Bayesian network models can

be done smoothly. This property is particularly important for complex systems that require a lot of time to re-model.

Expert knowledge One very important property of Bayesian network is that it can code expert knowledge as its prior distribution. This property practically allows Bayesian network to combine expert knowledge with statistical data. Domain experts thus can easily give their contributions by estimating the prior distribution of the Bayesian network, or by changing the structure of the Bayesian network.

Clear Interpretation Bayesian networks have clear interpretations of its structures and parameters. This is different comparing to other techniques, e.g. neural network models acting like a "black box". Bayesian network can be constructed purely using expert knowledge without learning from data. On the other hand, if we have a Bayesian network learned from data, it can be understood by domain experts.

5.3 MODEL DRIVEN SAMPLING

The model we build is a Bayesian network model containing a Bayesian network structure B*s* and the conditional probability distributions B*p*. B*s* models the training data qualitatively, and B*p* models the data quantitatively.

5.3.1 WORK FLOW OF MODEL DRIVEN SAMPLING

The core part in the MDS approach is to build an accurate model. There are three ways to build a model - 1) building model from data, 2) building model from domain knowledge,

3) building model from both data and domain knowledge. The workflow of model driven sampling classification is described in Figure 5-3.

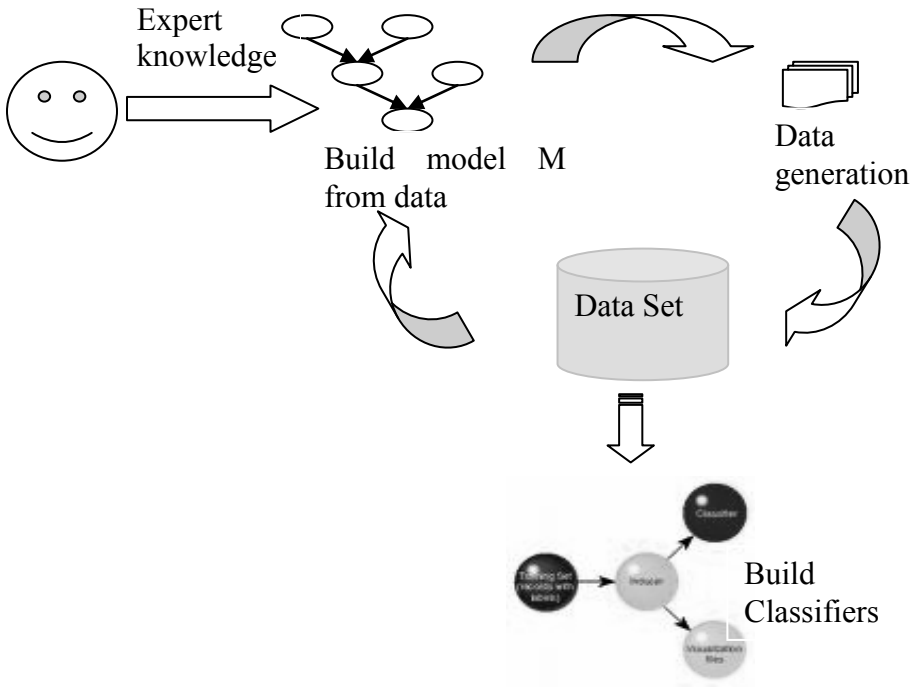


Figure 5-3 Work flow in model driven sampling classification

1) We first build a Bayesian network model M from the training data set or from the domain knowledge or both using multiple methods;
2) Model M generates new data samples;
3) The generated data is combined with the original data to form a new training data set to train classifiers.

5.3.2 ALGORITHM OF MODEL DRIVEN SAMPLING

The assumptions in MDS are – the training database is D, the data size is N, the minority data size is N', and the domain knowledge is K; the optimal data distribution is the perfectly balanced data distribution with imbalance ratio *i*=0.5; the evidence in data generation step is the minority class. The objective is to build a model M from training data D for better data sampling. The algorithm of model driven sampling approach is as following:

Model Driven Sampling Algorithm:

Given: the training data D, the data size N, the number of minority data N', the domain knowledge K

1) *Calculate the imbalance ratios i = N'/ N;*

2) *So the number of minority instances to be sampled is (0.5-i)N in order to achieve a balanced training data set;*

M-Step: Model building step M = (Bs, Bp)

3) *Learning Bayesian network structures from D; the best performing structure Bs is selected to ensure the correctness of the model;*

4) *Learning Bp for Bs from D, using Simple Estimator algorithm;*

5) *Update (Bs, Bp), if there is domain knowledge available*

S-step: Data sampling step

6) *Setting the minority value as the class evidence, and the number of instances to be generated is (0.5-i)N. The data D' containing (0.5-i)N instances is sampled from the model M(Bs, Bp) using Pearl MCMC [115]method;*

C-step: Data combination step

7) *Combining data D and D' to form a balanced training data BD;*

B-step: Build classifier

8) *Building a classifier from data BD to do data classification on the testing data.*

In the first step, we calculate the imbalance ratio of the training data i=N'/N, and thus the number of minority instances to be generated is (0.5-i)N in order to produce a balanced training data set.

In the M-step, we build different models using various structure learning algorithms such as K2 [32], Hill Climbing and CI algorithms with two scoring metrics - BDeu (Bayesian Dirichlet equivalent uniform) [17] and Bayes from Weka [2]. The reason we use different algorithms is that certain algorithm may perform better than others in certain cases. By using different algorithms, the best performing model (B*s*, B*p*) is selected as the final model M for data generation. The conditional probability tables of

the Bayesian network are learned using simple estimator by estimating directly from data once the structure is known. The simple estimator produces direct estimates of the conditional probabilities that is shown in the following equation:

$$P(x_i = k|pa(x_i) = j) = \frac{N_{ijk}+N'_{ijk}}{N_{ij}+N'_{ij}} \qquad \text{Equation 5-2}$$

We use N_{ijk} $(1 \leq i \leq n,\ 1 \leq j \leq q_i,\ 1 \leq k \leq r_i)$ to denote the number of records in D for which $pa(x_i)$ takes its jth value and for which x_i takes its kth value (r_i is the cardinality of x_i, and q_i is the cardinality of nodes in $pa(x_i)$). $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$. N'_{ijk} is the alpha parameter with non negative value, and we get the maximum likelihood estimates [102] when alpha =0. The domain knowledge in M step can help construct the model in two ways: 1) alter the structure Bs by arc operations, including deleting arcs, adding arcs, and changing arc directions 2) Estimating the prior distribution for Bp.

In the S-step, we make use of Pearl MCMC [115] method to generate instances. The model is built from M-step (Bs, Bp). We assume that the observed evidence in Bs is the minority class. We name the generated data as D'. The original training set D and the generated data D' are combined to form the new balanced training set BD. Traditional classifiers e.g. decision tree, Bayesian network, or support vector machine etc. can then be built on data BD.

5.3.3 BUILDING MODEL

5.3.3.1 Building model from domain knowledge

Domain knowledge is commonly used in medical decision support systems. Domain knowledge could come from scientific laws, expert opinions, accumulated personal experiences, common sense knowledge, etc. Domain knowledge is usually verified by life experiments and applications. Thus domain knowledge is assumed to be true in model building.

There are many works incorporating domain knowledge in machine learning models [108, 160]. When the data is sparse, or when we do not have any data available, data sampling methods are generally not effective. When a large amount of data is missing, or when multiple hidden nodes exist, learning parameters in Bayesian networks from data becomes extremely difficult [89]. However in MDS, we still can create models from domain knowledge. A model contains both qualitative representation and quantitative representation. The qualitative representation is the structure of Bayesian Network. The structure can be represented as topological constraints [62] which can be derived from domain knowledge. Quantitative representation of the model refers to the parameters of a Bayesian Network, which can also be estimated from domain knowledge [88] as shown in Appendix C.3.

5.3.3.2 Building model from data

We can also build model from training data set only. In this case, we need to learn the structure and the parameters for the model. There are two types of methods for learning

structures – score based methods and constraint based methods. Score based methods include Greedy Search, K2, MCMC, Hill Climbing etc, and constraint based methods include CI, ICS, etc. Without loss of generality, we chose to use K2, Hill Climbing method and CI algorithms for structure learning. K2 is using hill climbing adding arcs with a fixed ordering of variables. We used random order in our experiments. Hill climbing [18] adds and deletes arcs with no fixed ordering of variables. CI algorithm is to test whether variables x and y are conditionally independent given a set of variables Z for all combinations of x and y.

Given the BN structure is known, there are two categories of parameter learning problems – learning from complete data and learning from incomplete data which are described Appendix C.2. In this dissertation, we assume that our data are complete. We use simple estimator for parameter learning as introduced in section 5.3.2.

5.3.3.3 Building model from both domain knowledge and data

Building model from both domain knowledge and training data set is an added advantage of Bayesian Network. We can learn the initial structure and parameter set from the domain knowledge, and then we can update them using the training data. The special characteristics of Bayesian Network enable us to update the structure and parameter easily. The structure can be updated by arc operations including adding, deleting and reversing. The parameters can be verified and updated from experts' experiences.

5.3.4 DATA SAMPLING

The data sampling method we used is Pearl MCMC method [115] from the package of Bayesian network in Java (BNJ) [3]. MCMC method is also known as the Gibbs sampler as described in Appendix C.6.4. MCMC method can simulate realizations from complicated stochastic models in high dimensions by making use of the model's conditional distributions, which usually generates a much simpler and more manageable form as shown in the following data sampling step.

Data Sampling Step

Suppose we want to obtain samples of $X = \{x_1, \dots, x_n\}$ from the model (B*s*, B*p*) - a joint distribution $p(x_1, \dots, x_n)$ where $x_n = e$.

Step 1: We denote the *i*th sample by $X^{(i)} = \{x_1^{(i)}, \dots, x_n^{(i)}\}$ and we begin with some initial value $X^{(0)}$ for each variable;

Step 2: For *i*th sample where $i = \{1 \dots k\}$, sample each variable $x_j^{(i)}$ from the conditional distribution $p(x_j^{(i)} | x_1^{(i)}, \dots, x_{j-1}^{(i)}, x_{j+1}^{(i-1)}, \dots, x_n^{(i-1)})$.

The input for data sampling will be the model we built from the previous step (B*s*, B*p*) and the evidence file. In the evidence file, we specify that the observed evidence is the class variable associated with minority value, regardless of values for the rest of the variables in the network B*s*. We then sample each variable from the distribution of that variable conditioned on all other variables, making use of the most recent values and

updating the variable with its new value once it has been updated. The output is (0.5-*i*)N number of generated minority instances D' with the same features as those from the training space D. The generated data set D' is combined with D to form the new balanced training data set BD.

5.3.5 BUILDING CLASSIFIER

With the balanced training data BD, building a classifier from it is trivial. Most of the existing machine learning algorithms can be used for building a classifier on the balanced training data. We have experimented on different classification techniques including Support Vector Machines (SVM), C4.5 decision tree, and Bayesian networks. For consistency purpose and stable performance, we mainly use Bayesian network classifier in our experiments.

5.4 POSSIBLE EXTENSIONS

5.4.1 PROGRESSIVE MDS

In progressive MDS, the assumption of the best data distribution is not the balanced data distribution. Instead, the optimal data distribution is selected by progressively running MDS on different data distributions. The best performing data distribution will be selected as the optimal data distribution. Progressive MDS extends MDS in that a better performing data distribution is chosen instead of balanced data distribution. The details about progressive MDS will be discussed in Chapter 8.

5.4.2 CONTEXT SENSITIVE MDS

Context sensitive MDS is an extension of Model Driven Sampling in that the model is built with respect to a certain context. The technology we used is context sensitive Bayesian network as described in Appendix C.4. Context sensitive MDS can better model the small disjuncts by building sub-models for each of them. The sub-models will be more accurate for each small disjunct, and thus the data generated is more meaningful. The details of context sensitive MDS will be discussed in Chapter 9.

5.5 SUMMARY

In this chapter, we have proposed a new technology – Model Driven Sampling (MDS). We make use of Bayesian networks to construct our model. Because of the properties of Bayesian networks, we can construct models from data or from domain knowledge or both; we can also construct context sensitive models by using context sensitive Bayesian networks. The advantage of MDS is that it can make use of the whole training space to generate data samples, and it can also make use of domain knowledge to generate data samples. Thus MDS uses a much stronger knowledge base than other data sampling approaches.

The main limitation of MDS is that generally it is not efficient to be used for very high dimensional data space. This is because Bayesian network learning is exponential with respect to the data dimensions. So for high dimensional data space, we need to do feature selection before applying MDS. Normally, feature selection can reduce noise, and

focus on the selected important features. Therefore the model built with selected sub-feature set is more accurate [157].

CHAPTER 6: EXPERIMENT DESIGN AND SETUP

6. EXPERIMENT DESIGN AND SETUP

We have run and compared different machine learning algorithms including but not limited to C4.5 decision tree [122, 123], Support Vector Machine (SVM) [118], and Bayesian Network [31, 32, 143] (as described in section 2.1.4) on four data sets with three different sampling techniques including Random Sampling (RS), Synthetic Minority Over-sampling Technique (SMOTE) [23] and Model Driven Sampling (MDS). As it was shown in section 2.1.5 and 2.2.4.3, Bayesian network classifier was more stable in imbalanced data learning. Therefore, for consistency and length limitation, we report the running results from Bayesian network classifier only in this chapter. Each experiment was conducted by using 10 fold stratified cross validation, which made use of 90% of the data as the training data and the other 10% of the data as testing data.

6.1 SYSTEM ARCHITECTURE

The architecture of our system is shown in Figure 6-1. The training data set was first split into ten folds. Each time, we combined nine folds of the data as training data, and then tested on the remaining one fold. This procedure was repeated ten times, and we then derived the average performance value. Then each part underwent MDS sampling, and formed new balanced and enriched training data which was used to train the predictive

model. Part of the following descriptions and some preliminary results were reported in [159].

The system used for the experiments has an Intel 2.33GHz CPU, and 3.25 GB of memory. The software tools used for our experiments include Weka [2], Netica [1] and BNJ [3].

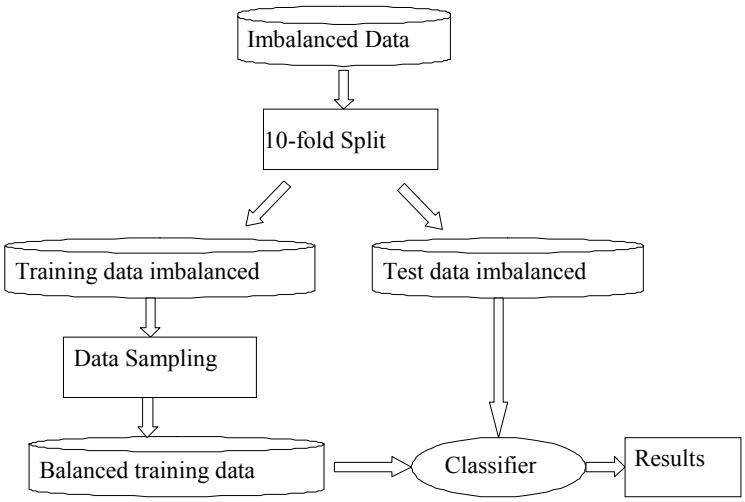


Figure 6-1 Experiment design for comparing different approaches

In all the experiments, we assumed that the optimal data distribution is a balanced distribution. Therefore, the sampled training data set is always balanced. In particular, in random over sampling, the parameter of bias to uniform is set to 1; in SMOTE sampling, the number of nearest neighbors to be generated is set to 5; in MDS, the evidence for data generation is set as positive for the class variable.

6.2 DATA SETS

The datasets used include simulated and real datasets. Without loss of generality, we simulate a circle for two dimensional data, and a sphere for three dimensional data.

6.2.1 SIMULATED DATA SETS

In the simulated data, we generated two dimensional data (simulating a circle), three dimensional data (simulating a sphere), and multi-dimensional data from the ALARM network [13]. The dimensions of the data set reflect data complexities. We have shown that MDS performs well on data sets with different dimensionality.

6.2.1.1 Two dimensional data

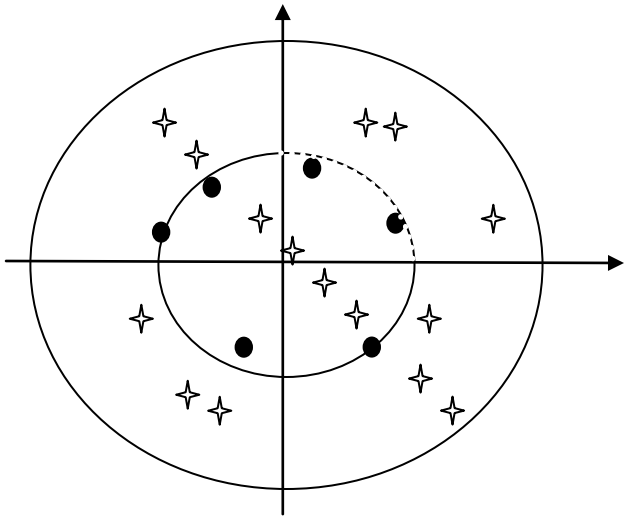


Figure 6-2 Two dimensional data set

We randomly generate the two dimensional data inside a circle. As shown in Figure 6-2, the four point stars represent majority data, and black spots represent minority data. The inner circle is the circle A centered at (0, 0) with radius 1, and the outer circle is the circle

B centered at (0, 0) with radius 2. Minority data locate around the inner circle A. Majority data spread inside circle A or between circle A and circle B. In this data set, we have generated 720 samples with imbalance ratio of 0.028. We assume that the domain knowledge is the approximate observations from the first quarter of the inner circle. The model built on this domain knowledge will randomly sample approximate data from the dashed curve (as marked in Figure 6-2) with a small error value of ε.

6.2.1.2 Three dimensional data

The three dimensional data is randomly sampled from the half sphere which is centered at (0, 0, 0) and with a radius of 1. There are 202 positive data samples generated approximately around the half sphere with an error value less than 0.07, and 811 negative data samples generated which are either outside the half sphere or inside the half sphere. There are 67 noisy data, including 56 false positive data samples and 11 false negative samples. The domain knowledge we assumed is the approximate observations from the first quarter of half sphere ($x>0$, $y>0$, $z>0$). The classification problem is defined to correctly identify the distributions for the minority data (samples on sphere) and majority data (samples off sphere), and the minority distribution is more critical than majority distribution.

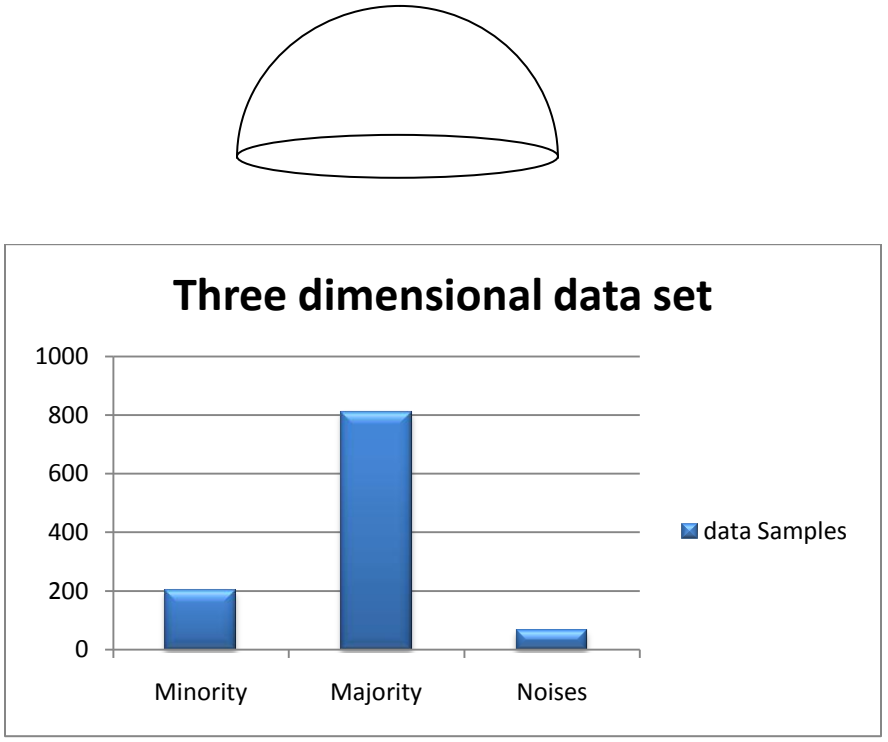


Figure 6-3 Three dimensional data - half sphere

6.2.1.3 Multi – dimensional data

We make use of ALARM network as shown in Figure 6-5 to generate the multi-dimensional data set. The ALARM network was first introduced by Beinlich, et al. [13]. It has 37 random variables and 46 arcs. The class variable is "FIO2" marked by a dashed rectangle in the network. We use ALARM network to generate 10,000 samples using Netica [1]. The training data contains 9718 majority samples with normal FIO2, 93 minority samples with low FIO2 and 189 (2%) missing samples (as shown in Figure 6-4). The domain knowledge we assumed is the 1000 approximately observed minority samples from the ALARM network. The classification problem is defined to correctly identify FIO2 value for each patient.

