Gene Expression Comparisons for Class Prediction in Cancer Studies

Donald Geman[†§], Christian d'Avignon[†‡]
Daniel Q. Naiman[†§], Raimond L. Winslow[†‡], Arnaud Zeboulon

June 2004

[†] *Center for Cardiovascular Bioinformatics and Modeling,*
Whitaker Biomedical Engineering Institute
[§] *Department of Applied Mathematics and Statistics*
[‡] *Department of Biomedical Engineering*
The Johns Hopkins University, Baltimore, Maryland, USA

Abstract

We introduce a new method for analyzing gene expression microarray data which is based entirely on comparing the mRNA counts of selected pairs of genes. The results are invariant to normalization and, apart from gene pair selection, there are no parameters to estimate, thereby avoiding some liabilities of standard techniques, such as sensitivity to pre-processing and inflated estimates of performance. In addition, the decisions are highly transparent, even to nonspecialists in class prediction and statistical learning. On the other hand, due to a search over all *pairs* of genes, certain issues must be addressed, such as computational complexity and tests of significance.

We have applied the method to detecting disease, identifying tumors and predicting treatment response. Some results with breast, leukemia and prostate cancer data are summarized, including a comparison with a variety of other methods (PAM, k-NN, SVM and DLDA) in predicting relapse from breast cancer. In general, we achieve comparable or even higher prediction rates with far fewer genes.

1 Introduction

We introduce a new method for classifying gene expression profiles in which predictions are based entirely on *pairwise comparisons*. Consequently, we attempt to differentiate between two classes by finding pairs of genes whose expression levels typically invert from one class to the other. Our approach is a particular instance of a larger class of rank-based methods in which the expression levels are immediately replaced by their corresponding ranks (i.e., most heavily expressed, second most heavily expressed, etc.) determined across all genes assayed using a single DNA microarray. Such methods are robust to quantization effects and are *invariant* to pre-processing designed to overcome chip-to-chip variation, such as normalization methods [26], under the very mild assumption that normalization preserves ordering.

We will focus on the simplest example of comparison-based classification – the "top-scoring pair(s)" or TSP classifier. The participating pairs are those which achieve the largest "score" relative to a simple measure of discrimination and each of these pairs "votes" for the class which makes the observed ordering within that pair the most likely. *There are no parameters to tune.*

Clearly information is lost using a rank-based procedure. However, the results reported here, and in our technical report [12], demonstrate that the amount of information residing in the ordering of gene expression levels is more than sufficient to perform classification at least as well as other methods. Indeed, in many cases, including the prostate and two breast cancer studies featured here, accurate prediction can be achieved by comparing the expression levels of a single pair of genes.

One motivation for our work is the *small-sample dilemma* in the statistical analysis of microarray data, which was one of the principal themes of *Interface 2004* and is well-documented in the literature [8, 18, 19, 25]. When measured against the number of features, and the complexity of the biological systems under study, the amount of data available for modeling and inference is severely limited. Discovery of the underlying structure within these data, in particular correlation patterns or even higher-dimensional interactions, is exceedingly difficult in this small-sample regime. Moreover, there is already evidence [8] that relatively simple classification methods (as in [22]) are competitive with more complex ones, such as as neural networks [1, 2, 14], decision trees [4, 6, 28] and support vector machines [17, 27]. Furthermore, reported success rates for class prediction are likely to be inflated [8, 19] unless all aspects of learning a classifier, in particular all choices of parameters, are properly validated. In our case there are no parameters to cross-validate.

Another motivation is to avoid opaque decision-making. Standard methods in statistical learning typically result in predictions based on nonlinear functions of many expression values, and consequently highly complex decision boundaries between the classes of interest. Such boundaries are then difficult to summarize in simple terms or to characterize in a manner which is biologically meaningful. In contrast, the *TSP* classification rule is evidently transparent.

We demonstrate the efficacy of this method in two comparisons: First, we summarize results from [12] on several gene expression data sets involving breast, prostate and leukemia cancers, providing prediction rates from source material as well as for the *TSP* classifier. (Some rates reported in the cited references are not properly validated (e.g., with a test set or by full cross-validation) and are considered to be biased [19, 8].) Second, we consider a recent study about the prognosis for relapse from breast cancer and compare the *TSP* classifier with a variety of other popular classification methods using publicly available software. We believe these results demonstrate that our method is clearly more efficient in terms of the number of genes utilized while maintaining at least comparable accuracy.

