Generalization of Nonlinear Integrals and Its Applications

WANG, Jinfeng

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Doctor of Philosophy
in
Computer Science and Engineering

The Chinese University of Hong Kong
February 2010

Thesis Assessment Committee

Professor WONG Man Hon (Chair)

Professor LEUNG Kwong Sak (Thesis Supervisor)

Professor LEE Kin Hong (Thesis Supervisor)

Professor LEUNG Ho Fung (Committee Member)

Professor KWONG Tak Wu (External Examiner)

Abstract of thesis entitled:

Generalization of Nonlinear Integrals and Its Applications

Submitted by WANG, Jinfeng

for the degree of Doctor of Philosophy

at The Chinese University of Hong Kong in December 2009

Nonlinear Integral (NI) is a useful integration tool. It has been applied to many areas including classification and regression. The classical method relies on a large number of training data, which lead to large time and space complexity. Moreover, the classical Nonlinear Integral has many limitations. For dealing with different situation, we propose Double Nonlinear Integrals and Nonlinear Integrals with Polynomial Kernel to deal with the problems transversely and longitudinally.

When the data to be classified have special distribution in the data space, the projection may overlap and the classification accuracy will be lowered. For example, when one group of the data is surrounded by the data of another group, or the number of classes for the data is large. To handle this kind of problems, we propose a new classification model based on the Double Nonlinear Integrals. Double Nonlinear Integral means projecting to a 2-Dimensional space by using the Nonlinear Integral twice in succession and classifying the virtual values in the 2-D space corresponding to the original data. Double Nonlinear Integrals can lessen loss of information due to the intersection of different classes on real axis. Accuracy will also be increased accordingly.

The classical Nonlinear Integrals implement projection along a line with respect to the features. But in many cases the linear projection cannot achieve good performance for classification or regression due to the limitation of the integrand. The linear function used for the integrand is just a special type of polynomial functions with respect to the features. We propose Nonlinear Integral with Polynomial Kernel (NIPK) in which a polynomial function is used as the integrand of Nonlinear Integral. It enables the projection to be along different types of curves on the virtual space, so that the virtual values gotten by the Nonlinear Integrals with Polynomial Kernel can be better regularized and easier to deal with. Experiments show that there is evident improvement of performance for NIPK compared to classical NI.

Another extension of Nonlinear Integral, Upper and Lower Nonlinear Integrals, which is a pair of extreme nonlinear integrals to contain all types of Nonlinear Integrals in the same scheme, is also proposed. It can give a set of upper and lower bounds which include all types of Nonlinear Integrals. We tried to find a solution with the smallest distance between the upper and lower bounds and the smallest error which is a NP hard problem. So we use the multi-objective optimization method to find a set of results for the regression model based on the Upper and Lower Nonlinear Integrals. We can just select one or more optimal solution(s) for a specific problem from the set of results. A weather predictor based on this model has been constructed to predict the next days temperature changing trend and range.

Finally, a NI based data mining framework has been established for identifying the chance of developing liver cancer based on the Hepatitis B Virus DNA sequence data. We have shown that the framework obtains the best diagnosing performance amongst many existing classifiers.

論文題目： 非线性积分扩展及其应用

作者 ： 王金鳳

學校 ： 香港中文大學

學系 ： 計算機科學與工程學系

修讀學位： 哲學博士

摘要 ：

非線性積分是一種很有用的求和工具，目前已經被應用到了分類和回歸等很多領域。它可以將原始數據投影到一個虛擬空間，然後用簡單的線性分類器來分類虛擬值，所對應的原始數據也隨之分類。本文中，我們在傳統非線性積分的算法基礎上分別從橫向和縱向上提出了兩種擴展方法---雙重非線性積分（DNI）和多項式非線性積分（NIPK）。

現實數據中存在投影重疊的情況，例如一類數據被另一類數據包圍，用傳統的非綫性幾分進行投影會影響分類準確度。因此，我們創建一種基於雙重非綫性積分的分類器。雙重非線性積分即是用傳統非線性積分連續投影兩次映射到二維空間，然後再將二位空間的數據進行分類。雙重非綫性積分能夠減少由於類間交疊所引起的信息丢失，從而提高準確率。

經典的非綫性積分是沿直綫進行投影，但在一些實際情況中，數據分佈是較爲複雜的，單一的直綫很難覆蓋同一類卻不在同一直綫上的數據，這樣就會影響分類或者回歸的性能。本文我們提出了指數非綫性積分，就是用一個指數函數來代替傳統的一次綫性函數作爲非綫性積分的積分函數，即非綫性積分的核。指數非綫性積分可以將原始數據沿著不同的曲綫投影到映射空間，使得所得到的投影值能夠更好的描述原始數據。

我們還提出了另一種非線性積分的擴展形式，即上下限非線性積分，這是一對能夠包含各種非線性積分的極限非線性積分，它能夠給出一對非線性積分值的上下界。我們試圖找到一個較好的解使得上下界的距離和界外點距離最近邊界的距離達到最小，這是一個 NP-hard 問題，因此我們使用多目標優化方法爲基于上下限積分的回歸模型找到一組優化解。基於此模型，我們建立了一個天氣預測器，它能夠提供給我們第二天氣溫的變化趨勢和範圍。

最後，我們建立一個基於非綫性積分的數據挖掘框架用來進行肝癌患病率的預測, 通過和傳統算法比較，證實我們的數據挖掘框架具有最好的診斷性能。

Acknowledgements

First of all, I would like to thank my dissertation supervisors, Prof. Kwong Sak LEUNG and Prof. Kin Hong LEE, for their guidance, encouragement and inspiration on my research from the beginning of our professional and educational relationship. They helped me find my way round research and have always been available for help and advice.

I would like to give my special thanks to Prof. Zhenyuan Wang at the University of Nebraska at Omaha who have given me lots of advices and guidance for establishing the theoretic basis of the thesis.

I should express my appreciation to Prof. Ho Fung LEUNG and Prof. Man Hon WONG at our department, and Prof. Sam Tak Wu KWONG at the Hong Kong City University. They serve in my thesis committee and have been giving me valuable suggestions on my research for many years.

I am also in deep gratitude to all of my colleagues and friends. They have given me enjoyable years of study by their pure-hearted encouragements and helps. I have enjoyed working with Yong Liang, Wing Ho Shum, Wenye Li, Gang Li, Bing Ni, Tak-Ming Chan, David, Peter, Ricky, and all the other team members that joined our group during the past four years.

Finally, I owe a special thanks to my husband, WenZhong Wang, for his tolerance and support during the last four years. His love, patience, and understanding have always been an invaluable source of tranquility and motivation. I am indebted to my parents for they have given me a family full of love, care, and support. I would like to thank my parents-in-law for supporting in the most difficult period of ours.

To My Parents and Husband

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Knowledge integration and decision making by humans are often done in environments where information are interdependent or interactive. Artificial intelligence researchers have been attempting to emulate this capability in computer systems to handle information fusion which assumes the input variables are indepedent. Nevertheless, the interaction among the information is ubiquitous in practical databases. This thesis is mainly devoted to a comprehensive investigation on the innovative data mining methodologies which merge the advantages of the Fuzzy Measure and Nonlinear Integral in representation of nonlinear relationship to deal with the interaction among features with respective to contribution.

This chapter gives a brief research background and introduction of this thesis. The model building, linearity, and nonlinearity in data mining are first introduced in Sections 1.1 and 1.2, respectively. The state-of-the-art of data mining models are reviewed and compared in Section 1.3. In Sections 1.4 and 1.5 the motivation and the problem definition respectively. The contribution of this thesis is presented in Section 1.6. Finally, the organization of this thesis is given in the last section.

1.1 Model Building in Data Mining

Data mining is an interdisciplinary field with a general goal of predicting outcomes and eliciting relationships in data. The main goal of data mining is to extract knowledge from data. This knowledge is usually represented by means of a particular data model that is extracted from the database. There are diverse tools employing sophisticated algorithms to discover hidden patterns, associations and structures from large amount of data stored in data bases or other information sources. Huge amount of data have been collected and stored in databases of financial investment, medical care, industry manufacturing, telecommunication, scientific research, and last but not least. the World Wide Web. Explosive growths and diverse forms of stored data would be useless until we have new techniques and tools that can intelligently and automatically assist us in eliciting valuable knowledge from raw data. Regression Analysis, Rule based Classifiers, and Neural Networks are some of the well known models. Information fusion is useful in the process of model building.

Data mining uses information fusion techniques for improving the quality of the extracted knowledge. The three distinct uses of information fusion are[1]:

Information fusion in preprocessing: Fusion is used to increase the quality of raw data prior to the application of data mining methods.

Information fusion for building models: The model built from data uses some kind of information fusion technique (e.g. a particular aggregation operator to fuse partial results).

Information fusion is used to extract information: The knowledge extracted from data is the result of a particular information fusion technique.

E.g., an aggregated value computed from the data.

Information fusion techniques can be used in two modes [1]. One way is to define the model using fusion techniques. A particular aggregation operator combines a set of inputs to obtain an output. An aggregation operator takes data from information sources and computes an aggregated value. If we denote the data by the information source x_i in $a_i = f(x_i)$, then the aggregated value can be represented by $\mathcal{F}(a_1, a_2, ..., a_n)$, where $\mathcal{F}$ expresses the integration operator. Given a set of l examples, each example for $i \in \{1, 2, ..., l\}$ consists on the values for variables $(a_1^i, a_2^i, ..., a_n^i)$ and the outcome y^i. A model of this data is build using an aggregation operator F. Building a model need finding the operator $\mathcal{F}$ and the parameters so that $\mathcal{F}(a_1^i, a_2^i, ..., a_n^i)$ is more similar to y^i for all the examples. The other way using fusion techniques is to combine several data models. This is described in [2, 4, 6]. The operators considered in combination modules include voting [5, 6] and the weighted mean [6]. Bagging [7] and Boosting [8] are well-known examples of machine learning algorithms for learning such complex data models.

1.2 Linearity and Nonlinearity

Most data mining problems are based on aggregation models to describe how determinant features influence determination. The traditional aggregation tool is linear as $y = a_1x_1 + a_2x_2 + ... + a_nx_n$, where y is the aggregation value for determination, x_i is the value of predictive feature, a_i is the unknown coefficients for $i = 1, 2, ..., n$, and n is the number of predictive features. The basic assumption in such a linear model is that there exists no interaction among predictive features to determination. This means that influence to decision from predictive features are independent such that the global contribution of the set of all predictive features to decision is just a simple sum of their respective contributions. However, a question is how to get the aggregation when predictive features are dependent.

Actually in many real-world problems, the interaction among predictive features to decision cannot be ignored. For example, consider two workers, A and B. A can produce 20 products per day individually; while B's daily yield is 30. If they work independently, the total product will be $20 + 30 = 50$ products. When they work cooperately, we need to consider the relationship between A and B. There may be two possibilities. One is that they yield more than 50 products because they can cooperate harmoniously so that their joint working efficiency is increased; the other is the contrasting situation which produces less than 50 products because their discordant relationship reduces their cooperative efficiency.

The ubiquitous nonlinearities in databases have been investigated explicitly or implicitly by many existing data mining approaches. They have been represented as the productive rules in rule-based systems, the activation functions and the weights in Neural Networks, the causality in Bayesian networks, the transitions firing in Petri nets, etc.

1.3 State-of-the-Art Methods

The requirements on fuzzy information description and nonlinear relationship representation have stimulated researchers to work on the representation and manipulation of fuzzy knowledge. Their efforts have been incorporated by many existing models. In this section, we will have a brief review on some representatives with greater interests. They are Decision Tree, Neural Network, Support Vector Machine and Naïve Bayes. The following paragraphs are the brief descriptions of the four classical methods which have been used to compare with our new methods.

1.3.1 Decision Tree [9]

A decision tree is a tree-structured classifier. Decision Tree method learns a decision tree using a recursive tree growing process. Each test corresponding to an feature is evaluated on the training data using a test criteria function. The test criteria function assigns each test a score based on how well it partitions the dataset. The test with the highest score is selected and placed at the root of the tree. The subtrees of each node are then grown recursively by applying the same algorithm to the examples in each leaf. The algorithm terminates when the current node contains either all positive or all negative examples. For our experiments, we use the widely available package-See5.0, which is the state-of-the-art of the Decision Tree classifier.

1.3.2 Neural Network [10]

An Artificial Neural Network (ANN), or commonly just called Neural Network (NN) is an interconnected group of artificial neurons that uses a mathematical or computational model for information processing based on a connectionistic approach to computation. In most cases an NN is an adaptive system that changes its structure or weights of the interconnections based on external and internal information that flows through the network. In more practical terms, NNs are nonlinear statistical data modeling for decision making and classification tools. They can be used to model complex relationships between inputs and outputs or to find patterns in data. However, it is essentially a black box approach and it is not easy to interpret how NNs function.

1.3.3 Support Vector Machine[11]

Support Vector Machines (SVMs) are a set of related supervised learning methods used for classification and regression. Viewing input data as two sets of vectors in an n-dimensional space, an SVM will construct a separating hyperplane in that space, one which maximizes the margin between the two data sets. To calculate the margin, two parallel hyperplanes are constructed. one on each side of the separating hyperplane, which are "pushed up against" the two data sets. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the neighboring data points of both classes, since in general the larger the margin the smaller the generalization error of the classifier.

The original optimal hyperplane algorithm proposed by Vladimir VaNIPKk in 1963 was a linear classifier. The classification model produced by SVM (as described above) only depends on a subset of the training data. because the cost function for building the model does not care about training points that lie beyond the margin. In our thesis, the software used is obtained from[11].

1.3.4 Naïve Bayes[12]

A Naïve Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong (naïve) independence assumptions. A more descriptive term for the underlying probability model would be "independent feature model".

Depending on the precise nature of the probability model, Naïve Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naïve Bayes models uses the method of maximum likelihood; in other words, one can work with the Naïve Bayes model without believing in Bayesian probability or using any Bayesian methods.

In spite of their over-simplified assumptions, Naïve Bayes classifiers often work much better in many complex real-world situations than one might expect. Recently, careful analysis of the Bayesian classification problem has shown that there are some theoretical reasons for the apparently unreasonable efficiency of Naïve Bayes classifiers[13]. An advantage of the Naïve Bayes classifier is that it requires a small amount of training data to estimate the parameters (means and variances of the variables) necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined and not the entire covariance matrix.

1.3.5 Comparisons and Discussions

In above subsections, we have reviewed four connectionist model-based methodologies which are able to deal with uncertain knowledge in different degrees. All of them have their respective advantages as well as drawbacks. For handling data mining problems, they may handle certain issues will but fail in other ones.

Decision Tree is one commonly used nonparametric tool for pattern classification. It is readily interpretable and easily understandable. All these features have been passed to the fuzzy decision tree, which can process fuzzy knowledge, such as linguistic terms. It is well suited to solve classification problems where fuzzy data are involved. However, it cannot provide any information about the intersection regions where the pattern classes are overlapped. Furthermore, it is very sensitive to training data and not efficient to deal with non symbolic data.

Neural Network seems to be the most popular methodology in data mining domain. They have been broadly applied to many data mining problems, such as classification, regression, clustering, and so on. Many advantages of Neural Network have been extended to fuzzy neural approaches so that they possess powerful learning ability and high flexibility to deal with different degrees of uncertainty. These uncertainties, represented as certainty degree values or linguistic variables, may appear at input nodes, weights, or output nodes. The main weakness of NNs is lack of abilities of knowledge interpretation due to their black-box nature.

The naïve Bayes classifier is an efficient classification model that is easy to learn and has a high accuracy in many domains. However, it has two main drawbacks: (i) its classification accuracy decreases when the features are not independent, and (ii) it can not deal with nonparametric continuous features. A optimal naïve Bayes classifier [14] was proposed. This method includes two phases, discretization and structural improvement, which are repeated alternately until the classification accuracy can not be improved. Discretization is based on the minimum description length principle.

Potential drawback of the SVM is that it is only directly applicable for two-class tasks. Therefore, algorithms that reduce the multi-class task to several binary problems have to be applied. So multi-class SVM was proposed to assign labels to instances by using support vector machines, where the labels are drawn from a finite set of several elements. The dominating approach for doing so is to reduce the single multiclass problem into multiple binary problems. Each of the problems yields a binary classifier, which is assumed to produce an output function that gives relatively large values for examples from the positive class and relatively small values for examples belonging to the negative class.

1.4 Motivation

From the discussion, we see that although various methodologies which can manage different situations, all currently existing models are not able to represent the interaction and relationship among features explicitly and efficiently.

Ideally, a comprehensive method should be able to:

- sufficiently represent and directly handle interactive information better;
- have more powerful self-learning ability; and
- describe the internal nonlinear relationship of knowledge.

These are also the motivations of this thesis. We have developed new algorithms based on the Nonlinear Integral with respect to a Fuzzy Measure to achieve these goals. Nonlinear Integral has been proved to be a powerful aggregation tool to handle data with interactive information in classification, regression, and decision making problems. In these applications, the Nonlinear Integral reveals a combination of many advantages which cannot be all possessed by any one of the existing methodologies. These advantages are robust self-learning ability, powerful nonlinear representation, and explicit description of interaction among features.

However, the classical Nonlinear Integrals only support linear integrands to be appropriate for the linear distributed data, and are helpless when facing the overlapping situation after projection. To extend the advantages of the Nonlinear Integral to deal with more complicated distributed data, we have extended the classical Nonlinear Integrals in different way in order to solve specific problems. These extended models are generalizations of the classical Nonlinear Integrals. The detailed descriptions will be given in the following chapters respectively.

In short, the generalizations of the Nonlinear Integral presented in this thesis are promising tools for data fusing problems where complex data are involved.

1.5 Problem Definition

Generalizations of the Nonlinear Integral in this thesis include three scopes. the longitudinal extension—Double Nonlinear Integrals (DNI), the transverse extension—Nonlinear Integrals with Polynomial Kernel (NIPK) and the overall extension—Upper and Lower Nonlinear Integrals (ULNI). All of them are able to deal with complex data better than the original one. Briefly, the DNI, the NIPK and the ULNI are generalizations of the original Nonlinear Integral. This statement has been proved by the theoretic derivations of them in this thesis. Relationship among the Classical Nonlinear Integral (CNI), the DNI, the NIPK and the ULNI is demonstrated in Figure 1.1.

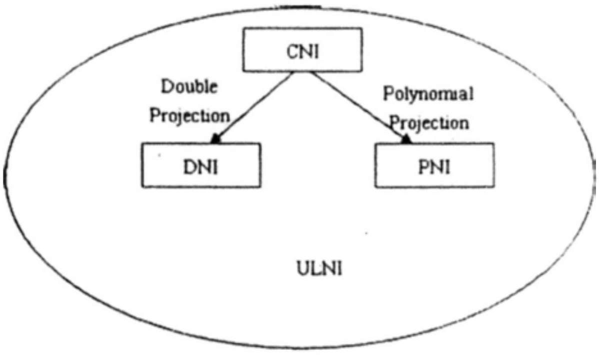


Figure 1.1: The relationship of the Generalized Nonlinear Integral and Classical Nonlinear Integral

Facing with the same data fusing problem, performances of the Classical Nonlinear Integral, the DNI, the NIPK and the ULNI, are different. Due to the different forms of integration results, the DNI, the NIPK and the ULNI have their distinct theoretic analyses, implementation algorithms, and application scopes, which have been investigated in this thesis respectively.

General conditions under which the DNI, the NIPK, the ULNI and the Classical Nonlinear Integral are employed are stated as follows: For a data fusing problem where a set of n predictive features (denoted by $x_1, x_2, ..., x_n$) are used to determine a set of m targets (denoted by $y_1, y_2, ..., y_m$) the value of x_i is denoted by $f(x_i), i = 1, 2, ..., n$, and the value of y_j is denoted by $g(y_j). j = 1, 2, ..., m$.

1.6 Contributions of this thesis

Three generalizations of Nonlinear Integral have been comprehensively investigated in this thesis. These generalizations are the Doubled Nonlinear Integral (DNI), the Nonlinear Integral with Polynomial Kernel(NIPK) and the Upper and Lower Nonlinear Integral (ULNI). All of them have some extensions, but the first one gives a double process of the classical Nonlinear Integral; the second one uses a polynomial integrand; and the last one gives out an interval value of integral. They are respectively designed to be applicable to different requirements of mining on heterogeneous data. Keeping the nonadditivity property of the Signed Fuzzy Measure in their definitions, the DNI, the NIPK and ULNI are able to elicit the interactive relationship among features. Due to these reasons, compared with other existing approaches, they possess more powerful ability on knowledge interpretation and flexibility when applied to different data mining applications or to show the characteristic. The main contributions of this thesis are listed as follows:

1. The classical methods transform the original classification problem from n-dimensional space to a one-dimensional space problem through the optimal projection based on Nonlinear Integrals. But plenty of information

may be missed in the projecting process. In some special cases, there exist projection overlapping when the data to be classified have special distributions in the data space, for example one group of the data is surrounded by the data of another group, or the number of classes for the data is large. This will lead to lower classification accuracy. So we propose a new classification model based on the Double Nonlinear Integrals to solve this problem. The Double Nonlinear Integrals can lessen loss of information coming from the intersection of different classes on real axis in classical one. Classification accuracy has been shown to increase accordingly.

2. We introduce the polynomial function as nonlinear integrand to fix the limitation of the classical Nonlinear Integrals. The Nonlinear Integrals with Polynomial Kernel with respect to polynomial integrand was proposed. This revolution can extend the projection from linear line to more shapes of curves which can cover more complicated data. The accuracy of classification model is not neccessarily increased with the degree of the polynomial. So we need to learn the optimal polynomial index for the Nonlinear Integrals with Polynomial Kernel. Nevertheless, we have shown the complexity of the Nonlinear Integrals with Polynomial Kernel is not greater than the classical Nonlinear Integrals.

3. A new nonlinear multi-regression model based on a pair of extreme Nonlinear Integrals, Upper and Lower Nonlinear Integrals, is established in this thesis. A data set of predictive features and the relevant objective feature is required for estimating the regression coefficients. Due to the nonadditivity of the model, a Genetic Algorithm or other soft computing technique should be adopted to search the optimized solution in the

regression problem. Applying such a nonlinear multi-regression model, an interval prediction for the value of the objective feature can be made once a new observation of predictive features is available.

4. A NI based data mining framework which includes molecular evolution analysis, clustering, feature selection, classifier learning and classification is introduced. This framework has been applied successfully to a Hepatitis B Virus(HBV) study. Our research group has collected HBV DNA sequences, either genotype B or C, from over 200 patients specifically for this project. In the molecular evolution analysis and clustering, three subgroups have been identified in genotype C and a clustering method has been developed to separate the subgroups. In the feature selection process, potential markers are selected based on Information Gains for further classifier learning. Initially meaningful rules are learnt by our algorithm called the Rule Learning which is based on Evolutionary Algorithm. Later, two new classification methods based on the Nonlinear Integrals and the Nonlinear Integral with Polynomial Kernel have been developed. Good performance of the new methods come from the use of the Fuzzy Measure and the relevant Nonlinear Integrals. The nonadditivity of the Fuzzy Measure reflects the importance of the predictive features as well as their interactions. Both classifiers give explicit information on the importance of the individual mutated sites and their interactions towards the classification (potential causes to liver cancer in our case). A thorough comparison study of these new methods with existing methods is also detailed.

1.7 Thesis Navigation

There are totally seven chapters in this thesis. In this chapter, we have given the research background of this thesis. We have listed the main challenges and the major research approaches in data mining problems. The remainder of this thesis concentrates on the generalizations of the Nonlinear Integral and their abilities in data mining.

Chapter 2 gives the background knowledge about several concepts and methodologies which pave the way for us to develop the generalizations of the Nonlinear Integral.

Chapter 3 concentrates on the investigation of the Double Nonlinear Integral(DNI). The formal definition of the DNI with the Signed Fuzzy Measure will be introduced. We develop a numerical calculation algorithm to derive the integration result of the DNI efficiently. Then the classification model based on DNI is discussed. We propose a GA-based double optimization algorithm to optimize the related parameters, including the element values of the Signed Fuzzy Measure. Its performance on synthetic data sets and real-life data from UCI database is presented at the end of the chapter.

In Chapter 4, a Nonlinear Integrals with Polynomial Kernel (NIPK) based classifier which is able to classify the complicated distributed data efficiently and effectively, is presented. Mechanism of the NIPK projection is illustrated first. After that, a GA-based adaptive classifier-learning algorithm is described. Performance of the NIPK based classifier is then verified by both synthetic and realistic datasets and compared with several traditional approaches.

Chapter 5 focuses on the Upper and Lower Nonlinear Integral based multiregression model. We give the formal definition of the ULNI based on the extension principle. We illustrate how to calculate the ULNI and what is the meaning of ULNI. Then, we present a new multi-regression model based on the ULNI. It is implemented by a multiobjective optimization algorithm based on Genetic Algorithm. Performance of the ULNI based multi-regression model is then evaluated by applying to synthetic datasets and a weather problem.

Chapter 6 presents a framework which includes Nonlinear Integrals and Rule Learning for a bioinformatics problem, data mining on the HBV DNA sequences. Our new methods can obtain better sensitivity which is cared most by Doctor. We discover some important sites in the sequences for the disease diagnosis by using L1-norm regularization.

Finally, in Chapter 7, we conclude this thesis with its contributions and limitations. We also point out some research directions and feasibilities for future work.

Chapter 2

Background Knowledge

Generalization of the Nonlinear Integral is a multidisciplinary study which combines different elements in Nonlinear Integral, data mining, and evolutionary computation. In this chapter, selective background knowledge is provided to help readers to understand the contents in the subsequent chapters. The earliest challenge to classical measure theory came from a French mathematician-G. Choquet[15], for which he coined the name theory of capacities. A Choquet capacity is a set function that associates a real number with each subset of the universal set employed and is continuous and monotonic with respect to set inclusion[16]. The two main concepts of this chapter are Fuzzy Measures and Nonlinear Integrals which were envisioned by Sugeno[17, 18] in his efforts to compare membership grade functions of fuzzy sets with probability. Similar to Choquet's capacities, Fuzzy Measures are too loose to allow us to develop a theory that would capture their full generality and yet, are of pragmatic utility. Moreover, it is too restrictive for several special types of Fuzzy Measures in some application contexts. More generalization was proposed by Wang[19, 20].

Required background in Set Theory will be introduced in section 2.1.

In Section 2.2, some elementary concepts on the Fuzzy Measure are given. The classical Nonlinear Integral is introduced in Section 2.3. As a basic optimization tool used in the thesis, Genetic Algorithms and some relevant improvement operators are demonstrated in Section 2.4.

2.1 Set Function

We denote the set of real numbers by $\mathcal{R}$ and the set of non-negative real numbers by $\mathcal{R}_+$. All functions we deal with are real-valued. In our thesis, X is assumed to be a finite set. X is called the universe of discourse. The elements of X are called points denoted by $x_1, x_2, ..., x_n$. A set containing no point is called the empty set, and is denoted by $\emptyset$.

Definition 2.1 *The class of all subsets of X is called the power set of X and is denoted by $\mathcal{P}(X)$.*

Definition 2.2 *A function ξ defined on a family of sets is called as a set function.*

Let ξ be a set function defined on $\mathcal{P}(X)$.

(i) The set function ξ is said to be additive if for every pair of disjoint subsets A and B of X

$$\xi(A \cup B) = \xi(A) + \xi(B)$$

(ii) The set function ξ is said to be monotone if for every pair of subsets A and B of X such that $A \subset B$

$$\xi(A) \leq \xi(B)$$

(iii) The set function ξ is said to be normalized if

$$\min\{\xi(A) | A \subset X\} = 0 \text{ and } \max\{\xi(A) | A \subset X\} = 1$$

If ξ is additive, then $\xi(\emptyset)$ =0 since $\xi(\emptyset) = \xi^+(\emptyset) + \xi^-(\emptyset)$. A non-negative additive set function is monotone; if ξ is non-negative and additive. and if $A \subset B \subset X$, then $\xi(B) = \xi(A \cup (B|A)) = \xi(A) + \xi(B|A) \geq \xi(A)$, where $B|A = \{x|x \in B, x \notin A\}$, since $\xi(B|A) \geq 0$. Since X is a finite set, an additive set function ξ defined on $\mathcal{P}(X)$ can be represented as

$$\xi(A) = \sum_{x \in A} \xi(\{x\}) \text{ for } A.$$

Definition 2.3 *For a set function ξ defined on $\mathcal{P}(X)$ such that $\xi(\emptyset)$ =0, its conjugate set function $\overline{\xi}$ is defined as*

$$\overline{\xi}(A) = \xi(X) - \xi(A^c) \textit{ for all } A$$

where A^c is the complement of A.

Based on these set theories, the Fuzzy Measure will be induced.

2.2 Fuzzy Measure and Signed Fuzzy Measure

The traditional aggregation tool for information fusion is the weighted average method, which is a linear integral (i.e., the Lebesgue integral) essentially. It is based on the assumption that the involved information sources are not interactive, and hence, their weighted effects are viewed as additive. Obviously, this assumption is not realistic in many applications. To describe the interaction among information sources, a new mathematical tool - the Fuzzy Measure and its generalization - the Signed Fuzzy Measure[16, 64] have been developed.

Fuzzy Measure is a generalization of classical measure theory. Let X be a universal set and $\mathcal{F}$ be δ-algebra of X. Here, δ-algebra $\mathcal{F}$ is a nonempty collection of subsets of X such that the following hold:

1) The empty set is in $\mathcal{F}$;

2) If A is in $\mathcal{F}$, then so A^c is the complement of A;

3) If A_n is a sequence of elements of F, then the union of the $A_n s$ is in $\mathcal{F}$.

$(X, \mathcal{F})$ is called a measurable space[16, 64]. The universal set X is not necessarily finite. However, in most real problems, when a set of features is considered as the universal set, of course, X is finite. In this case, $\mathcal{P}(X)$, the power set of X, is usually taken as $\mathcal{F}$. In the thesis, the set functions will be always defined on a given δ-algebra, $\mathcal{F}$. When X is finite, $\mathcal{F}$ will always be $\mathcal{P}(X)$.

Definition 2.4 *A measure on X is a non-negative additive set function defined on $\mathcal{P}(X)$. A normalized measure is called a probability measure. A signed measure on X is an additive set function defined on $\mathcal{P}(X)$.*

A measure measures the size of sets. The number of elements in a set is a kind of measure of the size of sets.

Definition 2.5 *A set function $\mu : \mathcal{P}(X) \to [0, \infty]$ is called a Fuzzy Measure if it is monotonic as $\mu(A) \leq \mu(B)$ $\forall A \subset B$ on $\mathcal{P}(X)$, and vanishes at empty set, that is, $\mu(\emptyset) = 0$.*

A classical illustration on Fuzzy Measure is an example about workers. For convenience, we recite it[22] as follows:

Example 2.1 *Let X be the set of all workers in a workshop, and suppose they produce the same products. For each $A \subset \mathcal{P}(X)$, we consider the situation that the members of group A work in the workshop. Each group may have various ways to work: various combinations of joint work and individual work. Let $\mu(A)$ be the number of the products made by group A in one hour. Then the set function $\mu : \mathcal{P}(X) \to [0, \infty]$ is monotone and $\mu(\emptyset) = 0$, and therefore it is a Fuzzy Measure. The Fuzzy Measure is not additive necessarily. Let A*

Table 2.1: An example of Fuzzy Measure defined on $X = \{x_1, x_2, x_3\}$

Sets	Value of μ	Sets	Value of μ
$\emptyset$	0	$\{x_3\}$	0.2
$\{x_1\}$	0.2	$\{x_1, x_3\}$	0.5
$\{x_2\}$	0.4	$\{x_2, x_3\}$	0.9
$\{x_1, x_2\}$	0.5	$\{x_1, x_2, x_3\}$	1.0

and B be disjoint subsets of X, and consider the productivity of the coupled group $\mu(A \cup B)$. If A and B work separately, then $\mu(A \cup B) = \mu(A) + \mu(B)$. But, since they generally interact on each other, the equality may not always hold. The effective cooperation of members of $\mu(A \cup B)$ yields the inequality $\mu(A \cup B) > \mu(A) + \mu(B)$. On the other hand, the incompatibility between A's operation and B's yields the opposite inequality $\mu(A \cup B) < \mu(A) + \mu(B)$.

