Cancer Classification Using Gene Expression Data

Ying Lu Jiawei Han
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
Email: {yinglu, hanj}@uiuc.edu

Corresponding author: Ying Lu (yinglu@uiuc.edu), DCL, Department of Computer Science, 1304 Springfield Ave., University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA. Phone: (217) 244-3570 Fax: (217) 244-6500

Abstract

The classification of different tumor types is of great importance in cancer diagnosis and drug discovery. However, most previous cancer classification studies are clinical-based and have limited diagnostic ability. Cancer classification using gene expression data is known to contain the keys for addressing the fundamental problems relating to cancer diagnosis and drug discovery. The recent advent of DNA microarray technique has made simultaneous monitoring of thousands of gene expressions possible. With this abundance of gene expression data, researchers have started to explore the possibilities of cancer classification using gene expression data. Quite a number of methods have been proposed in recent years with promising results. But there are still a lot of issues which need to be addressed and understood.

In order to gain deep insight into the cancer classification problem, it is necessary to take a closer look at the problem, the proposed solutions and the related issues all together. In this survey paper, we present a comprehensive overview of various proposed *cancer classification* methods and evaluate them based on their computation time, classification accuracy and ability to reveal biologically meaningful gene information. We also introduce and evaluate various proposed *gene selection* methods which we believe should be an integral preprocessing step for cancer classification. In order to obtain a full picture of cancer classification, we also discuss several issues related to cancer classification, including the *biological significance vs. statistical significance* of a cancer classifier, the *asymmetrical classification errors* for cancer classifiers, and the *gene contamination* problem.

Keywords: cancer classification, gene expression data.

1 Introduction

Cancer research is one of the major research areas in the medical field. Accurate prediction of different tumor types has great value in providing better treatment and toxicity minimization on the patients.

Previously, cancer classification has always been morphological, and clinical based. These conventional cancer classification methods are reported to have several limitations [Azu00] in their diagnostic ability. It has been suggested that specifications of therapies according to tumor types differentiated by pathogenetic patterns may maximize the efficacy of the patients [Aea00, GST+99, VDBea02, PTGea02, Zea01, Sea01, DGB02, VJ02, Aea00, DPBea96]. Also, the existing tumor classes has been found to be heterogeneous and comprises of diseases that are molecularly distinct and follow different clinical courses.

In order to gain a better insight into the problem of cancer classification, systematic approaches based on global gene expression analysis have been proposed. The expression level of genes are known to contain the keys to address fundamental problems relating to the prevention and cure of diseases, biological evolution mechanisms and drug discovery. The recent advent of microarray technology has allowed the simultaneous monitoring of thousands of genes, which motivated the development in cancer classification using gene expression data [GST+99, STM+00, LA01, NR02, Ber00]. Though still in its early stages of development, results obtained so far seemed promising .

Different classification methods from statistical and machine learning area have been applied to cancer classification, but there are some issues that make it a nontrivial task. The gene expression data is very different from any of the data these methods had previously dealt with. First, it has very high dimensionality, usually contains thousands to tens of thousands of genes. Second, publicly available data size is very small, all below 100. Third, most genes are irrelevant to cancer distinction. It is obvious that those existing classification methods were not designed to handle this kind of data efficiently and effectively. Some researchers proposed to do gene selection prior to cancer classification. Performing gene selection helps to reduce data size thus improving the running time. More importantly, gene selection removes a large number of irrelevant genes which improves the classification accuracy [GWB+00]. Due to the important role it plays in cancer classification, we also study the various proposed gene selection methods in this paper.

Besides gene selection, there are several issues related to cancer classification that are of great concern to researchers. These issues are derived from the biological context of the problem, and the medical importance of the result. These issues include statistical relevance vs. biological relevance of cancer classifiers, asymmetrical classification errors and the gene contamination problem. We believe that in order to have an in-depth understanding of the problem, it is necessary to study both the problem and its related issues and look at them all together.

The paper is organized as follows: In Section 2, we give a biological background information and problem definition. In Section 3, we first give a detailed description of the proposed gene classification methods, followed by an evaluation of the methods and end with a unified view of the methods. We present the description and evaluation of the various kinds of gene selection methods in Section 4. The related issues of cancer classification are discussed in Section 5. We conclude the paper in Section 6.

2 Background Information and Problem Statement

In this section, we first provide some basic biological background knowledge . Readers familiar with the background can skip this part. Then we introduce some terminologies and define the problem of cancer classification using gene expression data. Some classification challenges that are unique to the gene expression data are stated at the end.

2.1 Biological Background Information

We first give some fundamental knowledge in molecular biology. *Cells* are the fundamental working units of every living system. All the instructions needed to direct their activities are contained within the chemical *deoxyribonucleic acid* or DNA. A DNA molecule is a double-stranded polymer composed of four basic molecular units called *nucleotides*. DNA from all organisms is made up of the same chemical and physical components. Each *nucleotide* comprises a phosphate group, a deoxyribose sugar and one of the four *nitrogen bases*. The nitrogen bases are adenine(A), guanine(G), cytosine(C) and thymine(T). The halves of the double helix structures are held together by the hydrogen bonds between the nitrogen bases through the *base pairs* : A with T, C with G. Each strand in the DNA double helix can be seen as a chemical "mirror image" of the other. If there is an A on one strand, there will always be a T opposite it on the other, if there is a C on one strand, then its partner will always be G.

DNA sequence is a particular arrangement of the base pairs in the DNA strand, e.g., CTTGAATCCCG. The arrangement spells out the exact instructions required to create a particular organism with its own unique characteristics. DNA is called the blueprints of all living organisms, since the components of the strand encode all the information necessary for building and maintaining life, from simple bacteria to remarkably complex human beings. The unusual double helix structure of DNA molecules gives DNA special properties. These properties allow the information stored in DNA to be preserved and passed from one cell to another and from parents to offspring. When a cell divides to form two new daughter cells, DNA is *replicated* by untwisting the two strands of the double helix and using each strand as a template for building its chemical mirror image.

The entire DNA sequence that codes for a living thing is called its *genome*. The genome is an organism's complete set of DNA. Genomes vary widely in size: the smallest known genome for a free-living organism (a bacterium) contains about 600,000 DNA base pairs, while human and mouse genomes have about 3 billion DNA base pairs. Except for mature red blood cells, all human cells contain a complete genome. The genome does not function as one long sequence, but is divided into a set of genes.

A *gene* is a small, defined section of the entire genomic sequence, each has a specific and unique purpose. There are three types of genes, namely the protein-coding genes, the RNA-specifying genes and the untranscribed genes. Protein-coding genes are templates for generating molecules called proteins. RNA-specifying genes are templates for chemical machines. The RNA-specifying genes provides the template for the synthesis of a variety of RNA molecules. Untranscribed genes are regions of genomic DNA that have some functional purpose but do not achieve that purpose through transcription or

translation for the creation of new molecules.

2.1.1 Gene Expression and DNA Microarray Technology

DNA act as a template for making copies of itself but also as a blueprint for a molecule called RNA(ribonucleic acid). The genome provides a template for the synthesis of a variety of RNA molecules. The main types of RNA are messenger RNA(mRNA), transfer RNA(tRNA), and ribosomal RNA(rRNA).

The *expression* of the genetic information stored in the DNA molecule occurs in two stages: (i) *transcription* stage where the DNA molecule is transcribed into mRNA, (ii) translation stage where mRNA is translated into the amino acid sequences of the proteins that perform various cellular functions.

The process of transcribing a gene's DNA sequence into RNA is called *gene expression*. A gene's expression level indicates the approximate number of copies of that gene's RNA produced in a cell and it is correlated with the amount of the corresponding proteins made. It has been shown that specific patterns of gene expression occur during different biological states such as embryogenesis, cell development, and during normal physiological responses in tissues and cells [Rus00]. Thus the expression of a gene provides a measure of activity of a gene under certain biochemical conditions.

It is known that certain diseases, such as cancer, are reflected in the change of the expression values of certain genes. Normal cells can evolve into malignant cancer cells through a series of mutations in genes that control the cell cycle, apoptosis and genome integrity, etc. [BDBF$^+$00]. Studies on the use of DNA microarrays have supported the effectiveness of gene expression patterns for identifying different gene functions and cancer diagnosis.

Microarrays and serial analysis of gene expressions are two recent technologies for measuring the thousands of genome-wide expression values in parallel. The former, which consists of cDNA microarrays [SSDB95] and high-density oligonucleotide arrays [LDB$^+$96], measures the relative levels of mRNA abundance between different samples, while the latter measures the absolute level.

cDNA microarray analysis is a relatively new molecular biology method that expands on classic probe hybridization methods to provide access to thousands of genes at once. Therefore allowing the recording of expression levels of thousands of genes simultaneously. cDNA microarrays consists of thousands of individual DNA sequences printed in a high density array on a glass microscope. Each data point produced by a DNA microarray hybridization experiment represents the ratio of expression levels of a particular gene under two different experimental conditions. The result, from an experiment with m genes on a single chip, is a series of m expression level ratios. The numerator of the ratio is the expression level of the gene in the varying conditions of interest, and the denominator is the expression level of the gene in some reference condition.

Serial analysis of gene expression, or SAGE, is a technique designed to take advantage of high-throughput sequencing technology to obtain a quantitative profile of cellular gene expression. The SAGE technique does not measure the expression level of a gene, but quantifies a "tag" which represents the transcription product of that gene. A tag in this case is a nucleotide sequence of defined length.

The original length of the tag was nine bases, current SAGE protocols produce a ten to eleven base tag. The data product of the SAGE technique is a list of tags, with their corresponding count values, which is a digital representation of cellular gene expression.

Most existing cancer classification methods uses DNA microarray expression data. All the proposed methods in this paper tested their performance on such data.

2.2 The Cancer Classification Problem

Classification problem has been extensively studied by researchers in the area of statistics, machine learning and databases. Many classification algorithms have been proposed in the past, such as the decision tree methods, the linear discrimination analysis, the bayesian network, etc. For the last few years, researchers have started paying attention to the cancer classification using gene expression [GST+99, BDBF+00]. Studies have shown that gene expression changes are related with different types of cancers.

Most proposed cancer classification methods are from the statistical and machine learning area, ranging from the old nearest neighbor analysis, to the new support vector machines. There is no single classifier that is superior over the rest. Some of the methods only works well on binary-class problems and not extensible to multi-class problems, while others are more general and flexible. One thing to note for most of those proposed algorithms on gene classification is that the authors are only concerned with the accuracy of the classification and did not pay much attention to the running time(in fact, most gene classifiers proposed are quite computationaly expensive).

Cancer classification using gene expression data stands out from the other previous classification data due to its unique nature and application domain. Through this survey, we hope to gain some insight into the problem of cancer classification in aid of further developing more effective and efficient classification algorithms.

2.2.1 Terminologies and Problem Statement

We define and introduce some terminologies and notations that we will use throughout the section for the problem of cancer classification using gene expression data, termed *cancer classification*, for brevity.

