SIAM REVIEW
Vol. 45, No. 4, pp. 706–723

An Analytical Method for Multiclass Molecular Cancer Classification*

Ryan Rifkin[†][‖][‡‡]
Sayan Mukherjee[†][‖][‡‡]
Pablo Tamayo[†][‡‡]
Sridhar Ramaswamy[†][‡]
Chen-Hsiang Yeang[†][**]
Michael Angelo[††]
Michael Reich[†]
Tomaso Poggio[‖]
Eric S. Lander[†][¶]
Todd R. Golub[†][§]
Jill P. Mesirov[†]

Abstract. Modern cancer treatment relies upon microscopic tissue examination to classify tumors according to anatomical site of origin. This approach is effective but subjective and variable even among experienced clinicians and pathologists. Recently, DNA microarray-generated gene expression data has been used to build molecular cancer classifiers. Previous work from our group and others demonstrated methods for solving pairwise classification problems using such global gene expression patterns. However, classification across multiple primary tumor classes poses new methodological and computational challenges. In this paper we describe a computational methodology for multiclass prediction that combines class-specific (one vs. all) binary support vector machines. We apply this methodology to the diagnosis of multiple common adult malignancies using DNA microarray data from a collection of 198 tumor samples, spanning 14 of the most common tumor types. Overall classification accuracy is 78%, far exceeding the expected accuracy for random classification. In a large subset of the samples (80%), the algorithm attains 90% accuracy. The methodology described in this paper both demonstrates that accurate gene expression-based multiclass cancer diagnosis is possible and highlights some of the analytic challenges inherent in applying such strategies to biomedical research.

*Received by the editors July 26, 2002; accepted for publication (in revised form) April 18, 2003; published electronically October 31, 2003. This work was supported in part by grants from Affymetrix Inc., Millennium Pharmaceuticals Inc., Bristol-Myers Squibb Company (ESL), a Harvard/NIH Training Grant in Molecular Hematology (SR), and the SLOAN/DOE Fellowship in Computational Molecular Biology.

http://www.siam.org/journals/sirev/45-4/41198.html

[†] Whitehead Institute/Massachusetts Institute of Technology Center for Genome Research, Cambridge, MA 02139 (rif@genome.wi.mit.edu, sayan@mit.edu, tamayo@genome.wi.mit.edu, sridhar@genome.wi.mit.edu, chyeang@mit.edu, michaelr@genome.wi.mit.edu, lander@genome.wi.mit.edu, golub@genome.wi.mit.edu, mesirov@genome.wi.mit.edu).

[‡] Department of Adult Oncology, Dana-Farber Cancer Institute, Boston, MA 02115.

[§] Department of Pediatric Oncology, Dana-Farber Cancer Institute, Boston, MA 02115.

[¶] Department of Biology, Massachusetts Institute of Technology, Cambridge, MA 02139.

[‖] McGovern Institute, Center for Biological and Computational Learning, Massachusetts Institute of Technology, Cambridge, MA 02139 (tp@ai.mit.edu).

[**] Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139.

[††] X-Mine, Brisbane, CA 94005 (mangelo@xmine.com).

[‡‡] These authors contributed equally to this work.

Key words. multiclass classification, support vector machine, tumor, molecular classification, pattern recognition, cancer, computational biology

AMS subject classifications. 68T10, 62P10, 92B15, 92C50

DOI. 10.1137/S0036144502411986

1. Introduction.

The accurate classification of human cancer is an important component of modern cancer treatment. It is estimated that upwards of 40,000 cancer cases per year in the United States are difficult to classify using standard clinical and histopathologic approaches. Molecular approaches to cancer classification have the potential to effectively address these difficulties. However, decades of research in molecular oncology have yielded few useful tumor-specific molecular markers. An important goal in cancer research, therefore, continues to be the identification of tumor-specific genes for the purpose of molecular cancer classification.

Proteins in cells are essential to all of the functions of life: energy production, biosynthesis of molecules that form and maintain complex cellular and subcellular structures, response to environmental changes, proliferation, etc. Protein production is controlled by the coordinated "expression" of genes by means of "transcription" of DNA sequences into messenger RNA that is in turn "translated" into protein sequences. In this context, the regulation of protein production by induction and repression of gene expression plays a critical role in the major biological decisions of a cell such as division, differentiation, and death. For example, although each cell in an individual's body contains the same genome, they are distinct (i.e., some cells are liver cells while others are heart cells). Gene expression has been shown to be critical to making this differentiation. A particularly striking example is the MyoD gene, a transcription factor whose expression can turn essentially any cell into a muscle cell. Human diseases are also correlated with changes in the expression levels of genes. For example, in diseases such as cystic fibrosis or Huntington's disease, point mutations can result in the absence of gene transcripts for particular genes, which in turn results in disease. Based on this knowledge, there has been a significant effort to develop methods for measuring and analyzing the level of gene expression for many genes simultaneously in biological samples. The rationale for such an approach is that the level of gene expression is critical for determining the biological properties of cells [8, 9].

Several technologies for the high-throughput analysis of gene expression have been developed over the past 10 years. The two most common platforms are spotted cDNA and oligonucleotide microarrays. cDNA microarrays, pioneered by Brown and colleagues at Stanford [12], involves the hybridization of fluorescently labeled cDNA derived from sample RNAs to glass slides onto which DNA strands corresponding to genes of interest have been robotically deposited. The second platform, oligonucleotide arrays, developed at Affymetrix, Inc., involves the hybridization of fluorescently labeled RNAs to short 25-long oligonucleotides of known sequence that are photolithographically synthesized on a solid surface using technology similar to that used to make silicon chips [20]. Alternative oligonucleotide array strategies employing longer oligos (60–70 mers) generated by ink-jet printing and other technologies appear similarly effective. While few head-to-head comparisons of these methods have been made, the technologies are sensitive, quantitative, and reproducible.

Thus, microarrays permit the simultaneous measurement of thousands of expressed genes in biological samples and have made possible the creation of large datasets of molecular information that represent molecular "snapshots" of biological systems of interest (see, e.g., Dougherty's article in *SIAM News*, May 2002 [11]). These gene expression profiles can then be characterized by the application of large-scale data analysis and may serve as fingerprints for accurate molecular classification as well as improve our understanding of normal and disease states [16, 29, 15, 4, 7].

Previously, the Cancer Genomics program at the Whitehead Institute's Center for Genome Research developed computational approaches (unsupervised and supervised learning) using gene expression to accurately distinguish between two common blood cancer classes: acute lymphocytic and acute myelogenous leukemia [16, 29, 23]. The classification of primary solid tumors (e.g., breast cancer), in contrast, is a harder problem due to limitations concerning sample availability, identification, acquisition, integrity, and preparation. Moreover, a solid tumor is a heterogeneous cellular mix, and gene expression profiles might reflect contributions from nonmalignant components, confounding classification. In addition, there are some intrinsic complexities in making multiclass, as opposed to binary class, distinctions.