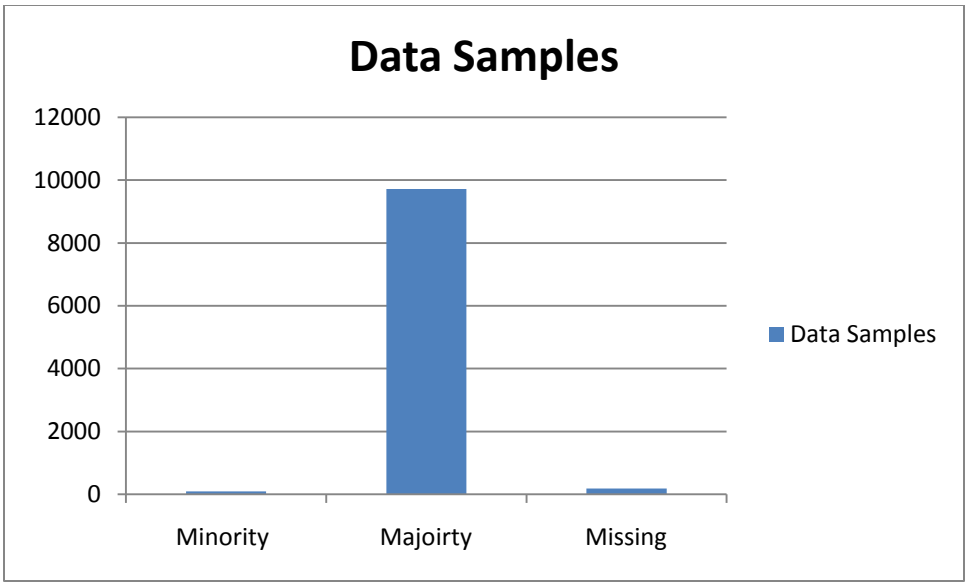


Figure 6-4 Multi dimensional data set

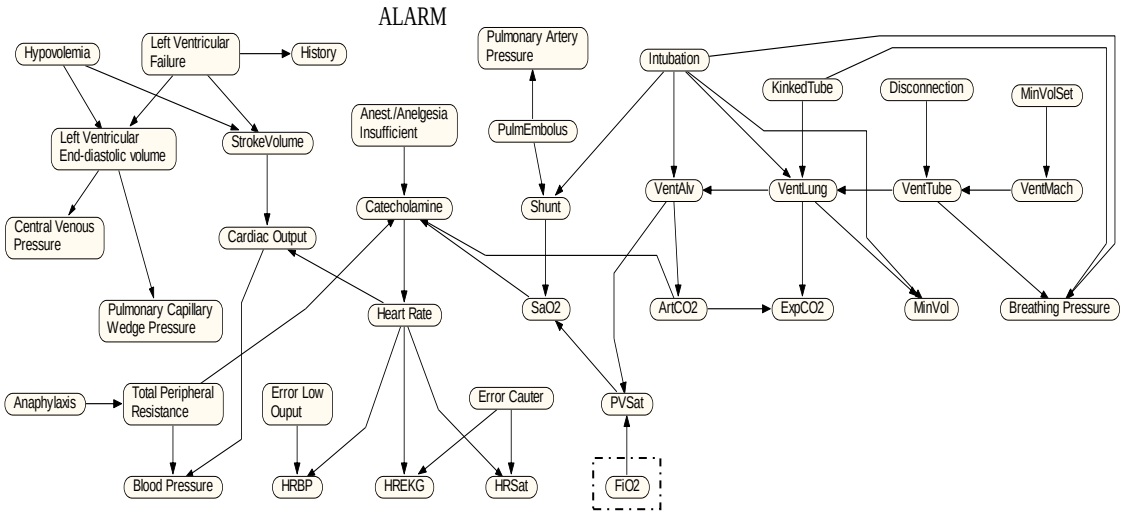


Figure 6-5 A Logical Alarm Reduction Mechanism [ALARM]

6.2.2 REAL LIFE DATA SETS

The real life data sets selected for the analysis span a wide spectrum in terms of complexity or dimension, imbalance ratio, and size; they are meant to illuminate the strengths and limitations of the algorithms studied under different conditions in medical domains. The data sets are Asia, Mammography, Indian Diabetes, Head Injury data, and Mild Head Injury data. The Asia data set is commonly used in machine learning communities as examples illustrating Bayesian Network learning. The head injury data and mild head injury data were described in Chapter 2. The other two data sets are from the UCI Machine Learning repository [14] which were used for imbalanced data learning [23, 57, 152]. The characteristics of the data sets are: binary data, unevenly distributed with different imbalance ratios (IB) as shown in Figure 6-6 and Table 6-1. IB ratio is equivalent to the percentage of minority examples in the training data; the lower of the value the more imbalanced the data is.

Table 6-1 - Class distributions (in numbers)

	Majority	**Minority**	**IB ratio**	**Features**
Asia	530	42	0.073	7
Indian Diabetes	500	268	0.349	8
Mammography	10923	260	0.023	6
Mild Head Injury	1776	29	0.016	45
Head Injury	307	184	0.375	17

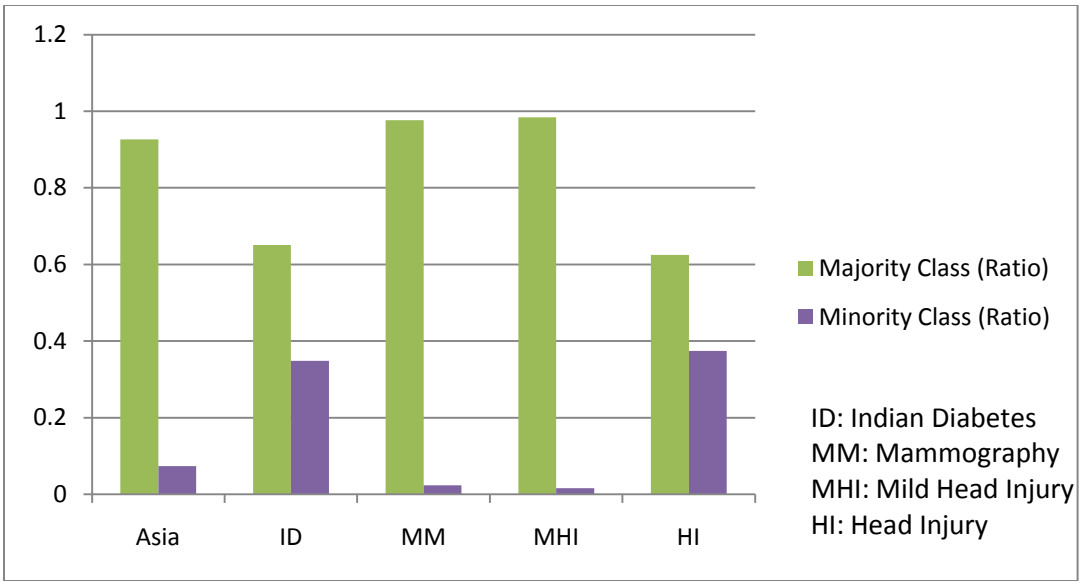


Figure 6-6 - Data class distributions (in relative ratios)

The Asia data set is about people who visited Asia and whether they had developed dyspnea or not. In our experiment, the Asia data set includes 42 positive cases, and 530 negative cases.

The Pima Indian Diabetes [14] data set includes 2 classes and 768 samples. The data is used to identify the positive diabetes cases in a population near Phoenix, Arizona. There are only 268 positive class samples.

The Mammography data set has a high skewed ratio: 10923 negative examples versa 260 positive examples. The trained classifier needs to be highly sensitive to detect the positive cases.

Head injury data set has a less imbalanced level of IB 0.375, while mild head injury data set has a high imbalance level of IB 0.016. We discussed how helping the

traditional machine learning methods in solving them in Chapter 2. In this chapter we are using imbalanced data learning techniques.

6.3 EXPERIMENTAL RESULTS

6.3.1 RUNNING RESULTS ON SIMULATED DATA

In model driven sampling, the model can either be built from the training data using machine learning method or from domain knowledge. The domain knowledge in the simulated data is the partial observation of the model. For example in the circle data, the domain knowledge we assumed is one quarter of the circle as shown in Figure 6-2, and in the sphere data, the domain knowledge is the one quarter of the sphere as shown in Figure 6-3. We compared both MDS based on data (MDS-Data) and MDS based on domain knowledge (MDS-Knowledge) in the simulated data.

6.3.1.1 Circle data

In circle data set, MDS has the same G-Mean value as random sampling. MDS with domain knowledge performs much better than the rest and it has a relatively balanced TP value of 0.901 and TN value of 0.75. (Original data refer to the data set without any sampling, and RS stands for “random sampling”)

Table 6-2 Running Results on Circle Data (P-value < 0.01)

	Original Data	**RS**	**SMOTE**	**MDS-Data**	**MDS – Knowledge**
TP[3]	0.45	1	0.45	1	0.901
TN[4]	1	0.574	1	0.574	0.75
G-Mean	0.671	0.758	0.671	0.758	0.822

6.3.1.2 Half-Sphere data

In half-sphere data, both MDS with machine learning and MDS with domain knowledge perform better than other sampling approaches. MDS with domain knowledge performs better than MDS with machine learning.

Table 6-3 Running Results on Half-Sphere Data (P-value <0.05)

	Original Data	**RS**	**SMOTE**	**MDS-Data**	**MDS-Knowledge**
TP	0.296	0.493	0.493	0.557	0.75
TN	0.999	0.864	0.864	0.809	0.66
G-Mean	0.544	0.652	0.652	0.671	0.703

6.3.1.3 ALARM data

In the ALARM data set, the domain knowledge assumed is the partial observation of the model which is a sub set of instances approximately generated from the true model using MCMC method. MDS achieves the best performance on the minority data and in overall. MDS based on domain knowledge performs much better than the other approaches too, ranked as the second best as shown in Table 6-4. Interestingly, we realized that MDS-

[3] TP is true positive rate for predicting minority samples.

[4] TN is true negative rate for predicting majority samples.

Knowledge performed worse than MDS based on data. This is because that the domain knowledge in ALARM data is encoded by approximately (MCMC) generating partial observations from the ALARM network. ALARM network itself is a large Bayesian network, therefore it needs a super large sample set in order to generate the simulation model that is close to the underlying true distribution. The partial observations used are a small subset of the simulated samples, which can be biased and are not necessarily better than the data generated from our MDS model. Therefore, MDS can sometimes perform better than MDS with domain knowledge.

Table 6-4 Running Results on ALARM Data (P-value < 0.05)

	Original Data	**RS**	**SMOTE**	**MDS-Data**	**MDS-Knowledge**
TP	0.366	0.366	0.376	0.777	0.591
TN	0.841	0.86	0.964	0.86	0.856
G-Mean	0.554	0.561	0.602	0.817	0.711

6.3.2 RUNNING RESULTS ON REAL LIFE DATA SETS

6.3.2.1 Asia data

The Asia data set has the lowest number of minority examples and the second lowest imbalance ratio 0.073. As shown in Table 6-5, the original data set without any sampling has a high prediction rate on its majority samples (98.7%), but a low prediction accuracy on its minority samples (7.1%), thus the overall performance is the lowest at 26.5%. Random sampling and SMOTE both significantly improve the predictions on minority

samples and achieve a much better overall performance. MDS achieves the best performance 88% overall and 90.5% on minority data set.

Table 6-5 Asia data running results

	Original Data	**RS**	**SMOTE**	**MDS**
TP	0.071	0.881	0.69	0.905
TN	0.987	0.863	0.925	0.856
G-Mean	0.265	0.872	0.799	0.88

6.3.2.2 Indian Diabetes data

The Indian Diabetes data is a relatively balanced data set with the highest imbalance ratio at 34.9%. Therefore, without any sampling, the original data set can achieve a satisfying performance on minority data and a good overall performance. The three sampling approaches equally improve the performance especially on the minority. The overall performance however is not much improved by MDS. (As shown in Table 6-6)

Table 6-6 - Indian Diabetes data running results

	Original Data	**RS**	**SMOTE**	**MDS**
TP	0.669	0.783	0.787	0.752
TN	0.836	0.741	0.745	0.783
G-Mean	0.748	0.762	0.766	0.767

6.3.2.3 Mammography data

Although the Mammography data set has the lowest imbalance ratio 0.023, it is still relatively simple as it has only 6 features which result in a low data complexity. In Table 6-7, the original data set can achieve 85% overall performance. The other approaches can equally improve the minority prediction by 15%. SMOTE has the best overall

performance 89%, and MDS has a comparable performance of 88.5%. However, MDS has the highest performance on minority data, with TP value equal to 0.901.

Table 6-7 - Mammography data running results

	Original Data	**RS**	**SMOTE**	**MDS**
TP	0.735	0.888	0.873	0.901
TN	0.981	0.857	0.908	0.869
G-Mean	0.849	0.872	0.89	0.885

6.3.2.4 Head Injury data

The Head Injury data is a relatively less imbalanced data set. The overall performance of its original data is reasonably good. All three data sampling techniques can improve the overall performance, particularly on the minority data. Among all the approaches, MDS performs the best.

Table 6-8 Running results for Head Injury data

	Original Data	**RS**	**SMOTE**	**MDS**
TP	0.674	0.713	0.717	0.728
TN	0.847	0.823	0.821	0.824
G-Mean	0.755	0.766	0.767	0.774

6.3.2.5 Mild Head Injury data

The Mild Head Injury data has the highest imbalance level – the lowest imbalance ratio 0.016 among all the five data sets. The performance on the original data is very poor. Both random sampling and SMOTE can improve the system performance. However, MDS is the best performing approaches. MDS also improves the minority prediction accuracy greatly and it has the highest true positive rate of 0.621.

Table 6-9 Running results for Mild Head Injury data

	Original Data	**RS**	**SMOTE**	**MDS**
TP	0.207	0.552	0.414	0.621
TN	0.994	0.844	0.853	0.831
G-Mean	0.453	0.683	0.594	0.718

6.4 SUMMARY

There are three important factors affecting an imbalanced data set: 1) the imbalance ratio, 2) the absolute size of the minority data, and 3) the dimension of the data set. The three factors are common in most medical data sets, and they vary among the five representative data sets chosen in this work. We have examined relatively easy problems which are less imbalanced, low dimensional, with sufficient minority samples (e.g., Indian Diabetes and Mammography datasets), to hard problems which are highly imbalanced, high dimensional (e.g., Head Injury problem), or with scarce minority samples (e.g., Asia).

The three different data sampling approaches discussed represent a wide range of data sampling efforts in tackling the imbalanced problems. They can be categorized by their learning scopes. Random sampling duplicates the data without creating new information; the SMOTE algorithm creates new synthetic data based on local information – the nearest neighbors; the MDS approach generates data based on global information – the knowledge model built from the full training space. As illustrated in Figure 6-7, random sampling produces data from a single data point; SMOTE generates data over

two data points; MDS generates data from a model built from all labeled data or domain knowledge.

As shown in Figure 6-8, MDS or MDS based on domain knowledge performs better than the other approaches. Typically, in the circle data, MDS is equivalent to random sampling, and MDS based on domain knowledge performs the best; in the sphere data set, MDS based on domain knowledge performs the best, and MDS performs the second best; in the ALARM data set, MDS performs the best, and MDS based on domain knowledge performs the second best.

As shown in Figure 6-9, all three different sampling approaches can improve classification performance on imbalanced data sets, especially on minority data. Comparing these three sampling approaches, random sampling is easy to implement and efficient; SMOTE will perform well especially when the minority data is dense; MDS will perform well when we have a reasonable accurate model to generate minority data, and this model could be from our medical domain knowledge or learning from existing data or both. Thus MDS can potentially address imbalanced problems with scarce or sparse minority data. As shown in section 6.3.1, MDS with domain knowledge is usually the best performing approaches with statistically significant improvement. However, as the lack of domain knowledge, we did not report results for MDS with domain knowledge in real life data sets. In future work, we will incorporate domain knowledge into our model for real life data sets. This capability is a major difference from and is a potential advantage over the other generative sampling approaches [91].

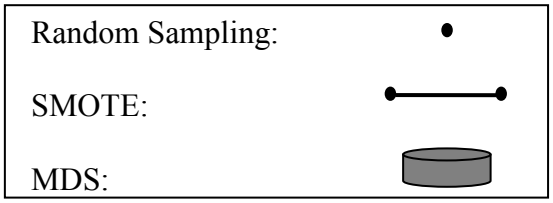


Figure 6-7 Learning scopes for 3 sampling approaches

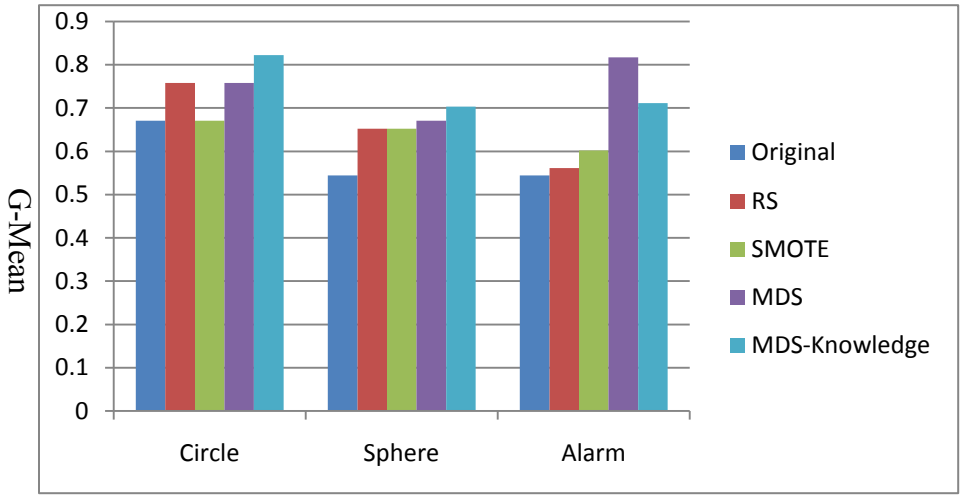


Figure 6-8 Overall comparisons among simulated data

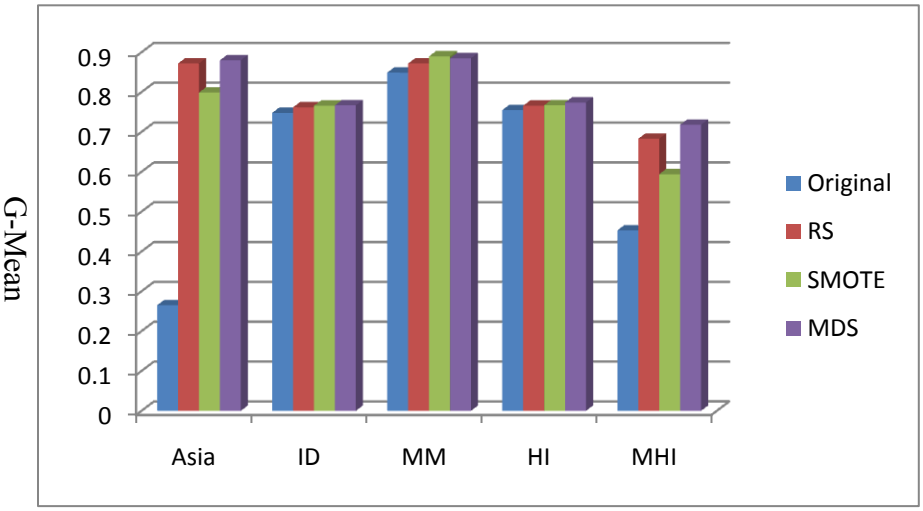


Figure 6-9 Overall performance (G-Mean) comparison

CHAPTER 7: MDS IN ASTHMA CONTROL

7. MDS IN ASTHMA CONTROL

In this chapter, we report on a case study in a real life project – the asthma control problem. Asthma is a chronic disease, and asthma control is about controlling the disease by giving necessary and adequate continuous treatment. It is costly and critical if the disease is not under control. In the asthma project, we try to predict whether the patient's asthma is under control or not before the next visit, so that necessary precautions can be taken ahead to reduce the risk of control failure. However, control failures occur only occasionally in all asthma patients. We will make use of random sampling, SMOTE sampling and MDS on asthma control predictions.

