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

UCSC-CRL-00-04

Terrence S. Furey[†][*], Nello Cristianini[‡], Nigel Duffy[†], David W. Bednarski[¶]
Michèl Schummer[¶], David Haussler[†]

[†]Department of Computer Science, University of California, Santa Cruz, Santa Cruz, CA 95064
[‡]Department of Engineering Mathematics, University of Bristol, Bristol, UK
[¶]Department of Molecular Biotechnology, University of Washington, Seattle, WA
booch@cse.ucsc.edu, nello@cse.ucsc.edu, nigeduff@cse.ucsc.edu,
michel@drschummer.de, haussler@cse.ucsc.edu

April 4, 2000

Abstract

DNA microarray experiments generating thousands of gene expression measurements are being used to gather information from tissue and cell samples about gene expression differences that will be useful in diagnosing disease. We have developed a new method to analyze 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. 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 analyze 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 analyzed. The results are comparable to those previously obtained. We show that other machine learning methods perform comparably to the SVM on many of those datasets as well.

Keywords: Support vector machines, microarray expression data, ovarian cancer

[*]Corresponding author: telephone (831) 458-1972, fax (831)459-4829

1 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 [26, 6, 9, 31, 22], or alternately evaluating each gene in a single environment but in different types of tissues, especially cancerous tissues [10, 1, 13, 21, 33, 30, 24, 32]. Those of the first type have allowed for identification of functionally related genes due to common expression patterns [5, 11, 31, 22], while the latter experiments have shown promise in classifying tissue types (diagnosis) and in the identification of genes whose expressions are good diagnostic indicators [13, 1]. In order to extract information from gene expression measurements, different methods have been employed to analyze this data including support vector machines [5, 20] clustering methods [11, 26, 1, 21, 2, 15], self-organizing maps [27, 13], and a weighted correlation method [13].

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 [5], detecting remote protein homologies [17], recognizing translation initiation sites [34], and breast cancer diagnosis and prognosis [19]. We have also recently become aware of two other current efforts that use SVMs in analyzing expression data [20] and (Jaakkola, personal communication). SVMs have demonstrated the ability to not only correctly separate entities into appropriate classes, but also in identifying instances whose established classification is not 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 analyzes 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. 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 mis-labeled, and another as an outlier, which is shown in the section 4 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 used in Golub *et al.*[13] 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 for ovarian cancer. However, further work needs to be done to identify the most effective feature selection/dimensionality reduction methods for this kind of data.

To test the generality of the approach, we also tested it on the leukemia data from Golub *et al.*[13] (72 patient samples) and the colon tumor data from Alon *et al.*[1] (62 tissue samples). Our results are comparable to those obtained in these 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 done. Indeed, the different kernels we tried performed nearly equally well and variants 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 datasets increase in size and complexity, we predict that our method will continue to demonstrate excellent performance, superior to that of simpler methods, but this is currently only speculation.

2 Microarray expression experiments

In recent years, several methods have been developed for performing gene expression experiments. In general, thousands of distinct DNA probes are attached to a microarray whose surface is typically made of coated glass or a type of membrane. Probes can be PCR products or oligonucleotides whose sequences correspond to target genes (or ESTs) of the genome being studied. RNA is extracted from the sample tissues or cells, reverse transcribed into labeled cDNA, which is then allowed to hybridize with the probes on the microarray. The cDNA corresponds to transcripts produced by genes in the samples, and the amount of a particular cDNA sequence present will be in proportion to the level of expression of its corresponding gene. The microarray is washed to remove non-specific hybridization, and the level of hybridization for each probe is calculated. From these measurements, an expression level for genes corresponding to the probes is derived. This level may represent a ratio between the expression of the gene under some control condition as compared to the test condition. This is repeated for each tissue or cell sample.

Certain experimental conditions can affect the accuracy of the expression measurements. There are problems inherent to PCR amplification that can result in probes that do not match the intended sequence or in differential amplification of cDNA. Cross-hybridization of repetitive sequences and non-specific hybridization to non-DNA features present on the array can lead to false-positive or false-negative signals. Lastly, tissue samples as opposed to cell samples introduce the possibility that expression levels being measured are due to the composition of the tissue rather than the expression of a particular gene in each cell.