In this context we asked whether it is possible to achieve a general, multiclass, molecular-based cancer classification based solely on gene expression profiles [25]. This paper describes the details of this approach. The most accurate computational methodology is described in detail, and technical limitations and challenges involved are stated. Our methodology is based on combining multiple binary support vector machine (SVM) classifiers trained to predict a sample's class membership. We apply this methodology to the diagnosis of multiple common adult malignancies using a collection of 198 DNA microarray tumor samples, spanning 14 common tumor types. In the next section we describe the sample collection and the associated experimental protocol. Then we describe the problems associated with multiclass prediction and introduce our methodology and results. A technical description of the SVM algorithm is included in the appendix. The methodology described in this paper suggests that systematic and unified cancer diagnosis is possible by the comparison of an unknown sample to a large reference database and provides a first look at the technical difficulties and computational challenges associated with such a system.

2. Sample Collection and Experimental Protocol. The gene expression datasets were obtained following a standard experimental protocol published elsewhere [24, 16] and described schematically in Figure 2.1. Tumors were biopsies from primary sites obtained prior to any treatment. RNA from each tumor was sequentially hybridized to Affymetrix Hu6800 and Hu35KsubA oligonucleotide microarrays (GeneChips$^{\text{TM}}$) containing a total of 16,063 probe sets. Microarrays were scanned using standard Affymetrix protocols and scanners. Expression values for each gene were calculated using Affymetrix GeneChip software.

Tumor data (198 samples) was organized into a training set with 144 samples and a test set with 54 samples (see Table 2.1). For more details on the experimental system and specimen characteristics, see [25]. The datasets are available on our Web site (www-genome.wi.mit.edu/MPR/GCM).

Supervised learning involves "training" a classifier to recognize the distinctions between classes and testing the accuracy of the classifier on an independent test set. Multiclass classification in this context is especially challenging for several reasons, including: (i) large dimensionality of the datasets, (ii) small but significant uncertainty in the original labeling, (iii) measurement noise, (iv) intrinsic biological variation from specimen to specimen, and (v) the small number of examples.

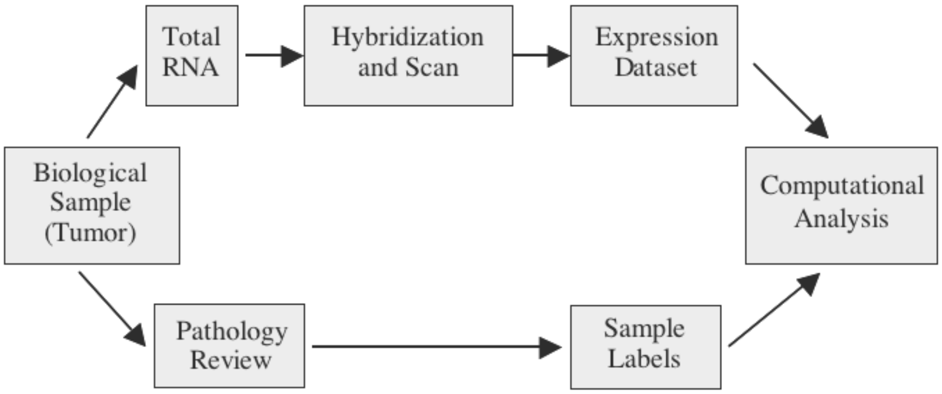


Fig. 2.1 *Experimental protocol and dataset creation.*

Table 2.1 *Number of tumor samples per class.*

Tumor class	# Train	# Test	Tumor class	# Train	# Test
Breast (BR)	8	3	Uterus (UT)	8	2
Prostate (PR)	8	2	Leukemia (LE)	24	6
Lung (LU)	8	3	Renal (RE)	8	3
Colorectal (CO)	8	5	Pancreas (PA)	8	3
Lymphoma (LY)	16	6	Ovary (OV)	8	3
Bladder (BL)	8	3	Mesothelioma (ML)	8	3
Melanoma (ME)	8	2	Brain (CNS)	16	4

As part of a preliminary feasibility study we explored a variety of prediction methodologies and algorithms and applied them to multiple tumor datasets. The results of this investigation were reported in [33]. Here we will describe in detail the technique that gave us the most accurate results.

3. Multiclass Supervised Classification. Multiple class prediction is intrinsically harder than binary prediction because the classification algorithm has to learn to construct a greater number of separation boundaries or relations. In binary classification an algorithm can “carve out” the appropriate decision boundary for only one of the classes; the other class is simply the complement. In multiclass classification each class has to be explicitly defined. Errors can occur in the construction of any one of the many decision boundaries, so the error rates on multiclass problems can be significantly greater than those of binary problems. For example, in contrast to a balanced binary problem where the accuracy of a random prediction is 50%, for K classes the accuracy of a random predictor is of the order of 1/K.

There are basically two types of multiclass classification algorithms. The first type deals directly with multiple values in the target field. For example, naïve Bayes, k-nearest neighbors, and classification trees are in this class. Intuitively, these methods can be interpreted as trying to construct a conditional density for each class, then classifying by selecting the class with maximum a posteriori probability. The second type decomposes the multiclass problem into a set of binary problems and then combines them to make a final multiclass prediction. This class contains support vector machines, boosting [28], and weighted voting algorithms and, more generally, any binary classifier. Boosting in fact can be used to combine intrinsic multiclass classifiers as well [27]. In certain settings the latter approach results in better performance than the multiple target approaches. Intuitively, when we have a high-dimensional input space and very few samples per class, we expect that it will be very difficult to construct accurate densities and that the second approach will perform better. Our dataset belongs to this category.

The basic idea behind combining binary classifiers is to decompose the multiclass problem into a set of easier and more accessible binary problems. The main advantage in this divide-and-conquer strategy is that any binary classification algorithm can be used. Besides choosing a decomposition scheme and a base classifier, one also needs to devise a strategy for combining the binary classifiers and providing a final prediction. The problem of combining binary classifiers has been studied in the computer science literature [19, 1, 17] from theoretical and empirical perspectives. However, the literature is inconclusive, and the best method for combining binary classifiers for any particular problem is undecided.

