Support Vector Machine Classification and Validation of Cancer Tissue Samples Using Microarray Expression Data

Article *in* Bioinformatics · February 2001

DOI: 10.1093/bioinformatics/16.10.906 · Source: CiteSeer

4 authors, including:

Terrence S Furey
University of North Carolina at Chapel Hill
244 PUBLICATIONS **78,460** CITATIONS

SEE PROFILE

David Haussler
University of California, Santa Cruz
420 PUBLICATIONS **106,773** CITATIONS

SEE PROFILE

BIOINFORMATICS

Vol. 16 no. 10 2000
Pages 906–914

Support vector machine classification and validation of cancer tissue samples using microarray expression data

Terrence S. Furey[1,*], Nello Cristianini[2], Nigel Duffy[1], David W. Bednarski[3], Michèl Schummer[3] and David Haussler[1]

[1]Department of Computer Science, University of California, Santa Cruz, Santa Cruz, CA 95064, USA, [2]Department of Engineering Mathematics, University of Bristol, Bristol, BS8 ITH, UK and [3]Department of Molecular Biotechnology, University of Washington, Seattle, WA 98195, USA

Received on April 4, 2000; accepted on May 19, 2000

Abstract

Motivation: *DNA microarray experiments generating thousands of gene expression measurements, are being used to gather information from tissue and cell samples regarding gene expression differences that will be useful in diagnosing disease. We have developed a new method to analyse this kind of data using support vector machines (SVMs). This analysis consists of both classification of the tissue samples, and an exploration of the data for mis-labeled or questionable tissue results.*

Results: *We demonstrate the method in detail on samples consisting of ovarian cancer tissues, normal ovarian tissues, and other normal tissues. The dataset consists of expression experiment results for 97 802 cDNAs for each tissue. As a result of computational analysis, a tissue sample is discovered and confirmed to be wrongly labeled. Upon correction of this mistake and the removal of an outlier, perfect classification of tissues is achieved, but not with high confidence. We identify and analyse a subset of genes from the ovarian dataset whose expression is highly differentiated between the types of tissues. To show robustness of the SVM method, two previously published datasets from other types of tissues or cells are analysed. The results are comparable to those previously obtained. We show that other machine learning methods also perform comparably to the SVM on many of those datasets.*

Availability: *The SVM software is available at http://www.cs.columbia.edu/~bgrundy/svm.*

Contact: *booch@cse.ucsc.edu*

Introduction

Microarray expression experiments allow the recording of expression levels of thousands of genes simultaneously. These experiments primarily consist of either monitoring each gene multiple times under many conditions (Spellman *et al.*, 1998; Chu *et al.*, 1998; DeRisi *et al.*, 1997; Wen *et al.*, 1998; Roberts *et al.*, 2000), or alternately evaluating each gene in a single environment but in different types of tissues, especially cancerous tissues (DeRisi *et al.*, 1996; Alon *et al.*, 1999; Golub *et al.*, 1999; Perou *et al.*, 1999; Zhu *et al.*, 1998; Wang *et al.*, 1999; Schummer *et al.*, 1999; Zhang *et al.*, 1997; Slonim *et al.*, 2000). Those of the first type have allowed for the identification of functionally related genes due to common expression patterns (Brown *et al.*, 2000; Eisen *et al.*, 1998; Wen *et al.*, 1998; Roberts *et al.*, 2000), while the latter experiments have shown promise in classifying tissue types (diagnosis) and in the identification of genes whose expressions are good diagnostic indicators (Golub *et al.*, 1999; Alon *et al.*, 1999; Slonim *et al.*, 2000). In order to extract information from gene expression measurements, different methods have been employed to analyse this data including SVMs (Brown *et al.*, 2000; Mukherjee *et al.*, 1999) clustering methods (Eisen *et al.*, 1998; Spellman *et al.*, 1998; Alon *et al.*, 1999; Perou *et al.*, 1999; Ben-Dor *et al.*, 2000; Hastie *et al.*, 2000), self-organizing maps (Tamayo *et al.*, 1999; Golub *et al.*, 1999), and a weighted correlation method (Golub *et al.*, 1999; Slonim *et al.*, 2000).

Support vector machines (SVMs), a supervised machine learning technique, have been shown to perform well in multiple areas of biological analysis including evaluating microarray expression data (Brown *et al.*, 2000), detecting remote protein homologies (Jaakkola *et al.*, 1999), and recognizing translation initiation sites (Zien *et al.*, 2000). We have also recently become aware of another effort that uses SVMs in analyzing expression data (Mukherjee *et al.*, 1999). SVMs have demonstrated the ability to not only correctly separate entities into appropriate classes, but also to identify instances whose established classification is not

*To whom correspondence should be addressed.

supported by the data. Expression datasets contain measurements for thousands of genes which proves problematic for many traditional methods. SVMs, though, are well suited to working with high dimensional data such as this.

