Classification of Tissue Samples Using Mixture Modeling of Microarray Gene Expression Data

Shili Lin and Roxana Alexandridis

Abstract

Accurate classification of tissue samples is an essential tool in disease diagnosis and treatment. The DNA microarray technology enables disease classification based only on gene expression analysis, without prior biological insights. We present a classification method based on modeling the distribution of the gene expression profile of a test sample as a mixture of distributions, each of which characterizes the levels of gene expression within a class. Class assignment for a test sample is based on the predictive probabilities of class memberships. We believe that this general modeling framework is a flexible scheme for multi-type classification. Since most of the thousands of genes whose expression levels are measured do not contribute to the separation between types of tissue samples, we also explore several measures for gene selection, including T, NPT, BW, NPBW, and a mixture modeling approach based on Markov chain Monte Carlo (MCMC) estimation of parameters. For a classifier based on a gene selection measure, such as the T classifier, the number of genes selected is achieved by cross-validation. The methods are applied to a leukemia dataset; our results are comparable with the best results achieved in a comparative study done by Professor Terry Speed and colleagues.

Keywords: microarray; gene expression; classifier; mixture; EM; MCMC

1 Introduction

DNA microarrays are biotech chips that enable researchers to measure the expression levels of thousands of genes simultaneously; see Schena [15] and The Chipping Forecast [5]. These measurements are obtained by quantifying the hybridization of the mRNA extracted from tissue samples to an array of spotted cDNA (cDNA arrays) or oligonucleotide probes (oligonucleotide arrays) immobilized on the surface of the chip. Details can be found in Schena *et al.* [16] for cDNA arrays and Lockhart *et al.* [9] for oligonucleotide arrays.

After proper image analysis, data processing and normalization (which entails nontrivial efforts, see for example, Dudoit *et al.* [4], Schadt *et al.* [14], Newton *et al.* [11], and Yang *et al.* [17]), a single number, referred to as the level of expression, is obtained for each gene on a microarray.

Statistical methods are needed to address many of the questions for which researchers seek answers from microarray gene expression data, such as (1) identifying genes differentially expressed under two or more conditions, (2) grouping genes with similar expression patterns, (3) finding genes that differentiate one tissue from another, and (4) molecular classification of tissue samples, including class discovery and class prediction. We focus on statistical methods for addressing this last issue.

Accurate classification of tissue samples is an essential tool in disease diagnosis and treatment. DNA microarray technologies enable classification based only on gene expression analysis, without requiring prior biological insight; successful cancer classification by Golub *et al.* [6] provides an excellent example. The idea is to classify a tissue sample into one of K known classes/types, where a sample, also called gene expression profile, is a vector whose components are the levels of gene expressions in a given tissue. Therefore, the problem of classification can be defined as follows: given a set of training samples, *i.e.*, samples whose class memberships are known, and a set of test samples, predict the class assignments of the test samples.

Most of the thousands of genes that make up the gene expression profile of a tissue sample do not contribute to the distinction between classes. Considering such irrelevant genes introduces noise to the classification process, and increases computational hurdles due to the extremely large dimensionality of the data. The combined contribution of many nonsignificant genes could downplay or even cancel the effects of the significant ones [8]. In addition, with a large number of genes whose expression levels are used for classification purposes, the interpretability of the results becomes an issue. When only a few genes are found helpful for separating classes, insight might be gained into the biological significance of these genes, as shown in Golub *et al.* [6].

For binary classification problems, Ben-Dor *et al.* [2] suggest a gene selection algorithm with a single threshold value chosen by cross-validation. Golub *et al.* [6] select the genes that provide best distinction between the "standardized" means of two classes (although their standardization is not the typical kind of standardization in statistics). Dudoit *et al.* [3] propose to select genes that display the largest ratios of between-group to within-group sums of squares, which is applicable to gene selection for multi-type classification problems.

Numerous methods have been proposed to classify tissue samples based on gene expression data. Some are restricted to binary classification, such as the weighted voting scheme of Golub *et al.* [6], while others are applicable to multi-type classifications. Techniques of machine learning, such as nearest neighbor classifiers [3], and cluster analysis methods, including hierarchical clustering [1, 12], have been entertained. Classification trees or aggregation of classifiers built from perturbed versions of the training set using boosting, bagging or convex pseudo-data methods of perturbing the training set [3], are some other examples.

There are yet other classification techniques that are applicable to multi-type classification problems; these are based on modeling the class densities, such as the linear and quadratic discriminant analysis of Dudoit *et al.* [3], or the naive Bayes methods of

Keller *et al.* [8]. For a comprehensive review of the methods, see Keller *et al.* [8] and Dudoit *et al.* [3].

In this article, we propose a classification method based on modeling the gene expression profile of a test sample as arising from a mixture of distributions, each of which characterizes the expression profiles within a class. We believe that this general modeling framework is a flexible scheme for multi-type classification. It could also be extended to accommodate class discovery in addition to classification to known classes. We also explore several measures for gene selection, including a mixture modeling approach based on Markov chain Monte Carlo (MCMC) estimation of parameters.