For more in depth discussions of these techniques, see Lockhart *et al.*[18] which describes Affymetrix oligonucleotide arrays and Schummer *et al.*[24] which analyzes membrane arrays made from cDNA clones.

3 Support vector machine method

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.

3.1 Support Vector Machines

Support vector machines (SVMs) [8] are a relatively new type of learning algorithm, originally introduced by Vapnik and co-workers [4, 29] 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 bioinformatics.

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 input points be realizations of the random variable $\mathbf{X} =(X_1, ..., X_n)$, and let $\mathbf{x}^j = (x_1^j, ..., x_n^j)$ be the j^{th} input point. Let the input points be labeled by the random variable $Y = \{-1, +1\}$.

Let ϕ: $I \subseteq \Re^n \to F \subseteq \Re^N$ 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: $S = \{(\mathbf{x}^1, y^1), ..., (\mathbf{x}^m, y^m)\}$. 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, 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

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

It is easy to prove [8] 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

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

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 $K(\mathbf{x}^i, \mathbf{x}^j) = e^{-\frac{\|\mathbf{x}^i - \mathbf{x}^j\|}{\sigma^2}}$, 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 [7]. One of these techniques [25] 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 [25].

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 [5]. 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.[1]

[1] 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 implementation is based on that described in [17] 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.

3.2 Feature Selection

Our feature selection criterion is essentially that used in Golub *et al.*[13]. We start with a dataset S consisting of m expression vectors $\mathbf{x^i} = (x_1^i, .., x_n^i), 1 \leq i \leq m$, where m is the number of tissue or cell samples and n is the number of genes measured. Each sample is labeled from $Y = \{+1, -1\}$ (e.g. cancer vs. normal). For each gene x_j, we calculate the 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[2]

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

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 differ slightly from Golub *et al.* in our use of these scores. They calculate separately scores for genes whose expression level is greater on average in class 1 than in class 2, and similarly for those greater in class 2. In creating a set of optimal discriminating features, they then select an equal number from each set. We simply take the genes with the top $F(x_j)$ score regardless of the class in which x_j is expressed more.

3.3 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 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 a 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.

It should be noted that it is very important that when feature selection is performed, the sample being tested must not be included in this process. Each individual hold-one-out test requires a new ranking of features using only those samples that are to be used for training. Inclusion of the test sample when doing feature selection can cause a leak of information which invalidates the independence assumptions required to reasonably evaluate the methods performance. For this problem in particular, a lot of information can be leaked in this way. [3]

In the SVM tests reported here, the kernel used in all cases is simply the dot-product of

[2] This score is closely related to the Fisher criterion score for the j^{th} feature, $F(j) = (\mu_j^+ - \mu_j^-)^2/((\sigma_j^+)^2 + (\sigma_j^-)^2)$ [3].

[3] Thanks to Tomaso Poggio for pointing this out to us.

the two input vectors.[4] A more complex kernel is not required, which we attribute to the small number of examples. As increasingly complex datasets become available providing more examples, higher-order kernels may become necessary [20].

4 Ovarian data results

The microarray hybridization experiments were performed using 97,802 DNA probes or clones attached to membranes. Expression levels were measured for 31 tissue samples which 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 clones were 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 were done using all expression scores for each tissue. Diagonal factor settings of 0, 2, 5, and 10 were tested. Then clones were ranked in the manner described previously, and datasets consisting of the top 25, 50, 100, 500, and 1000 features were created. Experiments using similar diagonal factors as above were performed using these smaller feature sets. Table 1 displays the results from these experiments. The best classification is done using the top 50 features with a diagonal factor of 2, 5 or 10. The optimal score achieved using all features, though, is not significantly worse than those achieved by the smaller data sets.

