Bayesian Classification of DNA Array Expression Data

Andrew D. Keller[1,2]   Michel Schummer[3]

Lee Hood[2]   Walter L. Ruzzo[1]

Technical Report UW-CSE-2000-08-01

August, 2000

[1]Department of Computer Science and Engineering, University of Washington, Seattle, WA 98195

[2]Institute for Systems Biology, Seattle, WA 98105

[3]Department of Molecular Biotechnology, University of Washington, Seattle, WA 98195

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

University of Washington

Seattle 98195

Bayesian Classification of DNA Array Expression Data

Andrew D. Keller[1,2] Michel Schummer[3]
Lee Hood[2] Walter L. Ruzzo[1]

Abstract

DNA arrays yield a global view of the cell by enabling the measurement of expression levels of thousands of genes simultaneously. When used to compare normal tissues and tissues at various stages of disease, or diseased tissues with different responses to treatment, arrays present opportunities for improved disease diagnosis and a deeper understanding of the molecular basis of observed phenotypes. Several machine learning methods have been applied to array data to classify genes on the basis of their expression levels in particular samples, and to classify tissue samples on the basis of their global patterns of gene expression [2-4,9,12,21]. These tasks are made more difficult by the noisy nature of array data, and when classifying tissues, by the overwhelming number of gene attributes relative to the number of training samples. In this paper, we present a naive Bayes method for classifying tissues on the basis of DNA array data, and use a likelihood-based metric to select the most useful subset of genes for inclusion in the classifier. We applied this method to data sets with tissues of two different classes, and found its accuracy to exceed that of a recently described method [12,21] in two of the three cases. Furthermore, our method is easily extendible to multiclass classification, and performed well when applied to a data set with three different classes of tissues.

[1]Department of Computer Science and Engineering, University of Washington, Seattle, WA 98195
[2]Institute for Systems Biology, Seattle, WA 98105
[3]Department of Molecular Biotechnology, University of Washington, Seattle, WA 98195 **1.**

Introduction

DNA arrays now offer the ability to measure the levels of expression of thousands of genes simultaneously. These arrays consist of large numbers of specific oligonucleotides or cDNA sequences, each corresponding to a different gene, affixed to a solid surface at very precise locations. When an array chip is hybridized to labeled cDNA derived from a particular tissue of interest, it yields simultaneous measurements of the mRNA levels in the sample for each gene represented on the chip. Since mRNA levels are expected to correlate roughly with the levels of their translation products, the active molecules of interest, array results can be used as a crude approximation to the protein content and thus the 'state' of the sample. Ideally, one would like in addition to measure the levels of proteins in a cell directly, and such technology is currently being developed [13].

DNA arrays yield a global view of gene expression and can be used in a number of interesting ways. For example, clustering can be performed in order to identify genes that are regulated in a similar manner under a number of different environmental conditions [1,4,9]. Such analysis can be used to surmise the unknown functions of genes based upon the known functions of other genes in the same cluster. When applied to samples prepared at various times following specific environmental perturbations or in different genetic backgrounds, arrays can be used to infer regulatory pathways at the level of transcription. Toward that aim, Bayesian networks have recently been inferred from array data to elucidate probabilistic relationships between the expression of different genes [11]. DNA arrays can be used to characterize the cellular differences between different tissue types, such as between normal cells and cancer cells at different stages of tumor progression, or between cancers with different responses to treatment, or between control cells and cells treated with a particular drug. Such analysis can potentially yield useful diagnostic tools for classifying samples on the basis of their gene expression patterns [2,12].

Classification of tissues on the basis of DNA array data presents several algorithmic challenges. For example, the data often contain 'technical' noise that can be introduced at a number of different stages, such as production of the DNA array, preparation of the samples, hybridization between cDNA and array, and signal analysis and extraction of the hybridization results. Schena *et al.* [18] tried to reduce some of this noise by simultaneously hybridizing both a test and reference sample to an array, each labeled with a different color fluorescent dye. Additional 'biological' noise can come from non-uniform genetic backgrounds of the samples being compared, or from the impurity or misclassification of tissue samples. Furthermore, array data contain an overwhelming number of attributes relative to the number of training samples, since each experiment yields the levels of expression of thousands of genes. One expects that the majority of such genes are irrelevant to the class distinction one wants to learn. The combined effect of large numbers of irrelevant genes could potentially drown out the contributions of the relevant ones.

We describe here a naïve Bayes algorithm and gene selection scheme that has a probabilistic basis and should cope with the specific challenges inherent in array data such as noise and the large number of attributes. The method specifically identifies those genes that are most likely to confer high classifier accuracy, and hence those that could likely lend insight into the biological basis of class distinction. Unlike previously described methods to classify tissues using array data, the naïve Bayes method described here is easily generalized for classification among any number of classes. Furthermore, this method has scalable computation time and memory requirements and will likely be applicable as the amount of DNA array data greatly increases, and even to future experiments in which the levels of cell proteins rather than mRNAs are measured directly. In section 2 we discuss algorithms applied to DNA array data and compare them with our method. In section 3, we evaluate the performance of the naive Bayes classifier when applied to three data sets. In section 4 we discuss our findings and speculate on possible further improvements.

2. Classification Methods

Several methods have been used to classify tissues on the basis of DNA array data. The problem can be stated as follows, where a 'sample' consists of the levels of expression in a particular tissue of each gene represented on the array: Given a set of training samples drawn from some probability distribution, each assigned a class, and a test sample drawn from the same probability distribution, determine the class of the sample. Ben-Dor *et al.* explored using Nearest Neighbor classification, Support Vector Machines, Boosting, and a clustering based approach [3]. Each method involves a supervised learning phase, during which samples with known classes are used to 'learn' distinguishing features among the classes. Although they appear to perform roughly comparably on test data sets, each method has particular strengths and weaknesses with regard to DNA array data, as described below.

2.1 Nearest Neighbor (NN)

Nearest Neighbor is a lazy classifier in which computation is deferred until classification time; training merely involves storing all the training samples in memory [8]. Classification of a sample consists of assigning it the class of the training sample that is closest to it according to a distance metric, such as Pearson correlation used by [3]. Sensitivity to noise in the data can be greatly reduced by classifying a sample according to the majority class of the N closest training samples, where $N > 1$. However, most distance metrics, including the Pearson correlation, are expected to become less sensitive as the dimensionality of the 'noisy' data increases, thus limiting the performance of NN when applied to array data. In addition, NN does not identify genes most useful for class distinctions and has large memory requirements, since it must maintain all training data in memory.

2.2 Support Vector Machines (SVM)

Support Vector Machines are a method for finding a hyperplane in high dimensional space that separates training samples of each class while maximizing the minimum distance between that hyperplane and any training sample [4,5]. If the data are not linearly separable, they can be projected onto a higher dimensional 'feature' space in which they are separable. Upon training, the SVM identifies those samples that are closest to the hyperplane, and thus which play a greater role in classifying a test sample. The SVM is therefore particularly effective in cases with a large number of samples, such as the use of array data to classify genes rather than tissues. The method can perform well in the presence of noisy data and large numbers of attributes. However, it does not identify those attributes most useful for classification, and therefore could not lend any insight into the molecular basis for tissue class distinction.

2.3 Boosting

Boosting is a method of aggregating many models produced by a 'weak learner' into an effective classifier [10]. In each iteration of the algorithm, a new model trained to emphasize those training samples misclassified by models of the previous iterations is produced. The final aggregate classifies a sample according to votes from its models, each weighted according to its accuracy on the data with which it was trained. While the aggregation of models should reduce classifier error in general, boosting could perform poorly if some of the training samples are mis-labeled. This is because in each iteration, training emphasizes those samples, including perhaps mis-labeled samples, which had been misclassified during previous iterations. Ben-Dor *et al.* [3] used a single gene and a threshold value as their weak learner. If that gene's expression level in a sample is below the threshold, the model votes for one class, and if the gene's expression level is above, the model votes for the other class. Since training the weak learner results in the selection of a single gene, the boosting aggregate of such weak learners should perform well even in the presence of overwhelming numbers of attributes, and furthermore, identifies a subset of genes of potential biological interest. However, the method is computationally expensive since the entire training data set must be examined to train the weak learner during each iteration.

