Investigating the Process of Developing a KDD Model for the Classification of Cases with Cardiovascular Disease Based on a Canadian Database

by

Chenyu Liu

A thesis

presented to the University of Waterloo

in fulfillment of the

thesis requirement for the degree of

Master of Science

in

Health Studies and Gerontology

Waterloo, Ontario, Canada, 2012

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Medicine and health domains are information intensive fields as data volume has been increasing constantly from them. In order to make full use of the data, the technique of Knowledge Discovery in Databases (KDD) has been developed as a comprehensive pathway to discover valid and unsuspected patterns and trends that are both understandable and useful to data analysts.

The present study aimed to investigate the entire KDD process of developing a classification model for cardiovascular disease (CVD) from a Canadian dataset for the first time. The research data source was Canadian Heart Health Database, which contains 265 easily collected variables and 23,129 instances from ten Canadian provinces. Many practical issues involving in different steps of the integrated process were addressed, and possible solutions were suggested based on the experimental results. Five specific learning schemes representing five distinct KDD approaches were employed, as they were never compared with one another. In addition, two improving approaches including cost-sensitive learning and ensemble learning were also examined. The performance of developed models was measured in many aspects.

The data set was prepared through data cleaning and missing value imputation. Three pairs of experiments demonstrated that the dataset balancing and outlier removal exerted positive influence to the classifier, but the variable normalization was not helpful. Three combinations of subset generation method and evaluation function were tested in variable subset selection phase, and the combination of Best-First search and Correlation-based Feature Selection showed comparable goodness and was

maintained for other benefits. Among the five learning schemes investigated, C4.5 decision tree achieved the best performance on the classification of CVD, followed by Multilayer Feed-forward Network, K-Nearest Neighbor, Logistic Regression, and Naïve Bayes. Cost-sensitive learning exemplified by the MetaCost algorithm failed to outperform the single C4.5 decision tree when varying the cost matrix from 5:1 to 1:7. In contrast, the models developed from ensemble modeling, especially AdaBoost M1 algorithm, outperformed other models.

Although the model with the best performance might be suitable for CVD screening in general Canadian population, it is not ready to use in practice. I propose some criteria to improve the further evaluation of the model. Finally, I describe some of the limitations of the study and propose potential solutions to address such limitations through out the KDD process. Such possibilities should be explored in further research.

Acknowledgements

My deepest gratitude goes first to Professor Jose Arocha, who expertly guided me through my graduate education at the University of Waterloo and who generously shared the plentiful research resources with me. His unique insights to the domain inspired the idea of this thesis, and his inexhaustible patience and endless support led the smooth advent of this thesis.

I would like to thank my thesis committee members, Professor Kumaraswamy Ponnambalam and Professor Helen Chen for their guidance and help throughout this study. Their knowledgeable comments and suggestions led to shape the thesis in a right direction, and their expertise from engineering and health informatics clarified many issues appeared in the process of thesis completion.

My sincere appreciation also extends to all professors and lecturers who taught me various graduate courses, which greatly expanded my knowledge and timely assisted the commencement of the study.

I would also like to thank my dear friends and colleagues, especially Lingkai Zhu for helping me in mathematics and computer science. You made my work and life at the University of Waterloo a happy one.

I would like to express my eternal appreciation towards my parents and family in China, who have always been there for me no matter where I am, for all unconditional supports and encouragements. You are the greatest people in the world.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ACS: Acute Coronary Syndrome

ANN: Artificial Neural Network

ARFF: Attribute-Relation File Format

– An American standard code for information interchange text file that describes a list of instances sharing a set of attributes

AUC: Area Under the ROC Curve

– A measurement to describe the model accuracy visually

BF: Best First search

– A heuristic search method that explores the feature space by expanding the most promising variable subset

C4.5: An algorithm used to generate a decision tree based on information gain

CAD: Coronary Artery Disease

CFS: Correlation-based Feature Selection

– A filter approach for selecting highly relevant but barely redundant variables according to their correlation measurements

CHHD: the Canadian Heart Health Database

CI: Cardiac Ischemia

CSV: Comma Separated Value

– A file format that is often used to exchange data between disparate applications

CVD: Cardiovascular Disease

DT: Decision Tree

ECG: Electrocardiograph

EDA: Exploratory Data Analysis

EM: Expectation Maximization algorithm

– An univariate missing value imputation method

FN: False Negative

FP: False Positive

GA: Genetic Algorithm

– A random search method that mimics the process of natural evolution

GPL: General Public License

– A widely used free software license

HD: Heart Disease

HF: Heart Failure

ID3: An early type of decision tree algorithm, similar to C4.5

IDS: Imbalanced Datasets

IG: Information Gain

– A quantity that measures the usefulness of a variable to the model

J48: A name adopted in WEKA for C4.5 decision tree

KDD: Knowledge Discovery in Databases

KNN: K-Nearest Neighbor

LR: Logistic Regression

MDL: Minimum Description Length

– A principle for model selection assuming that the simplest, most compact representation of data is the best and most probable explanation of the data

MFN: Multilayer Feed-forward Network

MI: Myocardial Infarction

MLE: Maximum Likelihood Estimation

- A method tries to find the parameters for a model with largest likelihood

MLP: Multilayer Perceptron

– A feed-forward artificial neural network

NB: Naïve Bayes

ODESI: Ontario Data Documentation, Extraction Service and Infrastructure

– A digital repository for social science data, including polling data

OR: Odds Ratio

PCA: Principal Component Analysis

ROC: Receiver Operating Characteristics

- A curve indicating the model performance

SMOTE: Synthetic Minority Over-sampling Technique

– A technique for inserting synthetic cases along the line segments joining any/all of the k nearest neighbors of minority class.

SSE: Sum of Squared Errors

– The sum of squared differences between each observation and its prediction

SU: Symmetrical Uncertainty

SVM: Support Vector Machine

TN: True Negative

TP: True Positive

WEKA: Waikato Environment for Knowledge Analysis

– A KDD software developed at University of Waikato in New Zealand

CHAPTER 1
Introduction

Healthcare is one of the most information intensive industries because medical information, knowledge, and data are routinely generated and stored as part of the care process for administrative and research purposes (Bath, 2004). The data volume has been increasing explosively in recent years. A single healthcare episode or research study may yield hundreds of variables producing large amounts of data. The increasing availability of health and medical data is partially attributed to the increased knowledge of disease pathogenesis and the corresponding preventive and therapeutic methods. Healthcare professionals today gather more information on patients from an increasing variety sources, such as laboratory test results, medical images, and numeric and textual data. For example, if an abnormal expression of a certain gene is revealed to be the precursor of a disease, the patient's expression level of this gene is then required to assist diagnosis, generating a new diagnostic variable that was not available in the past. The increasing volume of health and medical data is also partially due to the advances in technology, including the adoption of novel auxiliary medical facilities, the widespread use of massive storage devices, and the related decline in costs.

Nonetheless, individual data items may be of little value in their own right, whereas valuable information contained in large data sets may not be immediately apparent (Bath, 2004). Fortunately, techniques exist that can extract useful information from health data, and analysis tools are available that can reveal underlying patterns in large health and medical data sets, which can be used to

improve the quality of the information accessible to healthcare professionals. Traditional frequency-based statistical methods, such as regression, are a good choice for data analysis most of the time, provided that the analyst possesses a priori notion of the expected relations among variables (Larose, 2004). However, traditional statistical hypothesis testing approaches have drawbacks as they can only be utilized to verify a preformed hypothesis in certain situations. When used this way, traditional statistical tools are employed in a confirmatory mode, where "a model is fitted to the data, and statistical summaries are obtained and tested against the probability that values as high as those obtained could have occurred by chance" (Hartwig & Dearing, 1980, p. 10). In other words, traditional statistical hypothesis testing approaches are less useful if no prior hypothesis exists and cannot be used in an exploratory mode, where to other possible patterns in the data are uncovered.

Exploratory data analysis (EDA) is a more preferable approach when researchers are confronted with large complex databases and no firm hypothesis exist that can direct the investigation, for this exploratory mode of analysis is open to a wide range of alternative explanations (Hartwig & Dearing, 1980). EDA employs a variety of graphical techniques to examine the interrelations among the variables, extract important variables, assess statistical assumptions, and detect outliers and anomalies. By doing so, the analyst could maximize insights into the dataset and suggest hypotheses to test (Larose, 2004).

EDA and the standard statistical hypothesis-testing paradigm provide analysts the ability to understand the datasets they are interested in and translate them into knowledge, yet the scope of data analysis can go beyond that. We may not only want to describe patterns and trends in databases and estimate numerical values of response

variables, but also may wish to predict the prognosis of a disease, classify cancer types, cluster patients with similar symptoms, find which variables are associated with one another and so on and so forth. Moreover, confronted with massive amounts of data increasing on a daily basis, we need the assistance of computers to reduce the workload imposed by large data sets.

The field of knowledge discovery in databases (KDD) has been developed in which traditional statistics and exploratory data analysis are all involved. KDD is an interdisciplinary field bringing together techniques from machine learning, pattern recognition, statistics, databases, and visualization. KDD uses automatic or semiautomatic means to discover valid and unsuspected patterns and trends that are both ultimately understandable and potentially useful to data analysts from a large quantity of data (Fayyad, Piatetsky-shapiro, & Smyth, 1996).

KDD is particularly suitable for the health and medicine domains. The increased availability of health and medical data makes it feasible to seek for new knowledge and to better understand the biological, biochemical, pathological, psychosocial, and environmental processes that mediate health and disease. In addition, medicine and health deal with complex organisms and higher-level processes, so traditional analysis applications conducted by reducing necessary high-level descriptors might be inappropriate and unhelpful. Moreover, many diseases and conditions, particularly noninfectious diseases, may have multiple causative agents or many risk factors, and risk factors themselves are usually interrelated, KDD is flexible to extract those complicated and even nonlinear relationships. Last but not the least, clinicians depend on clinical data and other information such as medical records to make decisions, yet

the large and complex search spaces generated in health/medicine is beyond the ability of clinicians to handle easily (Bath, 2004).

In the present study, several KDD approaches were used in a health scenario to identify individuals with cardiovascular disease (CVD) based on a Canadian data set. Cardiovascular health was selected as the application scenario because it is a major cause of premature death, admission to hospital, and disability in Canada (The Heart and Stroke Foundation, 2011). An estimated 1.29 million Canadians reported suffering from heart disease in 2005 (Lee et al., 2009). About 250,000 potential years of life are lost in Canada each year because of heart disease (The Heart and Stroke Foundation, 2011). Therefore, it is crucial to screen out individuals with cardiovascular disease based on his or her exposure level to a range of predictors in order to implement corresponding prevention campaigns and deliver better healthcare.

From a KDD perspective, identifying individuals with cardiovascular disease can be seen as a classification problem. In the KDD arsenal, many weapons are available to tackle this type of problems. In the proposed study, five distinctive approaches, namely Frequency-based Statistics (e.g., regression), Artificial Neural Networks (ANNs), Belief-based Statistics (e.g. Bayesian classifier), Decision Tress (DT) and Instance-based Learning, were chosen because they are most popular KDD approaches used in classification tasks (Nisbet, Elder, & Miner, 2009).

Traditional frequency-based statistics has become the foundation of data analysis, and its result can be used as the baseline for comparison with the other approaches. Artificial neural networks were inspired by biological neural network and were designed to simulate the information processing in the brain (Larose, 2004). Technically, Bayesian classifier belongs to statistical approach as well because it

attempts to maximize the posterior probability in determining the class (Soni, Ansari, Sharma, & Soni, 2011). Decision Trees generate classification rules that are understandable to the analyst and present them in an upside-down tree shape. Instance-based Learning classifies objects by comparing them with instances already observed in the feature space (Aha, Kibler, & Albert, 1991).

All approaches have been well established, and are widely adopted in many fields of data processing and modeling. They have also been used in the classification of CVD cases in various settings. However, most of them rely on the availability of clinical or medical information that is relatively difficult and expensive to obtain in a large scale (i.e. electrocardiograph). As a result, these models can seldom be applied to screen general population. To our knowledge, none of them have been tested to analyze the Canadian CVD population.

The aim of the present study was to investigate the entire process of developing a classification model for CVD screening based on data from the Canadian Heart Health survey, which contains questionnaire information and simple clinical measurements. The process we explored in this thesis includes several phases, such as variable pretreatment, missing value imputation, variable normalization, dataset balancing, outlier removal, variable subset selection, model construction for single classifier, and adoption of performance enhancement methods. Throughout the investigation, we described the practical issues that appeared in dealing with a heart health dataset and in constructing several classification models.

Five specific learning schemes representing five distinct approaches to KDD were applied in the construction of the models, as well as in developing the testing experiments. The schemes used were the following: (a) *Logistic Regression* (LR)

representing the traditional statistical approach; (b) *Multilayer Feed-forward Network* (MFN) representing the Artificial Neural Network approach; (c) *Naïve Bayes* (NB) representing the Bayesian classifier; (d) the *C4.5* (J48) decision tree method; and (e) the *K-Nearest Neighbor* (KNN) representing Instance-based Learning.

The rest of this thesis is organized in the following way: Chapter 2 reviews the current research that has used KDD methods in building classification models for cardiovascular disease. Chapter 3 gives the study rationale for the studies conducted. Chapter 4 presents the details of the methodology setting up for dataset preparation, including WEKA, the software toolkit used in the experiments; we describe the data set used, the data preprocessing phase, and the process of variable subset selection. Chapter 5 introduces candidate classifiers with detailed description of their properties. Chapter 6 defines the performance measurement adopted in the experiments. Chapter 7 shows the experimental results. Chapter 8 discusses the practical issues involved in the KDD process we conducted, providing some practical insights for researchers in this area. Finally, in the last chapter we present some conclusions regarding the entire study, address the limitations, and suggest recommendations for future work.

CHAPTER 2
Literature Review: KDD Approaches in Cardiovascular Disease

As an "information rich" but "knowledge poor" domain, the healthcare environment remains fresh to KDD approaches (Soni et al., 2011). Due to their popularity and relatively long history, the five proposed KDD learning schemes have been applied to solve problems in cardiovascular health on various settings, such as the diagnosis of cardiac ischemia at the emergency room. Nonetheless, the effectiveness of these methods in the cardiovascular domain have seldom reviewed and compared.

In this chapter, the current research on the application of the five approaches (regression, ANN, naïve Bayes, decision trees, and k-nearest neighbor) to cardiovascular health is reviewed. However, before describing this research, I will define a few basic notions about cardiovascular diseases in the following section.

2.1 Basic Notions about Cardiovascular Diseases

The term "cardiovascular disease" (CVD) refers to a broad category of diseases describing any abnormal condition characterized by dysfunction of the heart and blood vessels, including coronary heart disease, cerebrovascular disease, peripheral arterial disease, rheumatic heart disease, congenital heart disease, deep vein thrombosis and pulmonary embolism (Mendis, Puska, & Norrving, 2011)

Coronary heart disease (CHD), also known as coronary artery disease (CAD), is a narrowing of the small blood vessels that supply blood and oxygen to the heart. CHD is caused by the build-up of atherosclerotic plaques on the walls of the arteries feeding the heart and results in the deprivation of blood and oxygen to the heart (Longo et al., 2011). The pathological condition is defined as cardiac ischemia (CI), of which angina (i.e., chest pain associated with inadequate blood supply to the heart) is the most common manifestation (Longo et al., 2011). When the atherosclerotic plaque blocks blood flow to a part of the heart for a long enough time such that part of the heart muscle is damaged or dies an acute event occurs, called myocardial infarction (MI), or heart attack (HA). Damage to the heart muscle can also lead to heart failure (HF), which takes place when the pumping action of the heart cannot provide enough blood to the rest of the body. The term "acute coronary syndrome" (ACS) is used to refer to the collection of clinical signs and symptoms that includes chest pain or discomfort due to a myocardial infraction (heart attack) or unstable angina.

Cerebrovascular disease refers to a problem with the circulation of blood in the blood vessels of the brain. A blockage with effects lasting less than 24 hours is referred to as a "transient ischemic attack." A complete blockage with long-term effects is referred to as a cerebrovascular thrombosis (clot) or a stroke. Peripheral arterial disease affects the function of blood vessels supplying the arms and legs.

Rheumatic heart disease refers to the damage to the heart muscle and heart valves from rheumatic fever, caused by streptococcal bacteria. Congenital heart disease is a malformation of heart structures existing at birth may be caused by genetic factors or by adverse exposures during gestation. These anatomical defects

can be as simple as a small hole in one of the inside walls of the heart or they can be very complex, affecting the way blood flows through the heart and lungs.

Deep vein thrombosis is the formation of a blood clot in the deep leg veins, which can dislodge and move to the heart and lungs causing pulmonary embolism. Other cardiovascular diseases comprise tumors of the heart; vascular tumors of the brain; disorders of heart muscle, such as cardiomyopathy; heart valve disease; disorders of the lining of the heart; and other afflictions.

Finally, the term "heart disease" is used as a broad disease category, which includes any disorder that affects the heart's ability to function normally, including those mentioned above.

2.2 Studies on the Use of Candidate KDD Approaches for the Classification of CVD Cases

A search was conducted to identify studies that used the five approaches in the classification of cardiovascular disease cases. The search, conducted in Medline and Google Scholar resulted in eighteen studies. The 18 studies were reviewed for the purpose of understanding the recent research progress made of applications of the proposed KDD approaches in various cardiovascular health settings (See Appendix 1). Our search also indicated that most research studies focused on ANNs. Fourteen out of eighteen articles used artificial neural networks. In addition, not all of the approaches have been compared with one another; for example, k-nearest neighbor has not been compared with ANNs or with Logistic Regression. Moreover, none of these approaches have been applied to Canadian data sets for the CVD classification. The research studies are described below.

2.2.1 Studies Using Artificial Neural Networks only

Artificial neural networks have demonstrated their potential for analyzing data sets in several domains (Ashare & Chakraborty, 1994; Larose, 2004), from which some advantages in KDD applications have been identified, such as rapidity and classification precision. The approach has been applied to problems in cardiovascular medicine. In a review by Itchihaporia, Snow, Almassy, and Oetgen (1996), they showed that ANNs have been successfully applied to four areas of cardiovascular disease: coronary artery disease, electrocardiography, cardiac image analysis, and cardiovascular drug dosing (Itchhaporia, Snow, Almassy, & Oetgen, 1996).

In one study by Akay (1992) an ANN was developed to detect CAD from 100 subjects using a supervised approach. The results of the study showed that the network correctly detected 84% CAD cases and 89% of the normal subjects. Among the fifteen prediction parameters used, six of them were obtained from the analysis of the diastolic heart sounds associated with CAD, which were considered as very significant, yet rarely used by other researchers. The remaining nine parameters were taken from physical examination records (Akay, 1992).

However, the scope ANNs in cardiovascular disease goes beyond coronary artery disease and can be used to classify individuals with many types of cardiovascular diseases (CVD), such as myocardial infarction (MI), cardiac ischemia (CI), coronary artery disease (CAD), acute coronary syndrome (ACS), and heart failure (HF), and undifferentiated CVD taken as a whole.

Baxt and Skora (1996) recruited 1,070 emergency patients presenting anterior chest pain to an emergency department. The researchers applied a previously trained

ANN model to diagnose cases of acute myocardial infarction, which they compared to the performance of emergency physicians. Twenty variables were selected and used of the patients' current history, previous history, physical examinations, and ECG results. The ANN model showed en excellent diagnostic accuracy with both sensitivity and specificity of 96.0%. The model surpassed the emergency physicians' diagnostic performance, who only showed a sensitivity and a specificity of 73.3% and 81.1% respectively (Baxt & Skora, 1996).

In another study in emergency medicine, Kennedy et al. (1997) applied an ANN model to a diagnostic task using data from 290 patients with chest pain. The data used included clinical and ECG results available at presentation. Fifty-three binary variables were derived from 39 data items. After model training from 90 patients, the overall accuracy, sensitivity, and specificity for the final network on 200 test data sets were 91.8%, 91.2%, and 90.2% respectively (Kennedy, 1997), showing similar results to those found by Baxt and Skora (1996).

Baldassarre et al. (2004) evaluated the capacity of ANNs to recognize patients with or without a history of vascular events (coronary heart disease or cerebrovascular disease or peripheral vascular disease). A database was generated of 949 patients and 54 variables regarding vascular risk factors and carotid ultrasound characteristics. After the analysis, the results showed that when the ANN model included all 54 variables, it could accurately identify cases with vascular events with 83% accuracy, whereas when using the 37 most relevant variables only, the accuracy of the model increased to 85%, with a sensitivity of 92%. Also, the increased performance with fewer variables underscores the usefulness and necessity of variable selection (Baldassarre et al., 2004) when developing the model.