An analysis of the misclassified examples revealed that one normal ovarian tissue sample, N039, was misclassified in all instances. In addition, the margin of misclassification (distance from decision boundary) was relatively large, meaning the SVM strongly believed it to be a cancerous ovarian tissue. Figure 1 shows the margins for classifications performed using the top 50 features and a diagonal factor of two. The margin in this case is the discriminant value calculated by the SVM which has been trained using the other 30 samples. For our experiments, for each tissue sample $\mathbf{x}$ with known label y, this discriminant is

$$y(\sum_{i=1}^{30} \alpha_i y_i (\langle \mathbf{x}^i, \mathbf{x} \rangle + 1) \tag{10}$$

A positive value indicates a correct classification, while a negative value indicates a misclassification. When the origin of this tissue was researched, it was realized that a miscommunication had caused the incorrect labeling of this tissue, and that it should have been labeled cancerous.

With a corrected label, the above experiments were run again, but disappointingly, classification results did not improve significantly. A similar analysis as above identified a second tissue, called HWBC3, as being consistently misclassified by a large margin in these new tests. It was also strongly misclassified in the original tests, as can be seen in Figure 1. This tissue is a non-ovarian normal tissue, and the only tissue of its type. Therefore, it is reasonable to believe that training the SVM on tissues with no relation might give spurious results when testing this tissue. Therefore, we removed this tissue and repeated the experiments with the remaining 30

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

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	500	5	3	11	12
dot-product	2	500	4	3	11	13
dot-product	5	500	4	3	11	13
dot-product	10	500	4	3	11	13
dot-product	0	1000	7	3	11	10
dot-product	2	1000	5	3	11	12
dot-product	5	1000	5	3	11	12
dot-product	10	1000	5	3	11	12
dot-product	0	97802	17	0	14	0
dot-product	2	97802	9	2	12	8
dot-product	5	97802	7	3	11	10
dot-product	10	97802	5	3	11	12

Table 1: **Error rates for ovarian cancer tissue experiments.**
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).

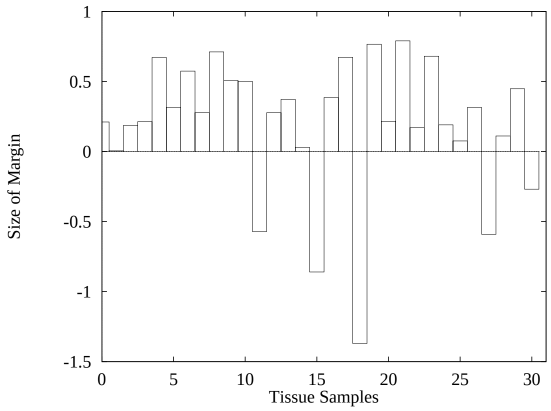

Figure 1: **SVM classification margins for ovarian tissues.** When classifying, the SVM calculates a margin which is the distance of an example from the decision boundary it has learned. In this graph, the margin for each tissue sample calculated using (10) is shown. A positive value indicates a correct 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.

Kernel	DF	Features	FP	FN	TP	TN
dot-product	0	25	1	3	12	14
dot-product	2	25	0	4	11	15
dot-product	5	25	0	4	11	15
dot-product	10	25	0	4	11	15
dot-product	0	50	0	3	12	15
dot-product	2	50	0	4	11	15
dot-product	5	50	0	4	11	15
dot-product	10	50	0	4	11	15
dot-product	0	100	1	4	11	14
dot-product	2	100	0	4	11	15
dot-product	5	100	0	4	11	15
dot-product	10	100	0	4	11	15
dot-product	0	500	3	3	12	12
dot-product	2	500	2	3	12	13
dot-product	5	500	1	3	12	14
dot-product	10	500	1	3	12	14
dot-product	0	1000	3	3	12	12
dot-product	2	1000	2	3	12	13
dot-product	5	1000	2	3	12	13
dot-product	10	1000	2	3	12	13
dot-product	0	97802	0	0	15	15
dot-product	2	97802	4	3	12	11
dot-product	5	97802	4	3	12	11
dot-product	10	97802	4	3	12	11

Table 2: **Error rates for ovarian cancer tissue experiments after classification of N039 as tumor and removal of HWBC3.**
For each setting of the SVM consisting of a kernel and diagonal factor (DF), each tissue was classified. Column three lists the number of features (clones) in the dataset. Reported are the number of normal tissues misclassified (FP), tumor tissues misclassified (FN), tumor tissues classified correctly (TP), and normal tissues classified correctly (TN).

tissue samples. Table 2 shows the results from these experiments. Perfect classification was achieved using all features and a diagonal factor of 0. No other setting, though, is able to make fewer than 3 mistakes, and therefore we cannot place much confidence in the one perfect experiment.