To further understand the practical meaning of the Fuzzy Measure we consider the elements in a universal set X as a set of predictive features to predict a certain objective. Then, for each individual predictive feature as well as each possible combination of the predictive features, a distinct value of a Fuzzy Measure is assigned to describe its influence to the objective. Due to the nonadditivity of the Fuzzy Measure, the influences of the predictive features to the objective are dependent such that the global contribution of them to the objective is not just the simple sum of their individual contributions.

Here is an example. Assume that we have observed three symptoms of a patient and want to determine which disease he or she is suffering. The symptoms are regarded as the information sources. which form the universal set denoted by $X = \{x_1, x_2, x_3\}$. Their individuals as well as joint influences on the prediction of disease are specified by a Fuzzy Measure μ defined in Table 2.1.

Here, $\mu(\{x_2, x_3\}) > \mu(\{x_2\}) + \mu(\{x_3\})$ indicates that the joint contribution of x_2 and x_3 to the diagnosis is greater than the sum of their individual contributions. This shows that the interaction between x_2 and x_3 is enhancing the influences of each other. On the other hand, $\mu(\{x_1, x_2\}) < \mu(\{x_1\}) + \mu(\{x_2\})$ shows that x_1 and x_2 are restraining each other. Note that, the essential properties of Fuzzy Measure are monotonicity and vanishing at the empty set. This implies that Fuzzy Measure only allows its value to be nonnegative.

Definition 2.6 *A Signed Fuzzy Measure is called a generalized Fuzzy Measure if it is nonnegative, that is*

$$\mu(A) > 0, \forall A \in \mathcal{P}(X)$$

Note that, the essential properties of Fuzzy Measure are monotonicity and vanishing at the empty set. This implies that Fuzzy Measure only allows its value to be nonnegative. However, the monotonicity and non-negativity of Fuzzy Measure are too restrictive for real applications. Thus, Signed Fuzzy Measure, which is a generalization of Fuzzy Measure, has been defined[22] and applied.

Definition 2.7 *A set function $\mu : \mathcal{P}(X) \to [-\infty, \infty]$ is called a Signed Fuzzy Measure provided that $\mu(\emptyset) = 0$.*

A Signed Fuzzy Measure allows its value to be negative and frees monotonicity constraint. Thus, it is more flexible to describe the individual and joint contribution rates from the predictive features in a universal set toward some target.

Example 2.2 *The normal time of stomach empty of human is about 267 ± 174 minutes per dining. In medicinal research, one or a set of drugs are adopted to adjust the time of patients stomach empty. When patient takes* *more than one drug, interaction among drugs make joint efficiencies toward the stomach empty time. The joint efficiency usually is not simply equal to the sum of efficiencies of each individual drug to the stomach empty time. Considering three drugs A, B, and C, a piece of research shows that individually taking drug A or C may decrease the stomach empty time while taking drug B may increase it. Jointly taking drugs A and B may increase the stomach empty time while taking A and C, or taking B and C, may decrease it. Taking drugs A, B, and C simultaneously may decrease the stomach empty time as well. The research also finds that within a proper range the efficiencies are constants approximately, that is, the effects are proportional to the amount of drug(s). Quantifying these observations, if μ is used to denote the efficiencies (number of minutes reduced by per unit of drug) of drugs toward the stomach empty time, we have $\mu(A) = 5.0$, $\mu(B) = -4.0$, $\mu(\{A, B\}) = -1.0$. $\mu(C) = 7.0$, $\mu(\{A, C\}) = 25.0$, $\mu(\{B, C\}) = 2.0$, and $\mu(\{A, B, C\}) = 15.0$. Here, μ is a Signed Fuzzy Measure defined on the power set of $X = \{A, B, C\}$.*

A Signed Fuzzy Measure can be decomposed as difference of two generalized Fuzzy Measures. Since we need not consider additivity now, such decomposition is much simpler than the Jordan decomposition where a signed measure is decomposed to be the difference of two measures[64].

Definition 2.8 *Let μ be a Signed Fuzzy Measure. A pair of two generalized Fuzzy Measures, ν^+ and ν^-, satisfying*

$$\mu(A) = \nu^+(A) - \nu^-(A) \quad \forall A \in \mathcal{P}(X)$$

(simply, we write $\mu = \nu^+ - \nu^-$), is called a nonnegative decomposition of μ.

We may omit word nonnegative in the above definition if there is no confusion, and simply call it a decomposition. For a given Signed Fuzzy Measure, there are infinite decompositions. Among them, there is one which is the smallest decomposition.

Definition 2.9 *The smallest decomposition of a Signed Fuzzy Measure is the composition, μ^+ and μ^-, such that $\mu^+ < \nu^+$ and $\mu^- < \nu^-$ for any decomposition, ν^+ and ν^-, of μ.*

The smallest decomposition of is unique. It can be expressed as

$$\mu^+(A) = \begin{cases} \mu(A) & if\ \mu(A) \geq 0 \\ 0 & otherwise \end{cases}$$

and

$$\mu^-(A) = \begin{cases} -\mu(A) & if\ \mu(A) \leq 0 \\ 0 & otherwise \end{cases}$$

for any $A \in \mathcal{P}(X)$. μ^+ and μ^- are called the positive part and the negative part of μ respectively.

Example 2.3 *The smallest decomposition of the Signed Fuzzy Measure in Example 2.2 is the composition of μ^+ and μ^-. Here, we have $\mu^+(A) = 5.0$, $\mu^+(B) = 0$, $\mu^+(A, B) = 0.0$, $\mu^+(C) = 7.0$, $\mu^+(A, C) = 25.0$, $\mu^+(B, C) = 2.0$, $\mu^-(A, B, C) = 15.0$, and $\mu^-(A) = 0$, $\mu^-(B) = 4.0$, $\mu^-(A, B) = 1.0$, $\mu^-(C) = 0$, $\mu^-(A, C) = 0$, $\mu^-(B, C) = 0$, $\mu^-(A, B, C) = 0$.*

In this thesis, we assume μ is a Signed Fuzzy Measure on $\mathcal{P}(X)$, i.e. $\mu : \mathcal{P}(X) \to [-\infty, \infty]$ and $\mu(\emptyset) = 0$. For convenience, $\mu(\{x_1\}), \mu(\{x_2\}), \ldots, \mu(\{x_n\})$, $\mu(\{x_1, x_2\}), \ldots, \mu(\{x_1, x_2, \ldots, x_n\})$ are sometimes abbreviated by $\mu_1, \mu_2, \ldots, \mu_n$, $\mu_{12}, \ldots, \mu_{12\ldots n}$, respectively.

2.3 Nonlinear Integral

Nonlinear Integrals, such as the Choquet Integrals[22, 23] with respect to Fuzzy Measure or Signed Fuzzy Measure, are recently becoming popular as powerful aggregation tools in the data mining study. This section is devoted to introducing some basic concepts on the Fuzzy Measure theory and the Nonlinear Integral with their applications in the data mining domain.

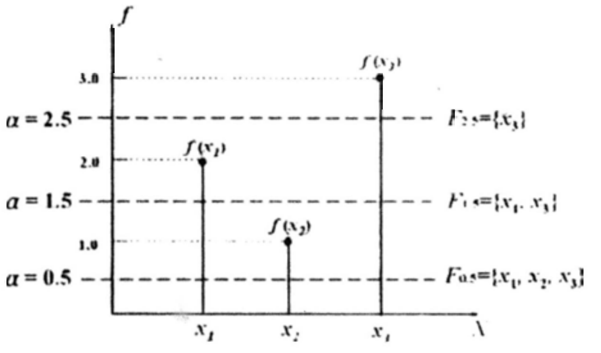


Figure 2.1: The α *cut* of a real-valued function.

2.3.1 Nonlinear Integral with Real-valued Integrand

Definition 2.10 *Let $(X, \mathcal{P})$ be a measurable space and be a Signed Fuzzy Measure defined on $(X, \mathcal{P})$. The Nonlinear Integral of a real-valued function $f : X \to [-\infty, \infty]$ is defined as*

$$\int f d\mu = \int_{-\infty}^{0} [\mu(F_\alpha) - \mu(X)]\, d\alpha + \int_{0}^{\infty} \mu(F_\alpha)\, d\alpha$$

where $F_\alpha = \{x | f(x) \geq \alpha\}$, for any $\alpha \in (-\infty, \infty)$, is called the $\alpha - cut$ of f.

The $\alpha - cut$ of f can be represented as a crisp set of X. For example, let $X = \{x_1, x_2, x_3\}$, and a real-valued function f is defined on X by $f(x_1) = 2.0$, $f(x_2) = 1.0$, and $f(x_3) = 3.0$, then the $\alpha - cut$ of f at $\alpha = 0.5, 1.5$ and 2.5 are crisp sets of X, described by $F_{0.5} = \{x_1, x_2, x_3\}$, $F_{1.5} = \{x_1, x_3\}$ and

$F_{2.5} = \{x_3\}$, respectively, as shown in Figure 2.1.

When $X = \{x_1, x_2, ..., x_n\}$ for any function $f : X \rightarrow [-\infty, \infty]$, both $[\mu(F_\alpha) - \mu(X)]$ and $\mu(F_\alpha)$ are functions of with bounded variance, and therefore, their Riemann integrals with respect α to exist and are finite. So, the Nonlinear Integral $\int f d\mu$ is well defined.

To calculate the value of the Nonlinear Integral of a given real-valued function f, usually the values of f, i.e.,$f(x_1), f(x_2), ..., f(x_n)$ should be sorted in a nondecreasing order so that $f(x_1') \leq f(x_2') \leq ... \leq f(x_n')$, where $(x_1', x_2', ..., x_n')$ is a certain permutation of $(x_1, x_2, ..., x_n)$. So the value of Nonlinear Integral can be obtained by

$\int f d\mu = \sum_{k=1}^{n} [f(x_i') - f(x_{i-1}')]\mu(x_i', x_{i+1}', ..., x_n')$, where $f(x_0') = 0$

For convenience, Wang[24] proposed a new scheme to calculate the value of a Nonlinear Integral with real-valued integrand by the inner product of two $(2^n - 1)$-dimension vectors as

$$\int f d\mu = \sum_{j=1}^{2^n - 1} z_j \mu_j$$

where,

$$z_j = \begin{cases} \min\limits_{i: frc(\frac{j}{2^i}) \in [\frac{1}{2}, 1]} f(x_i) - \max\limits_{i: frc(\frac{j}{2^i}) \in [0, \frac{1}{2}]} f(x_i), & if\ z_j \geq 0\ or\ j = 2^n - 1; \\ \\ 0 & otherwise \end{cases}$$

for $j = 1, 2, ..., 2^n - 1$ with a convention that the maximum on the empty set is zero. Here, $frc(\frac{j}{2^i})$ denotes the fractional part of $\frac{j}{2^i}$. In the above formula, if we express j in the binary form $j_n j_{n-1} ... j_1$, then $\{i | frc(\frac{j}{2^i}) \in [\frac{1}{2}, 1]\} = \{i | j_i = 1\}$ and $\{i | frc(\frac{j}{2^i}) \in [0, \frac{1}{2}]\} = \{i | j_i = 0\}$.

A significant advantage of this new calculation scheme is that it can easily discover the coefficients matrix of a system of linear equations with the unknown variables when the Nonlinear Integral is applied in further applications, such as regression and classification[24, 25, 26]. In those practical applications, values of the Signed Fuzzy Measure are usually considered as unknown parameters which are to be estimated using the training data sets. The adoption of this new scheme makes it convenient for using an algebraic method, such as the least square method, to estimate the value of μ, and furthermore, to reduce the complexity of computation.

Example 2.4 *Assumed that 60 products, 40 products and 10 products are produced respectively by worker A, worker B and worker C. Let x_1 denotes A, x_2 denotes B, and x_3 denotes C, a real-valued function f can be defined as $f(x_1) = 60, f(x_1) = 40$ and $f(x_3) = 60$. Then the synthetic effect of workers is just the Nonlinear Integral of f with respect to μ. Noting that $\mu_1 = \mu(\{x_1\}) = 5.0$, $\mu_2 = \mu(\{x_2\}) = 4.0$, $\mu_3 = \mu(\{x_1, x_2\}) = 1.0$, $\mu_4 = \mu(\{x_3\}) = 7.0$, $\mu_5 = \mu(\{x_1, x_3\}) = 25.0$, $\mu_6 = \mu(\{x_2, x_3\}) = 2.0$ and $\mu_7 = \mu(\{x_1, x_2, x_3\}) = 15.0$. Using the above equation, we have*

$min(f(x_1)) - max(f(x_2), f(x_3)) = 60 - 40 = 20 \Rightarrow z_1 = 20$

$min(f(x_2)) - max(f(x_1), f(x_3)) = 40 - 60 = -20 \Rightarrow z_2 = 0$

$min(f(x_1), f(x_2)) - max(f(x_3)) = 40 - 10 = 30 \Rightarrow z_3 = 30$

$min(f(x_3)) - max(f(x_1), f(x_2)) = 10 - 60 = -50 \Rightarrow z_4 = 0$

$min(f(x_1), f(x_3)) - max(f(x_2)) = 10 - 40 = -30 \Rightarrow z_5 = 0$

$min(f(x_2), f(x_3)) - max(f(x_1)) = 10 - 60 = -50 \Rightarrow z_6 = 0$

$min(f(x_1), f(x_2), f(x_3)) = 10 \Rightarrow z_7 = 10$

Then, the total number of products produced by workers can be computed, that is,

$$\int f d\mu = \sum_{j=1}^{2^n-1} z_j \mu_j = [20\ 0\ 30\ 0\ 0\ 0\ 10] \begin{bmatrix} 5.0 \\ -4.0 \\ -1.0 \\ 7.0 \\ 25.0 \\ 2.0 \\ 15.0 \end{bmatrix} = 220$$

Generally, the Nonlinear Integral is not linear due to the nonadditivity of the Signed Fuzzy Measure μ[16, 23, 27]. However, as a special case, when a Signed Fuzzy Measure μ is an additive measure, i.e., when μ is additive, the Nonlinear Integral coincides with the Lebesgue-like integral[64] and is linear. So, the Nonlinear Integral is a generalization of the Lebesgue-like integral. The following theorems give some properties of the Nonlinear Integral, including the decomposability, the continuity, and the monotonicity. They are useful for the discussions on the Generalization of the Nonlinear Integral in the subsequent chapters.

Theorem 2.1 *(Decomposability) For any given measurable function $f : X \to [-\infty, \infty]$ and the Signed Fuzzy Measure on $[X, \mathcal{F}]$,*

$$\int f d\mu = \int f d\mu^+ - \int f d\mu^-$$

where μ^+ and μ^- are decompositions of μ, if both integrals in the right side are well defined and "$(\infty) - (\infty)$ " or "$(-\infty) - (-\infty)$" does not occur.

Theorem 2.2 *(Continuity) The Nonlinear Integral $\int f d\mu$ is continuous with respect to the integrand f, that is , for any given $\epsilon > 0$, there exists $\delta > 0$, such that*

$$|\int f_1 d\mu - \int f_2 d\mu| < \epsilon$$

whenever $f_1 - f_2 < \delta$.

Theorem 2.3 *(Monotonicity) If μ is a Fuzzy Measure, then the Nonlinear Integral $\int f d\mu$ is monotonic with respect to the integrand f, that is,*

$$\int f_1 d\mu < \int f_2 d\mu \quad if\ f_1 < f_2.$$

Note that, the monotonicity property does not hold for the Nonlinear Integral with respect to a Signed Fuzzy Measure.

2.3.2 Applications of the Nonlinear Integral on Data Mining

As an aggregation tool, the Nonlinear Integral has become a powerful methodology to solve many data mining problems. In general, its applications can be categorized into the following groups:

Regression[28, 29, 30]: In statistics, regression analysis refers to techniques for modeling and analyzing several variables, when the focus is on the relationship between a dependent variable and one or more independent variables. More specifically, regression analysis helps us understand how the typical value of the dependent variable changes when any one of the independent variables is varied, while the other independent variables are held fixed. Most commonly, regression analysis estimates the conditional expectation of the dependent variable given the independent variables-that is, the average value of the dependent variable when the independent variables are held fixed. Less commonly, the focus is on the location parameter of the conditional distribution of the dependent variable given the independent variables. In all cases, the estimation target is a function of the independent variables called the regression function. In regression analysis, it is also of interest to characterize the variation of the dependent variable around the regression function, which can be described by a probability distribution.

Regression analysis is widely used for prediction (including forecasting of time-series data). Use of regression analysis for prediction has substantial overlap with the field of machine learning. Regression analysis is also used to understand which among the independent variables are related to the dependent variable, and to explore the forms of these relationships. In restricted circumstances, regression analysis can be used to infer causal relationships between the independent and dependent variables.

A large body of techniques for carrying out regression analysis has been developed. Familiar methods such as linear regression and ordinary least squares regression are parametric, in that the regression function is defined in terms of a finite number of unknown parameters that are estimated from the data. Nonparametric regression refers to techniques that allow the regression function to lie in a specified set of functions, which may be infinite-dimensional.

The performance of regression analysis methods in practice depends on the form of the data-generating process, and how it relates to the regression approach being used. Since the true form of the data-generating process is not known, regression analysis depends to some extent on making assumptions about this process. These assumptions are sometimes (but not always) testable if a large amount of data is available. Regression models for prediction are often useful even when the assumptions are moderately violated, although they may not perform optimally. However when carrying out inference using regression models, especially involving small effects or questions of causality based on observational data, regression methods must be used cautiously as they can easily give misleading results. Since Fuzzy Measure can describe the interaction among dependent variables, Nonlinear Integral has been popularly applied to the regression problems[31]. The most typical regression model based on the Nonlinear Integral built a nonlinear multi-regression network which was proposed in[25, 32]. A Signed Fuzzy Measure has been considered to improve the regression model of the Nonlinear Integral in[24, 33].

Classification[34]: An important task in Machine Learning is classification, also referred to as pattern recognition, where one attempts to build algorithms capable of automatically constructing methods for distinguishing between different exemplars, based on their differentiating patterns. Pattern classification tasks are often divided into several sub-tasks:

1). Data collection and representation;

2). Feature selection and/or feature reduction;

3). Classification.

Data collection and representation are mostly problem-specific. Therefore it is difficult to give general statements about this step of the process. In broad terms, one should try to find invariant features, which describe the differences in classes as best as possible. Feature selection and feature reduction attempt to reduce the dimensionality (i.e. the number of features) for the remaining steps of the task. Finally, the classification phase of the process finds the actual mapping between patterns and labels (or targets). In many applications the second step is not essential or is implicitly performed in the third step.

Many classification methods have been proposed based on various approaches [35]. Due to nonlinearity existing in the real world, some linear methods can not have high classification accuracy and satisfy the requirement. However, the contribution rate of each combination of predictive features including each singleton towards the decisive feature can be represented by a

Fuzzy Measure. The nonadditivity of the Fuzzy Measure reflects the interactions among the predictive features. Recently, many methods which attempt to use Nonlinear Integrals as aggregation tools[36, 37, 38, 39] has obtained encouraging results. In these existing methods, if there are m classes and n predictive features, then m sets of Fuzzy Measures are used and $m(2^n - 2)$ values of Fuzzy Measures are needed to be determined.

Unlike the methods above, another method called WCIPP (Weighted-Choquet-Integral based Projection Pursuit) uses a weighted Choquet Integral as a projection tool[26]. In WCIPP, only one Fuzzy Measure defined on the power set of the set of all predictive features is used to describe the importance of each predictive feature as well as their interactions[16, 40, 41] towards the classification of the records.

2.4 Genetic Algorithm

Genetic Algorithms (GAs) are general purpose search algorithms which use principles inspired by natural evaluation to evolve solutions to problems[42, 43]. This section aims to give a brief introduction on Genetic Algorithms which are used as a primary optimization method in the thesis.

2.4.1 What is a Genetic Algorithm

The Genetic Algorithm is a method for solving both constrained and unconstrained optimization problems that is based on natural selection, the process that drives biological evolution. A Genetic Algorithm starts off with a population of randomly generated individuals and repeatedly modifies a population of individual solutions. At each step, the genetic algorithm selects individuals of *chromosome* which represent candidate solutions to a problem at random from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution. We can apply the genetic algorithm to solve a variety of optimization problems that are not well suited for standard optimization algorithms, including problems in which the objective function is discontinuous, nondifferentiable, stochastic, or highly nonlinear.

The GA differs from a classical, derivative-based, optimization algorithm in two main ways, as summarized in Table 2.2.

The algorithm advances toward better individuals by applying genetic operators modeled on the genetic processes occurring in nature. The population undergoes evolution in a form of natural selection. During successive iterations, called generations, individuals in the population are rated for their adaptations as solutions, and on the basis of these evaluations, a new population of individuals is formed using a selection mechanism and specific genetic operators such as crossover and mutation. A fitness function must be derived for each problem to be solved. Given a particular individual of chromosome, the fitness function returns a single numerical fitness, which is supposed to be proportional to the utility or the adaptation of the solution which the individual represents. The canonical Genetic algorithm is formulated in Table 2.3 as pseudo code.

2.4.2 Coding representation

Coding representation is a key issue in GA approaches because GAs directly manipulates a coded representation of the problem and because the representation schema can severely limit the window through which a system observes its world[44].

Table 2.2: Comparison between Genetic Algorithm and Classical Algorithm

Classical Algorithm	Genetic Algorithm
Generates a single point at each iteration. The sequence of points approaches an optimal solution.	Generates a population of points at each iteration. The best point in the population approaches an optimal solution.
Selects the next point in the sequence by a deterministic computation.	Generates the next population by genetic operator(e.g. crossover, mutation) which uses random number generators.

Table 2.3: Pseudo Code of A Canonical GA

```
The Canonical GA(pseudo code):
 Choose initial population
 Evaluate each individual's fitness
 Determine population's average fitness
 Repeat
          Select best-ranking individuals to reproduce
          Mate pairs at random
          Apply mutation operator
          Evaluate each individual's fitness
          Determine population's average fitness
 Until terminating condition(e.g. until at least one individual has
          the desired fitness or enough generations have passed)
```

Fixed-length and binary coded strings for the representation solution have dominated GA research due to the theoretical results which show them to be the most effective ones[45] and they are convenient and simple to implement. The bit strings of parameters are concatenated together to give a single bit string (or "chromosome") which represents the entire vector of parameters. In biological terminology, each bit position corresponds to a gene of the chromosome, and each bit value corresponds to an allele. By binary coding, the problem being considered is translated into a combinatorial one where points of the search space are corners of a high-dimensional cube.

However, the GA's good properties do not stem from the use of bit strings. It would seem particularly natural to represent the genes directly as real numbers for optimization problems of parameters with variables in continuous domain[46]. Then an individual of chromosome is a vector of floating point numbers, the precisions of which will determine the corresponding precisions of solutions. The size of the chromosome is kept to be the same as the length of the vector. This is called the real coding method where each gene represents a variable of the problem. Values of the genes in a string of chromosome are forced to remain in the interval established by the variables which they represent, so the related genetic operators must observe this requirement.

2.4.3 Genetic Operators

Good performance of GA is achieved through diverse genetic operators. Typically, the genetic algorithm uses three main types of rules at each step to create the next generation from the current population:

* Selection rules select the individuals, called parents, which contribute to the population at the next generation.

* Crossover rules combine two parents to form children for the next generation.

* Mutation rules apply random changes to individual parents to form children.

This subsection describes some typical and specifically designed operators which are used in the thesis.

Selection Operators

The selection function chooses parents for the next generation based on their scaled values from the fitness scaling function. An individual can be selected more than once as a parent, in which case it contributes its genes to more than one child. The default selection option, stochastic uniform, lays out a line in which each parent corresponds to a section of the line of length proportional to its scaled value. The algorithm moves along the line in steps of equal size. At each step, the algorithm allocates a parent from the section it lands on.

A more deterministic selection option is Remainder, which performs two steps:

In the first step, the function selects parents deterministically according to the integer part of the scaled value for each individual. For example, if an individual's scaled value is 2.3, the function selects that individual twice as a parent.

In the second step, the selection function selects additional parents using the fractional parts of the scaled values, as in stochastic uniform selection. The function lays out a line in sections, whose lengths are proportional to the fractional part of the scaled value of the individuals, and moves along the line in equal steps to select the parents.

Note that if the fractional parts of the scaled values all equal zero, as can occur using top scaling, the selection is entirely deterministic. A selection scheme determines the probability of an individual being selected for reproduction and producing offspring by the crossover and/or mutation operators. Three typical selection schemes which includes Fitness proportionate selection, rank-based, and tournament selections will be introduced.

Fitness proportionate selection, also known as **roulette-wheel selection**, is the simplest selection scheme in Genetic Algorithms, also called the fitness proportionate selection.

It is a stochastic algorithm in which individuals are mapped to contiguous segments of a line, such that each individual's segment is equal in size to its

Individual	1	2	3	4	5
Fitness Value	4.0	3.0	2.0	1.0	0.0
Selection Probability	40%	30%	20%	10%	0%

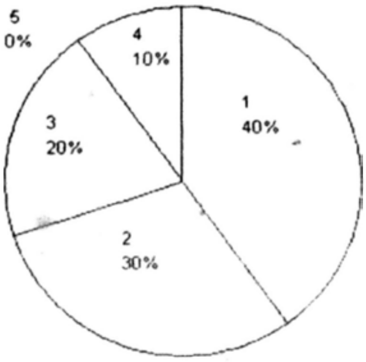


Figure 2.2: An Example of Roulette Wheel Selection

fitness. A random number is generated and the individual whose segment spans the random number is selected. If f_i is the fitness of individual i in the population, its probability of being selected is

$$p_i = \frac{f_i}{\sum_{i=1}^{n} f_i}$$

where n is the number of individuals in the population.

This technique is analogous to a roulette wheel with each slice proportional in size to the fitness, as depicted in Figure 2.2.

Rank-based selection: Individuals in population are sorted by objective values in rank-based fitness assignment. The fitness of each individual was not assigned according to its actual objective value but the position in the individuals rank. Hence the rank-based selection can maintain a constant selection pressure in the evolutionary search and avoid some of the problems encountered by roulette wheel selection.

There are many different rank-based selection schemes. We introduce two of them here. Assume the best individual in a population is ranked first. The probability of selecting individual can be calculated linearly as follows:

$$p_i = \frac{1}{2}(\eta_{max} - (\eta_{max} - \eta_{min})\frac{i-1}{n-1})$$

where n is the population size, η_{max} and η_{min} are two parameters satisfying conditions $\eta_{max} \geq \eta_{min} \geq 0$ and $\eta_{max} + \eta_{min} = 0$. Another rank-based selection scheme with a stronger selection pressure can be implemented by the nonlinear ranking scheme as[47]

$$p_i = \frac{i}{\sum_{j=1}^{n} j}$$

Tournament selection: Tournament selection involves running several "tournaments" among a few individuals chosen at random from the population. The winner of each tournament (the one with the best fitness) is selected for crossover. Selection pressure is easily adjusted by changing the tournament size. If the tournament size is larger, weak individuals have a smaller chance to be selected. Deterministic tournament selection selects the best individual (when $p = 1$) in any tournament. A 1-way tournament ($k = 1$) selection is equivalent to random selection. The chosen individual can be removed from the population that the selection is made from if desired; otherwise individuals can be selected more than once for the next generation.

Tournament selection provides selection pressure by holding a tournament among s competitors. s is the tournament size. The winner of the tournament is the individual with the highest fitness of the s tournament competitors, and the winner is then inserted into the mating pool. The mating pool which is comprised of tournament winners has a higher average fitness than the average population fitness. This fitness difference provides the selection pressure, which drives the GA to improve the fitness of each succeeding generation. Simply increasing the tournament size can increase selection pressure, as the winner from a larger tournament will, on average, have a higher fitness than the winner of a smaller tournament[48].

Tournament selection has several benefits: it is efficient to code, works on parallel architectures and allows the selection pressure to be easily adjusted.

Crossover Operators

Essence of a crossover operator is the inheritance of information (genes) from two or more parents to offspring. Crossover is a genetic operator used to vary the programming of a chromosome or chromosomes from one generation to the next. It is analogous to reproduction and biological crossover, upon which genetic algorithms are based. There are several kinds of crossover operators for different coding schemes introduced as follows.

Crossover for binary coding

Common recombination operators for binary strings include k-point crossover ($k \geq 1$) and uniform crossover, although there are many other variants.

One-point crossover: A single crossover point on both parents' organism strings is selected. All data beyond that point in either organism string is swapped between the two parent organisms. The resulting organisms are the children as Figure 2.3.

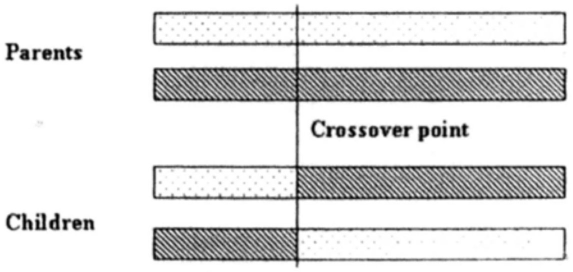


Figure 2.3: One-point Crossover

k-point crossover: Choose two individuals from current population as parents, cut both of them into $k+1$ segments at k randomly chosen points, respectively. Then switch their segments to form two new strings of chromosome as offspring. Figure 2.4 shows a k-point crossover when $k = 2$. in which everything between the two points is swapped between the parent organisms. rendering two child organisms.

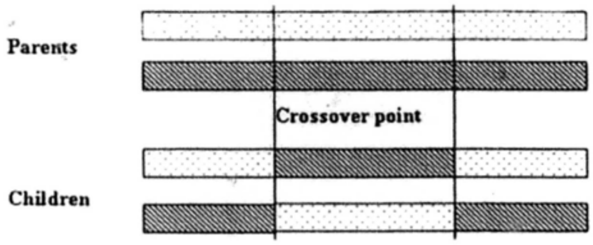


Figure 2.4: One-point Crossover

Uniform crossover: An offspring is generated by taking its each bit or character from the corresponding bit or character in one of the two parents. The parent from which the bit or the character is to be taken is uniformly randomly chosen.

Crossover for real coding

Assuming that $C_1 = (c_1^1.c_2^1,\ldots,c_n^1)$ and $C_2 = (c_1^2, c_2^2, \ldots, c_n^2)$ are two chromosomes that have been selected for applying the crossover operator.

We should point out that since each crossover operator generates a different offspring number; a selection mechanism for deciding the ones that shall be included in the population is sometimes needed. An offspring can be generated by randomly choosing from $[c_i^1, c_i^2]$[49]. If a position is randomly chosen, then two new individual will be produced by exchanging the value on this position in parents[50, 51]. More complicated methods exist in many situation, such as Arithmetical recombination[50], in which two offspring, $H_k = (h_1^k, h_2^k, \ldots, h_n^k)$, $k = 1, 2$, are generated, where $h_i^1 = \lambda c_i^1 + (1-\lambda)c_i^2$ and $h_i^2 = \lambda c_i^2 + (1-\lambda)c_i^1$ and is a constant (uniform arithmetical recombination) or varies with regard to the number of generations carried out (non-uniform arithmetical recombination). Linear recombination[51] generates three offspring. $H_k = (h_1^k, \ldots, h_i^k, \ldots, h_n^k)$, $k = 1, 2, 3$, where h_i^k is a linear combination of c_i^1 and c_i^2. With this type of recombination, an offspring selection mechanism is applied, which chooses the two most promising offspring of the three to replace their parents in the population.