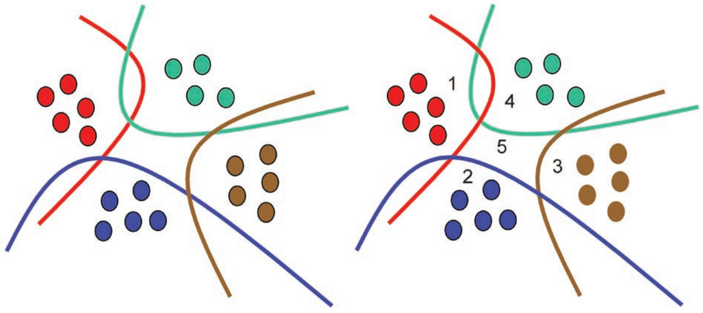
The decomposition problem in itself is quite old and can be considered an example of the collective vote-ranking problem addressed by Condorcet and others at the time of the French Revolution. Condorcet was interested in solving the problem of how to deduce a collective ranking of candidates based on individual voters' preferences. He proposed a decomposition scheme based on binary questions [21] and then introduced analytical rules to obtain a consistent collective ranking based on the individual's answers to these binary questions. It turns out that a consistent collective ranking is not guaranteed in all cases, and this led to the situation known as Condorcet's or Arrow's paradox [3].

Standard modern approaches to combining binary classifiers can be stated in terms of what is called "output coding" [10]. The basic idea behind output coding is the following: given K classifiers trained on various partitions of the classes, a new example is mapped into an output vector. Each element in the output vector is the output from one of the K classifiers, and a "codebook" is then used to map from this vector to the class label (see Figure 3.1). For example, given three classes, the first classifier may be trained to partition classes 1 and 2 from 3, the second classifier trained to partition classes 2 and 3 from 1, and the third classifier trained to partition classes 1 and 2 from 3.

Two common examples of output coding are the one-versus-all (OVA) and all-pairs (AP) approaches. In the OVA approach, given K classes, K independent classifiers are constructed where the ith classifier is trained to separate samples belonging to class i from all others. The codebook is a diagonal matrix and the final prediction is based on the classifier that produces the strongest confidence,

$$\text{class} = \arg \max_{i=1,\dots,K} f_i,$$

where f_i is the signed confidence measure of the ith classifier (i.e., the margin of the SVM). In the AP approach $K(K-1)/2$ classifiers are constructed with each classifier trained to discriminate between a class pair (i and j). This can be thought of as a $K \times K$ matrix, where the ijth entry corresponds to a classifier that discriminates between classes i and j. The codebook, in this case, is used to simply sum the entries



(a) (b)

	R	G	B	T
R	+1	-1	-1	-1
B	-1	+1	-1	-1
G	-1	-1	+1	-1
T	-1	-1	-1	+1

(c)

	R	G	B	T	class
1	.7	-.2	-3	-6	R
2	-.5	-1	.5	-5	B
3	-6	-1	-.3	.7	T
4	-.2	.4	-2	-1	G
5	-1	-.1	-1	-1	G

(d)

Fig. 3.1 *OVA classification.* (a) *Four binary classifiers are trained. The first discriminates the red from other classes, the second green from the other classes, the third blue, and the fourth tan.* (b) *The colored numbers are five new samples to be classified.* (c) *The codebook for OVA classification. The top row lists each OVA classifier; the numbers in the matrix are the ideal outputs for the class labels listed in the leftmost column.* (d) *The outputs of the four classifiers and the final class labels for the five new samples. The label assigned was that of the OVA classifier with the largest output.*

of each row and select the row for which this sum is maximal,

$$\text{class} = \arg \max_{i=1,\ldots,K} \left[\sum_{j=1}^{K} f_{ij} \right],$$

where, as before, f_{ij} is the signed confidence measure for the ijth classifier.

An ideal code matrix should be able to correct the mistakes made by the component binary classifiers. Dietterich and Bakiri [10] used error-correcting codes to build the output code matrix where the final prediction is made by assigning a sample to the codeword with the smallest Hamming distance with respect to the binary prediction result vector. There are several other ways of constructing error-correcting codes, including classifiers that learn arbitrary class splits and randomly generated matrices [5, 1, 17].

Intuitively, there is a tradeoff between the OVA and AP approaches. The discrimination surfaces that need to be learned in the AP approach are, in general, more natural and, theoretically, should be more accurate. However, with fewer training examples the empirical surface constructed may be less precise. The actual performance of each of these schemes, or others such as random codebooks, in combination with different classification algorithms is problem dependent.

Table 3.1 *Accuracy of different combinations of multiclass approaches and algorithms.*

Number of genes per classifier	Weighted voting OVA	Weighted voting AP	k-nearest neighbors OVA	k-nearest neighbors AP	SVM OVA	SVM AP
30	60.0%	62.3%	65.3%	67.2%	70.8%	64.2%
92	59.5%	59.6%	68.0%	67.3%	72.2%	64.8%
281	57.8%	57.2%	65.7%	67.0%	73.4%	65.1%
1073	53.5%	52.4%	66.5%	64.8%	74.1%	64.9%
3276	43.4%	48.4%	66.3%	62.0%	74.7%	64.7%
6400	38.5%	45.6%	64.2%	58.4%	75.5%	64.6%
All	—	—	—	—	78.0%	64.7%

In a preliminary empirical study of multiclass methods and algorithms [33] we applied the OVA and AP approaches with three different algorithms: weighted voting [16, 29], k-nearest neighbors, and SVMs. The results, shown in Table 3.1, demonstrate that the OVA approach in combination with SVM gave us the most accurate method by a significant margin, and we describe this method in detail below. See [33] for more details on using other algorithms.

4. Support Vector Machines. SVMs are powerful classification systems based on a variation of regularization techniques for regression [31, 14]. SVMs provide state-of-the-art performance in many practical binary classification problems [31, 14]. SVMs have also shown promise in a variety of biological classification tasks, including some involving gene expression microarrays [23, 6]. For a detailed description of the algorithm, see the appendix.

The algorithm is a particular instantiation of regularization for binary classification. Linear SVMs can be viewed as a regularized version of a much older machine-learning algorithm, the perceptron [26, 22]. The goal of a perceptron is to find a *separating hyperplane* that separates positive from negative examples. In general, there may be many separating hyperplanes. In our problem, this separating hyperplane is the boundary that separates a given tumor class from the rest (OVA) or that separates two different tumor classes (AP). The hyperplane computed by the SVM is the maximal margin hyperplane, that is, the hyperplane with maximal distance to the nearest data point. Finding the SVM solution requires training an SVM, which entails solving a convex quadratic program with as many variables as training points.

The SVM experiments described in this paper were performed using a modified version of the SvmFu package (http://www.ai.mit.edu/projects/cbcl/). The advantages of SVM, when compared with other algorithms, are their sound theoretical foundations [31, 14], intrinsic control of machine capacity that combats over-fitting, ability to approximate complex classification functions, fast convergence, and good empirical performance in general. Standard SVMs assume the target values are binary and that the classification problem is intrinsically binary. We use the OVA methodology to combine binary SVM classifiers into a multiclass classifier. A separate SVM is trained for each class, and the winning class is the one with the largest margin, which can be thought of as a signed confidence measure.