2 Comparison-Based Classification

Consider G genes whose expression levels $\mathbf{X} = \{X_1, X_2, ..., X_G\}$ are measured using DNA microarrays and regarded as random variables. Each profile $\mathbf{X}$ has a true class label in $\{1, 2, ..., C\}$. For simplicity, we assume $C = 2$, although the results extend to higher numbers of classes. Let $R[i]$ denote the rank of X_i relative to $\mathbf{X}$, where $R[i] = 1$ means that X_i is the smallest value in $\mathbf{X}$, $R[i] = 2$ denotes the second smallest value, etc. Assume no ties for simplicity, that is, $R[i] < R[j]$ if and only if $X_i < X_j$. Note that ranking is *within profiles* for each fixed experiment, not across experiments for each fixed gene as in nonparametric methods for detecting differentially regulated genes (see, e.g., [18]). Consequently, $\mathbf{R} = R[1], ..., R[G]$ is a permutation of $\{1, 2, ..., G\}$.

Obviously $\mathbf{X}$ contains more information than $\mathbf{R}$ and the mutual information between Y and $\mathbf{X}$ will be greater than that between Y and $\mathbf{R}$. Nonetheless, there may be enough information in $\mathbf{R}$ to strongly reduce the uncertainty about the class.

2.1 Pair Scoring

We will exploit discriminating information in the ranks $R[1], ..., R[G]$ by focusing on detecting "marker gene pairs" (i, j) for which there is a significant difference in the probability of $X_i < X_j$ from class 1 to class 2. Profile classification is then based on this collection of distinguished pairs. Notice that knowing the result of all pairwise comparisons ($R[i] < R[j]$) is equivalent to knowing $\mathbf{R}$. Here, the statistics of interest are $p_{ij}(c) = P(X_i \leq X_j | c),\ \ c = 1, 2$, i.e., the probabilities of observing $X_i \leq X_j$ in each class. These probabilities are estimated by the relative frequencies of occurrences of $X_i \leq X_j$ *within profiles* and over experiments. Consequently, for our analysis it is sufficient to know the *ranks* of the expression values within profiles on each microarray.

Let $\Delta_{ij} = |p_{ij}(1) - p_{ij}(2)|$ denote the "score" of (i, j). An example of computing a score is provided in Table 1. We seek gene pairs with "large" scores.

2.2 Pair Selection

Detection of marker gene pairs is a problem in feature selection, and plays the same role in our analysis as finding individual marker genes does in more standard methods [9, 18, 22, 21]. One option for pair selection might be to *first* select differentially-regulated or "marker genes" and only then proceed from individual genes to gene pairs by restricting the search for marker pairs to pairs of these marker genes. But two major drawbacks would ensue: 1) such post-filtering results would no longer be invariant to normalization; and 2) by construction, only differentially expressed genes could appear in the selected comparisons, thereby possibly losing

	$X_i \leq X_j$	$X_i > X_j$	
class 1	8	36	44
class 2	30	4	34

Table 1: An example of scoring a gene pair from the breast cancer prognosis study. Expression levels for about $25,000$ genes are obtained for 44 profiles associated with class 1 ("good prognosis") and 34 associated with class 2 ("poor prognosis"); see §5. For a particular pair (i, j) of genes we have identified, the 78 profiles are labeled according to the above 2×2 contingency table. These data lead to the probability estimates $p_{ij}(1) = 8/44$ and $p_{ij}(2) = 30/34$, which results in the score $\Delta_{ij} = |\frac{8}{44} - \frac{30}{34}| = .7005$. Since $p_{ij}(1) < p_{ij}(2)$, the classifier based on this gene pair votes for class 1 for a profile with $X_i > X_j$ and for class 2 otherwise.

discriminating pairs in which at most one gene is itself differentially expressed. We therefore adopt a more straightforward method based on direct search: We estimate Δ_{ij} for every distinct pair (i, j) and apply a selection rule based on the magnitude of Δ_{ij}. An example of such a decision rule is to rank the scores Δ_{ij} from largest to smallest and select all pairs achieving the top score.

(An indirect approach to scoring pairs can be found in [3], where feature selection based on gene pairs is investigated in the context of profile classification using linear discriminant analysis and nearest-neighbors. Rank statistics are not considered and, in particular, the expression levels within a pair are not compared.)

2.3 Classification

Pair selection results in a family $\mathcal{P}$ of distinguished pairs. For the sample sizes in the data sets we have treated, which range from $n = 49$ to $n = 102$, there are only one to three pairs which achieve the top score.

Any standard classification algorithm may then be implemented using $\mathcal{P}$ as input. We are interested in algorithms for which classification decisions have a simple interpretation. Voting is an example of such a decision algorithm, where *individual* votes are driven by maximum likelihood. In this method, given a new expression profile $\mathbf{X}$, an individual pair (i, j) in $\mathcal{P}$ votes for the class for which the observed ordering between X_i and X_j is more likely; see the example in Table 1. That is, if we observe $X_i \leq X_j$ in a new profile, then pair (i, j) votes for class 1 if $p_{ij}(1) \geq p_{ij}(2)$ and votes for class 2 otherwise. The class with the most votes is chosen. We refer to the resulting classifier as the *top scoring pair(s)* classifier, henceforth denoted *TSP*.