Harrison and Kennedy (2005) confirmed that ANNs could be used to identify ACS patients from data available at presentation. The authors developed models derived from 1,253 patients, and then tested the models on data collected prospectively from 1,894 patients. At the model training stage, inputs were selected separately from 43 potential variables to generate models using 8, 13, 20, and 40 variables. Increasing the number of predictors beyond 13 variables did not significantly increase the performance of ANN model, so the 13-predictor model, which included ECG data, was used for the model testing due to its marginal improvement over the 8-variable model. After a 10-fold cross-validation, the results showed that the ANN models had, on average, a sensitivity of 91.5%, a specificity of 91%, and an AUC of 0.94 (Harrison & Kennedy, 2005).

Colak, et al. (2008) produced and tested eight different ANN models from 237 patients who had been referred to the cardiology department for the purpose of CAD prediction. Seventeen predictor variables describing demographics, lifestyle and biochemical information were included in the models. Among eight networks used, the best performance was obtained with a model showing an accuracy of 92%, sensitivity of 96%, and specificity of 89% (Colak, Colak, Kocatürk, Sağiroğlu, & Barutçu, 2008).

2.2.2 Studies Comparing Artificial Neural Networks and Logistic Regression

As a basic approach, logistic regression has been used in six studies as a baseline scheme for comparison with ANNs in cardiovascular health. The studies reviewed showed that ANNs demonstrated excellent performance in classifying

individuals with various cardiovascular diseases, easily outperforming logistic regression in all six articles.

Selker, et al. (1995) prospectively collected clinical data from 5,773 patients who were admitted to hospital emergency departments with acute ischemia-like symptoms over a two-year period. Among the 200 relevant variables for the diagnosis of cardiac ischemia, only the 96 variables that could be obtained within the first 10 minutes of emergency care were included. When the models were limited to 8 variables only the AUC of the ANN model was 0.902 while the AUC of the logistic regression model was 0.887. However, when the constrain to limit the model to 8 variables was lifted, LR's AUC increased to 0.905, and ANN's AUC increased to 0.923. Although both models were capable of accurately predicting cardiac ischemia, the ANN model slightly outperformed the LR model (Selker, Griffith, Patil, Long, & D'Agostino, 1995).

Colombet, et al. (2000) evaluated the implementation and performance of a multilayer perceptron (MLP), a form of ANN, as compared with LR to classify the CVD cases. One of the strengths of the study results from the utilization of a huge dataset from the INDANA project (Individual Data Analysis of Antihypertensive Intervention Trials), consisting of 15,444 patients. A set of ten predictive variables describing demographic, physiological, smoking patterns, and electrocardiograph (ECG) information was used for modeling. The outcome was defined as the occurrence of myocardial infarction, stroke, or cardiovascular death. The MLP model correctly classified 76.0% cases in the test set, whereas the LR model only classified 65.9% cases correctly. Although both showed the same AUC, the MLP model had a

slightly narrower 95% confidence interval (0.75-0.81 vs. 0.75-0.80) (Colombet et al., 2000).

Baxt, et al. (2002) trained and tested an ANN and a LR classification model from 2,204 patients with chest pain. The data was restricted to those available at the time of initial patient contact in order to replicate the conditions of real-time evaluation. This resulted in 40 variables regarding patient history, physical examination, ECG, and a set of chemical markers. Although an average of 5% of all network required variables and 41% of chemical marker data were missing, the ANN demonstrated a good performance, with a sensitivity of 88.1% and a specificity of 86.2%, 16% and 8.8% respectively higher than logistic regression. The network also had an AUC of 0.900, 0.069 larger than LR (William G Baxt, Shofer, Sites, & Hollander, 2002a).

In a second study, Baxt, et al. (2002b) used the same population, the same inputs, and study procedure to train and test both ANN and LR model to classify individuals with MI. Consistently, the ANN model outperformed the LR model, correctly identifying 121 of the 128 patients with MI with a specificity of 95.9%. The AUC of the ANN was 0.982, and that for logistic regression was 0.870 (William G Baxt, Shofer, Sites, & Hollander, 2002b). The study by Baxt (2002b) suggests that ANN would be more suitable to classify or predict a specific disease like MI, rather than identifying patients belonging to a broad illness class, such as cardiac ischemia.

Phillips and Street (2005) determined the best prediction of heart failure outcomes, resulting from logistic regression and several other KDD methods including an ANN model. The models were built from 2,500 hospitalized heat failure patients, and more than 12 predictive variables were obtained from insurance claims

to describe patient demographics, diagnoses and treatments. The neural network algorithm yielded an AUC of 0.802, whereas the logistic regression model yielded an AUC of 0.734 (Phillips & Street, 2005).

Green, et al. (2006) compared neural networks and multiple logistic regressions to predict ACS from 634 patients presenting in an emergency department with chest pain. Only 38 variables that were immediately available at patient presentation were used, including ECG data and clinical data. For each approach, the authors produced several models based on the variables used and construction method. When all 38 variables were used, the neural network with the best performance had an AUC of 0.791 while the logistic model had an AUC of 0.757. Nonetheless, when the variables were limited to 16 ECG data only, the network with best performance showed an increased AUC of 0.802, but the AUC of logistic model decreased to only 0.705, indicating the presence nonlinearities in the ECG data that the logistic regression model could not capture (Green et al., 2006).

2.2.3 Studies Using Logistic Regression, Naïve Bayes, and Decision Tree

The performance of Naïve Bayes (NB) and Decision Tree (DT) in detecting cardiovascular disease cases has not been greatly examined, and hence is rarely compared with that of Logistic Regression (LR). Four recent studies, which we review in this section, showed that no method among the ones compared consistently outperformed the other classifiers.

The study by Long (1993) compared the performance of LR to DT, (using ID3, an early type of DT algorithm, similar to C4.5), on the classification of patients with acute cardiac ischemia. The dataset used in the study was collected from six hospitals

containing 5,773 consenting adults (3,453 as training set, and 2,320 as test set). A set of 7 variables for constructing LR was used out of fifty-nine clinical variables available in the emergency room. To generate the DT model all the variables were used except for 7 that were not generalizable or repeating. The LR model outperformed the default ID3 Decision Tree model, with an accuracy of 83.76% and 75.3% respectively on test set, although the later showed a much better performance than LR (accuracy: 92.24% vs. 77.58%) on training set, which suggested the over-fitting nature of default ID3 Decision Tree. The authors further developed a pruned tree which remained less precise than LR, with an accuracy of 80.13% and 83.76%, respectively (Long, Griffith, Selker, & D'Agostino, 1993).

A study by Tsien, Fraser, Long, and Kennedy, (1998) compared a DT model (using the C4.5 algorithm) and a LR model in the diagnosis of myocardial infarction in patients who presented to the emergency room complaining of chest pain. The database, with 1,752 records in total, was collected from hospitals in Scotland and England. Six hundred and thirty records were assigned to the training set, and the rest was reserved as two test sets according to the data origins. Forty-five clinical variables were considered as inputs. The experimental results showed that the DT performed equally well to the LR model on both test sets. The DT reached an AUC of 94.04% and 89.61% separately, whereas LR attained an AUC of 94.28% and 89.28%. The authors compared their tree model, based on C4.5 with the one built by Long et al, (1993), and concluded that their C4.5 tree was more accurate, smaller, and more clinically appropriate (Tsien, Fraser, Long, & Kennedy, 1998) than the one used in Long et al's study.

Anbarasi, Anupriya, and Iyengar (2010) attempted to predict the presence of heart disease using a reduced number of variables. Thirteen demographic and medical variables were originally involved in predicting the heart disease. The researchers utilized a genetic algorithm to determine the variables that contribute more to the diagnosis of heart disease, such that the number of tests needs to be taken by patients was reduced, resulting in the selection of 6 variables. The investigators tested DT (J48), NB, and classification via clustering, a method not included in our research. Observations showed that DT outperformed the other two classifiers after incorporating the variable subset selection, but took a longer time to build the model. The accuracy of three classifiers was 99.2%, 96.5%, and 88.3% for DT, NB, and classification via clustering, respectively. The results also showed that NB performed consistently both, before and after the reduction of variables with the same model construction time (Anbarasi, Anupriya, & Iyengar, 2010).

Ristov, et al. (2010) used machine-learning algorithms to make a better CVD risk management compared to a method defined by health insurance fund in Macedonia. The purpose of the study was to compare the capacity of different learning methods to evaluate and quantify the relationships among cardiovascular risk factors and cardiovascular disease. These learning algorithms consisted of Linear Regression, NB, DT (J48), and so on. The experiments were carried out on a Macedonian dataset containing 1,682 patients among whom 861 were diagnosed with CVD and 11 risk factors for risk assessment. Several results were observed. One of the simplest models, a NB classifier, reached 72.2354% accuracy. The DT (J48) correctly classified 1207 instances with an accuracy of 71.7598%. Unlike many other studies, LR surprisingly but not significantly outperformed all other algorithms with

an accuracy of 73.0083%. Thus, the authors have summarized that LR had the best predictive capacity and easiest interpretability for early CVD detection and prevention (Ristov & Peckov, 2008).

2.2.4 Studies Using Naïve Bayes, Decision Tree, Artificial Neural Network, and K-Nearest Neighbor

The classification performance of these four learning methods has been seldom competed against each other. In the only two articles found, naïve Bayes slightly outperformed other learning schemes, but artificial neural networks had never been compared with k-nearest neighbor.

Palaniappan, S. and Awang, R. (2008) developed a prototype Intelligent Heart Disease Prediction System (IHDPS) using three classification techniques, namely, DT, NB and ANNs. A total of 909 cases with 15 demographical and medical variables were achieved from an U.S. based heart disease database. The database was equally split to training set and test set in a random manner. After complementing three learning schemes, NB appeared to be the most effective classifier as it had the highest accuracy (86.53%) for patients with heart disease, followed by ANNs (86.12%) and DT (85.68%) without much difference. DT, however, appeared to be most effective for predicting patients without heart disease (89%) compared to the other two models. The researchers further concluded that all three models could be used to provide decision support to doctors for diagnosing patients and discovering medical factors associated with heart disease (Palaniappan & Awang, 2008).

Rajkumar, A. et al. (2010) investigated the performance of classifiers including NB, KNN, and Decision List in diagnosing heart diseases. The NB classifier and

KNN were pretty much the same as what are presented in classifier section. Decision list is a question in some formal system with a yes-or-no answer, depending on the values of some input parameters, but it does not attract our much attention because it is not included in our research for its poor popularity. The dataset the researchers used to train and test the classifiers consisted of 3,000 instances with 14 different variables with a split ratio of 2:1. The variables ranged from demographical information to clinical indexes. The experiment results uncovered that the NB outperformed KNN with the accuracy 52.33% versus 45.67%. The NB also consumed less time than KNN to build the model. Therefore the authors have concluded that naïve Bayes algorithm plays a key role in shaping improved accuracy of a heart disease database (Rajkumar & Reena, 2010).

2.3 Summary

The reviewed literature showed the current research progress made in the application of selected KDD tools in the study of cardiovascular disease. First, cardiovascular health remains a fresh domain for KDD as not many experiments have been conducted in this area. Second, more evidence is required to fully understand the usefulness of the different approaches, especially in regards to naïve Bayes, decision trees, and k-nearest neighbor methods. Third, from the present review, we observe that the performance of different learning methods appears to depend on the individual training set used, and no method can consistently outperform the others in all datasets under all conditions. Fourth, the classification accuracy of some KDD approaches have been shown to be comparable or even better than that of physicians using medical and laboratory data, at least in some settings. In this regard, we can

suggest that KDD can help improving the quality of physicians' clinical decisions. Finally, the review also underscores the potential of the variable subset selection as a helpful step in enhancing classification accuracy.

The studies reviewed were conducted using cardiovascular databases mainly from countries such as U.S. or the U.K., yet no study of cardiovascular disease to our knowledge has made use of a Canadian database, although some studies have investigated the KDD methods on Canadian data sets in other medical applications (Eapen, 2004). Therefore, there is a need to test the effectiveness of those methods on Canadian data sets for the classification of CVD. In the present study, we will try to apply the five approaches selected to a nation-wide Canadian dataset.

CHAPTER 3
Rationale

KDD has been proposed as a reliable pathway to discover useful and new information from large amounts of health and medical data, in cooperation with the latest computer technology. To date, many KDD approaches, such as regression, artificial neural networks, Bayesian statistics, decision trees, and k-nearest neighbor methods have been employed to aid the development of medical models. KDD is an integrated process, and a number of basic steps compose the overall KDD process. These steps are the following: dataset selection, preprocessing, transformation, data-mining methods application, and knowledge interpretation and evaluation (Fayyad et al., 1996).

Most of the current studies mainly focus on examining the effectiveness of each learning scheme in the classification of cardiovascular health cases, and somehow they have overlooked other aspects involved in the entire KDD process. In other words, the studies focused mostly on the data-mining step of the KDD method, while many practical issues arise during the procedure of discovering knowledge from cardiovascular health data sets, especially in the steps regarding data extraction from surveys. These issues have failed to attract enough attention from the KDD community. The present study attempts to examine the issues that appear throughout the complete process of developing a classification model for cases in the cardiovascular health domain. Furthermore, we try to provide some reasonable suggestions in relation to the whole process.

The studies reviewed are concerned mainly with clinical or medical variables

that are relatively difficult or expensive to obtain on a large scale (i.e. electrocardiograph or magnetic resonance imaging), thus the models built on these variables are unlikely to be useful when analyzing the general population, although they might be appropriate to detect individual cases in settings such as the emergency room. In addition, these studies did not consider the feasible contributions of people's knowledge or their awareness of cardiovascular health. Such lifestyle and cognitive variables could influence people's behavior, and consequently affect their exposure level to risk factors. The present study takes people's knowledge of cardiovascular disease into account, and involves only variables that can be easily collected through either questionnaires or simple clinical measurements. Therefore, the classification model developed in this study should be particularly useful to, and expandable in, screening CVD cases from a large population.

Another finding from above review is that the performance of many learning schemes in the application of CVD classifications was examined on databases generated from countries like U.S. or other European countries, but none of them were investigated on a Canadian database. In other words, it remains unclear how would these learning methods perform regarding cardiovascular health on databases generated from Canadian population, which has its own uniqueness due to their living environment, life style, genetic diversity, social development status, and so on so forth. Since the performance of different learning algorithms greatly relies on the dataset itself, it is expected to observe how will these proposed methods perform on a Canadian national survey, namely the Canadian Heart Health Database (CHHD), for the first time.

In short, the purpose of the present study is to examine the process of building a KDD model of cardiovascular disease classification with special emphasis on the stages of model development. The initiative comes from the utilization of a Canadian based cardiovascular health database, and the aptness for screening cases with cardiovascular disease.

CHAPTER 4
Methods

4.1 The Waikato Environment for Knowledge Analysis (WEKA) Toolkit

All experiments including developing models for different classifiers presented later on were conducted using the Waikato Environment for Knowledge Analysis (WEKA) toolkit. WEKA is a software developed at university of Waikato in New Zealand for the exploratory analysis of large data sets. As one of the most commonly used open source KDD software, WEKA is written in Java and distributed under the terms of the GNU General Public License (GPL). The software not only includes the most state-of-art machine learning algorithms, but also provides extensive support for the whole process of KDD, including data preprocessing, variable subset selection, evaluating learning schemes statistically, and visualizing the input data and the result of learning. WEKA is also well suited for developing new KDD algorithms. This diverse and comprehensive toolkit is accessed through a common interface so that the users can compare different learning schemes and identify those that are most appropriate for the research problem at hand. In this study, WEKA (version 3.7.5) was used as the only KDD tool.

4.2 Database Description

The Canadian Heart Health Database (CHHD) was first released in 1997. The CHHD is an integration of data from ten provincial heart health surveys that were conducted as part of the Canadian Heart Health Initiative between 1986-1992 across all Canadian provinces. The database was processed and released by Statistics Canada,

one of the most authoritative data source in the country.[1]

The database was chosen because it is relatively large and complex nature. The total numbers of cases included in the database is 23,129. The data set is suitable for testing the proposed approaches, with combinations of different types of variables; thereby provides more profound evidence of the overall performance of each candidate approach. Moreover, the samples recruited in the database well represented the general Canadian population, which replicates a real situation for population screening in which the candidate approaches and corresponding methods will be applied.

The CHHD database includes 265 variables, which can be broadly divided into 16 sections according to the different aspects under investigation. Since the objective of the proposed study is not to estimate the prevalence of CVD risk factors, the knowledge and awareness levels of CVD causes, consequences, or the associated risk factors and lifestyle behaviors at the provincial level, no weights need to be applied to the database. The original format of the datasets is comma-separated value (CSV), which was converted into ARFF format, a machine understandable data format for use in the WEKA toolkit.

4.3 Data Preprocessing

4.3.1 Variable Pretreatment

Much of the raw data contained in the database are incomplete and noisy. As mentioned earlier, the 265 variables in CHHD can be broadly categorized into 16

[1] The database was obtained from Statistics Canada via the Ontario Data Documentation, Extraction Service and Infrastructure (ODESI) under DLI license for free to academic community for research and teaching purposes. The identification numbers is: chhd_E_1986-1992

sections, and yet many of them are highly intercorrelated or even irrelevant to the response variable of CVD. Therefore, the raw data need to be cleaned up in order to minimize the unwanted information that gets into the later models and to shrink the database for better computational performance.

After scrutinizing every single variable included in the database, variables that have no relation to the occurrence of cardiovascular disease (e.g. sequence number) were eliminated. In addition, variables found to be overlapped to another one (e.g. education years to education level) were removed as well. Moreover, a number of variables in the database actually require prerequisites to be activated; they are open only to respondents who give a positive answer to another broader question. For instance, only those who have diabetes were qualified to answer sub-questions like the current treatment for diabetes. In this case, the broader variables were combined with the sub-variables, and were subsequently removed because of the redundancy of the information. Those variables that provide not enough information, such as those containing a very high proportion of missing values, were simply ignored.

The remaining variables were preserved as candidate predictor variables available for the further selection procedure described in the section on variable subset selection. It should be noticed that some retained variables might be recorded in both categorical and continuous form, but both of them were kept because each of them might be a preferred format to a given learning method. For example, in using decision trees, it is typically preferred to use categorical variables in order to execute splitting actions of each decision node. Also, the collinearity (intercorrelation) among the preselected variables may degrade the performance of certain learning algorithms such as the naïve Bayesian classifier, so this detrimental effect was examined and

diminished by using the *correlation-based feature selection method* presented in next section for the purpose of maintaining stability in the solution space and successfully implementing sensitive algorithms.

The preselected predictor variables are either continuous, or nominal although both of them were recorded as numeric. Thus they have to be converted into their corresponding format by certain filters in WEKA before putting into any KDD algorithm. For instance, the variable "age" was converted to a continuous type, but the variable of "age group" was converted to nominal type.

Given that the main task of the candidate KDD methods is to identify individuals with cardiovascular disease, a binary class indicating the heart health status is required to be the response variable. CHHD contains three such variables, indicating whether or not a subject ever had heart attack, stroke, or other heart diseases respectively. However, these three variables only hold in 749, 425, 1945 cases respectively at the outset, so a new combined response variable was created indicating all these diseases (heart attack, stroke, other cardiovascular disease).

4.3.2 Managing Missing Values

The CHHD data set contains many missing values for the input variables. However, the missing-data problem was solved at the outset by imputation. The term "imputation" refers to the substitution of some values for missing data, so it enables the analyst to proceed without further hindrance. To date, a variety of imputation techniques are available to handle missing values. For continuous variables, we do not wish to simply delete the instances with missing value because doing so is a wasteful lost of precious data and may lead to serious biases in the analyses. Replacing missing

values with the corresponding variable mean or group mean is the most commonly used method, but this naive imputation method may create more problems than it solves, distorting estimates, standard errors and hypothesis tests (see Schafer, 1999).

Other methods of dealing with missing values include the "hot deck" method, which consists of replacing the missing value with that of an observed value taken from a matched observation based on the non-missing variables, but the method has been found to be inadequate (Finch, 2010).

One of the most useful and best methods is the "multiple imputation" method. This method is used for continuous missing values, but it has been shown to be computationally heavy (Finch, 2010). The method, although useful, has been found to generate similar results to the Expectation Maximization (EM) method (Finch, 2010). Based on the review of the different methods for handling missing data, I believe that the multiple imputation and the EM methods are the most useful from a practical perspective. However, unlike the EM algorithm, the "multiple imputation" method is not supported by the WEKA platform. Therefore, EM algorithm, described in the following paragraphs, was adopted as the imputation method for continuous missing values.