In the experiments described in this paper there were few data points in many dimensions. Therefore, we used a kernel that corresponds to a linear (regularized) classifier as the SVM solution. Although we did allow the hyperplane to make misclassifications, in all cases involving the full 16,063 dimensions each OVA hyperplane fully separated the training data with no errors. In some of the experiments, which involved explicit feature selection with very few features, there were some training errors. This may indicate that we could select a very small number of features and then use a nonlinear kernel function to improve classification; however, preliminary experiments with this approach yielded no improvement over the linear case.

Many methods exist for performing feature selection. We obtained similar results from informal experiments using signal-to-noise ratio [29], recursive feature elimination (RFE) [18], and radius-margin-ratio [23, 32]. For the formal presentation we used RFE since it was the most straightforward to implement with the SVM. An SVM is trained using all features. The features are ranked according to the magnitude of the elements of the resulting hyperplane, so the importance of feature i is the weight of the ith element of the hyperplane.

5. Methodology and Results. As mentioned before, the most accurate methodology for our multitumor dataset is a combination of an OVA approach and SVMs using all genes. To evaluate the accuracy of this method as a function of the number of genes, RFE was used to select features (see Figure 5.3). See the appendix for more details about SVMs, RFE, and implementation. Leave-one-out cross validation and an independent test set were used to test the methodology.

The procedure is as follows:

1. Define each target class based on histopathologic clinical evaluation (pathology review) of tumor specimens.
2. Decompose the multiclass problem into a series of 14 binary OVA classification problems, one for each class. A new sample is assigned the label of the OVA classifier corresponding to the largest confidence (margin) class $= \arg\max_{i=1,\dots,K} f_i$.
3. Test OVA classifiers on an independent test set.

This procedure is described pictorially in Figure 5.1, where the bar graphs on the lower right side show two examples of actual SVM output predictions of each of the 14 SVMs for a lymphoma and a breast sample.

The confidence of the final call is the margin of the winning SVM. When the largest confidence is positive the final prediction is considered a “high confidence” call. If negative, it is a “low confidence” call that can also be considered a candidate for a no-call because no single SVM “claims” the sample as belonging to its recognizable class. This simple categorization of prediction calls as having high or low confidence is very useful in analyzing the multiclass prediction results and developing a basic understanding of the nature of errors. In a clinical classification setting, where a model of this type might be deployed to aid diagnosis, one would need to further refine the analysis of the prediction calls beyond this simple categorization. There are several ways to do this. For example, one can consider as model output a ranked list of the top predictors with confidence above a threshold. This would allow multiple calls of different confidence levels in some indefinite “gray” cases. Another possibility would be to consider the relative costs of misclassification by using a full cost matrix and then applying an ROC (receiver operating characteristics) analysis of the predictions in order to minimize total cost or to understand the tradeoff between different types of errors. Both of these alternatives require a more general study of how to effectively deploy a molecular classifier in the clinic, which is beyond the scope of this paper.

In the first example in the lower right-hand side of Figure 5.1, an example of a high confidence call, the lymphoma classifier attains a large positive margin and

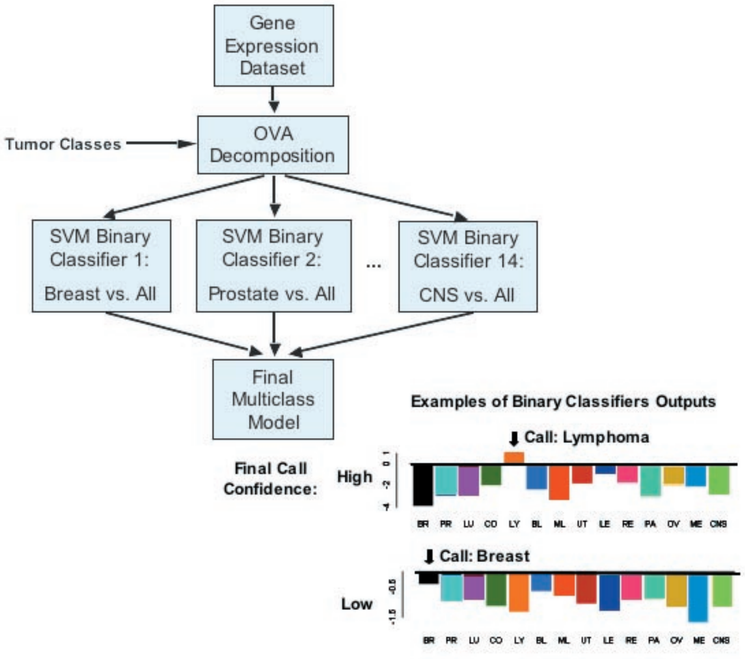

Gene Expression Dataset
Tumor Classes
OVA Decomposition
SVM Binary Classifier 1: Breast vs. All
SVM Binary Classifier 2: Prostate vs. All
...
SVM Binary Classifier 14: CNS vs. All
Final Multiclass Model
Examples of Binary Classifiers Outputs
Final Call Confidence:
High
Call: Lymphoma
1
0
-2
-4
BR PR LU CO LY BL ML UT LE RE PA OV ME CNS
Low
Call: Breast
-0.5
-1.5
BR PR LU CO LY BL ML UT LE RE PA OV ME CNS

Fig. 5.1 *Multiclass methodology using an OVA approach. The bar graphs on the lower right side show the results of a high and a low confidence prediction.*

Table 5.1 *Accuracy results for the OVA-SVM classifier.*

Dataset	Sample type	Validation method	Sample number	Total accuracy	Confidence High fraction	 Accuracy	 Low fraction	 Accuracy
Train	Multiple tumors	Cross val.	144	78%	80%	90%	20%	28%
Test	Multiple tumors	Train/test	54	78%	78%	83%	23%	58%

recognizes the samples in this class. No other classifier produces a comparable positive prediction. The second example is a low confidence call where the winning classifier is the one corresponding to the breast class. In this case the sample is classified as breast but the confidence of the prediction is much lower than in the first example. Repeating this procedure, we created a multiclass OVA-SVM model with all genes using the training dataset and then applied it to the two test datasets. The results are summarized in Table 5.1.