Let $X_1, X_2, \ldots, X_m$ be random variables for genes $G_1, G_2, \ldots, G_m$ respectively, where X_i has domain $\mathrm{dom}(X_i)$ which is the range of expression values for gene G_i. Let C be the random variable for the class labels, and $\mathrm{dom}(C) = \{1, \ldots, K\}$, where K denotes the total number of classes.

Let $t = \{t.X_1, t.X_2, \ldots, t.X_m\}$ denotes a size m tuple of expression values for m genes. Let $T = \{(t_1, c_1), (t_2, c_2), \ldots, (t_n, c_n)\}$ denoting a *training set* of n tuples, where $i = \{1, 2, \ldots, n\}$,$c_i \in \mathrm{dom}(C)$ is the class label of tuple t_i, Let the test set be $S = \{t_1, t_2, \ldots, t_l\}$ where l is the size of the test set. A *classifier* is a function $Class$ with two arguments, T and s, where T denotes the training samples and s is a testing sample. Function $Class$ returns a class prediction for sample s. The *classification accuracy* is defined as the number of correct predictions made by the classifier on a set of testing tuples

using the function $Class$ trained on the training tuples.

Cancer Classification Problem: Given a training set $T = \{(t_1, c_1), (t_2, c_2), \ldots, (t_n, c_n)\}$, where t_is are independent m-dimensional random data tuples of gene expression values, m is the total number of genes, $t_i = (t_i.X_1, t_i.X_2, \ldots, t_i.X_m)$, $m \gg n$ and $c_i \in \text{dom(C)}$ is the class label of the ith tuple. Given a test set $S = \{s_1, s_2, \ldots, s_l\}$. Each s_i is a gene expression data tuple of length m. Each s_i is in the form of $\{s_i.X_1, s_i.X_2, \ldots, s_i.X_m\}$, where x_j is the expression value of gene j. Find a classification function $Class$, that gives maximal classification accuracy on S.

2.2.2 The Challenges

There have been extensive studies done in the past on the classification problem by the statistical, machine learning and database research community. But gene classification as a new area of research poses new challenges due to its unique problem nature. Here we elaborate on some of these challenges.

First challenge comes from the unique nature of the available gene expression data set. Though the successful application of cDNA microarrays and the high-density oligonucleotides have made fast simultaneous monitoring of thousands of gene expressions possible and inexpensive, the publicly available gene expression data set size still remains small. Most of these data, such as the Colon tissue samples, the Leukemia data set, etc., has sample size below 100, On the contrary, the attribute space, or the number of genes, of the data is enormous: there are usually thousands to hundred thousands of genes present in each tuple. If the samples are mapped to points in the attribute space, then the samples can be viewed as very sparse points in a very high dimensional space. Most existing classification algorithms were not designed with this kind of data characteristics in mind. Such a situation of sparseness and high dimensionality is a big challenge for most classification algorithms. Overfitting is a major problem due to the high dimension, while the small data size makes it worse. Also, with so many genes in the tuple, it will be a big challenge on the computation time. Therefore, developing an effective and efficient classification algorithm for cancer classification is not an easy task.

Second challenge comes from the presence of noise inherent in the data set. These noise can be categorized into *biological noise* and *technical noise* [BDBF$^+$00]. Biological noise refers to the noise introduced by genes that are not relevant for determination of the cancer classes. In fact, most of the genes are not related to the cancer classes. Technical noise refers to the noises that are introduced at the various stages of data preparation whereas biological noise are associated with the non-uniform genetic backgrounds of the samples or the misclassification of the samples. Coupled with small sample size, the presence of noise makes accurate classification of data difficult.

Third challenge involves dealing with a huge number of irrelevant attributes(genes). Though irrelevant attributes are present in almost every kind of data sets researchers have dealt with previously, but the ratio of irrelevant attributes to the relevant attributes is not as huge as that in the gene expression data. In most gene expression data set, the number of relevant genes only occupy a small portion of the total number of genes. Most genes are not cancer related. The presence of these irrelevant genes interferes with the discrimination power of those relevant attributes. This not only incurs extra computation time in both the training and testing phase of the classifier, but also increases the classification

difficulty. One way to handle this is to incorporate a gene selection mechanism to select a group of relevant genes. Then cancer classifiers can be built on top of these selected genes. Another way is to incorporate the selection of relevant genes inside the training phase of the classifier. Performing cancer classification efficiently and effectively using either way is a nontrivial process, thus requiring further exploration.

Fourth challenge arises from the application domain of cancer classification. Accuracy is important in cancer classification, but it is not the only goal we want to achieve. Biological relevancy is another important criterion, since any biological information revealed during the process can help in further gene function discovery and other biological studies. Some useful information can be gained from the classification process is the determination of the genes that work as a group in determining the cancerous tissues or cells or the genes that are under-expressed or over-expressed in certain tissues or cells. All these would help biologists in gaining more understanding about the genes and how they work together and interact with each other. Therefore biologists are more interested in classifiers that not only produce high classification accuracy but also reveal important biological information.

2.3 Publicly Available Cancer Data Sets from cDNA Microarray

Currently, there is no central repository for human expression data. Below are several publicly available gene expression data from DNA microarray that are widely used by reseachers for cancer classification experiments.

The first data set is the *Colon cancer* data(http://microarray.princeton.edu/oncology). This data consists of 62 samples of colon epithelial cells from colon-cancer patients. The samples consists of tumor bipsies collected from tumors, and normal biopsies collected from healthy part of the colons of the same patient. The number of genes in the data set is more than 2000.

The second data set is the *Ovarian cancer* data. It consists of 32 samples, 15 of which are ovary biopsies of ovarian carcinomas, 13 of which are biopsies of normal ovaries and 4 samples belong to other tissues. 28 of the samples are labeled.

The third data set is the *Leukemia* data(http://www.genome.wi.mit.edu/MPR). This data consists of 72 samples. The samples consists of two types of leukemia, 25 of AML and 47 of ALL. The samples are taken from 63 bone marrow samples and 9 peripheral blood samples. There are 7192 genes in the data set.

The fourth data set is the *Lymphoma* data set(http://genome-www.stanford.edu/lymphoma). This data consists of 81 samples with 4,482 genes. In the 81 samples, 29 of them belong to class 1, 9 sample belong to class 2 and 43 samples belong to class 3.

The fifth data set is the *NCI* data set. This data consists of 60 samples of 9,703 genes from the National Cancer Institute(NCI)'s anti-cancer drug screen. The 60 samples belong to a variety of classes: 7 belongs to breast class, 5 belongs to central nervous system class, 7 belongs to the colon class, 6 belongs to leukemia class, 8 in melanoma class, 9 in non-small-cell-lung-carcinoma class, 6 in ovarian class, 2 in prostate class, 9 in renal class and 1 of unknown class.

There is another data set from NCI. It consists of 218 normal tissue samples and 90 cancerous tissue samples spanning across 14 different cancer types. There are 16,063 genes in this data set.

3 Cancer Classification Methods

In this section we first give a detailed description of some common classification methods used for cancer classification. We then follow with a preliminary evaluation of the methods based on their performance and biological relevance. we hope that the addressing of some of the important issues in cancer classification would assist in developing better techniques in the future. We conclude the section with a summarization of the methods.

3.1 Fisher's Linear Discriminant Analysis(FLDA)

FLDA was originally proposed and applied by [Fis36]. It is a nonparametric method that finds a *projection matrix* P which reshapes the data set to maximize the class separability. *Class separability* is defined to be the ratio of the between-class scatter matrix to the within-class scatter matrix. This projection defines features that are optimally discriminating.

Let $\bar{x_i}$ be a set of N column vectors of dimension D. The *mean* of the data set is $\bar{\mu}_x = \frac{1}{N}\sum_{i=1}^{N}\bar{x}_i$. Suppose there are K classes, $c_1, c_2, \ldots, c_K$. Then the *mean of class* k having N_k members is $\bar{\mu}_{xk} = \frac{1}{N_k}\sum_{\bar{x}_i \in C_k}\bar{x}_i$.

Therefore, the between-class scatter matrix can be defined as:

$$S_e = \sum_{k=1}^{K} N_k(\bar{\mu}_{xk} - \bar{\mu}_k)(\bar{\mu}_{xk} - \bar{\mu}_x)^T$$

And the within-class scatter matrix defined as :

$$S_n = \sum_{k=1}^{K}\sum_{\bar{x}_i \in C_k}(\bar{x}_i - \bar{\mu}_{xk})(\bar{x}_i - \bar{\mu}_{xk})^T$$

The transformation matrix that repositions the data to be most separable is the matrix P that maximizes $\frac{det(P^T S_e P)}{det(P^T S_n P)}$.

Let $\bar{w}_1, \bar{w}_2, \ldots, \bar{w}_D$ be the generalized eigenvectors of S_e and S_n. Then $P = [\bar{p}_1, \bar{p}_2, \ldots, \bar{p}_D]$. This gives a projection space of dimension D. A projection space of dimension $d \leq D$ can be defined by using the generalized eigenvectors with the largest d eigenvalues to give $P_d = [\bar{p}_1, \bar{p}_2, \ldots, \bar{p}_d]$. The projection of vector $\bar{x}$ into a subspace of dimension d is then $\bar{y} = W_d^T\bar{x}$. Thus the generalized eigenvectors are the eigenvectors of $S_e S_n^{-1}$.

[DFS00] applied FLDA to the cancer classification problem. Given a training set T of size n, each tuple is in the form (t_i, c_i), where t_i is in the form $(t_i.X_1, t_i.X_2, \ldots, t_i.X_m)$ which is a vector of the expression values of m genes in tuple i and c_i is the class label associated with t_i. t_is can be viewed as an $n \times m$ matrix, M, of gene expression values, where row i corresponds to the ith tuple and column

j corresponds to the expression values of gene j in the n samples. This method tries to find the linear combination Ma of the columns of M that has large ratio of *between-class* sum of squares to *within-class* sum of squares, where a is the transformation matrix.

Since S_e and S_n are the between-class scatter matrix and within-class scatter matrix respectively, Ma has the ratio of between-class sum of squares to within-class sum of squares given by $a'S_ea/a'S_na$.

The extreme values of $a'S_ea/a'S_na$ is obtained from the eigenvalues and eigenvectors of the matrix $S_n^{-1}S_e$. $S_n^{-1}S_e$ has at most $h = min(K-1, m)$ non-zero eigenvalues, $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_h$, and corresponding linearly independent eigenvectors,$v_1, v_2, \ldots, v_h$. Note that K denotes the number of classes present in the data, and m is the attribute space. Then, for any sample $\boldsymbol{t}$, the *discriminant variables* are defined to be $u_k = \boldsymbol{t}\boldsymbol{v}_l$, where $l = 1, 2, \ldots, h$ and $\boldsymbol{v}_l$ maximizes $a'S_ea/a'S_na$.