The EM algorithm is based on Maximum Likelihood Estimation (MLE), which tries to find the parameters for a model with the largest likelihood, such that the appearance of current observations is most probable. The EM algorithm is a technique that finds maximum likelihood estimates in parametric models for incomplete data. It works recursively by repeating the following two steps: (1) *Expectation*, or E-step, in which given the observed data and the parameter estimates, the E-step calculates the conditional expectation of the complete-data log likelihood (the log function is

adopted because it increases monotonically); (2) *Maximization*, or M-step, where given the complete-data log likelihood, the M-step finds the parameter estimates that maximize the complete-data log likelihood from the E-step. The two steps are iterated until the iterations converge. More details about the algorithm description and related equations can be found in a gentle tutorial of the EM algorithm (see Bilmes, 1998).

For nominal variables, there is no consensus about the appropriate imputation methods. The EM algorithm cannot be applied for nominal variables because it assumes a multivariate normal distribution, which is not the case for nominal variables. A common method is to replace the missing nominal values with the variable mode, which is the value that occurs most frequently in a dataset or in a probability distribution. In the CHHD, most missing nominal values in the CHHD data set were not collected in certain provinces at the first place, so replacing missing nominal values with modes denotes that the respondents of entire province have exactly the same behavior to these variables, which does not make sense from a practical point of view. An alternative method is to impute the missing values of a specific nominal variable based on the pattern of this variable in certain provinces, but such information is quite limited as it is unavailable for most variables and provinces, especially for variables describing knowledge and awareness. It is also difficult to assign a specified value to an individual instance even in cases where such patterns exist. As a result, in the present study, decided to delete the instances with missing nominal values. Although this may lead to bias, we made the decision because of the large number of records in the data set.

Other records were also deleted. In particular, subjects whose blood specimen were not taken during the survey in CHHD were removed because they failed to offer

measurements of many crucial biomarkers that contribute to the occurrence of cardiovascular disease. Similarly, individuals who did not attend clinics to take measurements such as height or weight were excluded as well.

4.3.3 Handling of an Imbalanced Data Set

The CHHD database is imbalanced in the sense that the number of negative cases overwhelms the positive cases for the output variable. The phenomenon has nothing to do with the defects in the study design or recording errors, but simply reflects the real situation where the incidence of cardiovascular disease is smaller than the possibility of not having the condition. The imbalance nature of dataset does not affect the majority class too much, but minority classes are often what people care about the most. Unfortunately, imbalanced datasets (IDS) cause a huge deterioration in the performance of normal learning algorithms as, in this situation, they tend to classify non-rare cases more accurately than rare cases. For instance, Japkowicz and Stephen (2002) have demonstrated that decision trees and neural networks are rather sensitive to the negative effect brought by IDS (Japkowicz & Stephen, 2002).

To solve the problem caused by the imbalanced nature of the data set a technique, called Synthetic Minority Over-sampling Technique (SMOTE), has been used (Chawla, Bowyer, Hall, & Kegelmeyer, 2002). As explained in Chawla et al. (2002) SMOTE works by synthesizing rather than simply duplicating new samples by multiplying a random number between zero and one by the distances between samples of minority class and their k-nearest neighbors through all variables. Therefore, the effect is like randomly adding synthetic cases along the line segments joining any/all of the k-nearest neighbors of minority class (Chawla et al., 2002). Specifically, in the

present study the original CHHD dataset was balanced by over-sampling the instances of the minority class (cases without CVD). We conducted two separate experiments to test the effect of balancing on the dataset.

4.3.4 Variable Normalization Methods

The ranges of continuous variables can vary to a large extent. Differences in the variable ranges can result in the tendency for the variables with greater ranges to have excessive influence on the results. To solve this problem, all the continuous variables were normalized in one experiment to standardize the scale of the effect that each variable has on the result. This was done by a method called *min-max normalization* in WEKA (Witten, Frank, & Hall, 2011). Min-Max normalization works by subtracting the minimum value of a variable from each value of the variable and then dividing the difference by the range of the variable. These new values are multiplied by the new range of the variable and finally added to the new minimum value of the variable. These operations transformed the continuous data into a new range, generally [0.1].

Normalization of nominal variables is more problematic, as might be expected. Traditionally, they are converted to indicator (dummy) variables as recommended by Larose (2004). An indicator variable is one that takes the values 0 or 1 to indicate the absence or presence of some categorical effect that may be expected to shift the outcome. In the present study, the conversion was done through transforming a nominal variable to a series of binary variable according to its number of possible values. For instance, a nominal variable such as "marital status" could be converted to five binary variables, where each variable indicates the absence or the presence [0, 1]

of a marital status: never married, divorced, married, widowed, or separated. Note that should only one binary variable have a positive value (e.g. 1) if the data is not missing.

Although normalization has been recommended for balancing the unequal influences due to different magnitudes, in the present study I conducted two control experiments to compare the real effect of normalization on the dataset, as will be described later.

4.3.5 Dealing with Outliers

A common phenomenon in data sets is the presence of outliers; that is, observations that lie outside the overall pattern of a distribution. These abnormal values were detected by means of the Interquartile-Range filter in WEKA (Witten et al., 2011). The outlier was defined as a value larger than the third quartile plus 3 times of interquartile range, and a value smaller than first quartile minus 3 times of interquartile range. The exact effect of outliers on modeling remains unclear. Removing outliers from a data set is definitely beneficial to some algorithms that are sensitive to their presence, such as the k-nearest neighbor method. However, outliers may not represent extreme isolated values, but may also indicate a population with a heavy-tailed distribution (i.e., high kurtosis), so one should exercise a great deal of caution when dealing with outliers.

Probably the best way to deal with outliers is to compare the analyses of the whole data set with the outliers removed. As will be described later, two experiments with and without outliers were conducted to determine whether or not the outliers in the CHHD data set would harm the performance of the methods used.

4.4 Variable Subset Selection

Variable subset selection, also known as feature selection or attribute selection, is a procedure building robust KDD models which consists of selecting a subset of the variable in the data set that are considered the most relevant variables for the model. In theory, including one could think that including more variables should result in a model with more discriminating power. However as Witten et al. (2011) have warned, adding new variables into a model may actually deteriorate the performance of many state-of-the-art learning schemes, regardless of whether these variable are relevant to the model or not (Witten et al., 2011). Moreover, using more variables usually causes a much heavier computational burden and longer training time compared with a simpler and more parsimonious model. Thus, variable subset selection becomes an inevitable stage in KDD.

The aim of variable subset selection is to find the minimal subset of variables that is necessary and sufficient for the target class, and consequently, that reduces the dimensionality of the data with the goal of improving the prediction accuracy of a given classifier. Variable subset selection also speeds processing up, although this benefit may be outweighed by the computational effort involved in implementing the variable subset selection techniques. More importantly, dimensionality reduction yields a more compact, more easily interpretable representation of the research problem, focusing the analyst's attention on the most relevant variables.

In theory, a variable subset is first generated by searching the feature space and then assessed by applying an evaluation function afterwards. The candidate variable subset with best result in the evaluation function is then adopted as the final set of predictor variables for model development. Therefore, determining an appropriate

variable subset generation method becomes a necessary step before transferring the input variables to any downstream evaluation function. Generally speaking, there are three different generation mechanisms for completing this job: complete search, heuristic search, and random search (Dash & Liu, 1997).

As its name suggests, the *complete search* generation procedure conducts a complete search in the subset space. Unlike the other mechanisms, the complete generation procedure goes through entire space with the order of 2^N, where N is the subset size, and does not terminate at a certain point in feature space imposed by some stopping criterion, so it helps to find global optima (Dash & Liu, 1997). Theoretically, conducting an exhaustive search of the entire space would be the most precise method because it tries to assess every single possible combination of all variables according to some evaluation criteria. However, this approach is very costly to carry out in terms of computational complexity, even for a medium subset size (Dash & Liu, 1997), thereby complete searching is impractical for a data set containing a relatively large number of variables. Because of its limitations, we did not use complete search.

Heuristic search is a method that is optimized from exhaustive search and makes every iteration closer to the optimal variable subset. It employs some heuristic functions that minimize the search scope from 2^N to N^2 or less (Dash & Liu, 1997). By sacrificing completeness of search the heuristic method consequently might not always find the best solution globally, but it is guaranteed to find a good solution in reasonable time. Because of the significant increments in efficiency, the heuristic search method has been widely adopted for solving difficult problems that could not be tackled in any other way or that take a practically unreasonable time to compute.

A typical *heuristic search,* used in this study, is *best-first search*. It works in the

following way: a number of variable subsets are ordered according to their performance measures, and the first subset with the best performance measure is given the priority to do a further search by adding or deleting variables (forward selection or backward elimination). The new descendant subsets along with other unchanged subsets are then resorted according to their performance, and the new first best subset now takes the priority for further searching. In other words, the best-first method does not just stop when performance starts to drop but dynamically keeps a list of all variable subsets evaluated thus far, such that it can revisit an earlier variable configuration. Given enough time, this method allows us to explore the full variable space, unless it is prevented by assigning a stop criterion.

Random search is a relatively new approach compared to other two categories. An intuitive analogy of this type of methods is gopher-bashing game; the gopher heads out randomly without anticipation. Similarly, the candidate subset for evaluation is generated by randomly jumping from one spot to another in the search space. Its search space remains 2^N but the random search methods usually find a fewer number of variable subsets than 2^N by setting a maximum number of iterations that are allowed (Dash & Liu, 1997). The strengths of the random search method lies in that it uses fewer computational resources, and that it is of relatively easy implementation. A drawback of the method is that, because it depends on random factors, its experimental results are difficult to reproduce.

A representative *random search* method, employed in the study, is the *genetic algorithm* (GA). As its name suggests, GA is inspired by genetics and attempts to simulate, computationally, the processes by which natural selection operates (Larose, 2005). The "chromosome" in the GA refers to one of the candidate variable subsets,

which is encoded in digit strings, whereas the “gene” refers to a single digit of the candidate subset and represents a variable index. At initialization, a number of chromosomes are specified as the starting population. Each chromosome in the initial population is assesses by the evaluation function and subsequently assigned a probability to be selected as the parenthood chromosome, according to the proportion of its fitness to the total fitness summed over all the chromosomes. After selection, the crossover action occurs, creating two new offspring chromosomes by exchanging the sequences beyond a randomly chosen locus in two parenthood chromosomes (Figure 1). The mutation can also haphazardly take place altering the digits at a particular locus in an offspring chromosome with a very small probability. The three operations described above (i.e. selection, crossover, and mutation) work iteratively until the required amount of population members in one generation is satisfied. After several generations, GA iteratively updates the population until meeting the assigned generation numbers. In this study a population size of 20 and a number of generations of 20 were set as the parameters of GA, and probabilities of crossover and mutation were set to be 0.6 and 0.033 respectively. These are actually the default settings in WEKA.

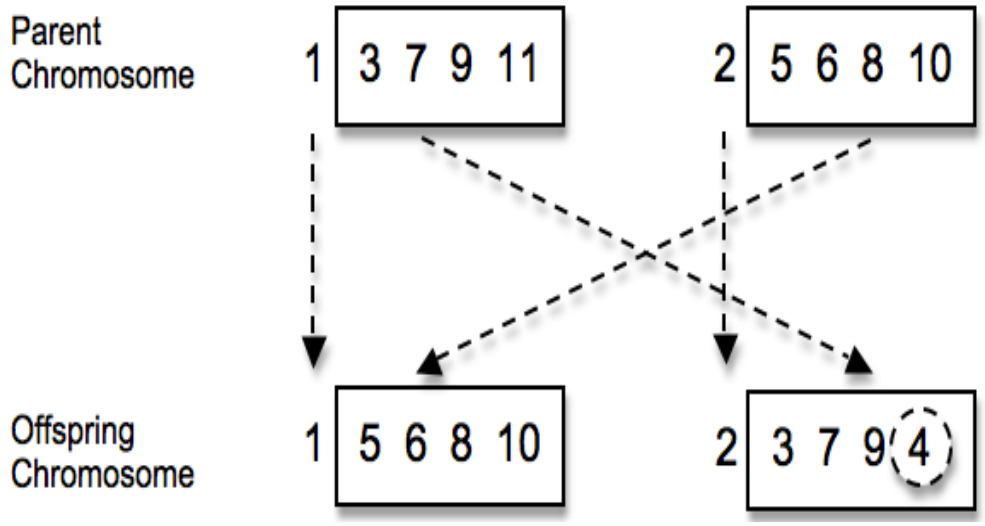

Figure 1: Performing crossover at locus two on the two parents with mutation on an offspring chromosome

As mentioned earlier, each variable subset generated should be passed down to the assigned evaluation function, and the subset with the best available performance is then selected as the final variable subset. The evaluation function is achieved through certain approaches. Two fundamentally different approaches exist for selecting a good variable subset. One is called the *wrapper method*, which ultimately employs a specific learning algorithm to evaluate the subset; the other is called the *filter method*, which makes an independent assessment based on general characteristics of the training data without involving any learning algorithm (Yu & Liu, 2003).

In the wrapper method, a predetermined learning algorithm is wrapped into the selection procedure as part of the evaluation function, and its performance estimated using accuracy estimation techniques is used to assess the fitness of each variable subset to the interested outcome. The variable subset with the highest evaluation is chosen as the final set on which to run the future classification algorithms. The most

commonly employed learning algorithms are linear regression, decision tree, and naïve Bayes, which are will known in the KDD community and represent three completely different approaches to learning (Kohavi & John, 1997; Nisbet et al., 2009). The wrapper approach works as a black box, and no knowledge of the algorithm is needed, just the interface. More details about the underlying mechanisms of these learning schemes can be found in next chapter.

The wrapper approach might be more understandable as long as people know how does the underlying learning scheme behave, but it also has a few drawbacks. First of all, the variable subset selected by a specific learning scheme can be barely generalized to other classifiers, for the wrapper approach tends to find variables better suited to the predetermined learning algorithm resulting in superior learning performance. In addition, the wrapper approach is quite computationally intensive. For the purpose of avoiding over-fitting, the cross-validation technique that is discussed in next chapter in detail is involved in learning procedure, so the entire selection process must be executed for multiple times. Moreover, the selected subset does not provide much confidence with respect to the relevance and redundancy, which are essential in determining classification accuracy. Kohavi and John (1997) indicated that many learning algorithms are rather sensitive to irrelevant or redundant variables such as decision tree and naïve Bayesian classifiers. Nonetheless, a wrapper with naïve Bayes approach was utilized in the study to discover the best available variable subset (Kohavi & John, 1997), after all only would a real experiment yield evidence regarding if the wrapper method is truly proper for domain of cardiovascular health.

The filter approach, on the other hand, is learning scheme independent and just uses necessary variables to partition the instance space in a way that separates all the interested classes. It intuitively makes sense to find the smallest variable subset that serves to distinguish all instances uniquely, so computationally it is less expensive to run and subsequently executes many times faster compared to wrappers, which gives the capability of scaling to datasets with a large number of variables (Hall, 1999). More importantly, filters are much more general than wrappers and do not need to re-run when switching from one learning to another. However, Witten et al. (2011) argued that the filters' bias toward consistency of the variable set may lead to over-fitting, for the algorithm may go to unnecessary lengths to repair an inconsistency, defined as two instances having the same variable values but different class labels, which is caused merely by noise in fact (Witten et al., 2011).

The representative filter employed in this study was the *correlation-based feature selection* (CFS). As Mark A. Hall has claimed, "A good feature subset is one that contains features highly correlated with (predictive of) the class, yet uncorrelated with (not predictive of) each other" (Hall, 1999, p. 4). In other words, a variable is said to be good if it is greatly relevant to the target concept but rarely redundant. Therefore, the CFS filter was determined because of its competency of selecting such a subset. In CFS filter, an information theory based correlation measure is applied to evaluate the goodness of variable subset based on the statement above. The filter first gauges the entropy, a measure of the uncertainty of a random variable. The entropy of a variable X is defined as

$$H(X) = -\sum_{i=1}^{n} P(x_i)\log_2(P(x_i)),$$

and the entropy of X after observing values of another variable Y is defined as

$$H(X \mid Y) = -\sum_{j=1}^{n} P(y_j) \sum_{i=1}^{n} P(x_i \mid y_j) \log_2 (P(x_i \mid y_j)),$$

where $P(x_i)$ is the prior probabilities for all values of X, and $P(x_i|y_i)$ is the posterior probabilities of X given the values of Y. The underlying idea is that if the observed values of X in the training set are divided according to the values of another variable Y, and the entropy of X with respect to the division induced by Y is less than the entropy of X prior to dividing, then there is a correlation between variable X and variable Y. The amount by which the entropy of X diminishes reflects incremental information about X provided by Y and is, hence, called the information gain (IG). IG is a symmetrical measure, so it is defined as

$$IG = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X).$$

According to the measure, if $IG(X|Y)>IG(Z|Y)$, the variable Y is considered more correlated to variable X then to variable Z. Nevertheless, information gain is biased in favor of variables with multi-values. Furthermore, the values have to be normalized to ensure they are comparable and have the same effect. Therefore, symmetrical uncertainty (SU) is adopted as the final correlation measure to compensate for IG's bias and normalize its value to the range [0,1], and it is defined as follows,

$$SU(X,Y) = 2\left[\frac{IG}{H(X) + H(Y)}\right].$$

4.5 Summary

In summary, in this chapter, the WEKA toolkit and the CHHD were introduced.

The former is a platform on which all experiments were conducted; the later is the Canadian Heart Health Database from which the KDD models were developed.

In addition, methods of dataset preparation were described. The input variables have to be pre-treated because not all data are suitable for applying a KDD approach. Data set preparation includes several aspects, such as handling of missing values, dealing with an imbalanced data set and data normalization, and managing outliers.

Missing values for continuous variables were imputed by the *expectation-maximization* method, and missing values for nominal variables were removed along with whole instance. The CHHD is essentially imbalanced, which may cause problem for the classification of a minority class, so the *Synthetic Minority Over-sampling Technique* (SMOTE) was applied to balance the database. In order to control the excessive influence brought by larger scales of some variables, the variable normalization was recommended in the literature as a useful procedure to employ. The existence of outliers is a thorny issue, for they imply rare yet real patterns but ones that generate noise in the data. Outliers could be removed or not depending on their overall impact on the analysis.

After preparing the dataset, it is crucial to execute variable subset selection, which comprises two components: a subset generation component and an evaluation function component. In present study, Best-First search and Genetic Algorithm were employed as subset generation methods, and Naïve Bayes and Correlation-based Feature Selection were utilized as the evaluation function, representing the two approaches to variable subset selection, the *wrapper* and the *filter* approach, respectively.

CHAPTER 5
Candidate KDD Learning Schemes and Model Construction

In this chapter, we present the specific KDD learning schemes representing the five approaches selected for comparison, namely traditional statistics, neural networks, Bayesian statistics, decision trees, and k-nearest neighbors. I also present some combined methods that make use of more than one approach to improve performance.

KDD approaches may be categorized as either supervised or unsupervised. In the supervised approach, the response variable is pre-identified; whereas in the unsupervised approach a search is conducted for patterns and structure among all the variables. Since the response variable in the proposed study has been pre-specified as *cardiovascular disease* and all real classification outcomes are known in advance, all candidate KDD learning algorithms executed are supervised methods.

The overall database is divided into training and test dataset. The training set is used to develop the classification model, and the test set is used to evaluate its performance. The training set includes the pre-classified values of the response variable in addition to the predictors, such that the KDD algorithms could provisionally learn the underlying patterns or relationships from training samples. In contrast, the values of the response variable are hidden in the test set. The provisional model formed from training set can be then carried out on the test set, and the efficacy of the classifications is then evaluated by comparing predicted values against the true values of the response variable.

Usually, the accuracy of the provisional model is not as high on the test sets as it is on the training set, and this phenomenon is referred to as *over-fitting* on the training set. Over-fitting occurs because the provisional model tries to account for every possible trend or pattern in the training set, even for idiosyncratic data points. The increased complexity resulting high accuracy on the training set leads to the degradation in the generalizability of the provisional model to the test set. To avoid the occurrence of over-fitting, that is increasing the classification accuracy on the test set, a technique known as *K-fold Cross-Validation* is employed in present study. At initialization, the technique randomly breaks the overall dataset into k partitions, called folds, and each time a single fold is chosen as a test set, in turns, while the of rest k-1 folds are used for training the model. After *k* iterations, each fold should be tested for exactly once and trained for *k-1* times. As a result, the technique generates k models developed from k slightly different training sets, and then the average accuracy or other performance measurements are reported. Because the fitting mechanism is no longer fixed, the k-fold cross-validation helps to learn the most essential patterns, which are more generalizable than before, to a specific set of questions. Moreover, the reported average evaluation results gives us more confidence about how the model performs in general, so k-fold cross-validation can be used for model selection. The parameter K was set to be 10 in present study because research on numerous different datasets, with different learning schemes, have shown that ten-fold cross-validation leads to the best estimate of error, and there is also some theoretical evidence to back this up (Witten et al., 2011).