As can be seen in the table, in cross validation the overall multiclass predictions were correct for 78% of the tumors. This accuracy is substantially higher than expected for random prediction (9% according to proportional chance criterion). More interestingly the majority of calls (80%) were high confidence, and for these the classifier achieved an accuracy of 90%. The remaining tumors (20%) had low confidence

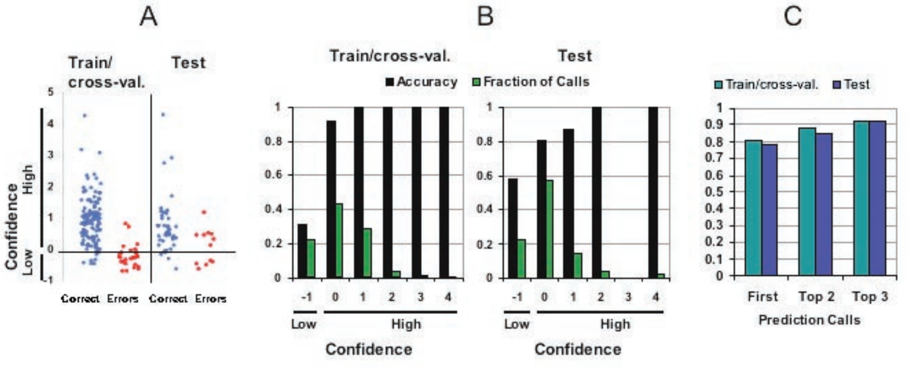


Fig. 5.2 *Individual sample errors and confidence calls* (A)*, accuracy bar graphs* (B)*, and performance considering second and third most confident prediction* (C) *for the train/cross validation and test datasets.*

calls and lower accuracy (28%). The results for the test set (test set 1) were similar to those obtained in cross validation: the overall prediction accuracy was 78% and the majority of these predictions (78%) were again high confidence with an accuracy of 83%. Low confidence calls were made on the remaining 22% of tumors with an accuracy of 58%. The actual confidences for each call and a bar graph of accuracy and fraction of calls versus confidence is shown in Figure 5.2 (A), (B). The confusion matrices are shown in Table 5.2.

An interesting observation concerning these results is that for 50% of the tumors that were incorrectly classified the correct answer corresponded to the second or third most confident (SVM) prediction. This is shown in Figure 5.2 (C).

We also analyzed the accuracy of the multiclass SVM predictor as a function of the number of genes (or features). The algorithm inputs all of the 16,063 genes in the array and each of them is assigned a weight based on its relative contribution to each OVA classification. Practically, all genes were assigned weakly positive and negative weights in each OVA classifier. We performed multiple runs with different numbers of genes selected using RFE. Results are shown in Figure 5.3. Note that total accuracy decreases as the number of input genes decreases for each OVA distinction. Pairwise distinctions can easily be made between some tumor classes using fewer genes, but multiclass distinctions among highly related tumor types are intrinsically more difficult. This behavior can also be the result of the existence of molecularly distinct but unknown subclasses within known classes that effectively decrease the predictive power of the multiclass method. Despite the increase in accuracy when utilizing more genes, significant but modest prediction accuracy can be achieved with a relatively small number of genes per classifier (e.g., about 70% with about 200 total genes).

To confirm the stability and reproducibility of the prediction results for this collection of samples, we repeated the train and test procedure for 100 random splits of a combined dataset. The results were similar to the reported case. Figure 5.3 shows the mean and standard variation of the error rate for the different test/train splits as a function of the total number of genes. Due to the fact that different test/train splits were obtained by reshuffling the dataset, the empirical variance measured is optimistic [13].

Table 5.2 *Confusion matrices for the OVA-SVM classifier on training and test data. The dashes stand for no errors.*

Train	BL	BR	CNS	CO	LE	LU	LY	ME	ML	OV	PA	PR	RE	UT	Totals
BL	5	-	-	1	-	-	-	1	-	1	-	-	-	-	8
BR	-	7	-	-	-	-	-	-	-	-	-	-	-	-	8
CNS	-	-	16	-	-	-	-	-	-	-	-	-	-	-	16
CO	-	1	-	6	-	-	-	-	-	1	-	-	-	-	8
LE	-	-	-	-	24	-	-	-	-	-	-	-	-	-	24
LU	1	1	-	1	-	4	-	-	-	1	-	-	-	-	8
LY	-	-	-	-	-	-	16	-	-	-	-	-	-	-	16
ME	-	-	-	-	-	-	-	8	-	-	-	-	-	-	8
ML	-	-	-	1	-	-	-	-	5	-	1	1	-	-	8
OV	1	-	-	-	-	-	-	-	1	3	-	-	1	2	8
PA	1	1	-	-	-	-	-	1	-	-	5	-	-	-	8
PR	-	-	-	-	-	1	-	-	1	-	-	6	-	-	8
RE	-	1	-	-	-	-	-	-	1	1	-	-	5	-	8
UT	-	-	-	-	-	-	-	-	-	1	-	-	-	7	8
Totals	8	11	16	10	24	5	16	10	8	8	6	7	6	9	144

Test 1	BL	BR	CNS	CO	LE	LU	LY	ME	ML	OV	PA	PR	RE	UT	Totals
BL	2	-	-	-	-	-	-	1	1	-	-	-	-	-	3
BR	-	2	-	-	-	-	-	-	-	1	1	-	-	-	4
CNS	-	-	4	-	-	-	-	-	-	-	-	-	-	-	4
CO	-	-	-	4	-	-	-	-	-	-	-	-	-	-	4
LE	-	-	-	-	5	-	-	-	-	-	-	1	-	-	6
LU	1	-	-	-	-	2	-	-	1	-	-	-	-	-	4
LY	-	-	-	-	-	-	6	-	-	-	-	-	-	-	6
ME	-	-	-	-	-	-	-	3	-	-	-	-	-	-	3
ML	1	-	-	-	-	-	-	-	1	-	-	-	-	-	2
OV	-	-	-	-	-	-	-	-	1	2	1	-	-	-	4
PA	-	1	-	-	-	-	-	1	-	-	2	-	-	-	3
PR	-	-	-	-	-	-	-	-	-	1	-	4	-	1	6
RE	-	-	-	-	-	-	-	-	-	-	-	-	3	-	3
UT	-	-	-	-	-	-	-	-	-	-	-	-	-	2	2
Totals	4	3	4	4	5	2	6	3	4	4	4	5	3	3	54

6. Conclusions.

The method described in this paper demonstrates that the simultaneous diagnosis of multiple tumor types is possible by the comparison of an unknown sample to a reference database of multiple tumor types. The combination of the OVA approach and SVM provides a powerful technique for molecular classification of cancer that appears to outperform other approaches. It is remarkable that a majority of the samples (80%) can be predicted with high confidence and accuracy (83%) based solely on gene expression and despite the presence of varying unknown proportions of nonneoplastic elements in the tumor specimens and the amount of noise, biological variation, and uncertainty in the labeling. Despite the success of the method some questions remain open.