It is noteworthy that for classification based on a single gene pair, the sum of misclassification probabilities over the two classes can be expressed as $1 - \Delta_{ij}$, which provides a natural justification for score maximization.

The procedure of tallying individual votes, while attractive from the point of view of simplicity [9], can also be derived as a maximum likelihood rule under the simplifying assumptions that (i) individual comparisons are conditionally independent given the class, and (ii) for some p we have either $p_{ij}(c) = p$ or $p_{ij}(c) = 1 - p$ for all $(i, j) \in \mathcal{P}$ and both classes $c = 1, 2$.

3 Estimation of Error and Tests of Significance

In validating our results, there are two paramount issues: unbiased estimates of the generalization error and tests of significance for both the top score and the estimated classification rate.

3.1 Estimation of Error

In estimating the (generalization) error rate of the TSP classifier, gene pair selection was performed *within the cross-validation loop*. With n samples and (leave-one-out) cross-validation (CV), this means choosing n separate subsets $\mathcal{P}$, one for each profile "held out" during training, then classifying that profile. (Other methods of estimating the error rate could, and perhaps should, be considered; see §6.) In particular, both the actual top-score, as well as the set of pairs which achieve it, may vary with the sample left out. The estimated prediction rate is then $1 - e/n$ where $e \in \{1, ..., n\}$ is the number of errors observed in the cross-validation.

For our procedure there are no parameters to select inside the CV loop. For other procedures that do require parameters, e.g., k-nearest neighbors, random forests and support vector machines, the estimated prediction rates may be severely biased if performance is sensitive to these parameters and they are not properly cross-validated (using an inner CV loop to choose parameter values) [8, 19, 25]. The TSP classifier avoids this source of bias.

3.2 Tests of Significance

Typically, the expression levels of thousands of genes are measured in a given study, and hence there is an enormous number of possible *pairs* of genes, even hundreds of millions (see e.g., the breast cancer prognosis study in §5). Naturally, the issue of *over-fitting* arises – finding high scores and (seemingly) discriminating pairs of genes due merely to chance. There are several ways to test for "significance." One is the cross-validation itself: When a sample is left out, if a high-scoring pair of genes is due entirely to chance, then it will correctly classify the sample left out with probability one-half. Consequently, high performance cannot be due to chance alone; see the discussion in [12].

In addition, a permutation analysis provides another important test of significance, both for the score and for the estimated rate. For any given study, artificial data sets can be constructed by randomly permuting the class labels, hence maintaining the sample sizes n_1 and n_2 of the two classes. The resulting top scores and cross-validated error rates are then indicative of those obtained when attempting to classify based on profile labels which cannot be predicted from the expression values while maintaining the overall statistical dependency structure among the genes.

In each of our experiments, the top score was computed for 1000 random assignments of the class labels to the n samples (preserving class sample sizes). Hence in each study a p-value can be associated with the top score on the actual data by taking the fraction of permuted data sets in which a score at least as large is obtained. This p-value can be interpreted as the probability of observing such a large score under the null hypothesis that the pairs are non-informative for classification. Similarly, the prediction rate of the TSP classifier itself can be estimated as in §3.1

Problem	Score	Genbank ID 1	t-stat 1	Genbank ID 2	t-stat 2
Breast Nodal	0.838	X03453	4.39	X82634	2.25
Prostate	0.902	M84526	7.46	M55914	4.13
Leukemia	0.979	L11373	1.99	X95735	10.92
Leukemia	0.979	D86976	1.60	X95735	10.92
Leukemia	0.979	J05243	7.87	M23197	6.62
Breast Prognosis	0.701	AW134553	2.55	NM_003963	1.70

Gene descriptions

X03453	Bacteriophage P1 cre gene for recombinase protein
X82634	Homo sapiens mRNA for hair keratin acidic 3-II
M84526	Human adipsin/complement factor D mRNA, complete cds
M55914	Homo sapiens c-myc binding protein (MBP-1) mRNA, complete cds
L11373	Human protocadherin 43 mRNA, complete cds for abbreviated PC43
X95735	Homo sapiens mRNA for zyxin
J05243	Human nonerythroid alpha-spectrin (SPTAN1) mRNA, complete cds
M23197	Human differentiation antigen (CD33) mRNA, complete cds
D86976	Human mRNA for KIAA0223 gene, partial cds
AW134553	MOB1, Mps One Binder kinase activator-like 2C
NM_003963	Homo sapiens transmembrane 4 superfamily member 5 (TM4SF5), mRNA

Table 2: The top scoring pair(s) for each study, together with the top score and individual t-statistics.

above for each random permutation and a p-value can therefore be associated with the rate on the correctly labeled data.