5.1 Statistical Approach: Logistic Regression

Logistic regression (LR) is used to approximate the relation between a categorical response variable and a set of predictor variables. To discern categorical response variables the sigmoid function that has non-linear S-shaped curve is utilized (see equation 1 in the appendix 2). From the equation we know that the value for sigmoid function is obtained between 0 and 1, so it is of a form that may be interpreted as a probability of belonging to a class for a given set of predictor variables.

Since the real values of the regression coefficients are unknown, maximum likelihood estimation is utilized, which finds estimates of the regression coefficients for which the likelihood of observing the actual data is maximized. The significance of predictor variables contained in the logistic regression model can be assessed by the likelihood ratio test (G-test) that follows a chi-square distribution with 1 degree of freedom, assuming that the null hypothesis is true that $\beta_i = 0$, or by Wald test (S-test) that follows a standard normal distribution under the same null hypothesis.

The predictor variables could either be categorical or continuous, but in case of polychomotous predictors the dataset should be coded using indicator variable and reference cell coding. Both types of predictors can generate odds ratio statistic according to the fact that $OR = e^{bi}$ $(i = 1,2,...,m)$, which provides a third way of measuring the significance of a predictor simply by checking whether the 95% confidence interval of OR crosses one or not. Consequently, the regression coefficient for a predictor can be derived and interpreted in terms of the natural log of its OR. For categorical predictive variable, the ln OR(x_i) denotes the change in probability of belonging to the class of interest when switching from x_i=0 to x_i=1. For continuous

predictive variable, ln OR(x_i) represents the probability change for an unit increase in the value of the predictor x_i.

The correct interpretation and application of ln OR for continuous variables are established on assumption of linearity, which means that the estimated OR is constant across the range of the predictor. If the assumption is violated the higher-order terms, such as quadratic and cubic variables, or their combinations can be then introduced into the regression model in order to handle nonlinearity. An alternative approach is to transform continuous variables into indicator ones.

5.2 Artificial Neural Networks: Multilayer Feed-forward Network

Multilayer Feed-forward Network (MFN) is one of the ANN algorithms for solving classification, prediction, and estimation tasks developed by simulating the biological neural network. A MFN consists of a layered, feed forward, and completely connected network of nodes. A real neuron uses dendrites to collect biological signals passed from other upstream neurons, reacting to those signals nonlinearly, and sending the new signal to other downstream neurons through the axon, if a response threshold is reached. Similarly, an artificial neural network uses nodes in one layer to collect inputs, processes them nonlinearly, and transmits the outputs to the nodes in next layer. Typically, the MFN is composed of three layers: an input layer, a hidden layer, and an output layer. It should be noticed that the input layer does not process the input and just provides input. In addition, the feed forward feature guarantees the single direction of data flow within the network and does not allow looping. Moreover, every node in a given layer is completely connected to all the nodes in next layer, and

each connection is assigned a weight with a randomly selected value at initialization (Figure 2).

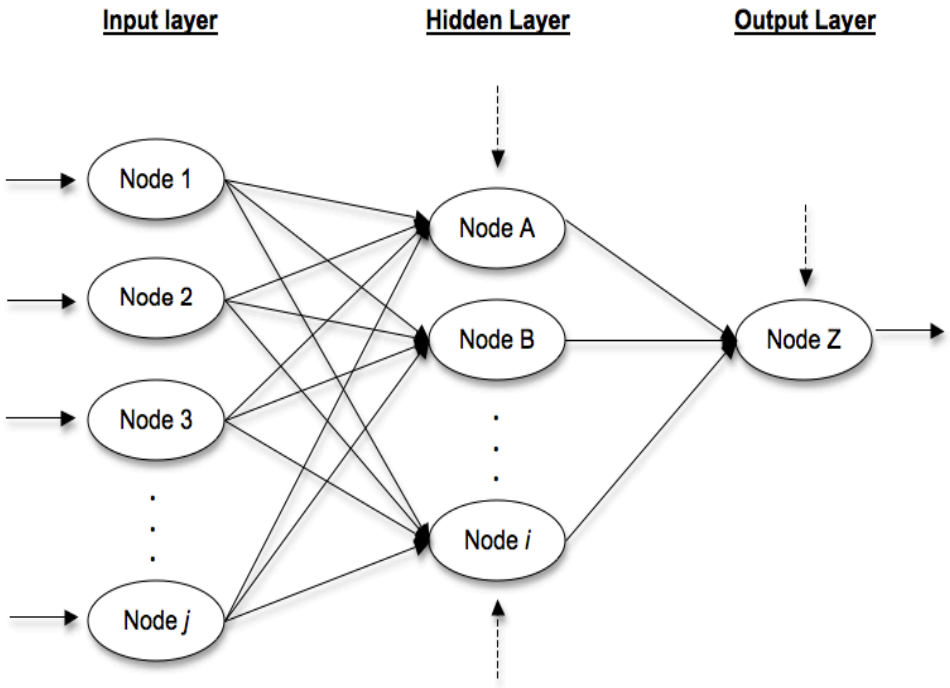

Figure 2: The structure of Multilayer Feed-Forward Network with 3 layers

Here, the inputs mainly refer to the predictor variables that are involved into the model, and all variable values must be encoded in a standardized manner, taking values between zero and one, even for categorical variables. This was done by min-max normalization and other methods described in data preprocessing section. For nodes in the hidden and the output layers, there are constant inputs as well, which are equivalent to the intercepts in logistic regression. Once all the inputs are collected and transmitted to the hidden layer or the output layer, the amount of inputs that each node in these layers received is decided by a combination function called net, which produces linear summation of the node inputs and the connection weights in a single

scalar value (see equation 2 in the appendix 2).

Since the artificial neural network simulates the real neurons that respond to the biological signals nonlinearly, the nodes in the hidden and the output layers use a sigmoid function as a nonlinear activation function, which combines nearly linear behavior, curvilinear behavior, and nearly constant behavior, depending on the value of the input (see equation 3 in the appendix 2). The consequence of applying a sigmoid function is a continuous value ranging from zero to one. The value is then used either as an input to the connected nodes in the output layer or as the final output.

As mentioned above, the weights assigned to each connection are randomly chosen at initialization. As a result, the initial final output or classification obtained according to these weights is undoubtedly rough. To measure how well the output predictions fit the actual target values, the MFN model uses the sum of squared errors (SSE), and the goal of MFN algorithm is to minimize SSE. Therefore, the MFN algorithm relies on the gradient descent method, which proceeds iteratively until the minimum SSE is achieved. The mathematical expression for the gradient descent method is the fourth equation in the appendix 2. The partial derivative of SSE represents the slope of the tangent at the point of current weights on SSE curve, and it should be adjusted to the direction of the slope of local optima that is zero. The learning rate decides how far one-step or one-iteration can move towards the local optima. Large learning rate could lead to overshooting, and small learning rate could result in a slow process. Note that all the weights should be updated simultaneously at one-iteration.

5.3 Naïve Bayes

Naïve Bayes method generally shows an optimal classification performance with less computational efforts, and works well on both categorical and continuous variables (Larose, 2005). It is a really simple classification method, and the underlying principle is assigning a given instance to the class with largest possibility under the presence of information provided by all predictors. Another way of understanding it is to regard the response variable as a random variable coming from a posterior distribution of possible values (either 0 or 1 for this study because of the binary classification essence), while observed data provides information on likely values. The posterior distribution is determined by Bayes' theorem (see equation 5 in the appendix 2). Since the feature subset selected by CFS may have more than one variable, the likelihood functions can be multiplied together under the assumption that the variables are conditionally independent, which is the case for this study. The value of response variable that maximizes the posterior probability, or the class label with largest possibility is then assigned as the predicted class.

5.4 Decision Tree: C4.5

The decision tree is one of the most popular classification algorithms in current use in KDD development. The attractiveness of decision trees lies in their interpretability, especially with respect to the construction of decision rules. Shaped as an upside-down tree, decision tree is a collection of decision nodes, connected by branches, extending downward from the root node until terminating in leaf nodes. Once a new instance enters the decision tree diagram, it is tested at the decision nodes in each level with respect to target variables, and passed downward to other decision

nodes in next level along branches until reaching a terminating leaf node, which indicates the class assignment. The effect of decision tree classification is to partition the dataset in accordance with the values of predictive variables. The decision tree is compatible to both continuous and categorical variables, but the continuous variable must take on values that are explicitly separated as either belonging to a particular class or not belonging. Figure 3 provides an example of a simple decision tree.

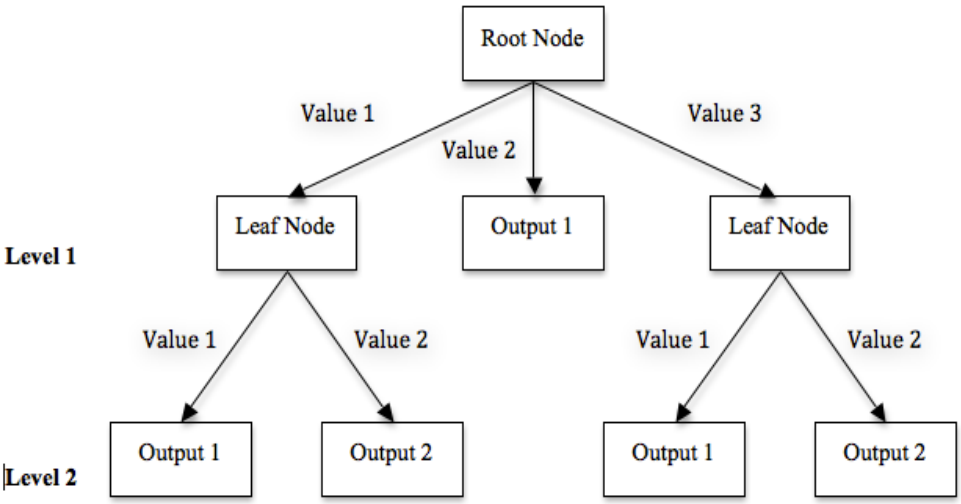

Figure 3: The structure of a simple Decision Tree

Decision tree seeks to create a set of leaf nodes that are as "pure" as possible, that is, each of the instances in a particular leaf node has the same classification. In this way, the decision tree needs to provide classification assignments with the highest measure of confidence available. Based on different methods for measuring leaf node purity, many algorithms for growing decision trees have been proposed, and the study employed a widely applied leading algorithm named C4.5 decision tree algorithm. The C4.5 algorithm recursively partitions the instances in the training dataset into subsets of instances with similar values for the target variable. The C4.5 algorithm

grows the tree by conducting for each decision node, an exhaustive search of all available variables and all possible splitting values, selecting the optimal split according to the information gain ratio (see equation 6 in the appendix 2). In short, information gain ratio measures the increase in information produced by partitioning the training dataset according to a candidate split, while biasing the decision tree against considering variables with a large number of distinct values. At each decision node, C4.5 algorithm chooses the optimal split whichever maximizes the information gain ratio over all possible splits. Unlike other decision tree construction algorithms, C4.5 algorithm is not restricted to binary splits, but for categorical variables it by default grows a separate branch for each value of these variables.

5.5 Instance-based Learning: K-Nearest Neighbor

The last basic classifier we shall investigate is the K-Nearest Neighbor (KNN) algorithm, which is most often used for classification, although it can also be used for estimation and prediction. KNN is a representative of instance-based learning. In it, a classification for a new unclassified instance is found simply by comparing it to the most k similar instances in the training set (Aha et al., 1991). More technically, if most of k nearest neighbors of a given instance in the feature space belongs to one particular class, then the given instance may probably belong to this class as well (Figure 4).

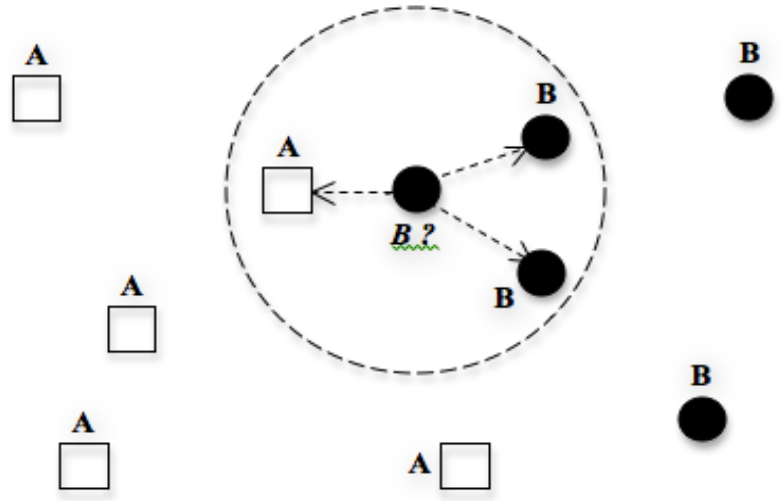

Figure 4: The structure of K-Nearest Neighbor

The selection of nearest neighbors depends on their nearness or similarity to the new unclassified instance. The nearness or similarity is measured by distance metrics that are real-valued functions. The distance metrics have following properties: non-negativity, commutativity, and triangle inequality (Larose, 2004). The most common distance metric is Euclidean distance, which stands for the usual way in which people conceives the distance in real world, so it was consequently adopted in constructing the KNN model in this study. It is worth to note that all selected neighbors must be pre-classified, or they must hold unambiguous class labels in advance. In addition, although continuous variables with large values may overwhelm the influence of other variables that are measured on a smaller scale, the normalization to continuous variables in dataset preparation stage could avoid this detrimental effect. Moreover, the Euclidean distance metric is inappropriate for categorical variables, so another function named as “different from” is then substituted instead, with value 0 indicating no difference and a value of 1 showing otherwise (Larose, 2004). The effect of

applying distance metric is to identify k instances that are of minimum distance from the new unclassified instance in feature space.

The problem remains for choosing an appropriate value of k, which can be solved by various heuristic techniques (Rajkumar & Reena, 2010). Generally speaking, larger values of k limit the influence of noise on the classification behavior; yet make decision boundaries less distinct. Thus, the best value of k in this study was selected iteratively by heuristic cross-validation technique, but an odd number was preferred for k because it prevented tied votes in binary classification.

With a method of determining which instances are most similar or close to the new unclassified instance in terms of the Euclidean distance, a combination function that incorporates all neighbors and provides a final class assignment for the new unclassified instance is required. A simpler method is to vote for the classification decision by k nearest neighbors equally or unweightedly. A more sophisticated method takes the weights of each nearest neighbor into account, that is their distances from the new unclassified instance matters. Closer neighbors have a heavier vote in the final class assignment than do more distant neighbors. The implementation of this weighted voting is that the influence of a given instance is inversely proportional to the distance of the instance from the new instance to be classified.

5.6 Cost-sensitive Learning: MetaCost Algorithm

As mentioned in the data-preprocessing phase, an imbalanced data set (IDS) has been shown to be a detrimental phenomenon appearing in the CHHD database for the candidate classifiers. Synthetic Minority Over-sampling Technique (SMOTE) was used to increase the sample size for the minority class from a dataset perspective. In

addition to over-sampling approach, the other approach for solving problems associating with IDS focuses at the algorithm level, which aims to improve original algorithms or to design new learning schemes with better performance. Researchers have summarized that there are currently four major methods: cost-sensitive learning, support vector machine, one-class learning, and ensemble learning (Lin, Hao, & Yang, 2008; Ye, Wen, & Lv, 2009). In the study, cost-sensitive learning was chosen as the main approach at the algorithm level, for it considers the non-uniform costs of misclassification that corresponds to the necessity of real situation in cardiovascular health, where the false positive is far less costly than false negative. Ensemble learning method was also applied in this study as to improve the overall performance for basic learning schemes, and more details about it are presented in following section.

Cost-sensitive learning method in this study is achieved through MetaCost algorithm that wraps a cost-minimizing procedure around an arbitrary classifier (Domingos, 1999). Rather than making any change to the classifier, the MetaCost algorithm takes the misclassification cost into account by giving a cost matrix *C*, where *C(i,j)* represents the cost that an instance actually belonging to class *j* is misclassified to class *i*. The algorithm predicts a given instance *x* to the class that minimizes the overall cost, which is calculated through conditional risk (see equation 7 in appendix 2). $P(j|x)$ can be obtained according to proportion of each class to the total votes through bagging technique that is further discussed in ensemble learning section. In present experiment, ten new training sets were allowed to be produced by bagging procedure, and the number of cases in each new training set was set to be equivalent to the original training set size, which allowed it to be more general. For

the successful application of MetaCost algorithm, the core is to create a proper cost matrix $C(i|j)$ that reflects the real situation. Unfortunately, the current research regarding misclassification cost of cardiovascular disease in reality does not bear many fruits. Sampurno assumed that the false negative cases cost four times bigger than false positive cases in her thesis regarding cardiovascular disease in patients with type 1 diabetes (Sampurno, 2006). The scale was consequently applied in this research as the default cost matrix, but the effects of various scales from 4:1 to 1:7 were investigated as well. The candidate learning scheme with the best performance examined above was employed as the base classifier for MetaCost sensitive learning algorithm.

5.7 Ensemble Modeling

Ensemble modeling, also known as ensemble learning or bundling, refers to a machine-learning frame in which multiple learning schemes trained individually are combined to solve the same problem. In other words, ensemble models try to combine a set of hypotheses whereas ordinary learning algorithms try to learn only one hypothesis from the training data (Zhou, 2009). The underlying idea of ensemble modeling is quite simple in that the model constructed by the combination of a set of base classifiers trained from different datasets often outperforms the models build by the best single classifier in terms of accuracy; something that has been supported by a number of studies (Zhou, 2009).

Literally, any learning scheme can be used as the base classifier including the classification algorithms adopted in present study, as long as it is more accurate than random guessing (accuracy>0.5), but only the classifier with the best performance in

this study was used in the ensemble learning in order to build a model as accurate as possible. Although increasing the number of base classier to infinite would technically lead to zero error rate for the final ensemble model (Freund & Schapire, 1996), only ten base classifiers were combined in the research due to the practical consideration. To date, several ensemble-modeling techniques have been developed. In present study two of them were employed, namely Bagging and Boosting.

Bagging is a technique that trains a number of base classifiers each from a different bootstrap sample. A bootstrap sample is generated by repeatedly duplicating cases from the original dataset with the possibility of copying for more than once until meets the size of the original. Thus, the instances from original dataset may or may not appear in the bootstrap sample. After obtaining the base classifiers, the bagging technique combines them by majority voting and the class with most voting is then assigned for a new instance (Nisbet et al., 2009; Zhou, 2009). In present study, the iterations of generating bootstrap samples are set to be ten.

Boosting is a technique that combines a number of base classifiers with variety. The variety is created from weighting cases based on which ones are easier or harder to model correctly. The specific AdaBoost M1 method was applied for boosting because it is more promising for binary classification problem (Freund & Schapire, 1996). In AdaBoost M1, the different training sets are generated through adjusting the weights of corresponding instances. At initial, the weights for each instance are equal, and a base classifier 1 is trained under current sample distribution. For those instances misclassified by weak classifier 1, their weights are increased; for those correctly classified instances, their weights are decreased. In this way, the misclassified instances become more salient and hence get a new sample distribution (training set).

At meanwhile, the base classifier 1 is assigned a weight as well according to its classification performance in order to indicate the importance of it, the more accurate the more weights. Then the weak classifier 2 is trained in new sample distribution, and the learning process keeps running for *n* iterations, which results in *n* base classifiers with *n* corresponding weights. The number of ten iterations was designated in present study. The final step is to combine those weak classifiers based on their weights, and to produce the final anticipated strong classifier (Freund & Schapire, 1996).

From previously described details of AdaBoost algorithm, a few characteristics of the technique should be noticed: (1) each iteration only alters the sample distribution rather than re-sampling as bagging does; (2) the alteration of sample distribution merely depends on classification status; and (3) the final model is generated from the weighted combination of weak classifiers. In summary, AdaBoost is an outstanding technique in terms of its efficiency and simplicity, because it significantly improves classification accuracy as reported by others (Freund & Schapire, 1996) and is compatible to any basic learning schemes (Freund, Schapire, & Abe, 1999).

5.8 Summary

In this chapter, five basic yet popular supervised KDD learning schemes and two approaches for improving classification performance were presented in detail. The logistic regression is a function that fits most observed data points in the feature space and measures the relationship between the dependent variable and a set of independent variables. The multilayer feed-forward network is a type of artificial

neural network model that maps a set of observed instances onto a set of appropriate outputs through a biologically inspired network. The naïve Bayes algorithm is a simple probabilistic classifier based on Bayes' theorem with its assumption of conditional independence of each predictor variable. The decision tree maps observations about an instance to conclusions about its target class, and is developed by splitting the dataset into subsets according to values of variables. The k-nearest neighbor is an algorithm for classifying new instances based on closest similar training data points in the feature space.