2 A Multi-type Classification Method

Mixture modeling of test samples

Let K denote the number of known classes (or sub-types, *e.g.*, leukemia sub-types) for which training samples exist. We use $Y_{ki} = (Y_{ki1}, \cdots, Y_{kiG})'$ to denote the column vector of gene expressions of the ith sample from class k, where G is the number of genes. Hence $\{Y_{ki}, i = 1, \cdots, T_k\}$ is the collection of data from class k, where T_k is the sample size, $k = 1, \cdots, K$. For each $i = 1, \cdots, T_k$, we assume $Y_{ki} \sim f_k(. \mid \theta_k)$, where θ_k is the vector of parameters of the component density function, which can be estimated, for example, from the training samples.

Let $\{X_i = (X_{i1}, \cdots, X_{iG})', i = 1, \cdots, T\}$ denote the gene expression data from T test samples, whose class membership assignments are unknown and the subject of interest. We model X_i as i.i.d. observations from a mixture distribution with component density functions f_k but unknown component weights π_k, $k = 1, \cdots, K$, $\sum_{k=1}^{K} \pi_k = 1$. That is,

$$f(X_i \mid \theta) = \sum_{k=1}^{K} \pi_k f_k(X_i \mid \theta_k), i = 1, \cdots, T,$$

where θ is the vector of unknown parameters including the π_k.

Two likelihood formulations are considered. If we assume that the parameters of each component density are to be estimated from the corresponding training samples, then the likelihood formulation is based on known component densities. The parameter vector is thus $\theta = \{\pi_k, k = 1, \cdots, K\}$, with the constraint $\sum_{k=1}^{K} \pi_k = 1$, and will be estimated using the test samples only. The likelihood function is

$$L_1(\theta) = \prod_{i=1}^{T} \left[\sum_{k=1}^{K} \pi_k f_k(X_i \mid \theta_k) \right]. \tag{1}$$

Alternatively, we can estimate the parameters θ_k in each component density together with the component weight π_k using data from training samples and test samples jointly. The parameter vector is thus $\theta = \{\pi_k, \theta_k, k = 1, \cdots, K\}$, again with the constraint

$\sum_{k=1}^{K} \pi_k = 1$. The likelihood function is then

$$L_2(\theta) = \prod_{k=1}^{K} \prod_{i=1}^{T_k} f_k(Y_{ki} \mid \theta_k) \prod_{i=1}^{T} \left[\sum_{k=1}^{K} \pi_k f_k(X_i \mid \theta_k) \right]. \tag{2}$$

Under the latter formulation, data from the test samples also contribute to the estimation of the parameters in each component density. In the next section, we focus on estimation of the parameters under this formulation. Parameter estimations under formulation (1) can be carried out similarly.

EM estimation of parameters

We assume that each component density is multivariate normal with mean vector $\mu_k = (\mu_{k1}, \cdots, \mu_{kG})$ and variance-covariance matrix Σ_k, that is, $\theta_k = \{\mu_k, \Sigma_k\}$. We further assume that the expression levels among different genes are independent, therefore $\Sigma_k = \text{diag}(\sigma_{k1}^2, \cdots, \sigma_{kG}^2)$ is a diagonal matrix of the variances. To find the maximum likelihood estimates (MLEs) of the parameters in (2) with normal component densities, the EM algorithm is highly suited [10].

Let Z_i, which takes a value from the set $\{1, 2, \cdots, K\}$, denote the unobserved class assignment for test sample $i = 1, \cdots, T$. Then $\{(X_i, Z_i), i = 1, \cdots, T\} \cup \{(Y_{ki}, k), k = 1, \cdots, K, i = 1, \cdots, T_k\}$ can be regarded as a representation of the complete data. The corresponding complete data likelihood is

$$L_c(\theta) = \prod_{k=1}^{K} \prod_{i=1}^{T_k} f_k(Y_{ki} \mid \theta_k) \prod_{i=1}^{T} \prod_{k=1}^{K} [\pi_k f_k(X_i \mid \theta_k)]^{I(Z_i = k)},$$

where $I(Z_i = k)$ is the indicator function that takes the value 1 if $Z_i = k$ and 0 otherwise. The EM iterates for the parameters are easily obtained and are given by:

$$\pi_k^{(t+1)} = \sum_{i=1}^{T} E[I(Z_i = k) \mid X_i, \theta^{(t)}]/T, \tag{3}$$

$$\mu_k^{(t+1)} = \frac{\sum_{i=1}^{T_k} Y_{ki} + \sum_{i=1}^{T} E[I(Z_i = k) \mid X_i, \theta^{(t)}] X_i}{T_k + \sum_{i=1}^{T} E[I(Z_i = k) \mid X_i, \theta^{(t)}]}, \tag{4}$$

$$(\sigma^2)_{kg}^{(t+1)} = \frac{\sum_{i=1}^{T_k} (Y_{kig} - \mu_{kg}^{(t)})^2 + \sum_{i=1}^{T} E[I(Z_i = k) \mid X_i, \theta^{(t)}](X_{ig} - \mu_{kg}^{(t)})^2}{T_k + \sum_{i=1}^{T} E[I(Z_i = k) \mid X_i, \theta^{(t)}]}, \tag{5}$$

$$k = 1, \cdots, K, g = 1, \cdots, G,$$

where

$$E[I(Z_i = k) \mid X_i, \theta^{(t)}] = \frac{\pi_k^{(t)} f_k(X_i \mid \theta_k^{(t)})}{\sum_{k=1}^{K} \pi_k^{(t)} f_k(X_i \mid \theta_k^{(t)})}.$$

From a starting parameter configuration $\theta^{(0)}$, we compute the sequence of estimates $\theta^{(t)}$ iteratively using equations (3)-(5) until convergence. The resulting parameter configuration is the estimated MLEs and is denoted by $\hat{\theta}$. Note that the estimates of the component density parameters involve data from the test samples as well as those from the training samples, as pointed out earlier.