2.4 Clustering-based Classification

Ben-Dor *et al.* [3] describe a clustering based approach to classification of DNA array data, whereby training samples and the test sample to be classified are mixed together and clustered without supervision to produce a specified number of clusters. The class of the test sample is then determined according to the majority class of the training samples with which it is clustered. The optimal number of clusters must be determined by trying several different values and evaluating the homogeneity of the resulting clusters with respect to class. This method should perform well in the presence of noisy data as long as the number of genes is limited, since like NN, the method relies on having a sensitive metric for the distance between two samples. In order to reduce the dimensionality of the data, [3] implements an orthogonal gene selection step prior to clustering. Genes are chosen according to how accurately they can partition the training samples along class lines with a single threshold value, such that training samples of one class have values of that gene above the threshold, and training samples of the other class, below. This selection improves the performance of the method, and identifies a set of genes important for classification. However, there is no evidence that their choice of gene selection is optimal for their algorithm.

2.5 Golub-Slonim (G-S) algorithm

A promising classification method for DNA array data was recently described by Golub *et al.* [12] and Slonim *et al.* [21], and will be referred to here as the 'G-S algorithm'. The algorithm, which is only applicable to data sets with two classes, uses the training data to compute a mean and standard deviation for each gene's level of expression among samples of each class. The class of a test sample is then determined according to how close its gene values are to the respective gene value means for each class.

The G-S algorithm includes a gene selection step to reduce the dimensionality of the data prior to classification. Genes are chosen that display the best separation between means for the two classes, as measured by the 'G-S correlation' metric:

$$\text{G-S correlation (gene } g) = (\mu^g_1 - \mu^g_2)/(\sigma^g_1 + \sigma^g_2)$$

where μ^g_1, σ^g_1 and μ^g_2, σ^g_2 are the mean and standard deviation for values of gene g among training samples of class 1 and 2, respectively. Genes with the most positive and most negative G-S correlation values are selected in parallel and grouped together in equal numbers in the final classifier.

Given a classifier with n genes and test sample vector $\boldsymbol{x} = \{ x_1, x_2, ..x_g.., x_n \}$, where each component x_g is the value of expression of gene g in that sample, classification is achieved by computing the difference between each gene's vector component of $\boldsymbol{x}$ and the average of its two class means, $(\mu^g_1 + \mu^g_2)/2$. The predicted class is then determined according to the sign of the sum of such differences over all genes in the classifier, each weighted by its G-S correlation:

$$\text{class}(\boldsymbol{x}) = \text{sign} \sum_{\text{genes } g} \{ [x_g - (\mu^g_1+\mu^g_2)/2] \ [(\mu^g_1 - \mu^g_2)/(\sigma^g_1 + \sigma^g_2)] \}$$

where a positive value of the sum indicates class 1, and a negative value, class 2.

The G-S algorithm is simple and appears to work quite well. However, it has some shortcomings. For example, the choice of threshold value for each gene, $(\mu^g_1 + \mu^g_2)/2$, is not readily justifiable; there could perhaps be a better choice of that divider. Another weakness is that it does not lend itself naturally toward a method for multiclass classification. Finally, its method of gene selection tends to avoid genes for which class values have large standard deviations with respect to the training data. Such cases may be quite prevalent in array data, however, and among the most relevant and biologically informative. For example, many cancerous cells are associated with elevated rates of somatic mutation [16]. One might expect some genes that are tightly regulated in normal tissues (and thus have a small standard deviation of expression values) to have highly variable levels of expression in a genetically heterogeneous population of cancerous cells (and thus have a large standard deviation of values). Additionally, in cases in which a labeled tissue type, for example a cancer tissue, is actually impure, or composed of two different sub-types, one might find that some genes are expressed at 'normal' levels in one cancer sub-type, and at 'cancer' levels in the other sub-type. Such genes would likely have low G-S correlation scores despite the fact that they contain useful information for classification and could potentially lead to an increased biological understanding of the nature of class differences.

2.6 Naive Bayes (NB) algorithm

In this section, we describe a naive Bayes classifier for array data and a gene selection scheme explicitly designed to optimize it. This selection, based upon a likelihood metric, is applicable to data with any number of classes.

2.6.1 Naive Bayes Classifier

The naive Bayes method (NB) is a simple approach to probabilistic induction that has been successfully applied in a number of machine learning applications [8]. According to the method, given various class models for the data, for example model M_i for class i, and a test sample vector $\boldsymbol{x}$ =

$\{x_1,x_2,..x_g...x_n\}$ drawn from some probability distribution, one can classify $\boldsymbol{x}$ according to the model with maximum *a posteriori* probability (or log *a posteriori* probability), given the sample:

$$\text{class}(\boldsymbol{x}) = \underset{i}{\text{argmax}}\ (\log p(M_i \mid \boldsymbol{x}))$$

where $p(M_i \mid \boldsymbol{x})$ is the Bayesian *a posteriori* probability that M_i is true given the test sample $\boldsymbol{x}$. By Bayes rule,

$$p(M_i \mid \boldsymbol{x})\, p(\boldsymbol{x}) = p(\boldsymbol{x} \mid M_i)\, p(M_i)$$

and assuming equal prior probabilities, $p(M_i)$, for each model, we obtain:

$$\text{class}(\boldsymbol{x}) = \underset{i}{\text{argmax}}\ (\log p(\boldsymbol{x} \mid M_i))$$

i.e. the computed class of the sample is the model for which the sample has the greatest likelihood. Finally, the naive Bayes method makes the additional assumption that, given the class model, values for each component of $\boldsymbol{x}$ are independent of one another, so the above becomes:

$$\text{class}(\boldsymbol{x}) = \underset{i}{\text{argmax}}\ (\sum_{g} \log p(x_g \mid M_i))$$

This assumption of class attribute independence greatly facilitates computation of the likelihoods for the data, given each model, since it is much easier to infer individual class attribute value probabilities from the training data than it is to infer joint class attribute value probabilities. This simplification has been used successfully in a number of domains, including some with known class attribute dependencies [6].

In the case of DNA array data, we model each class as a set of Gaussian distributions, one for each gene computed from the training samples of that class:

$$M_i = \{ M^1_i, M^2_i, \dots M^g_i \dots M^n_i \}$$

where M^g_i is the class i Gaussian distribution for gene g. The class of a test sample $\boldsymbol{x}$ is then given by:

$$\text{class}(\boldsymbol{x}) = \underset{i}{\text{argmax}}\ (\sum_{\text{gene } g} \log p(x_g \mid M^g_i))$$

which, when substituting M^g_i for a Gaussian distribution with sample mean μ^g_i and standard deviation σ^g_i, becomes:

$$\text{class}(\boldsymbol{x}) = \underset{i}{\text{argmax}}\ \{ \sum_{\text{gene } g} [-\log(\sigma^g_i) - 0.5\ ((x_g - \mu^g_i)/\sigma^g_i)^2] \}$$

since $p(x_g \mid M^g_i)$ is proportional to $(1/\sigma^g_i) \exp(-0.5((x_g - \mu^g_i)/\sigma^g_i)^2$, if interpreted as the probability that the gene g component of $\boldsymbol{x}$ is within some small non-zero interval centered at x_g. Furthermore, if one again assumes equal prior probabilities for all models, the relative log probabilities between any two models M_a and M_b with respect to $\boldsymbol{x}$ can be expressed simply as the difference between their log likelihoods:

$$\log p(M_a \mid \boldsymbol{x}) - \log p(M_b \mid \boldsymbol{x}) = \log p(\boldsymbol{x} \mid M_a) - \log p(\boldsymbol{x} \mid M_b) =$$

$$\sum_{\text{gene } g} [-\log(\sigma^g_a) - 0.5\ ((x_g - \mu^g_a)/\sigma^g_a)^2 + \log(\sigma^g_b) + 0.5\ ((x_g - \mu^g_b)/\sigma^g_b)^2]$$

Such a difference can be used as a confidence measure for choosing class a over class b.