To cope with the fact that false negative classifications are more costly than false positive classifications, cost-sensitive learning was implemented through MetaCost algorithm that alters the underlying cost matrix. The study also intended to apply ensemble-modeling technique because it usually enhances the performance of single classifier. Two approaches, namely bagging method and AdaBoost M1 algorithm, were employed to develop the ensemble model. The former generates a number of slightly different training sets, and the later adjusts the weight of each instance and base classifier. All experiments were conducted with 10-fold cross-validation technique in order to acquire a more general realization.

CHAPTER 6
Classifier Performance Measurements

The performance measurements adopted in the proposed study will be accuracy, sensitivity, specificity, precision, kappa statistic, and area under the curve (AUC), all commonly used in the KDD literature. In addition, the computational time of each method will also be reported.

Since the response variable is binary, an instance would be classified as only one of four possible outcomes, given a classifier:

If it is a positive instance, it will be classified as positive and counted as true positive (TP); if it is a positive instance and it is identified as negative, it will be counted as a false negative (FN); likewise, an instance could also be classified as a true negative (TN); or as a false positive (FP). Given a classifier and a set of instances, these four possible outcomes would consist of a two-by-two table called confusion matrix that is reported by all classifiers. The simplest measurement is the accuracy that just divides the number of correctly classified instances by the number of total instances.

More sophisticated measurements are sensitivity, specificity and precision. Sensitivity, also called hit rate, or recall rate in some fields, measures the ability to correctly identify positive instances from all positive instances. Therefore, the sensitivity is calculated as the number of true positives (TP) over the total number of true positives and false negatives (TP+FN). Similarly, specificity measures the ability to correctly identify negative instances from all negative instances, and it is expressed as the number of true negatives (TN) over the total number of true negatives and false

positives (TN+FP). The two measurements can be derived from the confusion matrix directly. On the other hand, precision gauges the fraction of all classified instances that are correct or hold true under one class label, so it is calculated as TP/TP+FP or TN/TN+FN. The larger the number, the better the performance, such that good results correspond to large numbers down the main diagonal and small, ideally zero, off-diagonal elements. In present study, average precision for both classes is reported.

The overall accuracy is an easy but insufficient measurement for evaluating the real capacity of classifiers, for it is unclear the extent to which the result could be equally achieved by a random classifier. The Kappa statistic is often used to measure the agreement between classified and actual categorizations of a dataset, while correcting for an agreement that occurs by chance (Witten et al., 2011). It is expressed as the difference between the instances correctly classified by target model (TP+TN) and instances could be correctly classified by chance, over the difference between all instances (TP+FP+TN+FN) and instances could be correctly classified by chance.

The area under the curve (AUC) is another commonly utilized performance measure. The curve here refers to the receiver operating characteristics (ROC) graph. It is useful for organizing classifiers intuitively and visually and particularly suitable for skewed class distributions (Fawcett, 2006). ROC graphs involves the third parameter, false positive rate or false alarm rate of a classifier, which refers to the number of negatives included in the sample, expressed as a percentage of the total number of negatives (FP/(FP+TN)). With this information, a ROC graph plots the sensitivity on the vertical axis against the false positive rate on the horizontal axis. Once a new case enters the model, it alters the class distribution in confusion matrix, so a (sensitivity, false positive rate) pair corresponding to a single point in ROC curve

is produced. The process continuous until all cases enters the model. As a result, the ROC curve depicts the tradeoff between benefits (true positives) and costs (false positives) (Fawcett, 2006). We naturally expect to gain more benefits with less costs, so a good classifier should generate a curve bulging to the upper left corner, and the farther the better. Consequently, the larger the area under the curve (AUC) the better the model classifies.

In short, six performance measurements indicating different aspects were adopted in present study. To summarize, they are accuracy, sensitivity, specificity, average precision, kappa statistic, and area under the curve (AUC). In addition, the computational time of each method will also be reported.

CHAPTER 7
Experimental Results

7.1 Data Preprocessing

7.1.1 Variable Pretreatment

First of all, a new response variable was created. In the original CHHD database, three distinct variables exist that indicate cardiovascular disease, namely *heart attack*, *stroke*, and *other heart disease*. Specifically, these three variables were EVERSTR, EVERHA, OTHHD (See Appendix 3 for the list of variables). For the present study, we considered cardiovascular disease as a single entity. Consequently, a new response variable labeled as CVD was then produced through combining and replacing three existed variables.

Several variables in the original CHHD database were found to be irrelevant for the classification of CVD cases, and were therefore deleted directly from the database. Seven variables fell directly into this category, namely SEQNO (Sequence number), INBLOOD (Blood taken), INCLINIC (Clinic attended), PWGTQ (Probability weight for questionnaire), PWGTC (Probability weight for clinic/blood specimen), HRS (Hours since last meal), FAST (Fasting), for they did not have indication to the occurrence of cardiovascular disease at all. In addition, variable PROV (Province) had to be deleted as well, for some provinces such as Nova Scotia no longer had records left in the database after removing instances with missing values, which made the variable worthless. Moreover, one variable believed to be unhelpful for model building, RXHEART (Treatment for heart disease), was deleted, for people who were

taking any prescribed medication for heart definitely suffer from cardiovascular disease.

More often, a few broader variables were actually summarized or accounted for a number of other variables with overlapping meaning, so that the overlapping variables were eliminated in this step. More specifically, several more refined variables were calculated or derived from thirty-five more detailed variables in the database. For example, the variable EDUC (Highest education level completed) was generated from variable EDUCYRS (Education years) according to the Canadian education system, so the variable EDUCYRS was deleted. Thirty-five variables of this type were ignored in the process of developing our model.

From time to time, the identical information was recorded in different scales in CHHD database, so variables indicating the same content as other variables but in a broader range were eliminated. For instance, two categorical variables grouping subjects' age were recorded, but only the one that grouped age in ten years was retained, namely GPAGE2. After scrutinizing all variables in the database, eleven variables were moved out from the CHHD database due to this reason.

In some cases, certain variables contain too scarce responses to provide adequate information for developing our classification model, thus these variables were removed in this step. In general, these variables required prerequisites only open to certain respondents. These variables involved branch questions describing specific aspects of variables at a higher level, which inevitably led to the appearance of large amounts of missing values in them. For example, the variable BPRX1 tells if respondents take medication for high blood pressure, but the variable only opens to

people who suffer from high blood pressure and who actually take corresponding treatments. This character held forty-two variables in original CHHD database.

Occasionally, several variables blended in with their direct branch questions in lower level. In this way, these variables in upper level were taken as specific values for branch variables in lower level and removed subsequently. For instance, the variable DBNOTRT (No current diabetic treatment) was only open to subjects who reported having diabetes according to variable DIABCAT (Diabetes status), so the variable DIABCAT was then combined with the variable DBNOTRT, such that the value of 0 that initially represented "not applicable" in DBNOTRT now meant that the present subject had no diabetes. As a result, fourteen variables like DIABCAT were removed from the CHHD database after blended in with their direct branch variables.

Finally, a few variables were seldom recorded in text format in the first place, hence neglected from model construction process. Specifically, these variables were BPHLTHOW (How high BP affects health), BPCAUSE (Cause of high BP), and CHOLINFO (Information about cholesterol).

In summary, 114 variables were removed from the CHHD in advanced in the database due to various reasons exposed above. Also, 151 predictor variables and 1 response variable were retained or created in CHHD database for further variable subset selection. The more details about each variable can be found in Appendix 3.

7.1.2 Dealing with Missing Values

Missing data is present in the CHHD data set. Some missing values are present because individual respondents did not provide certain information. Some survey

participants failed to provide their blood sample for biomarker testing such as cholesterol; or did not attend clinics for checking their blood pressure and gauging personal measurements. Although these data are certainly relevant for determining cardiovascular disease status, they were removed from the CHHD database according to their values on variables INBLOOD (Blood taken) and INCLINIC (Clinic attended).

Most missing values in nominal variables occurred because some information was not collected in all provinces. Replacing this kind missing values with modes is inappropriate, for the underlying patterns of those nominal variables would be distorted across the population. Therefore, the instances for nominal variables with missing values that were not collected by all provinces were excluded from the CHHD database.

These two manipulations resulted in totally 4,967 instances left in the CHHD database, including 4,405 negative cases and 562 positive cases respectively. The remaining missing values were only for continuous variables and then imputed by means of the *expectation-maximization algorithm* described earlier based in the section on multivariate normal distribution.

7.1.3 Effect of Balancing the Dataset

As explained earlier in chapter four, balancing the CHHD database is expected to overcome the unpromising classification performance for the minority class brought by the imbalanced nature of the dataset. Two experiments were hereby conducted to examine the real effect of balancing the CHHD dataset through over-sampling technique on overall performance of classifiers.

In order to compare experimental results on the same basis, all experiment settings were kept identical for both classifiers, except for the ratio of positive cases to negative cases in two training sets. Specifically, no normalization or outlier removal was applied to either dataset, but the instances of the minority class in one experiment were over-sampled by means of the SMOTE method (see chapter 4 for details) to a roughly 1:1 ratio with the majority class. After preparing two datasets, the variable subset selection procedure (Correlation-based Feature Selection with Best-First search) was then applied to both datasets. From a theoretical reasoning point of view, the default subset selection method might pick a different subset for each dataset, as the prior probabilities for all values of a variable and the posterior probabilities of the variable given the values of another variable might change after applying over-sampling (see chapter 4 for details). However, it was observed that the same variable subset was selected for both of them (Table 1).

The effect of balancing the CHHD dataset was tested through developing two contrastable models by certain classifier using the same variable subset. Logistic Regression was designated as the default classifier in the experiments accompanying by 10-fold cross-validation, so if the variable was recorded in both continuous and nominal types originally, the continuous ones were preferred.

The results showed that the overall accuracy actually decreased from 88.77% to 84.42% after applying over-sampling. However, if we take a closer look, it is obvious that the logistic regression model built from the dataset without over-sampling tended to classify negative cases more accurately than to classify positive cases, which suggested over-fitting for negative instances, but under-fitting for positive instances. The specificity for the negative class and sensitivity for the positive class respectively

in this model were 0.996 vs. only 0.037! In addition, the kappa statistic only achieved a low 0.0565, which meant that about 6% of the classifications made agreed with perfect classifier, after reducing the influence of chance. Moreover, although the average precision of the model developed without the minority class over-sampling was slightly better than that of the other model; something that is possibly attributed to the extreme high sensitivity (Table 1).

In contrast, the model built with over-sampling for the minority class classified both classes equally well. The sensitivity and specificity were 0.863 and 0.826 respectively. The area under the AUC curve and the kappa statistic all greatly outperformed the former model (Table 1). Therefore, balancing the imbalanced dataset by over-sampling was proved to be extremely beneficial and hence adopted for the following study.

Table 1: The performances of classification models developed either with or without applying over-sampling for minority class[2]

	Accuracy	Sensitivity	Specificity	Average Precision	AUC	Kappa
Model without over-sampling	88.77%	0.037	0.996	0.852	0.713	0.0565
Model with over-sampling	84.42%	0.863	0.826	0.845	0.915	0.6885

[2] Variable subset selected by the Best-First search and Correlation-based Feature Selection: AGE, EMPLOY, LANG, MSYS, MDIAS, BMI, WHR, PULSE, HDL, TRIG, CIG_DAY, BPWHEN, BPWHO, BPFRIED, WGTLOSE, SALTCOOK, SALTOTH, CHOLART, BCHOL, HDSALT, HDSMOKE, HDART, STRPREV.

7.1.4 Effect of Normalization

Based on the review of the literature on KDD methods, it was unclear whether variable normalization was advantageous or not for discovering knowledge in the CHHD database. Therefore, two experiments were executed to investigate the effect of normalization on classifier performance. In order to make the results comparable, the experimental settings were maintained unchanged in both experiments except for the normalization process. The over-sampling procedure for the minority class was added to both experiments due to the advantage of over-sampling demonstrated above. Logistic regression was specified as the default classifier in the experiments based on 10-fold cross-validation.

The experimental setting resulted in the construction of two models either with or without variable normalization. The control group model was the same one developed from the balanced dataset on previous step, and the other model was build by simply adding a normalization procedure, as described in chapter 4. After applying the exactly same method (BF+CFS) to both experiments, we observed that the same variable subset chosen in the previous step was once again selected regardless of the normalization procedure. The phenomenon occurred because the prior probabilities and posterior probabilities of variables did not change at all.

Interestingly, the results illustrated that the normalization did not exert any influence on classifiers' performance at all in the current dataset. All measurements were absolutely the same as in the control group model developed from the balanced CHHD dataset (Table 1). In addition, the time taken to build the model increased from 1.24 seconds to 2 seconds after normalization, which was attributed to the creation of the extra binary variables for the original nominal variables.

Two factors might contribute to the lack of improvement after normalization. First, when taking a closer look at the logistic regression equation developed without normalization, we can see that it treated each value of the nominal variables as a new variable by default, which was equivalent to the conversion of a nominal variable to a number of binary variables. Second, the ranges of the continuous variables were not large enough to exert an excessive influence on the results. For example, the largest range observed was for variable MSYS with only 104.5. As a result, we suggest not adopting the normalization process in preparing future datasets.

7.1.5 Effect of Outlier Removal

As explained earlier, outliers could be advantageous or disadvantageous because they can represent precious yet unusual information that may exist in real situations, while they may add noise to the main pattern found in the data. Therefore, without clear evidence it was not reasonable to exclude or replace outliers from the CHHD in developing the classification models. In this case, two additional experiments were carried out to assess the effect of outliers on the performance of the classifiers.

The dataset for the control group was prepared by over-sampling only, which was the same control group model used in last step. The other dataset was prepared by using both over-sampling and outlier removal. The outlier was defined as the value larger than the third quartile plus 3 times of interquartile range, and the value smaller than first quartile minus 3 times of interquartile range. Unexpectedly, 1,278 instances were identified as outliers out of 4,967 instances in dataset before over-sampling, which was about one fourth of the entire database. All outliers were removed from the

dataset consequently. After over-sampling the minority class by the SMOTE method, 3,260 negative instances and 3,260 positive instances were contained in database. Note that the outlier removal operation led to the deletion of two variables (CIGCAT and CIG_DAY), for they all had a single value now that was no longer informative to differentiate positive and negative cases. The variable subset selection method (*Correlation-based feature selection* with *best-first search*) was then applied to the dataset without outliers. This time it selected a subset slightly different from the subset chosen in the control model. Logistic regression was once again designated as the default classifier, and the performances of both models are shown in Table 2.

The model developed without outliers slightly outperformed the control model in terms of all six measurements. The overall accuracy of model without outliers was 85.49% whereas the overall accuracy of control model was 84.42%. For the model without outliers, it reached slightly higher sensitivity and specificity (0.844 for class 0 and 0.866 for class 1), and the sensitivity and specificity for the control model were 0.826 and 0.863 respectively. In addition, the average precision and AUC of model developed without outliers were 1% higher than that of control model. Moreover, the kappa statistic of model excluding outliers was more than 2% higher than that of control model, implying a larger agreement between classified and actual categorizations of a dataset, while correcting for an agreement that occurred by chance (Table 2).

In summary, the experimental results demonstrated the efficacy of outlier removal on the improvement of the classifier performance. Therefore, the dataset prepared by over-sampling of minority class and outlier removal without normalization was determined as the final dataset for later variable subset selection

and model construction by different learning algorithms.

Table 2: The performances of classification models developed either with or without outlier removal[3]

	Model with outliers	Model without outliers
Accuracy	84.42%	85.49%
Sensitivity	0.863	0.866
Specificity	0.826	0.844
Average Precision	0.845	0.855
AUC	0.915	0.925
Kappa	0.6885	0.7098

7.2 Comparing Three Variable Subset Selection Methods

As described above, the variable subset selection method comprises two components: feature space search method and evaluation function. Two feature space search methods were employed in present study, namely best-first search (BF) and genetic algorithm (GA), representing heuristic search and random search respectively. Similarly, two evaluation functions were applied as well, namely correlation-based feature selection (CFS) and naïve Bayes (NB), representing the filter and wrapper approaches individually. In order to find the optimal variable subset by these methods, three combinations of them were explored on the dataset prepared in last step. They

[3] Variable subset selected by the Best-First search and Correlation-based Feature Selection: AGE, EMPLOY, LANG, MSYS, MDIAS, BMI, WHR, PULSE, HDL, TRIG, CIG_DAY, BPWHEN, BPWHO, BPFRIED, WGTLOSE, SALTCOOK, SALTOTH, CHOLART, BCHOL, HDSALT, HDSMOKE, HDART, STRPREV for the model with outliers; AGE, EMPLOY, MARITAL, LANG, MSYS, MDIAS, WHR, PULSE, HDL, TRIG, BPWHEN, BPWHO, BPSATFAT, WGTLOSE, SALTCOOK, SALTDK, CHOLFAST, CHOLART, BCHOL, HDSALT, HDSTRESS, HDSMOKE, HDART, STRPREV for the model without outliers.

were best-first search plus correlation-based feature selection, genetic algorithm plus correlation-based feature selection, and best-first search plus naïve Bayes.

The experiments were separated into two steps. First, all three combinations were applied to the dataset presented above; so three distinct variable subsets were generated subsequently. Second, four classification models were developed from the datasets restricted by the three distinct variable subsets, as well as the dataset with all variables. In this way, the goodness of each combination was expressed as the performance of the corresponding model. All models were developed by logistic regression with 10-fold cross-validation technique in order to acquire a more general realization about each combination.

Since 10-fold cross-validation was utilized, ten different variable subsets were generated after ten iterations by each combination in step one. In WEKA, the selection result for each method was reported as a list of all variables with their summarized numbers, indicating how many times were they selected in ten iterations. In other words, the times that the variables were selected was a sign of their relative importance, so the variables selected more times than others were to be included into the final subset. Unfortunately, it was difficult to determine this threshold by looking at the empirical literature, for no studies have ever determined an appropriate number. For practical reasons, a five-fold was set as the threshold in the present study: variables that were selected for more than five times out of ten iterations were included into the final subset, and therefore transferred to the model construction step. Note that it was possible to include both continuous variables and nominal variables with same content. In this case, only continuous input variables were maintained in

order to perform logistic regression. The variable subset selected by each combination is listed in Table 3.

Table 3: The variable subsets selected by three methods: BF+NB, BF+CFS, GA+CFS[4]

Selection Methods	Variable Subset Selected
BF+NB	EDUC, EMPLOY, WORKTYPE, LANG, NPHARM, TCHOL, LDL, TRIG, BPWHEN, BPWHO, BPOKNOW, BPSATFAT, DBINS, WGTLOSE, WGTMEALS, SALTCOOK, SALTDK, FATOBS, CHOLFAST, CHOLART, BCHRX, BCHFAT, BCHWGT, HDWGT, HDSTRESS, HDTIRED, HDHBP, HDART, HDPREV, STRPREV, FEMRX
BF+CFS	AGE, EMPLOY, MARITAL, LANG, MSYS, MDIAS, WHR, PULSE, HDL, TRIG, BPWHEN, BPWHO, BPSATFAT, WGTLOSE, SALTCOOK, SALTDK, CHOLFAST, CHOLART, BCHOL, HDSALT, HDSTRESS, HDSMOKE, HDART, STRPREV
GA+CFS	AREA, SEX, AGE, EDUC, INCADEQ, WORKTYPE, MARITAL, LANG, MSYS, MDIAS, BMI, WHR, PULSE, TCHOL, HDL, TRIG, SMOKECAT, CIG_DAY, ALCOHOL, BPWHEN, BPHIGHTR, BPHEALTH, BPOTHMTS, BPDK, DBOTHER, WGTOTH, WGTDK2, SALTEDEM, SALTHA, SALTKIDN, SALTRX, SALTART, SALTDK, FATOBS, FATBP, CHOLPOUL, CHOLBEEF, CHOLSEA, CHOLMILK, CHOLOTH, CHOLDK1, CHOLART, CHOLOTHR, CHOLDK2, BCHHIGH, BCHSTRES, BCHRX, BCHOTH, HDDIET, HDWGT, HDSALT, HDCHOLBD, HDTIRED, HDHERED, HDHBP, HDDK, HDOTH

The three distinct variable subsets selected by three candidate methods were then used for developing logistic regression models. The efficacy of each variable subset was assessed through the classification performance of the corresponding logistic regression model. The experimental results were listed in Table 4.

The results demonstrated that there was no significant difference in terms of the efficacy among three selection methods. The most accurate classification for CVD was achieved by the subset selected by BF and NB with the accuracy of 88.08%, although the other two combinations brought about comparable results, with the

[4] BF = Best-First search; GA = Genetic Algorithm; NB = Naïve Bayes; CFS = Correlation-based Feature Selection

accuracy of 88.02% for GA and CFS, and 85.49% for BF and CFS. The sensitivity and specificity also implied that three selection approaches performed equally well, for the change amplitudes of them were approximately 2% and 4% respectively. The best sensitivity and specificity were observed for the BF+NB combination (0.876, 0.885), whereas the sensitivity and specificity for next two combinations were (0.884, 0.877) and (0.866, 0.844) respectively. The same conclusion could be drawn from the results of other performance measurements such as average Precision, AUC, and Kappa statistic as well.