Let $s = (s.X_1, s.X_2, \ldots, s.X_m)$ be a test sample, where $s.X_i$, $i = 1, 2, \ldots, m$, denotes the expression value of gene i. Let $\bar{c}_k$ be a $1 \times m$ vector of average expression values of m genes in training tuples belonging to class k. Let $\bar{c}_k$ denotes the vector of average gene expression values for tuples in class k. The correlation between s and each class is measured using the squared Euclidean distance of s and $\bar{c}_k$, denoted as $d_k(\boldsymbol{s})$, where

$$d_k(\mathbf{s}) = \sum_{l=1}^{h}((\boldsymbol{s} - \bar{c}_k)\boldsymbol{v}_l)^2$$

Class k is assigned to s if the Euclidean distance between s and $\bar{c}_k$ is minimum.

Formally, given training samples T and a test sample s, the FLDA classifies s using the following classification function:

$$Class(T, \boldsymbol{s}) = argmin_k d_k(\boldsymbol{s})$$

3.2 Weighted Voting of Informative Genes - GS Method

The weighted voting method is proposed by Golub and Slonim et al. [GST+99, STM+00] for classifying binary class data. The GS method is a correlation based classifier. The assignment of classes is based on the weighted voting of the expression values of a group of "informative genes" in the test tuple.

The informative genes are genes that have high correlation with the class labels. Let the expression values of gene g in n training samples be represented by an expression vector $g = (e_1, e_2, \ldots, e_n)$, where e_i denotes the expression value of g in tuple i. Let vector $c = (c_1, c_2, \ldots, c_n)$ be the class vector denoting the classes of tuple i. Let $(\mu_1(g), \sigma_1(g))$ and $(\mu_2(g), \sigma_2(g))$ denote the mean and the standard deviation of the log_{10} of the expression values of g in class 1 and class 2 respectively. Then, the level of correlation, $P(g, c)$, between the expression values of gene g and the class vector c is measured using "signal-to-noise" ratio(SNR).

$$P(g, c) = (\mu_1(g) - \mu_2(g))/(\sigma_1(g) + \sigma_2(g))$$

Intuitively, this metric favors genes with expression values that span a big range, has small variation within the same class and big variation between different classes. The value of $|P(g, c)|$ is proportional to the correlation between the gene expression vector and the class vector. The sign of $P(g, c)$ denotes

which of the two classes the gene is more correlated with and the magnitude denotes the degree of correlation. Positive P-values denotes higher correlation with class 1 and negative P-values denotes higher correlation with class 2. The larger the magnitude, the stronger the correlation.

The "informative genes", IG, are selected as follows: let L be the user input parameter for the number of informative genes to be selected. Then the GS method selects $L/2$ genes having the highest positive P values and $L/2$ genes having the highest negative values.

For each $g \in IG$, define parameters (a_g, b_g), where $a_g = P(g, c)$, $b_g = (\mu_1(g) + \mu_2(g))/2$. a_g reflects the correlation of the expression values of g in the training data with the classes. b_g denotes the average of the mean log_{10} expression values of g of training tuples in the two classes. Let μ and σ denote the mean and the standard deviation of the expression values of gene g in the training tuples. Given a test tuple s, where $s = (s_1, s_2, \ldots, s_m)$. The class label of s is determined as follows: For each gene $g \in IG$ with expression value in s denoted by s_g, the *normalized* log_{10} expression value of g is defined as $Nor_g = log_{10}((s_g - \mu)/\sigma)$. Define the *vote* of gene g as $v_g = a_g(Nor_g - b_g)$, where the sign of the vote indicates the class(positive for class 1 and negative for class 2).

Intuitively, each "informative gene" casts a "weighted" vote for one class, where the magnitude depends on the expression level of the gene in the test tuple and the degree of correlation of that gene has over the training set.

The total vote for class 1, V_1, by IG is the sum of all the positive votes, and the total vote for class 2, V_2, is the sum of all the absolute values of the negative votes. Let V_{win} be the total vote of the class that has the higher total votes, and V_{lose} be the total vote of the class with lower total votes. Then the *prediction strength*, PS, of the vote cast by IG is defined as $PS = (V_{win} - V_{lose})/(V_{win} + V_{lose})$. PS denotes the relative margin on victory over the vote.

A "prediction strength threshold", pst, is used to determine if the prediction of the weighted voting is strong enough to assign the majority class to the test tuple. If $PS \geq pst$, then the winning class is assigned to be the class label of s, otherwise, the weighted voting is considered to be too weak to assign the test sample to the voted class, thus assigning "Uncertain" as the class label to the test tuple.

3.3 Artificial Intelligence Approaches

Below we describe the Naive Bayesian classifier and the artificial neural network method.

3.3.1 Probabilistic Induction: Naive Bayes(NB) Method

In general, Naive Bayes method uses probabilistic induction to assign class labels to test tuples, assuming independence among the attributes.

[KSHR00, FNP00] used the Naive Bayes algorithm for gene classification. In applying Naive Bayes method to gene classification, the method models each class as a set of Gaussian distributions: one for each gene from the training samples. Let K be the number of classes, and C_k denotes class k. Each class C_k is modeled using a set of Gaussian distributions, one for each gene in the training data set:

Then C_k is given by the formula:

$$C_k = \{C_k^1, C_k^2, \ldots, C_k^m\}$$

where C_k^i, is the Gaussian distribution of class k for gene i.

Given a test tuple, s, the class label of s, is obtained as:

$$class(s) = argmax_i^m(\sum_{g=1}^{m} logP(s_g|C_i^g))$$

Let μ_i^g amd σ_i^g be the mean and standard deviation of the Gaussian distribution for the class i distribution for gene g. Since $p(s_g|C_i^g)$ is proportional to $(1/\sigma_i^g)^{-0.5}((s_g - \mu_i^g)/\sigma_i^g)_2$, the class label is given by the function:

$$class(s) = argmax_i^m \sum_{g=1}^{m}[-log(\sigma_i^g) - 0.5((s_g - \mu_i^g)/\sigma_i^g)_2]$$

3.3.2 Neural Networks

[KWR+01] used neural networks for cancer type prediction. The method consists of three major steps: principle component analysis, relevant gene selection and artificial neural network prediction.

Principle component analysis [Jol86] is used for dimensionality reduction which helps to avoid "overfitting" error in the supervised regression model. They observed that inclusion of class labels into the reduction process does not provide optimal performance but introduces bias in the data. Thus, class labels are excluded from the dimensions that undergo reduction.

Only a set of relevant genes are selected from the group of genes in the expression profiles for the training of the neural networks. A model dependent analysis method is used for checking the relevancy of each genes, which is defined through sensitivity function. For a data set of N samples and K classes denoted as $c_1, c_2, \ldots, c_k$, the sensitivity of a gene g_i with respect to the class labels is defined as

$$S_i = \frac{1}{N}\frac{1}{K}\sum_{j=1}^{N}\sum_{m=1}^{K}\left|\frac{\partial c_m}{\partial g_i}\right|$$

This formula gives the importance of a gene with respect to the total classification. In addition, they also specified sensitivity of each gene, g_i, with respect to each class, c_j, defined as

$$S_{ij} = \frac{1}{N}\frac{1}{K}\sum_{m=1}^{N}\left|\frac{\partial c_j}{\partial g_i}\right|$$

where c_j is the jth class label and g_i is the ith gene. For each S_{ij}, they also defined a sign that signals if the largest contribution to the sensitivity is due to positive or negative terms. A positive sign implies that increasing the expression level of the gene increases the possibility of the sample belonging to this cancer class and vice versa for the negative sign. The S_i and S_{ij} values of genes are calculated, and genes are ranked both according to their importance with respect to the total classification (termed

as total rank) and to their importances with respect to each individual cancer class (termed as separate rank). Based on the separate rank, the genes are then classified according to the cancer class in which they are highly expressed.

While deciding the number of genes to be selected for the classification process, it was observed that selection of 96 genes gives the best performance for the data set they used(88 samples of 6567 genes in which 63 are used in the training process and 25 used in the test).

The class prediction was done using an Artificial Neural Network(ANN) classifier[Bis95]. The ANN classifier consists of linear perceptrons of 10 input nodes, which corresponds to 10 principle components, and 4 output nodes, which corresponds to the 4 different class labels in the input data. In total, they used 44 parameters. 3-fold cross validation were used for the prediction procedure. Samples were shuffled and split into 3 equal sized groups where 2 of them were used as the training set and 1 was used as the testing set. The experiment was repeated for 1250 times by random shuffling the samples. The class label of each test sample was assigned using majority voting of the results obtained for each test sample in the 1250 tests. To allow the rejection of assigning a class to a test sample, they defined a distance function, d_k, which measures the distance from a sample to the ideal vote for each cancer class

$$d_k = 1/2 \sum_{i=1}^{k} (o_i - \delta_{i,k})^2$$

where k is the cancer class, o_i is the average vote for cancer class i and $\delta_{i,k}$ is unity if i corresponds to class k and 0 otherwise. The distance is normalized such that the distance between two ideal samples belonging to different classes is unity. For each class, the empirical probability distribution of its distance was generated. They defined the 95th percentile of the probability distribution as the cutoff value for confidence. Samples outside of the 95th percentile of probability distribution of the classes were not assigned any class.

3.4 Decision Tree - Recursive Partitioning

Decision tree, also known as classification trees, is a well know classification method[BFOS84]. It has been widely used in classification applications and many extensions/variations have been proposed[RS98, Utg89, SAM96, GGRL99].

A decision tree consists of a set of internal nodes and leaf nodes. The internal nodes are associated with a splitting criterion which consists of a splitting attribute and one or more splitting predicates defined on this attribute. The leaf nodes are labeled with a single class label. The construction of the decision tree is usually a two-phase process. In phase 1, the growing phase, an overgrown decision tree is built from the training data. The splitting criterion at each internal node is chosen to split the data sets into subsets that have better class separability, thus minimizing the misclassification error. In phase 2, the pruning phase, the tree is pruned using some heuristics to avoid overfitting of data which tends to introduce classification error on the test data.

[ZYSX01] proposed a recursive partitioning cancer classification method based on decision tree. This method constructs a binary decision tree. A purity based entropy function is used to determine

the splitting criterion at each internal node. The function is defined as

$$Plog(P) + (1 - P)log(1 - P)$$

where P is the probability of a tuple being normal. During the process of tree construction, this function is applied to find the best gene to split and the best splitting criterion for the chosen gene. This is done by test each unused gene on all of its possible splitting points using the purity entropy function and select the one that gives the best result.

For testing the classification accuracy, they used 5-way *localized* cross-validation to reduce the chances of overfitting due to small data set size. The localized procedure keeps the same splitting gene in each internal node, but does cross validation on different splitting point of that gene. The performance test was done on the colon tissue data set from NCI and the result of the cross validation showed a misclassification rate of 6-8%.

3.5 Similarity Based Methods - NN and CAST

Here, we describe two multiclass classification methods based on similarity measure. Compared to the other methods, these two methods are less prone to noise and bias in the data But they have the disadvantage of not being able to scale well.