7.1 BACKGROUND

Asthma is still an important cause of ill health in today's population. It consumes a lot of heath service resources [150], because that many patients who have poorly controlled asthma require unscheduled visits to hospitals for urgent nebulizations, emergency department visits or admissions to hospital. Therefore, to reduce the usage of acute health services for asthma is very important in health care.

Asthma management is about how to control the disease. One important topic is to correctly predict whether the patient is going to be under control or not in the near

future. ACT [55] is used for measuring the severity of the disease using five self evaluation questions. It is designed for patients at home conveniently evaluating whether his asthma is under control or whether he needs to see a doctor. ACT is particularly useful for evaluating patients' current situation, i.e. whether asthma is under control or not at the time of taking ACT test. However, it is also very useful for clinicians or patients to know the potential risk of getting out of control in the near future. Correctly knowing the future can help clinicians or patients to be better prepared and carefully plan the treatment to avoid a costly situation and to prevent suffering. In this project, we have completed an initial study on how to correctly predict asthma control failure in the future based on the information provided at each clinic visit. The outcome measures for control failure are unscheduled physician visits for urgent nebulization or hospitalization [55, 103] that appear in any of the subsequent clinic visit. If they do not have any subsequent visit in our patients' records, then we assume that the patient is under control.

7.2 DATA SETS

7.2.1 DATA DESCRIPTION

The data sets we used are collected by a local hospital in Singapore under proper approval and usage guidelines from April 2001 to July 2006. There are two data sets – asthma first visit data and asthma subsequent visit data.

Table 7-1 Data sets collected from our asthma program

	Collection Duration	Data Size	Minority Data	Majority Data	Imbalance Ratio	Attributes
First Visit	20/04/2001~ 02/06/2006	942	213	729	0.226	138
Subsequent Visit	18/05/2001~ 13/07/2006	5294	1247	4047	0.236	115

These two data sets share the common characteristic that they are both imbalanced. The imbalance ratio for asthma first visit data is 0.226 and the imbalance ratio for asthma subsequent visit data is 0.236.

The asthma first visit is a patient based outcomes analysis problem. The data set records the information when asthma patients visit the respiration centre for the first time. It has 138 attributes recording the patients' general information, asthma history, treatment history, etc. There are 213 positive samples out of a total of 891 samples. The main problem is to determine whether a patient will encounter any control failure in the future based on the information provided on his first visit.

The asthma subsequent visit is a visit based outcomes analysis problem. The data contains the patients' information at each visit. The problem is to predict asthma control given a patient's current visit information. The assumption in this problem is that the future control failure is independent of a patient's past visits given his current visit information.

7.2.2 DATA PREPROCESSING

7.2.2.1 Feature selection

Feature selection is necessary for data sets with high dimensions to reduce problem complexity and also to remove noisy parameters. It is particularly useful for the asthma data because there are more than 100 parameters in each of the data sets, and a lot of parameters such as patient's ID number, visit date etc will not help in predicting control. Instead they add noise to the built model and affect the efficiency and effectiveness. We used Bayesian Networks to build predicting models by selecting 40 features (chi-squared value > 4.5) out of 138 (Table 7-2) and selecting 20 features (chi-squared value > 9.7) out of 138 (Table 7-3). Generally the 20-feature model performs better than the 40-feature model, except for SMOTE, whose performance drops slightly. The 20-feature model is also much less complex than the 40-feature model and therefore is more efficient. In this experiment, we made use of Chi-square feature selection i.e., evaluating the worth of the attributes by computing the value of the chi-squared statistics with respect to the class. The selected features for asthma first visit are shown in Appendix A. The features for asthma subsequent visit are shown in Appendix B.

Table 7-2 Asthma first visit running results- 40 features out of 138

	Original Data	RS	SMOTE	MDS
TP	0.419	0.576	0.448	0.59
TN	0.852	0.732	0.805	0.732
G-Mean	0.598	0.649	0.6	0.657

Table 7-3 Asthma first visit running results - 20 features out of 138

	Original Data	RS	SMOTE	MDS
TP	0.423	0.606	0.39	0.643
TN	0.877	0.708	0.85	0.719
G-Mean	0.609	0.655	0.576	0.68

However, it is not necessary that the fewer features perform the better. Fewer features contain less information, though they reduce the system complexity. The optimal feature set contains the maximum information while effectively reducing the system complexity. To determine the optimal number of features, we make use of progressive feature selection methods to empirically decide the best set of features.

7.2.2.2 Discretization

Data discretization is necessary for dealing with continuous variables. Continuous variables are in contrast with nominal variables. A continuous variable can take any possible value with its range. For example, the variable age can take any integer value from 0 to 100. Data discretization is to convert the continuous variables into nominal variables. The data discretization algorithm that we used is minimum description length (MDL) algorithm [49]. Data dicretization helps improve the system efficiency and accuracy.

7.3 RUNNING RESULTS

We have experimented with various sampling approaches including MDS using different feature sets. Each of the features selected are verified by domain experts to ensure that

they are meaningful. 10-fold cross validation is used for reporting the experimental results.

7.3.1 ASTHMA FIRST VISIT DATA

In asthma first visit data, we used the patients' first visit information to predict whether the patient is under control in the future. The outcome variable is "control". A patient is under control if there is no urgent nebulization or hospitalization.

The running results for asthma first visit data are shown as in Table 7-2 (40-feature model) and Table 7-3 (20-feature model). The performance for MDS ranks as the best among all approaches. MDS also has the best minority prediction rate among all approaches. We also notice that MDS with 20 features perform better than MDS with 40 features. It could be the reason that the network built from high dimensional data set is too complex and the data generated might contain more errors.

We further reduce the features to a 7-feature set, without any drug changing features. The results are shown as in Table 7-4. In this set of features, all approaches improve their performances except that original data's drops. Among all feature combinations, MDS performs the best among all approaches. MDS also achieves the best True Positive rate of 0.723.

Table 7-4 Asthma first visit data running results with 7 features

	Original Data	RS	SMOTE	MDS
TP	0.305	0.657	0.573	0.723
TN	0.894	0.69	0.724	0.649
G-Mean	0.522	0.673	0.644	0.685

We also tried other feature combinations, and we found 7-feature set (including Patient's Record No, Asthma Duration, MC, Nebulisation Count, UNebulisation Freq Oral Steriods Count, DrugSubvention) is the optimal feature set for asthma first visit data. Interestingly, we notice that Patient's Record No is also an important factor. This is because Patient's Record No records the chronicle order of the patients visiting the clinic. Treatment and healthcare improves over time and thus over the Patients' Record No.

7.3.2 ASTHMA SUBSEQUENT VISIT DATA

In the asthma subsequent visit, we make use of the patients' current visit information (including but not limited to their first visit), to predict the whether the patient is going to be under control in the future. The outcome variable is same as in the asthma first visit data.

We compared 40 features (chi-squared value > 5.5), 21 features (chi-squared value > 16) and 6 features (chi-squared value > 55) using Bayesian Network classifiers with different sampling approaches. The running results for asthma subsequent data with 40 features are shown in Table 7-5. The performance for asthma subsequent data with 21 features is shown in Table 7-6. The performance on 6-feature set is shown in

Table 7-7. We can tell that all approaches' performances improve on 21-feature set over 40-feature set, and the performances are further improved on 7-feature set.

Table 7-5 Asthma Sub Visit Results (40-feature set)

	Original Data	RS	SMOTE	MDS
TP	0.213	0.42	0.287	0.483
TN	0.907	0.783	0.866	0.753
G-Mean	0.44	0.574	0.498	0.603

Table 7-6 Asthma Sub Visit Results (21-feature set)

	Original Data	RS	SMOTE	MDS
TP	0.473	0.582	0.508	0.578
TN	0.951	0.842	0.894	0.842
G-Mean	0.671	0.7	0.674	0.698

Table 7-7 Asthma Sub Visit Results (6-feature set)

	Original Data	RS	SMOTE	MDS
TP	0.49	0.559	0.548	0.565
TN	0.974	0.933	0.941	0.929
G-Mean	0.691	0.722	0.718	0.725

In all the different feature sets, we discovered that MDS performs better than other approaches or at least equivalent to random sampling in Table 7-6. We also tried other different feature combinations, but the best overall performance is achieved by MDS on 6-feature set. The six features are "MV Followup Wks", "MV UNebulisation", "MV Events", "MV Nights with wheeze/cough/SOB", "MV Days with

wheeze/cough/SOB", and "MV Activity stopped". Six feature set is also clinically efficient, and physicians can key in minimum parameters to get the best accurate predictions.

7.4 SUMMARY

One important characteristic for a clinically useful diagnosis system is to minimize the features utilized. So the physicians can make use of the minimal set of information to determine the possible outcomes efficiently and effectively. From the above experiments, that the optimal number of features for asthma first visit data is 7 and the optimal number of features for asthma sub visit data is 6. In both cases, MDS performs the best among all the considered approaches. MDS can make use of the minimal information, and produce better results.

This case study also shows that feature selection is important particularly for imbalanced data learning. High dimensional data often generates complex network structures, and can easily cause more noise in the data generated. Feature selection can select the minimum optimal feature set and thus ensure that the model built by MDS is clean, containing less noise.

CHAPTER 8: PROGRESSIVE MODEL DRIVEN SAMPLING

8. PROGRESSIVE MODEL DRIVEN SAMPLING

Data sampling can often improve the overall performance of the system by balancing the data distribution in the training space. However, it is not true that balanced data distribution always gives the best performance. The optimal data distribution can be imbalanced in certain domains. Progressive model driven sampling is to generate data progressively such that an approximately optimal data distribution can be discovered instead of blindly using balanced data distribution as the optimal data distribution. Since the model we build is based on the existing training data or expert knowledge, it is usually not a perfect model. The data generated from the model contain noise, and the usage of the generated data shall be limited to a certain degree. Progressive model driven sampling can discover the minimum amount of generated data to be used. It improves the system accuracy.

8.1 CLASS DISTRIBUTION MATTER

Class distribution plays an important role in imbalanced data learning. Different class distributions usually give very different performances. We study the relationships between class distribution and system performance in order to find an optimal class

distribution, such that minimum amount of generated data can give the best possible performance.

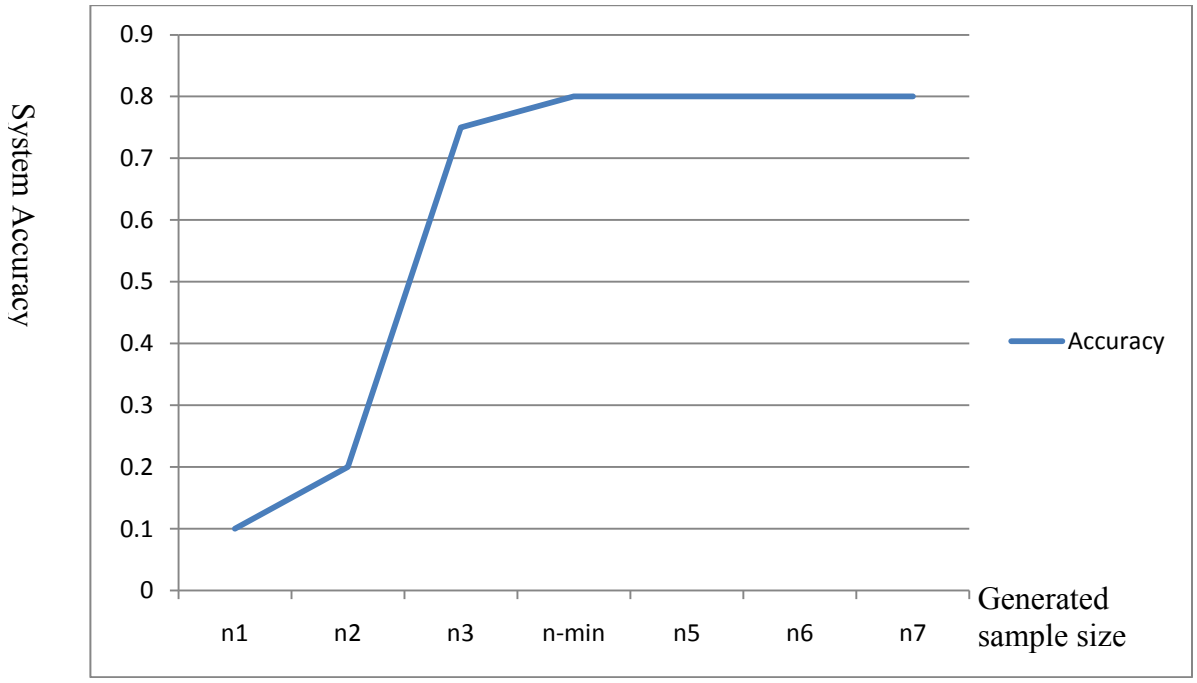


Figure 8-1 System accuracy versa the number of generated samples

We discovered from our empirical studies that the system performance curve with respect to the generate data size is similar to the curve shape in Figure 8-1. The horizontal axis represents the number of synthetic samples generated, and the vertical axis represents the system accuracy. A learning curve usually has a steep slope in the first portion followed by a gentle slope, and then a plateau. The plateau occurs when the system performance cannot be increased any more adding more data. The data distribution at n-min is the optimal data distribution. The assumption for the learning curve in Figure 8-1 is that the generated data is perfect data without noise. However, it is

not always true in reality, as the synthetic data generated by either SMOTE or MDS contains noise. Therefore, there is often a descending slope after the plateau in reality.

8.2 DATA SETS AND CLASS DISTRIBUTIONS

In this section we use classification results to learn the relationships between class distribution and classifiers' performance. The classifier used is the Bayesian Network classifier. 10-fold cross validation is used for reporting the experimental results.

8.2.1 DATA SETS

The data sets we experimented on include Circle Data, Sphere Data, Asthma First Visit Data (7 features), and Asthma Subsequent Visit Data (5 features) as shown in Table 8-1. These data are built in a way that we purposely decrease the original minority data size, so that we can better study the effect of progressive sampling.

Table 8-1 Data summaries for progressive sampling

	Features	Majority	Minority	Distribution
Circle Data	2	631	17	0.026
Sphere Data	3	730	182	0.20
Asthma First Visit	7	660	187	0.22
Asthma Sub Visit	5	4438	326	0.068

8.2.2 DATA DISTRIBUTIONS

The data distributions ranged from the original distribution to the balanced data distribution. At each step, we generated new minority data and added them into the

training space producing a new data distribution with imbalance ratio increased by a small percentage (2-5%), until the balanced distribution was reached. If the system performance improves obviously, we use a bigger incremental step (5% for example), otherwise if the performance drops or improves little, we use a smaller incremental step (2% for example).

Table 8-2 Progressive sampling distributions for Circle data

Majority data	Minority data	Generated	Data Distribution
631	17	0	0.026234568
631	17	53	0.1
631	17	109	0.166446499
631	17	141	0.2
631	17	172	0.230487805
631	17	204	0.259389671
631	17	235	0.285390713
631	17	253.4286	0.3
631	17	322.7692	0.35
631	17	450	0.425318761
631	17	614	0.5

Table 8-3 Progressive data distributions for Sphere

Majority data	Minority data	Generated	Data Distribution
730	182	0	0.199561404
730	182	61.33333	0.25
730	182	130.8571	0.3
730	182	169.4815	0.325
730	182	211.0769	0.35
730	182	304.6667	0.4
730	182	415.2727	0.45
730	182	548	0.5

We then test the system on each data distribution, and choose the distribution that gives the best performance result. The different data distributions generated for each data set is shown in Table 8-4 to Table 8-5.

Table 8-4 Progressive data distributions for asthma first visit

Majority data	Minority data	Generated	Data Distributions
660	187	0	0.220779221
660	187	33	0.25
660	187	95.85714	0.3
660	187	168.3846	0.35
660	187	253	0.4
660	187	353	0.45
660	187	473	0.5

Table 8-5 Progressive data distributions for asthma sub visit

Majority data	Minority data	Generated	Data Distributions
4438	326	0	0.068429891
4438	326	167.1111	0.1
4438	326	457.1765	0.15
4438	326	783.5	0.2
4438	326	1153.333	0.25
4438	326	1576	0.3
4438	326	2063.692	0.35
4438	326	2632.667	0.4
4438	326	3305.091	0.45
4438	326	4112	0.5

8.3 EXPERIMENT DESIGN IN PROGRESSIVE SAMPLING

The data were generated progressively as shown in previous section. The workflow for our experiment is shown in Figure 8-2. For each data distribution generated, we used three different sampling methods (random sampling, SMOTE and MDS) to sample the required amount of data, Bayesian network classifier's performance was recorded. The best performing data distribution for each sampling approach was then chosen. The performance of progressive MDS can be compared with the other sampling approaches.

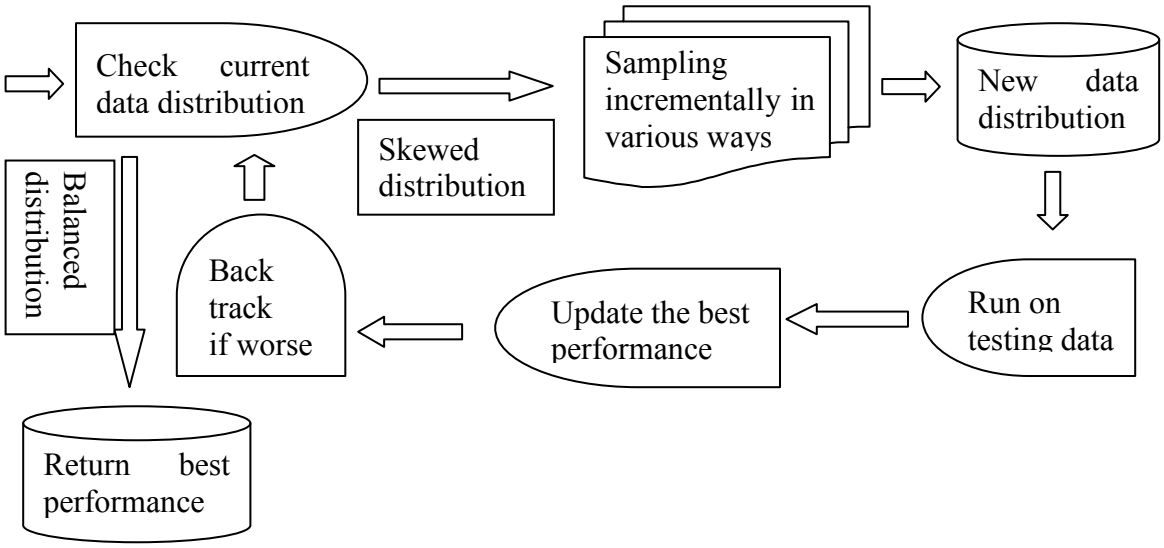


Figure 8-2 System flow for progress sampling

Progressive sampling algorithm:

Initialization: distribution DIS; best performance BP;

Loop:

Generate a new distribution DIS;

If DIS = 0.5, then return BP;

Otherwise, sample data to the new distribution DIS

Get the classifier performance BP' from the new distribution DIS

If BP' > BP, then update BP with BP'

Otherwise, backtrack the data distribution with a small percentage

Go to Loop.

In the progressive sampling algorithm, we start with the original data distribution, and progressively generate and more balanced data distributions. For each data distribution, we apply three different sampling algorithms to get the sampled distribution. The performance on the new sampled data is compared with the current best performance. If the performance improved, the current best performance will be updated; otherwise, we miss the best performing distribution which locates between the current distribution and the previous distribution, so we need to back track the data distribution. Once we reach the balanced distribution, we can return the best performance as the optimal performance and the best performing data distribution as the optimal distribution.

8.4 EXPERIMENTAL RESULTS

The approaches tested in progressive sampling are Random Sampling (RS), Synthetic Minority Over-sampling Technique (SMOTE), and Model Driven Sampling (MDS). The

g-Mean value for various approaches in progressive sampling on circle data is summarized in Table 8-6 and Figure 8-3.

8.4.1 EXPERIMENTAL RESULTS FOR CIRCLE DATA

The experimental results for MDS on Circle data set reach the best performance at the data distribution of 28.5% with g-Mean value equal to 0.776. The best performance for SMOTE approach is 0.767 at the imbalance ratio of 25.9%; the best performance for random sampling approach is 0.771 at the imbalance ratio of 25.9%.

Table 8-6 g-Mean value for progressive sampling running results in Circle 20 data

Imbalance Ratio	MDS	SMOTE	Random Sampling
2.60%	0.671	0.671	0.671
10.00%	0.671	0.671	0.671
16.60%	0.7	0.671	0.65
20.00%	0.664	0.678	0.644
23%	0.689	0.734	0.739
25.90%	0.73	0.767	0.771
28.50%	0.776	0.726	0.761
30%	0.739	0.738	0.743
35%	0.705	0.758	0.758
42.50%	0.758	0.758	0.758
50%	0.758	0.758	0.758

8.4.2 EXPERIMENTAL RESULTS FOR SPHERE DATA

The g-Mean value results for progressive sampling in Sphere data are summarized in Table 8-7 and Figure 8-4. The experimental results for MDS on Sphere data reach the best performance at the data distribution of 32.5% with g-Mean value equal to 0.671. The best performance for SMOTE approach is 0.652 from the imbalance ratio of 0.3

onward. The best performance for Random Sampling approach is 0.652 from the imbalance ratio of 0.25 onward.