2.6.2 Likelihood Selection of Genes for the NB classifier: Two class case

In the two class case, genes in the NB classifier each vote for the likelihood of alternative models, M^g_1 and M^g_2, given the test sample vector component x_g. Intuitively, we want genes that can distinguish between samples of each class, finding $M^g{}_1$ more likely than $M^g{}_2$ given a sample of class 1, and $M^g{}_2$ more likely than $M^g{}_1$ given a sample of class 2. We define two relative log likelihood scores, $LIK_{1\rightarrow 2}$ and $LIK_{2\rightarrow 1}$ for gene g:

$LIK_{1\rightarrow 2} = log\ p(M^g{}_1 \mid \boldsymbol{X_1}) - log\ p(M^g{}_2 \mid \boldsymbol{X_1})$, where $\boldsymbol{X_1}$ are training samples of class 1

$LIK_{2\rightarrow 1} = log\ p(M^g{}_2 \mid \boldsymbol{X_2}) - log\ p(M^g{}_1 \mid \boldsymbol{X_2})$, where $\boldsymbol{X_2}$ are training samples of class 2

The 'ideal' gene for the NB classifier would be expected to have both LIK scores much greater than zero, indicating that it on average votes for class 1 on training samples of class 1, and for class 2 on training samples of class 2. If a test sample is selected from the same probability distribution as the training data, then one can expect this gene to likewise vote for class 1 on average for test samples of class 1, and for class 2 for test samples of class 2. The greater the values of the LIK scores are above zero, the greater contribution one expects the gene to make toward the correct classification of a test sample.

In practice, it is difficult to find genes for which both LIK scores are far greater than zero (see Discussion). Instead, one can select two sets of genes, $\text{GENES}_{\mathbf{1\rightarrow 2}}$ and $\text{GENES}_{\mathbf{2\rightarrow 1}}$, each maximizing one of the two LIK scores while merely requiring the other to be greater than zero:

$\text{GENES}_{\mathbf{1\rightarrow 2}}$: $LIK_{1\rightarrow 2} >> 0$ and $LIK_{2\rightarrow 1} > 0$

$\text{GENES}_{\mathbf{2\rightarrow 1}}$: $LIK_{1\rightarrow 2} > 0$ and $LIK_{2\rightarrow 1} >> 0$

Genes in each set are ranked according to their value of the LIK score maximized by that set. An NB classifier with n genes is then produced by combining the $n/2$ top ranking genes from each set.

2.6.3 Generalizing Likelihood gene selection to the case of more than two classes

This method for using LIK scores to select genes for a naive Bayes classifier extends beyond the case of two classes. In the general case where the number of classes is c, we define $c(c\text{-}1)$ different LIK scores:

$LIK_{j\rightarrow k} = \log p(M^g{}_j \mid \boldsymbol{X_j}) - \log p(M^g{}_k \mid \boldsymbol{X_j})$, where $\boldsymbol{X_j}$ are training samples of class j

and $1 \le j,k \le c,\ j \ne k$. Similarly, we select $c(c\text{-}1)$ distinct sets of genes, each maximizing one particular LIK score while merely requiring all others to be greater than zero:

$\text{GENES}_{\mathbf{j\rightarrow k}}$: $LIK_{j\rightarrow k} >> 0$

$LIK_{j'\rightarrow k'} > 0 \quad j' \ne k',\ 1 \le j',k' \le c$

Genes in each $\text{GENES}_{\mathbf{j\rightarrow k}}$ set should therefore best distinguish test samples of class j with respect to the alternative model $M^g{}_k$.

When equal numbers of genes from all $c(c\text{-}1)$ $\text{GENES}_{\mathbf{j'\rightarrow k'}}$ sets are combined, the resulting NB classifier should again have the desired properties. Consider a test sample $\boldsymbol{x}$ of class j. Genes in the $c\text{-}1$ different $\text{GENES}_{\mathbf{j\rightarrow k'}}$ sets, $1 \le k' \le c,\ k' \ne j$, on average make a contribution to the log likelihood term of $M^g{}_j$ that is much larger than its contribution to the term of $M^g{}_{k'}$, and at least as large as that to all other terms. Genes in the other $(c\text{-}1)^2$ sets of $\text{GENES}_{\mathbf{j'\rightarrow k'}}$, $1 \le j',k' \le c,\ j' \ne k',\ j' \ne j$ will on average make a contribution to the log likelihood term of $M^g{}_j$ at least as large as that to terms of the alternatives. As a result, the summed log likelihood term of $M^g{}_j$ will on average be larger than that of all other models, so

$\underset{i}{\operatorname{argmax}} \log p(\mathbf{x} \mid M_i) = j$ and the classifier votes for class j.

3. Results

3.1 Data sets

We tested the Likelihood selection and NB classifier, as well as the previously described G-S classifier, on the three different data sets described in Table 1. The colon data [2] contains 62 samples of which 22 are from normal colon tissues and the remaining from colon cancer, each including gene expression values for 2000 different genes measured using Affymetrix array technology. These genes were selected from a total of 6600 by [2] based upon their having strong signals. The ovary data [19,20] contains 31 samples, 15 of which are derived from normal tissues, and the remaining 16 from ovarian cancers in various stages of malignancy. Each sample includes the levels of expression for 97802 cDNA clones of which approximately one third are unique. Data was produced from hybridization to filter arrays, and values represent absolute levels of mRNA. Finally, the leukemia data set [12,21] contains a training set composed of 27 samples of acute lymphoblastic leukemia (ALL) and 11 samples of acute myeloblastic leukemia (AML); each includes the levels of expression of 7129 genes using Affymetrix array technology. These two classes of leukemia arise from different cell lineages, and differ in their prognosis and response to treatment. This data set also includes an independent test set containing 20 ALL and 14 AML samples. The ALL samples can be further characterized as belonging to either the distinct ALL-B or ALL-T subclass, according to whether they arise from a B or T cell lineage. The ALL samples in the training set are composed of 19 ALL-B and 8 ALL-T samples, and those in the test set are composed of 19 ALL-B and 1 ALL-T samples. The leukemia data set thus can be used both to test two-class classification for distinguishing ALL from AML samples, or to test three-class classification for distinguishing the ALL-B, ALL-T, and AML classes.

Data set	*Number of genes*	*Classes*	*Training data*	*Test data*	*Reference*
Colon	2000	normal cancer	22 40		http://www.molbio.princeton,edu/colondata
Ovary	97802	normal cancer	15 16		[19, 20]
Leukemia 2-class	7129	ALL AML	27 11	20 14	http://waldo.wi.mit.edu/MPR/data_sets.html
Leukemia 3-class	7129	ALL-B ALL-T AML	19 8 11	19 1 14	http://waldo.wi.mit.edu/MPR/data_sets.html

Table 1: Data sets used in this study.

3.2 Evaluation of classifiers

The NB and G-S classification methods each describe a means for selecting genes included in the classifier as well as a means for using those selected genes to classify test samples. Since the gene selection and classification steps are orthogonal to one another, we evaluated all four combinations of selection (LIK or G-S) and classification (NB or G-S), implemented as described below:

Likelihood Selection

Genes are selected in parallel for all $c(c\text{-}1)$ $\mathrm{GENES}_{\mathbf{j}\rightarrow\mathbf{k}}$ sets, where c is the number of classes. Each $\mathrm{GENES}_{\mathbf{j}\rightarrow\mathbf{k}}$ set, $1\leq j,k\leq c,\ j\neq k$, maintains a list of genes ordered by decreasing value of $\mathrm{LIK}_{j\rightarrow k}$, the particular LIK value to be maximized by that set. Genes are added to the $c(c\text{-}1)$ lists only if all of their $c(c\text{-}1)$ LIK scores computed from the training data are greater than zero. A classifier with total number of genes n is then constructed by grouping together the top ranking $n/(c(c\text{-}1))$ genes from each set.