Mutation Operators

In Genetic Algorithms, mutation is a genetic operator used to maintain genetic diversity from one generation of a population of chromosomes to the next. It is analogous to biological mutation. This diversity allows the exploration of larger regions of the search space. The classic example of a mutation operator involves a probability that an arbitrary bit in a genetic sequence will be changed from its original state. A common method of implementing the mutation operator involves generating a random variable for each bit in a sequence. This random variable tells whether or not a particular bit will be modified. The purpose of mutation in GAs is to allow the algorithm to avoid local minima by preventing the population of chromosomes from becoming too similar to each other, thus slowing or even stopping evolution. This reasoning also explains the fact that most GA systems avoid only taking the fittest of the population in generating the next but rather a random (or semi-random) selection with a weighting toward those that are fitter.

The mutation operator for binary strings is simple. An individual of chromosome is selected from current population as parent. The offspring will be formed by changing or replacing several bits from 0 to 1 or from 1 to 0. The mutation operators used for vectors of real values usually change values on some positions based on certain probability distributions, such as uniform. lognormal, Gauss (normal) and Cauchy distributions[50]. A random walk method is used to replacing the selected gene c_i by c_i' which is defined by $c_i = c_i \pm \Delta$. Δ is a small real value determined by users with its default as 0.005.

2.5 Chapter Summary

In this chapter, we have outlined some basic concepts and properties of the Nonlinear Integrals and the Genetic Algorithms. Some specifically designed genetic operators have also been demonstrated. This chapter paves the way for our investigation on the Generalization of the Nonlinear Integral in the subsequent chapters.

Chapter 3

Double Nonlinear Integrals

In this Chapter, a new classification model based on projection with Double Nonlinear Integrals(DNI) is proposed. Many classification methods have been proposed based on various approaches [35]. Due to nonlinearity existing in the real world, some linear methods can not satisfy the requirement with high classification accuracy. However, the contribution rate of each combination of predictive features including each singleton towards the decisive feature can be represented by a Fuzzy Measure. The nonadditivity of the Fuzzy Measure reflects the interactions among the predictive features. Recently, many methods which attempt to use Nonlinear Integrals as aggregation tools[36, 37, 39, 38] has obtained quite encouraging results. In these existing methods, if there are m classes and n predictive features, then m sets of Fuzzy Measures are used and $m(2^n - 1)$ values of Fuzzy Measures are needed to be determined.

Unlike the methods above, another method called WCIPP (Weighted-Choquet-Integral based Projection Pursuit) use a weighted Choquet Integral as a projection tool[26]. In WCIPP, only one Fuzzy Measure defined on the power set of the set of all predictive features is used to describe the importance of each predictive feature as well as their interactions[13, 16, 40] towards the classification of the records. The original classification problem in n-dimensional space is transformed to a one-dimensional space problem through the optimal projection based on Nonlinear Integrals. But plenty of information may be missed in the projecting process. In some special cases which will be described in the following sections, there exists projection overlapping when the data to be classified have special distribution in the data space. For example, one group of the data is surrounded by the data of another group, or the number of classes for the data is large. This will lead to lower classification accuracy. To solve this problem, we propose a new classification model based on Double Nonlinear Integrals in this chapter. Double Nonlinear Integrals can lessen loss of information due to the intersection of different classes on real axis in WICPP. Accuracy will be increased accordingly. Although computation complexity will be linearly increased, it is still acceptable.

This chapter is organized as follows. Section 3.1 gives out the theoritical knowledge of Double Nonlinear Integral. Construction of the classification model based on projection with Double Nonlinear Integrals is described in details in section 3.2. The experimental results of the new classification model for each dataset are presented in Section 3.3. Section 3.4 presents the conclusions and some directions for future work.

3.1 Double Nonlinear Integrals

In classification, we are given a data set consisting of l example records, called training set, where each record contains the value of a decisive feature, Y, and the value of predictive features $x_1, x_2, ..., x_n$. Positive integer l is the data size. The decisive feature indicates the class to which each example belongs, and it is a categorical feature with values coming from an unordered finite domain. The set of all possible values of the decisive feature is denoted by $C = c_1, c_2, ..., c_m$, where each c_k, $k = 1, 2, ...m$, refers to a specified class. The predictive features are numerical, and their values are described by an n-dimensional vector, $(f(x_1), f(x_2), \cdots, f(x_n))$. The range of the vector. a subset of n-dimensional Euclidean space, is called the feature space. The j^{th} observation consists of n predictive features and the decisive features can be denoted by $(f_j(x_1), f_j(x_2), \cdots, f_j(x_n), Y_j)$, $j = 1, 2, \cdots, l$.

The following are the details of these basic concepts and the mathematical model for the classification problem.

3.1.1 Definition of Double Nonlinear Integral

Let $X = x_1, x_2, \cdots, x_n$ be a nonempty finite set of predictive features and $\mathcal{P}(X)$ be the power set of X.

A signed Fuzzy Measure defined in chapter 2 allows its value to be negative and frees of the monotonicity constraint. Thus, it is more flexible to describe the individual and joint contribution rates from the predictive features in a universal set towards some target. The classical Nonlinear Integral has been defined in the last chapter. Based on the basic concept, we proposed a new extension version as follows.

Definition 3.1 *Let μ, ν be two Fuzzy Measures on $\mathcal{P}(X)$. The double Nonlinear Integral of a function $f : X \to [-\infty, \infty]$ with respect to μ and ν is defined by*

$$\int f d\nu =< \int f d\mu, \ \int f d\nu >$$

where ν is determined after the values of μ.

The value of Double Nonlinear Integral is the coordinates of the virtual data in the 2-D space projected by the Nonlinear Integrals. In fact, the Double Nonlinear Integrals is the superposed version of the classical Nonlinear Integrals.

3.1.2 Projection based on Double Nonlinear Integrals

Based on the Nonlinear Integrals, we can build an aggregation tool that projects the feature space onto a virtual space which maybe 1-dimenstional, 2-Dimensional or more dimensional. Under the projection, each point in the feature space corresponds a value of the virtual space.

A point $(f(x_1), f(x_2), \cdots, f(x_n))$ is projected to be $\hat{Y}$, the value of the virtual variable, on a real axis through a Nonlinear Integral defined by $\hat{Y} = \int f d\mu$. Once the values of the Fuzzy Measures are determined, we can calculate virtual value $\hat{Y}$ from f. Figure 3.1 illustrates the projection from a 2-D feature space onto a real axis, L, by the Nonlinear Integral. The contours being broken are due to the nonaditivity of the Fuzzy Measure. The points on the same projection line have the same set of Fuzzy Measure values, so they can be projected onto the same location. In our model, we used the signed Fuzzy Measure in Nonlinear Integrals, so the direction of projection can be shown differently due to different signs of Fuzzy Measures.

Projection to 1-D makes the original information simple. But some useful information for classification may be left out, which leads to overlapping situation like those shown in Figure 3.2. That star in the right position represents a point misclassified by projection to 1-D. We can not classify it with the other points around it very well.

As described above, overlapping of classification on 1-D space exists in real world problems indeed. When this situation comes up, we need additional

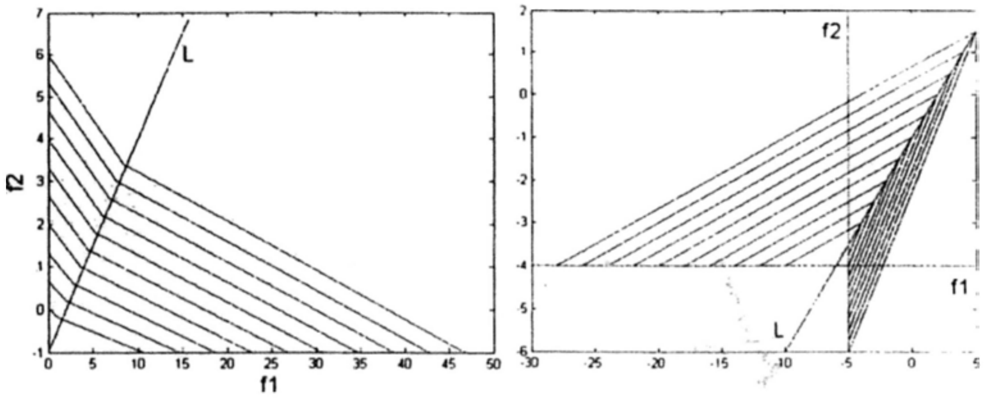


Figure 3.1: Projection onto Axis by Nonlinear Integrals

information to classify. So the 1-D space is extended to 2-D space. Similar to 1-D case that the first Fuzzy Measure μ is learned, another Fuzzy Measure ν must be introduced into the classification model. The learning process of the second Fuzzy Measure is similar to the previous one. The real axis on the 1-Dimension space is used as one axis of the 2-Dimension space. Then we learn the second Fuzzy Measure ν using GA and the value of the Nonlinear Integral respect to ν is distributed on the other dimension in 2-Dimension space. The linear classifier is used as fitness function on to classify the points projected to 2-Dimension space with the Double Nonlinear Integrals. The graphical representation of projection with the Double Nonlinear Integrals to 2-D is shown to Figure 3.3. The example case in Figure 3.2 can be projected again onto the 2-Demension space in Figure 3.3 and separated into two classes.

3.2 Classification Model by DNI

In this section, a new classification model based on Double Nonlinear Integrals will be presented. It can be viewed as a general methodology of projecting the points in the feature space onto a virtual space by Double Nonlinear

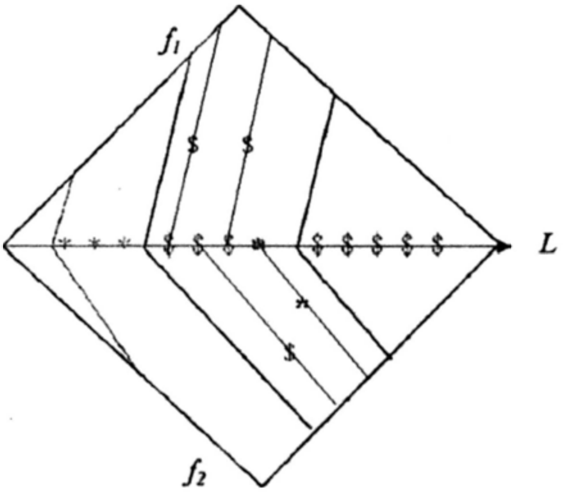


Figure 3.2: Overlapping in Projection to Real Axis

Integral, and then using a linear classifier to classify these points according to a certain criterion optimally. The parameters are obtained by using an adaptive Genetic Algorithm. Good performance of this method comes from the use of the Fuzzy Measure and the relevant Nonlinear Integral, since the nonadditivity of the Fuzzy Measure reflects the inherent interactions of the predictive features towards the discrimination of the points. In fact, each predictive feature has respective important index reflecting their amounts of contributions towards the decision. Furthermore, the global contribution of several predictive features to the feature of classification is not just the simple sum of the contribution of each feature to the decision, but may contribute cooperatively nonlinearly. A combination of the predictive features may have mutually restraining or a complementary synergy effect on their contributions towards the classification. So the Fuzzy Measure defined on the power set of all predictive features is a proper representation of the respective importance of the predictive features and the interactions among them, and a relevant Nonlinear Integral is a good fusion tool to aggregate the information coming

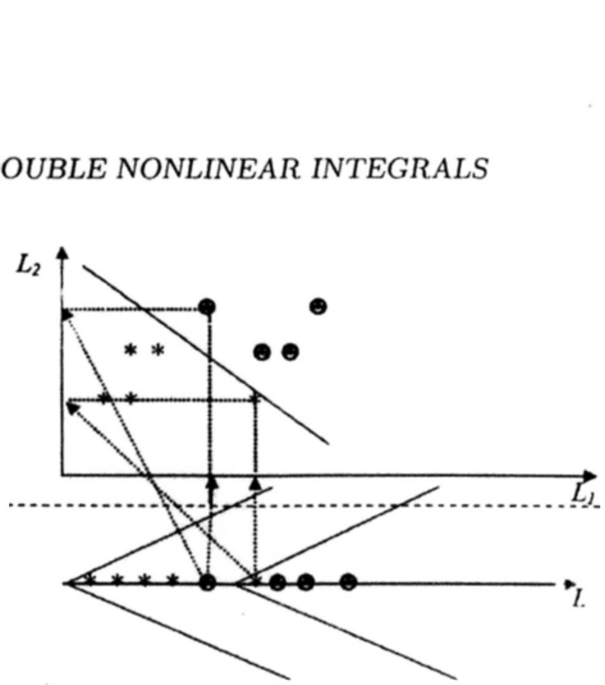
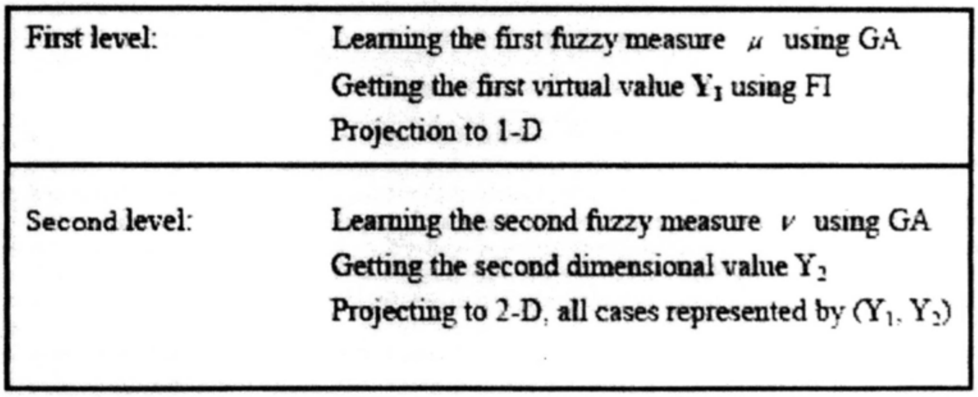


Figure 3.3: Projection to 2-D space Due to Overlapping in Projection in 1-D space

from the predictive features for the classification. The process of the new classification model is summerized in Figure 3.4. The detailed description of each part will be introduced in the subsections.



First level:	**Learning the first fuzzy measure μ using GA** **Getting the first virtual value $\mathbf{Y}_1$ using FI** **Projection to 1-D**
Second level:	**Learning the second fuzzy measure ν using GA** **Getting the second dimensional value $\mathbf{Y}_2$** **Projecting to 2-D, all cases represented by $(\mathbf{Y}_1, \mathbf{Y}_2)$**

Figure 3.4: The Process of Classification Model based on Double Nonlinear Integrals

3.2.1 GA-based Adaptive Classifier

Based on the Nonlinear Integral, we want to find an appropriate formula that projects the n-dimensional feature space onto a real axis, L, such that each point $f = (f(x_1), f(x_2), \cdots, f(x_n))$ which can be written in brief format

$f = (f_1, f_2, \cdots, f_n)$ becomes a value of the virtual variable that is optimal with respect to the classification. In such a way, each classification boundary is just a point on real axis L.

The classification process can be divided into two parts to implement.

1) The Double Nonlinear Integral based classifier depends on the Fuzzy Measures μ and ν, so determining the optimal values of μ and ν is in the first place of our work;

2) When the Fuzzy Measures μ and ν are determined, the virtual points (Y_1, Y_2) can be obtained by using Nonlinear Integral. So, we must decide how to classify these virtual points on 2-Dimension space.

The following will focus on the above problems respectively.

Here we discuss the optimization of the Fuzzy Measure μ under the criterion of minimizing the corresponding global misclassification rate which is obtained in the second part above.

In our GA model, we use a variant of original integrand f, $f' = af + b$, to substitute f, where a is a vector to shift the coordinates of data and b is a vector to scale the values of predictive features. Here, a and b attempt to balance the scales of the predictive features in case that they have different measurement units. Each chromosome represents Fuzzy Measure μ , shifting vector a and scaling vector b. A signed Fuzzy Measure equals 0 at empty set. If there is n features in training data, a chromosome has $2^n - 1 + 2n$ genes. Traditional genetic operations(e.g. crossover, mutation) are used. At each generation, for each chromosome, all variables are fixed and the virtual values of all training data are calculated using the Nonlinear Integral with respect to the signed Fuzzy Measure, so the classification function used in the second part is used as the fitness function and the misclassification rate is used as the fitness value.

In our model, projection to 2-Dimension based on Double Nonlinear Integrals is adopted for higher accuracy. So we must repeat above described optimization process. In the first step, we get the optimal value of the first Fuzzy Measure and the feature space is projected to 1-Dimension space. The real axis on the 1-Dimension space is used as one axis of the 2-Dimension space in the step 2. Then we learn the second Fuzzy Measure ν using GA and the value of the Nonlinear Integral respect to ν is distributed on the other dimention in 2-Dimension space. The linear classifier is used as fitness function on 2-Dimension space.

3.2.2 Linear Classifier for the Virtual Values

After determining the Fuzzy Measure μ and ν , shifting vector a, scaling vector b and the respective classification function from the training data in GA, original data in the n-dimensional feature space are projected onto 2-Dimension space using the Double Nonlinear Integrals. One linear classifier is needed to classify the virtual points $\hat{Y} = ((y'_{11}, y'_{12}), (y'_{21}, y'_{22}), \ldots, (y'_{l1}, y'_{l2})$. Discriminant analysis is introduced in details[53].

We use Fishers linear discriminant[54] function to perform classification in projected space. Positive and negative centroids for projected data are determined by the following formulas.

$$m_+ = \frac{\sum_{i:y_i=1} x_i}{\sum_{i:y_i=1} 1}; \quad m_- = \frac{\sum_{i:y_i=-1} x_i}{\sum_{i:y_i=-1} 1}$$

Ronald Fisher defined Scatter Matrices as

$S_\pm \equiv$ [illegible] $(x_i - m_\pm)(x_i - m_\pm)'$

$S_W = S_+ + S_-$is callec [illegible] Class Scatter Matrix. Similarly, the Between Class Scatter Ma [illegible] ned as $S_B \equiv (m_+ - m_-)(m_+ - m_-)'$.

So this result in an equivalen [illegible] ession for Fishers discriminant criterion

Table 3.1: The Example for Double Nonlinear Integral

x_1	x_2	Class
1	2	1
3	1	2
2	4	2
4	3	1

Table 3.2: The Preset Fuzzy Measure

Subsets	Value of μ	Value of ν
$\emptyset$	0	0
$\{x_1\}$	3	3
$\{x_2\}$	2	4
$\{x_1, x_2\}$	2	1

is a ratio between two quadratic forms as $J(\omega) = \dfrac{\omega' S_B \omega}{\omega' S_W \omega}$, in which ω represents the direction of the projection space, i.e. the one-dimensional space. We can solve the programming problem by maximizing $J(\omega)$. The optimal ω can be represented as $\omega = S_W^{-1} * (m_+ - m_-)$. So the Fishers discriminant function is formulated as:

$$y = \omega * (x - n_+ * m_+ - n_- * m_-)$$

in which $n_\pm$ is the sum of observations in each class respectively. Finally, a threshold needs to be fixed in order to define a complete classifier.

3.2.3 Example for classification based on the Double Nonlinear Integrals

In this section, we walk through a simple example to explain the work principle of the classification model based on the Double Nonlinear Integrals.

Given a data set with two features and two classes as shown in Table 3.1, there exists the Fuzzy Measure as Table 3.2.

The value of the classical Nonlinear Integral can be computed as :

$$y_1 = f(x_1) * \mu(x_1, x_2) + (f(x_2) - f(x_1)) * \mu(x_2) = 1 * 2 + 1 * 2 = 4$$
$$y_2 = f(x_2) * \mu(x_1, x_2) + (f(x_1) - f(x_2)) * \mu(x_1) = 1 * 2 + 2 * 3 = 8$$
$$y_3 = f(x_1) * \mu(x_1, x_2) + (f(x_2) - f(x_1)) * \mu(x_2) = 2 * 2 + 2 * 2 = 8$$
$$y_4 = f(x_2) * \mu(x_1, x_2) + (f(x_1) - f(x_2)) * \mu(x_1) = 3 * 2 + 1 * 3 = 9$$

Then the visual value by projecting is drawn in Figure 3.5. Obviously, the data can not be classified by one point into two classes. So we need to introduce the sencond set of the Fuzzy Measures ν shown in Table 3.2 to perform the second projection. The second set of values by projectiong can be computed in the same method.

$$y_1 = f(x_1) * \mu(x_1, x_2) + (f(x_2) - f(x_1)) * \mu(x_2) = 1 * 1 + 1 * 4 = 5$$
$$y_2 = f(x_2) * \mu(x_1, x_2) + (f(x_1) - f(x_2)) * \mu(x_1) = 1 * 1 + 2 * 3 = 7$$
$$y_3 = f(x_1) * \mu(x_1, x_2) + (f(x_2) - f(x_1)) * \mu(x_2) = 2 * 1 + 2 * 4 = 10$$
$$y_4 = f(x_2) * \mu(x_1, x_2) + (f(x_1) - f(x_2)) * \mu(x_1) = 3 * 1 + 1 * 3 = 6$$

The projection to 1-D space by the classical Nonlinear Integral is transferred into 2-D space by the Double Nonlinear Integral. The transfermation process is shown in Figure 3.5. The star points stand for class 1 and the dot points stand for class 2. We can not classify the four points into two classes by one point in 1-D space shown in top part of Figure 3.5. After using Double Nonlinear Integral the points in the 1-D space are streched and scattered in 2-D space so that the data can be classified correctly by a straight line. For real life problems, the two sets of Fuzzy Measure will be learned using GA algorithm introduced in the above section.

3.3 Experiments

In this section, we present the results of classification model based on projection with the Double Nonlinear Integral to classify two kinds of datasets. One

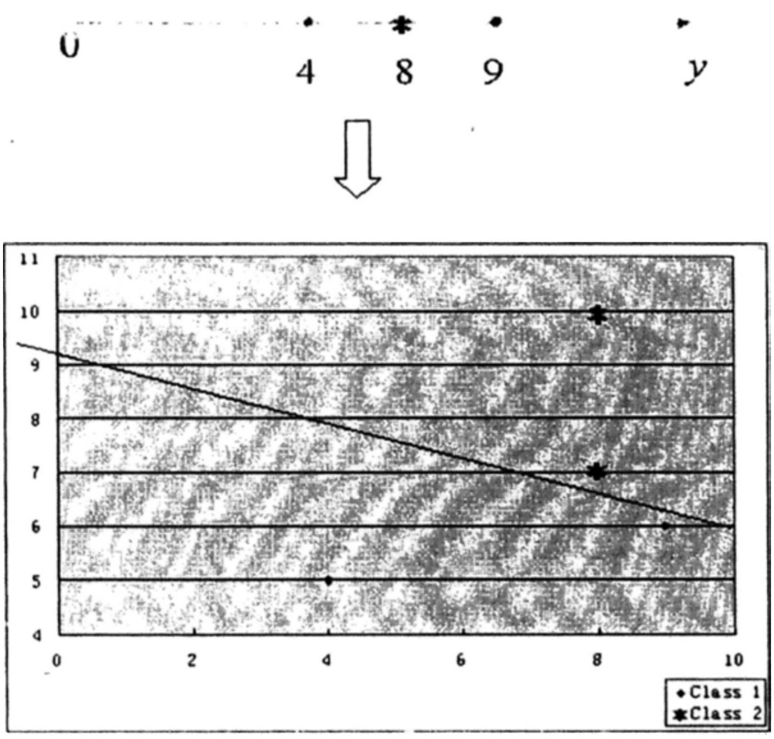


Figure 3.5: Transfermation from 1-D Space to 2-D

kind is synthetic datasets which contain 100 cases and 200 cases respectively with two classes and show ying-yang distribution as Figure 3.6. We apply new algrithom to the synthetic data to evaluate the improved performance of Double Nonlinear Integral. The other kind of datasets are benchmark problem selected from UCI[55]. The information about data are listed in Table 3.3 which is used for comparing the Double Nonlinear Integral with the classical methods. The brief description of benchmark datasets are given as follows.

Heart Disease(Heart): This dataset has 76 raw features, only 14 of them are actually used. The "goal" field refers to the presence of heart disease in the patient. The names and social security numbers of the patients were recently removed from the database, replaced with dummy values. The actual data used in our experiments has been processed which contain 270

Table 3.3: Description of Datasets

Datasets	Examples	features	Classes
Syn_Data1	100	2	2
Syn_Data2	200	2	2
Heart	270	13	2
Pima	768	7	2
Wdbc	569	30	2
Breast-cancer-winson	699	9	2
Echocardiogram	132	13	2

examples.

Pima-indians-diabetes (Pima): This dataset contains 768 examples. Each sample represents a patient who may show symptoms of diabetes is described by 8 features, which are: 1) number of times pregnant, 2) plasma glucose concentration, 3) diastolic blood pressure, 4) triceps skin fold thickness, 5) two-hour serum insulin, 6) body mass index, 7) diabetes pedigree function, and 8) age. There are 500 samples from patients who do not have diabetes and 268 samples from patients who are known to have diabetes.

Wisconsin Diagnostic beast cancer (Wdbc): This dataset contains 569 samples with 32 features which has ID, diagnosis and 30 real-valued input features. The ID feature is removed and diagnosis is used as class. Class distribution is 357 samples for benign and 212 samples for malignant.

Breast Cancer: This breast cancer databases was obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg. It contains 669 samples which are partitioned into two classes, 458 benign samples and 241 malignant samples. Each sample is described by 9 features: a) clump thickness, b) uniformity of cell size, c) uniformity of cell shape, d) marginal adhesion, e) single epithelial cell size, f) bare nuclei, g) bland chromatin, h) normal nucleoli, and i) mitoses. In this database, the values of the sixth feature in 16 samples are missing. We have ignored these 16 samples when conducting experiments.

Echocardiogram Data: The problem addressed by past researchers was to predict from the other variables whether or not the patient will survive at least one year. It contains 132 instances with 13 features which are all numeric-valued. They are 1) survival, 2) still-alive, 3) age-at-heart-attack, 4) pericardial-effusion, 5) fractional-shortening, 6) epss, 7) lvdd, 8) wall-motion-score, 9) wall-motion-index, 10) mult, 11) name, 12) group and 13) alive-at-1.

For implementing our new classifier based on the Double Nonlinear Integrals, GA tool in Matlab v7.2 Programming is called combining with Fishers discriminant function. In our experiments, all parameters in GA are set with the default values. We design the generation limit which is 100 as the stopping criteria.

We adopt 10-fold cross validation method to make sure that the testing data can cover the whole dataset. That means that we randomly break the data into 10 sets of size of $N/10$, train on 9 sets and test on 1, and each 10-fold experiment is repeated 10 times and the mean result is recorded. After ten iterations, all data are used for testing and the average can be computed to evaluate the performance of the classifier based on the Double Nonlinear Integrals.

Table 3.4 shows that the result of the classical Nonlinear Integral and the Double Nonlinear Integral for synthetic data. We can see that the Double Nonlienear Integral has improved the accuracy compared to the classical Nonlinear Integral, especially for dataset 2 which has more cases. Figure 3.7 illustrates the classification situation for the dataset with 200 cases. The left one is the first projection to 1-D space by the classical Nonlinear Integral. Apparently, many star points scatter into the dot ones. After projecting onto

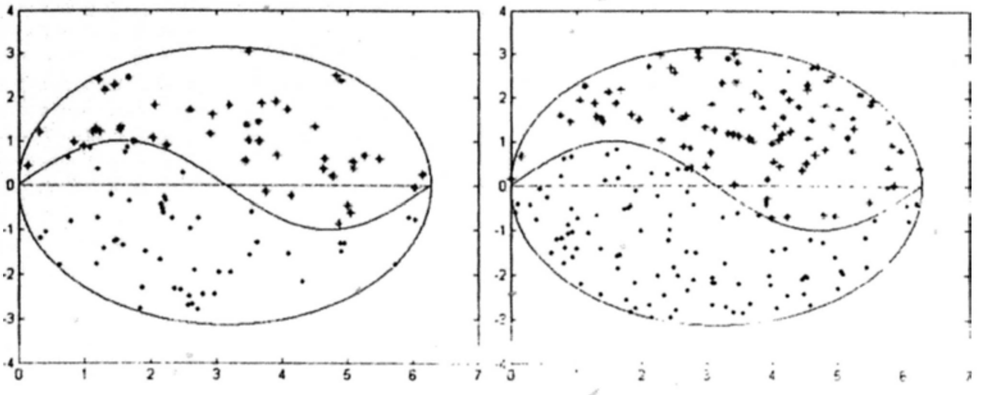


Figure 3.6: The Synthetic Data Distribution

2-D space by the Double Nonlinear Integral, all data in the two classes are streched and classified efficiently by a straight line. The error is decreased greatly.

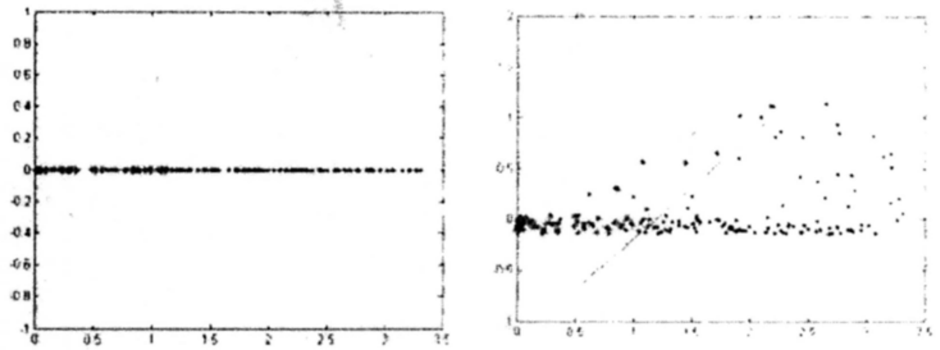


Figure 3.7: The Graphical Representation of Classification for the Synthetic Dataset with 200 Cases

Table 3.5 show the results of Projection with the Double Nonlinear Integrals to 2-D compared with the classical methods which include Decision Tree(DT)[9], Support Vector Machine(SVM)[11], Naïve Bayes(NB)[12], Nueral Network(NN) [10]. The best results are highlighted in bold. We can see that our new algorithm has higher accuracy than the classical methods for most cases and comparable in other cases.

Table 3.4: Comparison Results between 2-D Projection and 1-D Projection

Datasets Algorithms	Dataset1(100 cases)		Dataset2(200 cases)	
	Training accuracy	Testing accuracy	Training accuracy	Testing accuracy
Projection to 1-D	0.987	0.987	0.964	0.965
Projection to 2-D	0.987	0.987	**0.994**	**0.990**

Table 3.5: Comparison Results between 2-D Projection and Classical Methods

Datasets	Algorithms	DT	SVM	NB	NN	DNIC
Heart	Train accuracy	0.857	0.801	0.850	0.850	**0.867**
	Test accuracy	**0.847**	0.790	0.837	0.843	0.844
Pima	Train accuracy	**0.783**	0.770	0.776	0.782	0.782
	Test accuracy	0.747	0.762	**0.770**	0.765	0.752
Wdbc	Train accuracy	0.922	0.925	0.918	0.922	**0.930**
	Test accuracy	0.896	0.922	0.916	0.915	**0.958**
Breast-Cancer-winson	Train accuracy	0.962	0.958	0.968	0.959	**0.964**
	Test accuracy	0.941	0.954	**0.957**	0.956	0.956
EchocardiogramR	Train accuracy	0.941	0.908	0.938	**0.946**	0.907
	Test accuracy	0.881	0.888	0.925	**0.928**	0.886

Note: NIC stands for Classical Nonlinear Integrals; DNIC stands for Double Nonlinear Integrals; and NB stands for Naïve Bayes.