4 Experiments with Breast, Leukemia and Prostate Cancer Data

The TSP classifier was initially evaluated on three class prediction problems: *Predicting the status of lymph nodes in patients with breast tumors* (**Breast Nodal** study; [25]); *Classifying profiles into leukemia subtypes* (**Leukemia** study; [13]); *Distinguishing prostate tumors from normal profiles* (**Prostate** study; [20]). Details involving these data (references, chips, samples sizes, web addresses, etc.) can be found in [12] and in Table 3.

4.1 Top-Scoring Pairs

There are three top-scoring pairs for the **Leukemia** data and only one for the **Breast Nodal** and **Prostate** data; the actual top scores, and corresponding gene pairs, are identified in Table 2, together with their individual t-statistics. Some of these genes would not be regarded as "differentially regulated" on the basis of their individual t-statistics. Notice that the same gene may appear in more than one pair.

No score among the 1000 permutation trials came near the top score actually observed on either the **Leukemia** or **Prostate** data, and hence the estimated p-values are virtually zero. For the **Breast Nodal** data the estimated p-value of the top score is 0.001.

Problem	**G**	**n**	**TSP** (# genes)	**Previous Results** (# genes)
Breast Nodal	7129	49	79% (2)	41%-88% (10-4459)
Leukemia	7129	72	94% (5)	85%,95% (50)
Prostate	12600	102	95% (2)	86%-92% (4-256)

Table 3: Some comparisons of performance between the TSP classifier and previously reported prediction rates: G is the total number of probes; n is the sample size; and # genes is number of genes used by the classifier.

4.2 Prediction Results

The estimated (correct) prediction rate of the TSP classifier for the first three studies is displayed in Table 3 along with other reported results for these data. All TSP results are based on leave-one-out cross-validation. In predicting the status of lymph nodes (affected or non-affected) in the **Breast Nodal** study, the estimated classification rate of 79% corresponds to nine errors and three ties out of 49 cross-validation loops; random tie-breaking then results in 10.5 errors on average. Estimated error rates for these data based on leave-one-out cross-validation using a wide variety of common machine learning techniques are summarized in Chapter 3 of [8] for varying numbers of pre-filtered genes: m = 10, 50, 100, 200, 500, 1000, and $m = 7129$. Most parameter choices are external to the cross-validation in estimating the error rates listed in [8]; see the comprehensive discussion there. These external parameters include those which are method-specific as well as the choice of the number of genes that are pre-filtered. For example, in the case of support vector machines, there are 48 experiments presented in [8], corresponding to choosing the kernel, the penalty, the filtering method and the number genes to be filtered; the number of errors varies according to the protocol (e.g., $7 - 11$ errors with $m = 10$ genes, $12 - 18$ errors with $m = 50$ genes, etc.). All of these methods are more complex than the TSP classifier and relatively few parameter choices yield better results. Moreover, it is not clear that even these differences would remain after proper cross-validation of the other methods.

For the **Leukemia** study, the two stated rates, 85% and 95%, refer, respectively, to validation on the test set and leave-one-out cross-validation on the training set [13]. For the **Prostate** study [20] a k-nearest neighbor classifier was applied to m genes (for selected values of m from 1 to 256) identified by measuring differential expression between normal and tumor samples using a variation of the signal-to-noise statistic [13]. For each m, prediction error was estimated using leave-one-out cross-validation; the range $86\% - 92\%$ corresponds to $4 \leq m \leq 256$. The parameter choices are not cross-validated, and hence the best estimated prediction rates are biased upwards.

Additional information about each comparison can be found in [12], including a discussion of the biological purpose and significance of the genes in the top-scoring pairs. For example, in the **Prostate** study, one of the genes in the top scoring pair using the TSP classifier was adipsin, which was identified as one of the top 50 marker genes in Singh et al.[20] but the other gene in the pair, c-myc, was not. Nonetheless, the joint behavior of c-myc and adipsin is highly discriminative of non-tumor versus prostate tumor samples, yielding a prediction rate of 95% (with $p < 0.001$).

5 Experiments with Breast Cancer Prognosis Data

In [24], the authors attempt to utilize the expressions values of 24,481 genes and ESTs obtained from cDNA microarrays in order to predict the existence of future metastases among patients with primary breast tumors but who do not have tumor cells in local lymph nodes at the time of diagnosis. Among $n = 78$ patients in the study, there are $n_2 = 44$ patients who remained disease-free for at least five years ("good prognosis group") and $n_1 = 34$ patients who developed distant metastases within five years ("poor prognosis group"). The objective is to identify profile signatures which predict, at the time of diagnosis, to which class the tumor belongs.