5 Ovarian feature analysis

After ranking the features using the procedure described above on all of the 31 samples, we attempted to sequence the top-ranked 10 clones. Using these 10 clones, we were able to achieve perfect classification in hold-one-out experiments, thus we felt that these clones may be significant in the identification of cancerous tissue. As stated above, this classification performance is overly optimistic due to the information leaked during feature selection. The fact that performance using just these 10 features is good merely serves to support the possibility that they are meaningful clones.

As Table 3 shows, three of the clones did not yield a readable sequence. Two of these clones could not be amplified and one represents more than one gene, in which case the sequencing reaction fails. We did not attempt to re-sequence these clones. Of the remaining seven clones, five represent expressed genes and two constituted repetitive sequences. Repetitive sequences occur naturally at 3' ends of messenger RNAs, some being as long as 1000 bp, some being as short as 10. Another source for repeats is the chromosomal DNA, which is a by-product of the mRNA preparation that can be reduced but hardly ever avoided. Since these two clones show a stronger signal than that in the normal tissues, the RNA prepared from tumor tissue must therefore contain more repetitive sequences. We can only speculate that genomic rearrangements in the tumor cells result in short chromosomal DNA fragments that can easily contaminate the mRNA preparation.

Out of the five clones that match to expressed genes, two were homologous to ESTs and three matched to known genes. For these 5 sequences, information is thus available that might tell us whether the gene is cancer-related (either a known or assumed tumor gene, or presence in cDNA libraries from tumor tissues in the case of ESTs). The cancer-relatedness of a feature helps us assess the quality of the clone ranking. Both EST-matching clones have homologies to ESTs that overwhelmingly come from tumor libraries. Likewise, one of the three clones with homology to known genes matches to ferritin H (GenBank accession number L20941), a known cancer gene [28]. Another clone matches to LYVE-1 (GenBank accession number NM_006691.1) a lymphatic gene which is more highly expressed in the tumors due to the lymphocytes infiltrating the tumors. The third known gene is poly(rC)-binding protein 2 (PCBP2, GenBank accession number NM_005016). This gene is under-expressed in the tumors. In summary, three of the five clones with homology to expressed genes are cancer-related and one is related to the presence of white blood cells in the tumor.

In order to evaluate this finding in the context of the accuracy of the clone scoring, we compared the identities of the top 1000 ranking clones to the bottom 1000. Since we did not have the means to sequence such as large number of clones, we had to content ourselves with the sequences generated in earlier random sequencing experiments (55 among the top 1000 and 28 among the bottom 1000). Table 4 shows that the top ranked clones are enriched for clones that did not yield a readable sequence (bad sequences), as well as for repeats and for tumor genes (such as ferritin H, CDC2 [GenBank accession number D88357] and the SET translocation gene [GenBank accession number NM_003011.1]). The number of tumor-related ESTs did not increase. Interestingly, the level of Immunoglobulin genes remains essentially the same. These genes are uniquely expressed by tumor-infiltrating white blood cells and one would have expected a higher showing in the top 1000. Another interesting finding is that the genes of the metabolism (mitochondrial genes and ribosomal proteins) which are commonly found to show elevated expression in tumors, and which here, if at all, tend to be found in the bottom 1000. Thus, the scoring enriches the tumor-related genes but also the non-specific sequences. From a tumor biologist's point of view, the accumulation of tumor-related genes at the top is a very useful feature when it comes to screening for novel cancer genes.

The above analysis seems to suggest that the feature selection method is able to identify clones that are cancer-related, and rank them highly. In addition, though, some clones seemed to obtain a high ranking while not having a meaningful biological explanation, and some known tumor genes are not ranked as high as would be expected. Given this and the inability of this feature selection method to significantly improve classification performance, additional effort is needed to develop ways of identifying meaningful features in these types of datasets.