3.4 Chapter Summary

In this chapter, a new classification model based on projection with the Double Nonlinear Integrals has been proposed. This method has good performance based on the Double Nonlinear Integral with the signed Fuzzy Measure, because the nonadditivity of the Fuzzy Measure reflects the importance of the predictive features, as well as their inherent interactions. Projection to 2-Dimension based on the Double Nonlinear Integrals enhances the performance on classification due to solving the intersection situations in projection onto 1-Dimension space. However, the computation complexity has also increased linearly, which is acceptable.

In future work, we may compare our new model with more classical methods to evaluate its performance and extend projection to 2-D based on the Nonlinear Integrals to more dimensional space if necessary so that better performance can be obtained.

Chapter 4

Nonlinear Integrals with Polynomial Kernel

Nonlinear Integrals(NIs) are useful integration tools. It can get a set of virtual values by projecting original data to a virtual space for classification purpose using Nonlinear Integrals. The classical Nonlinear Integrals implement projection along a line with respect to the features. But in many cases the linear projection cannot achieve good performance for classification or regression due to the limitation of the integrand. The linear function used for the integrand is just a special type of functions with respect to the features. In this chapter, we propose a Nonlinear Integrals with Polynomial Kernel(NIPK). A polynomial function with respect to the features is used as the integrand of Nonlinear Integrals. It enables the projection to be along different types of curves to the virtual space, so that the virtual values gotten by Nonlinear Integrals can be better regularized and have higher separation power for classification.

We use Genetic Algorithms (GAs) in Matlab to learn the Fuzzy Measures so that a larger solution space can be searched.

To test the capability of the NIPK, we apply it to classification on several benchmark datasets and a Bioinformatics project described in Chapter 6. Experiments show that there is evident improvement on performance for the NIPK compared to the classical NIs.

This chapter is organized as follows:

Section 4.1 gives a brief introduction about this chapter. In section 4.2, the basic concepts related to Fuzzy Measures and Nonlinear Integrals are introduced. We extend the integrand from classical functions to polynomial kernels and establish the corresponding Nonlinear Integrals with Polynomial Kernel based model. Then the main algorithm of the Nonlinear Integral with Polynomial Kernel for classification is presented in section 4.3. The experimental background and results are shown in section 4.4 with detailed analyses. Finally, some conclusions are given.

4.1 Introduction

Nonlinear Integrals are known to have good results on classification and regression despite of the large computational complexity. Since Fuzzy Measure was first introduced by Sugeno[17], many versions of Nonlinear Integrals with respect to Fuzzy Measures were proposed by researchers and applied to classification and regression on real world data, e.g. Choquet, Sugeno, Twofold, t-conorm integral [36, 37, 38, 39]. In these methods, the Nonlinear Integrals are used as confidence fusion tools. Given an object $X = \{x_1, x_2, ..., x_n\}$, for each class c_k, $k = 1, 2, ...m$, a Fuzzy Measure is needed to fuse the n degrees of confidence for statement : X belongs to class C based on the value of each x_i, $i = 1, 2, ...n$. So m Fuzzy Measures are used and the values of Fuzzy Measures are needed to be determined. Moreover, these methods need a large number of training data with multiple classes. It has high time and space complexity.

Unlike the methods above, another method called WCIPP (Weighted-Choquet-Integral based Projection Pursuit) use a weighted Choquet Integral as a projection tool [26]. In WCIPP, only one Fuzzy Measure defined on the power set of the set of all features is used to describe the importance of each feature as well as their interactions[13, 16, 40] towards the classification of the records. The classical classification problem in n-dimensional space is transformed to a one-dimensional space problem through the optimal projection based on Nonlinear Integrals. We used a generalized WCIPP with respect to the Signed Fuzzy Measure in previous research. The Signed Fuzzy Measure can describe the interaction and contribution of features for better decision. f represents the value of each feature which will be used as component in integrand. The integrand is represented by $f' = af + b$ which is a transformation of f and the Fuzzy Measure is extended to the Signed Fuzzy Measure. This part will be described in detail in the following sections. But there is a limitation for the generalized WCIPP. Integrand is just one special linear type of integrand with respect to predictive features. So we propose a formal generalization of the classical integrand, $f' = af + b$, which is linear. A brief description is given below.

In this chapter, we use the polynomial kernel instead of the linear function in the above Nonlinear Integral with Polynomial Kernel as a nonlinear integrand to describe the projection path. This can project the original data in a high dimensional space onto a linear virtual space along different curves according to the degree of the polynomial integrand. As a result, the virtual data may be separated more easily and accurately due to the polynomial effects which have equivalent power of creating flexibly curve boundary surfaces. Additionally, when the feature number is very large, the computation complexity of Nonlinear Integrals will be unacceptable. So we use feature selection as preprocessing to reduce the number of features and lower the complexity, which enables the application of Nonlinear Integrals to be extended to more realistic problems.

4.2 Basic concepts

In classification, we are given a data set consisting of L example records. called training set, where each record contains the value of a decisive feature, Y, and the value of predictive features $X = \{x_1, x_2, ..., x_n\}$. The positive integer L is the data size. The decisive feature indicates the class to which each example belongs, and it has categorical values coming from an unordered finite domain. The set of all possible values of the decisive feature is denoted by $C = c_1, c_2, ..., c_m$, where each c_k, $k = 1, 2, ..., m$, refers to a specified class. The predictive features are numerical, and their values are described by an n-dimensional vector, $(f(x_1), f(x_2), \cdots, f(x_n))$. The range of the vector, a subset of n-dimensional Euclidean space, is called the feature space. The j^{th} observation consists of n predictive features and one decisive feature, which can be denoted by $(f_j(x_1), f_j(x_2), \cdots, f_j(x_n), Y_j)$. $j = 1, 2, \cdots, l$. Before introducing the model, we give out the fundamental concepts as follows.

4.2.1 Fuzzy Measure and Nonlinear Integral

Let $X = x_1, x_2, \cdots, x_n$, be a nonempty finite set of features and $\mathcal{P}(X)$ be the power set of X. The Signed Fuzzy Measure and Nonlinear Integral with the Signed Fuzzy Measure have been defined in Chapter 2. We would not repeat them. Here we just give out simple example to explain these definitions.

Example 4.1 *Let* $X = \{x_1, x_2, x_3\}$. *Set function* $\mu : \mathcal{P}(X) \to (-\infty, \infty]$ *is* *given as*

$$\mu(E) = \begin{cases} 0 & if\ E = \emptyset \\ 2 & if E = \{x_1\} \\ -3 & if\ E = \{x_2\} \\ -1 & if\ E = \{x_1, x_2\} \\ 5 & if\ E = \{x_3\} \\ 4 & if\ E = \{x_1, x_3\} \\ -2 & if\ E = \{x_2, x_3\} \\ 3 & if\ E = X \end{cases}$$

Then μ is a signed efficiency measure on $\mathcal{P}(X)$.

A Signed Fuzzy Measure allows its value to be negative and free from the monotonicity constraint. Thus, it is more flexible to describe the individual and joint contribution rates from the predictive features in a universal set towards some targets.

4.2.2 Nonlinear Integrals with Polynomial Kernel

Definition 4.1 *Let μ be a non-monotonic Fuzzy Measure on X and f be a nonnegative function. The Polynomial Nonlinear Integral with respect to μ is given by*

$$(pc) \int f d\mu = \sum_{j=1}^{n} [f'^{p}(x_i) - f'^{p}(x_{i-1}))] \mu(\{x_i, x_{i+1}, ..., x_n\})$$

where $\mu(\{x_i, x_{i+1}, ..., x_n\})$ is the same as the definition of Nonlinear Integral in Chapter 2, f^p is the polynomial integrand to replace the classical linear one and p is an integer to represent the exponent for all features.

From Figure 4.1, we can see the simple graphical representation of the projection by the classical Nonlinear Integrals. In this section, we discuss the detailed situation of the projection by Nonlinear Integrals with Polynomial Kernel with different degrees of polynomial integrands. We design the polynomial integrands as $(af + b)^p$. When $p = 1$, the Nonlinear Integrals with Polynomial Kernel is consistent with the classical Nonlinear Integrals. So it can be viewed as a generalized form. For simplicity, we limit our discussions in two dimensional spaces in this section. The discussions would apply to higher dimensional feature spaces.

A. p=1

When $p = 1$, the projection axis is linear and projection contours are piecewise linear. In 2-dimensional space, the projection axis satisfies the equation . The slope of the projection axis can be positive or negative. Let us see an example for illustrating the situation with respect to the classical Fuzzy Measure.

Example 4.2 *Let $\mu_1 = 0.2, \mu_2 = 0.6, \mu_{12} = 1.0$. The other parameters are $a_1 = 1, b_1 = 4; a_2 = 2, b_2 = 6$ So the real axis L can be computed by solving equation $a_1 f + b_1 = a_2 f + b_2, a \neq 0, b \neq 0$.*

L: $f_2 = \dfrac{b_1 - b_2}{a_2} + \dfrac{a_1}{a_2} f_1 = -1 + 0.5 f_1$

The contours can be computed using the Nonlinear Integral with Polynomial Kernel defined above. When $a_1 f + b_1 < a_2 f + b_2$, the contours represented by $y = 0.4f_1 + 1.2f_2 + 5.2$ are above L. For example $f_1 = 0, f_2 = 6$ then the projection value $y = 0.4 * 0 + 1.2 * 6 + 5.2 = 12.4$. When $a_1 f + b_1 > a_2 f + b_2$. the contours represented by $y = 0.2f_1 + 1.6f_2 + 5.2$ are below L. For example $f_1 = 5, f_2 = 0$, then the projection value $y = 0.2 * 5 + 1.6 * 0 + 5.2 = 6.2$. This projection is shown in Figure 4.1.

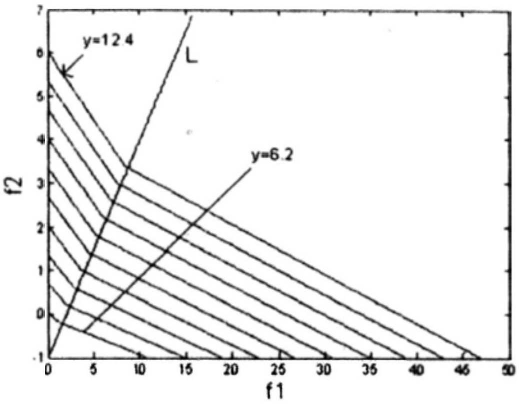


Figure 4.1: $p = 1$ with the classical Fuzzy Measure

In an extended model, we extend the Fuzzy Measure to a generalized Fuzzy Measure-the Signed Fuzzy Measure. It means the joint contribution of multiple features may not necessarily be larger than the individual's. The directions of projections can be changed according to the signs of the Signed Fuzzy Measures in the graph. When μ is nonnegative and regular, the slopes of the contours must be less than or equal to 0. The extension relaxed this restriction so that the slopes of projections can be flexibly varied from positive to negative and vice versa. This situation is explained by Example 4.3 and graphically described in Figure 4.2.

Example 4.3 *Let* $\mu_1 = -1, \mu_2 = -2, \mu_{1,2} = 3.0$ *and* $a_1 = 1, b_1 = 4; a_2 = 2, b_2 = 6$. *Then the projection axis is the same as that in Example 4.2,*

$$i.e.\ L\text{: } f_2 = \frac{(b_1 - b_2)}{a_2} + \frac{a_1}{a_2} f_1 = -1 + 0.5 f_1$$

Because the Fuzzy Measure is the Signed Fuzzy Measure which is different from previous example, the contour lines are different from those in Figure 4.1. The corresponding graph on projections by Nonlinear Integrals with re-

spect to the Signed Fuzzy Measure is shown in Figure 4.2.

For contours, above L, $y = \mu_{12} * (a_1 f_1 + b_1) + \mu_2 (a_2 f_2 + b_2) = 3f_1 - 4f_2$

For contours, below L, $y = \mu_{12} * (a_2 f_2 + b_2) + \mu_1 (a_1 f_1 + b_1) = 6f_1 - 1f_2 + 14$

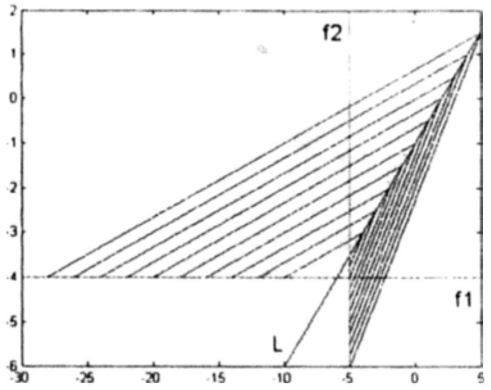


Figure 4.2: $p = 1$ with the Signed Fuzzy Measure

B. p**=2**

When $p = 2$, the polynomial integrand is represented as $(af+b)^2$. The projection axis can be computed similarly with $p = 1$ which satisfies $(a_1 f_1 + b_1)^2 = (a_2 f_2 + b_2)^2$, $a \neq 0$, $b \neq 0$. So there are two projection axes by solving above equation, i.e. $L: \ f_2 = \pm \left(\frac{b_1 - b_2}{a_2} + \frac{a_1}{a_2} f_1 \right)$. Projection contour may be parabola, hyperbola or ellipse depending on the sign of parameters. Let us see the example in Figure 4.3. The data which have $(a_1 f_1 + b_1)^2 < (a_2 f_2 + b_2)^2$ are in the areas I and III and those which have $(a_1 f_1 + b_1)^2 > (a_2 f_2 + b_2)^2$ are in the areas II and IV. The blue projection curves in areas I and III follow the function

$y = \mu_{12} * (a_1 f_1 + b_1)^2 + \mu_2 * ((a_2 f_2 + b_2)^2 - (a_1 f_1 + b_1)^2)$

$= (\mu_{12} - \mu_2) * (a_1 f_1 + b_1)^2 + \mu_2 * (a_2 f_2 + b_2)$

The red projection curves in areas II and IV follow the function

$$y = \mu_{12} * (a_2 f_2 + b_2)^2 + \mu_1 * ((a_1 f_1 + b_1)^2 - (a_2 f_2 + b_2)^2)$$
$$= (\mu_{12} - \mu_1) * (a_2 f_2 + b_2)^2 + \mu_1 * (a_1 f_1 + b_1)$$

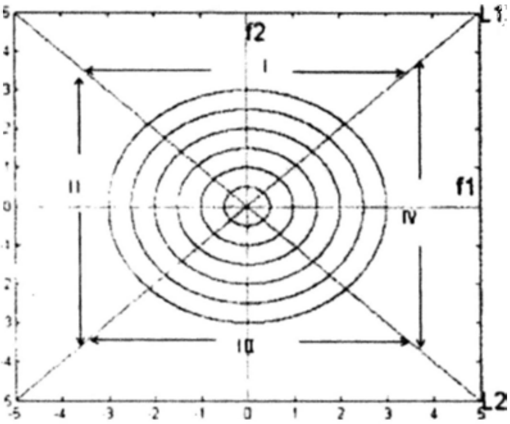


Figure 4.3: $p = 2$

C. p=3

When $p = 3$, the polynomial integrand can be represented as $(af + b)^3$. The projection axis needs to satisfy $(a_1 f_1 + b_1)^3 = (a_2 f_2 + b_2)^3$, $a \neq 0$, $b \neq 0$. Due to the odd exponent, there is only one line as in the situation of $p = 1$. The difference between $p = 1$ and $p = 3$ is in just the projection paths. The former ones are simple straight lines, but the latter ones are along the curves of the polynomial functions of degree 3. Figure 4.4 shows the representative curves.

When $p = 4$, the situation is similar to that of $p = 2$; when $p = 5$, the situation is similar to that of $p = 3$. So the detailed descriptions and figures will be skipped.

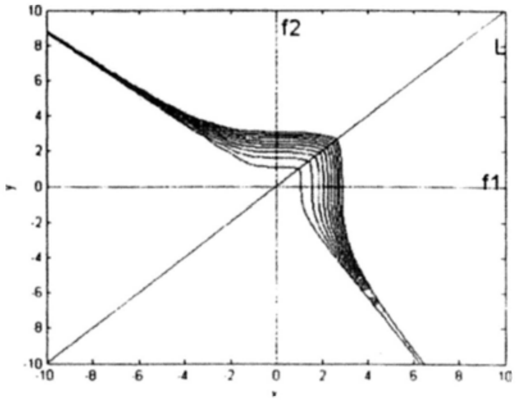


Figure 4.4: $p = 3$

4.3 Projection based on Nonlinear Integral with Polynomial Kernel for classification

Based on the Nonlinear Integral, we can build an aggregation tool that projects the feature space onto a virtual 1-dimenstional space. Under the projection, each point in the feature space becomes a value of the virtual variable.

A point $(f(x_1), f(x_2), \cdots, f(x_n))$ is projected to be Y, the value of the virtual variable, on a real axis through a Nonlinear Integral defined by $Y = \int f d\mu$. Once the values of μ are determined, we can calculate the virtual values $\hat{Y}$ from f.

Figures 4.1 and 4.2 illustrate the projection from a 2-D feature space onto a real axis, L, by the classical Nonlinear Integral. The contours being broken are due to the nonaditivity of the Fuzzy Measure. Although there are other kinds of projection axes using Nonlinear Integral with Polynomial Kernel as in Figures 4.3 and 4.4. Given the training examples, our algorithm will learn the optimal coefficients of the polynomial function by GA. We can classify the cases according to the virtual values on the axis projected by Nonlinear

Integrals.

Our approach can be divided into three steps.

Step 1. Learn the Fuzzy Measures and polynomial index using the training dataset;

Step 2. Construct a linear classifier based on the virtual data which obtained by Nonlinear Integrals with polynomial Kernel;

Step 3. Evaluate the learnt model using the testing examples.

In our GA model, a Signed Fuzzy Measure is 0 at empty set. The fitness function can be defined as:

$FitnessFun = Error$

where $Error$ is the misclassification rate for each dataset.

The pseudo code for the whole algorithm of the Nonlinear Integral with Polynomial Kernel based classifier is listed as follows:

```
--------------------------------------------------------------------------------
n: features number;
L: dataset size;
Input: training data $f_{ij}$ and $y_j, i = 1, 2, \cdots, n; j = 1, 2, \cdots, L$
Output: error $final\_e$
Begin:
    While $final\_e >$ threshold
        Learning Fuzzy Measure $\mu$, parameters $a$ and $b$, and degree $p$ in GA;
        For each data $j$
            Compute NIPK;
            Classify data $j$
        End
        $final\_e$=fitness;
    End
    Output $\mu, a, b$ and $p$;
    For a new observation
        Compute NIPK;
        Classify;
    End
    Output the $final\_e$;
End
```

4.4 Experiments and analysis

We have performed two parts of experiments. The first part is testing our algorithm on the synthetic Datasets and 3 Monkey problems: the second part is to apply Nonlinear Integral with Polynomial Kernel(NIPK) to seven real-life problems selected from UCI [55]. We compare the performance of NIPK with the classical Nonlinear Integral classifier

4.4.1 Model Building

To implement the learning algorithm of our new classifier based on NIPK, we use the GA tool in Matlab v7.2 Programming. All the parameters used in our GA for our experiments are default values. We set the generation limit to be 100 as the stopping criteria.

We adopt 10-fold cross validation method to make sure that the whole dataset can be used as testing data in turn and over-training (over-fitting) can be avoided. It means that we randomly partition the N data into 10 sets of size of $N/10$, train the model on 9 sets and test on the remaining set. We

Table 4.1: Description of Synthetic Datasets and Monk Series Datasets in Part 1

Datasets	Examples	Features	Classes
SYN1	200	2	2
SYN2	500	3	2
MONK1	556	6	2
MONK2	601	6	2
MONK3	554	6	2

then repeat the training and testing 9 times in turn and take the mean result.

4.4.2 Results and Analysis

In this part, there are two synthetic datasets, three Monk series datasets and seven datasets selected from the UCI repository [55]. The synthetic datasets have the same distribution on both sides of the ying-yang shape in 2-Dimensional and 3-Dimensional spaces and different dataset sizes of 200 (as shown in Figure 4.5) and 500 respectively. The detailed information is shown in Tables 4.1 and 4.2 respectively. Two of these datasets, breast-cancer-winson and echocardiogram, have the noisy data processed by substituting them by the most common value or mean value.

We can see that the number of features of each dataset is rather large for Nonlinear Integrals to deal with. It will take very long time to learn the Fuzzy Measures. So feature selection is a necessary step. We adopt Information Gains based ranking method to select the features for classification. We just select the top features as the subset to be used in the model, which can greatly improve the efficiency of Nonlinear Integrals because the time of learning the Signed Fuzzy Measure is reduced greatly. Actually for these datasets, we have experimented with 3, 4 and 5 features. The results show no significant difference in performance. Based on the Occam razor principle,

Table 4.2: Description of Datasets in UCI Repository

Datasets	Abbr.	Examples	Features	Classes
Heart	Heart	270	13	2
Pima	Pima	768	7	2
Wisconsin Diagnostic Breast Cancer	Wdbc	569	30	2
Wisconsin Prognostic Breast Cancer	Wpbc	699	9	2
Echocardiogram	Echo.	132	13	2
Tae	Tae	151	5	2
Sonar	Sonar	208	60	2

3 is the optimized choice for the number of features for these datasets. For consistency, the same dataset is used in other methods. The main algorithm of classification model is implemented using Matlab v7.2 [57].

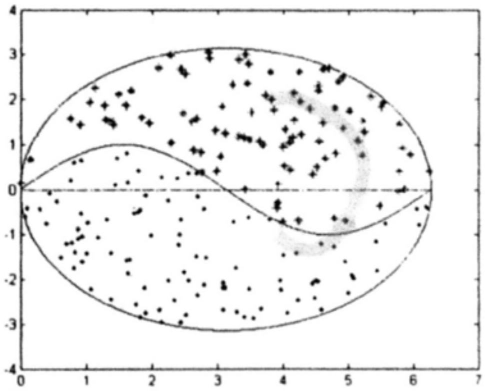


Figure 4.5: The Synthetic Data Distribution

Firstly, we test the performance of this model respectively when p equals 1 to 5. The results in each situation are shown in Tables 4.3 and 4.4. We can see the performance of the Classifier based on Nonlinear Integrals with Polynomial Kernel (NIPKC) is not necessarily the best when the index is the largest. So the accuracy is not augmented linearly as the index p is increased. It also shows that the index is not fixed for the optimal situation. It depends on the irregularity of the boundary required to separate the clusters of data

Table 4.3: The Results of NIPKC with Different Degrees for Datasets in Part1

Datasets	performance	p=1	p=2	p=3	p=4	p=5
Syn-Data1	train accuracy	**0.964**	0.959	0.954	0.952	0.947
	test accuracy	0.912	**0.935**	0.925	0.905	0.929
Syn-Data2	train accuracy	0.953	**0.958**	0.947	0.950	0.946
	test accuracy	0.932	**0.954**	0.926	0.944	0.932
Monk1	train accuracy	0.867	**0.890**	0.880	0.883	0.827
	test accuracy	0.789	0.793	0.744	**0.886**	0.797
Monk2	train accuracy	**0.720**	0.703	0.680	0.670	0.660
	test accuracy	**0.677**	0.646	0.611	0.644	0.646
Monk3	train accuracy	0.954	0.967	0.972	0.971	**0.978**
	test accuracy	0.950	0.964	0.975	0.975	**0.986**

for classifications.

Since there is no one fixed value for p, we propose to learn p together with the Fuzzy Measures. The results in each situation are shown in Tables 4.5 and 4.6 as follows. Table 4.5 gives the comparison results of NIPKC and the Classical NIC. Table 4.6 shows the performance of NIPKC compared with several classical methods, namely, Neural Network (NN), Decision Tree (DT), Naïve Bayes (NB), and Support Vector Machine (SVM).

From Table 4.5, we can see the NIPKC has better the performance than NIC in most cases, especially for complex datasets. According to the comparison of the results with the traditional methods in Table 4.6, the NIPKC has the highest accuracy for some datasets and is comparable in the others. For complex data, NIC cannot get better results than those traditional algorithms and need to be extended to NIPKC. Although maybe there is some over training for some datasets, NIPKC contains not only the characteristic of NIC which can describe the interaction of features in the contribution towards classification but has competitive classification power provided by the polynomial projection.

Table 4.4: The Results of NIPKC with Different Index for Datasets in Part2

Datasets	performance	p=1	p=2	p=3	p=4	p=5
Heart	train accuracy	0.859	**0.860**	0.854	0.844	0.844
	test accuracy	**0.856**	**0.856**	0.844	0.830	0.826
Pima	train accuracy	0.780	**0.781**	0.780	0.772	0.771
	test accuracy	0.766	**0.768**	0.753	0.751	0.733
Wdbc	train accuracy	**0.926**	0.925	0.925	0.923	0.923
	test accuracy	0.905	**0.919**	0.910	**0.919**	0.914
Wpbc	train accuracy	0.953	**0.958**	0.947	0.950	0.946
	test accuracy	0.932	**0.954**	0.926	0.944	0.932
Echo.	train accuracy	**0.948**	0.943	0.932	0.919	0.918
	test accuracy	0.895	0.917	**0.932**	0.894	0.894
Tae	train accuracy	0.769	0.762	0.765	**0.771**	0.758
	test accuracy	0.674	**0.709**	0.666	0.658	0.628
Sonar	train accuracy	0.801	0.802	**0.807**	0.798	0.796
	test accuracy	0.739	0.727	0.750	0.751	**0.765**

4.5 Chapter Summary and Future Work

In this chapter, we have alleviated the limitation of the classical Nonlinear Integrals on integrand and introduce the polynomial function as the nonlinear integrand. This generalization has extended the projection from linear lines to various shapes of curves which can handle more complicated data classification. We can see the accuracy of classification model does not necessarily increase with the degree of the polynomial. So we also need to learn the optimal degree of the Nonlinear Integrals with Polynomial Kernel in the training process. By extending the Nonlinear Integral to higher degrees, we call the new model: the Nonlinear Integral with Polynomial Kernel. Moreover, the complexity of the Nonlinear Integral with Polynomial Kernel Classifier is not greater than that of the classical Classifier.

We have applied Nonlinear Integral with Polynomial Kernel to Benchmark

Data . All results showed that Nonlinear Integral with Polynomial Kernel can get better or comparable classification performance than the classical Nonlinear Integral and the other traditional classifiers.

Table 4.5: The Comparison Results between NIC and NIPKC

Datasets	performance	NIC	NIPKC
Syn1	train accuracy	**0.964**	0.963
	test accuracy	0.912	**0.929**
Syn2	train accuracy	0.953	**0.958**
	test accuracy	0.932	**0.942**
Monk1	train accuracy	0.867	**0.871**
	test accuracy	0.789	**0.873**
Monk2	train accuracy	0.720	**0.722**
	test accuracy	0.677	**0.711**
Monk3	train accuracy	0.954	**0.977**
	test accuracy	0.950	**0.969**
Heart	train accuracy	0.859	**0.860**
	test accuracy	**0.856**	0.851
Pima	train accuracy	0.780	**0.781**
	test accuracy	0.766	**0.768**
Wdbc	train accuracy	0.926	**0.929**
	test accuracy	0.905	**0.912**
Wpbc	train accuracy	0.953	**0.969**
	test accuracy	0.932	**0.957**
Echo	train accuracy	0.948	**0.951**
	test accuracy	0.895	**0.911**
Tae	train accuracy	**0.769**	0.768
	test accuracy	0.674	**0.693**
Sonar	train accuracy	**0.801**	0.783
	test accuracy	0.739	**0.768**

Table 4.6: The Comparison Results of Several Methods

Datasets	performance	NIPKC	NIC	DT	NB	NN	SVM
Heart	train accuracy	**0.860**	0.859	0.857	0.801	0.850	0.850
	test accuracy	0.851	**0.856**	0.847	0.790	0.837	0.843
Pima	train accuracy	0.781	0.780	**0.783**	0.770	0.776	0.782
	test accuracy	0.768	0.766	0.747	0.762	**0.770**	0.765
Wdbc	train accuracy	**0.929**	0.926	0.922	0.925	0.918	0.922
	test accuracy	0.912	0.905	0.896	**0.922**	0.916	0.915
Wpbc	train accuracy	**0.969**	0.953	0.962	0.958	0.968	0.959
	test accuracy	**0.957**	0.932	0.941	0.954	**0.957**	0.956
Echo.	train accuracy	**0.951**	0.948	0.941	0.908	0.938	0.946
	test accuracy	0.911	0.895	0.881	0.888	0.925	**0.928**
Tae	train accuracy	0.768	0.769	0.737	0.705	0.729	**0.824**
	test accuracy	**0.693**	0.674	0.594	0.671	0.685	0.653
Sonar	train accuracy	0.783	0.801	0.774	0.747	0.793	**0.819**
	test accuracy	**0.768**	0.739	0.716	0.695	0.759	0.743

Chapter 5

Upper and Lower Nonlinear Integrals

A new nonlinear multi-regression model based on a pair of extreme Nonlinear Integrals, the Upper and Lower Nonlinear Integrals, is established in this chapter. A data set of predictive features and the relevant objective feature is required for estimating the regression coefficients. Due to the nonadditivity of the model, a Genetic Algorithm or other soft computing technique should be adopted to search for the optimized solution in the regression problem. Applying such a nonlinear multi-regression model, an interval prediction for the value of the objective feature can be made once a new observation of predictive features is available. We apply our model on synthetic data and weather problem. The results testify the performance of the multi-regression based on the Upper and Lower Nonlinear Integrals.

To show such a new multi-regression model, this chapter is arranged as follows. Section 5.1 gives an introduction. In Section 5.2, the concept of the Signed Fuzzy Measure is recalled and an overview of Nonlinear Integrals defined on finite sets are given. The new nonlinear multi-regression model based on the Upper and Lower Nonlinear Integrals is described in Section 5.3. This section also shows the multi-objective optimization method for the regression model to make an interval prediction. In Section 5.4, simulation on synthetic data and application on weather data are presented. Finally, some conclusions are given in Section 5.5.

5.1 Introduction

The classical aggregation tool in information fusion is the weighted sum. It is a linear model. Using this linear model needs a basic assumption: there is no interaction among the contributions from predictive features towards the objective feature such that the joint contribution from a group of predictive features is just the simple sum of contributions from each individual feature in the group. However, in many real problems, such an interaction cannot be ignored. Fortunately, a non-classical mathematical tool, nonadditive set function, has been successfully used to describe the above-mentioned interaction in information fusion and data mining [16, 17, 37, 63, 65, 68, 70]. Replacing the classical weighted sum which is the classical Lebesque integral defined on a certain finite set, the aggregation tool should be a certain type of abstract integrals, which is generally nonlinear due to the nonadditivity of the involved set function. In this case, as one of the inverse problems of the information fusion, the multi-regression should be established in terms of Nonlinear Integrals. Such an idea has been recently realized in some works where the Choquet integral is adopted [24, 68, 69, 71].