5.1 Previous Results

The authors in [24] develop a "prognosis classifier" constructed by: i) gene screening to reduce the number of genes to around 5000; ii) selection of 231 ("reporter") genes among these 5000 based on computing a correlation coefficient of expression with disease outcome; iii) rank ordering these 231 correlations and successively adding small groups to a correlation-based classifier, measuring improvements by leave-one-out CV. Finally, classification is based on correlating the expression profile of the left-out sample based on seventy genes with those of the remaining samples. The authors report an 83% classification rate (i.e., 13 errors out of 78 samples). However, this rate was revised downward to 73% (21 errors) upon realizing that the initial estimate of 83% was not properly cross-validated; for instance, the set of reporters was not re-computed in each loop of the cross-validation (online supplement to [24]).

5.2 TSP Classifier

There are missing values for about half of the genes; for instance, one missing out of 78 for 8397 genes, two missing for 1826 genes, etc. (The treatment of missing data is not discussed in [24].) We decided to use only the 13,547 genes for which there are no missing values; another approach would have been to utilize all the data but normalize the scores to account for differing sample sizes. As indicated in Table 1, the top score on the whole training set is $\Delta = 0.7005$ and there is only one pair achieving this score (see the discussion below). Based on 1000 permutations of the class labels, the estimated p-value of this score is $p = 0.003$; see Figure 1.

5.2.1 Biological Context

The pair of genes with the highest score is indicated in Table 2. The protein encoded by gene TM4SF5 is a cell-surface glycoprotein mediating signal transduction events involved in the regulation of cell proliferation and motility. It may play a role in uncontrolled growth of tumor cells, and is known to be over-expressed in pancreatic as well as other cancers [16]. Probe AW134553 is a member of Unigene cluster Hs.97927 encoding the human the Mps one binder kinase activator-like 2C (MOBKL2C, MOB1) protein. In yeast, MOB1 is a protein kinase member of the mitotic exit network involved in spindle body duplication and mitotic checkpoint regulation via interactions with Dbf2 and Cdc15 [15].

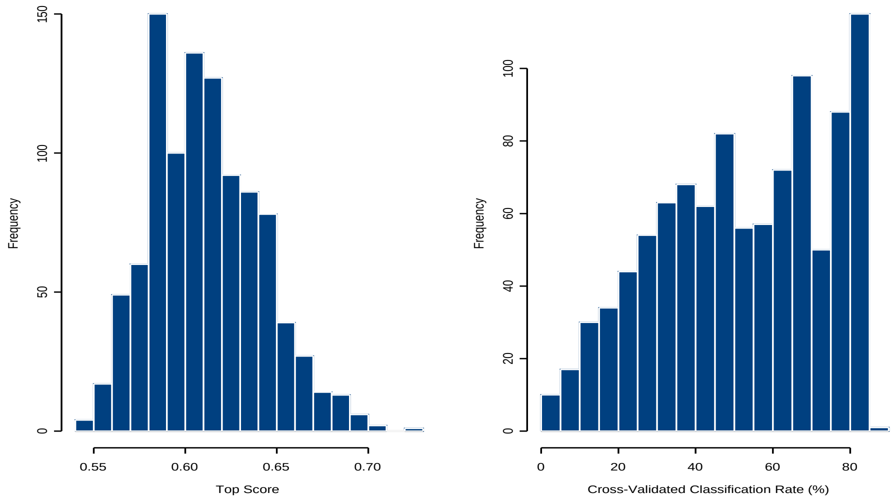

Figure 1: Histograms of top scores (left panel) and classification rates (right panel) for 1,000 randomly relabeled gene expression profiles from the breast cancer data. For the actual data, the top score obtained is .7005 and the classification rate is 84.0%.

5.2.2 Prediction Rate

As before, and as in [24], the prediction rate of the TSP classifier was estimated by leave-one-out CV; of course the top-scoring pairs will in general depend on the sample left out. The TSP classifier makes 12 errors and there is one tie (i.e., in one loop, half the top-scoring pairs vote for class 1 and half vote for class 2). Assuming a random vote in the case of ties results in a predicted classification rate of $65.5/78 = 84\%$, with $p = 0.012$ (the fraction of permutations in which the TSP classifier achieved a rate at least this high). In Figure 1 we also show the histogram of the classification rates (estimated by leave-one-out CV) for the 1000 permutations.