G-S Selection

Genes are selected in parallel for those with the most positive and negative values of G-S correlation, respectively. Each set maintains a list of genes in ranked order, one in decreasing magnitude of G-S correlation (Positive Correlation genes), and the other in increasing magnitude (Negative Correlation genes). A classifier with total number of genes n is then constructed by grouping together the top ranking $n/2$ genes from each set. The G-S algorithm is only applicable to data sets with two classes.

NB Classification

$$\mathrm{class}(\boldsymbol{x}) = \underset{i}{\mathrm{argmax}}\ \{ \sum_{\text{gene } g \text{ in classifier}} [-\log(\sigma^g{}_i) - 0.5\ ((x_g - \mu^g{}_i)/\sigma^g{}_i)^2] \}$$

where $\mu^g{}_i$ and $\sigma^g{}_i$ are calculated for each gene g from training samples of each class i.

G-S Classification

$$\mathrm{class}(\boldsymbol{x}) = \mathrm{sign}\ \{ \sum_{\text{gene } g \text{ in classifier}} [x_g - (\mu^g{}_1+\mu^g{}_2)/2]\ [(\mu^g{}_1- \mu^g{}_2)/(\sigma^g{}_1+ \sigma^g{}_2)] \}$$

(positive for class 1 and negative for class 2) where $\mu^g{}_1$, $\sigma^g{}_1$ and $\mu^g{}_2$, $\sigma^g{}_2$ are computed for each gene g from training samples of class 1 and 2, respectively.

Classifiers with a range of different numbers of genes were tested on the three data sets. For the classifiers trained on 2-class data, this number ranged from 10 (two sets of 5 genes) to 1000 (two sets of 500 genes). In the case of the colon and ovary data sets, the classifiers were assessed by a leave-one-out cross validation method, in which different classifiers are constructed, each trained on the samples excluding a different single sample. The classifier is then used to classify the sample not used in training. For the leukemia data set that has separate training and test samples, the training samples were used to construct the classifiers and the test set, for evaluation. Classifier accuracy was computed as the fraction of test samples classified correctly.

In the case of the ovary and leukemia data sets, the LIK/NB classifier outperformed the G-S/G-S classifier over a wide range of classifier sizes (Figure 1b,c). In contrast, the G-S/G-S classifier had higher accuracy than LIK/NB in the case of the colon data set (Figure 1a). In the majority of cases, each classifier worked best with its own method of selecting its genes. However there were exceptions such as the ovary

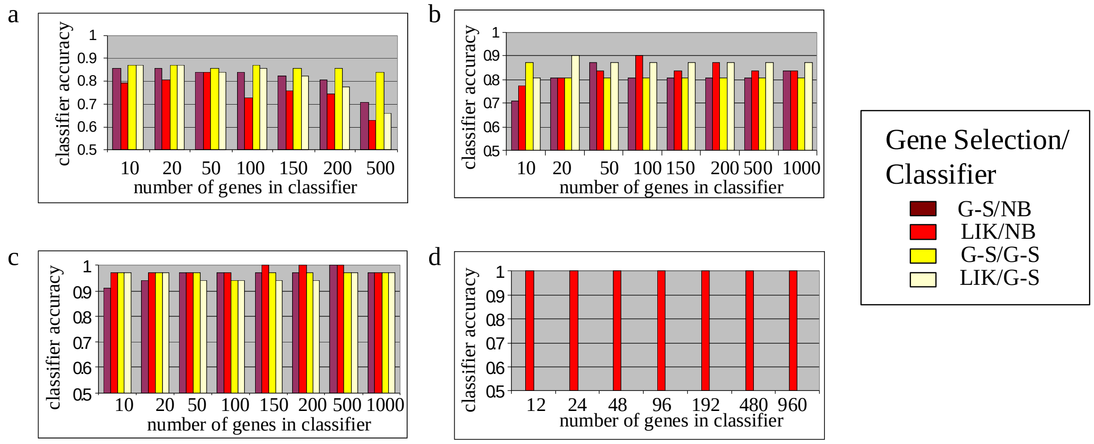


Figure 1: Accuracy rate of classifiers when applied to: a. colon data set; b. ovary data set; c. leukemia 2-class data set; d. leukemia 3-class data set. Four different classifiers are evaluated on the first three data sets, each resulting from a different combination of one of two gene selection schemes (LIK or G-S) and one of two classification schemes (NB or G-S). Only the LIK/NB classifier is evaluated on the three-class leukemia data set (part d).

data set, in which the G-S classifier worked better with the Likelihood selected genes (LIK/G-S classifier), and the colon data set, in which the NB classifier worked better with the G-S selected genes (G-S/NB classifier).

Figure 1d shows the results of the three-way classification of the leukemia data into the ALL-B, ALL-T, and AML classes using the LIK/NB classifier. Over a wide range of classifier sizes, 100% accuracy was observed. The greater accuracy of that classifier relative to the case of the two-class classification of the same data set may reflect a better fit of the data when two Gaussians are used to model the ALL class samples rather than just a single Gaussian.

In order to gain some insight into why the LIK/NB classifier performed better on some data sets than on others, the $LIK_{j\rightarrow k}$ scores of genes from each $\mathbf{GENES_{j\rightarrow k}}$ data set were plotted as a function of their rank in the set (Figure 2). The larger classifier sizes include progressively more genes, and thus include genes with progressively lower rank and LIK values. One can see a correlation between the magnitude of LIK scores observed (as a function of gene rank) and the performance of the LIK/NB classifier. Genes selected in the colon data set had the lowest overall LIK scores, those selected in the ovary data set, intermediate LIK scores, and those selected in the leukemia data set (either 2-class or 3-class), the highest. Thus in each case, the higher the LIK scores of the selected genes (as a function of gene rank), the better performance observed with the LIK/NB classifier on that data set.

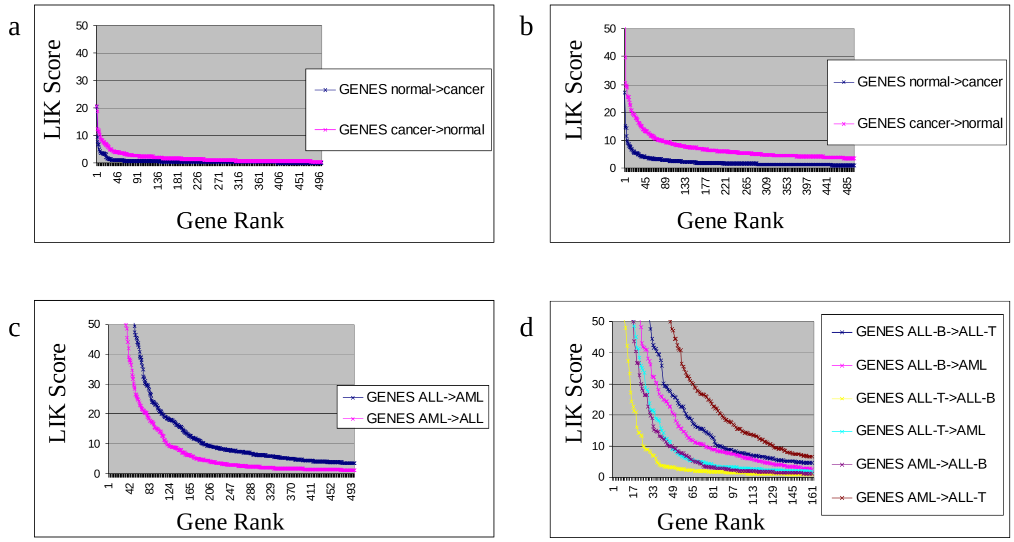


Figure 2: LIK scores of Likelihood selected genes from: a. colon data set; b. ovary data set; c. leukemia 2-class data set; d. leukemia 3-class data set. LIK scores for each gene set shown are plotted against the rank of that gene (by LIK score) in the set. Note that LIK values for many high-ranking genes selected from the leukemia data sets are off-scale in the figure.

The relevance of gene LIK scores to the performance of the LIK/NB classifier is supported by evaluating the performance of classifiers constructed from the leukemia 2-class data set using genes with various ranges of LIK values. Classifiers were made using the top ranking 25 genes from each of the two LIK-selected gene sets after first disregarding the top N genes, where N was gradually increased from 0 by increments of 12. Thus, as N was increased, genes in the resulting classifiers had progressively lower average LIK values. Figure 3 shows that, when applied to test samples and evaluated for accuracy, classifiers with genes having higher average LIK scores performed better in general than those with genes having lower average LIK scores. However, some classifiers with genes having low average LIK scores did perform well.