The different type of Nonlinear Integrals corresponds to the different partition rules that describe the different coordination scheme. Generally, when a data set is available, people do not know the concrete coordination. Thus, a new question appears: why should we choose the Choquet integral in the nonlinear multi-regression model, since there are various types of Nonlinear Integrals, such as the upper integral and the lower integral [67, 70]. The upper and the lower integrals form an extreme pair, the maximum and the minimum, among all Nonlinear Integrals [67]. That is to say, any type of Nonlinear Integrals is between the upper integral and the lower integral, in terms of their values. Hence, a natural idea is to use the upper integral and the lower integral to "control" the observed values of the objective feature and, then, establish a new type of nonlinear multi-regression.

5.2 Upper and Lower Nonlinear Integrals

In section 5.2.1, we review some related concepts described in previous chapters. We introduce the Upper and Lower Nonlinear Integrals in section 5.2.2. In the following sections, we will use some examples to help understanding the theory.

5.2.1 Signed Fuzzy Measures

Denote the set of all information sources by X and call it the universal set. Then $(X, \mathcal{P}(X))$ is a measurable space, where $\mathcal{P}(X)$ is the power set of X. In almost all real problems, the universal set is finite. For example, in any database, the number of features is always finite. Thus, throughout this chapter we assume that $X = \{x_1, x_2, ..., x_n\}$, where each x_i, $i = 1, 2, ..., n$, is an feature. In a multi-regression problem, $x_1, x_2, ..., x_n$ are called predictive features. They are usually numerical. There is another numerical feature Y called the objective feature in the database as a fusion target. The observation value of Y is denoted by y generally. While in classification problems,

$x_1, x_2, \ldots, x_n$ are called predictive features and Y is the objective feature that is categorical.

Definition 5.1 *Any set function, $\mu : \mathcal{P}(X) \to (-\infty, \infty]$, is called a signed efficient measure if $\mu(\emptyset) = 0$, where $\emptyset$ is the empty set. Any nonnegative signed efficient measure is called efficient measure. Any monotone efficient measure is called monotone measure.*

The Signed Fuzzy Measure, efficiency measure, and monotone measure are also called the Signed Fuzzy Measure, generalized Fuzzy Measure, and Fuzzy Measure respectively [16, 17, 23, 63, 65, 70].

Example 5.1 *Let $X = \{x_1, x_2, x_3\}$. Set function $\mu : \mathcal{P}(X) \to (-\infty, \infty]$ is given as*

$$\mu(E) = \begin{cases} 0 \quad if\ E = \emptyset \\ 2 \quad if E = \{x_1\} \\ -3 \quad if\ E = \{x_2\} \\ -1 \quad if\ E = \{x_1, x_2\} \\ 5 \quad if\ E = \{x_3\} \\ 4 \quad if\ E = \{x_1, x_3\} \\ -2 \quad if\ E = \{x_2, x_3\} \\ 3 \quad if\ E = X \end{cases}$$

Then μ is a Signed Fuzzy Measure on $\mathcal{P}(X)$.

Any Signed Fuzzy Measure μ can be decomposed as the difference of two efficiency measures, $\mu = \mu^+ - \mu^-$.

Definition 5.2 *Let $\mu : \mathcal{P}(X) \to (-\infty, \infty]$ be a Signed Fuzzy Measure. $\mu = \mu^+ + \mu^-$ is called the reduced decomposition of μ if both μ^+ and μ^-*

are efficiency measures on $\mathcal{P}(X)$ and $\mu^+(E) \cdot \mu^-(E) = 0$ for every $E \subseteq X$.

The pair of μ^+ and μ^- is also simply called the reduced decomposition of μ, where μ^+ is called the positive part of μ, while μ^- is called the negative part of μ. For any given Signed Fuzzy Measure, the reduced decomposition is unique. Equality $\mu^+(E) \cdot \mu^-(E) = 0$ means that at least one of $\mu^+(E)$ and $\mu^-(E) = 0$ must be zero. In fact,

$$\mu^+(E) = \begin{cases} \mu(E) & if\ \mu(E) > 0 \\ 0 & otherwise \end{cases}$$

and

$$\mu^-(E) = \begin{cases} -\mu(E) & if\ \mu(E) < 0 \\ 0 & otherwise \end{cases}$$

The reduced decomposition, μ^+ and μ^-, of a Signed Fuzzy Measure μ is its minimal decomposition in the following sense:

(1) μ^+ and μ^- is a decomposition of μ;

(2) if ν^+ and ν^- is a decomposition of μ, then $\mu^+ \leq \nu^+$ and $\mu^- \leq \nu^-$.

Example 5.2 *Consider a Signed Fuzzy Measure shown in Example 1, the reduced decomposition of μ is*

$$\mu^+(E) = \begin{cases} 0 & if\ E = \emptyset \\ 2 & if E = \{x_1\} \\ 0 & if\ E = \{x_2\} \\ 0 & if\ E = \{x_1, x_2\} \\ 5 & if\ E = \{x_3\} \\ 4 & if\ E = \{x_1, x_3\} \\ 0 & if\ E = \{x_2, x_3\} \\ 3 & if\ E = X \end{cases}$$

and

$$\mu^-(E) = \begin{cases} 0 & if\ E = \emptyset \\ 0 & ifE = \{x_1\} \\ 3 & if\ E = \{x_2\} \\ 1 & if\ E = \{x_1, x_2\} \\ 0 & if\ E = \{x_3\} \\ 0 & if\ E = \{x_1, x_3\} \\ 2 & if\ E = \{x_2, x_3\} \\ 0 & if\ E = X \end{cases}$$

The following example explains that a Signed Fuzzy Measure can be used for describing the interaction among the contributions from the information sources towards a certain target. A similar example of workers appeared in [23] first.

Example 5.3 *Let $X = \{x_1, x_2, x_3\}$ be the set of three workers. They are hired for manufacturing a certain type of toys. Their individual and joint efficiency $\mu : \mathcal{P}(X) \to [0, \infty]$ is given as follows:*

$$\mu(E) = \begin{cases} 0 & if\ E = \emptyset \\ 5 & if E = \{x_1\} \\ 3 & if\ E = \{x_2\} \\ 10 & if\ E = \{x_1, x_2\} \\ 4 & if\ E = \{x_3\} \\ 4 & if\ E = \{x_1. x_3\} \\ 6 & if\ E = \{x_2, x_3\} \\ 9 & if\ E = X \end{cases}$$

Set function μ is an efficiency measure that describes the interaction among the contributions from individual workers towards the target, the total number of toys manufactured by these workers. In this example, $\mu(\{x_1, x_2\}) > \mu(\{x_1\}) + \mu(\{x_2\})$ shows that workers x_1 and x_2 cooperate well and, therefore, the interaction between their contributions is mutually promoting. While $\mu(\{x_1, x_3\}) < \mu(\{x_1\}) + \mu(\{x_3\})$, even $\mu(\{x_1, x_3\}) < \mu(\{x_1\})$, shows that workers x_1 and x_3 cooperate very bad and the interaction between their contributions is mutually inhibitive. This set function is not monotonic.

5.2.2 Nonlinear Integrals on finite sets

Let $X = \{x_1, x_2, ..., x_n\}$, $\mu : \mathcal{P}(X) \to (-\infty, \infty]$ and $\nu : \mathcal{P}(X) \to (-\infty, \infty]$ be Signed Fuzzy Measures, and $f : X \to (-\infty, \infty]$ and $f : X \to (-\infty, \infty]$ be nonnegative functions.

Definition 5.3 *[67] A set function $\pi : \mathcal{P}(X) - \{\emptyset\} \to (-\infty, \infty]$ is called a partition of f if*

$$f(x) = \sum_{A|x \in A \subseteq X} \pi(A) \quad \forall x \in X.$$

Definition 5.4 *[67] Each type of integrals with respect to μ is characterized by a rule r , by which, for any given f., a partition π of f can be obtained. Regarding both π and μ as $(2^n - 1)$-dimensional vectors, the value of the integral of f under rule r, denoted by $(r)\int f d\mu$ and is called the r-integral of f with respect to μ, is the inner product of π and μ, that is, $(r)\int f d\mu = \pi \cdot \mu$, where (r) indicates the type of integral.*

The r-integrals have the following properties that the classical Lebesgue integral has:

(R1)$(r)\int cfd\mu = c(r)\int fd\mu \quad \forall c \in [0, \infty)$;

(R2)$(r)\int fd\mu \leq (r)\int gd\mu \quad if\ \mu \geq 0\ and\ f \leq g$;

(R3)$(r)\int fd\mu \leq (r)\int fd\nu \quad if\ 0 \leq \mu \leq \nu$.

Properties (R2) and (R3) are called the monotonicity of the r-integral.

However, the r-integrals given in Definition 5.4 are usually nonlinear with respect to the integrand, that is, equality

$$(r)\int(f+g)d\mu = (r)\int fd\mu + (r)\int gd\mu$$

may not hold. A counterexample can be found in [67].

There are infinitely many types of r-integrals, among them, the Lebesgue-like integral (simply called the Lebesgue integral [64], if there is no confusion) and the Choquet integral [15, 16, 23, 24, 63, 70] is a pair of extreme Nonlinear Integrals in terms of the manner of coordination among features [67]. Another pair of extreme Nonlinear Integrals, which is in terms of the amount of integration value, is the upper and the lower integrals. They are given in Definition 5.5.

Definition 5.5 *[67, 70] The upper integral of f with respect to μ, denoted by*

$(U)\int f d\mu$, is defined by

$$(U)\int f d\mu = sup\{\sum_{j=1}^{2^n-1} a_j \cdot \mu(A_j) | \sum_{j=1}^{2^n-1} a_j \chi_{A_j=f}\} \qquad (1)$$

while the lower integral of f with respect to μ, denoted by $(L)\int f d\mu$, is defined by

$$(L)\int f d\mu = inf\{\sum_{j=1}^{2^n-1} a_j \cdot \mu(A_j) | \sum_{j=1}^{2^n-1} a_j \chi_{A_j=f}\} \qquad (2)$$

where $a_j \geq 0$ and $A_j = \bigcup_{i:j_i=1} \{x_i\}$ if j is expressed in binary digits as $j_n j_{n-1} \cdots j_1$ for every $j = 1, 2, ..., 2^n - 1$.

The calculation of the upper and the lower integrals is just the procedure of solving the following linear programming problems respectively:

$$\text{Maximize (or Minimize) } z = \sum_{j=1}^{2^n-1} a_j \cdot \mu_j$$

$$\text{subject to } \sum_{j=1}^{2^n-1} a_j \chi_{A_j}(x_i) = f(x_i),\ i = 1, 2, ..., n \qquad (3)$$

$$a_j \geq 0,\ j = 1, 2, ..., 2^n - 1,$$

where $\mu_j = \mu(A_j)$ for $j = 1, 2, ..., 2^n - 1$, and $a_1, a_2, ..., a_{2^n-1}$ are unknown parameters. The above n constraints can also be rewritten as

$$\sum_{j:x\in A_j\subseteq X} a_j = f(x) \quad \forall x \in X.$$

Defining set function $\pi : \mathcal{P}(X) \to (-\infty, \infty]$ by $\pi(A_j) = \pi(j) = a_j$ for $j = 1, 2, ..., 2^n - 1$, we may see that π is a partition of f. So, the upper and the lower integrals are just two special types of r-integral. Its corresponding partitioning rules are "divide the integrand in such a way that the integration value is maximized" and "divide the integrand in such a way that the integration value is minimized" respectively.

From the definition directly, we may have a property of the upper and the lower integrals.

Property (UL1) For any r-integral, $(L)\int f d\mu \leq (r)\int f d\mu \leq (U)\int f d\mu$

Example 5.4 *We use the data given in Example 5.3. Assume that these workers are hired for 3, 8, and 5 hours respectively, that is,*

$$f(x) = \begin{cases} 3 & if\ x = x_1 \\ 8 & if\ x = x_2 \\ 5 & if\ x = x_3 \end{cases}$$

in a certain day. If there is an excellent manger who can well arrange their work, then the number of manufactured toys in this day may be, from formula (1), as many as

$(U) \int f d\mu = 3 \times \mu(\{x_1, x_2\}) + 5 \times \mu(\{x_2\}) + 5 \times \mu(\{x_3\}) = 65$

Anyway, these three workers can manufacture at least

$(L) \int f d\mu = 3 \times \mu(\{x_1, x_3\}) + 2 \times \mu(\{x_2, x_3\}) + 6 \times \mu(\{x_2\}) = 42$

toys. As for the other two types of Nonlinear Integrals, the Lebesgue integral and the Choquet integral, of function f, we have

$\int f d\mu = 3 \times \mu(\{x_1\}) + 8 \times \mu(\{x_2\}) + 5 \times \mu(\{x_3\}) = 65$

and

$(C) \int f d\mu = 3 \times \mu(\{x_1, x_2, x_3\}) + 2 \times \mu(\{x_2, x_3\}) + 3 \times \mu(\{x_3\}) = 48.$

These results also confirm Property (UL1).

In general, we can use the difference between the upper and the lower integrals, $(P) \int 1 d\mu - (L) \int 1 d\mu$ to indicate the uncertainty associated with Signed Fuzzy Measure μ. As a special case, when μ is additive, then any r-integral, including the upper integral, the lower integral, and the Choquet integral, coincides with the Lebesgue integral, and is linear with respect to the integrand. Hence, the uncertainty associated with any additive measure is 0.

5.3 Multi-regression based on the Upper and Lower Nonlinear Integrals

One of the Nonlinear Integrals, the Choquet integral, has been applied as the aggregation tool in multi-regressions [24, 68, 69, 71]. From Chapter 2, we know that the Choquet integral is just one of the Nonlinear Integrals. It has a very special cooperation manner, maximal cooperation, among predictive features. Hence, a new question appears: for a given data set, why such a cooperation manner is suitable? That is, why should we adopt the Choquet integral but not some other types of the Nonlinear Integrals? Indeed, we have no sufficient reason to defend the choice of the Choquet integral, which exists in our previous works. If there is no additional information focused on the cooperation manner of the predictive features, we are really unable to say what the most suitable one is among all of r-integrals. Fortunately, any r-integrals are dominated by the upper integral and the lower integral. Hence, we may use them to roughly represent any actual r-integral and to describe the possible error.

5.3.1 Model Building

Before establishing a model of multi-regression based on the Upper and Lower Nonlinear Integrals, we need to recall the concept of interval number.

Definition 5.6 *Any closed interval $[a, b]$ is called an interval number, where a and b are real numbers satisfying a.*

Any real number a is a special case of interval numbers. It can be rewritten as $[a, a]$.

To consider the regression problem, assume that a complete data set has the form

$$\begin{array}{cccc} x_1 & x_2 & ,\ldots, x_n & Y \\ f_{11} & f_{12} & ,\ldots, f_{1n} & y_1 \\ f_{21} & f_{22} & ,\ldots, f_{2n} & y_2 \\ f_{l1} & f_{l2} & ,\ldots, f_{ln} & y_l \end{array} \qquad (4)$$

is available, where Y is the target feature, the j^{th} row

$$f_{j1} \quad f_{j2} \quad ,\ldots, f_{jn} \quad y_j$$

is the j^{th} observation of features $x_1, x_2, \ldots, x_n$ and Y, $j = 1, 2, \ldots, l$. Positive integer l represents the size of the data, and should be much larger than n. Each observation of $x_1, x_2, \ldots, x_n$ can be regarded as a function $f : X \to (-\infty, \infty]$. Thus, the j^{th} observation of $x_1, x_2, \ldots, x_n$ is denoted by f_j, and we write $f_{ji} = f_j(x_i)$, $i = 1, 2, \ldots, n$ for $j = 1, 2, \ldots, l$.

Now we assume $f_j \geq 0$ $\forall i = 1, 2, \ldots, n$ first. To describe how objective feature Y depends on predictive features $x_1, x_2, \ldots, x_n$, a new multi-regression model can be expressed as follows:

$Y = c + [(L) \int (a + bf) d\mu, (U) \int (a + bf) d\mu] + N(0, \sigma^2)$,

where c is a constant, $N(0, \sigma^2)$ is a normally distributed random perturbation with mean 0 and variance σ^2, and $[(L) \int (a + bf) d\mu, (U) \int (a + bf) d\mu]$ is an interval number, in which functions $a : X \to (-\infty, \infty]$ and $b : X \to (-\infty, \infty]$ can be expressed as n-dimensional vectors, i.e., $a = (a_1, a_2, \ldots, a_n)$ and $b = (b_1, b_2, \ldots, b_n)$, and are used to balance the various phases and scales of predictive features, while μ is a Signed Fuzzy Measure. Functions a and b should satisfy the following constraints:

$$a_i \geq 0 \ for \ i = 1, 2, \ldots, n, \ with \ \min_{1 \leq i \leq n} a_i = 0$$
$$0 \leq b_i \leq 1 \ for \ i = 1, 2, \ldots, n, \ \max_{1 \leq i \leq n} b_i = 1.$$

In this regression model, constant c, vectors a and b, and Signed Fuzzy Measure μ are unknown parameters and are called regression coefficients. The model is nonlinear generally. Once the data set shown in (4) with sufficient large size is available, these parameters can be optimally determined by minimizing a vector of objectives including the squared errors expressed as

$$Min\ E, \qquad E = [e_1, e_2],$$

where $e_1 = \sum\limits_{j=1}^{l} e_{1j}^2$,

$$e_{1j} = \begin{cases} 0 & if\ y_j - c \in [(L)\int(a+bf)d\mu, (U)\int(a+bf)d\mu] \\ min(|y_j - c - (L)\int(a+bf)d\mu|, |y_j - c - (U)\int(a+bf)d\mu| & otherwise \end{cases}$$

that describes the random error, and $e_2 = \sum\limits_{j=1}^{l} e_{2j}^2$

$$e_{2j} = (U)\int(a+bf_j)d\mu - (L)\int(a+bf_j)d\mu$$

that describes the uncertainty carried by the Signed Fuzzy Measure μ. After determining all regression coefficients, once a new observation f is available, the prediction for the target Y is an interval number

$\hat{Y} = [c + (L)\int(a+bf)d\mu, c + (U)\int(a+bf)d\mu]$.

The percentage of wrongly predicting can be estimated by

$$e_3 = \frac{\sum\limits_{j=1}^{l} e_{3j}}{l},$$

where

$$e_{3j} = \begin{cases} 1 & if\ y_j - c \in [(L)\int(a+bf)d\mu, (U)\int(a+bf)d\mu] \\ 0 & otherwise \end{cases}$$

In case the assumption $f_j \geq 0\ \forall i = 1, 2, ..., n$ is not true, denoting $m(f) = \min\limits_{j=1,2,...,l;i=1,2,...,n} f_{ji}$, we may replace f_{ji} by $f_{ji} - m(f)$ for each $j = 1, 2, ..., l$ and $i = 1, 2, ..., n$.

5.3.2 Multiobjective Optimization using Genetic Algorithm for the multi-regression model

The problem presented in Section 5.3.1 is a nonlinear multiobjective optimization problem with respect to unknown regression coefficients which is NP hard. Using an analytical and/or algebraic method to solve such a multiobjective optimization problem is difficult. We have to use a soft computing technique to search for a vector of objectives which must be trade off in some way. The Genetic Algorithm is one of the feasible methods. Genetic Algorithms are well suited to multiobjective optimization problems as they are fundamentally based on biological processes which are inherently suitable for multiobjective problem since it provides multi-model optimal and suboptimal solutions [72]. This method based on Genetic Algorithm can search for a set of Pareto optima for multiobjective minimization.

In the genetic algorithm, each parameter is presented by a gene denoted by g with a subscript. Let genes $g_1, g_2, ..., g_n$ present $\mu_1, \mu_2, ..., \mu_{2^n-1}$ respectively, where μ_k denotes $\mu(A)$, in which $A = \bigcup_{k_i=1} x_i$, $k = 1, 2, ..., 2^n - 1$. Genes $g_1, g_2, ..., g_{2^n-1}$ form a chromosome.

In this chapter, we used the GA with constraints function in Matlab. In our GA model, a Signed Fuzzy Measure is 0 at empty set. If there are n features in training data, a chromosome has $2^n - 1$ genes which are set to random real values randomly at initialization. Genetic operations used are default ones. At each generation, for each chromosome, all variables are fixed and the objective values of all training data are calculated using Nonlinear Integral.

The pseudo code for the whole algorithm of the multi-regression based on the Upper and Lower Nonlinear Integrals is as follows:

```
$n$: features number;
$l$: dataset size;
Input: training data $f_{ij}$ and $y_j$ $i = 1, 2, ..., n$; $j = 1, 2, ..., l$
Output: error $e$
Begin:
    While $final\_e >$ threshold
        Learning Fuzzy Measures $\mu$ in GA;
        For each data $j$
            Compute Upper Integral;
            Compute Lower Integral;
            Compute error $e_{1j}$ and $e_{2j}$
        End
        $final\_e$=fitness;
    End
    Output the Fuzzy Measures $\mu$;
    For a new observation
            Compute the Upper Integral;
            Compute the Lower Integral;
            Compute the error $e_1$ and $e_2$
    End
  End
```

5.4 Simulations and Applications

In this section, we describe the implementation of simulations and applications respectively. The simulations are operated on four synthetic datasets. The applications are implemented on the weather problems. All experiments are coded in Matlab. For each dataset, we adopt 10-fold cross validation method to make sure that the whole dataset can be used as testing data in turn and over-training (over-fitting) can be avoided.

5.4.1 Simulation for synthetic datasets

In the explanatory experiments, we use synthetic datasets to evaluate the performance of the proposed Upper and Lower Nonlinear Integrals as a regression tool. We generated synthetic datasets by Choquet Integral and random distribution respectively which include 200 individuals with 2 predictive features and 500 individuals with 3 predictive features. For keeping the uniqueness, we preset the constant $c = 0$, all shifting parameters a as 0 and all scaling parameters b as 1. The parameters are preset as $\mu = \{1.000, 5.000, 3.000\}$ for datasets with 2 features and $\mu = \{1.000, 5.000, 3.000, \ 2.000, 4.000, 3.000, 6.000\}$ for datasets with 3 features. We search for the optimal solutions of Fuzzy Measures with the Upper and Lower Nonlinear Integrals based Regression Model. The results are listed in Tables 5.1-5.4 which are sets of Pareto optima. e_1 is the distance between the points to the nearest bound, upper or lower one; e_2 is the range of the Upper and Lower Nonlinear Integrals; and e_3 is the percentage of points being outside the upper and lower bounds. We set the parameter controlling the size of optimal solutions as 10.

All solutions are sorted by e_1 in the training results. The top one solution for each dataset marked in bold is closest to the preset values. The

Table 5.1: The Results for Random Data with 2 Features Random Data(Original Values for μ_1, μ_2 and μ_3 are 1.0, 5.0, 3.0 respectively)

μ_1	μ_2	μ_3	Training			Testing		
			e_1	e_2	e_3	e_1	e_2	e_3
0.998	**5.015**	**2.816**	**0.000**	**1.058**	**0.000**	**0.000**	**0.980**	**0.000**
1.121	4.864	3.132	0.000	0.944	0.117	0.000	0.875	0.050
0.973	4.890	3.628	0.003	0.740	0.272	0.001	0.685	0.150
1.139	4.377	4.103	0.033	0.468	0.578	0.034	0.433	0.400
1.072	4.162	4.214	0.065	0.338	0.728	0.070	0.313	0.550
1.197	4.318	5.023	0.132	0.163	0.833	0.075	0.151	0.900
0.877	4.462	5.163	0.146	0.058	0.950	0.073	0.054	0.900
0.846	4.462	5.288	0.172	0.006	0.983	0.089	0.006	1.000
1.300	4.046	5.347	0.227	0.000	1.000	0.153	0.000	1.000
1.300	4.046	5.347	0.227	0.000	1.000	0.153	0.000	1.000

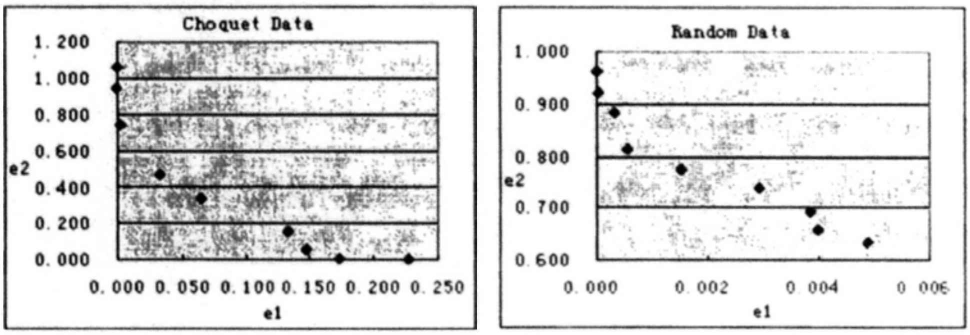


Figure 5.1: The Set of Noninferior Solutions for Datasets with 2 Features

corresponding e_1 in training and testing are zero, which means all data are included in the range of the Upper and Lower Nonlinear Integrals. So the percentage of points being outside the upper and lower bounds is zero too. Each e_2 is close to the true distance between the Upper and Lower Nonlinear Integrals. If recovering the original model is needed, the top ones are acceptable. On the other hand, the bottom solution has the smallest e_2 in training and testing. If we ignore e_1 and e_3, the upper and lower bounds nearly coincide with each other to form a point which can fit the original data well.

Table 5.2: The Results for Choquet Data with 2 Features(Original Values for μ_1, μ_2 and μ_3 are 1.0, 5.0, 3.0 respectively)

μ_1	μ_2	μ_3	Training			Testing		
			e_1	e_2	e_3	e_1	e_2	e_3
0.996	**4.968**	**2.989**	**0.000**	**0.961**	**0.000**	**0.000**	**1.125**	**0.000**
0.996	4.968	2.989	0.000	0.961	0.000	0.000	1.125	0.000
0.931	4.911	2.997	0.000	0.919	0.011	0.000	1.076	0.000
0.746	4.968	2.989	0.000	0.880	0.022	0.000	1.031	0.000
0.726	4.812	3.020	0.001	0.813	0.150	0.000	0.952	0.050
0.735	4.790	3.125	0.002	0.775	0.367	0.001	0.908	0.550
0.746	4.718	3.176	0.003	0.739	0.406	0.002	0.865	0.600
0.647	4.520	3.020	0.004	0.693	0.194	0.002	0.812	0.200
0.580	4.577	3.112	0.004	0.660	0.311	0.002	0.774	0.400
0.474	4.613	3.117	0.005	0.636	0.322	0.002	0.745	0.450

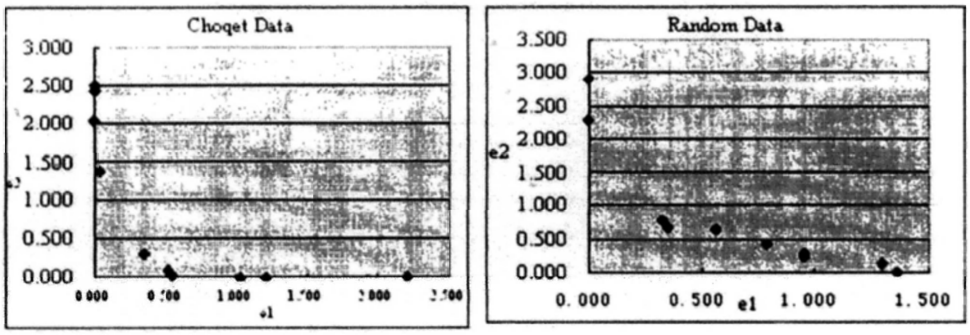


Figure 5.2: The Set of Noninferior Solutions for Datasets with 3 Features

The corresponding Fuzzy Measure is far away from the initial set value. This situation is different from the true model. In this case, the regression model based on the Upper and Lower Nonlinear Integrals is just performing like the classical regression model.

From the tables. we can see that the values of Fuzzy Measures for data with 2 features are closer to the real ones than those for datasets with 3 features. It means that the complexity increases as the number of features ascends so that it is difficult to find the real model with a large feature size.

Table 5.3: The Results for Random Data with 3 Features(Original Values for $\mu_1 - \mu_7$ are 1.000,5.000,3.000,2.000,4.000,3.000,6.000 respectively)

μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7	Training			Testing		
							e_1	e_2	e_3	e_1	e_2	e_3
0.873	**4.980**	**3.191**	**2.025**	**5.354**	**2.091**	**6.129**	**0.000**	**2.901**	**0.000**	**0.000**	**2.452**	**0.000**
1.018	4.586	3.744	1.908	5.615	3.318	6.096	0.003	2.276	0.104	0.002	1.984	0.120
1.934	3.465	5.513	1.767	5.221	4.523	6.778	0.327	0.794	0.707	0.285	0.741	0.740
2.064	3.431	5.376	2.085	4.872	4.347	7.636	0.353	0.678	0.764	0.353	0.585	0.780
2.375	3.567	5.561	2.055	5.141	4.603	7.330	0.565	0.652	0.793	0.550	0.564	0.740
2.625	3.304	6.011	1.796	5.236	4.703	7.347	0.790	0.435	0.884	0.702	0.405	0.840
2.812	3.216	6.315	1.930	5.054	4.680	7.484	0.954	0.310	0.904	0.882	0.273	0.860
2.869	3.150	5.874	1.769	5.072	4.677	7.844	0.954	0.254	0.922	0.868	0.232	0.940
3.209	3.197	6.364	1.824	4.972	4.635	8.098	1.296	0.149	0.949	1.179	0.121	0.900
3.266	2.925	6.197	1.726	4.993	4.646	7.912	1.357	0.004	1.000	1.239	0.003	1.000

The Pareto frontier plot gathers all optima for two kinds of datasets as shown in Figures 5.1 and 5.2. The horizontal axis represents e_1 and vertical axis represents e_2. According to the requirements, the appropriate solutions can be selected from the set of points on the charts.

5.4.2 Application on weather data

Weather prediction has been one of the most challenging problems around the world for more than half a century. Besides its practical value in meteorology, the weather prediction is also a typical "unbiased" problem of time series forecasting in scientific research. The weather records include temperature ranges, humidity, cloud density or rainfall of the days before the day being considered. The weather prediction has been traditionally based on the numerical models[106]. Such a simulation often requires intensive computations involving complex differential equations and computational algorithms. Besides that, the accuracy of the prediction is bounded by the adoption of incomplete boundary conditions, model assumptions, and numerical instabilities. Those difficulties in the weather prediction bring challenges not only to

Table 5.4: The Results for Choquet Data with 3 Features(Original Values for $\mu_1 - \mu_7$ are 1.000,5.000,3.000,2.000,4.000,3.000,6.000 respectively)

μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7	Training			Testing		
							e_1	e_2	e_3	e_1	e_2	e_3
1.028	**4.678**	**2.941**	**1.806**	**5.686**	**3.015**	**5.810**	**0.000**	**2.502**	**0.000**	**0.000**	**2.568**	**0.000**
1.005	4.635	2.976	1.820	5.596	3.077	5.806	0.000	2.430	0.004	0.000	2.495	0.000
1.015	4.623	2.945	1.806	4.716	3.326	5.974	0.001	2.037	0.040	0.000	2.062	0.020
0.756	3.611	3.432	2.514	5.006	4.193	7.638	0.044	1.374	0.262	0.016	1.430	0.180
0.819	3.874	4.657	2.158	3.351	5.515	6.911	0.345	0.311	0.804	0.253	0.327	0.760
0.919	3.883	4.601	2.133	2.903	5.946	6.752	0.520	0.097	0.947	0.400	0.102	0.960
0.758	3.859	4.640	2.136	2.882	5.978	6.760	0.544	0.015	0.996	0.423	0.016	1.000
0.419	2.536	2.955	3.985	4.400	6.530	6.962	1.022	0.008	0.998	0.715	0.009	0.980
0.415	2.580	2.987	4.139	4.553	6.716	7.125	1.200	0.003	0.996	0.881	0.003	1.000
0.512	2.959	3.469	4.568	5.082	7.526	8.042	2.196	0.002	1.000	1.859	0.002	1.000

Note: e_1-the distance of the points to the nearest bound, upper or lower one

e_2 -the range of the Upper and Lower Nonlinear Integral

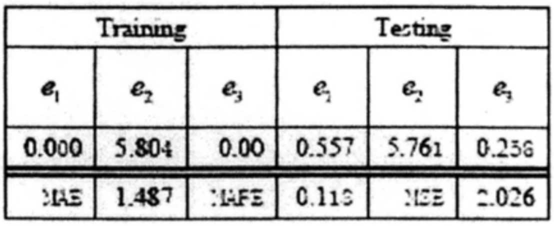
e_3 -the percentage of points being outside the upper and lower bounds

meteorologists but also to researchers in data mining. Diverse methods and models[107, 108, 109] have been proposed to solve the weather prediction problem.