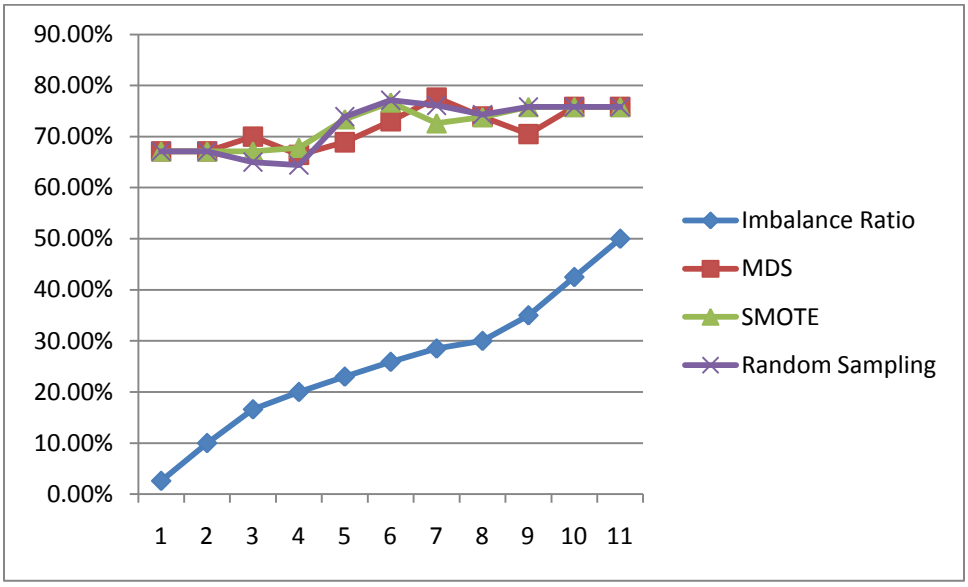


Figure 8-3 Progressive sampling results for various approaches in Circle data

Table 8-7 g-Mean value for progressive sampling in Sphere data

Imbalance Ratio	MDS	SMOTE	Random Sampling
0.2	0.544	0.544	0.544
0.25	0.646	0.648	0.652
0.3	0.66	0.652	0.652
0.325	0.671	0.652	0.652
0.35	0.652	0.652	0.652
0.4	0.653	0.652	0.652
0.45	0.649	0.652	0.652
0.5	0.652	0.652	0.652

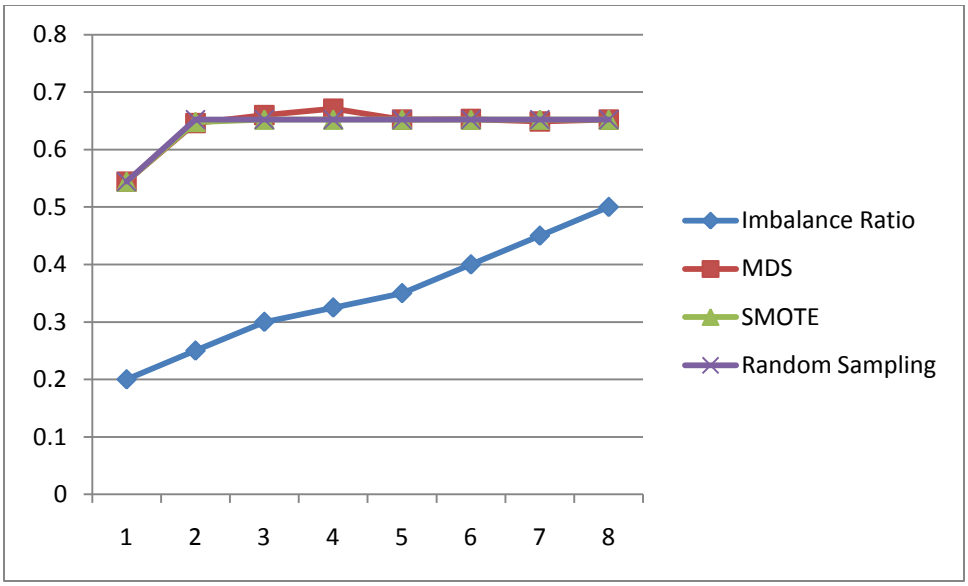


Figure 8-4 Experimental results for progressive sampling in sphere

8.4.3 EXPERIMENTAL RESULTS FOR ASTHMA FIRST VISIT DATA

The g-Mean values for asthma first visit data are summarized in Table 8-8 and Figure 8-5. The experimental results for MDS on asthma first visit data reach the best performance at the data distribution of 50% with g-Mean value of 0.685. The best performance for SMOTE approach is 0.649 at the imbalance ratio of 0.4; the best performance for Random Sampling approach is 0.691 at the imbalance ratio of 0.45.

Table 8-8 g-Mean value for progressive sampling in asthma first visit data

Imbalance Ratio	MDS	SMOTE	Random Sampling
0.22	0.522	0.522	0.522
0.25	0.561	0.561	0.632
0.3	0.605	0.597	0.633
0.35	0.641	0.631	0.676
0.4	0.651	0.649	0.683
0.45	0.677	0.639	0.691
0.5	0.685	0.642	0.673

8.4.4 EXPERIMENTAL RESULTS FOR ASTHMA SUB VISIT DATA

The g-Mean values for progressive sampling in asthma subsequent visit data (with 6 feature-set) are summarized in Table 8-9 and Figure 8-6. The experimental results on asthma sub visit data using MDS reach the best performance at the imbalance ratio of 0.45 with g-Mean value of 0.736. The best performance for SMOTE approach is 0.726 at the imbalance ratio of 0.25; the best performance for Random Sampling approach is 0.722 at the imbalance ratio of 0.5. It is interesting to note that only random sampling' performance is always increasing with new data generated, which means random sampling reaches the optimal performance at balanced data distribution. However, MDS and SMOTE reach their optimal performance before balanced data distribution.

Table 8-9 g-Mean value on progressive data sampling in asthma sub visit data

Imbalance Ratio	MDS	SMOTE	Random Sampling
0.068	0.691	0.691	0.691
0.1	0.686	0.702	0.701
0.15	0.688	0.712	0.701
0.2	0.7	0.722	0.703
0.25	0.694	0.726	0.704
0.3	0.698	0.725	0.708
0.35	0.727	0.717	0.712
0.4	0.719	0.68	0.715
0.45	0.736	0.676	0.719
0.5	0.729	0.63	0.722

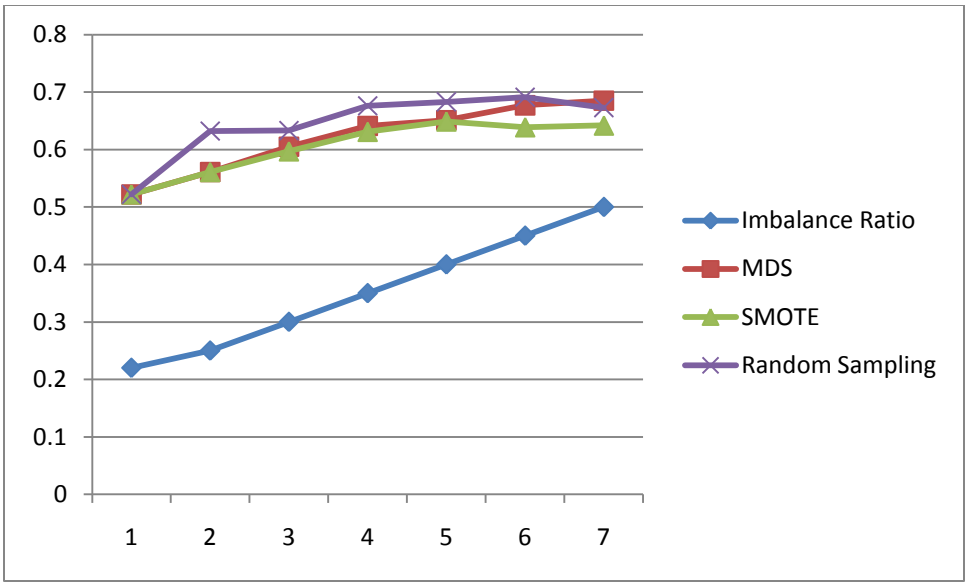


Figure 8-5 Experimental results in progressive sampling for asthma first visit data

Table 8-10 Optimal data distributions for various approaches

	MDS	SMOTE	Random Sampling
Circle 20	0.285	0.259	0.259
SphereN	0.325	0.3	0.25
Asthma First Visit Data	0.5	0.4	0.45
Asthma Sub Visit Data	0.45	0.25	0.5

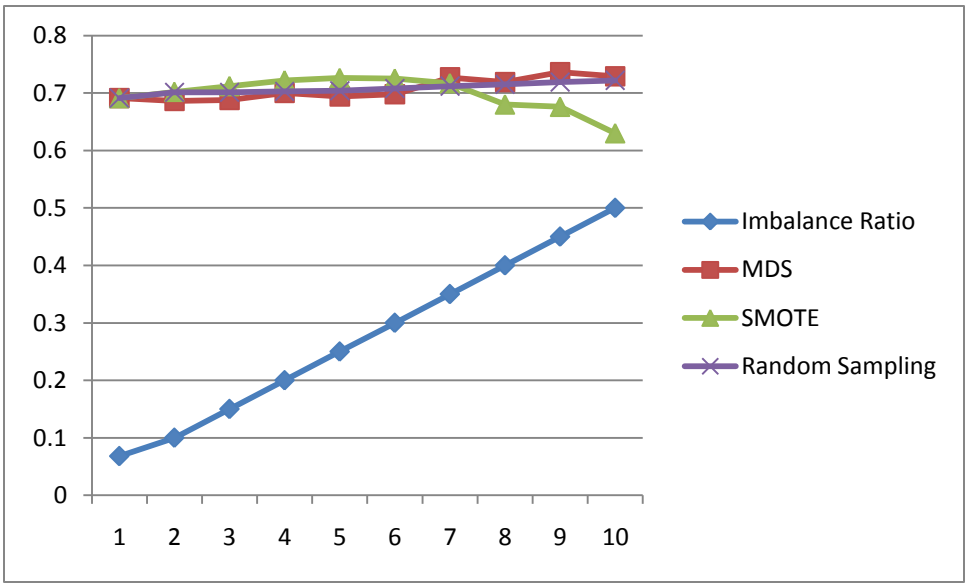


Figure 8-6 Experimental results for progressive sampling in asthma sub visit

8.5 SUMMARY

From the above experimental results, the optimal data distributions for various data sets and approaches are summarized in Table 8-10. We can see that most of the time, the optimal performance is not achieved at the balanced data distribution. Progressive sampling can help to identify the optimal or near optimal data distribution, and produce better results. In asthma first visit data and asthma sub visit data, the best data distribution is 0.5. This is because these two data sets are highly imbalanced and complicated, and more simulated data are usually preferred. In this scenario that balanced data distribution is optimal.

CHAPTER 9: CONTEXT SENSTIVE MODEL DRIVEN SAMPLING

9. CONTEXT SENSITIVE MODEL DRIVEN SAMPLING

The context we used in this thesis is defined or given by a domain expert. It usually refers to a certain scenario or environment which can be used to partition the original model to reduce the problem complexity and to fine tune the model. Context sensitive MDS is an example of domain knowledge based MDS. Context sensitive MDS is to build sub models based on the contexts in a complicated problem and generate data from the sub models. The reason is that it is hard to correctly describe a complicated problem using a single model; instead, building sub models for individual contexts can effectively and efficiently model the problem.

9.1 CONTEXT SENSITIVE MODEL

In hospitals, for leucocythemia patients, if they receive marrow transplants operation, they have a chance of 50% to survive with good care, and 50% chance to die with life extension without good care. But if they do not receive marrow transplants operation, they will die for sure in a short time.

Considering the above made up example, the context is "marrow transplants operation" for predicting patients' recovery. If we know the status of the context - marrow transplants operation, then the sub models can be built for each case instead of building a full model without considering the context. The advantage is that, the sub models are less complicated and more accurate then the model built without context. This is because that, often, the context information itself can give fully accurate prediction about the outcomes. For example, if we know the patient did not receive marrow transplants operation, we are then one hundred percent sure that this patient will die.

We build context sensitive models using Bayesian networks. Context sensitive Bayesian Networks can be represented in multiple methods, such as Bayesian multinets, similarity networks [53], tree structure [15] and context sensitive network [77] etc. Detailed technical information about context sensitive Bayesian network methods are summarized in Appendix C.4.

9.2 CONTEXT IN IMBALANCED DATA

In contrast to simplifying representation structures in Bayesian Network, a good context in imbalanced data set shall be able to reduce the imbalance ratio or data complexity and therefore produce a better performance. Specifically, there are two criteria for choosing a good context – either by reducing data imbalance level or by reducing problem complexity. In order to reduce the data imbalance level, the context shall be able to physically split the training data into two smaller data sets, such that the imbalance level

is substantially reduced in one data, while the other data set contains negligible minority cases; In order to reduce the problem complexity, the context shall be able to logically divide the data into two data sets, such that data complexity is reduced in both data sets.

For example in the sphere data shown in Figure 9-1 and Table 9-1, the total sample space has an imbalance ratio of 4%. There exists a context C which splits the data into two – data A under context C and data B under non context C. The imbalance ratio is 38.1% for data A, and is 0.3% for data B. The context C can help us model data A and B separately. Since the sample space for data A is much less imbalanced (38.1%) comparing to the original data (4%), it should be much easier to predict the minorities. Even though we may lose predictions on the minorities inside data B, we can still get a very good overall accuracy, as the minorities in data B is small.

The above example showed a context that can physically divide the training space into small portions and therefore make the sub models more adapted to each scenario. As shown in the following sessions, the context can logically divide the training space and reduce the concept complexity without reducing the data size.

9.3 DATA SETS

The data sets we used include sphere data, asthma first visit data and asthma sub visit data. In sphere data, we illustrate the context that can divide the training space to build sub models adapted to different local scenarios. In asthma first visit data, the context can separate the training space into two sub spaces, where sub models are used to generate

synthetic data to build a combined MDS model. In asthma sub visit data, the context we used does not reduce the training space; instead it can reduce the concept complexity to improve the performance.

9.3.1 SIMULATED DATA

As shown in Figure 9-1, the minority data spreads around the sphere and the majority data either spreads inside or outside the sphere. The imbalance ratio of the total space is 4%. In the context of upper sphere, the imbalance ratio is 0.381, and in the context of lower sphere, the imbalance ratio is 0.003.

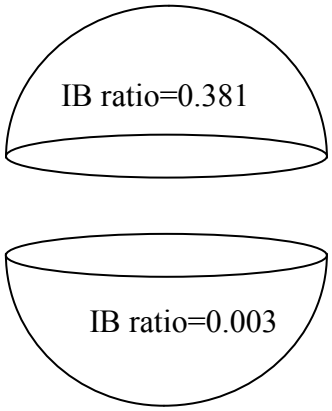


Figure 9-1 Simulated Context Specific Data

Table 9-1 Data samples of the sphere

	Minority	Majority
Total sample space	40	1000
Context - upper half sphere	37	60
Context - lower half sphere	3	940

The data in the lower half sphere contain very few minority samples, but with most majority samples. The data in the upper half sphere contain most of the minority samples, but with fewer majority samples.

9.3.2 ASTHMA FIRST VISIT DATA

The context we used in asthma first visit data is whether the patient takes theophyline or not. As shown in Table 9-2 and Figure 9-2, the imbalance ratio for the sub data under the context of "theophyline" is slightly decreased. Although the imbalance ratio for data under context "without theophyline" increased, the data set is too small to build a meaningful model.

Table 9-2 Asthma first visit data distribution w/o context

	Positive	Negative	Imbalance Ratio
No Context	213	729	0.226
Theophyline = yes	190	696	0.214
Theophyline = no	23	33	0.41

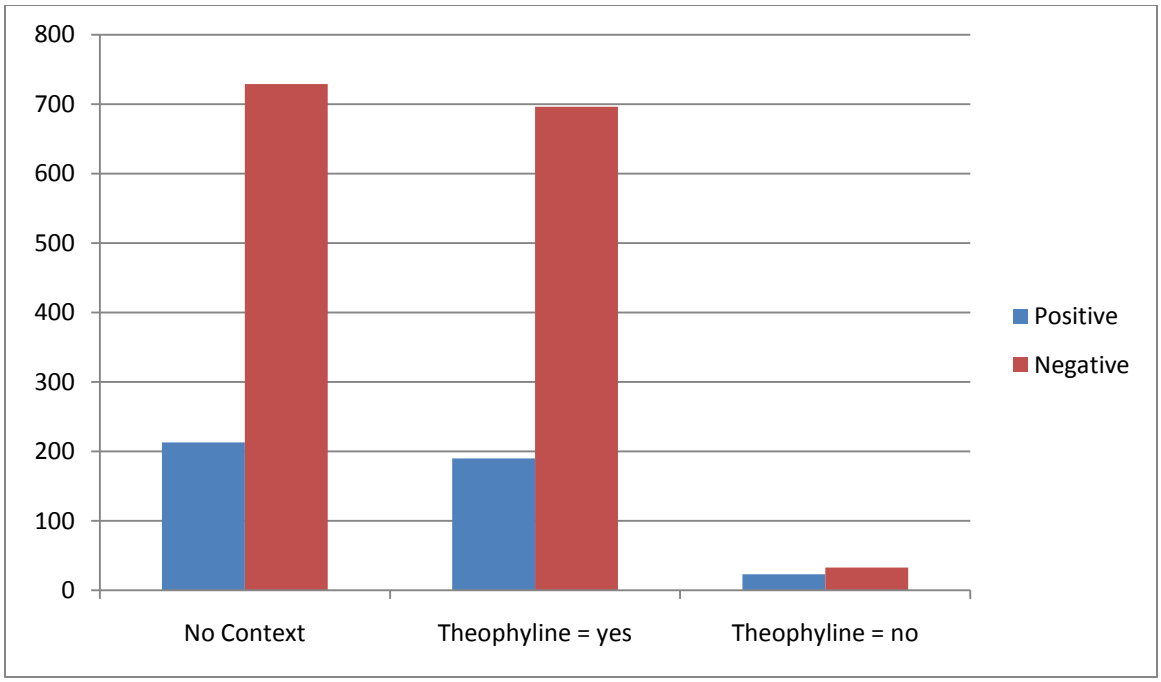


Figure 9-2 Asthma first visit data distribution with context

9.3.3 ASTHMA SUB VISIT DATA

The context in this case study is the outcome measure – urgent nebulization or hospitalization. These contexts share the same training space. They partition the training space logically instead of partitioning the sampling space physically. By separating these two outcome measures, the training space becomes more precise for each individual outcome measure as shown in Table 9-3 and Figure 9-3.