5.2.3 Computational Complexity

Computing the TSP classifier requires $O(G^2)$ operations since this requires computing a score for every gene pair. For the breast cancer data, this calculation took 67 CPU seconds using an itanium2 1.3 GHz processor. A naive approach to computation of the classification rate by leave-one-out CV would require n times the effort of computing the TSP. (Recall that n is the number of sample profiles.) However, this factor can be reduced to approximately two by using a preliminary calculation in which a list of gene pairs is identified that must contain any top scoring pair of genes in all of the CV iterations. This list typically represents a very small fraction of all gene pairs. (In fact, for the breast cancer data this list contains only three pairs.) As a consequence, the remaining computational effort in calculating the

Method	*1 gene*	*11 genes*	*49 genes*	*86 genes*	*184 genes*
DLDA	21	34	29	28	29
1-NN	28	37	27	25	26
3-NN	27	34	30	23	23
SVM	21	33	23	25	31

Table 4: The number of errors (out of 78) in leave-one-out CV with three common methods (see text) for varying numbers of genes corresponding to screening based on five thresholds (see text) on the t-statistics. The TSP classifier uses two genes and makes 12.5 errors.

TSP inside the CV loops is negligible. For the breast cancer data, leave-one-out CV took 132 CPU seconds using the above-mentioned processor.

5.3 Benchmarks

We benchmark our results and those in [24] against two well-known software packages for classification from microarray data:

- *BRB Array Tools* (Simon, R. and Lam, A., $http://linus.nci.nih.gov/BRB-ArrayTools.html$); and
- *Prediction Analysis of Microarrays* (*PAM*) [22]

Both packages perform gene selection internally to the CV; *BRB Array Tools* uses leave-one-out CV whereas *PAM* performs 10-fold CV in order to estimate the generalization error. In both cases missing data were treated the same way as for the TSP classifier. (Rank-based versions of all these methods could be considered, but only the standard versions were implemented.)

5.3.1 BRB Array Tools

BRB Array Tools offers a choice of several classification techniques. We chose three common methods: *k-Nearest Neighbors* (*k-NN*), *Diagonal Linear Discriminant Analysis* (*DLDA*) and *Support Vector Machines* (*SVM*). Possible values of k for *k-NN* are $k = 1$ and $k = 3$ and the distance metric is the Euclidean distance. The SVM kernel is linear and we kept the default value of 1 for the "cost" parameter. For each classification technique, gene selection is done via the t-test, whose significance threshold is set by the user. We chose five values for this threshold : 10^{-5}, 10^{-4}, 5×10^{-4}, 2.5×10^{-3}, 5×10^{-3}, yielding a selection of respectively $1, 11, 49, 86$ and 184 genes on the whole tumor set (the number of genes selected during each step of the CV being similar to those numbers). The results are shown in Table 4; the values in the table are the number of leave-one-out CV errors determined by *BRB Array Tools*.

Note that these figures are likely to be optimistic estimates of future performance. Indeed, as mentioned in §3.1, unbiased estimation would require two nested loops of cross-validation, with the "optimal" number of genes (and value of k for k-NN) being chosen internally to the first (inner) loop of CV and the performance being computed during the second (outer) loop of CV [8]. Even so, the best result in Table 4 is 21 errors, obtained by both $DLDA$ and SVM using one gene.

5.3.2 PAM

PAM [22] is a variant of *DLDA*. There are two differences. The first is the addition of a small positive constant (”fudge factor”) to the denominator of the expression for the distance from the observation to the class mean. The second is the use of ”shrunken” rather than ordinary centroids as prototypes for each class. The fudge factor guards against the possibility of spurious large distances for genes with low expression values. The shrunken centroids and consequent ”soft thresholding” eliminate uninformative genes from the prediction rule as the shrinkage parameter (”threshold”) is increased; this has performed better than ”hard thresholding” (as is the case for the t-test) in other settings [7]. The CV performance is provided as a function of the threshold (or, equivalently, of the number of genes used in the classifier), allowing one to easily find the ”optimal” value of the threshold (or of the number of genes).

The estimated prediction rate is fairly constant over all thresholds (equivalently, from one gene to all $13,547$ genes). The best result is 27 errors (65% prediction accuracy) with 15 genes.

5.3.3 Notes

We also ran both *BRB Array Tools* and *PAM* with missing data imputed with *PAM's* k-NN imputer (the default value of $k = 10$ was kept), which has been shown to be an excellent imputation technique [23]. The results are very similar. The best result among *DLDA, k-NN* and *SVM* is still 21 errors, obtained with *3-NN* using 115 genes, and the optimal performance of *PAM* is again 27 errors, obtained with 21 genes.

Finally, it is somewhat surprising that none of the classifiers in the benchmark experiment performs better than the classifier in [24]. A possible explanation is that since the initial selection of around 5000 genes is external to the cross-validation, the prediction rate in (the supplement to) [24] remains biased.