What is the exact reason for the remaining errors made by the classification model? We found that most of the errors correspond to low confidence calls and that many of these were made on samples with moderately or poorly differentiated (high-grade) carcinomas. Those tumors are usually difficult to classify using traditional methods since they often lack the characteristic morphological hallmarks of the organ from which they arise. The mostly even distribution of errors throughout the solid tumor classes (see Table 5.2) suggests that total accuracy might increase by

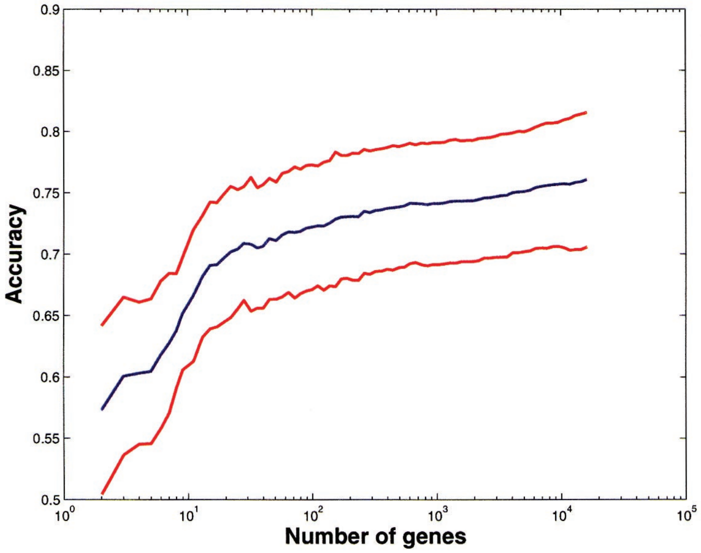


Fig. 5.3 *Mean classification accuracy and standard deviation plotted as a function of number of genes used by the classifier. The blue curve is the mean accuracy and the red curves are one standard deviation above and below the mean accuracy. The prediction accuracy decreases with decreasing number of genes.*

considering a larger number of samples in the training set. Another source of error might be associated with the inclusion of histopathologically similar, but genetically distinct, tumors in various training set classes. In addition, the fact that 50% of the incorrectly classified tumors belong to the class with the second or third most confident prediction suggests that these errors might be corrected by an incremental improvement (e.g., by acquiring some extra data points for some classes or perhaps by an incremental improvement of the algorithm).

The OVA-SVM–based classification strategy may not be the optimal method when a larger sample collection becomes available. As discussed above, there is a tradeoff between the OVA and AP approaches. One can speculate that with a larger number of samples for a fixed class number, there will be a crossover point beyond which the AP-SVM method will be the best. A larger number of tumors with detailed clinical annotation will be required to answer all these questions and fully explore the limitations of our approach.

One can envision a centralized gene expression database that will allow classification accuracy to improve as the model is trained by larger and larger amounts of data. In this scenario the clinical classification and diagnosis of tumors can be improved by considering objective, systematic, and standardized computational methods like the one described here. A more refined analysis of prediction results, for example, using a more detailed categorization of calls by confidence level and class, or by using ROC analysis, might increase the effectiveness of such a system in a clinical setting. This expression-based cancer classification will not be a substitute for traditional diagnostics, but it may represent an important adjunct. The hope is that the molecular characteristics of a tumor sample may for the most part remain the same despite atypical clinical or varying histological features. If this is indeed the case, then the deployment of these computational methods will enable powerful new strategies for a more uniform and comprehensive molecular classification of primary and metastatic tumors.

Appendix. Support Vector Machines. The problem of learning a classification boundary given positive and negative examples is a particular case of the problem of approximating a multivariate function from sparse data. The problem of approximating a function from sparse data is ill-posed, and regularization theory is a classical approach to solving it [30].

Standard regularization theory formulates the approximation problem as a variational problem of finding the function $\hat{f}$ that minimizes the functional $I[f]$, that is,

$$\min_{f \in \mathcal{H}} \frac{1}{\ell} \sum_{i=1}^{\ell} V(y_i, f(\mathbf{x}_i)) + \lambda ||f||_K^2,$$

where $V(\cdot,\cdot)$ is a loss function, $||f||_K^2$ is a norm in a reproducing kernel Hilbert space defined by the positive function $K(\mathbf{x}_i, \mathbf{x}_j)$ [2], ℓ is the number of training examples, and λ is the regularization parameter. Under rather general conditions the solution to the above functional has the form

$$f(\mathbf{x}) = \sum_{i=1}^{\ell} \alpha_i K(\mathbf{x}, \mathbf{x}_i).$$

SVMs are a particular case of the above regularization framework [14].

Here we describe in more detail how the SVM algorithm works. The SVM runs described in this paper were performed using a modified version of SvmFu (http://www.ai.mit.edu/projects/cbcl/). We also describe the feature selection method used to study how the accuracy changes as a function of the number of genes. For a more comprehensive introduction to SVM, see [14, 31].

For the SVM the regularization functional minimized is the following:

$$\min_{f \in \mathcal{H}} \frac{1}{\ell} \sum_{i=1}^{\ell} (1 - y_i, f(\mathbf{x}_i))_+ + \lambda ||f||_K^2,$$

where the hinge loss function is used, $(a)_+ \equiv \max(a, 0)$. The solution again has the form

$$f(\mathbf{x}) = \sum_{i=1}^{\ell} \alpha_i K(\mathbf{x}, \mathbf{x}_i),$$

and the label output is simply $\text{sign}[f(\mathbf{x})]$.

There is an intuitive geometric interpretation of the SVM approach in the case of a linear classifier, which corresponds to $K(\mathbf{x}, \mathbf{x}_i) = \mathbf{x} \cdot \mathbf{x}_i$. A hyperplane is defined via its normal vector $\mathbf{w}$. Given a hyperplane $\mathbf{w}$ and a point $\mathbf{x}$, define $\mathbf{x}_0$ to be the closest point to $\mathbf{x}$ on the hyperplane—the closest point to $\mathbf{x}$ that satisfies $\mathbf{w} \cdot \mathbf{x}_0 = 0$ (see Figure A.1). We then have the following two equations:

$$\begin{aligned} \mathbf{w} \cdot \mathbf{x} &= k \text{ for some } k, \\ \mathbf{w} \cdot \mathbf{x}_0 &= 0. \end{aligned}$$

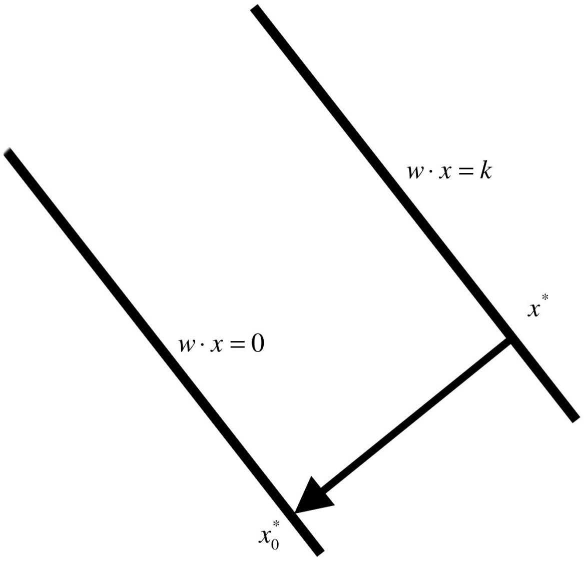


Fig. A.1 *The distance between points x and x_0.*

Subtracting the above equations we obtain

$$\mathbf{w} \cdot (\mathbf{x} - \mathbf{x}_0) = k.$$

Dividing by the norm of $\mathbf{w}$, we have

$$\frac{\mathbf{w}}{||\mathbf{w}||} \cdot (\mathbf{x} - \mathbf{x}_0) = \frac{k}{||\mathbf{w}||}.$$