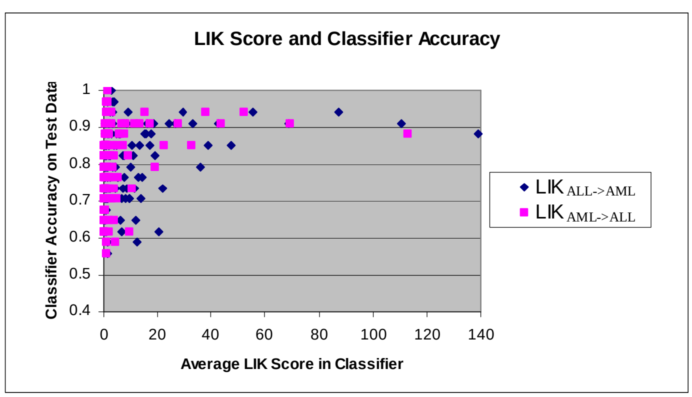


Figure 3: Relationship between classifier accuracy and average LIK score. NB classifiers containing genes with progressively lower LIK scores were tested for accuracy on the leukemia 2-class data set. Classifiers each contained 50 genes total, two sets of 25, with average LIK scores indicated.

The LIK measure offers a standardized way to limit the number of genes in NB classifiers trained on various data sets. One can require that each $\mathrm{GENES}_{\mathbf{j}\to\mathbf{k}}$ set, $1 \leq j,k \leq c,\ j \neq k$, contain identical numbers of genes, and that genes selected into each $\mathrm{GENES}_{\mathbf{j}\to\mathbf{k}}$ set have a value of $\mathrm{LIK}_{j\to k}$ greater than or equal to a pre-specified minimum value. These restrictions will limit each set to a common size, and hence will limit the size of the overall classifier. Table 2 shows resulting characteristics of NB classifiers trained on the various data sets while requiring a minimum LIK score value of 3.0. Notably, a small classifier resulted in the case of the colon data, as expected, since that data set had the fewest genes with high LIK scores. For each data set, accuracy of the classifier with the minimum 3.0 LIK threshold was near the maximal observed previously for any classifier size tested.

Data set	*Number of genes in classifier*	*Classifier accuracy*
Colon	39.4 +/- 3.6	0.84
Ovary	988 +/- 68	0.84
Leukemia 2-class	492	1.0
Leukemia 3-class	900	1.0

Table 2: Results of NB classifiers employing the minimum 3.0 LIK value criteria for gene selection. For the colon and ovary data sets, cross validation produced a set of classifiers, each with its own number of genes; hence the mean and s.d. of number of genes for the set is indicated.

3.3 Likelihood Selected genes

The LIK scores and G-S correlation values are alternative metrics for selecting genes in a classifier, and likely favor different subsets of genes. Figure 4a shows the fraction of genes in common to classifiers trained on the various data sets using either Likelihood or G-S selection. For comparison, the fraction expected by chance alone is shown as well. One can readily see that in all cases, classifiers using Likelihood and G-S selection share many more genes than expected by chance alone. However, it is also clear that they contain a significant percent of genes not shared by the other. This is supported by a direct analysis of G-S correlation scores for the Likelihood selected genes from the leukemia data set. Figure 4b shows that there is a wide variation in G-S correlation for genes throughout all ranks in the Likelihood selected sets, suggesting that classifiers of all sizes with Likelihood selected genes contain many genes not included in all but the largest classifiers using G-S selection. Such genes likely contribute productively to the accuracy of the NB classifier since Likelihood selection proved more effective than G-S selection for the NB classifier when applied to the ovary and leukemia data sets (Figure 1). Thus Likelihood selection could potentially identify additional genes that contribute to the biological basis for class distinction.

a

b

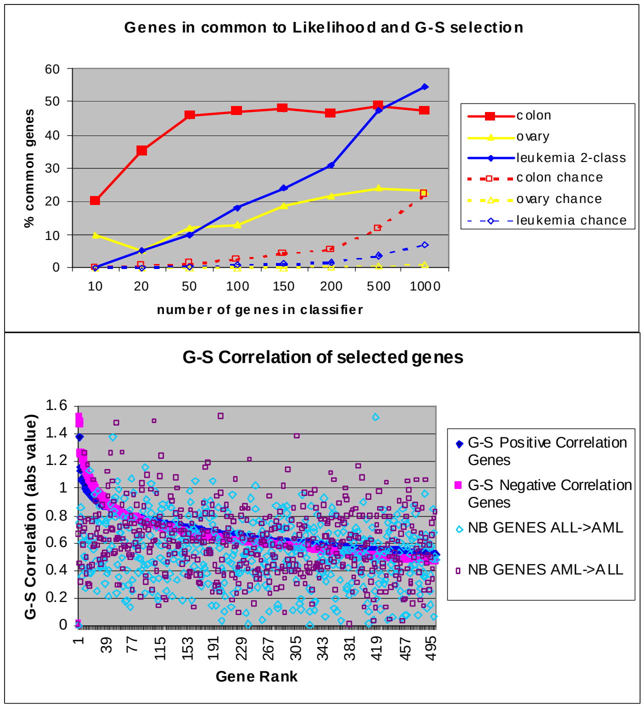


Figure 4: Comparison between Likelihood and G-S selected genes. a. Percent of genes in common between classifiers trained on various data sets using either Likelihood or G-S selection. Also shown is the percent of common genes expected by chance. b. G-S correlation scores of genes in G-S selected Positive and Negative Correlation sets, and in Likelihood selected $GENES_{ALL \rightarrow AML}$ and $GENES_{AML \rightarrow ALL}$ sets, as a function of gene rank within that set.

Table 3 lists the top 10 genes selected on the basis of their $LIK_{ALL \rightarrow AML}$ values from the leukemia 2-class data set, and similarly the top 10 genes selected on the basis of their $LIK_{AML \rightarrow ALL}$ values. Many of these genes have class Gaussian distributions with large standard deviations and low G-S correlation values, as reflected in the low G-S ranks indicated. They include several genes already implicated as having a role in cancer such as TCL1, associated with T-cell malignancies [23], CD24, an early tumor marker [14], and amphiregulin, an epidermal growth factor-related protein with tumor inhibitory activity [17]. The utility of these genes in classification warrants future experiments aimed at determining their roles, if any, in the generation and progression of the ALL and AML leukemias.

$GENES_{ALL \to AML}$ rank	*G-S Positive Correlation rank*	*Gene Description*
1	110	CD24 signal transducer
2	73	T-cell leukemia/lymphoma 1 (TCL1)
3	49	Immunoglobulin lamda gene locus
4	1097	T-cell receptor delta chain
5	306	T cell factor 1
6	830	Thymocyte AgCd1b
7	213	T cell Ag receptor gene t3delta
8	526	MAL (t-cell specific proteolipid protein)
9	605	Neuropeptide y
10	1629	Integrin alpha 6
21	235	CD2
97	36	CD19
114	573	CD7 (copy 1)
170	1154	CD3
236	905	CD10
375	1588	CD7 (copy 2)
996	145	CD22

$GENES_{AML \to ALL}$ rank	*G-S Negative Correlation rank*	*Gene Description*
1	165	Amphiregulin
2	116	Cathepsin G (serine protease)
3	537	Interleukin bsf-2
4	1072	Trypsinogen IV b form gi
5	81	Cystatin a
6	625	B cell stimulating factor 2
7	150	Neutrophil elastase
8	280	Connective tissue activation peptide III
9	5	Adipsin complement factor D
10	123	Prostaglandin endoperoxide synthase-2
46	4	CD33
345	83	CD13
1748	1771	CD14

Table 3: Rankings of Likelihood selected genes from the leukemia 2-class data set. Top 10 ranking genes, together with known marker genes, are shown for the $GENES_{ALL \to AML}$ and $GENES_{AML \to ALL}$ gene sets. Gene ranks (out of a total of 7129) indicated in bold typeface denote genes included in the NB classifier employing the minimum 3.0 LIK value requirement. Also indicated are the ranks of those genes in the Positive and Negative Correlation sets resulting from G-S selection. The data set contained two different attributes corresponding to the CD7 gene, indicated as 'copy 1' and 'copy 2'.