Table 9-3 Asthma sub visit data distribution w/o context

	Positive	Negative	Imbalance Ratio
No Context	1247	4047	0.236
Context = Hospitalization	973	4321	0.184
Context = Unebulization	1241	4053	0.234

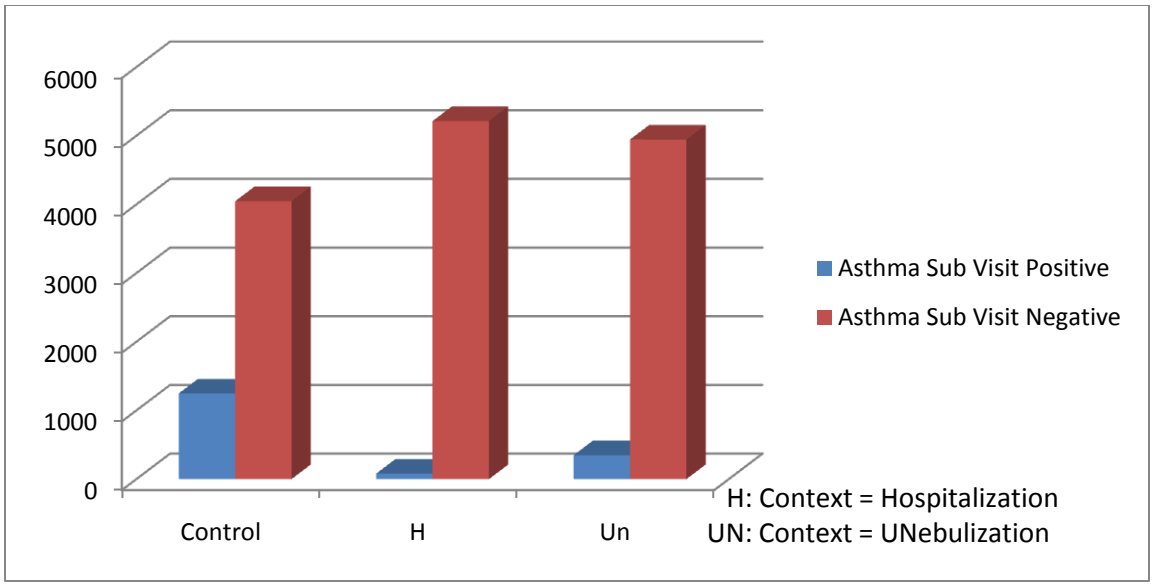


Figure 9-3 Asthma subsequent visit data distribution with context

9.4 EXPERIMENT DESIGN

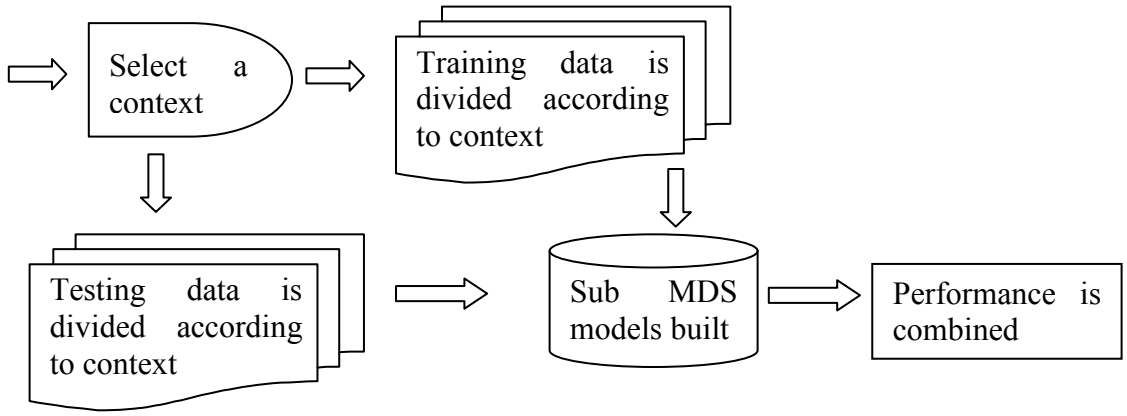


Figure 9-4 Work flow for context sensitive sampling

The workflow for context sensitive sampling is shown in Figure 9-4. First we need to select a context, which can separate the data as much as possible, and decrease the imbalance level of at least one generated data as much as possible. Training space can

then be divided according to the context to build sub models. The sub models can be combined to run on the testing space under different contexts. The algorithm for context sensitive MDS is as following:

Context Sensitive MDS algorithm

Context C is selected;

Training data is divided into two parts – TD under C, and TD' under C' (negation of C)

IF both TD and TD' are significantly large enough

THEN MDS Models built separately for TD and TD'

AND run the testing data for context C and C'

OHTERWISE sub models built for TD and TD' to form one combined MDS model

AND run the testing data

As shown in the algorithm, sub MDS models can only be built when there are enough training data in sub training data TD and TD' divided according to the context. If any of the training data are not significant enough to build a model, then one combined MDS is built instead, making use of the sub models to generate data. The data are

generated by the sub models proportionally to the amount of minority data inside TD and TD'.

9.5 EXPERIMENTAL RESULTS

9.5.1 SPHERE DATA

Without considering the context, the total sample space has a highly skewed distribution with imbalance ratio of 4%. The g-Mean value for our Bayesian net classifier without sampling is 0.543; and the g-Mean value for model driven sampling is improved to 0.624.

Table 9-4 Results without context

No sampling		Model Driven Sampling		
a=sphere	b=others	a=sphere	b=others	Actual Class
12	28	16	24	A=sphere
18	982	28	972	B=others
TP=0.3 TN=0.982 g-mean=0.543		TP=0.4 TN=0.972 g-mean=0.624		Results

As shown in Table 9-1, if we consider the context of upper sphere and under sphere, the data distribution for upper sphere will be relatively balanced; However, the data distribution for under sphere is extremely skewed, because most of the minority data is in upper sphere.

The upper sphere is relatively balanced, thus it has good performance with g-Mean value equal to 0.84. Model driven sampling slightly improves the accuracy of minority predictions, but drops in majority predictions. So overall, model driven

sampling cannot improve the performance when the data set is relatively balanced as shown in Table 9-5. The under sphere is extremely imbalanced, and the minority values are unpredictable with or without sampling as shown in Table 9-6. This is because the minority has too few values to be meaningful, and they are considered as noise in the classifier.

Table 9-5 Running results for upper sphere

No sampling		Model Driven Sampling		
a=sphere	b=others	a=sphere	b=others	Actual Class
28	9	31	6	A=sphere
4	56	21	39	B=others
TP=0.757 TN=0.933 g-mean=0.84		TP=0.838 TN=0.65 g-mean=0.738		Results

Table 9-6 Running results for under sphere

No sampling		Model Driven Sampling		
a=sphere	b=others	a=sphere	b=others	Actual Class
0	3	0	3	A=sphere
0	940	0	940	B=others
TP=0 TN=1 g-mean=0		TP=0 TN=1 g-mean=0		Results

Table 9-7 Running Results for total sphere with context

No sampling		Model Driven Sampling		
a=sphere	b=others	a=sphere	b=others	Actual Class
28	12	31	9	A=sphere
4	996	21	979	B=others
TP=0.7 TN=0.996 g-mean=0.834		TP=0.775 TN=0.979 g-mean=0.871		Results

As shown in Table 9-7, the overall performance for the whole data set using context sensitive learning is 0.834 which is highly improved comparing to the result of 0.543 without context. Context MDS can improve the performance from 0.624 to 0.871.

9.5.2 ASTHMA FIRST VISIT DATA RESULTS

Since the sub data shown in Table 9-3 and Figure 9-3 either get more imbalanced or are not significant enough to build an accurate model. We build two sub models from the two data sets, and generated synthetic samples to build a combined MDS model. The synthetic data are generated proportionally to the size of minorities in the sub training data. A MDS model is built on top of the synthetic data. Since the sub models are customized to their local context, the synthetic data created should be more relevant to the context. The result shown in Table 9-8 is slightly improved over the data generated without context with the highest performance of 0.685 as shown in Chapter 7.

Table 9-8 Confusion matrix for context sensitive MDS in asthma first visit data

Actual = control	Predicted = failure		
156	57	Actual = control	TP= 0.653
251	478	Predicted = failure	TN= 0.628
			G-Mean= 0.693

9.5.3 ASTHMA SUB VISIT DATA RESULTS

The data distributions for asthma subsequent visit with respect to context of different types of emergency department visit is shown as in Table 9-3. Although the sub data imbalance level is not increased and the sub data size is not reduced, the concept complexity does drop. Instead of using a combined outcome measure, now each sub data

has its own precise outcome measure. To combine the results from the sub data sets, we check the predictions for each sample from both sub models. By definition of the asthma control failure as shown in section 7.1, if either prediction is true, then the sample is predicted to be true in the total space, otherwise the sample is false. As shown in Table 9-9, context sensitive MDS does increase the performance to 0.76 comparing with the performance of 0.725 without context (the highest MDS score for asthma sub visit data as shown in Chapter 7).

Table 9-9 Asthma subsequent visit data's performance with context

Context= Hospitalization		Context= Unebulization		Combined results		
a= positive	b= negative	a= positive	b= negative	a= positive	b= negative	Actual Class
651	322	711	530	863	384	A=positive
194	4127	308	3745	664	3383	B=negative
TP=0.669 TN=0.955 g-mean=0.799		TP=0.573 TN=0.924 g-mean=0.728		TP=0.692 TN=0.836 g-mean=0.76		Results

9.6 DISCUSSIONS

In this chapter, we have described three different types of context sensitive MDS methods. They are empirically shown to improve the overall performance by isolating the minority data into a much smaller data space, or by producing a smaller and more precise model to generate synthetic data, or by reducing the concept complexity.

It is essential to select a good context. The important characteristic of a good context is that it can partition the sampling space, to produce smaller sub training spaces,

with more concrete concepts in each sub training space. For the three data sets we have examined, the contexts for sphere data and asthma sub visit data are relatively good, they can improve the system performance substantially. However, for asthma first visit data, the context is not well chosen, and one of the separated spaces is not significant comparing to the overall training space, therefore the improvement is negligible.

CHAPTER 10: CONCLUSIONS

10. CONCLUSIONS

10.1 REVIEW OF EXISTING WORK

In this thesis, we have reviewed existing approaches in imbalanced data learning, both on algorithm level approaches and data level approaches. We focused on the more promising data sampling approaches. The popular approaches include random sampling, SMOTE sampling, and progressive sampling. Random sampling creates duplicated data to bias to the minority. SMOTE sampling creates synthetic data using the nearest neighbor to bias to the minority. Progressive sampling finds the near optimal distributions using any sampling method. Random sampling and SMOTE sampling focus on how to sample data, while progressive sampling is a method telling how much to sample. Progressive sampling generally can be applied to other data sampling methods.

Existing approaches make use of only the data sample itself or its nearest neighbor, while in reality, with consideration of other data samples, we can generate more meaningful data. In real life experiments, we usually have domain experts' input in addition to the training data space. However, as far as we have seen, none of the existing approaches ever make use of the domain expert knowledge, such as experts' input, context information etc. to help data generation.

10.2 COUNTRIBUTIONS

In this thesis, we have proposed a model driven sampling approach (MDS). MDS creates synthetic data from the model built from the whole training space and domain knowledge. MDS has been empirically shown to be performing better than other approaches in most cases. Even in the worst case scenario, it performed comparably to the existing best approach.

10.2.1 THE GLOBAL SAMPLING METHOD

MDS is a global sampling approach. Existing approaches mostly either make replications or make use of its nearest neighbors – the local knowledge to generate data. Data sampled from local sampling methods are often not accurate. However, it is not trivial to sample data directly from the whole training space. In this thesis, we use probabilistic graph to model the whole training space and then generate synthetic data from the model thereafter. The data sampled from the global model can better simulate the true reality. We have experimentally shown that MDS generally performed better than other sampling approaches including random sampling and SMOTE sampling methods, on both the simulated data sets and the real life data sets.

10.2.2 MDS WITH DOMAIN KNOWLEDGE

Existing sampling approaches mainly make use of only the training data to generate synthetic data. MDS, however, can also use domain experts' knowledge to generate synthetic data. Bayesian network allows probabilistic uncertainties to be represented in a

graphical structure. The model built is explicit to experts allowing domain knowledge to be more readily combined into the model. For example, experts can make necessary changes to the structure and conditional probability table according to their experience, which is particularly useful when the training space is limited or noisy. By integrating domain knowledge into the model, the data generated becomes more accurate. It is extremely useful for sparse data with high imbalance ratios, or for cases without enough training data – absolute rarity. Domain knowledge can make up for the lack of data and can usually help build a more accurate model.

Domain experts can also provide the "context" information to the model. Context sensitive Bayesian network allows MDS to create models separately under different contexts. The sub models created are normally more concrete and more adapted to its context. A good context sensitive model can reduce the system complexity; meanwhile it can improve the system efficiency and accuracy by adapting to its local environment.

In this dissertation, domain knowledge based MDS has been empirically shown to outperform other sampling approaches in various situations. However, from our experimental results, domain knowledge based MDS may not necessarily outperform the MDS method, due to possible deficiencies in the domain knowledge. Therefore, correctly selecting good domain knowledge is very important in the model creation step.

10.2.3 MDS COMBINED WITH PROGRESSIVE SAMPLING

Progressive sampling is an effective way in determining the optimal data distributions for data sampling. MDS can be combined with the progressive sampling method. The synthetic data generated from the model can be incrementally added to the training space until an optimal data distribution is discovered. Progressive MDS can guarantee an optimal data distribution found for MDS, instead of using the balanced data distribution which may not be optimal. As shown in our experiments, most of the data reached their optimal performance with imbalance ratio less than 50% (balanced data distribution).

10.2.4 CONTEXT SENSITIVE MDS

One type of very useful domain expert knowledge or knowledge from literature is the context in a training space. Context sensitive MDS can make use of context sensitive Bayesian network to build models. We have shown that three different types of contexts can be applied and they did improve the system performance over the cases without contexts. Context sensitive MDS can simplify the problem by building smaller but more accurate models under various contexts. Therefore, the synthetic data created is more specific under a certain context and thus is more accurate and meaningful. It has been shown in asthma sub visit data that context sensitive MDS can decompose the training space logically, instead of reducing the training data size, to reduce the sub models' complexity.

10.3 LIMITATIONS

MDS assumes that the balanced data distribution is the optimal data distribution for our targeted imbalanced data set, which is often inaccurate in real cases. We have shown in our experiments on progressive MDS that, the optimal data distributions for most data sets we experimented on are not balanced.

MDS also assumes that the model built from the training space or domain knowledge is reasonably accurate. However, noisy training space or noisy experts' knowledge often result in noisy models, and therefore the synthetic data created might also contain a lot of noise. This might in turn degrade the system performance.

10.4 FUTURE WORK

Future work includes testing our work and adapting it to real life clinical usage; further research is needed in context sensitive model driven sampling and a comprehensive context sensitive system should be able to minimize the workload and produce better synthetic data. Model correctness checking in MDS and MDS with domain experts' interactions are also potential research areas especially in clinical domains.

10.4.1 FUTURE WORK IN ASTHMA PROJECT

Rather than using traditional statistical analysis, we have used machine learning method combining domain knowledge and training data to create a knowledge model which could allow physicians to view and modify it explicitly. The synthetic data created

improved the predictions for minorities and the overall performance. In future, we aim to integrate our work into a platform for the practical usage in asthma treatment. A more detailed study using ROC curves as the evaluation metrics is necessary, to help identify the tradeoff between the positive prediction rate and the negative prediction rate.

10.4.2 FUTURE WORK IN MDS

Although we have used a mixed expert approach for model selections in MDS, the model correctness is not checked. In future work, we can design a model checking and verification mechanism in MDS, to make sure that the synthetic data generated are reasonably clean and correct.

Context sensitive Bayesian network is a relatively new research area, so is the context sensitive MDS. In future work, we can make use of the adaptive context sensitive Bayesian network (e.g. context sensitive network - CSN in [77]) to build adaptive context sensitive model driven sampling system. Further exploration of context sensitive model driven sampling should make sampling more efficient and effective. The synthetic data created should be more adapted to their local environments.

Knowledge based model driven sampling will be further studied, especially in clinical domains. Knowledge acquisition, knowledge representation, and domain experts' interaction in model driven sampling also need further research work.

Another research direction in MDS is to systematically combine context sensitive MDS with progressive sampling. Making use of the latest progressive sampling

approaches [147] could enable us to develop a progressive, context sensitive and adaptive MDS system that can better manage the imbalanced data problems.

APPENDIX A: ASTHMA FIRST VISIT ATTRIBTUES

Chi-Square Value	Serial number	Attribute
107.7204275	109	Nebulisation Count
58.31685305	124	Oral Steriods Count
39.67437921	137	DrugSubvention
24.41403192	1	Patient's Record No
23.8157766	101	MC
23.19437045	112	UNebulisation Freq
20.5933934	100	Asthma Duration
16.90825091	79	Change PDrugs LAB2
15.26578832	39	Trigger Factor Haze
13.91803023	62	Activity stopped
13.08053925	80	Change PDrugs Others
12.4672092	8	In Attendance Doctor
12.42476301	52	InhalerTurbuhaler
12.07582708	70	GINA
11.59475463	76	Change PDrugs Long Acting Theophyline
11.21030447	104	Intubations
10.95772934	60	Days with wheeze/cough/SOB
10.83167101	84	Change PDrugs ICS+LABA Dosage
10.70129567	46	Compliance Medication
9.778479921	108	Hospitalisation Count
9.237047006	77	Change PDrugs Oral Steroids
9.023499665	7	In Attendance Nurse
8.621066197	44	Trigger Factor Stress
8.565699306	118	UNebulisation Loc6 Hospital
8.514731932	113	UNebulisation Loc1 Home
7.80473233	129	Smoking Years
7.613835636	16	Current PDrugs SAB2
7.571389379	54	InhalerAccuhaler
7.321208867	9	Refer Source
7.257758541	18	Current PDrugs Others
7.156944846	47	Inhaler Techniques Skills
6.933595589	59	Absent from School
6.745369375	117	UNebulisation Loc5 GP
6.675081521	102	Fatal Asthma

6.41598551	92	Change PDrugs ICS+LABA Dosage Freq
5.831843595	107	Hospitalisation
5.153821328	37	Trigger Factor House dust
5.119909533	41	Trigger Factor Change Weather
4.977570285	119	UNebulisation Loc7 SAF
4.558494836	131	Quit Smoking Years
4.460246116	86	Change PDrugs Oral Steroids Dosage
4.312186356	15	Current PDrugs Oral Steroids
4.29671566	4	Patient's Race
4.201291251	38	Trigger Factor Animal dandens
4.049825363	63	Spirometry
3.911923349	11	Current PDrugs Budesonide
3.812551403	61	Nights with wheeze/cough/SOB
3.725273357	94	Change PDrugs Oral Steroids Dosage Freq
3.609613787	106	Intubations date
3.203153075	132	Remarks
2.988782459	98	Reinforcement by Asthma Nurse
2.74443775	13	Current PDrugs ICS+LABA
2.69501443	97	Written Action
2.358558676	121	Oral Steroid Use
2.357469532	111	UNebulisation
2.250441328	128	Cigarettes
2.244121461	14	Current PDrugs Long Acting Theophyline
2.015195733	50	InhalerMDI
1.828543773	82	Change PDrugs Budesonide Dosage
1.764324153	72	Change PDrugs BDP
1.681304796	75	Change PDrugs ICS+LABA
1.651949062	115	UNebulisation Loc3 MOPD
1.605494276	51	InhalerMDISkills
1.588585301	73	Change PDrugs Budesonide
1.478529795	20	Current PDrugs BDP Dosage
1.468701003	99	Next Visit
1.468701003	133	Patient Discharge
1.465246483	10	Current PDrugs BDP
1.38665092	55	InhalerAccuhalerSkills
1.376995488	17	Current PDrugs LAB2
1.282882364	114	UNebulisation Loc2 EMD
1.255279905	116	UNebulisation Loc4 Polyclinic
1.186615958	45	Trigger Factor Others
0.970255631	21	current PDrugs Budesonide Dosage

0.932169215	71	Change in Treatment
0.91504215	48	Today's PEFR
0.892796719	23	Current PDrugs ICS+LABA Dosage
0.859182307	74	Change PDrugs Fluticasone
0.777314069	3	Patient's Sex
0.71585618	19	Current PDrugs Others Define
0.590858913	127	Smoking
0.577066923	130	Quit Smoking
0.564997255	53	InhalerTurbuhalerSkills
0.498920964	78	Change PDrugs SAB2
0.493367966	85	Change PDrugs Long Acting Theophyline Dosage
0.471429603	12	Current PDrugs Fluticasone
0.464708952	136	Default
0.448926757	110	Nebulisation Date
0.437660255	103	Fatal Asthma Specify
0.393656292	126	Sinusities
0.385928934	25	Current PDrugs Oral Steroids Dosage
0.375894493	87	Change PDrugs LAB2 Dosage
0.354808136	58	InhalerOthersSpecify
0.336429655	29	current PDrugs Budesonide Dosage Freq
0.298278389	28	Current PDrugs BDP Dosage Freq
0.289745031	49	Device
0.238122963	95	Change PDrugs LAB2 Dosage Freq
0.235148392	43	Trigger Factor Exercise
0.225077009	105	Intubations Count
0.169984291	40	Trigger Factor Household Smoking
0.155377383	42	Trigger Factor Food
0.135891343	88	Change PDrugs Others Dosage
0.082514262	56	InhalerOthers
0.080887704	96	Change PDrugs Others Dosage Freq
0.05691553	24	Current PDrugs Long Acting Theophyline Dosage
0.053574973	31	Current PDrugs ICS+LABA Dosage Freq
0.045562091	57	InhalerOthersSkills
0.041694735	36	Trigger Factor Respiratory Infections
0.036525854	33	Current PDrugs Oral Steroids Dosage Freq
0.0341245	5	Patient's Race Others Define
0.033972836	123	LT Oral Steroids Dose
0.023832984	32	Current PDrugs Long Acting Theophyline