6 Other data results

To demonstrate that our method can perform well in general compared to other methods used to analyze expression datasets, we performed similar experiments using previously published

Seq ID	Way of action
repeat ALU	
ferritin H	ferritin is used in trials as marker for ovarian cancer
EST 1	matches to ESTs form tumor libraries
bad PCR	
repeat LINE1	
bad sequence	
bad PCR	
LYVE-1	expressed on the lymph vessel wall
PCBP2	required for translation of poliovirus RNA: binds and stabilizes mRNA of erythropoietin, hepatitis A and C virus, and tyrosine hydroxylase
EST 2	matches to ESTs form tumor libraries

Table 3: **Sequence homologies of the top 10 scoring clones** These sequences were considered the 10 top-ranked clones using the feature selection method described above.

Gene	Top	%	Bottom	%	Ratio
total sequences	55		28		
bad sequences	10	18	2	7	+
repeats	6	11	0	0	+
tumor genes	7	13	1	4	+
ESTs	9	16	5	18	
tumor-related ESTs	7	13	4	14	
metabolism genes	7	13	7	25	-
novel sequence	0	0	2	7	-
Immunoglobulin	4	7	3	11	
other known genes	5	9	5	18	

Table 4: **Comparison of top and bottom ranked clones.** A total of 55 of the top 1000 ranked clones and 28 of the bottom 1000 ranked clones were sequenced. Each is categorized into one of nine groups. The number of sequences in each category from the top and bottom rankings is listed, along with the percentage of the total sequenced that category contains. The last column shows whether the category is more prevalent in either the top (+) 1000 or bottom (-) 1000 ranked clones.

datasets. The first dataset involves expression experiments on samples taken from patients with human acute leukemia. Initial analysis was performed by Golub *et al.* [13], and the dataset can be obtained at http://waldo.wi.mit.edu/MPR/cancer_class.html. The second dataset is comprised of expression data measuring levels of expression of genes in human tumor and normal colon tissues. Alon *et al.*[1] originally analyzed this data, and it is available at their website, http://www.molbio.princeton.edu/colondata.

6.1 AML/ALL dataset

Bone marrow or peripheral blood samples were 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 bone marrow samples of which 27 are ALL and 11 are AML, and a test set consisting of 24 bone marrow and 10 peripheral blood samples, 20 ALL and 14 AML. The dataset provided contained 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.*[13], we normalize these scores for each gene by subtracting the mean and dividing by the standard deviation of the expression values for that gene.

Golub *et al.* report accuracy of classification on the training set using a weighted voting scheme[5] and also clustering using self-organizing maps (SOMs). Hold-one-out cross validation tests using the weighted voting scheme correctly classify all samples for which a prediction is made, 36 of the 38 samples, while declining to predict on the remaining two. A two-cluster SOM produced one cluster with 24 ALL and 1 AML sample, and the second with 10 AML and 3 ALL samples.

We also did a full hold-one-out cross-validation measurement of the accuracy of our method on the training set alone. The SVM method was able to correctly classify all samples of the training set with a diagonal factor of two. Retesting subsets with only the top-ranked 25, 250, 500, and 1000 features, it was also able to correctly classify all training samples correctly using a diagonal factor of two in all cases.

We then tried to classify samples in the test set using a classifier that had been trained only on the examples in the training set. Multiple dataset sizes and diagonal factor settings perform optimally on the training set, and testing each combination produced 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. The SVM predicted incorrectly on five samples in at least one of its tests, and of these five, none were given predictions by Golub *et al.*. Two samples, patients 54 and 66, were misclassified in all cases.

Information is provided as to whether the ALL samples were of B-cell lineage or T-cell lineage. Using all 47 ALL samples from both 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 misclassified only a single tissue when using a zero diagonal factor. Golub *et al.* use SOMs to create 4

[5] The weighted voting scheme uses a group of 50 genes selected and described in the subsection "Feature Selection" where each gene predicts a class for each sample. These predictions are combined with each being weighted by "the degree of that gene's correlation with the class distinction", which is the $F(g)$ score defined above. If this combination exceeds a threshold in favor of one class over the other, a prediction is made.

clusters using all examples in the training set, including the AML samples. The first cluster contains 10 AML samples, the second contains 8 T-lineage ALL samples and 1 B-lineage ALL sample, the third contains 5 B-lineage ALL samples, and the last one contains 13 B-lineage ALL samples and a single AML sample.