A Classification Scheme

For each sample to be classified, we compute the predictive probabilities that it belongs to each of the known classes given the observed expression data and the parameter estimates. Then the sample is assigned to the class that has the largest predictive probability. That is, we compute

$$P(Z_i = k \mid X_i, \hat{\theta}) \propto \hat{\pi}_k f_k(X_i \mid \hat{\theta}_k), k = 1, \cdots, K. \tag{6}$$

Then

$$\hat{Z}_i = \text{argmax}_k \{P(Z_i = k \mid X_i, \hat{\theta}), k = 1, \cdots, K\}.$$

For a test set with T samples with known underlying class assignments $z_i, i = 1, \cdots, T$, the quantities $r = \sum_{i=1}^{T} I(\hat{Z}_i = z_i)/T$ and $e = \sum_{i=1}^{T} I(\hat{Z}_i \neq z_i)$ give the prediction accuracy rate and the number of samples that are misclassified, respectively.

3 Methods for Building Classifier

Gene selection measures and cross-validation

Gene selection measures are summary statistics used to order or select genes according to the perceived importance in discriminating among known classes. Four measures are described below and their performances are evaluated. Other gene selection measures are also considered; see the Discussion section for details.

T: This measure is applicable to two-class discriminant problems only. The measure T_g is simply the two sample t-statistic for each gene $g = 1, \cdots, G$. That is,

$$T_g = \frac{\bar{Y}_{1.g} - \bar{Y}_{2.g}}{\sqrt{S^2_{1.g}/T_1 + S^2_{2.g}/T_2}},$$

where $\bar{Y}_{1.g} = \sum_{i=1}^{T_1} Y_{1ig}/T_1$ and $S^2_{1.g} = \sum_{i=1}^{T_1} (Y_{1ig} - \bar{Y}_{1.g})^2/(T_1 - 1)$ are the sample mean and sample variance of class 1, respectively, and similarly for $\bar{Y}_{2.g}$ and $S^2_{2.g}$. Then the N genes with the largest absolute T_g values are selected to form the T classifier of size N. We discuss how to select N through cross-validation below.

NPT: This is the non-parametric counterpart of T, and thus is also applicable only to two-class problems. Let $R_g =$ rank $\{Y_{1ig}, i = 1, \cdots, T_1, Y_{2ig}, i = 1, \cdots, T_2\}$ denote the vector of the ranks of all the samples, among both classes, of gene g. The *NPT* measure is defined as the difference of the average rank of the samples in class one ($\bar{R}_{1.g}$) and that in class two ($\bar{R}_{2.g}$), that is, $NPT_g = \bar{R}_{1.g} - \bar{R}_{2.g}, g = 1, \cdots, G$. The N genes with the

largest absolute NPT_g values are selected to form the NPT classifier of size N. This classifier is more robust than T to outlying expression levels.

BW: This is a classifier based on the ratio of between-class sum of squares to within-class sum of squares, as proposed by Dudoit *et al.* [3]. This classifier is applicable to multi-type classification problems. The BW classifier is equivalent to the T classifier for two-class problems when the sample sizes in the two classes are equal, and hence, it may be viewed as a generalization of the T classifier. Specifically, define

$$BW_g = \frac{\sum_{k=1}^{K} T_k(\bar{Y}_{k.g} - \bar{Y}_{..g})^2}{\sum_{k=1}^{K}\sum_{i=1}^{T_k}(Y_{kig} - \bar{Y}_{k.g})^2},$$

where $\bar{Y}_{k.g}$ is the sample mean of class k, and $\bar{Y}_{..g}$ is the overall mean of all samples across all classes. Then the N genes with the largest BW_g values are selected to form the BW classifier of size N.

NPBW: This is the non-parametric counterpart of the BW classifier, which is also applicable to multi-type problems. The gene selection measure $NPBW_g$ is similarly defined as in BW_g but with the individual expression levels or the means replaced by their corresponding ranks (across all samples) and the corresponding average ranks. Like NPT, this classifier is robust to outlying expression levels.

For each type of classifier, after the genes are ordered according to their relative importance in discriminating among known classes, the number of genes N to use for classifying new samples must be selected. This task is accomplished by Leave-One-Out Cross-Validation (LOOCV). For each competing classifier, we estimate the parameters of the mixture model using data from the training samples, but leaving one out as a test sample. Since the true class assignment of the test sample is known, we can score whether correct assignment is made. After cycling through all the training samples one at a time, the prediction accuracy rate may be computed. A classifier with high prediction accuracy rate from LOOCV will be used as a candidate for classification of new samples.