Nonetheless, what was unexpected was that the logistic regression model developed without variable subset selection outperformed other three models. It exhibited the best performance with an accuracy of 90.86%. Other measurements also verified slightly better performance of it (Table 4). Two plausible explanations might contribute to the phenomenon. Firstly, the used of logistic regression was not precluded by the possible intercorrelation in the dataset even though the relative importance of individual variable was hard to assess. Secondly, more information was provided for the model compared to models constructed after subset selection. Therefore, the variable subset selection procedure remained essential for improving the classification performance and reducing the computational burden.

Table 4: Classification performance for Logistic Regression models developed from the variable subsets selected by BF+NB, BF+CFS, GA+CFS respectively, as well as the dataset without selection[5]

Selection Methods	Accuracy	Sensitivity	Specificity	Average Precision	AUC	Kappa
BF+NB	88.08%	0.876	0.885	0.881	0.941	0.7617
BF+CFS	85.49%	0.866	0.844	0.855	0.925	0.7098
GA+CFS	88.02%	0.884	0.877	0.880	0.933	0.7604
No selection	90.86%	0.898	0.919	0.909	0.954	0.8172

Although the variable subset selected by BF and CFS performed slightly worse on LR among all candidates, the method was still chosen for constructing further models for the proposed classifiers because of the following reasons: First, the method guarantees to filter out irrelevant and redundant variables that were believed to be detrimental to learning algorithms, such as naïve Bayes. More importantly, given the task of dimension reduction, the method gave the fewer number of variables (Table 3), which remarkably reduced the computational effort. Furthermore, unlike the subset suggested by GA and CFS, eighteen out of twenty-four variables in the subset selected by BF and CFS were repeatedly picked in all ten iterations, and only two variables were chosen less than seven times. Based on these considerations, concluded that the subset selected by BF and CFS had more practical significance than the other variable subset.

[5] BF = Best-First search; GA = Genetic Algorithm; NB = Naïve Bayes; CFS = Correlation-based Feature Selection

7.3 Assessing the Performance of Proposed Single Classifiers

Five proposed learning schemes are presented in this section in order to develop a classification model for CVD with the best performance. The five learning schemes were logistic regression, multilayer feed-forward network, naïve Bayes, C4.5 decision tree, and k-nearest neighbor. The dataset was prepared as described above: the process included variable pretreatment, missing nominal values deletion, missing continuous values imputation by EM, minority class over-sampling by SMOTE, outlier removal, and variable subset selection by BF and CFS with 10-fold cross-validation. The five individual learning algorithms were then applied to the prepared dataset. The performances of the five schemes are listed in Table 5.

The table shows that no single learning algorithm could consistently outperform any other classifier in terms of all performance measurements. However, the priorities of the five learning schemes would still be able to sort out.

Overall, the C4.5 decision tree (J48) was the best classifier among all learning algorithms, for it achieved the best overall accuracy as well as sensitivity and specificity, which are the most commonly used indicators for a classifier's performance. The accuracy of J48 on classification of CVD cases was 88.22%, and the corresponding sensitivity and specificity were 0.887 and 0.878 respectively. The decision tree scheme also produced the best score for average precision with the value of 0.882. The kappa statistic showed that the classifications made by J48 agreed with the observed classifications to a 76.44%, after removing the influence of chance. Nonetheless, when the performance was assessed by AUC, J48 could only take the fourth place.

Multilayer feed-forward network with 3 layers might be next best classifier

because it exhibited the second best performance in terms of accuracy, precision, and kappa statistic. The accuracy of MFN on classification of CVD cases was 86.60%, while correctly classifying 85.1% positive instances and 88.1% negative instances. The average precision of MFN was 0.866, and it agreed with the observed classifications to a 73.19% after removing the influence of chance. Unfortunately, it was also the most time consuming learning algorithm and the worst classifier evaluated by AUC.

K-nearest neighbor ranked in third place among five learning algorithms. It reached an accuracy of 86.03%, but it should also be noted that although KNN showed the highest sensitivity for the positive class, it also showed the lowest specificity for the negative class, with the value of 0.806 and 0.915 respectively. The higher accuracy for identifying positive cases can be largely attributed to the application of SMOTE method, which inserted synthetic positive instances as the neighbors of existing positive cases. The operation was also responsible for the fact that KNN covered the largest area under the ROC curve. KNN also showed the third best performance in terms of average precision and kappa statistic with the values of 0.865 and 0.7206. Moreover, KNN was the least time consuming learning scheme.

As a default classifier for previous experiments, logistic regression was exactly the same model built in the variable subset selection stage. All measurements except for AUC proved LR to be the second worst classifier. It reached an accuracy of 85.49%, while the sensitivity and specificity were 0.866 and 0.844 respectively. Generally speaking, 85.5% of all classifications made by LR in each class corresponded to real observations. It also agreed with perfect classifier to degree of 70.89% without the effect of chance. However, the logistic regression model covered

92.5% area under the ROC curve, indicating that it acquired current sensitivity with less false positives compared to MFN, NB, and J48.

Naïve Bayes was illustrated by all statistics but AUC as the worst learning scheme for classifying cases with CVD in current dataset. The accuracy of it was as low as 81.14%, and the sensitivity and specificity of it were also the lowest among all classifiers, with the value of 0.794 and 0.829 respectively. However, the AUC ranked it in the middle with the value of 0.895.

Table 5: The classification performance of LR, MFN, NB, J48, KNN, Bagging and AdaBoost M1 model in terms of accuracy, sensitivity, specificity, average precision, AUC, kappa, and time taken for model construction

Model	Accuracy	Sensitivity	Specificity	Average Precision	AUC	Kappa	Time taken for model construction
LR	85.49%	0.866	0.844	0.855	0.925	0.7098	0.89s
MFN	86.60%	0.851	0.881	0.866	0.873	0.7319	11.48s
NB	81.14%	0.794	0.829	0.812	0.895	0.6227	0.09s
J48	88.22%	0.887	0.878	0.882	0.888	0.7644	1.41s
KNN	86.03%	0.915	0.806	0.865	0.944	0.7206	~ 0s
Bagging	89.52%	0.889	0.902	0.895	0.943	0.7905	4.32s
AdaBoost M1	89.98%	0.896	0.903	0.900	0.948	0.7997	6.84s

7.4 Assessing the Performance of MetaCost Cost-sensitive Learning

Based on the general performance measured by the six statistics and the time

taken to build model, J48 was proved as the best single classifier among all five learning schemes, hence it was determined as the base classifier for MetaCost algorithm as well as the subsequently ensemble modeling. In MetaCost sensitive learning, the default ratio of false positive to false negative was set to 1: 4 as suggested by other researchers (Sampurno, 2006), but the effects of different ratios from 5:1 to 1:7 were tested as well. In order to acquire a more general realization about the efficacy of cost-sensitive learning on CHHD, 10-fold cross-validation technique was applied through out all experiments.

From the results listed in Table 6, several conclusions could be drawn. The best performance of J48-based cost-sensitive learning was observed when the ratio was 3:1. This cost matrix led to the accuracy of 88.80%, and the average precision, AUC, and kappa statistic were also the best among all ratios. However, the underlying mechanism assumed that mistakenly classifying an actual negative case was three times more costly than mistakenly classifying an actual positive case. To our knowledge, the ratio somehow contradicted the common sense about cardiovascular disease, that is, the false negatives should be more expensive than the false positives. In addition, the cost-sensitive learning achieved by MetaCost failed to outperform single J48 classifier under most conditions (different ratios). If the model with highest accuracy was preferred, the sensitivity was then no longer as satisfactory as single J48 model. Moreover, modifying the cost matrix towards the direction of making false negative instances more expensive would constantly increase the sensitivity to infinite while decreasing the specificity.

Table 6: Classification performance of MetaCost sensitive learning when varying cost matrix of false positive and false negative in terms of accuracy, sensitivity, specificity, average precision, AUC, kappa, and time taken for model construction

FP : FN	Accuracy	Sensitivity	Specificity	Average Precision	AUC	Kappa	Time
5 : 1	88.01%	0.833	0.927	0.883	0.887	0.7601	4.7s
4 : 1	88.21%	0.840	0.925	0.885	0.891	0.7641	7.33s
3 : 1	88.80%	0.853	0.923	0.890	0.898	0.7761	5.81s
2 : 1	88.60%	0.872	0.899	0.886	0.90	0.7718	4.73s
1 : 1	87.93%	0.878	0.881	0.879	0.892	0.7586	4.42s
1 : 2	87.42%	0.894	0.854	0.875	0.883	0.7485	7.58s
1 : 3	86.83%	0.906	0.831	0.870	0.873	0.7365	4.55s
1 : 4	85.58%	0.913	0.799	0.860	0.866	0.7117	4.07s
1 : 5	85.58%	0.921	0.790	0.862	0.869	0.7117	3.87s
1 : 6	84.20%	0.922	0.762	0.851	0.859	0.6840	4.11s
1 : 7	82.81%	0.928	0.729	0.842	0.853	0.6561	4.32s

7.5 Assessing the Performance of Ensemble Modeling through Bagging and AdaBoost M1 Techniques

Using J48 as the base classifier, two ensemble-learning models were constructed by using either the bagging or the boosting technique, specifically AdaBoost M1 algorithm. A 10-fold cross-validation technique was once again employed through out all experiments. It was proved that ensemble modeling indeed ameliorated the performance of classifying CVD cases to a minor extent in contrast to single J48 model (Table 5). Both methods achieved comparable performance despite the fact that the model developed by AdaBoost M1 slightly outperformed the bagging model in terms of all measurements. The AdaBoost M1 model was as accurate to

almost 90%, which was the best overall accuracy observed so far, whereas the bagging model achieved an accuracy of 89.52%. Both models tended to identify positive cases somewhat more accurately than to identify negative cases, but the variation was negligible. In addition, AdaBoost M1 required more computation and hence ran slower than bagging.

7.6 Summary

In summary, in this chapter I presented the results of all experiments conducted on the entire dataset, including the preparation process, efficacy of variable subset selection methods, performance of five basic classifiers, and capability of cost-sensitive learning and ensemble learning. The dataset was prepared through variable pretreatment, missing nominal values deletion, missing continuous values imputation by EM, minority class over-sampling by SMOTE, and outlier removal. Among three variable subset selection methods, the method of best first search and naïve Bayes wrapper achieved the best performance on prepared dataset; however, the method of best-first search and correlation-based feature selection was determined as the final selection method because it helped to remove redundant and irrelevant variables, and the subset suggested was more significant than other subsets. After comparing the performance of five single classifiers, C4.5 decision tree was shown to be the best basic learning scheme on the classification of CVD cases in the CHHD, so it was adopted as the base learner for cost-sensitive learning and ensemble learning. Cost-sensitive learning was realized by MetaCost algorithm, and the results were unpromising no matter how cost matrix was assigned. Ensemble learning was accomplished by bagging technique and AdaBoost M1 algorithm. Both models

improved the overall performance on the classification of CVD cases on CHHD to a small extent, although the later was slightly better. As a result, we suggest that a C4.5 decision tree based boosting model achieved by AdaBoost M1 algorithm may be very useful for discovering knowledge in databases similar to the CHHD used in this study.

CHAPTER 8
Discussion

Despite the successful completion of building a classification model for cardiovascular disease using a Canadian population, many practical issues and study limitations were found during our exercise. This chapter addresses these issues according to the order in which they have appeared in the thesis, with detailed discussions of the issues involved, and also provides some hopefully inspiring suggestions for future research.

The first and foremost problem encountered during the development of our model was the lack of some potentially relevant data that may have produced more accurate results. As described earlier, the CHHD database explored in present study contains 265 variables and 23,129 records collected from all ten provinces across Canada, which we believe it to be large enough for discovering knowledge that may be valid and useful. However, numerous important data points in the feature space are missing due to a variety of reasons. Much of these data were biological markers and others were of a more social nature. For example, some variables that failed to be collected in several provinces at the first place included blood samples or some clinically-relevant measurements. The incompleteness of the database resulted in the deletion of valuable instances, and hence restricted the availability of data for the KDD approaches used in the study. It is reasonable to suggest that the more complete the data is, the more robust and generalizable the model would be, so it is greatly encouraged to mining datasets that are as complete and as large as possible.

When dealing with missing data in our work, the issue of "missingness mechanisms"—which is the term used in the KDD community to refer to the patterns of missing values—is overlooked. This is an important issue because the methods for dealing with missing values make assumptions about the underlying patterns for missing values. "Missingness mechanisms" serve to describe how the missing values are distributed within the dataset. In this regard, researchers have divided the missingness mechanisms into three broad categories: (1) Missing Completely at Random (MCAR) refers to the presence of missing values in one variable are unrelated to the value of the variable itself, or to values of any other variable observed in the dataset; (2) Missing at Random (MAR) refers to data where the presence of missing values in one variable is unrelated to missing data in another variable, but may be related to the value of the variable itself; and (3) Not Missing at Random (NMAR) occurs in all other cases (Nisbet et al., 2009; Scheffer, 2002).

The Expectation-Maximization algorithm employed in the present study imputes the missing values properly only under the assumption that the underlying "missingness mechanism" is MAR (Scheffer, 2002). Song et al. indicated that it is reasonable to assume the "missingness mechanism" as MAR even for small datasets (Song, Shepperd, & Cartwright, 2005), but the assumption may not hold true for the CHHD data set. In addition, Scheffer (2002) recommended not using case deletion as was done to several instances with missing nominal values in this study, unless the data is definitely MCAR. However, we have no confidence in holding this assumption for the CHHD data set. Therefore, future studies should explore the underlying "missingness mechanism" of the datasets researchers have available. Also, more evidence is needed about the effects of misassumptions of the underlying

“missingness mechanism."

The learning algorithms mechanically regard all data points in the feature space as what people have truly observed. However, a large amount of data is actually generated by missing value imputation methods and dataset balancing methods. For example, in our case the EM algorithm and SMOTE method were applied to produce the data for empty cells and synthetic data for minority class. Since the dataset is not completely real, people should not completely trust what has been learned from the data set, including the models we just developed. The best solution to this issue is what we suggested earlier, namely, mining datasets that are as complete as possible. But generating (or having access to) complete data sets is extremely difficult to achieve most of the time. We suggest reducing the weights of artificial data on modeling, while proposing more research related algorithms.

Since the medical domain, and more specifically cardiovascular health, is potentially one the major fields of application of KDD, analysts may have to deal with imbalanced datasets (IDS) most of the time because the occurrence of diseases is typically a rare incident. As discussed earlier, the IDS problem could be solved using two approaches: either at the algorithm level or at the dataset level. At the dataset level, our work only showed that the over-sampling worked well in improving the classification accuracy for the minority class. However, under-sampling has also been proposed as a proper dataset balancing technique. Under-sampling refers to the techniques that cut out noisy, boundary and redundant data points primarily for the majority class. The commonly used under-sampling methods are various (Lin et al., 2008). They heuristically utilize Euclidean distance, or K-nearest rules, to identify samples that can be deleted safely (Lin et al., 2008). However, domain experts (Lin et

al., 2008) have argued that noisy, boundary, and redundant samples only occupy a small portion of the feature space, so the effect of under-sampling may not be a promising strategy. For this reason, the under-sampling method was not employed in this study. However, the outlier detection result indicated that such data points in fact comprised a large proportion of the CHHD, so it may be necessary to test the effectiveness of under-sampling techniques in this dataset.

Furthermore, investigations that combine both techniques may be needed.

The variable subset selection was applied for the purpose of dimension reduction, but other methods exist for achieving this objective. A well-known method is Principal Component Analysis (PCA), which is commonly used in the analysis of multivariate data, and has been widely applied to reduce the dataset dimensionality (Larose, 2005; Witten et al., 2011). PCA tries to find a coordinate system that can partition all the data points in the feature space with maximum variances in each direction, indicating the degree of spread around the mean value in that direction, where each axis of this coordinate system is perpendicular to the others (Li, 2010; Witten et al., 2011). The total variance remain constant regardless of the coordinate system, and the axes called components account for their own share of total variances, so people just need to select the principal components making up most variance to reduce the dimensionality. The problem associating with PCA is that the principal components selected make no sense to analyst because they are different from the traditional coordinate system by which people understand the dataset; however, it is encouraged to investigate the effect of PCA on dimension reduction on current dataset.

In daily life we often see the case where people tend to seek for best solutions for the same problem from distinct approaches in which they are most skilled. The

situation also applies to KDD approaches because each of them accomplishes the same task from a completely different perspective, using different mechanisms (e.g., naïve Bayesian classifiers make predictions from conditional probabilities, whereas decision trees give results by inducing decision rules understandable to humans). Therefore, it is unclear about which learning mechanism is the best one for the task at hand until the different algorithms have actually been applied and compared, even though in many situations researchers and users often make the mistake by presuming the best approach (see Nisbet et al., 2009).

Moreover, only five classifiers were employed in the study due to their popularity and relatively long history of investigation, but many more learning schemes exist. To name a few, Support Vector Machine (SVM) and Rule-based classifiers are other possibilities to classify cases with cardiovascular disease in our dataset. Therefore, it would be recommended to examine the efficacy of other classifiers either from other approaches not mentioned in the study or different from the representative methods employed within same approach on the CHHD as well as others.

In this study, ensemble modeling was shown to be an effective tool for enhancing classifier performance. Researchers have given credits to three reasons to explain the phenomenon. The first reason is the insufficient information provided by the training data for choosing a single best classifier; the second reason is the imperfect search processes of the learning algorithms; the third reason is that ensembles can give some good approximations of true target function in hypothesis space (Zhou, 2009). The limitation of our work on ensemble modeling is that only homogeneous classifiers were combined despite the fact that there are some

differences on model parameters. However, Zhou (2009) has summarized the methods of heterogeneous ensemble learning, that is combining different classification algorithms such as Stacking method (Zhou, 2009), so it is appreciated to appraise the effect of heterogeneous ensemble modeling and hopefu2lly compare it with that of bagging and boosting methods.

The efficacy of cost-sensitive learning on the CHHD database is studied through changing the cost matrix in MetaCost algorithm. The experimental results demonstrated that the best performance was achieved when misclassifying negative instances costs three times more than misclassifying positive instances, but the observation does not correspond to current knowledge of cardiovascular disease. Moreover, although one study had assumed to use 1:4 as the default ratio of false positive to false negative as mentioned earlier (Sampurno, 2006), no research has ever confirmed what the real cost matrix should be. Therefore, the exact ratios of false negatives to false positives for not only the cardiovascular disease but also other disease are demanded urgently. Furthermore, the cost-sensitive learning did not significantly improve the classifier's performance in this study, but it might be due to the over-sampling procedure already used, so the impact of cost-sensitive learning should also be investigated on the original imbalanced CHHD. If it remained incapable of enhancing performance compared with the single classifier, then the cost-sensitive learning could be suggested to be inefficient in coping with IDS problem.

One significant limitation of the present research was the absence of a validation dataset. Recall that in the 10-fold cross-validation method, each fold has been taken as training set and testing set, but both of them are from the same dataset;

otherwise stated, the classification pattern was learned and tested by the same distribution, which led to generally better performance than the true accuracy. Therefore, an external validation set separate from the original dataset would be informative to evaluate how good the model truly is.

In summary, the chapter discussed the practical issues and study limitations in the process of developing the KDD model for the classification of cases with CVD. Even though the CHHD is a large database, the data was still found to be less than ideal due to the presence of large amounts of redundant and irrelevant data. When dealing with missing values, the underlying "missingness mechanism" was overlooked. Since some data points in the feature space were artificially generated, they should not be treated as the observed data. The IDS problem is pervasive in the medical and health domains, so it's worth to investigate the potential of under-sampling. Variable subset selection has been proven to be an efficient way of dimension reduction, but alternative methods such as PCA are available. Only five single classifiers and homogeneous ensemble learning methods were examined in the study, but attention should also go to other powerful KDD learning schemes as well as heterogeneous ensemble learning methods. Although cost-sensitive learning failed to demonstrate a promising result, more research on the real cost-matrix for cardiovascular disease is needed. One last limitation of the study comes from the absence of an external validation set, which would test the general validity of those classifiers.

CHAPTER 9
Conclusion and Recommendations

In the present study, the process of developing a KDD model for classifying cases with cardiovascular disease was investigated. The entire KDD process mainly comprises of five steps: dataset selection, dataset preprocessing, variable subset selection, data-mining methods application, and knowledge interpretation and evaluation methods, which were all covered.

The CHHD was chosen as the appropriate dataset because it is a Canadian-based heart health database and contains not only possible risk factors but also people's knowledge about CVD. The whole CHHD is a very large data set and consists of 265 variables and 23,129 instances. Additionally, all variables included in the CHHD can be easily obtained from either questionnaires or simple clinical measurements, so the eventual classification model might be appropriate to apply for population screening of CVD.