Here a systematic and principled method is introduced that analyses microarray expression data from thousands of genes tested in multiple tissue or cell samples. The primary goal is the proper classification of new samples. We do this by training the SVM on samples classified by experts, then testing the SVM on samples it has not seen before. We demonstrate how SVMs can not only classify new samples, but can also help in the identification of those which have been wrongly classified by experts. SVMs are not unique among classification methods in this regard, but we show they are effective. Our method is demonstrated in detail on data from experiments involving 31 ovarian cancer, normal ovarian and other normal tissues. We are able to identify one tissue sample as mislabeled, and another as an outlier, which is shown in the *Results* Section and illustrated in Figure 1. Though perfect classification is finally achieved in one instance, this performance is not consistently shown in multiple tests and, therefore, cannot be considered too significant.

We also experimented with the method introduced in (Golub *et al.*, 1999) to focus the analysis on a smaller subset of genes that appear to be the best diagnostic indicators. This amounts to a kind of dimensionality reduction on the dataset. If one can identify particular genes that are diagnostic for the classification one is trying to make, e.g. the presence of cancer, then there is also hope that some of these genes may be found to be of value in further investigations of the disease and in future therapies. Here we find that this dimensionality reduction does not significantly improve classification performance. It does reveal some genes that may be of interest in ovarian cancer. However, further work needs to be carried out to identify the most effective feature selection/dimensionality reduction methods for this kind of data.

To test the generality of the approach, we also ran experiments using leukemia data from Golub *et al.* (1999) (72 patient samples) and colon tumor data from (Alon *et al.*, 1999) (62 tissue samples). Our results are comparable to other methods used by the authors of those papers. Since no special effort was made to tune the method to these other datasets, this increases our confidence that our approach will have broad applications in analyzing data of this type.

It is difficult to show that one diagnostic method is significantly better than another with small data sets such as those we have examined. We have conducted a full hold-one-out cross-validation (jackknife) evaluation of the classification performance of the methods we tested. These include both SVM methods and variants of the perceptron algorithm. No single classification technique has proven to be significantly superior to all others in the experiments we have performed. Indeed, the different kernels we tried performed nearly equally well and variations of the perceptron algorithm are shown to perform comparably to the SVM on all tests. It is unfortunate that typical diagnostic gene expression datasets today involve only a few tissue samples. As more datasets become available with larger numbers of samples, we predict that our method will continue to demonstrate good performance.

Methods

In recent years, several methods have been developed for performing gene expression experiments. Measurements from these experiments can give expression levels for genes (or ESTs) in tissue or cell samples. For more in depth discussions of these techniques, see Lockhart *et al.* (1996) and Schummer *et al.* (1999). Datasets used for our experiments consist of a relatively small number of tissue samples (less than 100) each with expression measurements for thousands of genes.

Previous methods used in the analysis of similar datasets start with a procedure to extract the most relevant features. Most learning techniques do not perform well on datasets where the number of features is large compared to the number of examples. SVMs are believed to be an exception. We are able to begin with tests using the full dataset, and systematically reduce the number of features selecting those we believe to be the most relevant. In this way, we can show whether an improvement is made using smaller sets, thus indicating whether these contain the most meaningful genes.

To understand our method, a familiarity with SVMs is required, and a brief introduction follows. We explain below how we rank the features, and present an outline of how we use the SVM to perform classification and error detection.

Support vector machines

SVMs (Cristianini and Shawe-Taylor, 2000) are a relatively new type of learning algorithm, originally introduced by Vapnik and co-workers (Boser *et al.*, 1992; Vapnik, 1998) and successively extended by a number of other researchers. Their remarkably robust performance with respect to sparse and noisy data is making them the system of choice in a number of applications from text categorization to protein function prediction.

When used for classification, they separate a given set of binary labeled training data with a hyper-plane that is maximally distant from them (known as 'the maximal margin hyper-plane'). For cases in which no linear separation is possible, they can work in combination with the technique of 'kernels', that automatically realizes

a non-linear mapping to a feature space. The hyper-plane found by the SVM in feature space corresponds to a non-linear decision boundary in the input space.

Let the jth input point $\mathbf{x}^j = (x_1^j, \ldots, x_n^j)$ be the realization of the random vector $\mathbf{X}^j$. Let this input point be labeled by the random variable $Y^j \in \{-1, +1\}$.

Let $\phi : I \subseteq \Re^n \rightarrow F \subseteq \Re^N$ be a mapping from the input space $I \subseteq \Re^n$ to a feature space F. Let us assume that we have a sample S of m labeled data points: $S = \{(\mathbf{x}^1, y^1), \ldots, (\mathbf{x}^m, y^m)\}$. The SVM learning algorithm finds a hyper-plane $(\mathbf{w}, b)$ such that the quantity

$$\gamma = \min_i y^i \{\langle \mathbf{w}, \phi(\mathbf{x}^i)\rangle - b\} \tag{1}$$

is maximized, where $\langle , \rangle$ denotes an inner product, the vector $\mathbf{w}$ has the same dimensionality as F, $||\mathbf{w}||_2$ is held constant, b is a real number, and γ is called the *margin*. The quantity $(\langle \mathbf{w}, \phi(\mathbf{x}^i)\rangle - b)$ corresponds to the distance between the point $\mathbf{x}^i$ and the decision boundary. When multiplied by the label y^i, it gives a positive value for all correct classifications and a negative value for the incorrect ones. The minimum of this quantity over all the data is positive if the data is linearly separable, and is called the margin. Given a new data point $\mathbf{x}$ to classify, a label is assigned according to its relationship to the decision boundary, and the corresponding decision function is

$$f(\mathbf{x}) = \text{sign}(\langle \mathbf{w}, \phi(\mathbf{x})\rangle - b). \tag{2}$$