Noting that $\frac{\mathbf{w}}{||\mathbf{w}||}$ is a unit vector and the vector $\mathbf{x} - \mathbf{x}_0$ is parallel to $\mathbf{w}$, we conclude that

$$||\mathbf{x} - \mathbf{x}_0|| = \frac{k}{||\mathbf{w}||}.$$

Our goal is to maximize the distance between the hyperplane and the closest point, with the constraint that the points from the two classes lie on separate sides of the hyperplane. We could try to solve the following optimization problem:

$$\max_{\mathbf{w}} \min_{\mathbf{x}_i} \frac{y_i(\mathbf{w} \cdot \mathbf{x}_i)}{||\mathbf{w}||} \text{ subject to } y_i(\mathbf{w} \cdot \mathbf{x}_i) > 0 \ \forall \ \mathbf{x}_i.$$

Note that $y_i(\mathbf{w} \cdot \mathbf{x}_i) = k$ in the above derivation. For technical reasons, the optimization problem stated above is not easy to solve. One difficulty is that if we find a solution $\mathbf{w}$, then $c\mathbf{w}$ for any positive constant c is also a solution. In some sense, we are interested in the direction of the vector $\mathbf{w}$, but not its length.

If we can find any solution $\mathbf{w}$ to the above problem, for example, by scaling $\mathbf{w}$, we can guarantee that $y_i(\mathbf{w} \cdot \mathbf{x}_i) \geq 1k$ for all $\mathbf{x}_i$. Therefore, we may equivalently solve the problem

$$\max_{\mathbf{w}} \min_{\mathbf{x}_i} \frac{y_i(\mathbf{w} \cdot \mathbf{x}_i)}{||\mathbf{w}||} \text{ subject to } y_i(\mathbf{w} \cdot \mathbf{x}_i) \geq 1 \ \forall \ \mathbf{x}_i.$$

Note that the original problem has more solutions than this one, but since we are only interested in the direction of the optimal hyperplane, this would suffice. We now restrict the problem further: we are going to find a solution such that for any point closest to the hyperplane, the inequality constraint will be satisfied as an equality. Keeping this in mind, we can see that

$$\min_{\mathbf{x}_i} \frac{y_i(\mathbf{w} \cdot \mathbf{x}_i)}{||\mathbf{w}||} = 1.$$

So the problem becomes

$$\max_{\mathbf{w}} \frac{1}{||\mathbf{w}||} \text{ subject to } y_i(\mathbf{w} \cdot \mathbf{x}_i) \geq 1 \ \forall \ \mathbf{x}_i,$$

which can be transformed [31] into the equivalent problem

$$\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||^2 \text{ subject to } y_i(\mathbf{w} \cdot \mathbf{x}_i) \geq 1 \ \forall \ \mathbf{x}_i.$$

Note that so far we have considered only hyperplanes that pass through the origin. In many applications, this restriction is unnecessary, and the standard separable SVM problem is written as

$$\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||^2 \text{ subject to } y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 \ \forall \ \mathbf{x}_i,$$

where b is a free threshold parameter that translates the optimal hyperplane relative to the origin.

In practice, datasets are often not linearly separable. To deal with this situation, we add slack variables that allow us to violate our original distance constraints. The problem becomes

$$\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{\ell} \xi_i \text{ subject to } y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \ \xi_i \geq 0 \ \forall \ \mathbf{x}_i.$$

This new program trades off the two goals of finding a hyperplane with large margin (minimizing $||\mathbf{w}||$) and finding a hyperplane that separates the data well (minimizing $\sum_{i=1}^{\ell} \xi_i$). The parameter C controls this tradeoff. It is no longer simple to interpret the final solution of the SVM problem geometrically; the latter formulation, which corresponds to the original regularization problem of minimizing $I[f]$, works very well in practice. Even if the data at hand can be separated completely, it may be preferable to use a hyperplane that makes some errors if this results in a much smaller $||\mathbf{w}||$.

In general SVMs can find a nonlinear separating surface. This follows from the use of a general positive definite kernel. Again, there is an obvious geometric interpretation of this classic function approximation technique. The basic idea is to think of the kernel as a way to nonlinearly map the data to a feature space of high or possibly infinite dimension,

$$\mathbf{x} \rightarrow \phi(\mathbf{x}).$$

A linear separating hyperplane in the feature space corresponds to a nonlinear surface in the original space. We can write the program as follows:

$$\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{\ell} \xi_i \text{ subject to } y_i(\mathbf{w} \cdot \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i,\ \xi_i \geq 0\ \forall\ \mathbf{x}_i.$$

Note that as phrased above, $\mathbf{w}$ is a hyperplane in the feature space. In practice, we solve the Wolfe dual of the optimization problems presented and avoid having to work with $\mathbf{w}$ and $\phi(\mathbf{x})$, the hyperplane and the feature vectors, explicitly. In fact, it is well known from the classical theory of integral operators that every positive definite symmetric function of two variables can be represented, under mild conditions, as a dot product of the eigenvectors of the integral operator associated with it. Thus

$$K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j).$$

For example, if we use as our kernel function a Gaussian kernel,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(||\mathbf{x}_i - \mathbf{x}_j||^2).$$

This corresponds to mapping our original vectors $\mathbf{x}_i$ to a certain countably infinite-dimensional feature space when $\mathbf{x}$ is in a bounded domain and an uncountably infinite-dimensional feature space when the domain is not bounded.

Many methods exist for performing feature selection [29, 32, 18, 23]. In informal experiments, all resulted in similar performance. For the feature selection experiments described in this paper, we used RFE [18]. The method recursively removes features based upon the absolute magnitude of the hyperplane elements. Given microarray data with n genes per sample, the SVM outputs a hyperplane, $\mathbf{w}$, which can be thought of as a vector with n components, each corresponding to the expression of a particular gene. Loosely speaking, assuming that the expression values of each gene have similar ranges, the absolute magnitude of each element in $\mathbf{w}$ determines its importance in classifying a sample, since,

$$f(\mathbf{x}) = \sum_{i=1}^{n} \mathbf{w}^i \mathbf{x}^i + b,$$

where $\mathbf{w}^i$ is the ith component of vector $\mathbf{w}$ and $\mathbf{x}^i$ is the ith component of vector $\mathbf{x}$ and the class label is sign$[f(\mathbf{x})]$. The SVM is trained with all genes, the expression values of genes corresponding to $|\mathbf{w}^i|$ in the bottom 10% are removed, and the SVM is retrained with the smaller gene expression set.