We have collected the weather data of Tokyo in the period from January to December, 2007. We take the highest and the lowest temperature of each day as the predictive features to predict the mean temperature of the following day.

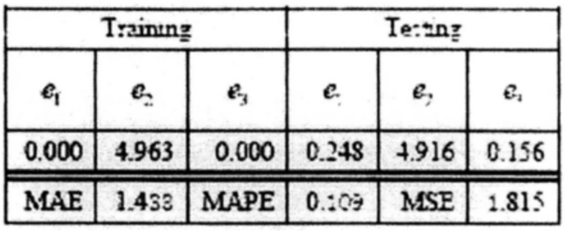
We use a non-overlapping window method to evaluate our model. This method divides all data into many folds with size k. $k - 1$ observations are used as training set and the last one is the testing datum. We have tried many values from 5 to 15 for k and select $k = 10$ to get an optimal result. We select the best result of each fold according to the fitness value. The averages of the best evaluation values for all folds are listed in the Tables. We use the mean of the Upper and Lower bounds predicted to compare with the mean temperature of the next day. Three evaluation criteria for evaluating the fitting

Table 5.5: The Result on Weather Problem with 2 Predictive Features



Training			Testing		
e_1	e_2	e_3	e_1	e_2	e_3
0.000	5.804	0.00	0.557	5.761	0.256
MAE	1.487	MAPE	0.113	MSE	2.026

Table 5.6: The Result on Weather Problem with 3 Predictive Features



Training			Testing		
e_1	e_2	e_3	e_1	e_2	e_3
0.000	4.963	0.000	0.248	4.916	0.156
MAE	1.433	MAPE	0.109	MSE	1.815

performance are computed which are the Mean Absolute Error(MAE), the Mean Absolute Percentage Error (MAPE) and the Mean Square Error(MSE) respectively. In statistics, the Mean Absolute Error(MAE) is a quantity used to measure how close forecasts or predictions are to the eventual outcomes. The MAE is given by $MAE = \frac{1}{n}\sum_{i=1}^{n}|e_i|$. The Mean Absolute Percentage Error(MAPE) is measure of accuracy in a fitted time series value in statistics, specifically trending. It usually expresses accuracy as a percentage. It can be denoted by $MAPE = \frac{1}{n}\sum_{i=1}^{n}|A_t - F_t|$, where A_t is the actual mean temperature and F_t is the mean value of the Upper and Lower Nonlinear Integrals. The mean square error(MSE) of an estimator is one of many ways to quantify the difference between an estimator and the true value of the quantity being estimated, which is denoted by $MSE = \frac{1}{n+1}\sum_{i=1}^{n}(A_t - F_t)^2$.

The results are acceptable as shown in Tables 5.5 and 5.6.

Figures 5.3 and 5.4 show the trends of predicting with the Upper and Lower Nonlinear Integrals using 2 predictive features and 3 predictive features respectively. We can see the most original points are covered within the

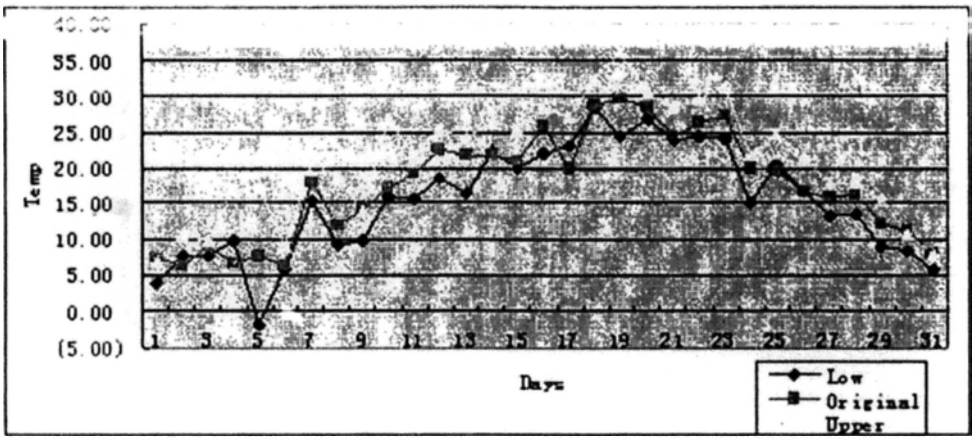


Figure 5.3: The Trend of Predicting on the Testing Data with 2 Predictive Features

predicted range. The fitting figures for mean values are drawn in Figures 5.5 and 5.6 for different predictive features' number respectively. The regression using the mean values of the Upper and Lower Nonlinear Integrals to fit the mean temperatures is satisfactory. The Mean Absolute Error(MAE) is 1.433 and the Mean Absolute Percentage Error(MAPE) is 0.109 in the best situation. This performance is good enough for a byproduct of the Upper and Lower Nonlinear Integral based multi-regression model.

5.5 Chapter Summary

The Upper and Lower Nonlinear Integrals(ULNI) form an extreme pair, the maximum and the minimum, among all Nonlinear Integrals. That is to say, any type of Nonlinear Integrals is between the Upper Integral and the Lower Integral, in terms of their values. Hence, a new type of nonlinear multi-regression is established to use the upper integral and the lower integral to "control" the observed values of the objective feature. Due to the nonadditivity of the Signed Fuzzy Measures, a Genetic Algorithm has been adopted

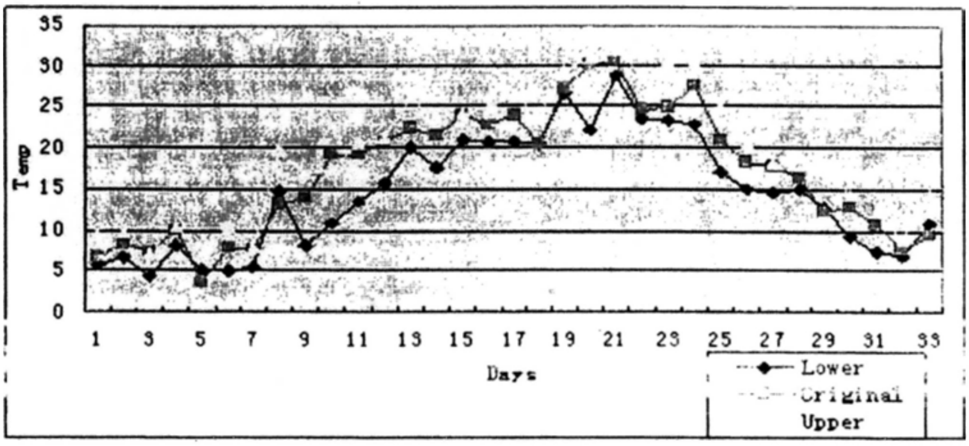


Figure 5.4: The Trend of Predicting on the Testing Data with 3 Predictive Features

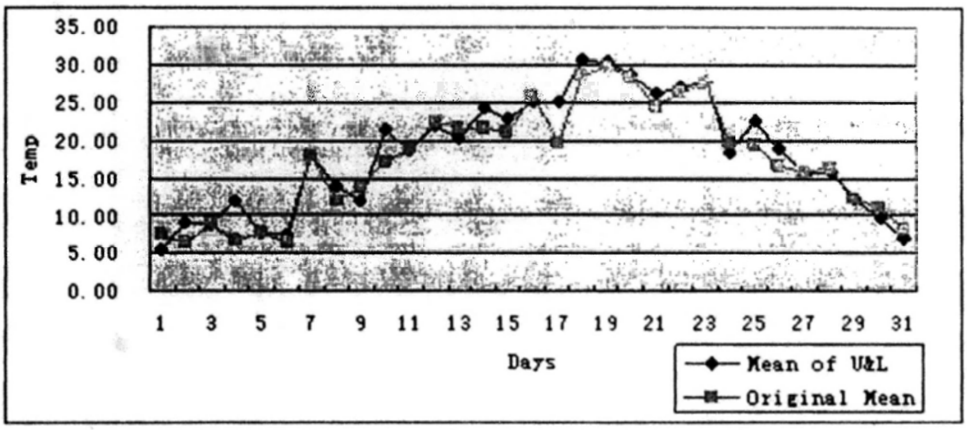


Figure 5.5: The Fitting Curve for the Mean Values with 2 Predictive Features

to learn the optimized parameters in the regression problem. Applying such a nonlinear multi-regression model, an interval prediction for the value of the objective feature can be made once a new observation of predictive features is available.

In this model, the total error consists of two types of errors. One is the squared error which represents the distance of the objective value from the nearest bound of the Upper and Lower Nonlinear Integrals to describe the random error. The other is the distance between the Upper and Lower

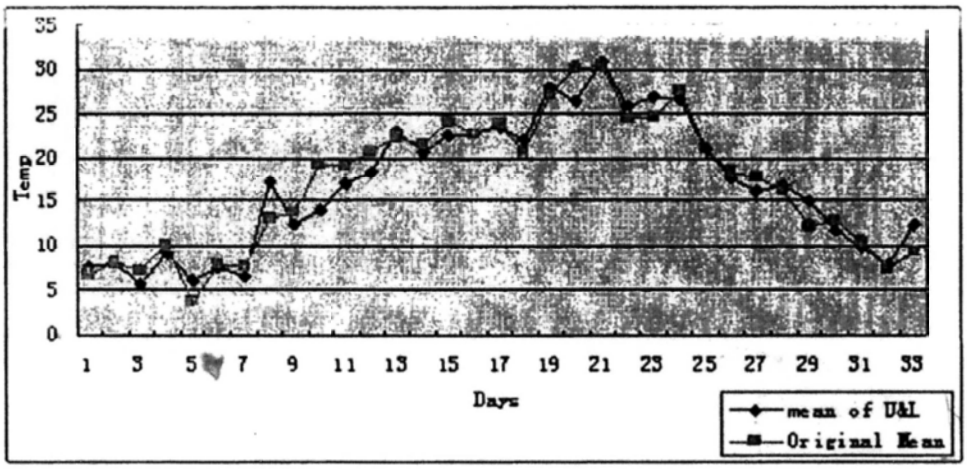


Figure 5.6: The Fitting Curve for the Mean Values with 3 Predictive Features

Nonlinear Integrals to describe the uncertainty carried by the Signed Fuzzy Measure.

We have created a set of synthetic datasets for evaluating the performance of the Multi-regression based on the Upper and Lower Nonlinear Integrals. In the experiments, we have used the multiobjective optimization method using GA to implement the Upper and Lower Nonlinear Integrals based regression model. The results have shown a set of optima for the model with the Upper and Lower Nonlinear Integrals. We can select different solutions from the pareto frontier obtained by the multi-criteria optimization for different criteria to satisfy different situations. The model is applied to a weather problem. The change trend of mean temperature of next day is predicted satisfactorily. The performance of our new model is testified by simulations and applications.

Chapter 6

Applications on Bioinformatics

Extraction of meaningful information from large experimental datasets is a key element in bioinformatics research. One of the challenges is to identify genomic markers in Hepatitis B Virus (HBV) that are associated with HCC (liver cancer) development by comparing the complete genomic sequences of HBV among patients with HCC and those without.

In this study, a data mining framework which includes molecular evolution analysis, clustering, feature selection, classifier learning and classification, is introduced. Our research group has collected HBV DNA sequences, either genotype B or C, from over 200 patients specifically for this project. In the molecular evolution analysis and clustering, three subgroups have been identified in genotype C and a clustering method has been developed to separate the subgroups. In the feature selection process, potential markers are selected based on Information Gains for further classifier learning. Then meaningful rules are learnt by our algorithm called the Rule Learning which is based on Evolutionary Algorithm. Also, two new classification methods based on the Nonlinear Integral and the Generalized Nonlinear Integrals have been developed. Good performance of these methods come from the use of the Fuzzy

Measure and the relevant Nonlinear Integral. The nonadditivity of the Fuzzy Measure reflects the importance of the predictive features as well as their interactions. These classifiers give explicit information on the importance of the individual mutated sites and their interactions towards the classification (potential causes to liver cancer in our case). A thorough comparison study of these three methods with existing methods is detailed.

For genotype B, genotype C subgroups C1, C2 and C3, important mutation markers (sites) have been found respectively. These two classification methods have been applied to classify never-seen-before examples for validation. The results show that the classification methods have more than 70% accuracy and 80% sensitivity for most datasets, which are considered high as an initial scanning method for liver cancer diagnosis.

This chapter is organized as follows. Section 6.1 gives out the problem statement. Section 6.2 describes the data mining framework which includes the new rule learning, the Nonlinear Integral classification model and the Nonlinear Integrals based classification model in detail. All the methods and datasets used in this project are detailed in Section 6.3. The experimental results and the comparative studies are presented in Section 6.4. Section 6.5 concludes with the summary and the discussion of some directions for future work.

6.1 Problem Statement

In Asia, infection of Hepatitis B virus (HBV) is a major health problem. At least 10% of the Chinese population (120 million people) are HBV carriers, and up to 25% of HBV carriers will die as a result of HBV-related complications including liver cirrhosis and hepatocellular carcinoma (HCC), i.e., liver cancer. Chronic infection by the hepatitis B virus (HBV) causes an increased risk of hepatocellular carcinoma (HCC) by more than 100 fold[101]. The relationship between HBV genotype and viral mutation with hepatocarcinogenesis is controversial. A case control study from Taiwan suggested that genotype C HBV is more closely associated with cirrhosis and HCC in those who are older than 50 years; whereas genotype B more common in patients with HCC aged less than 50 years [102]. Our previous cohort study of 426 cases of chronic hepatitis *B* patients also reviewed a higher risk of HCC and liver cirrhosis in genotype C infection [73]. On the other hand, reports from Japan and China did not confirm the higher malignant potential of genotype C HBV [103, 104]. The aim of this study is to find the genomic markers of the HBV and clinical information which are useful to predict occurrence of liver cancer and response to therapy.

In this chapter, we look into the clinical data prepared by the clinicians, and the HBV DNA genomes prepared by the biochemists of our research group [74, 78]. Patients taken part in this study are selected by the clinicians carefully, according to their age, sex, and past clinical status. Chronic hepatitis B patients recruited since 1997 were prospectively followed up for the development of HCC for avoiding selection bias. HCC was diagnosed by a combination of alpha fetoprotein, imaging, and histology. Liver cirrhosis was defined as ultrasonic features of cirrhosis together with hypersplenism, ascites, varices, and/or encephalopathy [73]. Clinical features for analysis were chosen by clinicians based on their expert knowledge. Primer Express software version 2.0 (PE applied Biosystems, Foster City, CA) was used to find suitable primers and probes. TaqMan real-time PCR technology was used to differentiate the nucleotide variant [74]. Because the focus of this chapter is on the study of data mining techniques, the selection process and criteria of patients and the research experiments run by our Biochemistry Department will not be discussed in detail.

In [93], HBV DNA sequences were taken from 13 patients. Authors of [94] amplified a conserved core region and a surface antigen region of HBV DNA by PCR from sera of 27 Korean chronic hepatitis B patients for detecting hepatitis B virus mutants. Our project is one of the biggest HBV DNA full sequence collection and analysis studies of its kind. We have collected DNA sequences from 98 Control (normal) and 100 HCC (cancer) patients specifically for this project. The DNA sequences of HBV are not exactly the same for each group, and they possess some individual nucleotide mutations that may or may not be related to HCC. From previous studies, HBV can be divided into seven genotypes where each of them has more than 8% difference of nucleotides to the others. In Hong Kong, genotypes B and C are the predominant types, and all the examples we have are of these two genotypes. To reduce the noise of genotypic difference amongst the sequences collected, we propose to analyze these DNA examples in each genotype separately.

Classification is one of the most studied data mining tasks. The objective is to predict the value (the class) of a user-specified goal feature based on the values of other features, called the predictive features. The goal feature might be the prediction of whether or not a patient has cancer, while the predictive predictive features might be the mutation sites of the patient's virus DNA.

The focus of this chapter is to identify genetic marker(s) for liver cancer (HCC) from Hepatitis B Virus (HBV) DNA sequences. There are similar medical researches reports, but all of them are focused on the specific gene positions, proteins or part of a virus genome. However, our project is the first study on the complete viral genome. One of the past researches is a HIV genomic study [75]. The researchers align each DNA sequence with a reference sequence, and then select the genes using their expert knowledge, and use Decision Tree and Support Vector Machine for analysis. In [76], the researchers focused on the identification of HBV DNA sequences that are predictive of response to one therapy. Some sites in sequences were observed to have caused the effect. Chan and others studied the risk factors in HBV sequences in respect of medicine [73, 74]. Here, we apply soft computing tools to predict positive patients and analyze the effective mutation sites in the HBV DNA sequences.

The aim of this chapter is to develop a data mining framework which contains an appropriate classifier for liver cancer based on HBV DNA and clinical data. We develop two new algorithms based on rule learning and Nonlinear Integrals. We then carry out a thorough comparative study on these two new models with existing classifiers. The classification model should have high sensitivity and acceptable accuracy and specificity for HCC diagnosis and prediction. The model learnt should also give clear indication of the degrees of influence of the features towards the classification goal and whether there are any interactions among the predictive features. In this chapter, we identified the important mutation sites (markers) in the HBV sequences that could have caused or related to liver cancer. We use information entropy for finding genetic markers of HCC in the HBV genome data and propose a new classification model based on the Nonlinear Integrals.

6.2 Data Mining Framework

The data mining framework developed is shown in Figure 6.1. There are nine modules. After the molecular evolutionary analysis, the data are passed to the Clustering Module to check whether clusters exist based on the phylogenetic tree analysis. If clusters are found, each cluster will be analyzed separately for potential genetic marker sites because it will minimize the noise produced by the genotype differences and give much better classification accuracy. For each cluster (or genotype), the data are divided into training and test sets. The training examples are then passed to the Feature Selection Module to find the useful features (genetic marker sites) for classification. The potentially useful features are extracted and passed to the Classifier Learning Module wherein a classifier is learnt. The features selected are also sent to the preprocessing module to extract the values of these features in the testing dataset for testing in the Classification module. Finally, the prediction results of the classifier are verified and evaluated based on the testing examples. If the evaluation results are unsatisfactory, i.e. stopping criteria are not satisfied, the learning process is repeated starting from the feature selection; otherwise, the classifier will be validated by never-seen-before examples. The following subsections will explain how the features are selected and the basic principles of the classifier.

6.2.1 Molecular evolutionary analysis

Serum examples from 49 patients infected with HBV genotype C, as determined by previous genotype-specific restriction fragment length polymorphism analysis, were studied [73]. All serum examples were kept in a −80 °C freezer for storage. All patients were ethnic Chinese and were followed up in the Hepatitis Clinic of the Prince of Wales Hospital (Hong Kong). All patients were positive for hepatitis B surface antigen for at least 6 months and had no evidence of hepatocellular carcinoma. Sixty-nine full-genome nucleotide sequences of HBV genotype C and 12 full-genome nucleotide sequences of nongenotype C HBV were also retrieved from the GenBank database for

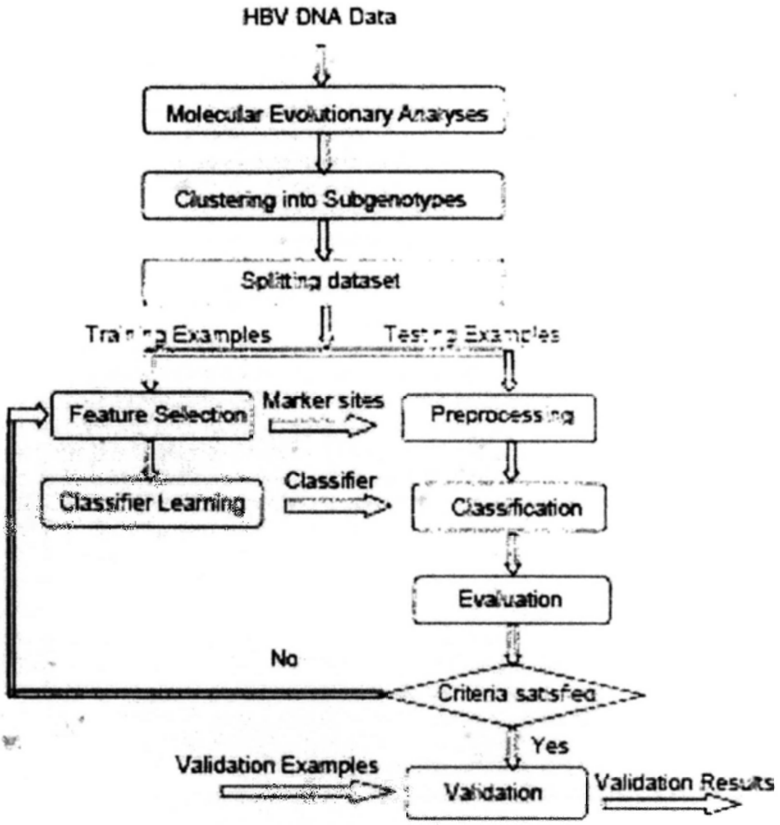


Figure 6.1: Data Mining Framework

comparison. All reference sequences from GenBank were derived from patients with chronic hepatitis B; HBV nucleotide sequences from patients with acute hepatitis B hepatocellular carcinoma or patients treated with antiviral agents were excluded. The geographical origins of patients harboring different HBV genotype *C* genomes in GenBank were retrieved from the respective original publications and the descriptions in the GenBank database.

The full-genome nucleotide sequences of the isolates of HBV genotype C from our center were compared with those of the isolates of HBV genotype C and nongenotype C HBV retrieved from the GenBank database. Nucleotide sequences are multiple-aligned using ClustalW version 1.83 [95] and corrected manually by visual inspection. Genetic distances are estimated by Kimura's two-parameter method and the phylogenetic trees are constructed by the neighbor-joining method [96, 97]. The reliability of the pairwise comparison and phylogenetic tree analysis is assessed and assured by bootstrap resampling with 1000 replicates. Phylogenetic and molecular evolutionary analyses are done using MEGA version 3.0 [98].

6.2.2 Clustering

Since different HBV subgroups are likely to be the results of divergence from genomic mutations over time, knowledge of the geographical distribution and genomic relatedness of the HBV genotype C subgroups will be useful in gaining an understanding of the spread of HBV in Asia. Hepatitis B virus genotype B (HBV/B) has been classified into 5 subgenotypes. In [77], a phylogenetic analysis of the complete genome sequences from the examples obtained from the Arctic and of those from Japan and Asia revealed 6 distinct clusters within HBV/B. Within each HBV genotype C subgroup, several clusters with genomic resemblance to one another can be identified. The most well-defined example is the cluster in Okinawa, where the prevalence of HBV genotype C is much lower than that in the rest of Japan [99].

There are two genotypes, B and C, in the two hundred plus HBV DNA sequences we collected specifically for this project. While genotype B HBV appears to be a homogenous group [105], the phylogenetic tree results show that there exist 3 main clusters in the genotype C among the HBV strains collected (Figure 6.2) [78]. We label them as C1, C2 and C3 respectively. Subgrouping of HBV genotype C was based on an intersubgroup difference of nucleotide sequence of >4% [100]. This is in concordance with our previous phylogenetic analyses with published full-length sequence in the GenBank. The main reason for us to find markers separately from within the clusters (subgenotypes) obtained from clustering analysis is that these subgenotypes exhibit mutations (nucleotide site differences) caused by geographical diversity which are not markers for carcinogenic diagnosis. If we were to analyze all these subgenotype data as one genotype group, their intergenotypic differences would become distracting noises in the data mining process for markers.

These three clusters can be identified by the combinations of 4 nucleotides. These three clusters will be analyzed separately in the classifier learning part.

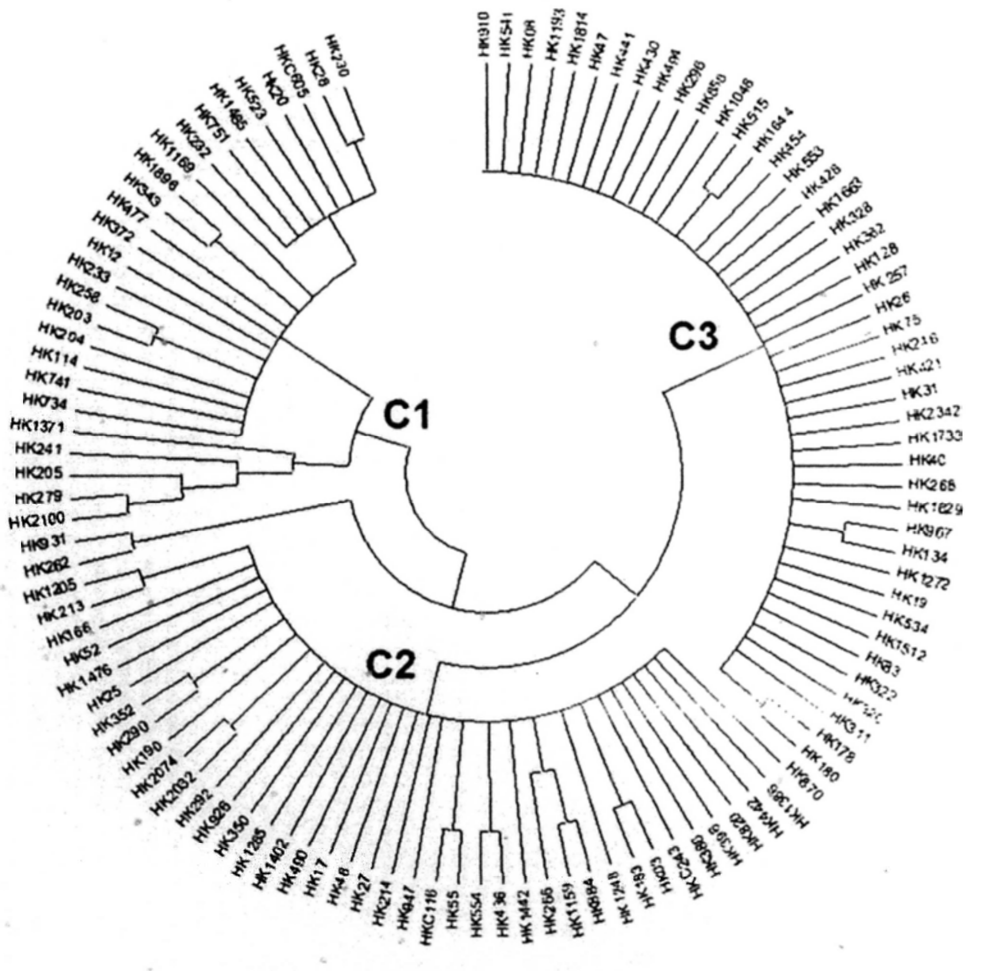


Figure 6.2: Phylogenetic Tree of Genotype *C*

6.2.3 Feature selection algorithm

The main purpose of feature selection [79, 80, 81, 82] is to reduce the number of features used in classification while maintaining acceptable classification accuracy. For example, the Sequential Forward Floating Selection (SFFS) algorithm proposed by Pudil et al. [83] was one of the commonly used algorithms [84]. The main advantage of this method is that it produces a hierarchy of feature subsets with the best selection for each dimension. However we aim at global performance of the whole framework, so we adopt a simpler algorithm based on information gain to select initial features.

In our approach, information gain criterion [9] is used to find the useful features to distinguish between the Control (normal) and the HCC (cancer) groups of HBV carriers.

Information gain is a common criterion for feature selection. The information gain of a feature is the uncertainty (entropy) that can be reduced if the feature is used for classification. Hence, the information gain should be the higher the better. Equation 1 is the entropy E of a feature X with n values $x_1, x_2, ..., x_n$ and $P(x_i)$ is the probability of the value x_i.

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

Specific to a typical DNA classification problem, we assume the data have m classes $C = c_1, c_2, ..., c_m$. For each aligned site position, it has K possible nucleotides $V_1, V_2, ..., V_K$. We define $|c_k|, k = 1, 2, ..., m$ as the number of sequences in class c_k. $|c_{ki}|$ is the number of sequences in Class c_k whose character at the aligned site is V_i, which can be A, T, G, or C in our case. The Remainder of X, $R(X)$ is defined as follows:

$$R(X) = \sum_{i=0}^{K} \frac{\sum_{k=1}^{m} c_{ki}}{\sum_{k=1}^{m} c_k} E(P(c_{1i}), P(c_{2i}), ..., P(c_{ki})) \qquad (2)$$

Information Gain IG_j of the aligned site j is the difference between the original information content $E(C)$ of the data set and the amount of information needed to classify all the unclassified data left in the data set after applying site j for classification:

$$IG_j = E(C) - R(j) \qquad (3)$$

The features are ranked by the information gains, and then the top ranked features are chosen as the potential features used in the classifier. A site with higher information gain will contribute more in the classification and be able to distinguish more examples (cases).

6.2.4 Popular classification algorithms

There are several common classification models such as Naïve Bayesian Network [10, 11, 12], Decision Tree, Neural Networks and Rule Learning using Evolutionary Algorithm [85]. The learning processes of Naïve Bayesian Networks and Decision Tree are faster. However, they cannot cope well with feature interactions. Neural Networks are treated as black box learning and it is difficult for human to understand or interpret the classification explicitly.

However, Rule Learning using Evolutionary Algorithm performs a global search and can cope with feature interactions better than the previous classification models [86, 87]. Also, the classification rules generated are simple and easily interpretable by human experts who frequently use the same reasoning approach ~~very~~ much similar to the rules. Therefore, the Rule Learning Using Evolutionary Algorithm approach is clearly a better choice in terms of interpretability of the knowledge acquired through the classifier learnt.

Rule learning tries to learn rules from a set of training data (examples). It can be modeled as a search problem of finding the best rules that classify the training examples with minimum error. However, the search space can be very large; a robust search algorithm is required. Here Generic Genetic Programming (GGP) [88, 91], which is a type of the Evolutionary Algorithms (EA), is adopted as our search and optimization algorithm. Firstly, a population is initialized by generating individuals (a set of rules) randomly. A fitness function is used to evaluate how good an individual is, that is, how many cases it can classify correctly. Then some individuals are selected to evolve (generate) new individuals with the genetic operators. Individuals become better and better through the evolution process until the termination criterion is met.

The input is the training dataset, and the output is a rule set, which can classify the training data with higher accuracy. We assume there are n features $X = x_1, x_2, ..., x_n$ and m classes $C = \{c_k | k = 1, 2, ..., m\}$. For each feature x_j, one of its K_j values can be taken. Each rule includes two components, the antecedent (IF part) and the consequence (THEN part), as follows:

IF $(x_1 = v_1) \wedge (x_2 = v_2) \wedge \cdots \wedge (x_l = v_l)$ THEN Class is $c = c_k$.

where the antecedent includes $l(l = [1, n])$ unique feature $x_1, x_2, ..., x_l \in \{x_1, x_2, ..., x_n\}, v_1, v_2, ..., v_l \in \{A, C, G, T\}$ and $c_k, k \in 1, 2, ..., m$, is a certain class to which the object is to be classified. In our case, we have only two classes, namely HCC and CONTROL. There are l unique features present in and $n - l$ features absent from each rule. Each feature present in the antecedent can only take one of its possible values, $\{A, C, G, T\}$. All the rules in the output rule set are connected by ELSE IF, meaning the order of application of the rules must be followed.