It is easy to prove (Cristianini and Shawe-Taylor, 2000) that, for the maximal margin hyper-plane,

$$\mathbf{w} = \sum_{i=1}^{m} \alpha_i y^i \phi(\mathbf{x}^i) \tag{3}$$

where α_i are positive real numbers that maximize

$$\sum_{i=1}^{m} \alpha_i - \sum_{ij=1}^{m} \alpha_i \alpha_j y^i y^j \langle \phi(\mathbf{x}^i), \phi(\mathbf{x}^j)\rangle \tag{4}$$

subject to

$$\sum_{i=1}^{m} \alpha_i y^i = 0, \alpha_i > 0, \tag{5}$$

the decision function can equivalently be expressed as

$$f(\mathbf{x}) = \text{sign}\left(\sum_{i=1}^{m} \alpha_i y_i \langle \phi(\mathbf{x}^i), \phi(\mathbf{x})\rangle - b\right). \tag{6}$$

From this equation it is possible to see that the α_i associated with the training point $\mathbf{x}^i$ expresses the strength with which that point is embedded in the final decision function. A remarkable property of this alternative representation is that often only a subset of the points will be associated with non-zero α_i. These points are called *support vectors* and are the points that lie closest to the separating hyper-plane. The sparseness of the α vector has several computational and learning theoretic consequences.

Notice that for a test point $(\mathbf{x}, y)$ the quantity $y\left(\sum_{i=1}^{m} \alpha_i y_i \langle \phi(\mathbf{x}^i), \phi(\mathbf{x})\rangle - b\right)$ is negative if the prediction of the machine is wrong, and a large negative value would indicate that the point $(\mathbf{x}, y)$ is regarded by the algorithm as 'different' from the training data.

The matrix $K_{ij} = \langle \phi(\mathbf{x}^i), \phi(\mathbf{x}^j)\rangle$ is called the *kernel matrix* and will be particularly important in the extensions of the algorithm that will be discussed later. In the case when the data are not linearly separable, one can use more general functions, $K_{ij} = K(\mathbf{x}^i, \mathbf{x}^j)$, that provide non-linear decision boundaries. Two classical choices are polynomial kernels $K(\mathbf{x}^i, \mathbf{x}^j) = (\langle \mathbf{x}^i, \mathbf{x}^j\rangle + 1)^d$ and Gaussian kernels $K(\mathbf{x}^i, \mathbf{x}^j) = \mathrm{e}^{-\frac{\|\mathbf{x}^i - \mathbf{x}^j\|}{\sigma^2}}$, where d and σ are kernel parameters. In our experiments, we use $K(\mathbf{x}^i, \mathbf{x}^j) = (\langle \mathbf{x}^i, \mathbf{x}^j\rangle + 1)$.

In the presence of noise, the standard maximum margin algorithm described above can be subject to over-fitting, and more sophisticated techniques are necessary. This problem arises because the maximum margin algorithm always finds a perfectly consistent hypothesis and does not tolerate training error. Sometimes, however, it is necessary to trade some training accuracy for better predictive power. The need for tolerating training error has led to the development of the soft-margin and the margin-distribution classifiers (Cortes and Vapnik, 1995). One of these techniques (Shawe-Taylor and Cristianini, 1999) replaces the kernel matrix in the training phase as follows:

$$K \leftarrow K + \lambda \mathbf{1}, \tag{7}$$

while still using the standard kernel function in the decision phase (6). We call λ the diagonal factor. By tuning λ, one can control the training error, and it is possible to prove that the risk of misclassifying unseen points can be decreased with a suitable choice of λ (Shawe-Taylor and Cristianini, 1999).

If instead of controlling the overall training error one wants to control the trade-off between false positives and false negatives, it is possible to modify K as follows:

$$K \leftarrow K + \lambda D, \tag{8}$$

where D is a diagonal matrix whose entries are either d^+ or d^-, in locations corresponding to positive and negative examples. It is possible to prove that this technique is equivalent to controlling the size of the α_i in a way that depends on the size of the class, introducing a

bias for larger α_i in the class with smaller d. This in turn corresponds to an asymmetric margin; i.e. the class with smaller d will be kept further away from the decision boundary (Brown *et al.*, 2000). In the case of imbalanced data sets, choosing $d^+ = \frac{1}{n^+}$ and $d^- = \frac{1}{n^-}$ provides a heuristic way to automatically adjust the relative importance of the two classes, based on their respective cardinalities.

The experiments presented in this paper were performed using a freely available implementation of the SVM classifier which can be obtained at http://www.cs.columbia.edu/~bgrundy/svm.[†] This implementation is based on that described in Jaakkola *et al.* (1999) and differs slightly from the above explanation in that it does not include a bias term, b, forcing all decision boundaries to contain the origin in feature space.