Several cell markers that are differentially expressed in the ALL and AML leukemias have been previously identified and used to distinguish the cancer types. They include the protein products of the CD2, CD3, CD7, CD10, CD19, and CD22 genes, expressed predominantly in ALL leukemias, and the protein products of the CD13, CD14, and CD33 genes, expressed predominantly in AML leukemias [24]. Table 3 shows that as a result of Likelihood selection, most of the ALL-specific genes had relatively high ranks in the GENES$_{\mathbf{ALL\rightarrow AML}}$ set (reflecting high LIK$_{\mathrm{ALL\rightarrow AML}}$ values), and most of the AML-specific genes had relatively high ranks in the GENES$_{\mathbf{AML\rightarrow ALL}}$ set. Genes with rank values less than or equal to 246, indicated in bold typeface, were those included in the NB classifier employing the minimum 3.0 LIK value restriction. The table also indicates the ranking for those genes in the G-S selected Positive and Negative Correlation sets. It is evident that most of the ALL specific markers have a higher rank (were selected more strongly) with respect to the Likelihood selection than the G-S selection. This likely reflects the greater heterogeneity among the ALL samples, which include both the ALL-B and ALL-T sub-classes. Some of the markers are expressed differentially in the ALL-B and ALL-T subclasses, leading to ALL class Gaussians with large standard deviations and hence lower G-S correlation scores. These observations further support the premise that the NB classifier is more robust than the G-S classifier with respect to less well-separated class Gaussian distributions.

4. Discussion

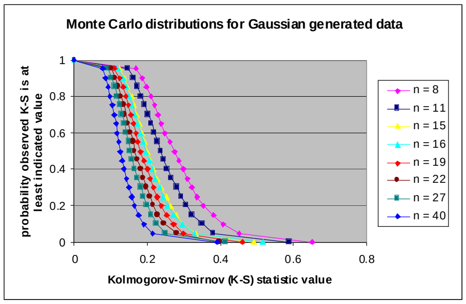
4.1 Improvements to the NB classifier

The NB classifier may not perform in practice as well as expected if some of its assumptions are not justified. For example, both Likelihood selection and NB classification assume equal prior probabilities for each class model. Additional assumptions are that class gene data fit a Gaussian distribution, and that class attribute values are independent. We discuss below whether or not these assumptions are justified, and how the algorithm could be improved if they are not.

4.1.2 Gaussian distribution assumption

A significant assumption of the Likelihood selection and NB classification scheme is that class values for expression levels of individual genes follow a Gaussian distribution. Violation of this assumption could potentially lead to suboptimal performance of the method. We used the Kolmogorov-Smirnov (K-S) statistic to estimate probabilities that observed data are generated according to a Gaussian distribution [22]. This statistic measures the maximum difference between an observed cumulative probability distribution (*e.g.* observed gene class values) and a calculated cumulative probability distribution (*e.g.* Gaussian distributions with mean and standard deviations computed from the gene class values). Initially, the probability distributions of K-S values for Gaussian-generated data sets of various sizes were computed by Monte Carlo simulation (Figure 6a). These distributions were then used to determine the percent of observed gene class distributions having calculated K-S values within 50% and 90% confidence intervals for the data being Gaussian generated. The results, shown in Figure 6b, indicate that the gene class values overall appear to fit Gaussian distributions reasonably well. With the exception of the colon data set about 2/3 of all class data fit a Gaussian distribution with over 50% confidence, and 1/4 with over 90% confidence. Interestingly, the leukemia 3-class data displayed a more confident fit to Gaussian distributions than did the leukemia 2-class data. This is consistent with the heterogeneous ALL class in the 2-class data being resolved into its ALL-B and ALL-T class components in the 3-class data.

a



b

Gene class distributions fit to Gaussian

data set	*% with >= 50% confidence*	*% with >= 90% confidence*
colon	39	7
ovary	82	28
leukemia 2-class	73	24
leukemia 3-class	78	28

Figure 5: Fit between gene class training data and Gaussian distributions. a. Probability distributions for Kolmogorov-Smirnov (K-S) statistic values resulting from Gaussian-generated data sets of indicated sizes (8,11,15,16,19,22,27,40) corresponding to the training class sizes for various data sets. b. Indicated for each data set are the percent of gene class data having K-S values within a 50% or 90% confidence interval for being Gaussian generated.

Although the assumption that class data fit a Gaussian distribution appears to be justified, it is still possible that more accurate classification could be achieved by rewarding those genes with class values having the highest confidence fit to Gaussian distributions. For example, one could require during the Likelihood gene selection process that all genes have class Gaussians with at least a minimum K-S confidence score. Alternatively, one could select genes as usual according to their LIK scores, yet during classification give higher weight to contributions from genes with more confident Gaussian fits to the training data. One could also use a non-Gaussian probability distribution to model gene class data with low K-S scores, such as discretized value counts [7] or kernel density estimation [15]. Alternatively, some data sets may be transformed to better fit a Gaussian distribution by taking the logarithm or square root of gene expression values. It will be interesting to implement these features in the future to determine their effects on classifier performance.

4.1.3 Naive Bayes assumption of class attribute independence

An additional assumption of the naive Bayes classifier is that the sample values of each gene in the classifier are independent of one another, given the class of that sample. We tested the validity of this assumption by evaluating the Pearson correlation (PC) between pairs of classifier genes with respect to training sample data of each class separately:

$$\text{PC (gene } g\text{, gene } g'\text{, class } A) = \sum_{\substack{\text{training sample } i \\ \text{of class } A}} [(x^g_i - \mu^g_i) / \sigma^g_i] [(x^{g'}_i - \mu^{g'}_i) / \sigma^{g'}_i]$$

A correlation of 1 indicates positive correlation between genes, -1 indicates negative correlation, and 0 indicates no correlation. Table 4 shows the average gene pair-wise correlation for genes within each $GENES_{j\rightarrow k}$ set of the two-class NB classifiers. In addition, an average pair-wise correlation was computed for pairs of genes randomly selected from each data set. It is evident that classifiers trained on all data sets contain genes with significant pair-wise correlation, suggesting that the independence assumption is violated. Likelihood selected genes from the leukemia data set display the least dependence followed by those from the ovary and colon data sets. The observed pair-wise correlation among genes in the ovary classifier could be due to the presence of redundant genes, since an estimated 2/3 of the genes in that data set are actually duplicates. Particularly noteworthy is the high degree of correlation among genes in the colon data set, even greater among randomly selected genes than among those selected by the NB classifier. This likely indicates a general problem with the data. For example, a large number of samples with very weak signal (background levels) might contribute to such a correlation. It is not known to what degree the observed deviations from the naive Bayes assumption of independence diminish the accuracy of the NB classifier. In recent years there have been several examples in which the naive Bayes classifier performs well, and even optimally, despite the existence of class attribute dependencies [6]. Nevertheless, there does appear to be a correlation between the degree of independence among genes within a classifier and that classifier's performance. In the future, one could increase gene independence by removing genes highly correlated with others in the classifier, or one could try to model the observed dependencies [11].

Data set	*Random Genes*	*$GENES_{1\rightarrow 2}$*	*$GENES_{2\rightarrow 1}$*
Colon	0.48 +/- 0.22	0.41 +/- 0.31	0.39 +/- 0.26
Ovary	-0.03 +/- 0.27	0.23 +/- 0.42	0.21 +/- 0.34
Leukemia 2-class	0.03 +/- 0.30	0.04 +/- 0.38	0.11 +/- 0.33

Table 4: Within-class Pearson Correlations for 300 pairs of genes either selected randomly, or from within the top 25 ranking genes from the Likelihood selected gene sets $GENES_{1\rightarrow 2}$ and $GENES_{2\rightarrow 1}$. Classes 1 and 2 represent normal and cancer, respectively, for the colon and ovary data sets, and ALL and AML, respectively, for the leukemia data set. The Pearson Correlations were computed as described in the text, over training data of the same class. Mean values and standard deviations are indicated.