An MCMC classifier

An alternative approach to gene selection for a two-class classifier is through mixture modeling and MCMC estimation and model selection. Suppose $(\bar{Y}_{k.g}, S^2_{k.g})$ are the sample mean and variance of gene g in class $k = 1,2; g = 1,\cdots,G$. If the sample size T_k is reasonably large, then $\bar{Y}_{k.g} \sim N(\mu_{kg}, S^2_{k.g}/T_k)$ approximately. Hence,

$$Y_g = \frac{\bar{Y}_{1.g} - \bar{Y}_{2.g}}{\sqrt{S^2_{1.g}/T_1 + S^2_{2.g}/T_2}}$$

follows a normal distribution with mean $\mu_{1g} - \mu_{2g}$, approximately. For a gene that is not differentially expressed in the two classes, $\mu_{1g} - \mu_{2g} = 0$. Thus, one may model Y_g as from a mixture of (univariate) normal distributions $(N(\mu_\lambda, \sigma^2_\lambda), \lambda = 1,\cdots,\Lambda)$ with an unknown number ($\Lambda \geq 1$) of components, with one of the components having mean zero

(referred to as the null component), representing those genes that are not differentially expressed.

The MCMC reversible jump method of Green [7] and Richardson and Green [13] is used to estimate the parameters of the model, including the number of components of the mixture. Then for each gene g, we compute the predictive probability that it belongs to each component of the mixture given Y_g and the estimated model, using a formula similar to (6). The gene is assigned to a component other than the null component if the predictive probability for that component is the largest and also larger than the weight of the null component. The collection of genes assigned to components other than the null component forms the MCMC classifier.

Following Richardson and Green [13], weak informative priors, chosen for computational convenience, are used for the model parameters. The priors for the means and variances of the component densities are assumed to be independent normals ($\mu_\lambda \sim N(\xi, \kappa^2)$) and inverse gammas ($\sigma_\lambda^2 \sim IG(\alpha, \beta)$), respectively. The hyperparameters ξ and κ are chosen to be the midpoint and half of the range (R) of the data interval, respectively, to make the prior for μ_λ to be rather flat. For σ_λ^2, we let $\alpha = 2$ and allow β to further follow a gamma distribution $G(l, h)$ with $l = 0.2$, and $h = 10/R^2$, to make σ_λ^2 similar but without being informative in their absolute size. The prior for the number of components (Λ) is assumed to be uniform between 1 and the pre-specified maximum number of components, taken to be 10 in our application. For the component weights, the prior is taken to be Dirichlet $D(1, 1, \cdots, 1)$. Further details can be found in Richardson and Green [13].

4 Leukemia dataset

The Leukemia dataset of Golub *et al.* [6] is the result of monitoring the expression levels of 7129 genes in two types of acute leukemia using Affymetrix high-density oligonucleotide array technology. The dataset consists of a training set which contains 27 samples of acute lymphoblastic leukemia (ALL), and 11 samples of acute myeloblastic leukemia (AML), and a test set comprising 20 ALL and 14 AML samples. The ALL samples could be further classified as ALL-B or ALL-T, depending on whether they arise from a B or T cell lineage. The 27 ALL training samples contain 19 ALL-B, and 8 ALL-T samples, while the 20 ALL test samples contain 19 ALL-B and one ALL-T sample. We refer to the problem of discriminating between ALL and AML as the two-class problem. Discriminating among ALL-B, ALL-T, and AML is referred to as the three-class problem.

5 Results

Checking the normality assumptions

In our mixture modeling of test samples, we assume that each component density

of the mixture follows a normal distribution. Therefore, we first checked whether this assumption is reasonable for the leukemia dataset. We assigned the test samples to the appropriate classes, because the true underlying class assignments for these samples are in fact known. Then, for the samples in each class, we computed the standardize expression levels (by subtracting the sample mean and dividing by the sample standard deviation within each class) for each gene, and they are plotted against normal scores. For the three-class problem, the results are shown in Figure 1. There are some obvious departures from normality, although they do not seem to be sufficiently bad to cast serious doubt on the validity of the assumption. The normality plots for the two-class problem are similar.

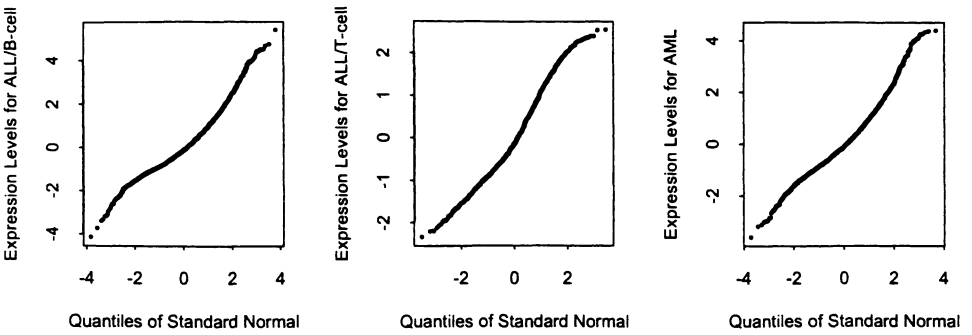

Figure 1: Normal probability plots for the samples in three classes, ALL-B, ALL-T, and AML.

Predictions for the two-class problem

For each of the four types of classifiers (*T*, *NPT*, *BW*, and *NPBW*) and a range of classifier sizes (1–200 genes), predictions of class assignments were carried out both for the training samples (through LOOCV) and the test samples using the mixture modeling approach. Figure 2(a) plots the prediction accuracy rates for the LOOCV of the training samples. The mixture modeling approach with the two types of non-parametric classifiers (*NPT* and *NPBW*) perform similarly; prediction accuracy rates of 1 are achieved in most of the range when the numbers of genes in the classifiers are more than 30. The mixture procedure does not perform as well with the parametric classifiers (*T* and *BW*), but the accuracy rates are still about 95% in most of the range. In summary, the results from LOOCV indicate that the prediction accuracy rates using the mixture modeling approach is not very sensitive to the type of classifier (among the four types that are considered here) nor the number of genes in a classifier, as long as the number is not very small.