Feature selection

Our feature selection criterion is essentially that used in Golub *et al.* (1999) and Slonim *et al.* (2000). We start with a dataset S consisting of m expression vectors $\mathbf{x^i} = (x_1^i, \ldots, x_n^i)$, $1 \leq i \leq m$, where m is the number of tissue or cell samples and n is the number of genes measured. Each sample is labeled with $Y \in \{+1, -1\}$ (e.g. cancer vs normal). For each gene x_j, we calculate the mean μ_j^+ (resp. μ_j^-) and standard deviation σ_i^+ (resp. σ_i^- using only the tissues labeled $+1$ (resp. -1). We want to find genes that will help discriminate between the two classes, therefore we calculate a score[‡]

$$F(x_j) = \left| \frac{\mu_j^+ - \mu_j^-}{\sigma_J^+ + \sigma_j^-} \right| \tag{9}$$

which gives the highest score to those genes whose expression levels differ most on average in the two classes while also favoring those with small deviations in scores in the respective classes. We then simply take the genes with the highest $F(x_j)$ scores as our top features.

Complete SVM method

The complete SVM method can be described as follows: we begin by choosing a kernel, starting with the simple dot-product kernel, and tune the diagonal factor to achieve the best performance on hold-one-out cross-validation tests using the full dataset. The SVM tuning procedure is then repeated with a specified number of the top-ranked features. In these cases, for each individual hold-one-out test, the features are ranked based on (9) using the scores from only the known samples, some number of the top features are extracted, and then these are then used to train the SVM and classify the unknown sample. Examples which have been consistently misclassified in all tests are identified. These examples can then be investigated by the biologist, and if it is determined that the original label is incorrect, a correction is made, and the process is repeated. Alternatively, an example may be deemed an outlier that is very different from the rest, and is therefore removed.

In the SVM tests reported here, only the simple dot-product kernel is used.[§] A more complex kernel is not required. As possibly more complex datasets become available providing more examples, higher-order kernels may become necessary (Mukherjee *et al.*, 1999).

Results

Our method is tested in detail using a previously unpublished ovarian tissue dataset. A short analysis of the feature selection is included. To demonstrate the generality of our method, we also performed experiments using previously published datasets. The first dataset contains examples of patients with human acute leukemia, originally analysed by Golub *et al.* (1999) with further results reported by Slonim *et al.* (2000). The dataset can be obtained at http://waldo.wi.mit.edu/MPR/cancer_class.html. The second dataset is comprised of human tumor and normal colon tissues. Alon *et al.* (1999), originally analysed this data which is available on their website, http://www.molbio.princeton.edu/colondata.

Ovarian dataset

Microarray expression experiments are performed using 97 802 DNA clones, each of which may or may not correspond to human genes, for 31 tissue samples. These samples are either cancerous ovarian tissue, normal ovarian tissue, or normal non-ovarian tissue. For the purpose of these experiments, the two types of normal tissue are considered together as a single class. The expression values for each of the genes are normalized such that the distribution over the samples had a zero mean and unit variance.

Hold-one-out cross-validation experiments are performed. The SVM is trained using data from all but one of the tissue samples. The sample not used in training is then assigned a class by the SVM. A single SVM experiment consists of a series of hold-one-out experiments, each sample being held out and tested exactly once.

Initially, experiments are carried out using all expression scores with diagonal factor settings of 0, 2, 5 and 10. The genes are then ranked in the manner described previously, and datasets consisting of the top 25, 50, 100, 500

[†] We use default values set in the software except for the diagonal factor, which varies, the convergence threshold, which we set to 10^{-11}, and using the 'noconstraint' option.

[‡] This score is closely related to the Fisher criterion score for the jth feature, $F(j) = (\mu_j^+ - \mu_j^-)^2/((\sigma_j^+)^2 + (\sigma_j^-)^2)$ (Bishop, 1995).

[§] We experimented with polynomial and radial basis kernels on the ovarian data, and found that on data containing the mis-labeled point, they performed worse than the linear kernel, but on the correctly labeled data, performance is similar to the linear kernel.

Table 1. Error rates for ovarian cancer tissue experiments

Kernel	DF	Feature	FP	FN	TP	TN
Dot-product	0	25	5	4	10	12
Dot-product	2	25	5	2	12	12
Dot-product	5	25	4	2	12	13
Dot-product	10	25	4	2	12	13
Dot-product	0	50	4	2	12	13
Dot-product	2	50	3	2	12	14
Dot-product	5	50	3	2	12	14
Dot-product	10	50	3	2	12	14
Dot-product	0	100	4	3	11	13
Dot-product	2	100	5	3	11	12
Dot-product	5	100	5	3	11	12
Dot-product	10	100	5	3	11	12
Dot-product	0	97 802	17	0	14	0
Dot-product	2	97 802	9	2	12	8
Dot-product	5	97 802	7	3	11	10
Dot-product	10	97 802	5	3	11	12

For each setting of the SVM consisting of a kernel and diagonal factor (DF), each tissue was classified. Column 2 is the number of features (clones) used. Reported are the number of normal tissues misclassified (FP), tumor tissues misclassified (FN), tumor tissues classified correctly (TP), and normal tissues classified correctly (TN).