4.1.2 Selecting multiple sets of genes

The assertion was made that in practice it is difficult to find genes having both $LIK_{1\rightarrow 2}$ and $LIK_{2\rightarrow 1}$ terms much greater than zero, where 1 and 2 are different classes. Hence, we justified the need to select and combine multiple sets of genes, each maximizing a single LIK score. If this assertion is not true, however, better performance could likely be achieved by selecting genes that simultaneously maximize all LIK scores. We directly assessed the relationship between $LIK_{1\rightarrow 2}$ and $LIK_{2\rightarrow 1}$ scores for all genes in the two-class data sets. From Figure 6, which plots the two LIK scores against one another, anticorrelation is apparent whereby genes with high $LIK_{1\rightarrow 2}$ scores tend to have low $LIK_{2\rightarrow 1}$ scores, and *vice versa*. This relationship can be explained by the observation that gene class distributions are often asymmetric since the class distribution with the greater mean value also has a greater standard deviation. In conclusion, our use of multiple gene sets in the NB classifier, each selected to maximize one of the LIK values, appears justified.

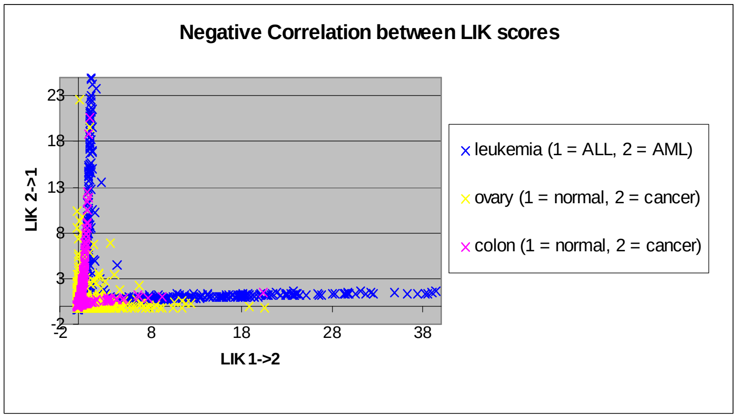


Figure 6: Negative correlation between $LIK_{1\rightarrow 2}$ and $LIK_{2\rightarrow 1}$ scores for genes in two-class data sets. Classes 1 and 2 represent ALL and AML, respectively, for the leukemia data set, and normal and cancer, respectively, for the colon and ovary data sets. Note: data with large LIK values (not shown on graph) follow same trend apparent in the figure.

4.2 Classification Strength

It is often advantageous for a classifier not only to make a decision about the class of a test sample, but to give a confidence measure of that decision (*i.e.* how likely that decision is true). One can define a metric for decision confidence and determine empirically (*i.e.* from the training data) the probability that a decision of any particular confidence value according to that metric is true. By employing a minimum confidence to classification, whereby test samples with decision confidence values below that threshold are left unclassified, one can decrease the number of false positives and false negatives at the expense of increasing the number of unclassified samples. The combination of a good metric for decision confidence and a good threshold value of that metric will result in a low false positive and/or low false negative rate without a concomitant high unclassified sample rate. The choice of appropriate decision confidence metric therefore ultimately depends on the particular classifier and how the classifier is employed. For example, one might want to minimize the false positive rate for some applications of the algorithm, and minimize the false negative rate for other applications. Possible decision strength metrics for the NB classifier include:

1. log likelihood difference of the winner class = $\log p(\mathbf{x} \mid M_{max}) - \log p(\mathbf{x} \mid M_{2nd})$

   where 'max' and '2nd' are the classes with the maximum and second largest log likelihoods, respectively.

2. relative log likelihood difference of the winner class =

$$[\log p(\mathbf{x} \mid M_{max}) - \log p(\mathbf{x} \mid M_{2nd})\,] \,/\, \sum_i \log p(\mathbf{x} \mid M_i)$$

3. absolute probability of the winner class = $p(\mathbf{x} \mid M_{max}) \,/\, \sum_i p(\mathbf{x} \mid M_i)$

In this study, we have refrained from employing decision confidence metrics and threshold values when evaluating classifiers in order to enable the most general comparisons between methods.

Ben-Dor *et al.* [3] evaluated several classifiers (NN, SVM, boosting, and a clustering based approach) on the same colon and ovary data set used in this study. However, they employed a decision confidence metric and thus left many 'low confidence' samples unclassified in an attempt to increase accuracy. As a result, we can only compare the performance of their classifiers with that of ours if we make some assumption, such as interpreting the fraction of their classified samples that were correctly classified as a best-case estimate of the accuracy of the classifier on all samples. With regard to the colon data set, the clustering based approach and SVM algorithm (estimated accuracy of .89 and .86, respectively) performed comparably to the G-S/G-S while NN and boosting (estimated accuracy of .81 and .80, respectively) performed comparably to LIK/NB. With regard to the ovary data set, SVM (estimated accuracy .95) performed better than both G-S/G-S and LIK/NB, boosting (estimated accuracy .89) comparable to LIK/NB, and the clustering based approach and NN (estimated accuracy .71 for both), worse than both algorithms. Thus, the SVM algorithm appears the most promising with respect to those two data sets. However, a more objective comparison among methods will require the actual rather than estimated accuracy of each method with respect to all samples of the data sets.

4.3 Bayesian interpretation of the G-S algorithm

It is interesting to note the similarities between the NB and G-S classifiers. For example, it has been argued that the G-S algorithm has a Bayesian interpretation under the assumptions of the NB classifier plus the additional restriction that for each gene, the two class standard deviations computed from the training data are equal, *i.e.* $\sigma_g = \sigma^g{}_1 = \sigma^g{}_2$ [21]. The argument justifies the G-S method under those assumptions, since the difference in log likelihoods of a test sample $\boldsymbol{x}$ given the two models is given by:

$$\sum_{\text{gene } g} [\, x_g - (\mu^g{}_1 + \mu^g{}_2)/2 \,]\, [(\mu^g{}_1 - \mu^g{}_2)/(\sigma_g{}^2)]$$

However, the G-S classifier does not actually compute the above sum, but rather:

$$\sum_{\text{gene } g} [\, x_g - (\mu^g{}_1 + \mu^g{}_2)/2 \,]\, [(\mu^g{}_1 - \mu^g{}_2)/(2\sigma_g)]$$

which is only equal to the difference in log likelihoods of the two models under the additional assumption that σ_g has a value of 2, and only proportional to the log likelihood difference if the standard deviations for all genes are the same. Nevertheless, we were interested in determining whether the assumption of equal class standard deviations was justifiable for the training data sets used in our study. For each data set with two classes, 1 and 2, we computed the average value of $max(\sigma^g{}_1/\sigma^g{}_2,\ \sigma^g{}_2/\sigma^g{}_1)$ among all genes in the data set, and among those genes selected by the G-S and NB classifiers. It is quite apparent that the two class standard deviations are rarely equal, even among those genes selected by the G-S classifiers (Table 5).

Data set	*All genes*	*G-S top 50*	*Likelihood top 50*
Colon	1.5 +/- 0.5	2.5 +/- 0.9	3.4 +/- 0.9
Ovary	1.3 +/- 0.4	1.9 +/- 0.7	5.8 +/- 1.8
Leukemia 2-class	1.8 +/- 2.3	4.6 +/- 3.2	22.6 +/- 12.1

Table 5: Ratios between standard deviations of gene values for alternative classes, $max(\sigma^g_1/\sigma^g_2, \sigma^g_2/\sigma^g_1)$, computed for indicated sets of selected genes.

Though the G-S classifier appears to have very limited Bayesian justification, it nonetheless performs remarkably well. Perhaps its selection scheme avoids overfitting the training data. Alternatively, it is possible that its evaluation function, $x_g - (\mu^g_1 + \mu^g_2)/2$, is more robust to the presence of non-Gaussian data or data with class attribute dependencies. It will be interesting to note in the future under what conditions the NB and G-S classifiers each perform best in order to gain further insight into how to produce the most accurate classifier for a particular data set.