3.5.1 Nearest Neighbor Analysis

This method is based on a distance metric between the testing tuples and the training tuples. The main idea of the method is for each testing sample s, find one training sample t with most similar expression value, according to a distance measure. The class label of t is then assigned to s. The distance metric can be any similarity measure based on attribute values, for example, the Pearson correlation function, the Euclidean distance function, etc.

[BDBF$^+$00] used the Pearson correlation as a measure of similarity. Let $\boldsymbol{s}$ be a testing sample, and $\boldsymbol{t}$ be a training sample. Then $\boldsymbol{s}$ and $\boldsymbol{t}$ can be viewed as vectors of m gene expression values. Let $E(\boldsymbol{x})$ and $E(\boldsymbol{x})$ represent the expected value and the variance of a vector $\boldsymbol{x}$, respectively. Then the Pearson correlation function between two vectors, $\boldsymbol{s}$ and $\boldsymbol{t}$, is given by:

$$P(\boldsymbol{s},\boldsymbol{t}) = \frac{E((\boldsymbol{s}_i - E(\boldsymbol{s}))(\boldsymbol{t}_i - E(\boldsymbol{t})))}{\sqrt{Var(\boldsymbol{s})Var(\boldsymbol{t})}}$$

Given a testing sample s, and a set of training tuples T containing pairs of the form (t_i, c_i) where t_is are the expression values of genes and c_i is the class label of the tuple, the Pearson correlation function is evaluated for every (s, t_i) pair, where $i \in \{1 \dots n\}$. The tuple t that has most similar expression value with s is the one that maximizes P, given by $argmax_i^n(P(s, t_i))$. The class label of t is then assigned to s.

Therefore, the nearest neighbor classifier can be formally expressed as:

$$Class(T, s) = class(argmax_i P(s, t_i))$$

where *class* returns the class of the training tuple that has the highest P value.

The Nearest Neighbor(NN) method can be extended to the K-Nearest Neighbor(KNN) method, as proposed by [EH51] and applied to gene classification by [DFS00]. KNN differs from NN that the class label of a testing sample s is assigned using majority vote from K training tuples that are most similar to s according to a distance measure function. In [DFS00] a correlation metric, R, is used to measure the similarity between pairs of samples. The correlation metric R is given by

$$R(\boldsymbol{s}, \boldsymbol{t}) = \frac{\sum_{j=1}^{m}(\boldsymbol{s}_j - \bar{\boldsymbol{s}})(\boldsymbol{t}_j - \bar{\boldsymbol{t}})}{\sqrt{\sum_{j=1}^{m}(\boldsymbol{s}_j - \bar{\boldsymbol{s}})^2}\sqrt{\sum_{j=1}^{m}(\boldsymbol{t}_j - \bar{\boldsymbol{t}})^2}}$$

The classification proceeds in two steps. In first step, for each test sample s, K training samples with highest similarity to s are picked using the correlation metric R. In step two, for each test sample s, its class is determined using majority voting of the classes of its K most similar neighbors.

Obviously the value of K affects the performance of the method. This is especially notable in the case of gene classification, since the data sets is small but has enormous number of attributes. [DFS00] determines the value of K using leave-one-out cross validation on the training set. Each training sample, t, is treated as a test sample, the distance measure is used to check its correlation with every other training tuples. The k most correlated training tuples are selected to vote for t's class. Every training tuple's original class is compared with its assigned class to get the total number of erroneous assignment for the current value of k.

This process is repeated for different values of k to find the one that gives least error to be used as in the classification of testing tuples. In [DFS00], the method checked for $k = 1, 2, \ldots, 21$.

NN is simpler than KNN since it only requires finding one most similar training tuple while the KNN method needs to first determine the value of k. But KNN has several advantages over NN. First, in the case of mislabeled training tuples, it will have much greater effect on the classification result of NN since one mislabel will result in misclassifying all the test tuples that are most similar to it. Also, KNN is less prone to bias in the data and more tolerable to noise since it makes use of several training tuples to determine the class of a test tuple instead of one tuple.

3.5.2 Cluster-based Method: CAST

[BDBF$^+$00] developed a cluster-based classifier, CAST. It is motivated from the observation made by Alon et al[ABN$^+$99] on the hierarchical clustering on the colon data (see data set description): the topmost division divides the samples into two groups: one group contains mostly normal samples and the other mostly tumor samples.

The main idea is to group the training tuples into different clusters based on their similarity with the tuples already in the cluster and remove those that no longer has much similarity with the current cluster. CAST uses a threshold parameter, p, to control the granularity of the clusters. During the training phase, a training tuple is grouped into a cluster if it has *high similarity* with the tuples inside the cluster. A tuple is said to have high similarity with a group of tuples if the similarity measure between them is at least p (the authors used Pearson correlation between the gene expression values

as a measure of similarity, though any similarity measure can be used). After inclusion of the new tuple into a cluster, the similarity score of the tuples in the cluster is again evaluated. A tuple will be "kicked out" if its new similarity score with the updated cluster is below p.

CAST creates one cluster at a time. During the training phase, it alternates between adding tuples with satisfying similarity scores to the cluster and removes those with unsatisfying similarity scores. Eventually, one cluster is formed with tuples all having similarity score above p. It then goes on to create another cluster until every tuple has been included inside a cluster.

CAST selects the best p value through a measure of cluster *structure compatibility*. Intuitively, compatibility measure penalizes the cluster structures that have tuples of the same classes separated into different clusters and tuples of different classes being grouped into the same cluster. They define the *compatibility score* as the sum of the number of pairs of tuples that have different classes and assigned to different cluster and the number of pairs of tuples that have the same class and assigned to the same clusters.

In selecting the best p value, the algorithm iteratively considers different values for p using binary search. For each p, it uses CAST to cluster the training tuples and then apply compatibility measure to the clusters. The p value that gives the highest compatibility score is chosen to used in the classification phase.

Given a test sample s, and a group of training tuples T, CAST tries different values of p to cluster s and T to find the best cluster structure (note that the compatibility measure is only done on the training tuples). The class label of s is determined through majority vote of the training tuples in the cluster where s belongs to. If there is no majority class in the cluster, or the majority is too small to be confident, then s is assigned to the "Uncertain" class.

3.6 Max-Margin Classifiers

The training process of a classifier can be viewed as a process of finding an hyperplane that separates the training tuples into different groups according to their classes. The *margin* of the hyperplane is defined as the distance from the hyperplane to the sets of points that are closest to it. A hyperplane with small margin is not as confident as the one with large margin. Given a slightly different training data, the hyperplane would change, thus changing the classification of points that lie close to the hyperplane. The definition of margin suggests that if a classifier is able to separate the points in a way that maximizes the margin, then it will be less subjective to overfitting and have better classification performance.

For the expression data used in cancer classification, they can be viewed as very sparse points in very high dimensional space. For such kind of data, it is very easy to find several hyperplanes that linearly separate the training tuples, yet easily subject to overfitting. Therefore, max-margin classifiers [FS98, SBS00] are good choices for dealing with this kind of data. Here we introduce two max-margin classification algorithms, Support Vector Machine and Boosting that has been applied on cancer classification.

3.6.1 Support Vector Machine

Support Vector Machine (SVM) was originally introduced by Vapnik and co-workers and used in many data mining applications[BGV92, Vap98, Bur98].

Given a set of binary class training tuples, each tuple x_i can be considered as a point in an m-dimensional input space, where m is the number of attributes. Associated with each point is a class label $c_i \in \{1, -1\}$. Two classes are linearly separable if there exists a hyperplane, $(\boldsymbol{w}, b)$ where $\boldsymbol{w}$ is a vector and b is a scalar, such that for point x_i,

$$\begin{cases} \boldsymbol{w}^T \boldsymbol{x}_i + b >= 0 & \text{for } c_i = 1 \\ \boldsymbol{w}^T \boldsymbol{x}_i + b <= 0 & \text{for } c_i = -1 \end{cases}$$

For small training data size and large attribute space, there exist many such hyperplanes. The SVM learning algorithm constructs a hyperplane with *maximum margin* that separates the positive tuples from the negative tuples. The points that lie closest to this max-margin hyperplane are called the *support vectors*. The hyperplane can be defined using these points alone and the classifier only makes use of these support vectors to classify tset tuples.

In the case of data that are not linearly separable in the input space, one solution is to map the *input space* into a higher dimensional *feature space*. Let ϕ: $I \subseteq \Re_m \rightarrow F \subseteq \Re_M$ be a mapping from the input space $I \subseteq \Re_m$ to the feature space F. Let $\langle x, y \rangle$ denotes the dot product of vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. Then the max-margin hyperplane is defined as the plane that have the value

$$\gamma = min_{i=1}^{n} c_i \langle \boldsymbol{w}, \phi(t_i) \rangle - b$$

maximized.

Here, the value $(\langle \boldsymbol{w}, \phi(t_i) \rangle - b)$ is the distance between the point t_i and the hyperplane in the feature space.

The sign of this value gives the side of the hyperplane that this point resides in. When this signed distance is multiplied with c_i, the class label of t_i, the result is positive if the tuple is correctly classified and negative if the tuple is incorrectly classified.

Recently, SVM has been widely used on gene expression data [Bea00, FKRWea01, MTS+99]. [CST00] showed that for a set of n training points x_i, with corresponding class label c_i, the hyperplane that maximized the margin has the vector $\boldsymbol{w}$ given by

$$\boldsymbol{w} = \sum_{i=1}^{n} \alpha_i c^i(\boldsymbol{x}_i)$$

where α_is are the positive numbers that maximize

$$\sum_{i=1}^{n} \alpha_i - \sum_{ij=1}^{n} \alpha_i \alpha_j c_i c_j \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$$

subject to the condition that

$$\sum_{i=1}^{n} \alpha_i c_i = 0, \alpha_i > 0$$

It turns out that only for the support vectors, their α_i values are non-zero. Therefore, only the support vectors are required to define the max-margin hyperplane. This implies that only the training tuples with non-zero α_i values have to be stored in memory for the classification process.

[FCD+01, BDBF+00, RTRea01] applied SVM technique to the cancer classification problem. Given a set of gene expression training tuples of size n, each tuple is a pair in the form of (t_i, c_i), where t_is are the vectors of the expression values of m genes, and c_is are the corresponding class labels and $c_i \in \{1, -1\}$. Let the max-margin hyperplane be denoted by the vector $\boldsymbol{w}_0$ and b_0. Given a test tuple s with class label c, the classification result produced by the SVM on s is given by its relationship with respect to the hyperplane generated, The result is given by:

$$Class(\boldsymbol{s}) = sign(c(\langle \boldsymbol{w}_0, \phi(\boldsymbol{s}) \rangle - b_0)$$

If the returned sign is positive, it means s is correctly classified, otherwise s is incorrectly classified.