Figure 2(b) plots the prediction accuracy rates for the test samples. Prediction accuracy rates of 1 are achieved only for the T classifiers with 19, 20, and 22 genes. Similar to the results in LOOCV, the performances of the procedures are not very sensitive to

the number of genes in the classifier as long as the number is not too small. Apart from *BW*, the results are not very sensitive to the types of classifiers in a wide range. The *BW* classifiers have not performed as well as the others.

A simulation study similar to that of Dudoit *et al.* [3] was carried out to further evaluate the performance of the mixture modeling procedure and to compare the four types of classifiers, including the effect of the size of a classifier. A total of 200 simulations (replications) were performed. For each simulation, 2/3 of the samples in each class (31 out of 47 ALL and 17 out of 25 AML) were randomly selected as the training samples, while the remaining served as the test samples. Each of the four types of classifiers with the sizes ranging from 1–200 was considered. The prediction results from the mixture procedures with all four types of classifiers are given in Figures 2(c) and 2(d). Specifically, the summary statistics for the number of test samples misclassified among the 200 replications for the T classifiers are plotted in Figure 2(c). For classifiers that are not very small, the results show that (1) there are no prediction errors in more than 25% of the replications, and (2) there are at most one prediction error in more than 75% of the replications. The performances for the three other types of classifiers are similar; full results are available from our web site (URL provided at the end of the article). The medians of the numbers of genes classified incorrectly (among 200 replications) for all four types of classifiers are plotted in Figure 2(d). We observe consistent results in all four classifier types for a wide range of classifier sizes.

We further examine the results from the simulation study by looking at each replication separately, instead of looking at the summary statistics, hoping to gain more insight into the relative performances of the four types of classifiers. Four pairs of classifiers are examined: T and NPT, BW and $NPBW$, T and BW, NPT and $NPBW$. One could examine other pairs, or higher number of classifiers jointly, but these four pairs seem the most appropriate ones to consider. For each pair and each replication, we classify the outcome into one of three categories: classifier 1 is better (the same, or worse) than classifier 2, depending on whether the number of samples incorrectly assigned under classifier 1 is smaller (the same, or larger) than that under classifier 2. The results are given in Table 1. Classifier T is slightly better than classifier NPT, while classifier $NPBW$ is slightly better than classifier BW, consistently for the three sizes of the classifiers examined. Furthermore, T seems to be better than BW, while their nonparametric counterparts perform almost exactly the same.

The results shown thus far are obtained under the mixture modeling formulation (2); that is, data in both the training samples and the test samples contribute to the estimation of mixture component density parameters as well as the mixing proportions. Results using formulation (1) are similar, especially for LOOCV as expected, although not quite as good in predicting the original test samples, which is not surprising either since there are almost as many test samples as there are training samples. The full results can be obtained from our web site.

MCMC classifier. A total of 100,000 iterations were performed. The first 50,000 iterations were discarded to allow for convergence; the remaining realizations were then

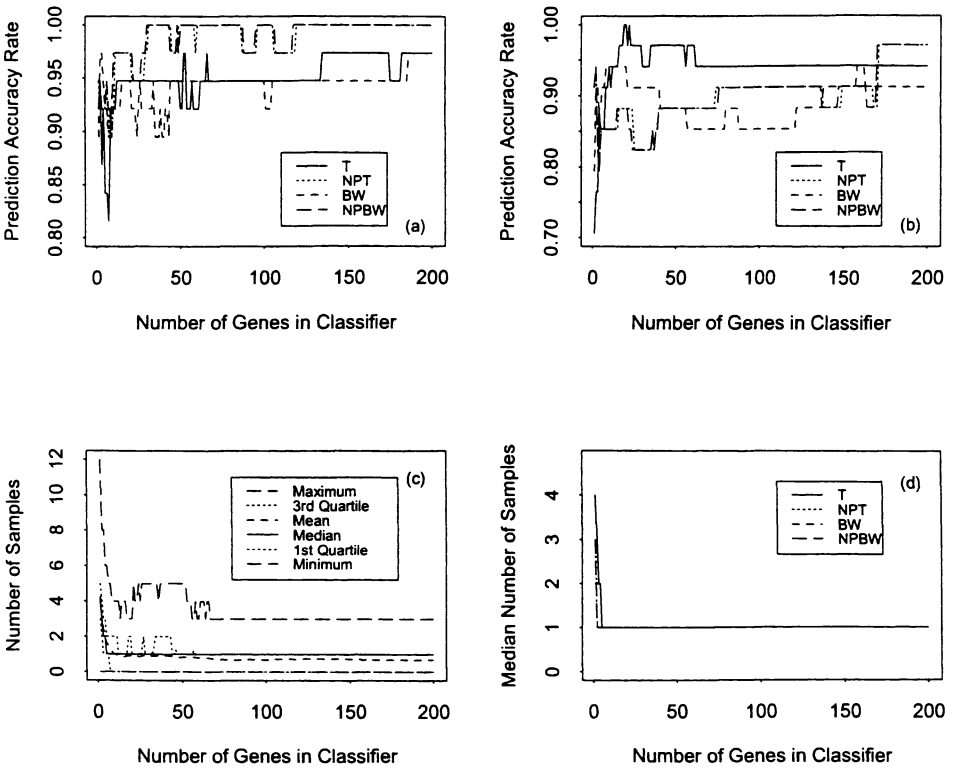

Figure 2: Prediction accuracy rates, or equivalently, the number of samples misclassified, for the training samples (through LOOCV) (a), the original test samples (b), and the simulated test samples (c and d), of the leukemia two-class problem. Figure 2(c) gives the summary statistics for classifiers based on T, and figure 2(d) plots the medians for T, NPT, BW, and $NPBW$.