		Dosage Freq
0.018199733	93	Change PDrugs Long Acting Theophyline Dosage Freq
0.018199733	22	Current PDrugs Fluticasone Dosage
0.017648226	26	Current PDrugs LAB2 Dosage
0.01678635	90	Change PDrugs Budesonide Dosage Freq
0.004853262	34	Current PDrugs LAB2 Dosage Freq
0.001632054	120	UNebulisation Loc8 Others
8.67E-04	30	Current Pdrugs Fluticasone Dosage Freq
7.87E-04	122	LT Oral Steroids
0	135	Patient Discharge Loc
0	6	Email
0	2	Hospital Database
0	83	Change PDrugs Fluticasone Dosage
0	69	FVC Predicted
0	81	Change PDrugs BDP Dosage
0	125	LT Oral Steroids DosePRN
0	134	Patient Discharge Date
0	89	Change PDrugs BDP Dosage Freq
0	91	Change PDrugs Fluticasone Dosage Freq
0	64	FEV1 Pre
0	27	Current PDrugs Others Dosage
0	35	Current PDrugs Others Dosage Freq
0	67	FVC Post
0	68	FEV1 Predicted
0	65	FVC Pre
0	66	FEV1 Post

APPENDIX B: ASTHMA SUBSEQUENT VISIT ATTRIBUTES

Chi-square Value	Serial Number	Attribute
66.839159	111	MV Followup Wks
63.418969	48	MV UNebulisation
61.116986	58	MV Events
59.437961	70	MV Nights with wheeze/cough/SOB
58.913096	69	MV Days with wheeze/cought/SOB
57.808841	71	MV Activity stopped
54.304761	68	MV Absent from School
35.975015	82	MV Change PDrugs Budesonide
32.08058	93	MV Change PDrugs ICS+LABA Dosage
31.128854	19	MV Current PDrugs ICS+LABA Dosage
28.881451	88	MV Change PDrugs LAB2
26.517257	1	Patient's Record No
23.287515	72	MV GINA
22.289848	61	MV Event Loc2 EMD
21.421383	110	MV Next Visit
20.540627	112	MV Patient Discharge
20.065187	51	MV UNebulisation Loc2 EMD
18.990223	84	MV Change PDrugs ICS+LABA
18.965559	14	MV Current PDrugs LAB2
18.943687	53	MV UNebulisation Loc4 Polyclinic
16.38451	6	MV Doc Attend
15.807197	64	MV Event Loc5 GP
15.791881	54	MV UNebulisation Loc5 GP
15.503807	86	MV Change PDrugs Oral Steroids
14.614197	63	MV Event Loc4 Polyclinic
14.224848	3	MV Visit Number
13.439703	89	MV Change PDrugs Others
10.258882	57	MV UNebulisation Loc8 Others
9.906678	55	MV UNebulisation Loc6 Hospital
9.03152	85	MV Change PDrugs Long Acting Theophyline
7.892892	8	MV Current PDrugs Budesonide
7.413237	11	MV Current PDrugs Long Acting Theophyline

7.022468	108	MV Reinforcement by Asthma Nurse
6.986097	50	MV UNebulisation Loc1 Home
6.306862	56	MV UNebulisation Loc7 SAF
6.036752	107	MV Written Action
6.015504	60	MV Event Loc1 Home
5.900364	65	MV Event Loc6 Hospital
5.602406	38	MV InhalerAccuhaler
5.58369	10	MV Current PDrugs ICS+LABA
5.202452	15	MV Current PDrugs Others
4.668484	44	MV Hospitalisation
3.491643	66	MV Event Loc7 SAF
3.458807	13	MV Current PDrugs SAB2
3.354906	32	MV Compliance Medication
3.131694	25	MV Current PDrugs Budesonide Dosage Freq
2.622827	17	MV Current PDrugs Budesonide Dosage
2.607965	36	MV InhalerTurbuhaler
2.480872	102	MV Change PDrugs ICS+LABA Dosage Freq
2.377684	12	MV Current PDrugs Oral Steroids
2.051057	91	MV Change PDrugs Budesonide Dosage
1.996892	34	MV InhalerMDI
1.484064	33	MV Inhaler Techniques Skills
1.407848	46	MV Nebulisation Count
1.352479	5	MV Nurse Attend
1.285623	87	MV Change PDrugs SAB2
1.031819	100	MV Change PDrugs Budesonide Dosage Freq
0.893954	59	MV Events Freq
0.738605	95	MV Change PDrugs Oral Steroids Dosage
0.686344	80	MV Change in Treatment
0.627169	49	MV UNebulisation Freq
0.588005	37	MV InhalerTurbuhalerSkills
0.569995	109	MV Physical Signs
0.450508	7	MV Current PDrugs BDP
0.373848	18	MV Current PDrugs Fluticasone Dosage
0.358638	52	MV UNebulisation Loc3 MOPD
0.33091	27	MV Current PDrugs ICS+LABA Dosage Freq
0.282603	67	MV Event Loc8 Others
0.271822	83	MV Change PDrugs Fluticasone
0.215102	62	MV Event Loc3 MOPD
0.177185	101	MV Change PDrugs Fluticasone Dosage Freq
0.176091	81	MV Change PDrugs BDP

0.16104	42	MV InhalerOthersSpecify
0.158453	39	MV InhalerAccuhalerSkills
0.115186	28	MV Current PDrugs Long Acting Theophyline Dosage Freq
0.113611	4	MV VisitDefault
0.095503	20	MV Current PDrugs Long Acting Theophyline Dosage
0.094022	92	MV Change PDrugs Fluticasone Dosage
0.087734	104	MV Change PDrugs Oral Steroids Dosage Freq
0.086523	26	MV Current PDrugs Fluticasone Dosage Freq
0.082263	105	MV Change PDrugs LAB2 Dosage Freq
0.062627	97	MV Change PDrugs LAB2 Dosage
0.03898	103	MV Change PDrugs Long Acting Theophyline Dosage Freq
0.037918	94	MV Change PDrugs Long Acting Theophyline Dosage
0.024218	47	MV Nebulisation Date
0.019744	22	MV Current PDrugs LAB2 Dosage
0.018761	106	MV Change PDrugs Others Dosage Freq
0.012412	98	MV Change PDrugs Others Dosage
0.009939	73	MV Spirometry
0.008972	30	MV Current PDrugs LAB2 Dosage Freq
0.008073	41	MV InhalerOthersSkills
0.005747	40	MV InhalerOthers
0.00503	21	MV Current PDrugs Oral Steroids Dosage
0.004325	16	MV Current PDrugs BDP Dosage
0.00391	23	MV Current PDrugs Others Dosage
0.003699	113	MV Patient Discharge Loc
0.002999	45	MV Hospitalisation Count
0.002011	9	MV Current PDrugs Fluticasone
0.001362	29	MV Current PDrugs Oral Steroids Dosage Freq
0.001322	35	MV InhalerMDISkills
0.000757	90	MV Change PDrugs BDP Dosage
0.000673	24	MV Current PDrugs BDP Dosage Freq
0.000605	31	MV Current PDrugs Others Dosage Freq
0	2	Hospital Database
0	114	MV DiagnosisDeath
0	77	MV FVC Post
0	76	MV FEV1 Post
0	79	MV FVC Predicted

0	78	MV FEV1 Predicted
0	99	MV Change PDrugs BDP Dosage Freq
0	43	MV Today's PEFR
0	75	MV FVC Pre
0	96	MV Change PDrugs SAB2 Dosage
0	74	MV FEV1 Pre

APPENDIX C: RELATED WORK - BAYESIAN NETWORK

In this appendix, we mainly discuss about the technology basis used in this dissertation, which is typically the Bayesian network [96, 115]. We discussed the Bayesian network learning including structure learning and parameter learning for building a Bayesian network model. We also introduced context sensitive Bayesian networks and discussed different sampling techniques in Bayesian network which can be used for data generation.

C.1. STRUCTURE LEARNING

Bayesian Network structure can be constructed from domain knowledge manually. There are generally two ways to automate the process of constructing a BN from knowledge base – one is score based method, and another is constraint based method.

In a score based approach, we can define our own model selection criteria. Learning a network structure can be considered as an optimization problem where a quality measure of a network structure given the knowledge base must be maximized according to our criteria. Some searching methods in a score based approach are Greedy Search, K2 [32], MCMC [115], etc.

Different from score based approach, constraint based approach mainly uncovers BN causal structure by conditional independence tests. The assumption is that there exists a network structure that exactly represents independencies in a system. It follows that if there is no edge between two variables, then a conditional independence can be identified

from knowledgebase. Once all edges are located, the directions of an edge can be adjusted so that the conditional independencies can be properly represented. Constraint based methods are based on the following assumptions: 1) Causal sufficiency: There are no hidden variables in the domain which are parents of observed variables; 2) Causal Markov: given present, future is independent of past. Popular available methods are PC algorithm, IC algorithm, etc.

C.2. PARAMETER LEARNING

Given BN structure is known, there are two categories of parameter learning problems – learning from complete data and learning from incomplete data.

Learning parameters from complete data and known structure is straightforward given that all parameters in the domain are independent. The close form solution can update each parameter values independently, e.g. Maximum Likelihood Estimation (MLE) [102]. Learning parameters from incomplete data is under the Missing-At-Random (MAR) assumption where missing values or patterns are dependent on the observed variable values. Obviously, when data is missing, parameters are not independent any more. There is no closed form solution in this case. Approximate methods including Expectation maximization (EM), Mont Carlo methods etc are used instead.

C.3. CONSTRUCTING FROM DOMAIN KNOWLEDGE

Domain knowledge is always useful in improving the machine learning models. It is especially useful for constructing Bayesian networks. Both the structures and conditional probability tables can be inferred directly from the domain knowledge [62, 90, 108].

To construct a Bayesian network from domain knowledge, there are three assumptions:

i. All variables are known in advance – the variables in the Bayesian network are determined;

ii. Domain knowledge can readily assert the causal relationships (typically correspond to the assertions of conditional dependencies) between variables – the edges in the Bayesian network can be determined by domain knowledge;

iii. The values of conditional probabilities can be estimated from domain knowledge.

Constructing Bayesian networks completely from domain knowledge is generally achieved in three main steps [36]:

i. Determine the number of variables and the meaning of these variables in the domain of interest;

ii. Determine whether there exist direct causal influence relationships between the variables in the domain; and

iii. Determine the conditional probability distributions given the structure of the Bayesian network from the first two steps.

Quite a few Bayesian networks have been constructed in this way, e.g. QMR-DT [99]. Various methods have been proposed to construct Bayesian networks with causal domain knowledge [36, 63].

C.4. CONTEXT SENSITIVE BAYESIAN NETWORK

C.4.1. CONTEXT DEFINITION IN BAYESIAN NETWORK

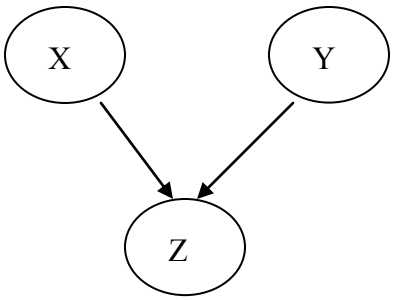
Bayesian networks have a lot of good properties in machine learning. However, there are certain properties that we cannot capture in Bayesian Network, for example, context specific independencies (CSI), i.e. given an assignment of a context variable, the networks structure can be much simplified. Context sensitive Bayesian network include Bayesian multinets [105], similarity networks [53], tree structure [15] and context sensitive network by Joshi, et al [77] etc.

Qualitatively, Bayesian Networks describe variable independencies – a variable is independent of its non-descendants given its parents. Quantitatively, Bayesian Networks represent probabilistic distributions that quantify inter-variable correlations. We specify a distribution by associating each note X a conditional probabilistic table (CPT) which represents conditional distribution of X given its parents. Let's consider the following example which demonstrates the deficiencies of Bayesian Network representation.

In hospitals, for leucocythemia patients, if they receive marrow transplants operation, they have a chance of 50% to survive with good care, and 50% chance to die with life extension without good care. But if they do not receive marrow transplants operation, they will die for sure in a short time.

As shown in Figure C-1, node X and Y are binary variables. Variable X represents *marrow transplants operation* (value 't' for receiving operation and 'f' for rejecting operation), variable Y represents care after operation to avoid virus infection (value 't' for good caring, 'f' for bad caring), and variable Z represents patient's status (value'z1' for "survival", 'z2' for "death with life extension", 'z3' for "death').

A CPT is usually in a tabular form, as shown in the figure. Since X, Y are binary variables, we need to specify four distributions for variable Z, which is exponential to the number of its parents. But if we examine the table carefully, we find that P(z|x, Y) = z3 when X=f regardless values of variable Y. So clearly, we need only three distributions rather than four, and the saving becomes essential when the network grows large [15].

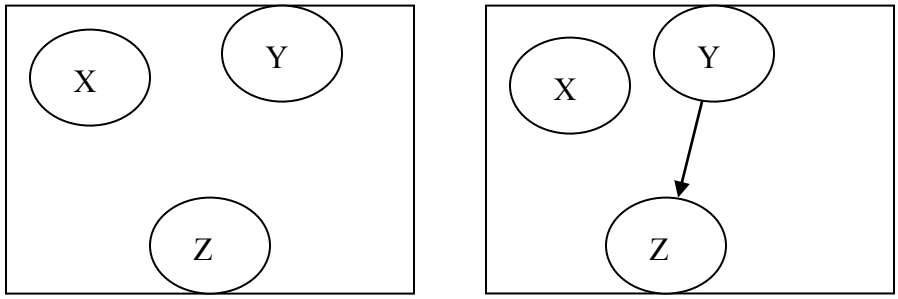


X	Y	P(z)
t	t	z1
t	f	z2
f	t	z3
f	f	z3

Figure C-1 Context Specificity in Bayesian Network

C.4.2. BAYESIAN MULTINET

Let P (C, U_1, … U_n) be a probability distribution, C be a context variable with values A_1, … A_k. If each graph D_i (1<=i<=k) corresponding to distribution P(U_1,… U_n| A_i) is a Bayesian Network, we say the set of all D_i (1<=i<=k) is a Bayesian multinet of P.



Rejecting operation Accepting operation

Figure C-2 A Bayesian multinet representation for leucocythemia example

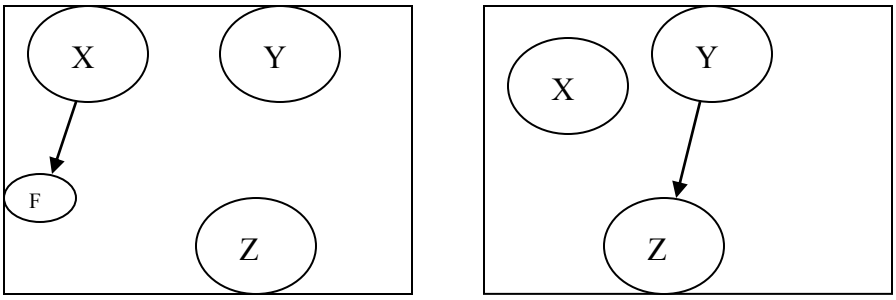
The Bayesian Network representation in Figure C-1 hides the fact that *leucocythemia patient*'s status is independent with the care received if he did not receive any operation. We can represent this example more explicitly by using two networks as shown in Figure C-2, whereby the first network represents a patient never receives

marrow transplants operation, and the second represents a patients has received the operation. Figure C-2 is obviously a much better representation than Figure C-1. It makes use of the context of whether a patient receiving marrow transplants operation and shows the dependence between care and patient's status only in the context of a patient receiving the operation. Also in Figure C-2, we only need three distributions rather than four. The saving can be substantially large when the network grows large, due to the fact that distributions grow exponentially with the number of variables growing, while the overhead of multi-network representations only grows linearly.

C.4.3. SIMILARITY NETWORKS

Bayesian multinet requires every variable to be in the local network, which adds inefficiencies as well as confusions in knowledge acquisition if some of the variables are not related to the hypothesis. On the other hand, we cannot simply eliminate those non-related variables from local networks, because doing so could dangerously lose valuable information. For example if we change values of variable X in the *leucocythemia* example as:

x1: reject operation because of no proper marrow match
x2: reject operation because of lack of money
x3: accept operation

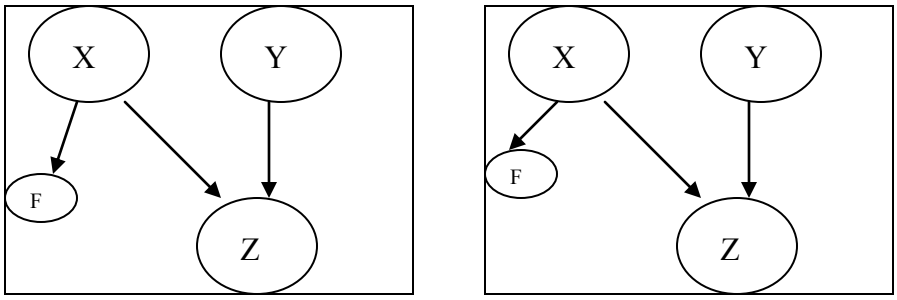


Operations rejected Operation accepted

Figure C-3 A similarity network representation

Then multinet representation of the example will become like in Figure C-3, (F is a binary variable with value either "this patient needs funding support from charity", or "this patient needs donation of proper matched marrow). So if we haphazardly remove unrelated nodes, this piece of information will be lost in the network. Similarity network [63] can help to resolve this problem.

A similarity network is very similar to a multinet, except that the nodes in each local network are only those that can "help to discriminate" the hypothesis under a certain context. "Context values" in all local networks are formed a connected cover of the value set of the context variable. The similarity network representation of the amended *leucocythemia* example is shown in Figure C-4.



x1: operation rejected because of lack of money
x3: operation accepted

x2: Operation rejected because of no marrow match
x3: operation accepted

Figure C-4 Similarity Network Representation of *leucocythemia*

A cover for a context variable C with value set C-set= $\{C_1, \ldots C_k\}$ means: the distributed union of context values that appeared in all local networks is C-set. A cover is connected if and only if all local networks can be connected by their common nodes with the context value. As shown in Figure C-4, $\{x_1, x_3\}$, and $\{x_2, x_3\}$ is a cover set of context value set. It is connected because it consists of the links x_1->x_3->x_2 to form a connected graph, and the two local networks are by chance the same in our example.

The advantages of similarity networks are: It tightens the local network representation much further comparing to multinet representation yet keeping all relevant information in the system, which is more efficient; By reducing irrelevant variables in local networks, confusing in knowledge acquisition from experts can be reduced; It prevents model builder to lose relevant information by forcing local networks a connected cover for context. For example, if the local networks of the similarity network is according to the cover set$\{x_1, x_2\}$, $\{x_3\}$, then the information why patients reject

operations would be lost. This property of similarity network was called *exhaustiveness* by Heckerman. Disadvantages of similarity network, connected cover set is not trivial, and it is rather tricky to select a good one.

C.4.4. TREE STRUCTURE REPRESENTATION

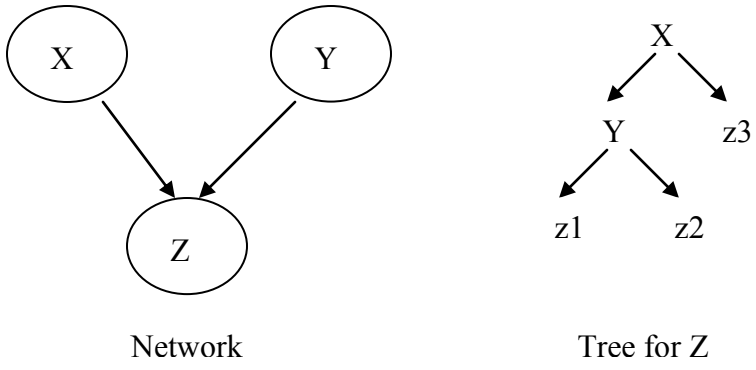


Figure C-5 Tree structure representation

Friedman et. al. [52] constructed a tree representation for CPT at variable Z in the simplest leucocythemia example as shown in Figure C-5. (Left arc represents true, and right arc represents false). In context ~x, clearly Y is rendered independent of Z. A path in the tree is the set of arcs from the roof node to a leaf node, the label of a path is the values of the variables occurring on that path, and a path is consistent with a context c if only if the label of a path is consistent with the assignment in context c. For example, in Figure C-5, path x->y->z_1 is consistent with context {x, y}. An edge is said to be redundant if the starting node is not lying on any path consistent with context c. For example, Y-> Z is redundant in the context of {~x}.

The advantage of tree representation is its naturalness and clearness, we can even directly read contexts information from paths of the tree; each path of a tree represents a distribution, and the distributions could be very compact after a proper pruning. However, the building of a tree can easily go exponential with large number of nodes, or each node with a large value set.