Sometimes when the training samples have mislabeled samples, the tuples may not be linearly separable in the feature space. In other times, for the sake of error tolerance and overfitting avoidance, perfect linear separation may not be desirable. For these cases, it is better to allow some training samples to fall to the wrong side of the hyperplane [Bea00, FCD+01]. *Soft margin* and *margin-distribution* [JTC99] SVM are developed for this purpose. Through some parameter tuning, which controls the overall training error or a specific false positive or false negative error, desired training error can be achieved.

The ability of SVM for producing hyperplane with maximized margin and for tuning the amount of training errors allowed has made SVM especially suitable for the gene expression data classification.

3.6.2 Aggregated Classifiers: Boosting

Aggregated classifiers are built from the aggregation of multiple versions of the class predictors using majority voting [Bre96, Bre98]. It is useful for improving the accuracy of the classifiers that are unstable to small changes of the learning set, for reducing the overfitting problem due to small training data set and for improving the accuracy of those "weak learner" classifiers.

The idea is to generate multiple versions of classifiers over the training data in iterations. In each iteration, the training data are sampled with replacement by putting more emphasis on the training tuples that were wrongly classified in the previous iteration. The final aggregated classifier is a weighted voting of the classifiers built during each iteration. The weight is dependent on the accuracy of the classifier built from the data it was being trained.

Here we describe AdaBoost algorithm proposed by [FS97] for *binary-class* problems. Given a training set $T = \{(t_1, c_1), (t_2, c_2), \ldots, (t_n, c_n)\}$, and a classification algorithm L. Since there are only two class labels, we could map one class label to -1 and the other to 1 for easier classification using boosting.

The initial distribution of weights of each tuple is the same, given by $W_1(t_i) = 1/n$, where $i = 1, \ldots, n$. In iteration i, call to L with weights distribution W_i will return a classifier L_i. The error, ϵ_i,

of L_i is given by $\sum_{j=1}^{n} W_i(t_j)\{c_i \neq L_i(t_j)\}$. Here, $\{c_i \neq L_i(t_j)\}$ checks if the classifier classifies tuple t_j correctly, it returns 1 if the statement is true (false classification) and 0 otherwise.

The weight of L_i is then $w_i = \frac{1}{2} log \frac{1-\epsilon_i}{\epsilon_i}$. The new weight distribution of T is then given by

$$W_{i+1}(t_i) \propto W_i(t_i) e^{-w_i c_i L_i(t_i)}$$

such that $\sum_i^n W_{i+1}(t_i) = 1$. Suppose there are x number of classifiers trained. Then given a test sample s, the class of s is given by

$$Class(s) = sign(\sum_{i=1}^{x} w_i L_i(s))$$

.

The margin m_i of a tuple t_i with class label c_i can be defined as:

$$m_i = c_i \sum_{j=1}^{K} w_j Class_i(t_i)$$

Positive m_i means t_i is correctly classified and otherwise t_i is incorrectly classified. The magnitude of m_i is proportional to the confidence of the classification. The smaller the magnitude, the less confident.

During the iterations of the classifier construction, if a tuple is misclassified in the current iteration, it will be assigned a greater weight in the next round. Thus, the goal of the algorithm is to "concentrate" on the hard samples that always get misclassified to achieve optimal classification. This actually corresponds to the effort of trying to enlarge the margin of the training tuples.

Theoretically, boosting can be thought of as a max-margin classification technique similar to SVM [BDBF$^+$00]. [SFBL98, MBB98] also showed that the generalization error of boosting depends on the distribution of margins of the training tuples.

[BDBF$^+$00, DFS00] applied boosting for cancer classification. [BDBF$^+$00] used a very naive decision function to act as a weak classifier. The decision function f is given as:

$$f(g:i,t,d) = \begin{cases} d & g[i] > t \\ -d & g[i] < t \end{cases}$$

f takes in three parameters, $g:i$, t, and d, where g denotes the vector of gene expression values, thus $g:i$ denotes the value of ith gene in the vector, t denotes the split point for gene i, and d is the class of g, $d \in \{-1, 1\}$. This decision function is applied to every gene and checks through every possible split point to find the one that gives the best class separation. The boosting process is thus iterations of split point evaluations for every gene in the tuples. The test samples are then classified according to the weighted voting of the classes given by the split point decision functions in each iteration.

[DFS00] applied boosting technique using the CART decision tree as the classifier. Both method achieved satisfactory results even though neither classification algorithm is well suited for the cancer data set. The credit should go to the max-margin idea in the boosting technique that concentrates on tuples that can easily get misclassified and try to find the separation that maximizes the margin.

3.7 Evaluation

We believe that for cancer classification, besides computation time and classification accuracy, biological relevance such as information about gene interaction, marker genes and etc(we will elaborate it in the next section of related issues) should also be one of the evaluation criteria.

This is due to the fact that biologists not only expect cancer classification gives correction classification but also want to generate more information about the genes for cancer-related studies such as drug discover and have a better understanding of tumor development.

Below, we give a rough evaluation of the cancer classifiers based on their classification performance and biological relevance.

3.8 Weighted Voting: GS Method

The GS method is simple and works well in some data such as the Leukemia data set. But its simplicity also results in some limitations. First, it is only applicable for binary class problem. Typical cancer classification involves identifying of more than two classes of cancer. In this case, GS algorithm will not be effective. Second, it chooses an equal number of genes that have high correlation with the two classes. This means that it is only effective in classifying data sets that are unbiased. Biased in this sense means that there are unequal number of genes favoring the two classes with the same correlation strength. If the majority genes of a data set are highly correlated with one class, then this method will not produce satisfactory classification result since it always choose an equal number of genes in both classes to vote for the majority.

3.9 Similarity-based Classifiers: KNN and CAST

The similarity methods(KNN and CAST) base their classification on the similarity of the expression values of each gene, this makes them less prone to noise and bias in the data. Both methods tries to find for a test tuple s, a set of training tuples that it has high similarity with and use majority voting of these training tuples to determine the class of s. The disadvantage of these methods is due to their non-scalability. Every testing tuple needs to be checked against every training tuple in order to find the most similar ones. These method works fine when the data size is small, but they suffer when the data size is large.

Comparatively, KNN is less computationally expensive than CAST since the similarity score is only evaluated once for every test and training pair. Whereas in CAST, it has to repeatedly evaluate the similarity score of each tuple in the clusters against that of the rest during the formation of each cluster. This is repeated many times in order to find the best granularity control parameter. We feel that it might not be practical to apply this method to cancer classification since it requires too much computational time. As the availability of gene expression data increases, this method tends to loose its advantage over other more scalable methods.

3.10 Max-Margin Classifiers: SVM and Boosting

Support Vector Machine has been well exploited by researchers in various fields from text categorization, pattern recognition to protein function prediction. Recently, SVM has been used in cancer classification. ability to deal with high dimensional data, avoidance of overfitting, and robustness with noise and sparseness of solution all make it well suited for application to cancer classification. Another benefit offered by SVM is scalability [Tre01]. The number of support vectors selected by the learning algorithm is usually small, even with large training set [BGL$^+$99, BDBF$^+$00, JTC99]. This is essential since the amount of available gene expression data will soon increase dramatically. One drawback that limits its original application is only for binary class problems. Some solutions were suggested to overcome this limitation, such as redefining the multiclass problem into binary, or iteratively performing binary classification until all classes has been separated, but none was very effective. Recently, extensions to multiclass SVM were being studied[CS00, LLW01, HL01], its effectiveness and performance is still an on-going research problem.

For the current available data characteristics, boosting achieves comparable classification performance with respect to other methods. Since boosting is usually applied to weak learners to improve classification accuracy through repeated classification of the weighted training tuples, it is quite time consuming. In addition, existing classification methods do produce comparable classification accuracy, thus we feel that the boosting method does not have any superiority over other methods.

3.11 Bayesian Network and Neural Networks

Bayesian network method is simple and can be applied to multiclass classification problem. But there are two issues that restricts its performance in cancer classification. First issue is its assumption of orthorgonality among genes. It is known that in most cases, the expression values of genes are correlated. Also, gene interaction is of important interest for the biologists. Assuming independence not only ignores important classification information embedded in the data set which might result in inaccurate classification, but it is also incapable of revealing any biological information in the data since the whole process is a black box. Another issue is that it assumes a Gaussian distribution of the data. And in the cases of data set that does not follow such a distribution, it proposed to do some transformation such as taking the logarithmic or square root of the gene expression values. But still, the assumption that the classes must follow a certain distribution is too restrictive, especially in the case of gene expression data which are hard to say what kind of distribution the data follow. We feel that though this method might be useful in classifying certain types of expression data, but its performance is limited due to its assumptions.

Neural network method has comparable result with the other methods. But similar to the naive bayes method, it does the classification in a black box manner. The user are not given any information on how the genes are correlated, which set of genes is more effective for classification, etc..

3.12 Decision Trees

Decision trees have been attractive in data mining environment for several reasons. First, their results are interpretable. Second, they do not require any parameter input. Third, the construction process is relatively fast. We believe that these advantages are also applicable to cancer classification using gene expression data. An additional advantage is the scalability of decision trees. There are several scalable decision tree algorithms [GGRL99, SAM96] available, as the date size increases, these algorithms might be useful for providing satisfying performance over large gene expression data.

3.12.1 Classification Accuracy

From the experiments conducted in[BDBF$^+$00], it found that SVM has the highest accuracy on the leukemia and ovarian data set, while CAST performs better for the Colon data set and Boosting using simple decision stump outperforms NN in all three data sets.

[KSHR00] compared NB method with GS method and found that NB outperforms GS method for the leukemia and ovarian data sets while GS outperformed NB in the colon data set. The NB classifier achieved 100% accuracy on the leukemia data set and 84%accuracy in the ovarian and colon data set.

[DFS00] observed that the classification accuracy for the NCI data set is much lower than that of leukemia and colon data set. They contributed it to the small data size (60) and wide range of classes (10). We believe that for such a small data size and large number of classes, overfitting is the main problem causing the degrading of performance.

Some of these approaches perform classification after gene selection. Experiments show that the quality and quantity of genes selected has quite big effect on the classification results, as expected.

Overall, the proposed methods all showed good performance on some data sets and there is no one that is superior than the rest on all data sets. Since the amount of gene expression data available is small, the classification accuracy of various algorithms cannot be compared extensively. It would be necessary to perform an extensive comparison of all the proposed methods on the available data sets in order to really judge which method gives the best classification accuracy. This could be one direction for future work.

3.12.2 Biological Relevance

One important difference between cancer classification and other previous classification applications is the amount of useful information revealed during the classification process. Obviously, accuracy is very important in cancer classification since it helps in accurate diagnosis of cancer patients. On the other hand, we still have limited knowledge about the cancer disease, biologists wish to gain more understanding of gene interaction and other biological information related to cancer development. Thus, cancer classification is not merely about classification accuracy, but also about identifying useful information for studying gene interactions. A cancer classifier that does not explore the biological information within the gene expression has only achieved part of the classification goal. Here, we give

a categorization of the methods described in this section to reflect their biological relevance in using and revealing the gene interaction information. The methods can be grouped roughly into three categories.