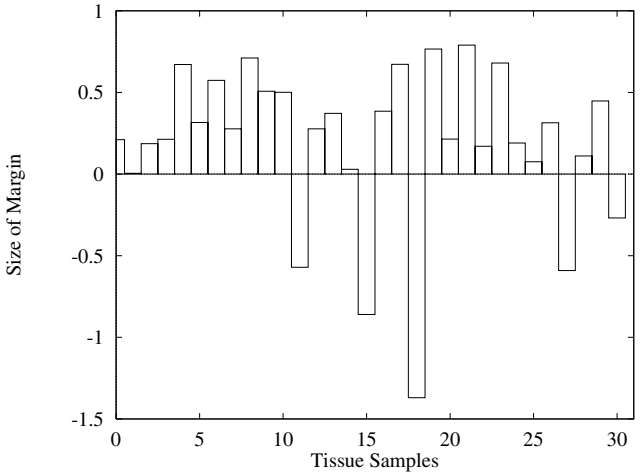

Fig. 1. SVM classification margins for ovarian tissues. When classifying, the SVM calculates a margin which is the distance of an example from the decision boundary it has learned. In this graph, the margin for each tissue sample calculated using (6) is shown. A positive value indicates a correct classification, and a negative value indicates an incorrect classification. The most negative point corresponds to tissue N039. The second most negative point corresponds to tissue HWBC3.

and 1000 features are created. Experiments using similar diagonal factors to those above are performed using these smaller feature sets. Table 1 displays the most significant results from these experiments. The best classification is done using the top 50 features with a diagonal factor of 2 or 5. Though the smaller datasets achieve slightly better scores compared to using all features, we do not believe this improvement to be significant.

An analysis of the misclassified examples reveals that one normal ovarian tissue sample, N039, is misclassified in all instances. In addition, the margin of misclassification, calculated using (6), is relatively large meaning the SVM strongly believes it to be cancerous. Figure 1 shows classification margins for experiments using the top 50 features and a diagonal factor of 2. Upon investigation, it is discovered that this tissue had been mistakenly labeled and is, in fact, cancerous.

With a corrected label, the above experiments are run again, but disappointingly, classification results do not improve. A second tissue, called HWBC3, is consistently misclassified by a large margin in these new tests, and was also strongly misclassified in the original tests, as shown in Figure 1. This non-ovarian normal tissue is the only tissue of its type, and an SVM trained on tissues with little similarity might give spurious classification results. Therefore, we remove this tissue and repeat the experiments. Perfect classification is achieved using all features and a diagonal factor of 0. No other setting is able to make fewer than three mistakes and most make at least four, therefore we can not place much confidence in one perfect experiment.

After ranking the features using all 31 samples, we attempt to sequence the ten top-ranked genes to determine if they are biologically significant. Three of these did not yield a readable sequence, and two are repetitive sequences which occur naturally at 3' ends of messenger RNAs and do not correspond to actual genes. Therefore, only five represent expressed genes for which cancer-relatedness information is thus available, either by its homology to a known or assumed tumor gene, or its presence in cDNA libraries from tumor tissues in the case of ESTs. Indeed, three of these five are cancer-related (Ferritin H and two cancer-library ESTs), and one is related to the presence of white blood cells in the tumor. This analysis seems to suggest that the feature selection method is able to identify clones that are cancer-related, and rank them highly. Some clones however, obtain a high ranking while not having a meaningful biological explanation. Random sequencing of some of the bottom-ranked clones also reveal some known tumor genes which would be expected to be ranked highly. Given this and the inability of this feature selection method to significantly improve classification performance, we conclude that additional effort is needed to develop ways of identifying meaningful features in these types of datasets. From a

tumor biologist's point of view however, the accumulation of tumor-related genes at the top is a very useful feature.

AML/ALL dataset

Bone marrow or peripheral blood samples are taken from 72 patients with either acute myeloid leukemia (AML) or acute lymphoblastic leukemia (ALL). Following the experimental setup of the original authors, the data is split into a training set consisting of 38 samples of which 27 are ALL and 11 are AML, and a test set of 24 samples, 20 ALL and 14 AML. The dataset provided contains expression levels for 7129 human genes produced by Affymetrix high-density oligonucleotide microarrays. The scores in the dataset represent the intensity of gene expression after being re-scaled to make overall intensities for each chip equivalent. Following the methods in Golub *et al.* (1999), we normalize these scores by subtracting the mean and dividing by the standard deviation of the expression values for each gene.

Golub *et al.* perform hold-one-out cross validation tests using a weighted voting scheme to classify the training set[¶] and also cluster this set using self-organizing maps (SOMs). The first method correctly classifies all samples for which a prediction is made, 36 of the 38 samples, while a two-cluster SOM produces one cluster with 24 ALL and one AML sample, and the second with 10 AML and three ALL samples.