C.4.5. NATURAL LANGUAGE REPRESENTATION

Liem Ngo et. al. [106] defined a language for representing context-sensitive knowledge declarative semantics. The language can be abstracted as follows:

Type:
*P|P** => *Denoting probability distribution*
Ai| Bj|... => *Denoting atoms*
X |Cap_ini => *Denoting domain variable (Cap_ini = names start with capital letter)*
p | q => *Denoting Predicates*

Values:
Predicate = Context Predicate | Probabilistic Predicate
Context Predicate = True | False (deterministic)
Context Atom = c-atom (atom formed from context predicate)
Context literal (c-literal) = c-atom | ~c-atom
Context Base = { C0 <- L1, L2, ...Ln} type of {c-atom <- c-literal, ... c-literal}
Probabilistic Atom = p-atom (atom formed from probabilistic predicate)
...
KB = <PD, PB, CB, CR>

A knowledge base consists of four basic units, namely there are: predicate declarations (PD), probabilistic base (PB), Context Base (CB), and Combining Rules (CR). Consider the example in Figure 2, the language representation will be like:

PD = { Z(Some result: Result, V), VAL(Z)={survival, death with life extension, death};
Y(Some care: Care, V), Val(Y)={good, bad}
}
PB = { P(Y(y, good)) = .5
P(Y(y, bad)) = .5
P(Z(z, survival) =1 <- Y(y, good), accept_operation(X)
P(Z(z, death) =1 <- reject_operation(X)
P(Z(z, death with life extension) =1 <- Y(y,bad), accept_operation(X)
}

CB = { accept_operation(X) <- ~reject_operation(X)
reject_operation(X) <- ~accept_operation(X)}

C.5. INFERENCING

In previous sessions we introduced compact representations of probability distributions - Bayesian Networks. A network describes a unique probability distribution *P*, and there are a lot of queries that we could answer about *P*.

We use inference as a name for the process of computing answers to such queries. There are many types of queries that we might ask, most of which involve evidences. An evidence *e* is an assignment of values to a set *E* variables in the domain. Without loss of

generality, we can represent *E* by a subset ***E*** = *[Xk+1, ..., Xn]*. The simplest query is to compute the likelihood of evidence:

$$P(\boldsymbol{e}) = \sum_{x_1} \cdots \sum_{x_k} P(x_1, \ldots, x_k, \boldsymbol{e})$$

Most of the time, we are interested in the conditional probability of a variable given the evidence:

$$P(X \mid e) = \frac{P(X, e)}{P(e)}$$

Which is the posteriori belief in *X* given evidence *e.* This query is useful in many cases:

- **Prediction**: what is the probability of an outcome given the starting condition; Target is a descendent of the evidence.
- **Diagnosis**: what is the probability of disease/fault given symptoms; Target is an ancestor of the evidence
- **Data Sampling:** Generate data samples that are realizations of *P*(x) given the evidence. Target is all the nodes in the probabilistic distribution *P*(x) except of the evidence.

C.6. DATA SAMPLING METHODS

The objective of sampling is to generate samples from a learned probabilistic distribution *P*(x). Here, the sample generated from a distribution *P*(x) is a single realization of x whose probability distribution is *P*(x), instead of a collection of realizations x as in statistics. It is assumed that *P*(x) can be evaluated such that $P(x)=P^*(x)/Z$. But *P*(x) is too

complicated for us to sample from it directly. We assume that we have a simpler density $Q(\mathrm{x})$ which we can evaluate to within a multiplicative constant where $Q(\mathrm{x}) = Q^*(\mathrm{x})/Z_Q$, and from which we can generate samples. The expectation of a $P(\mathrm{x})$ is given by Equation C-1.

$$\Phi = \langle \phi(\mathrm{x}) \rangle \equiv \int d^N \mathrm{x} P(\mathrm{x}) \phi(\mathrm{x})$$

Equation C-1 Expectation of function $P(\mathrm{x})$

We used Figure C-6 to Figure C-8 similar to McKay et. al. [98] to introduce different sampling techniques in the following sections.

C.6.1. IMPORTANCE SAMPLING

In importance sampling [98], we generate R samples from $Q(\mathrm{x})$. If these points were samples from $P(\mathrm{x})$ then we could estimate Φ by Equation C-1. But when we generate samples from Q, values of x where $Q(\mathrm{x})$ is greater than $P(\mathrm{x})$ will be over-represented in this estimator and where $Q(\mathrm{x})$ is less than $P(\mathrm{x})$ will be under-represented. Thus an "importance" factor $w_r \equiv \frac{P^*(x)}{Q^*(x)}$ is introduced to adjust each point, and $\widehat{\Phi} \equiv \frac{\sum_r w_r \phi(x^{(r)})}{\sum_r w_r}$.

A practical difficulty with importance sampling is that it is hard to estimate how reliable the estimator $\widehat{\Phi}$ is. The variance of $\widehat{\Phi}$ is hard to estimate, because the empirical variances of w_r and $w_r \phi(x^{(r)}$ are not necessarily a good guide to the true variances of the numerator and denominator in $\widehat{\Phi} \equiv \frac{\sum_r w_r \phi(x^{(r)})}{\sum_r w_r}$.

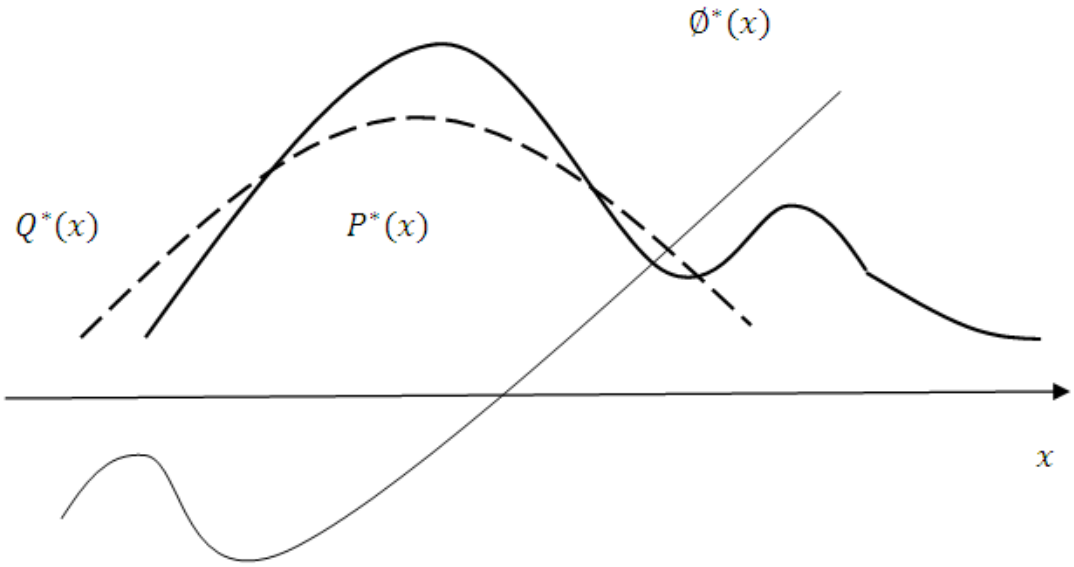


Figure C-6 Importance Sampling

C.6.2. REJECTION SAMPLING

In rejection sampling, we assume that we know the value of constant c such that for all x, $cQ^*(x) > P^*(x)$. A schematic picture of the two functions is shown in Figure C-7 (a). We generate two random numbers. The first, x, is generated from the proposal density $Q(x)$. We then evaluate $CQ^*(x)$ and generate a uniformly distributed random variable u from the interval $[0, cQ^*(x)]$. These two random numbers can be viewed as selecting a point in the two dimensional planes as shown in Figure C-7 (b).

We now evaluate $P^*(x)$ and accept or reject the sample x by comparing the value of u with the value of $P^*(x)$. If $u > P^*(x)$ then x is rejected; otherwise it is accepted.

Rejection sampling will work best if Q is a good approximation to P. If Q is very different from P then c will necessarily have to be large and the frequency of rejection will be large.

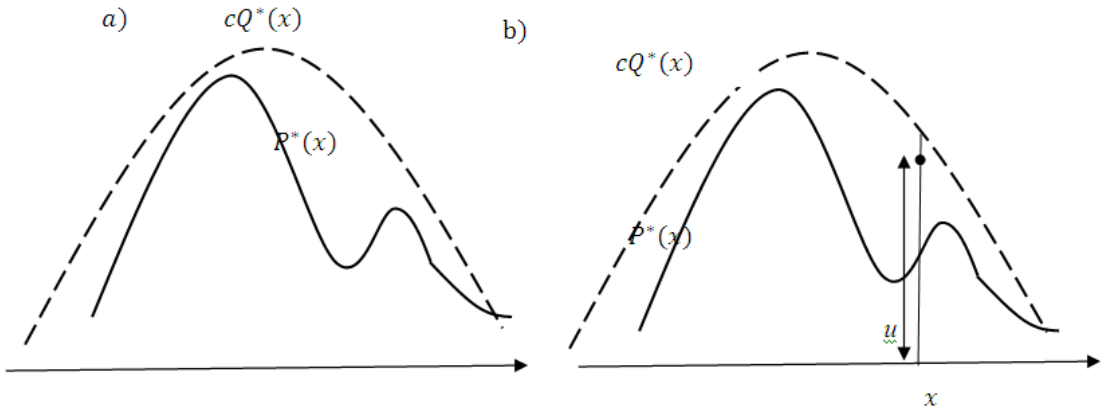


Figure C-7 Rejection Sampling

C.6.3. THE METROPOLIS METHOD

Importance sampling and rejection sampling only work well if the proposal density $Q(x)$ is similar to $P(x)$. In large and complex problems it is difficult to create a single density $Q(x)$ that has this property.

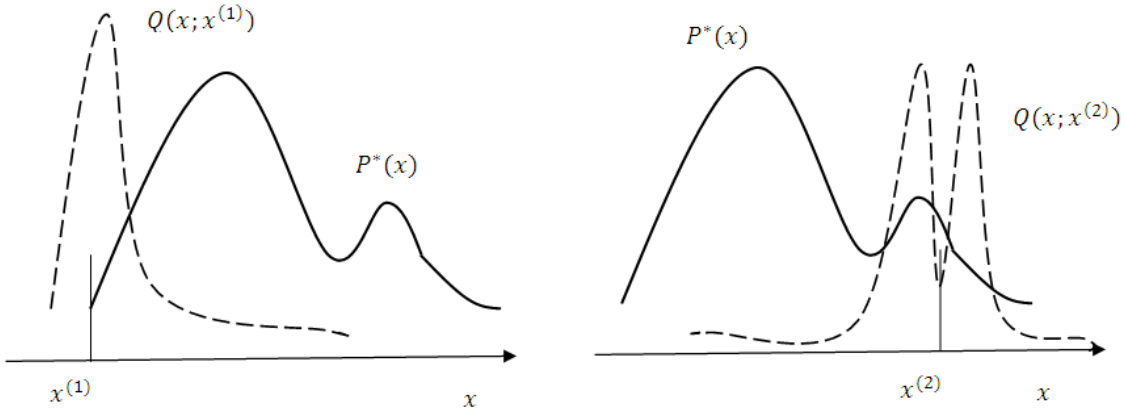


Figure C-8 Metropolis method, Q(x'; x) is here shown as a shape that changes with x

The metropolis method instead makes use of a proposal density Q which depends on the current state $x^{(t)}$. The density $Q(x';x^{(t)})$ might in the simplest case be a simple distribution such as a Gaussian centered on the current $x^{(t)}$. The proposal density $Q(x'; x)$ can be any fixed density. It is not necessary for $Q(x';x^{(t)})$ to look at all similar to $P(x)$. Figure C-8 shows the density $Q(x';x^{(t)})$ for two different states $x^{(1)}$ and $x^{(2)}$. A tentative new state x' is generated from the proposal density $Q(x';x^{(t)})$. To decide whether to accept the new state, we compute the quantity

$$a = \frac{P^*(x')}{P^*(x^{(t)})} \frac{Q(x^{(t)};x')}{Q(x';x^{(t)})}$$

If $a \geq 1$ then the new state is accepted.

Otherwise, the new state is accepted with probability a.

If the step is accepted, we set $x^{(t+1)} = x'$; otherwise then set $x^{(t+1)} = x^{(t)}$. The difference of metropolis sampling to rejection sampling is that rejection causes the current state to be written onto the lists instead of discarded. The metropolis method is an example of a Markov chain Monte Carlo method (MCMC). MCMC methods involve a Markov process in which a sequence of states is generated, each sample $x^{(t)}$ having a probability distribution that depends on the previous state $x^{(t-1)}$.

C.6.4. GIBBS SAMPLING

Gibbs sampling, also known as heat bath method, is a method for sampling from distributions over at least two dimensions. It can be viewed as a Metropolis method in which the proposal density Q is defined in terms of the conditional distributions of the joint distribution P(x). It is assumed that whilst P(x) is too complex to draw samples from directly, its conditional distributions $P(x_i|x_j, j\neq i)$ are tractable to work with.

We illustrate Gibbs sampling using two variables x_1, x_2. On each iteration, we start from the current state x^t, and x_1 is sampled from the conditional density $P(x_1|x_2)$, with x_2 fixed to x_2^t. A sample x_2 is then made from the conditional density $P(x_2|x_1)$, using the new value of x_1. This brings us to the new state $x^{(t+1)}$, and completes the iteration.

BIBLIOGRAPHY

1. Norsys, http://www.norsys.com/.
2. Weka, http://www.cs.waikato.ac.nz/ml/weka/.
3. Bayesian Network in Java, http://bnj.sourceforge.net/.
4. *Singapore Statistics*. July 2006; Available from: http://www.singstat.gov.sg/.
5. Abe, N. Invited talk: Sampling approaches to learning from imbalanced datasets: active learning, cost sensitive learning and beyond. In *ICML03 Workshop*. 2003.
6. Abe, N., B. Zadrozny, and J. Langford, An iterative method for multi-class cost-sensitive learning, In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2004, ACM: Seattle, WA, USA.
7. Akbani, R., S. Kwek, and N. Japkowicz. Applying support vector machines to imbalanced datasets. In *Proceedings of the 15th European Conference on Machine Learning* 2004.
8. Ali, K. and M. Pazzani, HYDRA-MM: learning multiple descriptions to improve classification accuracy. *International Journal of Artificial Intelligence Tools*, 1995: p. 4.
9. Andrews, P., et al., Predicting recovery in patients suffering from traumatic brain injury by using admission variables and physiological data: a comparison between decision tree analysis and logistic regression. *Journal of neurosurgery*, 2003. **97**: p. 326-336.
10. Angus, D.C. and N. Black, Improving care of the critically ill: institutional and health-care system approaches. *Lancet*, 2004. **363**(9417): p. 1314-1320.
11. Antoine, B., et al., Fast Kernel Classifiers with Online and Active Learning. *Journal of Machine Learning Research*, 2005. **6**: p. 1579-1619.
12. Batista, G.E., R.C. Prati, and M.C. Monard, A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explor. Newsl.*, 2004. **6**(1): p. 20-29.
13. Beinlich, I.A., et al. The ALARM monitoring system: a case study with two probabilistic inference techniques for belief networks. In *Second European Conference on Artificial Intelligence in Medicine*. 1989. London, Great Britain: Springer-Verlag, Berlin.
14. Blake, C. and C. Merz, UCI repository of machine learning databases, "http://www.ics.uci.edu/~mlearn/~MLRepository.html". 1998.
15. Boutilier, C., et al. Context-specific independence in Bayesian networks. In *Proceedings of UAI-1996*. 1996.
16. Breiman, L., et al. Classification and regression trees. In *Chapman and Hall/CRC Press*. 1984. Boca Raton, Fl.

17. Buntine, W., Theory refinement on Bayesian networks, In *Proceedings of the seventh conference (1991) on Uncertainty in artificial intelligence*. 1991, Morgan Kaufmann Publishers Inc.: Los Angeles, California, United States.
18. Buntine, W., A guide to the literature on learning probabilistic networks from data. *IEEE Trans. on Knowl. and Data Eng.*, 1996. **8**(2): p. 195-210.
19. Burges, C.J.C., A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 1998. **2**: p. 121-167.
20. Caruana, R. Learning from imbalanced data: rank metrics and extra tasks. In *Proc. Am. Assoc. for Artificial Intelligence (AAAI) Conf.* 2000.
21. Carvalho, D.R. and A.A. Freitas, A Genetic Algorithm-Based Solution for the Problem of Small Disjuncts, In *Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery*. 2000, Springer-Verlag.
22. Chan, P.K. and S.J. Stolfo. Toward scalable learning with non-uniform class and cost distributions: a case study in credit card fraud detection. In *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*. 2001.
23. Chawla, N.V., et al., SMOTE: Synthetic Minority Over-Sampling Technique *Journal of Artificial Intelligence Research*, 2002(16): p. 321-357.
24. Chawla, N.V., et al., Improving care of the critically ill: institutional and health-care system approaches. *Journal of Artificial Intelligence Research*, 2002. **16**: p. 321-357.
25. Chawla, N.V. C4.5 and imbalanced data sets: investigating the effect of sampling method, probabilistic estimate, and decision tree structure. In *Proceedings of the ICML03 Workshop on Class Imbalances*. 2003.
26. Chawla, N.V., et al., SMOTE: Synthetic Minority Over-Sampling Technique. . *Journal of Artificial Intelligence Research*, 2003(16): p. 321-357.
27. Chawla, N.V., et al. SMOTEBoost: Improving prediction of the minority class in boosting. In *Proceedings of Principles of Knowledge Discovery in Databases*. 2003.
28. Chawla, N.V., N. Japkowicz, and A. Kotcz, Editorial: special issue on learning from imbalanced data sets. *SIGKDD Explor. Newsl.*, 2004. **6**(1): p. 1-6.
29. Chen, K. and B.-l. Lu. Efficient classification of multilabel and imbalanced data using min-max modular classifiers. In *the International Joint Conference on Neural Networks* 2006.
30. Choi, S., et al., Prediction tree for severely head-injured patients. *Journal of neurosurgery*, 1991. **75**: p. 251–255.
31. Cooper, G.F. and E. Herskovits. A Bayesian method for constructing Bayesian belief networks from databases. In *Proceedings of the seventh conference (1991) on Uncertainty in artificial intelligence*. 1991. Los Angeles, California, United States: Morgan Kaufmann Publishers Inc.
32. Cooper, G.F. and E. Herskovitz, A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 1992. **9**: p. 309-347.

33. Domingos, P. MetaCost: a general method for making classifiers cost-sensitive. In *Proceedings of the fifth ACM SIGKDD international conference on knowledge discovery and data mining*. 1999. San Diego, California, United States: ACM Press.
34. Dora, C.S., et al. Building decision support systems for treating severe head injuries. In *IEEE International Conference on Systems, Man and Cybernetics*. 2001.
35. Drummond, C. and R.C. Holte. C4.5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on Learning from Imbalanced Data Sets II, International Conference on Machine Learning*. 2003.
36. Druzdzel, M.J. and L.C. Van Der Gaag, Building probabilistic networks: 'Where do the numbers come from?' Guest Editors' Introduction. *IEEE Transactions on Knowledge and Data Engineering*, 2000. **12**(4): p. 481-486.
37. Elkan, C. The foundations of cost-sensitive learning. In *Proceedings of the 17th international joint conference on artificial intelligence*. 2001. Seattle, WA, USA: Morgan Kaufmann Publishers Inc.
38. Engen, V., J. Vincent, and K. Phalp, Enhancing network based intrusion detection for imbalanced data. *Int. J. Know.-Based Intell. Eng. Syst.*, 2008. **12**(5,6): p. 357-367.
39. Engen, V. (2010). *Machine learning for network based intrusion detection: an investigation into discrepancies in findings with the KDD cup '99 data set and multi-objective evolution of neural network classifier ensembles from imbalanced data.* PH.D Thesis. Bournemouth University.
40. Ertekin, S., et al., Learning on the border: active learning in imbalanced data classification, In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. 2007, ACM: Lisbon, Portugal.
41. Ertekin, S., J. Huang, and C.L. Giles, Active learning for class imbalance problem, In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. 2007, ACM: Amsterdam, The Netherlands.
42. Estabrooks, A. (2000). *A combination scheme for learning from imbalanced data sets.* Master Thesis. Dalhousie University
43. Estabrooks, A. and N. Japkowicz. A mixture-of-experts framework for learning from unbalanced data sets. In *Proceedings of the 2001 Intelligent Data Analysis Conference* 2001.
44. Estabrooks, A., T. Jo, and N. Japkowicz, A multiple resampling method for learning from imbalances data sets. *Computational Intelligence*, 2004. **20**(1).
45. Fan, W., et al. AdaCost: misclassification cost-sensitive boosting. In *Proceedings of the Sixteenth International Conference on Machine Learning*. 1999.
46. Fawcett, T. and F. Provost, Adaptive Fraud Detection. *Data Min. Knowl. Discov.*, 1997. **1**(3): p. 291-316.
47. Fawcett, T. (2003). *ROC Graphs: Notes and Practical Considerations for Researchers.* Technical Report. HP Labs.