We use a simple example to illustrate the rules deduced by the Rule Learning. For HBV dataset B which will be introduced in the following section, we have learnt the rules for diagnosing lever cancer (HCC) and non-lever cancer

(CONTROL) cases. The rules are given as follows:

IF A1762 and G1764 and C53, then HCC

ELSE IF T1762 and A1764 and CG2712, then HCC

ELSE IF T1762 and A1764 and T2712 and C2525, then HCC

ELSE CONTROL

Although Rule learning based on EA can interpret the interaction of features, the degree of the interaction cannot be analyzed exactly by a measure. So we introduce the Fuzzy Measure to describe the interaction with respect to the classification. A new classification model is proposed based on the Nonlinear Integrals with respect to signed Fuzzy Measure in the following section.

6.2.5 Classification based on the Nonlinear Integrals and the Generalized Nonlinear Integral

In classification, we are given a data set consisting of N example records, called the training set, where each record contains the value of a classifying feature, Y, and the value of predictive features $x_1, x_2, ..., x_n$. Positive integer N is the data size. The classifying feature indicates the class to which each example belongs, and it is a categorical feature with values coming from an unordered finite domain. The set of all possible values of the decisive feature is denoted by $c_1, c_2, ..., c_m$, where each $c_k, k \in 1, 2, ..., m$, refers to a specified class. The predictive features are numerical, and their values are described by an n-dimensional vector, $(f(x_1), f(x_2), ..., f(x_n))$. The range of the vector, a subset of n-dimensional Euclidean space, is called the feature space. Thus, the j example record consists of the j^{th} observation for all predictive features and the classifying feature, and is denoted by $(f_j(x_1), f_j(x_2), ..., f_j(x_n), Y_j), j = 1, 2, ..., N$.

In this section, one method of classification based on the Nonlinear Integrals will be presented. It can be viewed as an idea of projecting the points in the feature space onto a real axis through a Nonlinear Integral, and then using a one-dimensional classifier to classify these points according to a certain criterion optimally. Andthoer method based on the Generalized Nonlinear Integral is a polynomial transformation of the classical Nonlinear Integral classifier. It implement projecting through not straight line but curve according to the polynomial integrand. The polynomial index will be learned with the signed Fuzzy Measure together. Our classifying features holding the discrete value of A, C, G or T is numericalized to be a virtual variable. All of these are realized under the guide of an adaptive genetic algorithm [13]. Good performance of this method comes from the use of the Fuzzy Measure and the relevant Nonlinear Integral, since the nonadditivity of the Fuzzy Measure reflects the importance of the predictive features, as well as their inherent interactions, towards the discrimination of the points. In fact, each predictive feature has respective important index reflecting their amounts of contributions towards the decision. Furthermore, the global contribution of several predictive features to classification is not just the simple sum of the contribution of each feature to the decision, but may vary nonlinearly. A combination of the predictive features may have mutually restraining or a complementary synergy effect on their contributions towards the classification decision. So the Fuzzy Measure defined on the power set of all predictive features is a proper representation of the respective importance of the predictive features and the interactions among them, and a relevant Nonlinear Integral is a good fusion tool to aggregate the information coming from the individual and the combinations of the predictive features for the classification. The details of these basic concepts about the Nonliear Integrals and the Generalized Nonlinear Integral and the mathematical models for the classification problem have been introduced in Chapter 2 and Chapter 4. Here we just give a brief review about classification model and corresponding revision acorrding to the realistic problem.

The classification process can be divided into two parts for implementation:

The Nonlinear Integral classifier depends on the Fuzzy Measure μ, so the first step is to determine the optimal values of μ by using GA tool. In fact, the fitness function comes from the linear classifier used in the second procedure. It is an iterative process. The optimal Fuzzy Measure will be output to the next step.

When the Fuzzy Measure μ is determined, the virtual value can be obtained using the Nonlinear Integral. Then we can classify these virtual values on real axis using a linear classifier.

The following paragraphs focus on the above problems respectively.

Here we discuss the optimization of the Fuzzy Measure μ under the criterion of minimizing the corresponding global misclassification rate.

In our GA model, we use a variant of the original function f, $f' = af + b$. where a is a vector to shift the coordinates of the data and b is a vector to scale the values of predictive features. Each chromosome represents Fuzzy Measure vector μ, shifting vector a and scaling vector b. A signed Fuzzy Measure is 0 at empty set. If there are n features in training data, a chromosome has 2^n+2n-1 genes which are set to random real values randomly at initialization. Genetic operations used are traditional ones. At each generation, for each chromosome, all variables are fixed and the virtual values of all training data are calculated using Nonlinear Integral. The fitness function can be defined as follows.

$$fitness = \omega_1 accuracy + \omega_2 sensitivity$$

ω_1 and ω_2 are the adjustment parameters given by users. Accuracy and sensitivity are determined in the second part of model.

6.3 Methods

We applied EA based Rule Learning [89], Nonlinear Integral classifiers and the Generalized Nonliner Integral-Nonlinear Integral with Polynomial Kernel to classify the HBV DNA data into liver cancer (HCC) and normal (CON, control) classes and then compare them with several classical classification methods which include See5.0 (Decision Tree), Neural Network, Support Vector Machine(SVM) and Naïve Bayes. As mentioned before, we do a detailed study on the Rule Learning and Nonlinear Integral classifier separately. These classical classification methods and the data sets used have been introduced in Chapter 2. Then the implementation details of the Nonlinear Integral (NI) classifier and the evaluation methodology will be introduced.

6.3.1 Implementation Details of Nonlinear Integral

To implement the learning algorithm of our new classifiers based on the classical Nonlinear Integrals and Nonlinear Integrals with Polynomial Kernel, we use the GA tool in Matlab v7.2 Programming combined with Fisher's discriminant function programming [90]. All the parameters of our GA in our experiments are shown in Table 6.1. We set the generation limit to be 100 as the stopping criteria.

Table 6.1: GATOOL Parameters in MATLAB

Parameter	Set value	Parameter	Set value
PopulationType	doubleVector	Stall generations	Inf
PopulationSize	20	Stall time limit	Inf
EliteCount	2	Tolerance	1.0000e-006
CrossoverFraction	0.8000	Constraint tolerance	1.0000e-006
CrossoverFunction	crossoverscattered	InitialPenalty	10
MigrationDirection	forward	PenaltyFactor	100
MigrationInterval	20	PlotInterval	1
MigrationFraction	0.2000	CreationFunction	gacreationuniform
Generations	100	FitnessScalingFunction	fitscalingrank

Table 6.2: The Details of HBV Data Sets

Datasets	Control	HCC	Total	%
B	51	37	88	43.878
C1	10	16	26	13.265
C2	18	22	40	20.408
C3	19	25	44	22.449
Total	98	100	198	

6.3.2 Data description

The dataset contain 98 control patients and 100 HCC patients. The HBV DNA sequences are obtained specifically for this study from these patients carefully selected by our medical experts to minimize the demographic bias. There are four datasets corresponding to the different clusters, namely B, C1, C2 and C3. The numbers of patients for each dataset are shown in Table 6.2 in which the last column represents the proportion of each dataset. For each dataset, an independent validation set is prepared to evaluate the performance of the classifiers. Table 6.3 shows the number of patients of the validation datasets.

Table 6.3: Summary of Validation Sets

Datasets	Control	HCC	Total
B	8	7	15
C1	7	5	12
C2	9	6	15
C3	5	5	10
Total	29	23	52

6.3.3 Evaluation Methodology

In classifying an unknown case, depending on the class predicted by the classifier and the true class of the patient (Control or HCC), four possible types of results can be observed for the prediction as follows:

i) True positive - The result of the patient has been predicted as positive (Cancer) and the patient has canner.

ii) False positive - the result of the patient has been predicted as positive (Cancer) but the patient does not have cancer.

iii) True negative - the result of the patient has been predicted as negative (Control), and indeed the patient does not have cancer.

iv) False negative - the result of the patient has been predicted as negative (Control) but the patient has cancer.

Let TP, FP, TN and FN denote respectively the number of true positive, false positives, true negatives and false negatives. For each learning and evaluation experiment, *Accuracy*, *Sensitivity* and $Specificity$ defined below are used as the fitness or performance indicators of the classification.

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

$Sensitivity = TP/(TP + FN)$

$Specificity = TN/(TN + FP)$

For screening tests, medical professional usually will prefer to have higher sensitivity, i.e., lower accuracy and specificity is an acceptable trade-off for high sensitivity as long as the accuracy and specificity are reasonable. It means that we rather send more people for confirmation tests than miss any true cancer patients. In the data sets, all features are categorical features. There are four symbolic values A, C, G and T for each feature. In order to use the nonlinear model, we use simple integer values, 0, 1, 2 and 3, as the numericalised initial values to represent the discrete values of the features respectively.

We adopt K-fold cross validation method to make sure that the whole dataset can be used as testing data in turn and over-training (over-fitting) can be avoided. It means that we randomly partition the N data into K sets of size of N/K, train on $(K-1)$ sets and test on the remaining set, and repeat K times in turn and take the mean result. After K runs, all data are used for testing and the average can be computed to evaluate the performance. The K-fold method is repeated 10 times for each experiment (10 runs in total) to obtain an overall average performance.

Our datasets are very small despite they are one of the biggest single studies. For example, C1 contains DNA sequences from 26 individuals. We must ensure that there is at least one positive case for each class in the testing dataset. If the number (K) of splits is too large, the size of the testing set will be too small and it may not even have a positive case. On the other hand, if the numbers of splits are too small, it will result in small training sets which may not contain sufficient information for training. So we need to find a balance between the sizes of training and testing sets in order to reduce the probability of over-training (over-fitting) and under-testing (i.e. not enough positive and negative examples for testing). We have tried several feasible K values for Nonlinear Integrals. The results are shown in Table 6.4. We can see that the testing accuracy and sensitivity are best by taking 10-fold. Consequently, we have chosen 10-fold method for our experiments. This 10-fold methodology is applied to all experiments including the classical classifier used in our comparison.

6.4 Experimental and Analysis

In this section, we first present the results of EA based Rule Learning [89] and Nonlinear Integral classifiers to classify the HBV DNA data into liver cancer (HCC) and normal (CON, control) classes and then compare them with several traditional classification methods which include See5.0 (Decision Tree) [9], Neural Network [10], Support Vector Machine(SVM) [11] and Naïve Bayes [12]. As mentioned before, we do a detailed study on the Rule Learning and the Nonlinear Integral based classifier separately because of the importance of their high interpretability of the models representing the knowledge acquired through the learning processes. The biochemists and doctors can see explicitly and clearly the influences of the mutated sites or markers and their potential interactions towards the formation of lever cancer.

For each dataset, we will use the 5 features which include those selected by the Rule Learning method and in some cases supplemented by those with the highest information gain obtained by Viewer [52] partially shown in Figure 6.9. For reducing computational complexity, we reduce the number of features by including feature selection method. We compared the results of the Nonlinear Integral based classifier with and without feature selection in Table 6.5. It shows that the feature selection is very useful.

6.4.1 Comparison between NIC and RL

Table 6.6 shows the comparison results of Rule Learning and the Nonlinear Integrals Classifier (NIC) and Table 6.7 shows the comparison results of our methods with several classic methods on datasets B, C1, C2 and C3. The results of Rule Learning (RL) and the classical Nonlinear Integral Classifier (NIC) for each dataset and a validation set, which contains the never-seen-before cases, are shown in Table 6.6.

In Table 6.6, sensitivity results of NIC are higher than those of RL in most cases and other values are comparable. Since sensitivity is more important for doctors to diagnose, the performance of NIC is considered to be better than that of RL. Furthermore, NIC not only can determine the important sites (markers) with regard to the diagnosis but also give their degrees of contribution in real values, which are relatively meaningful in biomedical research. This will be described in following section.

6.4.2 Results of Classifier based on Nonlinear Integrals and Generalized Nonlinear Integrals compared with other methods

Table 6.7 shows the comparison results of the Nonlinear Integrals Classifier and the Nonlinear Integrals with Polynomial Kernel based Classifier(NIPKC) with five classical algorithms which include Neural Network (NN), Decision Tree (DT), Naïve Network (NB), SVM and Rule Learning (RL).

We run six sets of experiment for each classifier. The first set of experiments uses the top 1 site (feature with the highest information gain), the second set the top 2 sites, the third one uses top 3 sites and so on. The results are evaluated mainly according to the test accuracy and sensitivity for HBV data. Finally, the best result out of the 6 sets of experiments for each method is selected for comparison. So we can see the optimal results of NIC and the NIPKC based on GA compared against other methods in Table 6.7.

For weighted average results in Table 6.7, the best ones are bolded. The weighted average is computed according to the number of cases in each dataset. The sensitivity and accuracy of our NIC and NIPKC are better than most algorithms or at least comparable. For comparing the performance of all algorithms graphically, we plot all the result in Figures 6.3 to 6.7. Meanwhile, we place all methods on a ROC space as Figure 6.9 for helping interpreting the results in Table 6.7.

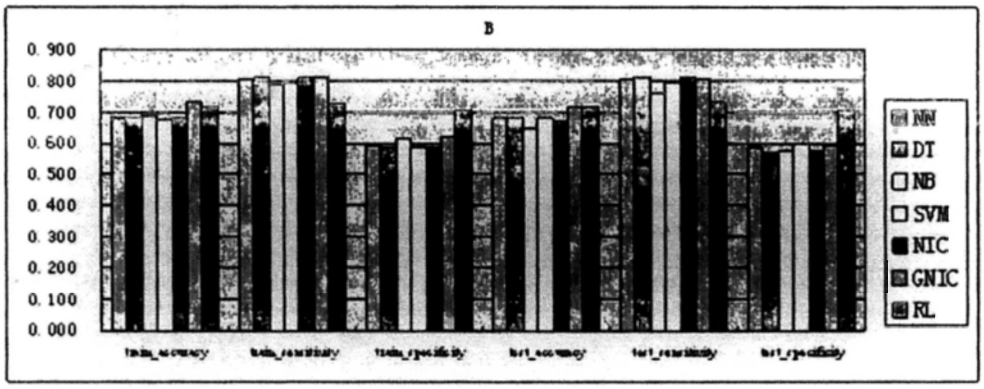


Figure 6.3: The Comparison of All Methods for Dataset B

6.4.3 Comments on results

Our framework includes Rule Learning (RL), Nonlinear Integral Classifier(NIC) and the generalized Nonlinear Integral Classifier-Nonlinear Integrals with Polynomial Kernel based Classifier(NIPKC). From Table 6.6 RL has slightly higher accuracy than NIC and NIPKC. It means that this method can have higher prediction power. But for doctors and clinicians, the sensitivity is more important than the accuracy, and NIC is better than RL on sensitivity.

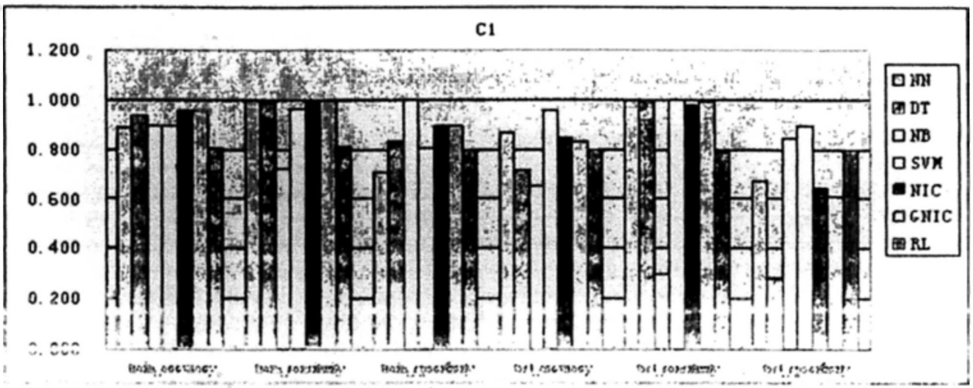


Figure 6.4: The Comparison of All Methods for Dataset C1

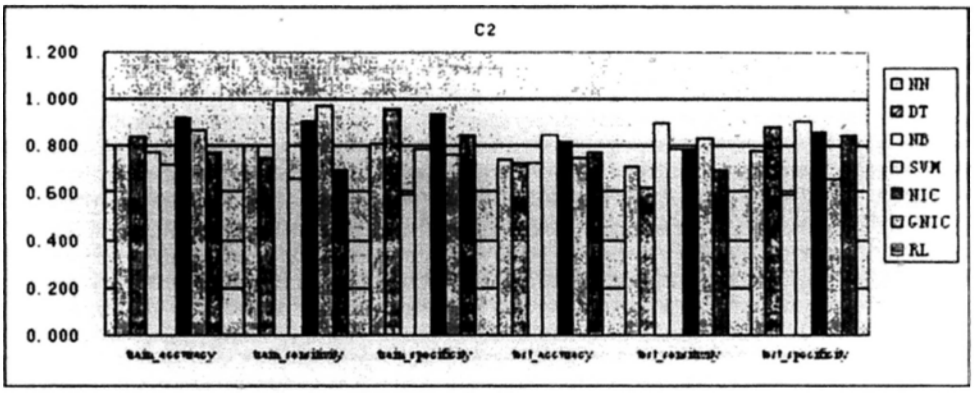


Figure 6.5: The Comparison of All Methods for Dataset C2

Compared with the four traditional methods namely Neural Network(NN). Decision Tree(DT), Naïve Bayes(NB) and Support Vector Machine(SVM), NIPKC shows the best diagnostic performance on the average evaluations. SVM shows better accuracy but inferior sensitivity for the test data. However, for screening tests, medical professional usually will prefer to have higher sensitivity, i.e., lower accuracy and specificity is an acceptable trade-off for high sensitivity as long as the accuracy and specificity are reasonable. It means that we rather send more people for confirmation tests than miss any true cancer patients. NIPKC and NIC not only have comparable accuracy, they can also show the interaction of features. How to identify the importance

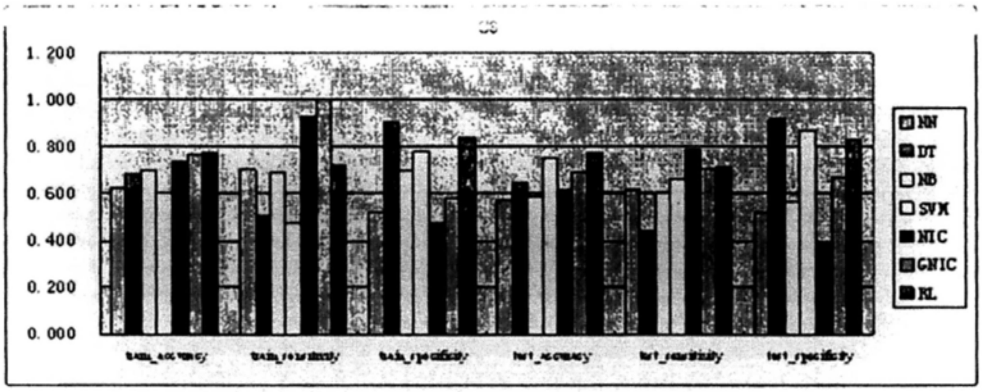


Figure 6.6: The Comparison of All Methods for Dataset C3

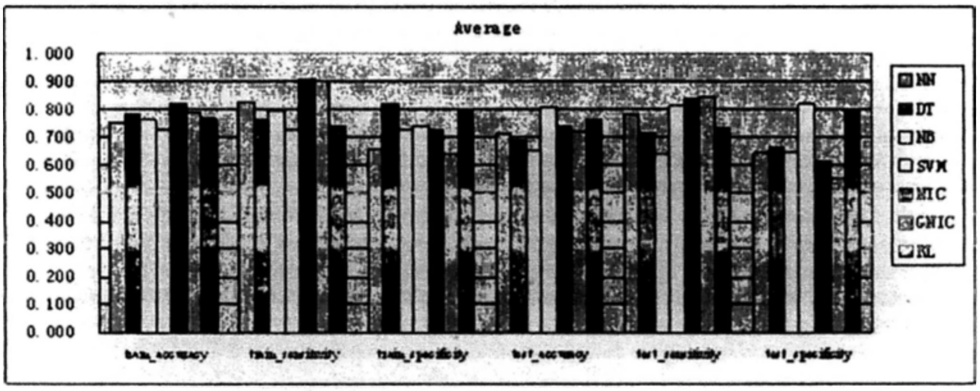


Figure 6.7: The Comparison of All Methods as Weighted Average

of features and their combinations will be introduced in the next section.

6.4.4 To identify important sites and interactions among them

Another important contribution of Nonlinear Integral classifier is that we can find some significant sites (markers) and interactions among them in the sequences for further wet laboratory analyses. According to the definition of Nonlinear Integrals, for each dataset we can get a set of linear equations about the signed Fuzzy Measures as variables. A solution with the fewest nonzero values can be obtain by solving linear equations based on L1-Norm regularization [92].

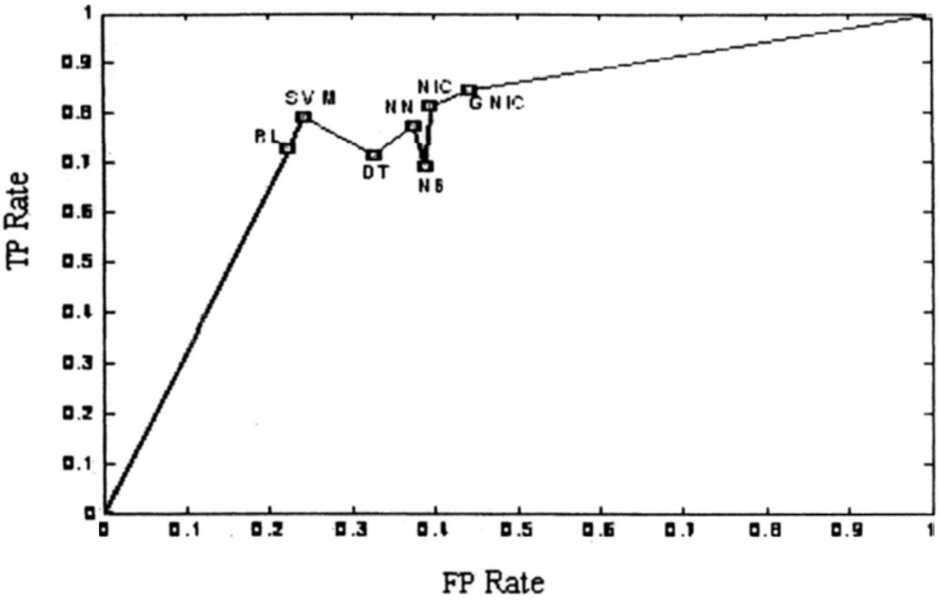


Figure 6.8: Classifiers in ROC space

The respective potential sites according to information gain in the sequences for x_i computed by Viewer[52] are listed in Table 6.8. Figure 6.9 is the screenshot from the Viewer for Dataset C1. The left column is the site numbers and the right column is the corresponding information gain values ranked in decreasing order.

For the B, C1, C2 and C3 datasets, the top 5 sites are used to formulate the set of linear equations. So we obtained the solutions which have the fewest nonzeros and filtered those positions with zero. In Table 6.9, we show the importance and relevance of the individual sites and their interactions.

From Table 6.9 we can see that many sites of sequence do not take effect individually or combined with others. The non-zero sites are important for diagnosing disease. This may be helpful for the bioinformatics and medical research. The L1-norm method is faster than using GA. But the results are not optimal. We use GA to search for the optimal results for Fuzzy Measure

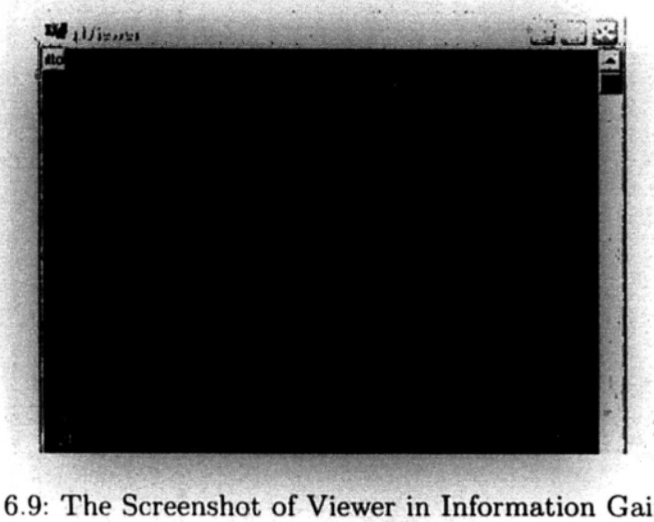


Figure 6.9: The Screenshot of Viewer in Information Gain Order

in most cases.

6.5 Chapter Summary

In this chapter, a data mining framework for DNA sequence biological datasets has been presented. It has been applied to the Hepatitis B Virus DNA datasets which are among the largest in the world and have been collected by our medical school specifically for this project. We have developed a framework for markers discovery. This framework has incorporated three algorithms, NIC, GNIC and RL. These classifiers can explicitly give the importance of the markers and their interactions and have shown good performance in cancer prediction.

Moreover, the details of the new classification methods based on the classical Nonlinear Integral and the Nonlinear Integral with Polynomial Kernel have been presented. These methods have good performance using the Fuzzy Measure, due to the nonadditivity of the Fuzzy Measure reflecting the importance of the individual predictive features as well as their inherent interactions. Besides the high interpretability the Nonlinear Integrals Classifier, the experimental results have shown that it is one of the best classifiers especially in terms of the sensitivity. It is very useful for preliminary diagnosis and screening test of liver cancer caused by HBV. In our model, we use GA for optimization which provides multi-modal solutions containing sets of best solutions. The final confirmation experiments, like many other bioinformatics problems, need to be carried out by biochemists to identify and study the true markers. Finally, we have used a L1-norm regularization method to get a solution with the fewest non-zero Fuzzy Measure values. It can provide some important individuals and combinations of key markers of the HBV DNA sequences. We believe that this information can be helpful to do further research for biochemists.

We hypothesize that the genomic makeup of HBV affects the carcinogenic potential of the virus. In this case-control study, we have demonstrated that some genotype-specific mutations are more commonly found among HCC patients than their age and gender matched controls. These markers can therefore be used as biomarkers to stratify the cancer risk of chronic hepatitis B patients. Our findings have been validated by independent datasets in the validation process. To confirm the biological role of these mutations, further experimental work using in situ mutagenesis of replicative HBV clones on their carcinogenicity in animal and cell line models will be required.

However, even though we have generated one of the largest datasets, the example sizes of the datasets are still small (less than 100) for each case. It is a challenge for the classifiers based on the Nonlinear Integral and the Generalized Nonlinear Integral to avoid overtraining.

Table 6.4: All Splits for Training and Testing on Nonlinear Integral Classifier

Data	Per	2fold	3fold	4fold	5fold	6fold	7fold	8fold	9fold	10fold
B	Acc	0.747	0.731	0.728	0.728	0.730	0.722	0.723	0.725	0.682
training	Sen	0.808	0.829	0.805	0.803	0.804	0.814	0.802	0.802	0.811
	Spe	0.702	0.660	0.672	0.672	0.677	0.656	0.665	0.669	0.588
B	Acc	0.649	0.646	0.647	0.643	0.621	0.637	0.634	0.625	0.674
testing	Sen	0.687	0.700	0.687	0.678	0.651	0.698	0.696	0.678	0.813
	Spe	0.621	0.606	0.617	0.615	0.600	0.593	0.588	0.587	0.574
C1	Acc	0.962	0.961	0.962	0.961	0.960	0.962	0.962	0.962	0.961
training	Sen	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	Spe	0.900	0.899	0.900	0.900	0.896	0.900	0.900	0.900	0.899
C1	Acc	0.785	0.815	0.840	0.843	0.848	0.845	0.856	0.849	0.847
testing	Sen	0.963	0.952	0.944	0.963	0.964	0.957	0.975	0.972	0.980
	Spe	0.500	0.594	0.683	0.660	0.650	0.650	0.656	0.661	0.640
C2	Acc	0.905	0.893	0.883	0.876	0.877	0.875	0.876	0.870	0.918
training	Sen	0.895	0.883	0.882	0.873	0.878	0.884	0.873	0.872	0.903
	Spe	0.915	0.903	0.883	0.879	0.876	0.866	0.878	0.869	0.937
C2	Acc	0.725	0.721	0.730	0.723	0.723	0.715	0.730	0.728	0.817
testing	Sen	0.645	0.624	0.655	0.650	0.660	0.669	0.648	0.657	0.788
	Spe	0.805	0.818	0.805	0.795	0.786	0.762	0.813	0.798	0.860
C3	Acc	0.780	0.774	0.772	0.767	0.765	0.761	0.765	0.761	0.731
training	Sen	0.724	0.753	0.738	0.715	0.743	0.739	0.717	0.738	0.913
	Spe	0.853	0.803	0.816	0.835	0.795	0.792	0.828	0.794	0.491
C3	Acc	0.651	0.609	0.606	0.624	0.604	0.627	0.624	0.606	0.600
testing	Sen	0.565	0.529	0.560	0.564	0.552	0.575	0.570	0.546	0.738
	Spe	0.763	0.712	0.670	0.698	0.674	0.695	0.698	0.670	0.410
Weighted	Acc	**0.814**	0.803	0.799	0.797	0.798	0.793	0.794	0.793	0.777
training	Sen	0.832	0.845	0.831	0.824	0.831	0.836	0.824	0.828	**0.877**
AVE.	Spe	**0.805**	0.772	0.777	0.780	0.772	0.761	0.775	0.767	0.678
Weighted	Acc	0.683	0.675	0.680	0.681	0.668	0.678	0.681	0.671	**0.709**
testing	Sen	0.688	0.680	0.686	0.685	0.672	0.699	0.695	0.683	**0.813**
AVE.	Spe	0.674	0.671	**0.676**	**0.676**	0.660	0.657	0.667	0.658	0.604

Note: Per=performance; Acc=Accuracy; Sen=Sensitivity; Spe=Specificity

Table 6.5: Comparison Results of Nonlinear Integral with and without Feature Selection

Datasets	Performance	with FS		without FS	
		Training	Testing	Training	Testing
B	Accuracy	**0.771**	**0.683**	0.687	0.617
	Sensitivity	**0.858**	**0.738**	0.337	0.233
	Specificity	0.708	0.646	**0.942**	**0.893**
C1	Accuracy	**0.922**	**0.790**	0.826	0.515
	Sensitivity	**1.000**	**0.920**	0.937	0.715
	Specificity	**0.798**	**0.600**	0.650	0.190
C2	Accuracy	**0.871**	**0.750**	0.793	0.626
	Sensitivity	**0.888**	**0.700**	0.813	0.698
	Specificity	**0.854**	**0.800**	0.769	0.530
C3	Accuracy	**0.828**	0.732	0.813	**0.750**
	Sensitivity	**0.843**	**0.705**	0.718	0.643
	Specificity	0.808	0.765	**0.937**	**0.900**

Table 6.6: Results of RL and NIC for Each Dataset

Datasets	Performance	RL			NIC		
		Training	Testing	Validation	Training	Testing	Validation
B	Accuracy	0.716	**0.716**	**0.769**	**0.771**	0.683	0.721
	Sensitivity	0.730	0.731	**0.800**	**0.858**	**0.738**	0.742
	Specificity	0.706	**0.707**	**0.750**	**0.708**	0.646	0.697
C1	Accuracy	0.808	**0.800**	**0.917**	**0.922**	0.790	0.712
	Sensitivity	0.812	0.790	**1.000**	**1.000**	**0.920**	0.854
	Specificity	**0.800**	**0.800**	**0.857**	0.798	0.600	0.570
C2	Accuracy	0.775	**0.775**	**0.917**	**0.871**	0.750	0.712
	Sensitivity	0.700	**0.700**	**1.000**	**0.888**	**0.700**	0.854
	Specificity	0.850	**0.850**	**0.857**	**0.854**	0.800	0.570
C3	Accuracy	0.773	**0.770**	**0.647**	**0.828**	0.732	0.639
	Sensitivity	0.720	**0.717**	0.700	**0.843**	0.705	**0.721**
	Specificity	**0.842**	**0.835**	**0.571**	0.808	0.765	0.523

Note: RL=Rule Learning; NIC=Nonlinear Integral Classifier

Table 6.7: Comparison Results with Classical Methods for All Datasets

Datasets	Performance	NN	DT	NB	SVM	NIC	NIPKC	RL
B training	Accuracy	0.681	0.682	0.689	0.674	0.682	0.731	0.716
	Sensitivity	0.805	0.811	0.790	0.794	0.811	0.812	0.730
	Specificity	0.591	0.588	0.617	0.589	0.588	0.618	0.706
B testing	Accuracy	0.680	0.681	0.650	0.680	0.674	0.714	0.716
	Sensitivity	0.806	0.812	0.758	0.795	0.813	0.804	0.731
	Specificity	0.589	0.571	0.573	0.597	0.574	0.591	0.707
C1 training	Accuracy	0.889	0.937	0.894	0.897	0.961	0.961	0.808
	Sensitivity	1.000	1.000	0.722	0.965	1.000	1.000	0.812
	Specificity	0.711	0.836	1.000	0.810	0.899	0.899	0.800
C1 testing	Accuracy	0.869	0.717	0.650	0.961	0.847	0.833	0.800
	Sensitivity	0.999	1.000	0.300	1.000	0.980	0.990	0.790
	Specificity	0.677	0.280	0.850	0.899	0.640	0.610	0.800
C2 training	Accuracy	0.805	0.839	0.773	0.725	0.918	0.870	0.775
	Sensitivity	0.799	0.749	0.993	0.665	0.903	0.971	0.700
	Specificity	0.813	0.953	0.589	0.785	0.937	0.769	0.850
C2 testing	Accuracy	0.746	0.728	0.727	0.848	0.817	0.748	0.775
	Sensitivity	0.715	0.615	0.897	0.789	0.788	0.835	0.700
	Specificity	0.783	0.880	0.592	0.907	0.860	0.660	0.850
C3 training	Accuracy	0.628	0.684	0.697	0.604	0.731	0.763	0.773
	Sensitivity	0.707	0.504	0.688	0.475	0.913	1.000	0.720
	Specificity	0.524	0.905	0.702	0.780	0.491	0.582	0.842
C3 testing	Accuracy	0.573	0.645	0.587	0.753	0.600	0.694	0.770
	Sensitivity	0.619	0.442	0.600	0.663	0.738	0.708	0.717
	Specificity	0.524	0.920	0.567	0.871	0.410	0.672	0.835
Weight Average training	Accuracy	0.722	0.748	0.735	0.698	0.777	**0.787**	0.753
	Sensitivity	0.808	0.755	0.799	0.720	0.877	**0.900**	0.732
	Specificity	0.637	0.765	0.680	0.700	0.678	0.641	**0.778**
Weight Average testing	Accuracy	0.695	0.687	0.652	**0.767**	0.709	0.721	0.751
	Sensitivity	0.772	0.715	0.691	0.791	0.813	**0.847**	0.729
	Specificity	0.625	0.673	0.612	0.760	0.604	0.556	**0.777**

Note: NN=Neural Network; DT=Decision Tree; NB=Naïve Network;
SVM=Support Vector Machine; RL=Rule Learning;
NIC=Nonlinear Integral Classifier.
NIPKC=Nonlinear Integral with Polynomial Kernel based Classifier.