We also did a full hold-one-out cross-validation tests on the training set, and our SVM method correctly classifies all samples with a diagonal factor setting of two. Retesting subsets containing the top-ranked 25, 250, 500, and 1000 features, perfect classification is obtained using a diagonal factor of two in all cases.

Using an SVM trained only with examples in the training set and the subsets of features that perform optimally on this training set, we classify examples in the test set producing results ranging between classifying 30 to 32 of the 34 samples correctly. Golub *et al.* use a predictor trained using their weighted voting scheme on the training samples, and classify correctly on all samples for which a prediction is made, 29 of the 34, declining to predict for the other five. In all tests, our SVM correctly classifies the 29 predicted by their method, and for the five unpredicted samples, each is misclassified in at least one SVM test. Two samples, patients 54 and 66, are misclassified in all SVM tests.

Lineage information, either T-cell or B-cell, is provided for the ALL samples. Using all 47 ALL samples from the training and test sets, the SVM achieves perfect classification using the 250 and 500 top-ranked features with multiple diagonal factor settings on hold-one-out cross-validation tests. Using the full dataset, the SVM misclassifies a single tissue using a zero diagonal factor. Golub *et al.* uses SOMs to create four clusters containing all training set examples, including the AML samples. The first cluster contains 10 AML samples, the second eight T-lineage ALL samples and one B-lineage ALL sample, the third five B-lineage ALL samples, and the last one 13 B-lineage ALL samples and a single AML sample. Additional tests in Slonim *et al.* (2000) use the weighted voting predictor to classify 33 samples of which it predicted on 32, all being correct.

Lastly, the success of chemotherapy treatments for 15 of the AML patients is available. Slonim *et al.* report that they were able to create a predictor which made only two mistakes using a single gene, HOXA9, but that other predictors using more than this gene generally had error rates above 30%. On hold-one-out cross-validation tests, the SVM is able to classify 10 of the 15 patients using the top 5 or 10 ranked features and a diagonal factor of two, thus performing only slightly better than chance. One misclassified sample, patient 37, is consistently misclassified by a relatively large margin.

Colon tumor dataset

Using Affymetrix oligonucleotide arrays, expression levels for 40 tumor and 22 normal colon tissues are measured for 6500 human genes. Of these genes, the 2000 with the highest minimal intensity across the tissues are selected for classification purposes and these scores are publicly available. Each score represents a gene intensity derived in a process described in Alon *et al.* (1999). The data is not processed further before performing classification. Alon *et al.* use a clustering method to create clusters of tissues. In their experiments, one cluster consists of 35 tumor and three normal tissues, and the other 19 normal and five tumor tissues.

Using the SVM method with full hold-one-out cross-validation, we classify correctly all but six tissues using all 2000 features and a diagonal factor of two. Using the top 1000 genes, the SVM misclassifies these same six samples which correspond to three tumor tissues (T30, T33, T36) and three normal tissues (N8, N34, N36). T30, T33, and T36 are among the five tumor tissues in the Alon *et al.* cluster with a majority of normal tissues, and N8 and N32 are in the cluster containing a majority of the tumor tissues.

Alon *et al.* define a muscle index based on the average intensity of ESTs that are homologous to 17 smooth muscle genes, and hypothesize that tumor tissues should have a smaller muscle index. In general, this proves correct with the notable exceptions that all tumor tissues have a muscle index less than or equal to 0.3 except for T30, T33,

[¶] The weighted voting scheme selects 50 genes as described in the subsection 'Feature selection'. Each gene predicts a class for each sample. These predictions are combined, each being weighted by the $F(g)$ score defined above, and if a threshold is exceeded in favor of one class over the other, a prediction is made.

and T36, and all normal tissues have an index greater than or equal to 0.3 except N8, N34, and N36. Two samples, N36 and T36, are especially interesting because their names indicate that they originate from the same patient, both are consistently misclassified by the SVM, and N36 has a muscle index or 0.1 and T36 has a muscle index of 0.7, contrary to the proposed hypothesis.

Comparison to perceptron-like classification algorithms

As discussed in the introduction, we do not claim that we can prove the superiority of the SVM method over other classification techniques on this type of dataset. The second family of algorithms we test are generalizations of the perceptron algorithm (Rosenblatt, 1958). This simple algorithm considers each sample individually, and updates its weight vector each time it makes a mistake according to

$$\mathbf{w}^{i+1} = \mathbf{w}^i + y^i \mathbf{x}^i. \tag{10}$$

The resulting decision rule is linear (no bias is used), and classification is given by $\text{sign}(\langle \mathbf{w}^i, \mathbf{x} \rangle)$. However, this algorithm requires modification when there is no perfect linear decision rule. Helmbold and Warmuth (1995) show that taking a linear combination of the decision rules used at each iteration of the algorithm is sufficient, and are able to derive performance guarantees. The final decision rule is $\text{sign}\left(\sum_t \langle \mathbf{w}^i, \mathbf{x} \rangle\right)$. Results for this modified perceptron are comparable to those for the SVM, and scores using all features in each dataset are given in Table 2.