The first category corresponds to methods that totally ignore the context of the data, the representatives in this category are FLDA (or any statistical based methods), the naive bayesian method and the neural network method. These methods only looks at the data as a set of distributions and make classification decision based on the distribution of data values regardless of the context meaning of the data.

The second category corresponds to methods that makes use of the correlation among genes in the data but do so in a "black box" fashion. NN, KNN, perceptrons, artificial neural networks, CAST and SVM belongs to this category. The classifiers in this category makes use of the correlation between the expression values of genes. But they do not give any insight into the structure of the data.

The third category corresponds to the methods that are capable of exploring and revealing correlations between the genes. The representative in this category is the classification tree based method-recursive partitioning. Classification trees reveal the relationships among the genes and do so through a stepwise predictor selection. This process of selection and splitting provides the biologists some insight into the structure of the data and the correlation information among the genes. The gene expression data have high dimensionality and small training data size, there might be times when decision tree method fail to give good classification accuracy due to noise and overfitting. Applying boosting with classification tree might be a good solution to improve accuracy in this type of situations though the information about gene correlations are less visible.

From this categorization, it can be seen that most cancer classifiers are still lacking in the biological relevance aspect. Further development in this direction is necessary.

3.13 A Unified View

Table 1 is a summarization of the various methods introduced in this section.

	Multi-class	Strategy	Biologically meaningful	Scalability
SVM	No	Max-Margin	No	Good
Boosting(Decision-tree)	Yes	Max-Margin	Yes	Classifier Dependent
Decision tree	Yes	Entropy Function	Yes	Good
KNN($K \leq 1$)	Yes	Similarity	No	Not Scalable
CAST	Yes	Similarity	No	Not Scalable
GS	No	Weighted Voting	Yes (gene selection)	Fair
FLDA	Yes	Discriminant Analysis	No	Fair
Neural Network	Yes	Perceptrons	No	Fair
Naive Bayes	Yes	Distribution modelling	No	Fair

Table 1: Summarization of cancer classifiers

It can be seen that some of the classification methods used for cancer classification do not provide much biologically relevant information to the user. In order to fully achieve the goals of cancer classification which are accuracy and bio-relevancy, it is necessary to either develop new classification algorithms aimed at providing good cancer classification accuracy while at the same time revealing more information for gene interaction, or modify the existing algorithms to take care of the bio-related aspects.

Additionally, since there are only a limited number of available data sets, and the size of these data are very small, thorough experiments cannot be conducted to really distinguish the better ones from the others. Right now we can only theoretically compare the algorithms on their computational time and effectiveness on cancer classification. More thorough comparisons for the computation time, classification accuracy and biological relevance needs to be done on the existing methods to provide a better picture of the current status. This would help us at getting better insight into the development of more effective and efficient algorithms aimed particularly at classifying cancerous data using gene expression values.

4 Gene Selection

Feature selection is an useful preprocessing technique in data mining, it's usually used to reduce the dimensions of the data and improve classification accuracy [KS96, SS88]. Currently, some feature selection methods on genomic data have been proposed [XJK01, CLT01, GWB$^+$00]. For the problem of cancer classification, we believe that a fair amount of attention should be paid on gene selection and make it an integral preprocessing step. This is well justified due to the following reasons.

First, gene expression data set has very unique characteristics which are very different from all the previous data used for classification. Most publicly available gene expression data has the following properties:

- high dimensionality: up to tens of thousands of genes,
- very small data set size: less than 100, and
- most genes are not related to cancer classification.

With such a huge attribute space, it is almost certain that all classifiers built upon it would prone to overfitting. The small data size makes it even worse. Since most genes are known to be irrelevant for class distinction, their inclusion would not only introduce noise and confuse the classifiers, but also increase the computation time. Gene selection prior to classification would help in alleviating these problems. With the "noise" from the irrelevant genes removed, the biological information hidden within will be less obstructed. During the classification process, classifiers such as decision tree will be able to provide a more precise view of genes interaction. Post-classification analysis can be done to find out some biologically relevant issues such as the sets of genes that are most related to certain kind of cancer or if there are other biological unknowns related to accurate cancer prediction. This would assist in drug discovery and early tumor discovery.

Second, experiments have shown that gene selection prior to classification improves the classification accuracy of most classifiers [DFS00, GST+99, GWB+00, CLT01]. It not only improves the performances of the classifiers that are weak in handling large number of attributes, but improves the performances of those that are capable of handling large attribute space, such as SVM[GWB+00]. Though there is no direct measure on the saving of running time, but the number of genes decreased from thousands to tens or below 10 after performing gene selection which would make a big difference on the running time.

Due to the above reasons, it should be clear that in order to achieve good classification performance, and obtain more useful insight about the biological related issues in cancer classification, gene selection should be well explored to both reduce the noise and avoid overfitting.

In this section, we take a look at the proposed gene selection methods in detail. Though researchers have used different feature selection methods for gene selection, but they can be broadly categorized into 2 groups: *individual gene ranking* approach, and *gene subset ranking* approach. Below, we first describe the main idea of each category and then followed by some gene selection methods in each. We conclude the section with some discussion.

4.1 Category I: Individual Gene Ranking

Feature ranking approach is the most commonly used for feature selection. In this approach, each feature/attribute is measured for correlation with the class according to some measuring criteria. The features/attributes are ranked and the top ones or those that satisfy a certain criterion are selected. The main characteristic of feature ranking is that it is based on *individual* feature correlation with respect to class separation.

In gene selection, methods in this category select genes with high *individual correlation* with the classes. No correlation among the genes are exploited. Genes selected in this case may have high correlation with the class as an individual, but act together, might not give the best classification performance. Also since each gene is considered separately, some genes may contain the same correlation information thus introduce redundancy. Finally, genes that are complement to each other in determining the class labels may not be selected if they don't exhibit high individual correlation.

4.1.1 Using Correlation Metric-GS Gene Selection Method

The most common and simple approach to feature selection is using correlation between the attribute values and the class labels. In this approach, the correlation is measured using a distance function, such Euclidean distance, Pearson correlation and etc.

The GS method [GST+99] proposed a correlation metric that measures the relative class separation produced by the expression values of a gene. It favors genes that have big between-class mean expression value and small within-class variation of expression value. For gene g, let μ_1 and μ_2 denote the mean expression values of g in the two classes, and let σ_1 and σ_2 denote the standard deviation of the expression values of g in class 1 and class 2. The gene selection metric is given as $P(g) = \frac{\mu_1 - \mu_2}{\sigma_1 - \sigma_2}$. The

P-value of each gene is measured and genes are grouped into positive value and negative value groups and ranked according to their absolute values. The top $k/2$ genes from the two groups are selected.

This method is simple and easy to implement, but have several drawbacks. First, in the case of classifying normal and cancerous data, the cancerous class may comprises cancers of different types. Therefore, the genes that are cancer-related might not have small within-class variations since the expression values for different tumors vary. This would result in mis-selection of the relevant genes. Second, it only checks individual gene correlation, thus unable to find an optimal set of genes in certain cases. Lastly, this method is not resistant to gene contamination (detailed discussion in next section). Genes that are not cancer-related but more composition related sometimes are selected as top ranking genes [GWB$^+$00].

4.1.2 Using the Weights of a Linear Discriminant Function

Some classifiers are trained as a linear discriminant function. A linear discriminant function is of the form of $D(x) = \boldsymbol{w} \cdot x + b$ where $\boldsymbol{w}$ is the vector of weights for each gene, and x is a matrix of input data and b is the bias. In this approach, the full set of genes are used to train a classifier in the form of a linear discriminant function. Genes having the top K weights are selected, where K is the desired number of genes.

The rationale is that for a trained classifiers, the weights of the features are proportional to their importance in determination of the class labels. That is, the higher the weight, the better in its distinction power of the class label. Therefore, given a trained classifier, we are able to obtain a set of K high ranking genes by selecting the genes with the top K weights. For the classifiers that in the form of a linear discriminant function. The top K genes can be selected from the weight vector $\boldsymbol{w}$.

4.1.3 Likelihood Gene Selection

Keller et. al. [KSHR00] used likelihood measurement for gene selection. Given a training data set, it tried to select genes whose expression values are a good indication of the class separation.

The genes are selected through a *log likelihood score* function, LIK. The LIK scores for each class are computed for every gene. Let's first consider the binary data case. Let C_1^g denote the distribution of class 1 for gene g, and C_2^g denotes the distribution of class 2 for gene g. Then, the relative log likelihood scores, can be defined as follows:

$$L1K_{1\to 2} = logp(C_1^g|X_1) - logp(C_2^g|X_1)$$

$$L1K_{2\to 1} = logp(C_2^g|X_2) - logp(C_1^g|X_2)$$

Where X_1 and X_2 refers to the training tuples in class 1 and class 2 respectively. A gene with ideal class discrimination power would have high values for both LIK scores, indicating that expression values of this gene in tuples belonging to class 1 will vote for class 1, and that in tuples belonging to class 2 will vote for class 2. This implies that if the test tuples are of the same distribution as the training sample, then the expression values of this gene will be a good indication of the class labels. The

higher the LIK scores are above 0, the better the class discrimination power the gene has. [KSHR00] pointed out that in practice, it is difficult to find genes for which both LIK scores are much greater than 0. They resort to select two sets of genes, $GENE_{1\to 2}$ and $GENE_{2\to 1}$, each maximizing one of the LIK scores, while requiring the other score to be above 0:

$$GENE_{1\to 2} : LIK_{1\to 2} \gg 0 \quad \text{and } LIK_{2\to 1} > 0$$
$$GENE_{2\to 1} : LIK_{2\to 1} \gg 0 \quad \text{and } LIK_{1\to 2} > 0$$

After computing the LIK scores for every gene, genes in each set is ranked according to their LIK scored in that set. The top $x/2$ genes are selected from each set, where x is the number of genes to be selected.

In generalizing to multiclass data, it requires computing $c(c-1)$ number of LIK scores for each gene, where c is the number of classes

$$LIK_{j\to k} = logp(C_j^g|X_j) - logp(C_k^g|X_j)$$

where X_js are the training tuples in class j, $1 \le j \le c$, and $j \ne k$.

Intuitively, it tried to find for every gene, its ability to discriminate each class from all the rest of the classes. As in binary class case, $c(c-1)$ sets of genes are grouped according to the LIK score they maximize and requiring the rest scores above 0:

$$GENE_{j\to k} : LIK_{j\to k} \gg 0 \text{and } LIK_{j'\to k'} > 0$$

where $j' \ne k'$, $1 \le j'$ and $k' \le c$. Equal number of genes are selected from each set to form the total set of genes selected for classification.

[KSHR00] compared the genes selected using NB and GS gene selection method. It found that NB's gene selection method gives better classification accuracy. Also, the genes selected by the NB method has more variety than that of GS. They concluded that this variety of genes contributed to the better classification accuracy.