Table 6.8: The Top 5 Sites No. of Sequences for Each Dataset

Data	site1	site2	site3	site4	site5
B	1762	1764	2712	1505	1627
C1	1915	1764	0928	1479	1461
C2	2170	2441	0799	2189	0814
C3	1768	1497	3098	1234	2768

Table 6.9: The Signed Fuzzy Measure of Each Site Used in Each Dataset

Sets of sites	B	C1	C2	C3
x_1	0.495	0.040	0.450	0.260
x_2	0.232	0.000	0.000	0.000
x_1, x_2	0.000	0.000	0.007	0.000
x_3	0.094	0.253	-0.183	0.000
x_1, x_3	0.175	0.000	0.860	0.000
x_2, x_3	-0.035	0.331	0.000	0.000
x_1, x_2, x_3	0.000	0.000	0.000	0.445
x_4	0.333	0.000	0.196	0.000
x_1, x_4	0.738	0.000	-0.604	0.000
x_2, x_4	0.102	0.000	0.000	0.000
x_1, x_2, x_4	0.000	0.000	0.000	0.000
x_3, x_4	0.252	0.000	0.000	0.000
x_1, x_3, x_4	0.566	0.000	0.000	0.000
x_2, x_3, x_4	-0.035	0.000	0.000	0.000
x_1, x_2, x_3, x_4	0.000	0.000	0.000	0.000
x_5	0.457	0.542	1.374	0.000
x_1, x_5	0.000	0.917	0.757	0.840
x_2, x_5	0.000	0.385	0.829	0.500
x_1, x_2, x_5	0.000	0.633	0.395	0.687
x_3, x_5	0.000	0.389	0.000	0.000
x_1, x_3, x_5	1.488	0.940	0.500	0.765
x_2, x_3, x_5	0.000	0.163	0.107	0.900
x_1, x_2, x_3, x_5	0.000	0.317	0.565	0.472
x_4, x_5	0.000	0.817	0.631	0.000
x_1, x_4, x_5	0.472	0.917	0.000	0.000
x_2, x_4, x_5	0.000	0.000	0.000	0.000
x_1, x_2, x_4, x_5	0.450	0.000	0.558	0.000
x_3, x_4, x_5	0.000	0.000	0.000	0.000
x_1, x_3, x_4, x_5	0.260	1.083	0.000	0.000
x_2, x_3, x_4, x_5	0.941	0.548	0.000	0.600
X	0.000	0.317	0.687	0.443

Chapter 7

Conclusion and Future Work

Knowledge integration and decision making often happen in environments where information concerned is interdependent or interactive. Artificial intelligence researchers have been attempting to emulate this capability in computer systems to handle information fusion assuming the input variables are independent. Nevertheless, the interaction among the information is ubiquitous in practical databases. Most existing approaches have limitations on dealing with the interaction among predictive features. The assumption about the independent features is not realistic for real-life problems. Nonlinear Integrals, such as Choquet Integral and Sugeno Integral, were proposed to solve the abovementioned limitations.

In this thesis, we have proposed various forms of generalization from the classical Nonlinear Integrals. The Generalized Nonlinear Integrals contain the Double Nonlinear Integral, the Nonlinear Integral with Polynomial Kernel and the Upper and Lower Nonlinear Integral which have been defined, implemented and applied to different benchmark datasets and some current projects. In all the models, we have adopted the Signed Fuzzy Measure with nonadditivity property to describe the contribution for decision of each predictive feature and the interaction among them. Compared to some traditional methods, the Generalized Nonlinear Integrals have promising performance for most problems and the best diagnostic performance for bioinformatics data, especially, for diagnosing cancers caused by Hepatitis B Virus(HBV).

7.1 Thesis contributions

The main contributions of this thesis have been described in chapter 1. We have proposed three kinds of extended versions of classical Nonlinear Integral for applications. They encompass the advantages of the classical Nonlinear Integrals and solve many limitations of the classical ones. We will give a detailed conclusion in the following parts.

- We have established a Double Nonlinear Integral model for classification. It implements the second projection by Nonlinear Integral to improve the inadequate performance coming from the first projection. This extension of Nonlinear Integral is based on the Signed Fuzzy Measure and GA learning parameters. Our model can choose automatically between the classical Nonlinear Integral and the Double Nonlinear Integral. When applying classical Nonlinear Integral to classification, there may be some important information missed which leads to the low classification accuracy. The Double Nonlinear Integral can solve this problem by projecting the virtual values again to get the second sets of virtual values. Each pair of the first and second values constructs the coordinates for each original datum. We can efficiently classify the data by making a linear classification on these virtual points in the 2-dimensional space.
- From another view, we have proposed an extension to the Nonlinear Integral with linear integrand-Nonlinear Integral with Polynomial Kernel.

It can project the original data along different types of curves instead straight lines in classical Nonlinear Integrals. It can deal with those data with more complicated distribution so that the better classification accuracy can be obtained. The shape of projection curves depends on the specific data, which means the polynomial index has to be learned by GA. together with the Fuzzy Measures. The complexity would not be increased significantly just adding one parameter, but the performance would be improved. It is testified not only on bench mark datasets but also on a real problem on bioinformatics.

- We have developed another formal extension of Nonlinear Integral called Upper and Lower Nonlinear Integrals. It can give a set of upper and lower bounds which subsume all kinds of Nonlinear Integrals. We use the multi-objective optimization method to find a set of optimal results for the regression model based on the Upper and Lower Nonlinear Integrals. We have tried to find a solution with the smallest distance between upper and lower bounds, and the smallest error from either of the bounds. This forms a NP hard problem. We can select one optimal solution for a specific problem from the set of results on the Pareto frontier. A weather predictor based on this model has been constructed. It can predict the following days' temperatures and ranges.

- A data mining framework has been established for the Hepatitis B Virus DNA sequence data. The framework includes molecular evolution analysis, clustering, feature selection, and classifier learning. In the feature selection process, potential markers are selected based on Information Gains for further classifier learning. Then meaningful rules are learnt by our algorithm called Rule Learning which is based on an Evolutionary

Algorithm. Two new classifiers based on classical Nonlinear Integrals and Nonlinear Integrals with Polynomial Kernels respectively have been introduced into the framework and obtained the best diagnosis performance.

7.2 Future work

Some improved and extended work concerning the current models and algorithms have been discussed in the previous chapters. In the following contents, suggestions for improvement are given.

- **Accelerating**: The current optimization algorithms of our models are rather time-consuming due to learning parameters which depends on the data. This problem is especially serious when the number of features is very large since the size of Fuzzy Measures is equivalent to the size of the power set of the features. We have to find a better accelerated strategies to improve the efficiency of learning the parameters technically and theoretically.

- **Fuzzifying**: Our algorithms deal with uncertain interaction of feature in the datasets with certain values. For some situations with uncertainty feature values, we need to introduce the fuzzifying scheme to the integrand and integral in the extensions of the Nonlinear Integrals.

- **More applications**: An important goal in designing the new regression and classification models is to solve real world problems. We plan to discover the potential power of our innovative models by finding more real applications. For example, we may use the multi-regression model to predict the stock market trends or forecast the trend of a certain financial index; we may use the projection classifier to determine the most important markers in the DNA sequences of more diseases such as diabetes.

Bibliography

[1] Torra, Vicen (Ed.), Information Fusion in Data Mining, *Series: Studies in Fuzziness and Soft Computing*, Vol. 123, 2003.

[2] Ishibuchi, H., Morisawa, T., Nakashima, T., Voting schemes for fuzzy-rule-based classification systems, *Proc. Of the Sixth IEEE Int. Conference on Fuzzy Systems*, 614-620, Barcelona, Catalonia, Spain,1996.

[3] Merz, C.J., Using Correspondence Analysis to Combine Classifiers, *Machine Learning*, 36 33-58, 1999.

[4] Webb, G.I., MultiBoosting: A Technique for Combining Boosting and Wagging, *Machine Learning*, 40 159-196, 2000.

[5] Bauer, E.,Kohavi, R., An Empiricial Comparison of Voting Classification Algorithms: Bagging, Boosting and Variants, *Machine Learning*, 105-139, 1999.

[6] Merz, C.J., Pazzani, M.J., Combining regression estimates, *Machine Learning*, 36 9-32, 1999.

[7] Breiman, L., Bagging Predictors, *Machine Learning*, 24 123-140, 1996.

[8] Schapire, R. E., The strength of weak learnability,*Machine Learning*, 5:2 197-227, 1990.

[9] Mitchell, Tom M., "Machine Learning". *The Mc-Graw-Hill Companies*, Inc., 1997.

[10] C. Eugene, Bayesian network without tears, *AI Magazine*, 12(4):5063, 1991.

[11] 11. D.M. Chickering D. Heckerman, D. Geiger, Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197243, 1995.

[12] W. Liu, J. Cheng and A.B. David, "An algorithm for Bayesian belief network construction from data". *Proceedings of the 6th International Workshop on Artificial Intelligence and Statistics*, 1997.

[13] Z. Y. Wang, K. S. Leung, and J. Wang, "A genetic algorithm for determining nonadditive set functions in information fusion." *Fuzzy Sets and Systems*, Vol. 102, pages 463-469, 1999.

[14] M. Martinez-Arroyo, L. E. Sucar, "Learning an Optimal Naïve Bayes Classifier," *18th International Conference on In Pattern Recognition*, Vol. 3 , pp. 1236-1239,2006.

[15] Choquet, G. Theory of capacities. *Annales de l'Institut Fourier*, 5, 131-295.

[16] Z. Wang and G. J. Klir, Fuzzy Measure Theory, *Plenum Press*, New York, 1992.

[17] Sugeno. M, "Theory of fuzzy integrals and its applications." *Ph.D. dissertation*, Tokyo Institute of Technology.

[18] Sugeno. M, "Fuzzy measures and fuzzy integrals: A survey." *Gupta, Saridis, and Gaines*, 89-102.

[19] Z. Wang, "the autocontinuity of set function and the fuzzy integral." *Journal of Mathematical Analysis and applications*, 99, 195-218.

[20] Z. Wang, "Asymptotic structural characteristics of fuzzy measure and their applications." *Fuzzy Sets and Systems*, 16,277-290,1985.

[21] P. R. Halmos, Measure Theory, *Van Nostrand*, New York, 1967.

[22] M. Grabisch, T. Murofushi and M. Sugeno (editors). Fuzzy Measures and Integrals: Theory and Applications, *Physica-Verlag*, 2000.

[23] T. Murofushi, M. Sugeno, and M. Machida, "Non-monotonic fuzzy measures and the Choquet integral,"*Fuzzy Sets and Systems*, vol. 64, no. 1, pp. 73-86, 1994.

[24] Z. Wang, "A new genetic algorithm for nonlinear multiregressions based on generalized Choquet integrals," *Proc. 12th IEEEIntern. Conf. Fuzzy Systems*, vol. 2, pp. 819-821, 2003.

[25] K. S. Leung, M. L. Wong, W. Lam, Z. Wang and K. Xu, "Learning nonlinear multiregression networks based on evolutionary computation," *IEEE Trans. On Systems, Man and Cybernetics*, Part B, vol. 32, no. 5, pp. 630-644, 2002.

[26] K. Xu, Z. Wang and K. S. Leung, "Classification by nonlinear integral projections," *IEEE Trans. Fuzzy Systems*, vol. 11, no. 2, pp. 187-201, 2003.

[27] Z. Wang, G. J. Klir, and W. Wang, "Monotone set functions defined by Choquet integral," *Fuzzy Sets and Systems*, vol. 81, no. 2, pp. 241-250, 1996.

[28] Richard A. Berk, Regression Analysis: A Constructive Critique, *Sage Publications*,2004.

[29] David A. Freedman, Statistical Models: Theory and Practice, *Cambridge University Press*,2005.

[30] R. Dennis Cook; Sanford Weisberg "Criticism and Influence Analysis in Regression", *Sociological Methodology*, Vol. 13. , pp. 313-361,1982.

[31] M. Grabisch, H. T. Nguyen and E. A. Walker, Fundamentals of Uncertainty Calculi with Applications to Fuzzy Inferences, *Kluwer Academic* Publishers, 1995.

[32] K. S. Leung, M. L. Wong, W. Lam, Z. Wang, "Discovering nonlinear-integral networks from databases using evolutionary computation and minimum description length principle," *Proc. 1998 IEEE Intern. Conf. Systems, Man and Cybernetics*, vol. 3. pp. 2354-2359, 1998.

[33] Z Wang, "A new model of nonlinear multiregressions by projection pursuit based on generalized Choquet integrals," *Proc. 2002 IEEE Intern. Conf. Fuzzy Systems*, vol. 2, pp. 1240-1204, 2002.

[34] Gunnar Ratsch, A Brief Introduction into Machine Learning, *Proceeding in 21st Chaos Communication Congress*, December. 2004.

[35] Hutter, M..: Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability, *Springer, Berlin*,2004.

[36] M. Grabisch, "The representation of importance and interaction of features by fuzzy measures," *Pattern Recognition Letters*, vol. 17. no. 6, pp. 567-575, 1996.

[37] M. Grabisch and J. M. Nicolas, "Classification by fuzzy integral: performance and tests," *Fuzzy Sets and Systems*, vol. 65, no. 2-3, pp. 255-271, 1994.

[38] L. Mikenina and H. -J. Zimmermann, "Improved feature selection and classification by the 2-additive fuzzy measure," *Fuzzy Sets and Systems*, vol. 107, no. 2, pp. 197-218, 1999.

[39] Keller, J., Yan, M. B. : Possibility expectation and its decision making algorithm.1st IEEE Int. Conf. *Fuzzy Systems*, pp 661-668.san Diago,1992.

[40] Wang, W., Wang, Z. Y., Klir.G. J.: Genetic algorithm for determining fuzzy measures from data. *Journal of Intelligent and Fuzzy Systems*, Vol. 6. , pp. 171-183,1998.

[41] Wang, Z. Y., Leung, K. S., Wang, J. : A genetic algorithm for determining nonadditive set functions in information fusion. *Fuzzy Sets and Systems*, Vol. 102. pp. 463-469,1999.

[42] D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning,*Addison-Wesley, New York* 1989.

[43] J. H. Holland, Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence, *London : MIT Press*, 1992.

[44] J. R. Koza, Genetic Programming, Cambridge, *MIT press*, 1994.

[45] D. E. Goldberg, "Real-coded genetic algorithms, virtual alphabets, and blocking," *Complex Systems*, vol. 5, no. 2, pp. 139-167, 1991.

[46] F. Herrera, M. Lozano and J. L. Verdegay, "Tuning fuzzy logic controllers by genetic algorithms," *International Journal of Approximate Reasoning*, vol. 12, pp. 299-315, 1995.

[47] X. Yao, "An empirical study of genetic operators in genetic algorithms," *Microprocessing and microprogramming*, vol. 38, pp.707-714, 1993.

[48] Brad L. Miller and David E. Goldberg, "Genetic Algorithms, Tournament Selection, and the Effects of Noise".

[49] N. J. Radcliffe, "Equivalence class analysis of genetic algorithms". *Complex Systems*, vol. 5, no. 2, pp. 183-205, 1991.

[50] S. Koziel and Z. Michalewicz, "Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization," *Evolutionary Computation*, vol. 7, no. 1, pp. 19-44, 1999.

[51] A. H. Wright, "Genetic algorithms for real parameter optimization," Foundations of Genetic Algorithms, (G.J.E. Rawlins editor), *Morgan Kaufmann Publishers*, pp. 205-218, 1991.

[52] K.S. Leung, Y.T. Ng, K.H. Lee, L.Y. Chan, K.W. Tsui, T. Mok, C.H. Tse, J. Sung.: Data Mining on DNA Sequences of Hepatitis B Virus by Nonlinear Integrals. *Proceedings Taiwan-Japan Symposium on Fuzzy Systems & Innovational Computing*, 3rd meeting, pp. 1-10. Japan, 2006.

[53] McLachlan, G. J.: Discriminant Analysis and Statistical Pattern Recognition. *Wiley*, New York,1992.

[54] Mika, S., Smola, A.J., Scho¨lkopf, B.: "An Improved Training Algorithm for Fisher Kernel Discriminants". *Proc. Artifical Intelligence and Statistics, AISTATS'01.* T. Jaakkaola and T. Richardson, eds. pp. 98-104, 2001.

[55] Merz C., Murphy, P.: UCI repository of machine learning databases.*[Online].Available:ftp//ftp.ics.uci.edu/pub/machine-learning-databases*, 1996.

[56] KS Leung, KH Lee, JinFeng Wang, Eddie YT Ng, Henry LY Chan, Stephen KW Tsui, Tony SK Mok, Chi-Hang Tse, Joseph JY Sung, Data Mining on DNA Sequences of Hepatitis B Virus. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 14 Jan. 2009.

[57] The MathWorks, *Natick*, MA, 2004.

[58] Data Mining Tools See5 and C5.0. Software available at http://www.rulequest.com/see5-info.html. May 2006.

[59] SAS Enterprise Miner (EM). Online: http://www.sas.com/technologies/analytics/datamining/miner/

[60] C.C Chang and C.J. Lin, LIBSVM : A library for support vector machines, Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm, 2001.

[61] C. Borgelt. Bayes Classifier Induction. Software available at http://fuzzy.cs.uni-magdeburg.de/ borgelt/bayes.html.

[62] Harry Zhang, The Optimality of Naïve Bayes, *Proceeding of 17th International FLAIRS Conference*, Florida, USA.

[63] D. Denneberg, Non-Additive Measure and Integral, *Kluwer Academic Publishers*, Dordrecht-Boston-London,1994.

[64] P. R. Halmos, Measure Theory, *Van Nostrand*, New York ,1967.

[65] E. Pap, Null-Additive Set Functions, *Kluwer Academic Publishers*, Dordrecht-Boston-London ,1995.

[66] Z. Wang and G. J. Klir, Generalized Measure Theory, *Springer*. New York, 2008.

[67] Z. Wang, K. S. Leung, and G. J. Klir, Integration on finite sets, *International Journal of Intelligent Systems*, 21, 1073-1092,2006.

[68] Z. Wang, K. S. Leung, and G. J. Klir, Applying fuzzy measures and nonlinear integrals in data mining, *Fuzzy Sets and Systems*, 156, 371-380, 2005.

[69] Z. Wang, K. S. Leung, M. L. Wong, J. Fang, and K. Xu, Nonlinear nonnegative multi-regressions based on Choquet integrals, *International Journal of Approximate Reasoning*, 25 , 71-87, 2000.

[70] Z. Wang, W. Li, K. H. Lee, and K. S. Leung, Lower integrals and upper integrals with respect to nonadditive set functions. *Fuzzy Sets and Systems*, 159 , 646-660, 2008.

[71] R. Yang, Z. Wang, P. A. Heng, and K. S. Leung, Fuzzified Choquet integral with fuzzy-valued integrand and its application on temperature prediction, *IEEE T. SMCB*, 38, No. 2 , 367-380, 2008.

[72] Deb, Kalyanmoy, "Multi-Objective Optimization using Evolutionary Algorithms," *John Wiley Sons, Ltd, Chichester*, England, 2001.

[73] Chan HLY, Hui AY, Wong ML, et al. Genotype C hepatitis B virus infection is associated with an increases risk of hepatocellular carcinoma. *Gut*, 53: 1494-1498, 2004.

[74] Henry L.Y. Chan, C.H. Tse, Eddie Y.T. Ng, K.S. Leung . K.H. Lee, K.W. Tsui, Joseph J. Y. Sung "Phylogenetic, Virological and Clinical Characteristics of Genotype C Hepatitis B Virus With Tcc

At Codon 15 Of The Precore Region," *Journal of Clinical Microbiology*. Vol. 44, No. 3, p. 681687, 2006.

[75] R. B. Potter and S. Draghici, A soft approach to predicting HIV drug resistance, Proc. of Pacific Symposium on Biocomputing, *PSB 2002, Kaua'i Marriott*, Kaua'i, Hawaii, 2002.

[76] A. Ciancio, A. Smedile, and M. Rizzetto, Identification of hbv dna sequences that are predictive of response to lamivudine therapy. *Hepatology*, 39:6473, 2004.

[77] Tomoyuki Sakamoto, Yasuhito Tanaka, Josephine Simonetti, Carla Osiowy, Malene L. Brresen, Anders Koch, Fuat Kurbanov, Masaya Sugiyama, Gerald Y. Minuk, Brian J. McMahon, Takashi Joh, and Masashi Mizokami, Classification of Hepatitis B Virus Genotype B into 2 Major Types Based on Characterization of a Novel Subgenotype in Arctic Indigenous Populations, *The Journal of Infectious Diseases*, 196:14871492, 2007.

[78] Herry LY Chan, Stephen KW Tsui, Eddie YT NG, Pete CH Tse, KS Leung , KH Lee, Tony Mok , Angeline Bartholomeuz, Thomas CC Au, Joseph JY Song, Epidemiological and Virological Characteristics of two subgroups of genotype C Hepatitis Virus. *Journal of Infect Disease*, vol. 191, pages 2022-2032, 2005.

[79] H. Almuallim and T. Dietterich, Learning boolean concepts in the presence of many irrelevant features. *Artificial Intelligence*, 69(1-2):179305, 1994.

[80] M. Dash and H. Liu, Feature selection for classification. *Intelligent Data Analysis*, 1997.

[81] G. John, R. Kohavi, and K. Pdlwfwe, Irrelevant features and the subset selection problem. *In Proceedings of the Eleventh International Conference on Machine Learning*, pages 121129, 1994.

[82] P. Langley, Selection of relevant features in machine learning. *In Proceedings of the AAAI Fall Symposium on Relevance*, pages 15, 1994.

[83] P. Pudil, J. Novovicoca, and J. Kittler, "Floating Search Methods in feature selection," *Pattern Recognition Letters*, vol. 15, pp. 1119-1125, Nov. 1994.

[84] Anil Jain and Douglas Zongker, "Feature selection: evaluation, application, and small example performance", *IEEE Transactions on Pattern analysis and machine intelligence*, vol. 19, No. 2, February 1997.

[85] W. Banzaf, P. Nordin, R. Keller, and F. Francone, Genetic Programming - An Introduction. *Palo Alto*, 1997.

[86] A. A. Freitas, A survey of evolutionary algorithms for data mining and knowledge discovery. In Ghosh A and Tsutsui S, editors, *Advances in Evolutionary Computation*, 2002.

[87] M.L. Raymer, W.F. Punch, E.D. Goodman, L.A. Kuhn, and A.K. Jain, Dimensionality reduce using genetic algorithms. *IEEE Transactions on Evolutionary Computing*, 4(2):164171, 2000.

[88] M.L. Wong and K.S. Leung, Learning recursive functions from noisy examples using generic genetic programming, *In Proceedings of the First Annual Conference*, pages 238-246. MA: MIT Press, Stanford, 1996.

[89] M.L. Wong and K.S. Leung, Genetic logic programming and applications, *IEEE Expert*, 10(5), pages 68-76, 1995.

[90] C.M. van der Walt and E. Barnard, Data characteristics that determine classifier performance. *Proceedings of the Sixteenth Annual Symposium of the Pattern Recognition Association of South Africa*, pages pp.160-165, 2006. Available [Online] http://www.patternrecognition.co.za

[91] M.L. Wong and K.S. Leung, Data Mining Using Grammar Based Genetic Programming and Applications, Kluwer Academic Publishers, *Genetic Programming Series*, Jan 2000.

[92] Mee Young Park, Trevor Hastie, L1-regularization path algorithm for generalized linear models, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 69 (4) , 659677 , 2007.

[93] Tomasz Laskus, Lian-Fu Wang, Marek Radkowski Hugo Vargas, Janusz Cianciara, Comparison of hepatitis B virus core promoter sequences in peripheral blood mononuclear cells and serum from patients with hepatitis B, *Journal of General Virology* , 78, 649653,1997.

[94] Won Kyoung Keum, Jee Youn Kim, Ja Young Kim, Sung Gil Chi, Hong Jung Woo,3,4 Sung Soo Kim, Joohun Ha and Insug Kang, Heterogeneous HBV mutants coexist in Korean hepatitis B patients, *EXPERIMENTAL and MOLECULAR MEDICINE*, Vol. 30, No 2, 115-122, June 1998.

[95] Thompson JD, Higgins DG, Gibson TJ. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through

sequence weighting, *position specific gap penalties and weight matrix choice.* Nucleic Acids Res, 22,467380, 1994.

[96] Kimura M. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *J Mol Evol*, 16:11120, 1980.

[97] Saitou N, Nei M. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol Biol Evol*, 4:40625, 1987 .

[98] Kumar S, Tamura K, Nei M. MEGA3: integrated software for molecular evolutionary genetics analysis and sequence alignment. *Brief Bioinform*, 5, 15063, 2004.

[99] Orito E, Ichida T, Sakugawa H, et al. Geographic distribution of hepatitis B virus (HBV) genotype in patients with chronic HBV infection in Japan. *Hepatology*, 34:5904, 2001.

[100] Bowyer SM, Sim JGM. Relationships within and between genotypes of hepatitis B virus at points across the genome: footprints of recombination in certain isolates. *J Gen Virol*, 81:37992. 2000.

[101] Beasley, R.P., L.Y. Hwang, C.C.Lin, C.S.Chien. Hepatocellular carcinoma and hepatitis B virus. A prospective study of 22 707 men in Taiwan. *Lancet* 2:1129-33. 1981.

[102] Kao, J.H., P.J. Chen, M.Y. Lai, D.S. Chen. Hepatitis B genotypes correlate with clinical outcome in patients with chronic hepatitis B. *Gastroenterology* 118:554-559, 2000.

[103] Sumi, H., O. Yokosuka, N. Seki, M. Arai, F. Imazeki, T. Kurihara, T. Kanda, K. Fukai, M. Kato, H. Saisho, Influence of hepatitis B virus genotypes on the progression of chronic liver disease. *Hepatology* 37:1926,2003.

[104] Yuen, M.F., Y. Tanaka, M. Mizokami, J.C. Yuen, D.K. Wong, H.J. Yuan, S.M. Sum, A.O. Chan, B.C. Wong, C.L. Lai. Role of hepatitis B virus genotypes Ba and C, core promoter and precore mutations on hepatocellular carcinoma: a case control study. *Carcinogenesis* 25:1593-8, 2004.

[105] Sugauchi, F., H. Kumada, H. Sakugawa, M. Komatsu, H. Niitsuma, H. Watanabe, Y. Akahane, H. Tokita, T. Kato, Y. Tanaka, E. Orito, R. Ueda, Y. Miyakawa, M. Mizokami. Two subtypes of genotype B (Ba and Bj) of hepatitis B virus in Japan. *Clin Infect Dis*, 38:1222-8. 2004.

[106] J.L. McGregor, K.J. Walsh and J.J. Katzfey, "Climate simulations for Tasmania," *Proc. 4th Int. Conf. Southern Hemisphere Meteorological Oceanography*, pp.514-515, 1993.

[107] Y. -P. Huang and T.-M. Yu, "The hybrid grey-based model for temperature prediction," *IEEE Trans. systems, Man, and Cybernetics*, Part B, vol. 27, no. 2, pp. 284-292, 1997.

[108] J. R. Hwang, S. M. Chen and C. H. Lee, "Handling forecasting problems using fuzzy time series," *Fuzzy Sets and Systems*, vol. 100, pp.217-228, 1998.

[109] R. Lee and J. Liu, "iJADE WeatherMan: A weather forecasting system using intelligent multiagent-based fuzzy neural network," *IEEE Trans. systems, Man, and Cybernetics*, Part C, vol. 34, no. 3, pp. 369-377, 2004.