6 Discussion

We have introduced a new classification methodology for microarray data based entirely on pairwise comparison of *relative* gene expression levels. Basing prediction on *ratios of concentrations* provides a natural link with biochemical activity. In fact, this link may help to explain why the *TSP* classifier appears to be more accurate than many other classifiers while at the same time using fewer genes. Moreover, ratios of concentrations will become more biologically meaningful when mRNA abundance is replaced by actual protein expression data. Indeed, the full potential of this method may not be realized until high-throughput protein comparisons become practical.

Besides the advantage of invariance to normalization, concrete hypotheses about the predictive significance of specific mRNA comparisons are generated naturally by the method, and follow-up studies could be focused on the corresponding list of gene pairs. An example was provided in the case of detecting prostate cancer.

We have chosen leave-one-out (“n-fold”) CV to estimate the error rate of the *TSP* classifier in order to provide an “apples-to-apples” comparison with the other work we cite. In addition, this method is well-known to have low bias. On the other hand, methods such as k-fold CV and bootstrap resampling techniques have been asserted to have smaller variance (see, e.g., [10, 11]), be more appropriate for microarray analysis [5], and be particularly well-suited to classifiers which exhibit various forms of "instability" (see below). For instance, with 10-fold CV, the estimated error rates should be unbiased for a training set of size $.9n$ (rather than of size n) although the variance (sensitivity to the training set) may be reduced; of course, performance is expected to degrade somewhat due to the smaller number of training samples for constructing the classifier. These tradeoffs will be investigated in future work.

We have focused our study on the TSP classifier in which predictions are based entirely on the top-scoring pairs. In most of the cases we have encountered there is in fact a unique top-scoring pair. However, there may be *many pairs* of genes whose relative expression values is informative. Moreover, the top-scoring pair may change when the training data is even slightly perturbed by adding or deleting a few samples. One avenue of future work is to find a more *stable*, comparison-based signature than *the* top-scoring pair or pairs. For example, one may also consider a slightly more complex $k - TSP$ classifier based on all pairs achieving the k best scores. In this case, k is a parameter that should be estimated using cross-validation, hence requiring a double loop of cross-validation to estimate the generalization error. An investigation of the $k-TSP$ classifier, and other extensions of the method introduced here, will be reported elsewhere.

With somewhat larger samples, say several hundreds, the induction of modest-depth decision trees, based on successive entropy reduction and using only comparison questions, becomes feasible, thereby maintaining results which are both easy to interpret and invariant to normalization. The corresponding decision rules would then be based on more complex mRNA comparisons involving more than two genes. The methodology extends almost without modification to more complex and heterogeneous data sets, for example consisting of mixed mRNA and protein abundances.

One could also envision modeling the statistical dependency structure among families of genes and proteins, for example regulatory pathways, based on observed order statistics. With very small amounts of data, it may only be possible to collect reliable estimates of pairwise comparisons among expression levels. More data could lead to estimating the order statistics of triplets, and so forth. This provides a natural, hierarchical family of models which can be adapted to the amount of data.

Acknowledgments: The authors wish to express their sincere thanks to Christina Yung for cheerfully processing some of the raw data related to this study. The contributions from Arnaud Zeboulon, an independent consultant, were supported in part by the Department of Applied Mathematics and Statistics. Daniel Naiman has been supported during the period of this research while on sabbatical at the Inherited Disease Research Branch of NHGRI and by NSF Grant ATM-0222238. Finally, this work was also supported by the Falk Medical Trust and NIH RO1-HL72488.

References

[1] S. Bicciato, M. Pandin, G. Didone, and C. Di Bello. Pattern identification and classification in gene expression data using an autoassociative neural network model. *Biotechnol. Bioeng.*, 81(5):594–606, 2003.

[2] G. Bloom, I. V. Tang, D. Boulware, K. Y. Kwong, D. Coppola, S. Eschrich, J. Quackenbush, and T. J. Yeatman. Multi-platform, multi-site, microarray-based human tumor classification. *Am. J. Pathol.*, 164(1):9–16, 2004.

[3] T. H. Bo and I. Jonassen. New feature subset selection procedures for classification of expression profiles. *Genome Biology*, 3(4):research0017.1–0017.11, 2002.

[4] A. L. Boulestiex, G. Tutz, and K. Strimmer. A CART-based approach to discover emerging patterns in microarray data. *Bioinformatics*, 19(18):2465–2472, 2003.

[5] U.M. Braga-Neto and E.R. Dougherty. Is cross-validation valid for small-sample microarray classification? *Bioinformatics*, 20:374–380, 2004.