used for inference. About 95% of the iterations picked three as the number of components for the mixture, with the second component corresponding to the distribution for genes that do not exhibit differential expressions (the null component), *i.e.*, with mean=0 for the component density. By applying our gene selection criterion, 23 genes were selected for the MCMC classifier. Class predictions for the test samples (using mixture formulation (2)) were then performed using the MCMC classifier. Out of the 34 samples in total, only one is classified incorrectly, giving a prediction accuracy rate

Table 1: Comparisons of classifiers for the two-class problem using the simulated data based on the leukemia dataset

#Genes	C1	C2	C1>C2[a]	C1=C2[b]	C1<C2[c]
25	*T*	*NPT*	59	96	45
	BW	*NPBW*	7	143	50
	T	*BW*	76	95	29
	NPT	*NPBW*	0	200	0
50	*T*	*NPT*	64	102	34
	BW	*NPBW*	15	161	24
	T	*BW*	72	99	29
	NPT	*NPBW*	4	196	0
100	*T*	*NPT*	58	100	32
	BW	*NPBW*	7	149	44
	T	*BW*	87	86	27
	NPT	*NPBW*	2	197	1

[a]This column gives the number of replications that result in smaller number of misclassified samples under classifier 1 than classifier 2.

[b]This column gives the number of replications that result in the same number of misclassified samples under both classifiers.

[c]This column gives the number of replications that result in larger number of misclassified samples under classifier 1 than classifier 2.

of 97%. On the other hand, under mixture formulation (1), in which only the training samples are used to estimate the component densities, five test samples are classified incorrectly.

Predictions for the three-class problem

Since *T* and *NPT* are applicable only to binary classification problems, they are not considered for further discriminating between ALL-B and ALL-T. For each multi-type feasible classifier (*BW* and *NPBW*) and a wide range of sizes (1–200 genes), the mixture modeling approach under formulation (2) were applied to classify the training samples (through LOOCV) as well as the test samples. Figure 3(a) and 3(b) plot the prediction accuracy rates for the LOOCV of the training samples and the test samples, respectively. Behavior similar to that observed in Figures 2(a) and 2(b) (for the two-class problem) is apparent in these figures. Namely, the prediction accuracy rates are not very sensitive to the type of classifier, nor the size of the classifier, and the class assignments of the samples are predicted quite accurately. *NPBW* performs better in smaller classifiers, especially in predicting the test samples, while *BW* does slightly better in LOOCV of the training samples. Overall, though, the performances of the two types of classifiers are similar.

A similar simulation study to that for the two-class problem was carried out. For

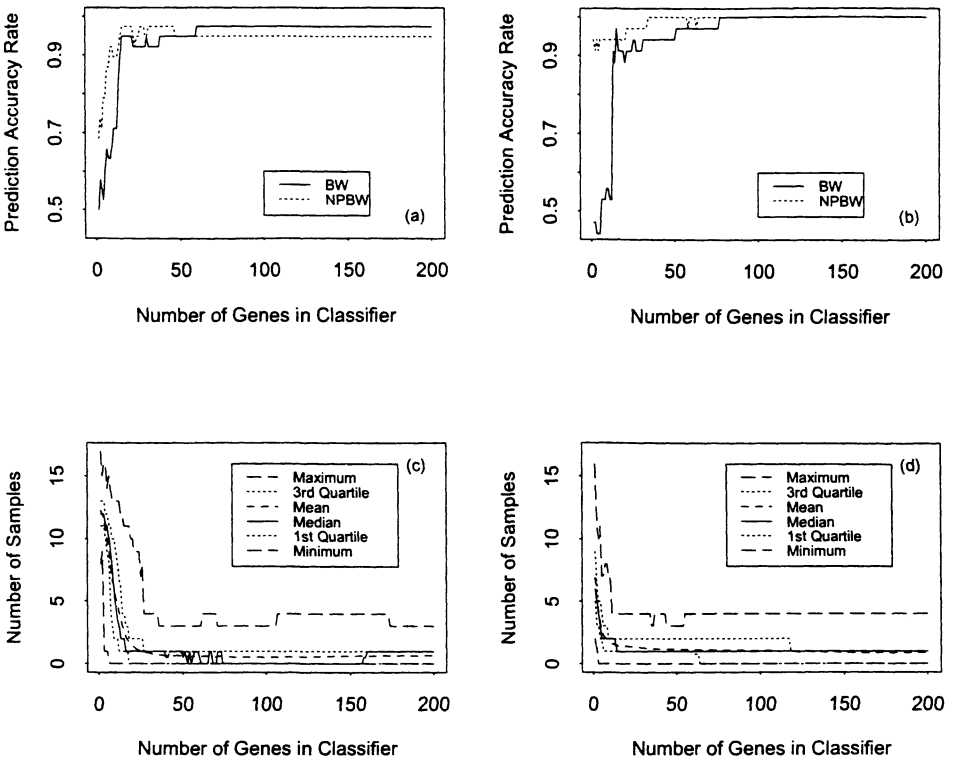

Figure 3: Prediction accuracy rates for the training samples (through LOOCV) (a), the original test samples (b), and the simulated test samples (c and d), of the leukemia three-class problem. Summary statistics for classifiers based on *BW* and *NPBW* are shown in (c) and (d), respectively.