9.1 Lessons Learned

The original CHHD is incomplete and noisy, so it had to be cleaned up before entering next step. In all, 114 out of 265 variables were removed from the dataset because they were not relevant for the current investigation, some have overlapping information with other variables, and some were recorded in different scales or formats. One new response variable, labeled as CVD, was created to indicate the CVD status as a whole, which included several cardiovascular diseases.

Missing values for continuous variables were imputed by the Expectation-

Maximization method, and missing values for nominal variables were removed along with whole instance. The CHHD was essentially imbalanced, which caused low classification accuracy for minority classes, so the Synthetic Minority Over-sampling Technique was applied to balance the database. The results showed that over-sampling the minority class significantly improved the overall classification performance. In order to control the excessive influence brought by larger scales of some variables, a variable normalization was utilized, but the control experiment showed no enhanced classification performance. Outliers indicate rare, yet real, patterns but brought noise to the main knowledge, and their real influence on the CVD classification model was investigated. The experimental result of this study demonstrated that the model developed without outliers slightly outperformed the control model in terms of all performance measurements, so we suggest that might be better to remove outliers in further KDD studies about CVD health, using similar data sets to the CHHD.

After preparing the dataset, it is crucial to apply variable subset selection, which comprises two components: subset generation and evaluation function. In the present study, we used best-first search and genetic algorithm as the subset generation methods, and naïve Bayes and correlation-based feature selection as the evaluation function, representing wrapper and filter approach respectively. For the purpose of finding the best selection approach, three combinations of them were compared in terms of classification performance on a logistic regression model. It was shown that the best-first search and naïve Bayes wrapper obtained the best accuracy, but the method did not guarantee the independence of each predictor variable. The method of genetic algorithm and correlation-based feature selection achieved the second best

result, but the subset selected was relatively insignificant and large, which failed to satisfy the requirement of variable subset selection. The method of best-first search and correlation-based feature selection was eventually determined as the proper method, for it inhibited irrelevant and redundant variables, and picked the smallest set of predictor variables with more significance, although the method produced a slightly worse result (less than 3%) than the results produced by previous two combinations.

Five KDD learning algorithms were used in this study, namely logistic regression, multilayer feed-forward network, naïve Bayes, C4.5 decision tree, and k-nearest neighbor. After comparing five single classifiers' performance including accuracy, sensitivity, specificity, average precision, AUC and kappa statistic, C4.5 decision tree was proved to be the best basic learning scheme on the classification of CVD cases on CHHD, for it achieved the highest accuracy of 88.22% and was sensitive to both classes. As a result, the C4.5 decision tree was adopted as the base learner for cost-sensitive learning and ensemble learning. Cost-sensitive learning was exemplified by the MetaCost algorithm, and the results were unpromising no matter what cost matrix was assigned. Ensemble learning was accomplished by the bagging technique and the AdaBoost M1 algorithm. Both models improved the overall performance on the classification of CVD cases on CHHD to a small extent, although the later was slightly better. Therefore, a C4.5 decision tree based boosting model achieved by AdaBoost M1 algorithm was recommended as a potentially useful tool for discovering knowledge in databases similar to CHHD explored in this study.

9.2 Limitations and Recommendations

It is necessary to acknowledge that the conclusion of our study might not hold true constantly as many internal factors may influence the validation of our model. Before any attempt to generalize the model to actual practice, further evaluations upon other criteria are highly recommended.

9.2.1 Validation Issues

First, in the variable pretreatment stage, 114 variables were deleted due to the aforementioned various reasons. However, these actions may introduce some bias to the model. For example, the variable "PROVINCE" might be related to the occurrence of cardiovascular disease to some degree, for it can reflect differences in the lifestyle of respondents from different regions that are not embodied by other retained variables. In addition, even though the data regarding the variable "PROVINCE" was incomplete (some provinces were deleted from the data set in the nominal missing values deletion stage), the data can still provide some valuable information. Moreover, some variables were deleted from the dataset because they overlapped with other related variables, but the analysis was conducted based on the assumption that they were identical to the ones deleted; however, this might not be always true. For instance, occasionally people who had been educated for 20 years might still not have a degree yet, in which case, the educational year does not reflect their educational level. With this in mind, it might be better to conduct the entire KDD process using the complete dataset with all variables as the baseline for comparison with the final model, i.e. the one developed by the AdaBoost M1 algorithm using C4.5 decision tree as the base classifier, although maintaining all

variables means to shrink the original CHHD to a very small scale.

Second, when removing or selecting variables they shall be appraised independently by the domain experts such as cardiologists and cardiovascular epidemiologists. Their domain expertise and experience provide another way to evaluate the quality of the final model. Their expert opinion would be helpful to determine the significance of the variables that were excluded from the data set at the variable pretreatment stage, or included in the variable subset at the feature selection stage.

Third, speaking of the variable significance, it is important to check the contribution of each selected variable to the total accuracy. This could be done by comparing the model developed from the dataset without the target variable with the original model, where the difference in performance would yield the significance of the target variable. Alternatively, it can also be done by identifying a small subgroup according to certain criteria, such as educational level or diabetes status because different subgroups may involve different predictors to the occurrence of CVD. If the model developed from the subpopulation includes the very same variables as our final model, they are then more significant in general than others.

Fourth, in the variable subset selection stage, the technique of 10-fold cross-validation was employed, and the fold number of five was set as the threshold for final subset selection based on the mechanism of majority voting. However, this setting is not the golden rule and could be narrowed down to a smaller number so that more variables would be selected into the subsets. Therefore, another model could be developed from the dataset with a larger scale of variable subset for comparison with our model, and if our model showed a better performance we can then conclude that

our model is suitable to discover knowledge from dataset like the CHHD with more confidence.

Finally, an external validation dataset is also crucial for assessing the real quality and characteristics of our model before deploy it into a production environment. The dataset was collected by different investigators from different institutions. External validation is a more rigorous procedure to determine whether our model will generalize to populations other than the one on which it was developed. Also, the model could be further adjusted based on the result generated from the external validation dataset.

9.2.2 Methodological Issues

As already discussed in the last chapter, several other improvements can be made to this study. First, the "missingness mechanism" was overlooked in this study, but our results might be established on the wrong assumption, so it is encouraged to verify the real "missingness mechanism" of the CHHD and other heart health databases.

Second, when mining the prepared dataset, the KDD learning algorithms treat all data points as observed data, which may mask the real performance of the classifiers. Thus we propose an idea of reducing the weights of artificial data on modeling, while encouraging research using related algorithms.

Third, our study only demonstrated the efficacy of the over-sampling method, but under-sampling of the majority class might also be promising in our study because of the existence of large amounts of outliers, so it is necessary to test the efficacy of under-sampling method as well.

Fourth, other dimension reduction methods that are commonly used in many domains such as Principal Component Analysis are recommended to compare with the variable subset selection method determined in present research.

Fifth, only five KDD learning schemes were examined, but a number of other classifiers from either other KDD approaches should also gain our attention, such as Support Vector Machine and Rule-based classifiers, so it is still too early to conclude that C4.5 decision tree is the best classifier for the CHHD.

Sixth, the study only combined several homogeneous classifiers with the variations in model parameters. However, it is also possible to combine different classification algorithms through other methods, such as Stacking (Zhou, 2009), which refers to heterogeneous ensemble learning, such that we can appraise the effect of heterogeneous ensemble modeling, and hopefully compare it with that of bagging and boosting methods.

Seventh, the studies of cost matrix in cardiovascular health are needed in order to implement cost-sensitive learning to classify CVD cases.

Last but not least, one limitation of present study is the absence of an external validation set, so it is recommended to test our eventual model on an external validation dataset in order to understand its real validity.

Bibliography

Aha, D. W., Kibler, D., & Albert, M. K. (1991). Instance-based learning algorithms. Machine Learning, 6(1), 37–66. doi:10.1007/BF00153759

Akay, M. (1992). Noninvasive diagnosis of coronary artery disease using a neural network algorithm. Biological cybernetics, 67(4), 361–7. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/1515514

Anbarasi, M., Anupriya, E., & Iyengar, N. (2010). Enhanced prediction of heart disease with feature subset selection using genetic algorithm. Int J Eng Sci Technol, 2(10), 5370–5376. Retrieved from http://www.ijest.info/docs/IJEST10-02-10-171.pdf

Ashare, a B., & Chakraborty, D. P. (1994). Artificial neural networks: better than the real thing? Journal of nuclear medicine : official publication, Society of Nuclear Medicine, 35(12), 2048–9. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/7989990

Baldassarre, D., Grossi, E., Buscema, M., Intraligi, M., Amato, M., Tremoli, E., Pustina, L., et al. (2004). Recognition of patients with cardiovascular disease by artificial neural networks. Annals of medicine, 36(8), 630–40. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/15768835

Bath, P. A. (2004). Data mining in health and medical information. Annual Review of Information Science and Technology, 38(1), 331 – 369. doi:10.1002/aris.1440380108

Baxt, W G, & Skora, J. (1996). Prospective validation of artificial neural network trained to identify acute myocardial infarction. Lancet, 347(8993), 12–5. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/8531540

Baxt, William G, Shofer, F. S., Sites, F. D., & Hollander, J. E. (2002a). A neural network aid for the early diagnosis of cardiac ischemia in patients presenting to the emergency department with chest pain. Annals of emergency medicine, 40(6), 575–83. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/12447333

Baxt, William G, Shofer, F. S., Sites, F. D., & Hollander, J. E. (2002b). A neural computational aid to the diagnosis of acute myocardial infarction. Annals of emergency medicine, 39(4), 366–73. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/11919522

Bilmes, J. (1998). A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models. International Computer Science Institute. Retrieved from http://lasa.epfl.ch/teaching/lectures/ML_Phd/Notes/GP-GMM.pdf

Chawla, N., Bowyer, K., Hall, L., & Kegelmeyer, W. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321–357. Retrieved from http://arxiv.org/abs/1106.1813

Colak, M. C., Colak, C., Kocatürk, H., Sağiroğlu, S., & Barutçu, I. (2008). Predicting coronary artery disease using different artificial neural network models. Anadolu kardiyoloji dergisi : AKD = the Anatolian journal of cardiology, 8(4), 249–54. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/18676299

Colombet, I., Ruelland, A., Chatellier, G., Gueyffier, F., Degoulet, P., & Jaulent, M. C. (2000). Models to predict cardiovascular risk: comparison of CART, multilayer perceptron and logistic regression. Proceedings / AMIA Annual Symposium. AMIA Symposium (pp. 156–60). Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2244093&tool=pmcentrez&rendertype=abstract

Dash, M., & Liu, H. (1997). Feature Selection for Classification. Intelligent data analysis, 1, 131 – 156. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.39.6038

Domingos, P. (1999). Metacost: A general method for making classifiers cost-sensitive. Retrieved from http://dl.acm.org/citation.cfm?id=312220

Eapen, A. G. (2004). Application of Data mining in Medical Applications. University of Waterloo.

Fawcett, T. (2006). An introduction to ROC analysis. Pattern Recognition Letters, 27(8), 861–874. doi:10.1016/j.patrec.2005.10.010

Fayyad, U., Piatetsky-shapiro, G., & Smyth, P. (1996). From Data Mining to Knowledge Discovery in. AI Magazine, 17(3), 37–54.

Finch, W. H. (2010). Imputation Methods for Missing Categorical Questionnaire Data : A Comparison of Approaches. Journal of Data Science, 8, 361–378.

Freund, Y., & Schapire, R. (1996). Experiments with a new boosting algorithm. Machine Learning: Proceedings of the Thirteenth International Conference. Retrieved from http://www.public.asu.edu/~jye02/CLASSES/Fall-

2005/PAPERS/boosting-icml.pdf

Freund, Y., Schapire, R., & Abe, N. (1999). A short introduction to boosting. Journal of Japanese Society for Artificial Intelligence, 14(5), 771–780. Retrieved from http://www.yorku.ca/gisweb/eats4400/boost.pdf

Green, M., Björk, J., Forberg, J., Ekelund, U., Edenbrandt, L., & Ohlsson, M. (2006). Comparison between neural networks and multiple logistic regression to predict acute coronary syndrome in the emergency room. Artificial intelligence in medicine, 38(3), 305–18. doi:10.1016/j.artmed.2006.07.006

Hall, M. (1999). Correlation-based feature selection for machine learning. Retrieved from http://www.lri.fr/~pierres/donn%E9es/save/these/articles/lpr-queue/hall99correlationbased.pdf

Harrison, R. F., & Kennedy, R. L. (2005). Artificial neural network models for prediction of acute coronary syndromes using clinical data from the time of presentation. Annals of emergency medicine, 46(5), 431–9. doi:10.1016/j.annemergmed.2004.09.012

Hartwig, F., & Dearing, B. E. (1980). Exploratory Data Analysis (Quantitative Applications in the Social Sciences) (p. 88). Sage Publications, Inc. Retrieved from http://www.amazon.com/Exploratory-Analysis-Quantitative-Applications-Sciences/dp/0803913702

Itchhaporia, D., Snow, P. B., Almassy, R. J., & Oetgen, W. J. (1996). Artificial neural networks: current status in cardiovascular medicine. Journal of the American College of Cardiology, 28(2), 515–21. doi:10.1016/0735-1097(96)00174-X

Japkowicz, N., & Stephen, S. (2002). The class imbalance problem: A systematic study. Intelligent Data Analysis, 6(5), 429–449. Retrieved from http://dl.acm.org/citation.cfm?id=1293951.1293954

Kennedy, R. (1997). An artificial neural network system for diagnosis of acute myocardial infarction (AMI) in the accident and emergency department: evaluation and comparison with serum myoglobin measurements. Computer Methods and Programs in Biomedicine, 52(2), 93 – 103. doi:10.1016/S0169-2607(96)01782-8

Kohavi, R., & John, G. (1997). Wrappers for feature subset selection. Artificial intelligence, (97), 273–324. Retrieved from http://www.sciencedirect.com/science/article/pii/S000437029700043X

Larose, D. T. (2004). Discovering Knowledge in Data: An Introduction to Data Mining (p. 240). Wiley-Interscience. Retrieved from http://www.amazon.com/Discovering-Knowledge-Data-Introduction-Mining/dp/0471666572

Larose, D. T. (2005). Data Mining Methods and Models (p. 344). Wiley-IEEE Press. Retrieved from http://www.amazon.com/Mining-Methods-Models-Daniel-Larose/dp/0471666564

Lee, D. S., Chiu, M., Manuel, D. G., Tu, K., Wang, X., Austin, P. C., Mattern, M. Y., et al. (2009). Trends in risk factors for cardiovascular disease in Canada: temporal, socio-demographic and geographic factors. CMAJ : Canadian Medical Association journal = journal de l'Association medicale canadienne, 181(3-4), E55–66. doi:10.1503/cmaj.081629

Li, L. (2010). Statistical Methods in Molecular Biology. (H. Bang, X. K. Zhou, H. L. Epps, & M. Mazumdar, Eds.) (Vol. 620, pp. 417–434). Totowa, NJ: Humana Press. doi:10.1007/978-1-60761-580-4

Lin, Z., Hao, Z., & Yang, X. (2008). Current state of research on imbalanced data sets classification learning. application research of computers, 25(2), 332–336.

Long, W. J., Griffith, J. L., Selker, H. P., & D'Agostino, R. B. (1993). A comparison of logistic regression to decision-tree induction in a medical domain. Computers and biomedical research, an international journal, 26(1), 74–97. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/8444029

Longo, D., Fauci, A., Kasper, D., Hauser, S., Jameson, J., & Loscalzo, J. (2011). Harrison's Principles of Internal Medicine: Volumes 1 and 2, 18th Edition (p. 4012). McGraw-Hill Professional. Retrieved from http://www.amazon.com/Harrisons-Principles-Internal-Medicine-Volumes/dp/007174889X

Mendis, S., Puska, P., & Norrving, B. (2011). WHO | Global atlas on cardiovascular disease prevention and control (pp. 1–164). Geneva: World Health Organization. Retrieved from http://www.who.int/cardiovascular_diseases/publications/atlas_cvd/en/index.html

Nisbet, R., Elder, J., & Miner, G. (2009). Handbook of Statistical Analysis and Data Mining Applications (p. 864). Academic Press. Retrieved from http://www.amazon.com/Handbook-Statistical-Analysis-Mining-Applications/dp/0123747651

Palaniappan, S., & Awang, R. (2008). Intelligent heart disease prediction system using data mining techniques. 2008 IEEE/ACS International Conference on Computer Systems and Applications, 8(8), 108–115. doi:10.1109/AICCSA.2008.4493524

Phillips, K. T., & Street, W. N. (2005). Predicting outcomes of hospitalization for heart failure using logistic regression and knowledge discovery methods. AMIA Annual Symposium proceedings / AMIA Symposium. AMIA Symposium (Vol. 2005, p. 1080). American Medical Informatics Association. Retrieved from /pmc/articles/PMC1560853/?report=abstract

Rajkumar, A., & Reena, G. S. (2010). Diagnsis of Heart Disease Using Datamining Algorithm. Global Journal of Computer Science and Technology, 10(10), 38–43.

Ristov, S., & Peckov, A. (2008). Machine Learning Approach for Early Detection of Cardiovascular Deceases (CVD). Retrieved October 1, 2012, from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4493524

Sampurno, F. (2006). Identifying risk factors associated with new onset cardiovascular disease in patients with type I diabetes using Classification Tree. The University of Melbourne. Retrieved from http://www.ms.unimelb.edu.au/publications/FannySampurnoHonoursThesis.pdf

Schafer, J. L. (1999). Statistical Methods in Medical Research. Statistical Methods in Medical Research, 8, 3–15. doi:10.1177/096228029900800102

Scheffer, J. (2002). Dealing with missing data. Research letters in the information and mathematical science, (3), 153–160. Retrieved from

http://equinetrust.org.nz/massey/fms/Colleges/College of Sciences/IIMS/RLIMS/Volume03/Dealing_with_Missing_Data.pdf

Selker, H. P., Griffith, J. L., Patil, S., Long, W. J., & D'Agostino, R. B. (1995). A comparison of performance of mathematical predictive methods for medical diagnosis: identifying acute cardiac ischemia among emergency department patients. Journal of investigative medicine : the official publication of the American Federation for Clinical Research, 43(5), 468–76. Retrieved from http://ukpmc.ac.uk/abstract/MED/8528758/reload=0

Song, Q., Shepperd, M., & Cartwright, M. (2005). A Short Note on Safest Default Missingness Mechanism Assumptions. Empirical Software Engineering, 10(2), 235–243. doi:10.1007/s10664-004-6193-8

Soni, J., Ansari, U., Sharma, D., & Soni, S. (2011). Predictive Data Mining for Medical Diagnosis: An Overview of Heart Disease Prediction. International Journal of Computer Applications, 17(8), 43–48. doi:10.5120/2237-2860

The Heart and Stroke Foundation. (2011). The Heart and Stroke Foundation's 2011 Annual Report on Canadians' Health. Retrieved from http://www.heartandstroke.com/atf/cf/{99452d8b-e7f1-4bd6-a57d-b136ce6c95bf}/lr-FINAL-Reportcard HSF_2011-english.pdf

Tsien, C. L., Fraser, H. S., Long, W. J., & Kennedy, R. L. (1998). Using classification tree and logistic regression methods to diagnose myocardial infarction. Studies in health technology and informatics, 52 Pt 1, 493–7. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/10384505

Witten, I. H., Frank, E., & Hall, M. A. (2011). Data Mining: Practical Machine Learning Tools and Techniques, Third Edition (The Morgan Kaufmann Series in Data Management Systems) (Third., p. 664). Morgan Kaufmann. Retrieved from http://www.amazon.com/Data-Mining-Practical-Techniques-Management/dp/0123748569

Ye, Z., Wen, Y., & Lv, B. (2009). A survey of imbalanced pattern classification problems. CAAI Transactions on Intelligent Systems, 4(2), 148–156.