48. Fawcett, T., An introduction to ROC analysis. *Pattern Recogn. Lett.*, 2006. **27**(8): p. 861-874.
49. Fayyad, U.M. and K.B. Irani, On the handling of continuous-valued attributes in decision tree generation. *Machine Learning*, 1992. **8**: p. 87-102.
50. Foster, P., J. David, and O. Tim, Efficient progressive sampling, In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*. 1999, ACM: San Diego, California, United States.
51. Friedman, J.H., R. Kohavi, and Y. Yun. Lazy decision trees. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*. 1996.
52. Friedman, N. and M. Goldszmidt, Learning Bayesian networks with local structure, In *Learning in graphical models*. 1999, MIT Press. p. 421-459.
53. Geiger, D. and D. Heckerman, Knowledge representation and inference in similarity networks and bayesian multinets. *Artificial Intelligence*, 1996. **82**(1-2): p. 45-74.
54. Gil-Pita, R., et al., Improving neural classifiers for ATR using a kernel method for generating synthetic training sets, In *Neural Networks for Signal Processing, 2002 (IEEE conference proceedings)*. 2002. p. 425 - 434.
55. GlaxoSmithKline. *Asthma Control Test*. 1997; Available from: www.asthmacontrol.com.
56. Graham, I.D., et al., Emergency physicians' attitudes toward and use of clinical decision rules for radiography. *Acad Emerg Med*, 1998. **5**: p. 134–140.
57. Guo, H. and H.L. Viktor, Learning from imbalanced data sets with boosting and data generation: the DataBoost-IM approach. *SIGKDD Explor. Newsl.*, 2004. **6**(1): p. 30-39.
58. Han, H., W.-Y. Wang, and B.-H. Mao. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In *Int'l Conf. Intelligent Computing* 2005.
59. Harmanec, D., et al. Decision analytic approach to severe head injury management. In *Proceedings of the 1999 AMIA Annual Symposium*. 1999.
60. He, H. and X. Shen. A ranked subspace learning method for gene expression data classification. In *Proceedings of the 2007 International Conference on Artificial Intelligence*. 2007.
61. He, H., et al., ADASYN: Adaptive synthetic sampling approach for imbalanced learning. *Neural Networks, IJCNN 2008*, 2008: p. 1322-1328.
62. Heckerman, D., D. Geiger, and D. Chickering, Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 1995. **20**(3): p. 197-243.
63. Heckerman, D.E., *Probabilistic similarity networks*. 1991: MIT Press. 234.
64. Holte, R., L. Acker, and B. Porter. Concept learning and the problem of small disjuncts. In *Proceedings of the 11th international joint conference on Artificial intelligence*. 1989: University of Texas at Austin.

65. Hong, X., S. Chen, and C.J. Harris, A kernel-based two-class classifier for imbalanced data sets. *IEEE Transactions on Neural Networks*, 2007. **18**(1): p. 28-41.
66. Hulse, J.V., T.M. Khoshgoftaar, and A. Napolitano, Experimental perspectives on learning from imbalanced data, In *Proceedings of the 24th international conference on machine learning*. 2007, ACM: Corvalis, Oregon.
67. Japkowicz, N., C. Myers, and M.A. Gluck. A novelty detection approach to classification. In *Fourteenth Joint Conference on Artificial Intelligence*. 1995.
68. Japkowicz, N. The class imbalance problem: significance and strategies. In *Proceedings of the 2000 International Conference on Artificial Intelligence* 2000. Las Vegas, Nevada.
69. Japkowicz, N. and S. Shaju, The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 2002. **6**(5): p. 429-449.
70. Japkowicz, N., Class imbalances: Are we focusing on the right issue?, In *Proceedings of the ICML-2003 Workshop: Learning with Imbalanced Data Sets II*. 2003.
71. Japkowicz., N., Supervised learning with unsupervised output separation. *International Conference on Artificial Intelligence and Soft Computing*, 2002: p. 321-325.
72. Japkowiz, N. Concept learning in the presence of between-class and within-class imbalances. In *Proceedings of the Fourteenth Conference of the Canadian Society for Computational Studies of Intelligence*. 2001: Springer-Verlag.
73. Jo, T. and N. Japkowicz, Class imbalances versus small disjuncts. *SIGKDD Explor. Newsl.*, 2004. **6**(1): p. 40-49.
74. Joshi, M.V., R.C. Agarwal, and V. Kumar. Mining needles in a haystack: classifying rare classes via two-phase rule induction. In *SIGMOD '01 Conference on Management of Data*. 2001.
75. Joshi, M.V., R.C. Agarwal, and V. Kumar. Predicting rare classes: can boosting make any weak learner strong? In *Proceedings of the Eighth International Conference on Knowledge Discovery and Data Mining*. 2002.
76. Joshi, M.V., V. Kumar, and R.C. Agarwal. Evaluating boosting algorithms to classify rare cases: comparison and improvements. In *First IEEE International Conference on Data Mining*. November 2001.
77. Joshi, R. (2009). *Context-sensitive network: a probabilistic context language for adaptive reasoning.* PH.D Thesis. National University of Singapore: Singapore.
78. Kang, P. and S. Cho, EUS SVMs: Ensemble of under-sampled SVMs for data imbalance problems. *Neural Information Processing*, 2006. **4232** p. 837-846.
79. Kohavi, R. Data Mining with MineSet: What Worked, What Did Not, and What Might. In *In Proceeding of the KDD-98 workshop on the Commercial Success of Data Mining*. 1998.
80. Kotsiantis, S., D. Kanellopoulos, and P. Pintelas, Handling imbalanced datasets: a review. *GESTS International Transactions on Computer Science and Engineering*, 2006. **30**(1): p. 25-36.

81. Kubat, M., R. Holte, and S. Matwin. Addressing the curse of imbalanced data sets: one sided sampling. In *Proceedings of the Fourteenth International Conference on Machine Learning*. 1997.
82. Kubat, M., R.C. Holte, and S. Matwin. Learning when negative examples abound. In *Lecture Notes in Artificial Intelligence* 1997: Springer.
83. Kubat, M., R.C. Holte, and S. Matwin, Machine learning for the detection of oil spills in satellite radar images. *Mach. Learn.*, 1998. **30**(2-3): p. 195-215.
84. Kukar, M. and I. Kononenko. Cost-Sensitive learning with neural networks. In *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98)*. 1998: John Wiley & Sons.
85. Le Cessie, S. and J.C. Van Houwelingen, Ridge estimators in logistic regression. *Applied Statistics*, 1992. **41**(1): p. 191-201.
86. Lewis, D.D. and J. Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Proceedings of the eleventh international conference on machine learning* 1994. San Mateo: Morgan Kaufmann.
87. Li, D.-C., C.-W. Liu, and S.C. Hu, A learning method for the class imbalance problem with medical data sets. *Comput. Biol. Med.*, 2010. **40**(5): p. 509-518.
88. Li, G.L. (2009). *Knowledge discovery with Bayesian networks.* PH.D Thesis. National University of Singapore: Singapore.
89. Liao, W. and Q. Ji, Exploiting qualitative domain knowledge for learning Bayesian network parameters with incomplete data. *CPR 2008. 19th International Conference on Pattern Recognition, 2008*, 2008: p. 1-4.
90. Liao, W. and Q. Ji, Learning Bayesian network parameters under incomplete data with domain knowledge. *Pattern Recogn.*, 2009. **42**(11): p. 3046-3056.
91. Liu, A., J. Ghosh, and C. Martin. Generative oversampling for mining imbalanced datasets. In *Proceedings of the International Conference on Data Mining*. 2007.
92. Liu, B., W. Hsu, and Y. Ma, Mining association rules with multiple minimum supports, In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*. 1999, ACM: San Diego, California, United States.
93. Liu, X.-Y., J. Wu, and Z.-H. Zhou, Exploratory under-sampling for class-imbalance learning, In *Proceedings of the Sixth International Conference on Data Mining*. 2006, IEEE Computer Society.
94. Liu, Y.-H. and Y.-T. Chen, Total margin based adaptive fuzzy support vector machines for multiview face recognition. *2005 IEEE International Conference on Systems, Man and Cybernetics* 2005. **2**: p. 1704 - 1711.
95. Liu, Y.-H. and Y.-T. Chen, Face recognition using total margin-based adaptive fuzzy support vector machines. *Neural Networks, IEEE Transactions*, 2007. **18**(1): p. 178 - 192.
96. Lucas, P. Bayesian networks in medicine: A model-based approach to medical decision making. In *Proceedings of the EUNITE workshop on Intelligent Systems in patient Care*. 2001. Vienna.

97. Maloof, M.A. Learning when data sets are imbalanced and when costs are unequal and unknown. In *Proceedings of the ICML-2003 Workshop on Learning from Imbalanced Data Sets II*. 2003.
98. McKay, D.J.C., Introduction to monte carlo methods, In *Learning in Graphical Models*, M.I. Jordan, Editor. 1998, Kluwer Academic Press. p. 175-204.
99. Middleton, B., et al., Probabilistic diagnosis using a reformulation of the internist-1/Qmr knowledge base I. *Medicine*, 1990. **30**: p. 241-255.
100. Miller, M. (1999). *Learning cost-sensitive classification rules for network intrusion detection using ripper.* Technical Report CUCS-035-99. Columbia University.
101. Murphy, P.M. and D.W. Aha, *UCI repository of machine learning databases*. 2004: Irvine, CA University of California, Department of Information and Computer Science.
102. Myung, I.J., Tutorial on maximum likelihood estimation. *J. Math. Psychol.*, 2003. **47**(1): p. 90-100.
103. Ng, T.-P., et al., Factors associated with acute health care use in a national adult asthma management program. *Annals of Allergy, Asthma and Immunology*, 2006. **97**: p. 784-793.
104. Ng, W. and M. Dash, An evaluation of progressive sampling for imbalanced data sets, In *Proceedings of the Sixth IEEE International Conference on Data Mining - Workshops*. 2006, IEEE Computer Society.
105. Ngo, L., P. Haddawy, and J. Helwig. A theoretical framework for context-sensitive temporal probability model construction with application to plan projection. In *Proc. UAI-95*. 1995: Morgan Kaufmann.
106. Ngo, L. and P. Haddawy, Answering queries from context-sensitive probabilistic knowledge bases, In *Selected Papers from the International Workshop on Uncertainty in Databases and Deductive Systems*. 1997, Elsevier Science Publishers B. V.: Ithaca, New York, Switzerland.
107. Nguyen, T.-N., Z. Gantner, and S.-T. Lars, Cost-Sensitive Learning Methods for Imbalanced Data. *International Joint Conference on Neural Networks*, 2010: p. p. 1--8.
108. Niculescu, R.S., T.M. Mitchell, and R.B. Rao, Bayesian network learning with parameter constraints. *J. Mach. Learn. Res.*, 2006. **7**: p. 1357-1383.
109. Nissen, J.J., Glasgow head injury outcome prediction program: an independent assessment. *Neurol Neurosung Psychiary*, 1999. **67**(3): p. 796–799.
110. Olson, D.L. and D. Delen, *Advanced Data Mining Techniques*. 1 ed. 2008: Springer. 138.
111. Pang, B.C., et al., Hybrid outcome prediction model for severe traumatic brain injury. *Journal of Neurotrauma*, 2007. **24**(1): p. 136 -- 146.
112. Pang, B.C., et al., Analysis of clinical criterion for "talk and deteriorate" following minor head injury using different data mining tools. *Submited to Journal of Neurotrauma*, 2007.

113. Papadopoulos, G.A., et al., Analysis of academic results for informatics course improvement using association rule mining, In *Information Systems Development*, Springer US. p. 357-363.
114. Park, S.-b., S. Hwang, and B.-t. Zhang, Mining the risk types of human papillomavirus (hpv) by adacost. *International Conference on Database and expert Systems Applications*, 2003: p. 403--412.
115. Pearl, J., *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. 1988: Morgan Kaufmann Publishers Inc. 552.
116. Pearson, R.K., G.E. Gonye, and J.S. Schwaber. Imbalanced clustering of microarray time-series. In *Proceedings of the ICML-2003 Workshop on Learning from Imbalanced Data Sets II*. 2003.
117. Peng, Y. and J. Yao, AdaOUBoost: adaptive over-sampling and under-sampling to boost the concept learning in large scale imbalanced data sets, In *Proceedings of the international conference on Multimedia information retrieval*, ACM: Philadelphia, Pennsylvania, USA.
118. Platt, J.C., Fast training of support vector machines using sequential minimal optimization, In *Advances in kernel methods*. 1999, MIT Press. p. 185-208.
119. Provost, F. and T. Fawcett, Robust classification systems for imprecise environments, In *Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*. 1998, American Association for Artificial Intelligence: Madison, Wisconsin, United States. p. 706-713.
120. Provost, F. Machine learning from imbalanced data 101. In *Proceedings of the AAAI-2000 Workshop on Imbalanced Data Sets*. 2000.
121. Qinand, A.K. and P.N. Suganthan, Kernel Neural Gas Algorithms with Application to Cluster Analysis, In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 4 - Volume 04*. 2004, IEEE Computer Society.
122. Quinlan, J.R., Introduction of decision tree. *Machine Learning*, 1986. **1**(1): p. 81-106.
123. Quinlan, J.R., *C4.5: Programs for machine learning*. 1993, San Mateo, CA.: Morgan Kaufmann Publishers.
124. Rao, R.B., et al. Clinical and financial outcomes analysis with existing hospital patient records. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2003. Washington, D.C.: ACM Press.
125. Raskutti, B. and A. Kowalczyk, Extreme re-balancing for SVMs: a case study. *SIGKDD Explor. Newsl.*, 2004. **6**(1): p. 60-69.
126. Raudys, S.J. and A.K. Jain, Small sample size effects in statistical pattern recognition: recommendations for practitioners. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1991. **13**(3): p. 252-264.

127. Riddle, P., R. Segal, and O. Etzioni, Representation design and brute-force induction in a Boeing manufacturing design. *Applied Artificial Intelligence*, 1994. **8**(125-147).
128. Signorini, D.F., et al., Predicting survival using simple clinical variables: a case study in traumatic brain injury. *Journal of Neurology, Neurosurgery, and Psychiatry*, 1999. **66**: p. 20–25.
129. Stiell, I.G. and G.A. Wells, The Canadian ct head rule for patients with minor head injury. *Lancet* 2001. **357**: p. 1391–1396.
130. Stiell, I.G. and C.M. Clement, Comparison of the Canadian ct head rule and the new orleans criteria in patients with minor head injury. *JAMA*, 2005. **294**: p. 1511–1518.
131. Su, C.-T. and Y.-H. Hsiao, An evaluation of the robustness of MTS for imbalanced data. *IEEE Trans. on Knowl. and Data Eng.*, 2007. **19**(10): p. 1321-1332.
132. Sun, Y., M.S. Kamel, and Y. Wang. Boosting for learning multiple classes with imbalanced class distribution. In *Proceedings of ICDM'2006*.
133. Sun, Y., et al., Cost-sensitive boosting for classification of imbalanced data. *Pattern Recogn.*, 2007. **40**(12): p. 3358-3378.
134. Swets, J., Measuring the accuracy of diagnostic systems. *Science*, 1988. **240**: p. 1285-1293.
135. Tan, A.C. and D. Gilbert, Multi-class protein fold classification using a new ensemble machine learning approach. *Genome Informatics*, 2003. **14**: p. 206--217.
136. Tang, Y. and Y.Q. Zhang, Granular SVM with repetitive undersampling for highly imbalanced protein homology prediction. *Int'l Conf. Granular Computing*, 2006: p. 457- 460.
137. Teramoto, R., Balanced gradient boosting from imbalanced data for clinical outcome prediction. *Statistical Applications in Genetics and Molecular Biology*, 2009. **8**(1).
138. Ting, K.M. The problem of small disjuncts: its remedy in decision trees. In *Proceeding of the Tenth Canadian Conference on Artificial Intelligence*. 1994.
139. Van Den Bosch, A., et al. When small disjuncts abound, try lazy learning: A case study. In *Proceedings of the Seventh Belgian-Dutch Conference on Machine Learning*. 1997.
140. Van Rijsbergen, C.J., *Information Retrieval*. 2nd ed. 1979, London: Butterworths.
141. Vapnik, V.N., *The nature of statistical learning theory*. 1995: Springer-Verlag New York, Inc. 188.
142. Vilariño, F., et al., Experiments with SVM and stratified sampling with an imbalanced problem: detection of intestinal contractions. *LNCS*, 2005. **3687**: p. 783--791.
143. Webb, G.I., J.R. Boughton, and Z. Wang, Not So Naive Bayes: Aggregating One-Dependence Estimators. *Machine Learning*, 2005. **58**(1): p. 5-24.

144. Weiss, G.M. Learning with rare cases and small disjuncts. In *Proceedings of the Twelfth International Conference on Machine Learning*. 1999. Morgan Kaufmann.
145. Weiss, G.M. Timeweaver: a genetic algorithm for identifying predictive patterns in sequences of events. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 1999. Orlando, Florida.
146. Weiss, G.M. and H. Hirsh, A quantitative study of small disjuncts. *Proceedings of the Seventeenth National Conference on Artificial Intelligence, AAAI Press, 2000*, 2000: p. 665-670.
147. Weiss, G.M. (2003). *The effect of small disjuncts and class distribution on decision tree learning.* PH.D Thesis. Rutgers University.
148. Weiss, G.M. and F. Provost, Learning when training data are costly: the effect of class distribution on tree induction. . *Journal of Artificial Intelligence Research*, 2003(19): p. 315-354.
149. Weiss, G.M., Mining with rarity: a unifying framework. *SIGKDD Explor. Newsl.*, 2004. **6**(1): p. 7-19.
150. Weiss, K.B. and S.D. Sullivan, The economic costs of asthma: a review and conceptual model. *PharmacoEconomics*, 1993(4): p. 14-30.
151. Witten, I.H. and E. Frank, *Data mining: practical machine learning tools and techniques with Java implementations*. 1999, San Francisco: Morgan Kaufmann.
152. Woods, K.S., et al., Comparative evaluation of pattern recognition techniques for detection of microcalcifications in mammography. *International Journal on Pattern Recognition and Artificial Intelligence*, 1993. **7**(6): p. 1417-1436.
153. Wu, G., Class-boundary alignment for imbalanced dataset learning. *ICML-KDD'2003 Workshop: Learning from Imbalanced Data Sets, 2003*, 2003.
154. Wu, G. and E.Y. Chang, Aligning boundary in kernel space for learning imbalanced dataset, In *Proceedings of the Fourth IEEE International Conference on Data Mining*. 2004, IEEE Computer Society.
155. Wu, G. and E.Y. Chang, KBA: Kernel Boundary Alignment considering imbalanced data distribution. *IEEE Trans. on Knowl. and Data Eng.*, 2005. **17**(6): p. 786-795.
156. Yan, R., et al. On predicting rare classes with SVM ensembles in scene classfication. In *IEEE International Conference on Acoustics, Speech and Signal Processing* 2003.
157. Yang, W.-H., D.-Q. Dai, and H. Yan, Feature extraction and uncorrelated discriminant analysis for high-dimensional data. *IEEE Trans. on Knowl. and Data Eng.*, 2008. **20**(5): p. 601-614.
158. Yin, H.L., et al. (2006). *Experimental analysis on severe head injury outcome prediction– a preliminary study.* Technical Report TRD9/06. School of Computing, National University of Singapore.
159. Yin, H.L. and T.-Y. Leong. A model-driven approach to imbalanced data sampling in medical decision making. In *Proceedings of the 2010 World*

Congress on Medical Informatics (MEDINFO 2010). 2010. Cape Town: IOS Press.
160. Yu, T., Incorporating prior domain knowledge into inductive machine learning: its implementation in contemporary capital markets. 2007, University of Technology, Sydney. Faculty of Information Technology.
161. Yuan, J., J. Li, and B. Zhang, Learning concepts from large scale imbalanced data sets using support cluster machines, In *Proceedings of the 14th annual ACM international conference on Multimedia*. 2006, ACM: Santa Barbara, CA, USA.
162. Zheng, J., Cost-sensitive boosting neural networks for software defect prediction. *Expert Syst. Appl.*, 2010. **37**(6): p. 4537-4543.
163. Zhou, Z.-H. and X.-Y. Liu, Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Trans. on Knowl. and Data Eng.*, 2006. **18**(1): p. 63-77.
164. Zhou, Z.-H. and X.-Y. Liu, On multi-class cost-sensitive learning. *Computational Intelligence*, 2010. **26**(3): p. 232–257.
165. Zhu, J. Active learning for word sense disambiguation with methods for addressing the class imbalance problem. In *Proceedings of ACL*. 2007.