Freund and Schapire (1998) demonstrate that kernels other than the simple inner product can be applied effectively to this algorithm, achieving performance comparable to the best SVM on a benchmark test of Hand-Written Digits. As in the case of SVMs, though, the use of a more complex kernel did not improve performance.

We also test an algorithm known as the p-norm perceptron (Grove *et al.*, 1997), using the same averaging procedure. Theoretical results suggest that these algorithms will perform well when good sparse hypotheses are available. The p-norm perceptron, though, did not perform as well as the theory might suggest (results not shown).

Table 2. Results for the perceptron using all features

Dataset	Features	FP	FN	SVM FP	SVM FN
Ovarian I	97 802	4.6	4.8	5	3
Ovarian II	97 802	4.4	3.4	0	0
AML/ALL train	7 129	0.6	2.8	0	0
AML treatment	7 129	4.8	3.5	3	6
Colon	2 000	3.8	3.7	3	3

Results are averaged over five shufflings of the data as this algorithm is sensitive to the order of the samples. The first column is the dataset and the second is the number of features considered. Ovarian I refers to the original full dataset with the incorrectly labeled N039 tissue, while Ovarian II is the dataset with the label corrected and the HWBC3 tissue removed. The ovarian and colon datasets show the number of normal tissues misclassified (FP) and the number of tumor tissues misclassified (FN). The AML/ALL training dataset report the number of AML samples misclassified (FP) and the number of ALL patients misclassified (FN). The AML treatment dataset shows the number of unsuccessfully treated patients misclassified (FP) and the number of successfully treated patients misclassified (FN). The last two columns report the corresponding SVM score using all features.

Conclusion

We have presented a method to analyse microarray expression data for genes from several tissue or cell types using SVMs. While our results indicate that SVMs are able to classify tissue and cell types based on this data, we show that other methods such as the ones based on the perceptron algorithm are able to perform similarly. The datasets currently available contain relatively few examples and thus do not allow one method to demonstrate superiority. The SVM performs well using a simple kernel, and we believe that as datasets containing more examples become available, the use of more complex kernels may become necessary and will allow the SVM to continue its good performance. As an added feature of our SVM method, we demonstrate that it can be used to identify mis-labeled data.

Microarray expression experiments have great potential for use as part of standard diagnosis tests performed in the medical community. We have shown along with others that expression data can be used in the identification of the presence of a disease and the determination of its cell lineage. In addition, there is a hope that predictions of the success or failure of a particular treatment may be possible, but so far, results from these types of experiments are inconclusive.

Acknowledgements

We used SVM software written by Bill Grundy and thank him for his assistance and for comments on an earlier draft. We are particularly grateful to Tomaso Poggio for pointing out a flaw in our method in earlier experiments. We thank Manuel Ares for suggesting we look at the Alon *et al.* data, and Dick Karp for putting us in contact with each other to study the ovarian cancer data. Finally, we are grateful to Al Globus, Computer Sciences Corporation at NASA Ames Research Center, for providing some of the computational resources required to perform our experiments.

References

Alon,U., Barkai,N., Notterman,D.A., Gish,K., Mack,S.Y.D. and Levine,J. (1999) Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA*, **96**, 6745–6750.

Ben-Dor,A., Bruhn,L., Friedman,N., Nachman,I., Schummer,M. and Yakhini,Z. (2000) Tissue classification with gene expression profiles. In *Proceedings of the 4th Annual International Conference on Computational Molecular Biology (RECOMB)* Universal Academy Press, Tokyo.

Bishop,C. (1995) *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford.

Boser,B.E., Guyon,I.M. and Vapnik,V.N. (1992) A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory* ACM Press, Pittsburgh, PA, pp. 144–152.

Brown,M., Grundy,W., Lin,D., Cristianini,N., Sugnet,C., Furey,T., M. Ares,J. and Haussler,D. (2000) Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc. Natl. Acad. Sci. USA*, **97**, 262–267.

Chu,S., DeRisi,J., Eisen,M., Mulholland,J., Botstein,D., Brown,P. and Herskowitz,I. (1998) The transcriptional program of sporulation in budding yeast. *Science*, **282**, 699–705.

Cortes,C. and Vapnik,V. (1995) Support-vector networks. *Machine Learning*, **20**, 273–297.

Cristianini,N. and Shawe-Taylor,J. (2000) *An Introduction to Support Vector Machines*. Cambridge University Press, Cambridge, www.support-vector.net.

DeRisi,J., Iyer,V. and Brown,P. (1997) Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, **278**, 680–686.

DeRisi,J., Penland,L., Brown,P., Bittner,M., Meltzer,P., Ray,M., Chen,Y., Su,Y. and Trent,J. (1996) Use of a cdna microarray to analyse gene expression patterns in human cancer. *Nat. Genet.*, **4**, 457–460.

Eisen,M., Spellman,P., Brown,P. and Botstein,D. (1998) Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA*, **95**, 14863–14868.

Freund,Y. and Schapire,R.E. (1998) Large margin classification using the perceptron algorithm. In *Proceedings of the 11th Annual Conference on Computational Learning Theory* ACM Press, New York, pp. 209–217.