Yu, L., & Liu, H. (2003). Feature selection for high-dimensional data: A fast correlation-based filter solution. Proceedings of the Twentieth International Conference on Machine Learning. Washington. Retrieved from https://www.aaai.org/Papers/ICML/2003/ICML03-111.pdf

Zhou, Z. (2009). Ensemble learning. Encyclopedia of Biometrics (pp. 270–273). Berlin: Springer. Retrieved from http://www.ai.kun.nl/aicourses/bki212a/materiaal/tr.pdf

Appendices

Appendix 1

Author	Year	Disease	Instances	Variables	Algorithm	Measurement	Country
Akay, M.	1992	CAD	100	15	ANN	Sensitivity: 84% Specificity: 89%	U.S.
Anbarasi M. et al.	2010	HD	909	6	DT	Accuracy: 99.2%	U.S.
					NB	Accuracy: 96.5%	
Baldassarre, D., et al.	2004	CVD	949	54	ANN	Accuracy: 83%	Italy
				37	ANN	Accuracy: 85%	
Baxt, W.G., et al.	1996	MI	1,070	20	ANN	Sensitivity: 96% Specificity: 96%	U.S.
Baxt, W.G., et al.	2002	MI	2,204	40	ANN	Sensitivity: 94.5% Specificity: 95.9% AUC: 0.982	U.S.
					LR	Sensitivity: 77.3% Specificity: 75% AUC: 0.870	
Baxt, W.G., et al.	2002	CI	2,204	40	ANN	Sensitivity: 88.1% Specificity: 86.2% AUC: 0.900	U.S.
					LR	Sensitivity: 72.1% Specificity: 77.4% AUC: 0.831	
Colombet, I., et al.	2000	CVD	15,444	10	ANN	Accuracy: 76% AUC: 0.78	France
					LR	Accuracy: 65.9% AUC: 0.78	
Colak, M.C., et al.	2008	CAD	237	17	ANN	Accuracy: 92%† Sensitivity: 96%† Specificity: 91%†	Turkey
Green, M.,	2006	ACS	634	38	ANN	AUC: 0.791	Sweden

et al.					LR	AUC: 0.757	
				16	ANN	AUC: 0.802	
					LR	AUC: 0.705	
Harrison, R.F., et al.	2005	ACS	3,147	13	ANN	Sensitivity: 91.5%§ Specificity: 91.0%§ AUC: 0.94§	U.K.
Kennedy, R.L., et al.	1997	MI	290	53	ANN	Accuracy: 91.8% Sensitivity: 91.2% Specificity: 90.2%	U.K.
Long, W.J.	1993	CI	5,773	7	LR	Accuracy: 83.76%	U.S.
				52	DT	Accuracy: 80.13%	
Palaniappan S. et al.	2008	HD	909	15	NB	Accuracy: 86.53%	U.S.
					ANN	Accuracy: 86.12%	
					DT	Accuracy: 85.68%	
Phillips, K.T., et al.	2005	HF	2,500	>12	ANN	AUC: 0.802	U.S.
					LR	AUC: 0.734	
Rajkumar A, et al.	2010	HD	3,000	14	NB	Accuracy: 52.33%	Unknown
					DT	Accuracy: 52%	
Ristov et al.	2010	CVD	1,682	11	KNN	Accuracy: 45.67%	Macedonia
					LR	Accuracy: 73.01%	
					NB	Accuracy: 72.24%	
					DT	Accuracy: 71.76%	
Selker, H.P., et al.	1995	CI	5,773	8	ANN	AUC: 0.902	U.S.
					LR	AUC: 0.887	
				96	ANN	AUC: 0.923	
					LR	AUC: 0.905	
Tsien, C.L. et al.	1998	MI	1,752	45	DT	Accuracy: 94.04% AUC: 94.28%	U.K.

		LR	Accuracy: 89.61% AUC: 89.28%

Abbreviations and acronyms
ACS: acute coronary syndrome
AUC: area under the curve.
CAD: coronary artery disease
CI: cardiac ischemia
CVD: cardiovascular disease
HD: heart disease
HF: heart failure
MI: myocardial infarction
†: Best model
§: Average for two datasets

Appendix 2

No.	Equation	Explanation
1	$h_\theta(z) = \frac{1}{1+e^{-z}}$, $z = \beta_0 + \beta_1 x_1 + ... + \beta_m x_m$	βi ($i=0,1,...,m$) is the regression coefficient, and xi ($i=1,2,...,m$) is the predictor variable
2	$Net_j = \sum W_{ij} X_{ij}$	X_{ij} represents the ith input to node j, and W_{ij} represents the weight associated with the ith input to node j
3	$f(net_j) = 1/(1 + e^{-X})$	X represents the summed value of Net_j obtained from previous equation
4	$W_{new}=W_{current} -\alpha(\partial SSE/\partial W_{current})$	W refers to weight, α refers to learning rate, and $\partial SSE/\partial W_{current}$ is the partial derivative of SSE to current weights
5	$p(Y \mid X) = p(X \mid Y)p(Y) / p(X)$	$p(X\|Y)$ represents the likelihood function, $p(Y)$ the prior probability, and $p(X)$ marginal probability of the data
6	$entropy = -\sum_{i=1}^{n} p_i \log_2(p_i)$ $info = \sum_{i=1}^{n} \frac{\|S_i\|}{\|S\|} \times entropy(S_i)$ $IG = info_{beforesplit} - info_{aftersplit}$ $IntrinsicInfo = -\sum \frac{\|S_i\|}{\|S\|} \log_2 \frac{\|S_i\|}{\|S\|}$ $IGRatio = IG \times IntrinsicInfo$	P_i denotes the proportion of instances in each branch to the total instances after splitting by a variable, n denotes the number of corresponding branch, and S and S_i represent the number of instances in a branch before and after splitting
7	$R(i \mid x) = \sum_i P(j \mid x)C(i \mid j)$	$P(j\|x)$ represents the probability of each class j, and $C(i,j)$ denotes the cost matrix

Appendix 3

Variable	Character[6]	Description
SEQNO	2	Sequence number
INCLINIC	2	Clinic attended or not
INBLOOD	2	Blood specimen taken or not
PROV	2	Province
AREA	1	Area designation based on 10,000+ population
SEX	1	Gender
AGE	1	Age in years
GPAGE	3	Age group
GPAGE2	1	Age grouped in 10 years
EDUCYRS	4	Years of education
EDUC	1	Grouped education
INCOME	4	Income level
INCADEQ	1	Income adequacy
EMPLOY	1	Employment status
WORKTYPE	1	Work classification
MARITAL	1	Current marital status
LANG	1	Language
HOUSEHLD	4	Number of persons in household
PWGTQ	2	Probability weight for questionnaire
PWGTC	2	Probability weight for clinic/blood specimen
BPS1	4	First systolic blood pressure reading
BPD1	4	First diastolic blood pressure reading
BPS2	4	Second systolic blood pressure reading
BPD2	4	Second diastolic blood pressure reading
BPCS1	4	First systolic blood pressure reading in clinic
BPCD1	4	First diastolic blood pressure reading in clinic
BPCS2	4	Second systolic blood pressure reading in clinic
BPCS2	4	Second diastolic blood pressure reading in clinic
MSYS	1	Average systolic blood pressure value
SYSCAT	1	Systolic blood pressure categories
MDIAS	1	Average diastolic blood pressure value
DIASCAT	1	Diastolic blood pressure categories
HYPER	4	Hypertensive status
H14090	5	Hypertensive status based on 140/90 standard
HATCS	6	High blood pressure awareness/treatment/control status
HATCSDS	6	High blood pressure awareness/treatment/control based on 140/90 standard
PHARM	1	On pharmacological treatment for high blood pressure?
NPHARM	1	On non-pharmacological treatment for high blood pressure?

[6] 1 = "retained variable", 2 = "unrelated variables to CVD", 3 = "variables recorded in broader value ranges", 4 = "overlapping variables", 5 = "merged variables", 6 = "variables with too few respondents", 7 = "textual variables"

BPPREV	5	Ever had blood pressure checked?
BPWHEN	1	Time since last blood pressure checked?
BPWHO	1	Who checked your blood pressure on last time?
BPINFO	1	How was blood pressure described?
BPPREVS	6	Last systolic blood pressure reading
BPPREVD	6	Last diastolic blood pressure reading
BPHIGH	5	Ever told you had high blood pressure?
BPHIGHTR	1	Any treatment prescribed for high blood pressure?
BPRX1	6	BP treatment prescribed: medication?
BPRXAND1	6	BP treatment prescribed: medicine and other?
BPSALT1	6	BP treatment prescribed: salt free diet?
BPWGT1	6	BP treatment prescribed: watch weight?
BPSTR1	6	BP treatment prescribed: avoid stress?
BPSMOK1	6	BP treatment prescribed: reduce/stop smoking?
BPALC1	6	BP treatment prescribed: reduce alcohol?
BPEXER1	6	BP treatment prescribed: start exercise?
BPBIO1	6	BP treatment prescribed: use biofeedback?
BPOTH1	6	BP treatment prescribed: other treatment?
BPDK1	6	BP treatment prescribed: don't know/not stated
BPPRG2	6	Same/different program for high blood pressure now?
BPRX2	6	BP treatment prescribed: medication?
BPRXAND2	6	BP treatment prescribed: medicine and other?
BPSALT2	6	BP treatment prescribed: salt free diet?
BPWGT2	6	BP treatment prescribed: watch weight?
BPSTR2	6	BP treatment prescribed: avoid stress?
BPSMOK2	6	BP treatment prescribed: reduce/stop smoking?
BPALC2	6	BP treatment prescribed: reduce alcohol?
BPEXER2	6	BP treatment prescribed: start exercise?
BPBIO2	6	BP treatment prescribed: use biofeedback?
BPOTH2	6	BP treatment prescribed: other treatment?
BPDK2	6	BP treatment prescribed: don't know/not stated
BPRXNOW	6	Now taking medication for high blood pressure?
BPRXEVER	6	Ever taken medication for high blood pressure?
BPOKNOW	1	Is your blood pressure normal now?
BPHEALTH	1	Do you think high blood pressure can affect health?
BPHLTHOW	7	How do you think high blood pressure can affect health?
BPCAUSE	7	What things can cause high blood pressure?
BPFOOD	5	Heard high blood pressure related to food/drinks?
BPSALT	1	Heard high blood pressure related to: salt?
BPSODIUM	1	Heard high blood pressure related to: sodium?
BPALC	1	Heard high blood pressure related to: alcohol?
BPFATS	1	Heard high blood pressure related to: fats?
BPSATFAT	1	Heard high blood pressure related to: saturated fats?
BPCHOL	1	Heard high blood pressure related to: cholesterol?
BPCALOR	1	Heard high blood pressure related to: calories/overeating?

BPADDIT	1	Heard high blood pressure related to: additives/preservatives?
BPCAFF	1	Heard high blood pressure related to: caffeine/coffee?
BPSUGAR	1	Heard high blood pressure related to: sugar/sweet foods?
BPSTARCH	1	Heard high blood pressure related to: starch/starchy foods?
BPPORK	1	Heard high blood pressure related to: pork?
BPOTHMTS	1	Heard high blood pressure related to: specific other meat?
BPMEATS	1	Heard high blood pressure related to: meats generally?
BPFRIED	1	Heard high blood pressure related to: fried/greasy/fast foods?
BPCALC	1	Heard high blood pressure related to: calcium?
BPOTHER	1	Heard high blood pressure related to: other foods/drinks?
BPDK	1	Heard high blood pressure related to: don't know
HGTC	4	Clinic: height in centimeters
WGTC	4	Clinic: weight in kilograms
BMI	1	Body mass index
BMICAT	1	BMI categories
BMI25	3	BMI, cut point of 25
BMI27	3	BMI, cut point of 27
WAIST	4	Clinic: waist in centimeters
HIP	4	Clinic: hip in centimeters
WHR	1	Waist/hip ratio
WHRMEN	4	Waist/hip ratio for males
WHRWOMEN	4	Waist/hip ratio for females
WHRCAT	1	Waist/hip ratio risk categories
PULSE	1	Clinic: pulse measurement
HRS	2	Clinic: hours since last meal
FAST	2	Fasting blood sample >= 8 hours?
HGT	4	Height in centimeters
WGT	4	Weight in kilograms
TCHOL	1	Total plasma cholesterol (mmol/l)
TCHOLCAT	1	TCHOL categories
TCHLFCAT	4	TCHOL categories, fasting subjects only
TCHOL52A	3	TCHOL categories, cut point=5.2, all subjects
TCHOL62A	3	TCHOL categories, cub point=6.2, all subjects
TCHOL52F	3	TCHOL categories, cut point=5.2, fasting subjects only
TCHOL62F	3	TCHOL categories, cut point=6.2, fasting subjects only
HDL	1	High density lipoproteins (mmol/l)
HDLCAT	1	HDL categories
HDL09F	3	HDL categories, cut point=0.9, fasting subjects only
LDL	1	Low density lipoproteins (mmol/l)
LDLCAT	1	LDL categories
LDL34F	3	LDL categories, cut point=3.4, fasting subjects only

LDL41F	3	LDL categories, cut point=4.1, fasting subjects only
TRIG	1	Triglycerides (mmol/l)
TRIGCAT	1	TRIG categories
TRIG23F	3	TRIG categories, cut point=2.3, fasting subjects only
MRF3A	4	Sum risk factors, SMOK+CHOL5.2+HYPER
MRF3B	4	Sum risk factors, SMOK+CHOL5.2+H14090
MRF4A	4	Sum risk factors, SMOK+CHOL5.2+HYPER+SEDENT
MRF4B	4	Sum risk factors, SMOK+CHOL5.2+H14090+SEDENT
DIABCAT	5	Diabetes status
DIABET	4	Ever told you have diabetes?
DIABAGE	6	Age when told you had diabetes
DBNOTRT	1	Diabetes treatment now: no current treatment
DBINS	1	Diabetes treatment now: insulin
DBRX	1	Diabetes treatment now: pills
DBDIET	1	Diabetes treatment now: diet
DBWGT	1	Diabetes treatment now: weight loss
DBOTHER	1	Diabetes treatment now: other treatment
DBDK	1	Diabetes treatment now: don’t know/not stated
ALCOHOL	1	Alcohol drinking status
ALCEVER	4	Ever taken an alcoholic drink?
ALC12MTH	4	Taken an alcoholic drink in past 12 months?
ALCMTH	6	Monthly frequency of alcoholic drinks in past year
ALCNUM	6	Average daily alcohol consumption
REGSMOK	4	Regular smoker?
SMOKECAT	1	Current smoking status
CIGCAT	1	Cigarettes smoked categories
CIG_DAY	1	Number cigarettes, cigars or pipes.
SMOK	4	Ever smoked cigarettes, cigars or pipes?
PIPE	4	Do you smoke a pipe now?
PIPENOW	6	Smoke pipe regularly or occasionally?
CIGAR	4	Do you smoke cigars now?
CIGARNOW	6	Smoke cigars regularly or occasionally?
CIGET	4	Do you smoke cigarettes now?
CIGETNOW	6	Smoke cigarettes regularly or occasionally?
CIGETNUM	4	Number of cigarettes usually smoked per day?
WGTLOSE	1	Ever tried to lose weight??
WGTNOW	5	Presently trying to change weight?
WGTDIET	1	To lose weight: dieting
WGTEXER	1	To lose weight: exercising
WGTMEALS	1	To lose weight: skipping meals
WGTPILLS	1	To lose weight: taking diet pills
WGTPRGM	1	To lose weight: attending weight control programs
WGTOTHER	1	To lose weight: something else
WGTDK1	1	To lose weight: don’t know/not stated
WGTLOOKS	1	Why lose weight: to be more attractive

WGTHLTH	1	Why lose weight: to improve general health
WGTHEART	1	Why lose weight: decreasing risk of heart attack
WGTBP	1	Why lose weight: maintain acceptable blood pressure
WGTCHOL	1	Why lose weight: have good cholesterol level
WGTART	1	Why lose weight: hardening of arteries
WGTDIAB	1	Why lose weight: decreasing risk of diabetes
WGTOTH	1	Why lose weight: other reasons
WGTDK2	1	Why lose weight: don't know/not stated
WGTLIKE	6	How much would you like to weight?
SALTCOOK	1	How often is salt added while cooking?
SALTFOOD	1	How often is salt added at the table?
SALTHLTH	5	Can amount on salt eaten affect health?
SALTBP	1	Salt effects: blood pressure would increase
SALTWGT	1	Salt effects: weight would increase
SALTEDEM	1	Salt effects: ankles may become swollen
SALTHA	1	Salt effects: increase risk of heart attack
SALTSTR	1	Salt effects: increase risk of stroke
SALTKIDN	1	Salt effects: increase risk of kidney problems
SALTRX	1	Salt effects: take blood pressure pills/medication
SALTART	1	Salt effects: increase hardening of arteries
SALTOTH	1	Salt effects: other problems
SALTDK	1	Salt effects: don't know/not stated
FATS	5	Aware health problems related to eating fat?
FATOBS	1	Fat health problems: overweighting/obesity
FATHEART	1	Fat health problems: heart disease/attack
FATCHOL	1	Fat health problems: high blood cholesterol
FATBP	1	Fat health problems: high blood pressure
FATART	1	Fat health problems: arteriosclerosis
FATDK	1	Fat health problems: don't know/not stated
FATOTH	1	Fat health problems: other problems
CHOL	5	Have you heard about cholesterol?
CHOLINFO	7	What have heard about cholesterol?
CHOLFOOD	5	Do you think cholesterol is in foods?
CHOLEGGS	1	Contains cholesterol: eggs/egg yolk
CHOLPOUL	1	Contains cholesterol: poultry
CHOLBEEF	1	Contains cholesterol: beef
CHOLPORK	1	Contains cholesterol: pork
CHOLSEA	1	Contains cholesterol: seafood
CHOLMILK	1	Contains cholesterol: milk
CHOLCHSE	1	Contains cholesterol: cheese
CHOLOTH	1	Contains cholesterol: other food
CHOLBUT	1	Contains cholesterol: butter
CHOLFAST	1	Contains cholesterol: fast food
CHOLDK	1	Contains cholesterol: don't know
CHOLHLTH	6	Can cholesterol in foods affect health?
CHOLINBL	1	Do you think cholesterol is found in blood?
CHOLBLD	6	Too much cholesterol in blood affects health?

CHOLART	1	Cholesterol affects health: hardening arteries?
CHOLBP	1	Cholesterol affects health: increase blood pressure
CHOLHA	1	Cholesterol affects health: heart attack
CHOLSTR	1	Cholesterol affects health: stroke
CHOLANG	1	Cholesterol affects health: angina/chest pain
CHOLOTHR	1	Cholesterol affects health: other problems
CHOLDK2	1	Cholesterol affects health: don't know/not stated
BCHOL	1	Ever had blood cholesterol measured?
BCHTOLD	6	Told what blood cholesterol level was?
BCHHIGH	1	Ever told your blood cholesterol was high?
BCHTRT	6	Prescribed treatment to lower blood cholesterol?
BCHDIET	6	Presently on diet to lower blood cholesterol?
BCHEXER	1	To lower cholesterol: exercise
BCHSTRES	1	To lower cholesterol: control stress/fatigue
BCHRX	1	To lower cholesterol: prescribed medications
BCHCHOL	1	To lower cholesterol: eat food with less cholesterol
BCHFAT	1	To lower cholesterol: eat less fatty foods
BCHWGT	1	To lower cholesterol: lose weight
BCHSKIM	1	To lower cholesterol: low fat dairy produce
BCHNONE	1	To lower cholesterol: nothing
BCHOTH	1	To lower cholesterol: other methods
BCHDK	1	To lower cholesterol: don't know/not stated
SEDENT	5	Sedentary?
EXER	4	Exercise regularly?
EXERSTRN	1	How much of this exercise is strenuous?
EXERLONG	1	How long do you usually exercise?
EXERWORK	6	Work requires strenuous physical activity?
HDDIET	1	Causes of heart disease: poor diet
HDWGT	1	Causes of heart disease: overweight
HDFATS	1	Causes of heart disease: excess fats
HDSALT	1	Causes of heart disease: excess salt
HDCHOLBD	1	Causes of heart disease: high blood cholesterol
HDCHOLFD	1	Causes of heart disease: foods with high cholesterol
HDSTRESS	1	Causes of heart disease: stress/worry
HDTIRED	1	Causes of heart disease: overwork/fatigue
HDEXER	1	Causes of heart disease: lack of exercise
HDSMOKE	1	Causes of heart disease: smoking
HDHERED	1	Causes of heart disease: heredity
HDHBP	1	Causes of heart disease: high blood pressure
HDART	1	Causes of heart disease: arteriosclerosis
HDDK	1	Causes of heart disease: don't know
HDOTH	1	Causes of heart disease: other causes
HDPREV	1	Believe heart disease can be prevented
EVERHA	5	Have you ever had a heart attack?
EVERSTR	5	Have you ever had a stroke?
STRPREV	1	Believe strokes can be prevented
OTHHD	5	Do you have any other heart disease?

RXHEART	2	Taking any prescribed medications for heart?
FEMRX	1	On oral contraceptives/hormonal pills?

Glossary

AdaBoost M1:

- A Boosting algorithm for ensemble modeling

Bagging:

- An ensemble modeling technique

Gradient Descent Method:

- An iterative process takes steps proportional to the negative of the gradient of the function in order to find a local minimum

Interquartile-Range filter:

- A filter for detecting outliers and extreme values based on interquartile ranges in WEKA.

MetaCost:

- A cost-sensitive learning algorithm achieved through the alteration of cost matrix

Min-Max Normalization:

- A process of taking data measured in its original scale and transforming it to a value between 0 and 1