[6] M. Dettling and P. Buhlmann. Boosting for tumor classification with gene expression data. *Bioinformatics*, 19(9):1061–1069, 2003.

[7] D. Donoho and I. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81:425–455, 2002.

[8] S. Dudoit and J. Fridlyand. Classification in microarray experiments. In T. Speed, editor, *Statistical Analysis of Gene Expression Microarray Data*. Chapman and Hall, 2003.

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

[10] B. Efron. Estimating the error rate of a prediction rule: Improvement on cross-validation. *J. Amer. Statist. Assoc.*, 78(382):316–331, 1983.

[11] B. Efron and R. Tibshirani. Improvements on cross-validation: The .632+ boostrap method. *J. Amer. Statist. Assoc.*, 92(438):548–560, 1997.

[12] D. Geman, C. d'Avignon, D.Q. Naiman, and R.L Winslow. Classifying gene expression profiles from pariwise mRNA comparisons. Technical report, Johns Hopkins University, 2004.

[13] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Collier, M. L. Loh, J. R. Downing, M. A. Caligiuri, and et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.

[14] J. Khan, J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer. Classification and diagnostic prediction of cancers using gene expression profiling and artifical neural networks. *Nat. Med.*, 7(6):659–659, 2001.

[15] A. S. Mah, J. Jang, and R. J. Deshaies. Protein kinase Cdc15 activates the Dbf2-Mob1 kinase complex. *Proc. Natl. Acad. Sci. USA.*, 98(13):7325–7330, 2001.

[16] F. Muller-Pillasch, C. Wallrap, U. Lacher, H. Freiss, M. Buchler, G. Adler, and T.M. Gress. Identification of a new tumor-associated antigen TM4SF5 and its expression in human cancer. *Gene*, 208:25–30, 1998.

[17] S. Peng, X. B. Ling, X. Peng, W. Du, and L. Chen. Molecular classification of cancer types from microarray data using the combination of genetic algorithms and support vector machines. *FEBS Lett.*, 555(2):358–362, 2003.

[18] P. Sebastiani, E. Gussoni, I. S. Kohane, and M. F. Ramoni. Statistical challenges in functional genomics. *Statistical Science*, 18(1):33–70, 2003.

[19] R. Simon, M. D. Radmacher, K. Dobbin, and L. M. McShane. Pitfalls in the use of DNA microarray data for diagnostic and prognostic classification. *Journal of the National Cancer Institute*, 95(1):14–18, 2003.

[20] D. Singh, P.G. Febbo, D.G. Jackson, J. Manola, C. Ladd, P. Tamayo, A.A. Renshaw, A.V. D'Amico, J.P. Richie, E.S. Lander, M. Loda, P.W. Kantoff, T.R. Golub, and W.R. Sellers. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1(2):203–209, 2002.

[21] G. Stolovitzky. Gene selection in microarray data: the elephant, the blind men and our algorithms. *Curr. Opin. Struct. Biol.*, 13:370–376, 2003.

[22] R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proc. Natl. Acad. Sci. USA*, 99(10):6567–6572, 2002.

[23] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R. B. Altman. Missing value estimation methods for dna microarrays. *Bioinformatics*, 17(6):520–525, 2001.

[24] L.J. van't Veer, H. Dai, M.J van de Vijver, Y. D. He, A.A.M. Hart, M. Mao, H.L. Peterse, K. van der Kooy, M.J. Marton, A.T. Witteveen, G.J. Schreiber, R.M. Kerkhoven, C. Roberts, P.S. Linsley, R. Bernards, and S.H. Friend. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415:530–536, 2002.

[25] M. West, C. Blanchette, H. Dressman, E. Huang, S. Ishida, R. Spang, H. Zuzan, Olson Jr. J.A., Marks. J.R., and J.R. Nevins. Predicting the clinical status of human breast cancer using gene expression profiles. *Proc. Natl. Acad. Sci. USA*, 98:11462–11467, 2001.

[26] Y.H. Yang, S. Dudoit, P. Luu, D.M. Lin, V. Peng, J. Ngal, and T.P. Speed. Normalization for cdna microarray data. In *Microarrays: Optical Technologies and Informatics, Proc. SPIE*, volume 4266, pages 141–152. 2001.

[27] C. H. Yeang, S. Ramaswany, P. Tamayo, S. Mukherjee, R. M. Rifkin, M. ngelo, M. Reich, E. Lander, J. Mesirov, and T. Golub. Molecular classification of multiple tumor types. *Bioinformatics*, 17 Suppl 1:S316–322, 2001.

[28] H. Zhang, C. Y. Yu, and B. Singer. Cell and tumor classification using gene expression data: construction of forests. *Proc. Natl. Acad. Sci. USA*, 100(7):4168–4172, 2003.