4.2 Category II: Gene Subset Ranking

The approach used by methods in this category is to find a *group of genes* that serve *together* to achieve the best classification result. The idea is to remove genes one by one, monitor the effect of removal of this gene on the *expected value of error*. The expected value of error is the error rate computed on an infinite number of samples. Given a training set, the expected value of error can be approximated by a cost function J computed on the training samples. The effect of computing the change in the expected value of error caused by removing a gene i is equivalent to bringing the weight of i to 0 in the decision function. Given a classifier with weights for each training tuples, the change in cost function from the removal of gene i is approximated by

$$DJ(i) = (1/2)\frac{\partial^2 J}{\partial w_i^2}(Dw_i)^2$$

where Dw_i is the change in the weight after removing gene i.

4.2.1 Recursive Feature Elimination (REF)

In [GWB+00], it proposed the use of *recursive feature elimination* approach to do gene selection. It iterates through following steps:

- Train the classifier (optimize the weights w_i with respect to cost function J)
- Compute the ranking criterion for all features ($DJ(i)$ or $(w_i)^2$)
- Remove the feature with the smallest ranking

From the elimination process, we can see that it tries to retain the set of features that has the highest classification power. The features that are eliminated last may not be necessarily the ones that are individually most relevant.

SVM has recently gained popularity in both classification and feature selection [WMC+00]. [GWB+00] used SVM to be the classifier for the cost function computation on subset ranking based gene selection. The SVM based REF gene selection method works as follows: apply the SVM classification algorithm on the training data; compute the change in cost function for the removal of each gene i; find the gene that has the minimum cost function change after its removal; remove that gene from the training data; repeat until the training data is empty.

It can be seen that this method is a backward feature ranking method. The group of genes that are removed last together gives the best classification result. But individually, they might not be the ones that has the best correlation. This is true by looking at the way the features are eliminated. The elimination function is the change in the expected error rate of the classifier. In this sense, it always remove the gene whose removal has the minimum change on the classification result.

4.3 Discussion

In this section, we compare the two categories of gene selection methods in several areas.

4.3.1 Redundancy and Complementary Issue

[GWB+00] found that the REF gene subset ranking (GSR) approach works better for cancer classification than the individual gene ranking approach(IGR). The former is capable of finding genes that are complementary which produce best classification result, whereas the latter only finds genes that have high individual correlation. Individual correlation does help in class distinction, but only to a certain extent. There are additional information relevant to tumor type differentiation that can only be revealed through the interactions among the genes. Some genes may not have individual high correlation but they complement each other in class distinction. Since the individual gene ranking approach does not have any mechanism in discovering the complementary roles among the genes, it can be considered as a handicapped approach. Also, since in IGR approach, each gene is assessed for its correlation individually and selected based on the correlation score, it is inevitable that in some cases, it would

select a set of genes with similar correlations(redundancy) yet miss out some other important genes. In the case of tumors that have different expressibility, i.e., some tumors are less distinctively expressed by genes and some tumors that more easily expressed, the IGR approach might select lots of genes that can distinguish one class but miss out those that are useful for distinguishing other not so highly distinguishable classes.

4.3.2 Computation Time

GSR approach seemed to provide a more optimal set of genes for cancer classification and other related biological studies. But it takes too long to compute. With so many genes in the data set, it has to iteratively perform subset ranking and eliminate genes one by one. [GWB$^+$00] suggested to trade accuracy for speed by initially removing chunks of genes each time, and resolve to remove one at a time towards the end. The reason is that though better results are obtained when removing genes one at a time than removing chunks at a time, the difference is only significant for small set of genes. Therefore, the time saving sub-optimal solution is to removes chunks of genes at one time and when the gene size reaches a few hundred then start to eliminate one by one.

4.3.3 Class Distinction Power vs Direct Relevance to Tumors

Another issue of concern is how relevant to cancer are the genes being picked by the two approaches. Some genes might give a good class distinction, but they might not be directly cancer related. Due to the fact that cancerous tissues and normal tissues usually have very different cell composition [GWB$^+$00, BDBF$^+$00], it is possible that those genes observed to have high distinction power are those that related with different cell composition instead of directly linked with different types of tumors.

[GWB$^+$00] analyzed the genes selected by the methods in both approaches and found that SVM REF gene selection method was able to pick up genes that are directly related with cancer, whereas other methods tends to pick up genes that are non-cancer related but due to the different cell compositions between the tissues. We believe that this is due to the fact that the SVM REF method uses subset ranking method instead of feature ranking method. This method tends to pick up a group of genes that when put together best determines the class type which means it best captures the gene interaction information.

It has also been observed that for the same gene selection mechanism, the accuracies of different classifiers do not differ much, whereas different gene selection methods have greater effect on the classification accuracy. This again proved how an important role gene selection plays in cancer classification.

In conclusion, we feel that a good gene selection method for cancer classification should observe the following properties:

- incorporation of the gene interaction information,
- selection criteria based on group performance instead of individual correlation,
- Inclusion of complementary genes,

- Picking of cancer related genes instead of cell composition related genes, and
- Reasonably efficient.

5 Related Issues

In this section, we discuss some important issues in cancer classification using gene expression data. They include the issue of statistical significance vs biological significance of a cancer classifier, detailed breakdown of classification results, the issue of sample contamination, and the effect of selection of marker genes on cancer classification.

5.1 Statistical vs. Biological Significance

In the past, classifiers are usually evaluated on the classification accuracy and efficiency. Not much attention was paid on the ability of the classifiers in discovering interactions between attributes. This is because the attributes in the earlier data used usually do not exhibit complex behavior or underlying meaning. Unfortunately, this is not the case for cancer classification.

As an area of bioinformatics study, the major goal of expression data analysis is to provide the biologists biologically meaningful information about the genes and related things. Through these information, biologists are able to discover unknowns and reaffirm previously knowledge. For cancer classification, information about gene interaction is of great biological relevance. It provides the biologists a clearer understanding of the roles a certain set of genes play in cancer development and related issues. This will greatly assists the discovering and early treatment of disease and drug discovery. It is a common consensus that biological relevance for a cancer classifier is as important as the classification accuracy.

One important issue is to find marker genes. *Marker genes* are genes whose expression values are biologically useful for determining the class of the samples. In other words, marker genes are genes that characterize the tumor classes.

The identification of marker genes are important due to the following reasons:

- Enable better classification performance
- Allow biologists further study the interaction of relevant genes in achieving a certain biological performance
- Study the functional and sequential behavior of known marker genes in order to facilitate the functionality discovery of other genes.
- Allow further study of relation of expression values of different genes with respect to the tumor class, is similar expression pattern always results in cancer or the combination of suppression of certain genes and expression of certain genes are a better indication of tumor, etc.

In section 3, we described several representative gene classification methods. No classifier is superior over all the others in the aspect of classification accuracy, but there is big difference in the biological meaningful aspect. It is obvious that statistical based classifiers do provide good accuracy. It might be a very good tool for classifying other types of data. But it is not a sufficiently good classifier in the case of cancer classification. Their accuracy is solely based on the statistical significance of the values of the attributes. For these methods, the context meaning of the values do not play any role thus unable to reveal any interactions among the genes nor the relation of the genes with the class labels were explored during the classification.

For the neural network approach, the propose naive bayesian method may work well in certain cases, but it uses the independence assumption of the genes. By this assumption, the method assume there is no interaction among the genes, and thus of course, it will not reflect any biological interaction of genes. SVM methods is a favorable gene classification method, but it suffers from its inability to provide biological meaning result to the biologist. The decision tree method provides step-wise classification process which gives the user some insight into how genes interact in determining the class labels.

5.2 Classification Errors

For the previously studied classification problems, classification error is just the number of tuples being misclassified. This is not adequate in the case of cancer classification.

In classifying normal vs cancerous data, the errors can be grouped into *misclassification* rate and non-classification rate. The misclassification rate is the ratio of tuples that is being wrongly classified. The non-classification rate is the ratio of tuples that could not be classified due to lack of confidence of the classifier in pin pointing a class label for that particular tuple. This rate is very important in cancer classification, since wrong judgement can be fatal. The misclassification error can be divided into *false positives* and *false negatives* [BDBF$^+$00]. False positives refers to negative samples that are classifiered as positive. False negatives refers to positive samples that are classifiered as negative. In other classification problems, false positives and false negatives makes not much difference, but it makes a big difference in cancer diagnosis applications. False positives are tolerable since further clinical experiments will be done to confirm, but false positives are detrimental since a cancer patient might be misclassified as normal. Therefore, in order to perform a fair and thorough comparison on the classification accuracy of different methods, it is necessary to compare both the non-classification rate, the false positives and the false negatives.

An ROC curve can be used to evaluate the "power" of a classification method with different asymmetric weights for the misclassification errors. A confidence parameter β can be used to control the confidence of the classification. The ROC curve can be used to plot the tradeoff between two types of errors as the confidence parameter varies.

By breaking the performance result into detailed pieces, it is easier to see which classifier has better performance in providing biologically meaningful information and accurate classification.

5.3 Sample Contamination

[BDBF+00] raised issue of *gene contamination*. Usually, cancerous and normal tissues have different compositions of cell types. Tumor and normal tissues may differ in cell composition (colon data set) [DFS00] since tumors are generally rich in epithelial cells, while normal tissues contain variety of cells, including large fraction of muscle cells These different cell composition may lead to genes that have different expression value in normal and cancerous tissues though they are not the consequence of the cancerous/normal difference. The usual mistake in gene selection or classification is splitting the samples based on genes that distinguishes the cell composition, instead of the cancer-related ones. This might lead to classification success, but does not provide any biological information about genes relating to cancer development.

In [BDBF+00], the authors checked the presence and absence of such genes in the Colon data set on the classification accuracy, found that in general, their absence does not affect accuracy. But the problem is some gene selection algorithms tend to pick up genes that have correlation due to difference in cell composition among the samples instead of those cancer-related ones. Further research is needed in selecting the relevant genes for cancer classification.

5.4 Availability and Standardization of Data: Public Repository of Gene Expression Data

DNA microarray and chip technology is creating considerable amounts of valuable data already, but these data are either not available publicly, or are scattered across the Internet.

Till present, though the number of genes in the gene expression data are huge, but the number of number of available data samples are very small. This has hindered the development of effective algorithms for cancer classification. With the limited amount of data sets available and small data size of these data sets, the scalability of the algorithms cannot be tested. Also, comparison of effectiveness of different algorithms cannot be done since they can be only compared on a very few data sets which may not be the representatives of the kind of expression data that will be available in the future.

Currently, the gene expression data came from different laboratories, this means the data across those laboratories may not be standardized. As more laboratories acquire this technology, the amounts of large-scale gene expression data and profiles will grow rapidly, leading to a gene expression data explosion. This might introduce the following two issues: First, data from different labs needs to be combined to create a larger data set. Non-standardization of data will introduce noise and error into the classification accuracy. Second, data set from different labs may contain different sets of genes. This means either the data will contain missing values or methods need to be developed to efficiently combine the data from different labs by selecting only the common gene expression data.