4.4 Conclusion

We have described a naive Bayes (NB) algorithm for classification of DNA array data, and a novel means of gene selection aimed at optimizing classification accuracy. We compared the accuracy of this NB method to the previously described Golub-Slonim (G-S) method when applied to three different data sets. We found that the NB classifier performed better on two of the data sets (ovary and leukemia), whereas the G-S performed better on the third (colon). We find a direct correlation between the performance of the NB classifier and the magnitude of the LIK scores of its selected genes. For example, the average LIK score of the top 5 genes from both gene sets in the NB classifiers trained on the colon data set (12.7) was 82 fold lower than that of the classifiers trained on the leukemia data set (1039), which had the highest observed accuracy. These results suggest that the average LIK value of selected genes could be used to determine whether the NB classifier should be used in preference to the G-S method on a particular data set.

We use a novel Likelihood gene selection scheme based upon relative log likelihood terms with respect to pairs of class models. The Likelihood selection favors genes that increase the expected accuracy of the NB classifier, including genes avoided by G-S selection for having large class value standard deviations. Table 5 compares the relative class standard deviations for Likelihood and G-S selected genes from the two-class data sets. It is evident that Likelihood selection includes genes with much greater discrepancy between class standard deviations relative to G-S selection.

The Likelihood selection offers a simple means to limit the number of genes in the NB classifier and generalizes to data sets with more than two classes in a straightforward manner. We incorporated a minimum 3.0 LIK value requirement into our gene selection scheme and achieved good results. On all data sets, the resulting classifiers performed with the maximal or near-maximal accuracy observed for any classifier tested with a pre-designated size. We also demonstrated excellent results when the NB classifier was applied to a three-way classification of the leukemia data. In general, the method requires evaluating a number of LIK scores that increases as the square of the number of classes, thus limiting its usefulness to a moderate number of classes. In practice, however, one does not envision the need to distinguish many more than a few tissue classes simultaneously. In many cases involving large numbers of classes, the data has a hierarchical structure enabling one to perform sequential classifications with fewer classes, classifying the 'parent' classes prior to their 'child' classes.

5. References

[1] Alizadeh, A., Eisen, M.B., Davis, R.E., Ma, C., Rosenwald, A., Sherlock, G., Boldrick, J.C., Sabet, H., Tran, T., Yu, X., Powell, J.I., Yang, L., Marti, G.E., Moore, T., Hudson, J., Chan, W.C., Greiner, T., Weisenberger, D., Tibshirani, R., Armitage, J.O., Lossos, I., Levy, R., Wilson, W., Grever, M., Byrd, J., Botstein, D., Brown, P.O. and Staudt, L. (2000). Identification of clinically distinct types of diffuse large B-cell lymphoma based on gene expression patterns. *Nature*, **403**, 503-511.

[2] Alon, U., Barkai, N., Notterman, D.A., Gish, K., Mack, S.Y.D., and Levine, J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor colon tissues probed by oligonucleotide arrays. *PNAS*, **96**, 6745-6750.

[3] Ben-Dor, A., Bruhn, L., Friedman, N., Nachman, I., Schummer, M., and Yakhini, Z. (2000). Tissue Classification with Gene Expression Profiles. In*: Proceedings of the 4th Annual International Conference on Computational Molecular Biology (RECOMB),* Tokyo, Japan: Universal Academy Press.

[4] Brown, M.P.S., Grundy, W., Lin, D., Cristianini, N., Sugnet, C., Furey, T.M., Ares, J., and Haussler, D. (2000). Knowledge-based analysis of microarray gene expression data by using support vector machines. *PNAS*, **97**, 262-267.

[5] Cortes, C. and Vapnik, V. (1995). Support vector machines. *Machine Learning*, **20**, 273-297.

[6] Domingos, P. and Pazzani, M. (1996). Beyond independence: Conditions for the optimality of the simple Bayesian classifier. In: *Proceedings of the Thirteenth International Conference on Machine Learning*, pp. 105-112. Bari, Italy. Morgan Kaufmann.

[7] Dougherty, J., Kohavi, R. and Sahami, M. (1995). Supervised and unsupervised discretization of continuous features. In: *Machine Learning: Proceedings of the Twelfth International Conference*, Morgan Kaufmann.

[8] Duda, R.O. and Hart, P.E. (1973). *Pattern Classification and Scene Analysis.* New York: John Wiley and Sons.

[9] Eisen, M., Spellman, P., Brown, P., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *PNAS*, **95**, 14863-14868.

[10] Freund, Y. and Schapire, R.E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *J. Computer and System Sciences*, **55**, 119-139.

[11] Friedman, N., Linial, M., Nachman, I., and Pe'er, D. (2000). Using Bayesian Networks to analyze expression data. In*: Proceedings of the 4th Annual International Conference on Computational Molecular Biology (RECOMB)*, pp. 127-135. Tokyo, Japan: Universal Academy Press.

[12] Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., Bloomfield, C., and Lander, E. (1999). Molecular Classification of Cancer: Class discovery and class prediction by gene expression monitoring. *Science*, **286**, 531-537.

[13] Goodlett DR, Bruce JE, Anderson GA, Rist B, Pasa-Tolic L, Fiehn O, Smith RD, Aebersold R. (2000). Protein identification with a single accurate mass of a cysteine-containing peptide and constrained database searching. *Anal Chem.*, **72**, 1112-1118.

[14] Huang, L-R. and Hsu, H.C. (1995). Cloning and Expression of *CD24* Gene in Human Hepatocellular Carcinoma: A Potential Early Tumor Marker Gene Correlates with *p53* Mutation and Tumor Differentiation. *Cancer Research.* **55**, 4717-4721.

[15] John, G.H. and Langley, P. (1995). Estimating continuous distributions in Bayesian classifiers. In: *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence.* San Mateo, CA. Morgan Kaufmann.

[16] Loeb, L. (1998). Cancer cells exhibit a mutator pheontype. *Adv.Cancer Res.*, **72**, 25-56.

[17] Ma, L., Gauville, C., Berthois, Y., Millot, G., Johnson, G.R. and Calvo, F. (1999). Antisense expression for amphiregulin suppresses tumorigenicity of a transformed human breast epithelial cell line. *Oncogene*, **18**, 6513-6520.

[18] Schena, M., Shalon D., Davis, R. and Brown P.O. (1995). Quantitative monitoring of gene expression patterns with a cDNA microarray. *Science*, **270**, 467-470.

[19] Schummer, M. (2000). Manuscript in preparation.

[20] Schummer, M., Ng, WL., Bumgarner, R., Nelson, P., Schummer, B., Hassell, L., Baldwin. R., Kerlan, B., and Hood, L. (1999). Comparative hybridization of an array of 21,5000 ovarian cDNAs for the discovery of genes over-expressed in ovarian carcinomas. *Gene*, **238**, 375-385.

[21] Slonim, D.K., Tamayo, P., Mesirov, J.P., Golub, T.R. and Lander, E.S. (2000). Class Prediction and Discovery Using Gene Expression Data. *Proceedings of the 4th Annual International Conference on Computational Molecular Biology (RECOMB)*, pp. 263-272. Tokyo, Japan: Universal Academy Press.

[22] Stephens, M.A. (1970). Use of the Kolmogorov-Smirnov, Cramer-Von Mises and Related Statistics Without Extensive Tables. *J. Royal Stat. Soc. ser B*, **32**, 115-122.

[23] Virgilio, L., Narducci, M.G., Isobe, M., Billips, L.G., Cooper, M.D., Croce, C.M., and Giandomenico, R. (1994). Identification of the *TCL1* gene involved in T-cell malignancies. *PNAS*, **91**, 12530-12534.

[24] Winkelstein, A., Sacher, R.A., Kaplan, S.S. and Roberts, G. (1998). *White Cell Manual, Edition 5*. Philadelphia, F.A. Davis Company.