The SVM code we used was SvmFu (http://www.ai.mit.edu/projects/cbcl/). The linear or dot-product kernel was used for all reported results:

$$K(\mathbf{w}, \mathbf{v}) = \mathbf{w} \cdot \mathbf{v}.$$

The data was preprocessed by dividing by 5,000 so that the dot product between all samples in the dataset was in the range of $[-1, 1]$. The regularization parameter of the SVM was set to 1, $C = 1$.

For the results reported using RFE, 10% of the features were removed at each iteration.

Acknowledgments. We thank Raphael Bueno, Kevin Loughlin, Scott Pomeroy, Margaret Shipp, and Phil Febbo for contributing tumor samples to this study. We also thank David Waltregny for initial review of pathology and Christine Huard and Michelle Gaasenbeek for expert technical laboratory assistance. We are indebted to members of the Cancer Genomics Group, Whitehead/MIT Center for Genome Research, and the Golub Laboratory, Dana-Farber Cancer Institute for many valuable discussions.

REFERENCES

[1] E. L. Allwein, R. Schapire, and Y. Singer, *Reducing multiclass to binary: A unifying approach for margin classifiers*, J. Mach. Learn. Res., 1 (2001), pp. 113–141.

[2] N. Aronszajn, *Theory of reproducing kernels*, Trans. Amer. Math. Soc., 686 (1950), pp. 337–404.

[3] K. J. Arrow, *Social Choice and Individual Values*, Wiley, New York, 1951.

[4] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and Z. Yakhini, *Tissue classification with gene expression profiles*, J. Comput. Biology, 7 (2000), pp. 559–584.

[5] R. C. Bose and D. K. Ray-Chaudhuri, *On a class of error-correcting binary group codes*, Inform. and Control, 3 (1960), pp. 68–79.

[6] M. P. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. Sugnet, M. Ares, Jr., and D. Haussler, *Support vector machine classification of microarray gene expression data*, Proc. Natl. Acad. Sci. USA, 97 (2000), pp. 262–267.

[7] A. Califano, G. Stolovitzky, and Y. Tu, *Analysis of gene expression microarrays for phenotype classification*, in Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology, San Diego, CA, 1999, pp. 75–85.

[8] *Chipping Forecast*, Nature Genetics, 21 (1999), Supplement.

[9] *Chipping Forecast*, Nature Genetics, 32 (2002), Supplement.

[10] T. Dietterich and S. Bakiri, *Error-correcting output codes: A general method for improving multiclass inductive programs*, in Proceedings of the Ninth AAAI National Conference on Artificial Intelligence, AAAI Press, Menlo Park, CA, 1991, pp. 572–577.

[11] E. Dougherty, *Pattern recognition and gene expression*, SIAM News, May 2002; available online from http://www.siam.org/siamnews/05-02/genes.htm.

[12] D. Duggan, M. Bittner, Y. Chen, P. Meltzer, and J. M. Trent, *Expression profiling using cDNA microarrays*, Nature Genetics Supp., 21 (1999), pp. 10–14.

[13] B. Efron and R. Tibshirani, *Introduction to the Bootstrap*, Chapman and Hall, New York, London, 1983.

[14] T. Evgeniou, M. Pontil, and T. Poggio, *Regularization networks and support vector machines*, Adv. Comput. Math., 13 (2000), pp. 1–50.

[15] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler, *Support vector machine classification and validation of cancer tissue samples using microarray expression data*, Bioinformatics, 16 (2000), pp. 906–914.

[16] 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. Caliguiri, C. D. Bloomfield, and E. S. Lander, *Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring*, Science, 286 (1999), pp. 531–537.

[17] V. Guruswami and A. Sahai, *Multiclass learning, boosting and error-correcting codes*, in Proceedings of the Twelfth Annual Conference on Computational Learning Theory, ACM Press, New York, 1999, pp. 145–155.

[18] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, *Gene selection for cancer classification using support vector machines*, Machine Learning, 46 (2002), pp. 389–422.

[19] T. Hastie and R. Tibshirani, *Classification by pairwise coupling*, in Advances on Neural Processing Systems 10, MIT Press, Cambridge, MA, 1998, pp. 507–513.

[20] R. J. Lipschutz, S. P. A. Fodor, T. R. Gingeras, and D. J. Lockhart, *High density synthetic oligonucleotide arrays*, Nature Genetics Supp., 21 (1999), pp. 20–24.

[21] P. Michaud, *Condorcet: A man of the avant-garde*, Appl. Stoch. Models Data Anal., 3 (1987), pp. 173–189.

[22] M. Minsky and S. Papert, *Perceptrons. An Introduction to Computational Geometry*, MIT Press, Cambridge, MA, 1972.

[23] S. Mukherjee and R. Rifkin, *Support Vector Machine Classification of Microarray Data*, CBCL Paper 182, Artificial Intelligence Lab. Memo 1676, MIT, Cambridge, MA, 1999.

[24] S. Ramaswamy and T. R. Golub, *DNA microarrays in clinical oncology*, J. Clinical Oncology, 20 (2001), pp. 711–719.

[25] S. Ramaswamy, P. Tamayo, R. Rifkin, C. Yeang, M Angelo, C. Ladd, M. Reich, E. Latulippe, J. Mesirov, T. Poggio, W. Gerald, M. Loda, E. Lander, and T. Golub, *A uniform approach to molecular cancer diagnosis using tumor gene expression signatures*, Proc. Natl. Acad. Sci. USA, 98 (2002), pp. 15149–15154.

[26] F. Rosenblatt, *Principles of Neurodynamics*, Spartan Books, New York, 1962.

[27] R. Schapire, Y. Freund, P. Bartlett, and W. S. Lee, *Boosting the margin: A new explanation for the effectiveness of voting methods*, Ann. Statist., 26 (1998), pp. 1651–1686.

[28] R. Schapire, *A brief introduction to boosting*, in Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, Morgan-Kaufmann, San Francisco, 1999, pp. 1401–1406.

[29] D. K. Slonim, P. Tamayo, J. P. Mesirov, T. R. Golub, and E. S. Lander, *Class prediction and discovery using gene expression data*, in Proceedings of the Fourth Annual International Conference on Computational Molecular Biology (RECOMB), Universal Academy Press, Tokyo, Japan, 2000, pp. 263–272.

[30] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*, W. H. Winston, Washington, DC, 1977.

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

[32] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik, *Feature selection for SVMs*, in Advances on Neural Processing Systems 13, MIT Press, Cambridge, MA, 2001, pp. 668–674.

[33] C. Yeang, S. Ramaswamy, P. Tamayo, R. Rifkin, M. Angelo, M. Reich, E. Lander, J. Mesirov, and T. Golub, *Molecular classification of multiple tumor types*, Bioinformatics Discovery Note, 1 (2001), pp. 1–7.