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

Grove,A.J., Littlestone,N. and Schuurmans,D. (1997) General convergence results for linear discriminant updates. In *Proceedings of the 10th Annual Conference on Computational Learning Theory* ACM Press, New York, pp. 171–183.

Hastie,T., Tibshirani,R., Eisen,M., Brown,P., Ross,D., Scherf,U., Weinstein,J., Alizadeh,A., Staudt,L. and Botstein,D. (2000) Gene Shaving: a new class of clustering methods for expression arrays. *Stanford University Technical report*.

Helmbold,D. and Warmuth,M.K. (1995) On weak learning. *J. Comput. Syst. Sci.*, **50**, 551–573.

Jaakkola,T., Diekhans,M. and Haussler,D. (1999) Using the Fisher kernel method to detect remote protein homologies. In *Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology* AAAI Press, Menlo Park, CA.

Lockhart,D., Dong,H., Byrne,M., Follettie,M., Gallo,M., Chee,M., Mittmann,M., Wang,C., Kobayashi,M., Horton,H. and Brown,E. (1996) Expression monitoring by hybridization to high–density oligonucleotide arrays. *Nature Biotechnol.*, **14**, 1675–1680.

Mukherjee,S., Tamayo,P., Mesirov,J., Slonim,D., Verri,A. and Poggio,T. (1999) Support vector machine classification of microarray data. *Technical Report CBCL Paper 182/AI Memo 1676 MIT*.

Perou,C., Jeffrey,S., van de Rijn,M., Rees,C., Eisen,M., Ross,D., Pergamenschikov,A., Williams,C., Zhu,S., Lee,J., Lashkari,D., Shalon,D., Brown,P. and Botstein,D. (1999) Distinctive gene expression patterns in human mammary epithelial cells and breast cancers. *Proc. Natl. Acad. Sci. USA*, **96**, 9212–9217.

Roberts,C., Nelson,B., Marton,M., Stoughton,R., Meyer,M., Bennett,H., He,Y., Dai,H., Walker,W., Hughes,T., Tyers,M., Boone,C. and Friend,S. (2000) Signaling and circuitry of multiple mapk pathways revealed by a matrix of global gene expression profiles. *Science*, **287**, 873–880.

Rosenblatt,F. (1958) The perceptron: a probabilistic model for information storage and organization in the brain. *Psych. Rev.*, **65**, 386–407.

Schummer,M., Ng,W., Bumgarner,R., Nelson,P., Schummer,B., Bednarski,D., Hassell,L., Baldwin,R., Karlan,B. and Hood,L. (1999) Comparative hybridization of an array of 21 500 ovarian cDNAs for the discovery of genes overexpressed in ovarian carcinomas. *Gene*, **238**, 375–385.

Shawe-Taylor,J. and Cristianini,N. (1999) Further results on the margin distribution. In *Proceedings of the 12th Annual Conference on Computational Learning Theory* ACM Press, New York.

Slonim,D., Tamayo,P., Mesirov,J., Golub,T. and Lander,E. (2000) Class prediction and discovery using gene expression data. In *Proceedings of the 4th Annual International Conference on Computational Molecular Biology (RECOMB)* Universal Academy Press, Tokyo, Japan, pp. 263–272.

Spellman,P., Sherlock,G., Zhang,M., Iyer,V., Anders,K., Eisen,M., Brown,P., Botstein,D. and Futcher,B. (1998) Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell*, **9**, 273–3297.

Tamayo,P., Slonim,D., Mesirov,J., Zhu,Q., Kitareewan,S., Dmitrovsky,E., Lander,E. and Golub,T. (1999) Interpreting patterns of gene expression with self-organizing maps. *Proc. Natl. Acad. Sci. USA*, **96**, 2907–2912.

Vapnik,V. (1998) *Statistical Learning Theory*. Wiley, New York.

Wang,K., Gan,L., Jefferey,E., Gayle,M., Gown,A., Skelly,M., Nelson,P., Ng,W., Schummer,M., Hood,L. and Mulligan,J. (1999) Monitoring gene expression profile changes in ovarian carcinomas using cDNA microarray. *Gene*, **229**, 101–108.

Wen,X., Fuhrman,S., Michaels,G., Carr,D., Smith,S., Barker,J. and Somogyi,R. (1998) Large-scale temporal gene expression mapping of central nervous system development. *Proc. Natl. Acad. Sci. USA*, **95**, 334–339.

Zhang,L., Zhou,W., Velculescu,V., Kern,S., Hruban,R., Hamilton,S., Vogelstein,B. and Kinzler,K. (1997) Gene expression profiles in normal and cancer cells. *Science*, **276**, 1268–1272.

Zhu,H., Cong,J., Mamtora,G., Gingeras,T. and Schenk,T. (1998) Cellular gene expression altered by human cytomegalovirus: global monitoring with oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA*, **95**, 14470–14475.

Zien,A., Rätsch,G., Mika,S., Schölkopf,B., Lemmen,C., Smola,A., Lengauer,T. and Müller,K. (2000) Engineering support vector machine kernels that recognize translation initiation sites. *Bioinformatics*, to appear.