each of the 200 replications, 2/3 of the samples in each class were randomly selected to form the training samples, and the remaining were assigned as test samples. For each classifier, the mixture approach under formulation (2) was applied to predict the class assignments of test samples. Summary statistics for the number of test samples classified incorrectly are plotted in Figure 3(c) for the *BW* classifiers, and 3(d) for the *NPBW* classifiers. Again, the mixture modeling approach yields good results for classifiers that are not too small, and *NPBW* performs slightly better for very small classifiers.

Table 2: Comparisons of classifier C1 (*BW*) and classifier C2 (*NPBW*) for the three-class problem using the simulated data based on the leukemia dataset

#Genes	C1>C2[a]	C1=C2[b]	C1<C2[c]
10	36	25	139
25	91	69	40
50	103	79	18
100	90	99	11
150	63	118	19
200	49	133	18

[a,b,c] See the footnotes of Table 1.

We further compare the performances of the two types of classifiers by examining each replication individually, in addition to the summary statistics across replications. For each replication, the outcome is classified into one of three categories: *BW* yielding smaller (same, larger) number of misclassified samples than *NPBW*. The results are shown in Table 2. We observe that, for smaller classifiers, there is a larger discrepancy between the two classifiers. Since *NPBW* is more robust to outlying expression levels, it is not surprising to see that it outperforms *BW* for the smallest classifier considered. As the number of genes in the classifiers increases, the two types of classifier become more similar, although *BW* continued to slightly outperform *NPBW* for larger classifiers.

The results shown thus far for the three-class problem are obtained using the mixture modeling formulation (2). Results using formulation (1) are similar, although not quite as good in predicting the original test samples, as what was observed for the two-class problem (full results available from our web site).

6 Discussion

In this article, we propose a method for classification of tissue samples by modeling the (multivariate) distribution of gene expression levels in a test sample as a mixture of distributions, each characterizing the distribution of the levels of gene expressions in a known class. This method can be paired with many gene selection methods (*i.e.*, methods for building classifiers) to reduce the dimensionality of the problem. Several classifiers are studied; results on *T*, *NPT*, *BW*, *NPBW*, and the MCMC classifier are presented in the current article, while results on several other binary classifiers can be found at our web site. Among the classifiers that are applicable to two-class problems, *T* performs well compared to the others in terms of prediction accuracy rates for the test samples (*T* achieving 100% accuracy rates for three classifier sizes) and the simulated samples in the leukemia dataset using the mixture modelling approach for classification. The MCMC classifier also performs well, with one prediction error out of a total of 34

test samples. Although the work on the MCMC classifier is still very preliminary, we are encouraged by these promising results, and effort is underway to extend it to handle multi-type classification problems. For multi-type feasible classifiers, *BW* is generally better than *NPBW* for predicting the training samples (through LOOCV) and the simulated samples, although *NPBW* is better in predicting the test samples, and *NPBW* classifiers were usually better than *BW* classifiers for smaller classifiers, again for the leukemia dataset. Note that the sizes of the classifiers that perform well are usually larger for the three-class problem than for the two-class problem, although they are all quite small (< 200) compared to the original number of genes. For predictions using the mixture modeling approach without first doing gene selection, three and four test samples are misclassified for the two-class and three-class problem, respectively, confirming the importance of gene selection.

Due to the lack of true test samples in the leukemia dataset, we were able to explore prediction accuracy rates for the test samples for a range of classifier sizes. In a real data analysis situation, however, we would proceed with the classification procedure proposed in this article in the following fashion. First, one would perform LOOCV with the training samples for a wide range of classifiers and sizes. Then a small set of classifiers that had performed well would be selected for classifying the test samples. We strongly recommend using more than one classifier so that consistency of prediction results can be checked. If several classifiers that had performed equally well in cross-validation had also produced consistent results in classifying the test samples, it would be an indication of satisfactory results, although there is no guaranteer that all assignments were correct. On the other hand, if discrepancies occur, then the biologists might be able to study the samples that caused the discrepancies more closely using other information.

Mixture modeling of test samples is a flexible means for multi-type classification of tissue samples. We have investigated two alternative formulations of the likelihood. It is not surprising to see that the one utilizing both the training samples and the test samples for parameter estimations (formula (2)) outperforms the one based on training samples only to estimate the parameters of the component densities, in many cases. Compared to other methods that have also been proposed for multi-type classifications, our approach performs at least as well with the leukemia dataset. For example, for predicting class membership of test samples, our approach yielded results with no prediction errors with medium-size classifiers (both *BW* and *NPBW*). The naive Bayes approach of Keller *et al.* [8] also yielded no misclassifications, for a small number of classifiers. Among the approaches discussed in Dudoit et al. [3], the best results had one misclassification of the simulated test samples, for both the median and the third-quartile, out of a total of 200 replications. Our simulation study using *BW* (for most of the classifiers ranging from 40 to 160 genes) resulted in zero and one misclassifications for the median and third-quartile, respectively, also out of 200 replications. Although one dataset and a limited simulation study do not warrant general conclusions, the results that we have obtained thus far show that the mixture modeling approach, coupled

with a gene selection measure such as *BW* (or its non-parametric counterpart if extreme observations are present), is promising. We plan to further evaluate its performance, especially its ability for classifying with larger numbers of classes.