6 Conclusion

Systematic and unbiased approach to cancer classification is of great importance to cancer treatment and drug discovery. Previous cancer classification methods are all clinical based and were limited in their diagnostic ability. It has been known that gene expressions contains the keys to the fundamental problems of cancer diagnosis, cancer treatment and drug discovery. The recent advent of microarray technology has made the production of large amount of gene expression data possible. This has motivated the researchers in proposing different cancer classification algorithms using gene expression data.

In this paper, we provided a comprehensive survey on the existing cancer classification methods and evaluated their performance in three aspects: computation time, classification accuracy and biological relevance. Gene selection as an important preprocessing step was also presented in detail and evaluated for their relevance in cancer classification. Related issues such as statistical significance vs biological significance, asymmetric classification errors, gene contamination and marker genes were also introduced.

Through this survey, we conclude that cancer classification using gene expression data has a promising future in providing a more systematical and unbiased approach in differentiating different tumor types. However, there is still a great amount of work that needs to be done in order to achieve the goal of cancer classification.

References

[ABN+99] U. Alon, N. Barkai, D. Notterman, K. Gish, and et.al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. In *Proc. Nat. Aca. Sci. USA*, volume 96, pages 6745–6750, 1999.

[Aea00] A. Alizadeh and et. al. Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. *Nature*, 403:503–511, 2000.

[Azu00] A. Azuaje. Interpretation of genome expression patterns: computational challenges and opportunities. *IEEE Engineering in Medicine and Biology*, 2000.

[BDBF+00] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and Z. Yakhini. Tissue classication with gene expression profiles. In *Proc. of the Fourth Annual Int. Conf. on Computational Molecular Biology*, 2000.

[Bea00] M. Brown and et al. Knowledge based analysis of micorarray gene expression data by using support vector machines. In *Proc. of the National Academy of Sciences*, volume 97, pages 262–267, Jan 2000.

[Ber00] A. Berns. Cancer: Gene expression in diagnosis. *Nature*, pages 491–492, Feb 2000.

[BFOS84] L. Breiman, J. Friedman, A. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth, Belmont, 1984.

[BGL+99] M. Brown, W. Grundy, D. Lin, N. Cristianini, C. Sugnet, M. Ares Jr., and D. Haussler. Support vector machine classication of microarray gene expression data. Technical report, Univ. of California at Santa Cruz, 1999.

[BGV92] B. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In *Proc. of 5th Annual ACM Workshop on Computational Learning Theory*, pages 144–152. ACM PRESS, 1992.

[Bis95] C. Bishop. *Neural Networks for Pattern Recognition*. Clarendon Press, OXFORD, 1995.

[Bre96] L. Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 1996.

[Bre98] L. Breiman. Arcing classifiers. *Annuals of Statistics*, 26:801–824, 1998.

[Bur98] C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[CLT01] C. Campbell, Y. Li, and M. Tipping. An efficient feature selection algorithm for classification of gene expression data, 2001.

[CS00] K. Crammer and Y. Singer. On the learnability and design of output codes for multiclass problems. *Computational Learning Theory*, pages 35–46, 2000.

[CST00] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines*. Cambridge University Press, 2000.

[DFS00] S. Dudoit, J. Fridlyand, and T. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. Technical report, Berkeley, June 2000.

[DGB02] W Dubitzky, M Granzow, and D Berrar. *Comparing Symbolic and Subsymbolic Machine Learning Approaches to Classification of Cancer and Gene Indentification*. Kluwer Academic, 2002.

[DPBea96] J. DeRisi, L. Penland, P. Brown, and et al. Use of a cdna microarray to analyse gene expression patterns in human cancer. *Natural Genetics*, 4:i457–460, 1996.

[EH51] E.Fix and J. Hodges. Discriminatory analysis, nonparametric discrimination: Consistency properties. Technical report, USAF School of Aviation Medicine, 1951.

[FCD+01] T. Furey, N. Cristianini, N. Duffy, D. Bednarski, M. Schummer, and D. Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 2001.

[Fis36] R. Fisher. The use of multiple measurements in taxomonic problems. *Annual of Eugenics*, 7:179–188, 1936.

[FKRWea01] K. Fujarewicz, M. Kimmel, J. Rzeszowska-Wolny, and et al. Improved classification of gene expression data using support vector machines. *Journal of Medical Informatics and Technologies*, 6, Nov 2001.

[FNP00] N. Friedman, M. Nachman, and D. Pe'er. Using baysian networks to analyze expression data. In *Proc. of the 4th Ann. Int. Conf. on Comp. Molecule Biology*, 2000.

[FS97] Y. Freund and R. Shapire. A decision-theoretic generalization of on-line learning and application to boosting. *Journal of Computer and System Sciences*, 55:119–139, 1997.

[FS98] Y. Freund and R. Schapire. Large margin classification using the perceptron algorithm. In *Proc. of the 11th Annual Conf. on Comp. Learning Theory*, 1998.

[GGRL99] J. Gehrke, G. Ganti, R. Ramakrishnan, and W. Loh. Boat-optimistic decision tree construction. In *Proc. of 1999 SIGMOD Conference*, 1999.

[GST+99] T.R. Golub, D.K. Slonim, P. Tamayo, M. Gaasenbeek C. Huard, J.P. Mesirov, H. Coller, M. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, and E.S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, pages 531–537, Oct 1999.

[GWB+00] I. Guyon, J. Weston, S. Barnhill, M. D., and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 2000.

[HL01] C. Hsu and C. Lin. A comparison on methods for multi-class support vector machines. Technical report, National Taiwan University,Taipei, Taiwan, 2001.

[Jol86] I. Jollife. *Principle Component Analysis*. Springer-Verlag, 1986.

[JTC99] J.Shawe-Taylor and N. Cristianini. Furthur results on the margin distribution. In *Proc. 12th Annual Conf. on Computational Learning Theory*, 1999.

[KS96] D. Koller and M. Sahami. Towards optimal feature selection. In *Machine Learning: Proc. of 13th Int. Conf.*, 1996.

[KSHR00] A. Keller, M. Schummer, L. Hood, and W. Ruzzo. Bayesian classification of dna array expression data. Technical report, University of Washington, August 2000.

[KWR+01] J. Khan, J. Wei, M. Ringner, L. Saal, and et. al. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine*, 2001.

[LA01] S. Lakhani and A. Ashworth. Microarray and histopathological analysis of tumours: the future the past? *Nature Reviews Cancer*, pages 151–157, Nov 2001.

[LDB+96] D. Lockhart, H. Dong, M. Byrne, M. Follettie, M. Gallo, M. Chee, M. Mittmann, C. Wang, M. Kobayashi, and H. Horton. Expression monitoring by hybridization to high-density oligonucleotide arrays. *Nature Biotechnology*, 14:1675–1680, 1996.

[LLW01] Y. Lee, Y. Lin, and G. Wahba. Multicategory support vector machines. Technical report, University of Wisconson-Madison, 2001.

[MBB98] L. Manson, P. Bartlett, and J. Baxter. Direct optimization of margins improves generalization in combined classifiers. Technical report, Australia National University, 1998.

[MTS+99] S. Mukherjee, P. Tamayo, D. Slonim, A. Verri, T. Golub, J. Mesirov, and T. Poggio. Support vector machine classification of microarray data, 1999.

[NR02] D. Nguyen and D. Rocke. *Classification of Acute Leukemia based on DNA Microarray Gene Expressions using Partial Least Squares*. Kluwer Academic, 2002.

[PTGea02] S. Pomeroy, P. Tamayo, M. Gassenbeek, and et al. Prediction of central nervous embryonal tumour outcome based on gene expression. *Nature*, pages 436–442, 2002.

[RS98] R. Rastogi and K. Shim. Public: A decision tree classifier that integrates building and pruning. In *Proc. of 1996 Int. Conf. on Very Large Databases(VLDB96)*, 1998.

[RTRea01] S. Ramaswamy, P. Tamayo, R. Rifkin, and et al. Multiclass cancer diagnosis using tumor gene expression signatures. *PNAS*, 98(26):15149–15154, Dec 2001.

[Rus00] P. Russel. *Fundamentals of Genetics*. Addison Wesly Longman Inc., 2000.

[SAM96] J. Shafer, R. Agrawal, and M. Mehta. Sprint: A scalable parallel classifier for data mining. In *Proc. of 1996 Int. Conf. on Very Large Databases(VLDB96)*, 1996.

[SBS00] A. Smola, P. Bartlett, and B. Scholkopf. *Advances in Large-Margin Classifiers*. MIT Press, 2000.

[Sea01] T. Sorlie and et al. Gene expression patterns of breast carcinomas distinguish tumor subclass with clinical implications. In *Proc. of National Academy of Science*, pages 10869–10874, 2001.

[SFBL98] R. Schapire, Y. Freund, P. Bartlett, and W. Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *The Annuals of Statistics 26*, 1998.

[SS88] W. Siedlecki and Sklansky. On automatic feature selection. *International Journal of Pattern Recognition and Artificial Intelligence*, 2:197–220, 1988.

[SSDB95] M. Schena, D. Shalon, R. Davi, and P. Brown. Quantitative monitoring of gene expression patterns with a complementary dna microarray. *Science*, 270:467–470, 1995.

[STM+00] D. Slonim, P. Tamayo, J. Mesirov, T. Golub, and E. Lander. Class prediction and discovery using gene expression data. In *Proc. 4th Int. Conf. on Computational Molecular Biology(RECOMB)*, pages 263–272, 2000.

[Tre01] V. Tresp. Scaling kernel-based systems to large data sets. *Data Mining and Knowledge Discovery*, 5(9), 2001.

[Utg89] P. Utgoff. Incremental induction of decision trees. *Machine Learning*, 4:161–186, 1989.

[Vap98] V. Vapnik. *Statistical Learning Theory*. Wiley, New York, NY, 1998.

[VDBea02] L. Veer, H. Da, M. Bijver, and et al. Gene expression profiling predicts cinical outcome of breast cancer. *Nature*, pages 530–536, 2002.

[VJ02] L. Veer and D. Jone. The microarray way to tailored cancer treatment. *Nature Medicine*, pages 13–14, Jan 2002.

[WMC+00] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik. Feature selection in svm. *Advances in Neural Information Processing Systems*, 2000.

[XJK01] E. Xing, M. Jordan, and R. Karp. Feature selection for high-dimensional genomic microarray data. In *Proc. of the 18th Int. Conf. on Machine Learning*, 2001.

[Zea01] D. Zajchowski and et al. Identification of gene expression profiles that predict the aggressive behavior of breast cancer cells. *Cancer Research*, pages 5168–5178, 2001.

[ZYSX01] H. Zhang, C. Yu, B. Singer, and M. Xiong. Recursive partitioning for tumor classification with gene expression microarray data. *PNAS*, 98:6730–6735, June 2001.