The mixture formulation is also flexible in that it can be extended to handle situations where there are no training samples (class discovery problems) or when there are training samples but some of the test samples do not belong to any of the known classes (joint analysis of classification and class discovery). The key is to modify the mixture likelihood so that it allows for components that do not correspond to any known classes.

In our demonstration of the usage of the mixture modeling approach, each component density is assumed to be multivariate normal. This assumption was made for convenience. This assumption was also made in other methods, such as the methods based on maximum likelihood discriminant analysis [3]. Although good results were obtained from our analyses of the leukemia dataset, we could have used other distributions that fit the data better, as our figures show that there are obvious departures from normality. If the EM procedure for obtaining maximum likelihood estimates is no longer feasible, other methods for obtaining the MLEs may be used, including MCMC methods. Furthermore, we assume that the genes in a classifier are independent. Again, this assumption can be lifted, as the likelihood formulation is completely general; the component densities can be true multivariate distributions.

Electronic-Database Information

The URL for the supplementary material is: `http://www.stat.ohio-state.edu/~statgen/PAPERS/GeneExpression.html`.

Acknowledgments

Shili Lin would like to thank Professor Terry Speed for his mentoring and encouragement during her years at Berkeley and beyond. This work was supported in part by NSF grant DMS–9971770 (to S. Lin).

Shili Lin, Department of Statistics, Ohio State University, Columbus,
`shili@stat.ohio-state.edu`

Roxana Alexandridis, Department of Statistics, Ohio State University, Columbus,
`roxana@stat.ohio-state.edu`

References

[1] A. A. Alizadeh, M. B. Eisen, R. E. Davis, C. Ma, I. S. Lossos, A. Rosenwald, J. C. Boldrick, H. Sabet, T. Tran, X. Yu, J. I. Powell, L. Yang, G. E. Marti, T. Moore,

J. Hudson Jr, L. Lu, D. B. Lewis, R. Tibshirani, G. Sherlock, W. C. Chan, T. C. Greiner, D. D. Weisenburger, J. O. Armitage, R. Warnke, R. Levy, W. Wilson, M. R. Grever, J. C. Byrd, D. Botstein, P. O. Brown, and L. M. Staudt. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, 403:503–511, 2000.

[2] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and Z. Yakhini. Tissue Classification with Gene Expression Profiles. *Journal of Computational Biology*, 7:559–584, 2000.

[3] S. Dudoit, J. Fridlyand, and T. P. Speed. Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data. *Journal of the American Statistical Association*, 97:77–87, 2002.

[4] S. Dudoit, Y. H. Yang, M. J. Callow, and T. P. Speed. Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *Statistica Sinica*, 12:111–139, 2002.

[5] The Chipping Forecast. *Supplement to Nature Genetics*, 21:1–60, 1999.

[6] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. 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*, 286:531–537, 1999.

[7] P. J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82:711–732, 1995.

[8] A. D. Keller, M. Schummer, L. Hood, and W. L. Ruzzo. Bayesian Classification of DNA Array Expression Data. Technical report, UW-CSE-2000-08-01, Department of Computer Science and Engineering, University of Washington, 2000.

[9] D. J. Lockhart, H. L. Dong, M. C. Byrne, M. T. Follettie, M. V. Gallo, M. S. 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.

[10] G. J. McLachlan and T. Krishnan. *The EM algorithm and extensions*. John Wiley & Sons, Inc. New York, 1997.

[11] M. A. Newton, C. M. Kendziorski, C. S. Richmond, F. R. Blattner, and K. W. Tsui. On Differential Variability of Expression Ratios: Improving Statistical Inference about Gene Expression Changes from Microarray Data. *Journal of Computational Biology*, 8:37–52, 2001.

[12] D. A. Notterman, U. Alon, A. J. Sierk, and A. J. Levine. Transcriptional Gene Expression Profiles of Colorectal Adenoma, Adenocarcinoma, and Normal Tissue Examined by Oligonucleotide Arrays. *Cancer Research*, 61:3124–3130, 2001.

[13] S. Richardson and P. J. Green. On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society, Series B*, 59:731–758, 1997.

[14] E. E. Schadt, C. Li, C. Su, and W. H. Wong. Analyzing High-Density Oligonucleotide Gene Expression Array Data. *Journal of Cellular Biochemistry*, 80:192–202, 2000.

[15] M. Schena, editor. *DNA Microarrays: A Practical Approach*. Oxford University Press, 1999.

[16] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown. Quantitative Monitoring of Gene Expression Patterns with a Complementary DNA Microarray. *Science*, 270:467–470, 1995.

[17] Y. H. Yang, S. Dudoit, P. Luu, and T. P. Speed. Normalization for cDNA microarray data. In M. L. Bittner, Y. Chen, A. N. Dorsel, and E. R. Dougherty, editors, *Microarrays: Optical Technologies and Informatics, Proc. SPIE*, volume 4266, pages 141–152, 2001.