Lastly, results of chemotherapy treatments for 15 AML patients is available. Treatment was considered successful if the patient went into remission for 46 to 84 months, otherwise it was considered a failure. Golub *et al.* report that they were unable to achieve accurate results using their weighted voting scheme. On hold-one-out cross-validation tests, the SVM was able to classify 10 of the 15 patients using the top 5 or 10 ranked features and a diagonal factor of two, thus performing slightly better than chance. One misclassified sample, patient 37, was consistently misclassified using multiple settings, and by a relatively large margin.

6.2 Colon tumor dataset

Using Affymetrix oligonucleotide arrays, expression levels for 40 tumor and 22 normal colon tissues were measured for 6500 human genes. Of these genes, the 2000 with the highest minimal intensity across the tissues were selected for classification purposes, and these were made publicly available. The scores in the dataset represent a gene intensity derived in a process described in Alon *et al.*[1]. The data was not processed further before performing classification. Alon *et al.* use a clustering method to create clusters of tissues. In their experiments, one cluster consisted of 35 tumor and 3 normal tissues, and the other 19 normal and 5 tumor tissues.

Using the SVM method with full hold-one-out cross-validation on the dataset of 62 tissues, we were able to correctly classify correctly all but six of the tissues using all 2000 features and a diagonal factor of two. Using only the top 1000 genes produced similar results at the same diagonal factor. The six misclassified in each of the optimal runs were exactly the same and consisted of three tumor tissues (T30, T33, T36) and three normal tissues (N8, N34, N36). T30, T33, and T36 are among the 5 tumor tissues that were clustered with the majority of the normal tissues by Alon *et al.*, and N8 and N32 were similarly in the cluster containing the majority of the tumor tissues.

Figure 2 plots the margins for the tissues based on the experiments using all of the data with a diagonal factor of two. This is analogous to Figure 1 above, which helped identify the mis-labeled tissue in the ovarian dataset. As we can see in Figure 2, none of the six misclassified tissues were borderline cases according to the SVM.

Alon *et al.* define a muscle index based on the average intensity of ESTs that are homologous to 17 smooth muscle genes. They explain that “normal tissue samples include a mixture of tissue types, while the tumor samples are biased to epithelial tissue of the carcinoma”. Therefore, it is expected that tumor tissues should have lower expression levels for these 17 ESTs and a smaller muscle index. In general, this proved to be true. Interestingly, though, all tumor tissues had a muscle index less than or equal to 0.3 except for T30, T33, and T36, and all normal tissues had an index of greater than or equal to 0.3 except N8, N34, and N36.

Without the assistance of the biologists who conducted these experiments, we cannot explore whether it is possible that one or more of these samples were bad or mis-labeled. Simply removing all six samples from the data and re-testing still produced classification errors. Two of the samples, N36 and T36, are especially interesting because their names indicate that they originated from the same patient, yet both are consistently misclassified by the SVM. Also, N36 has a muscle index or 0.1 and T36 has a muscle index of 0.7 which is counter-intuitive.

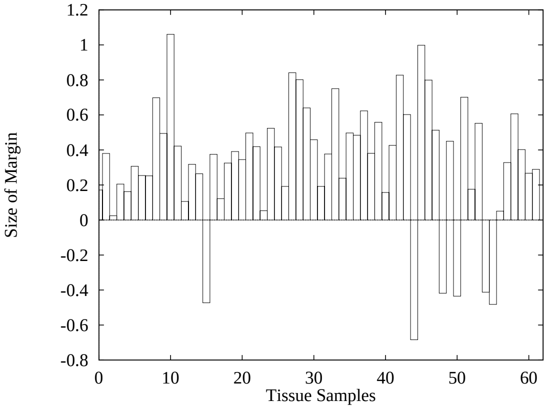

Figure 2: **SVM classification margins, colon tissues.** When classifying, the SVM calculates the margin which is the distance of an example from the separating hyper-plane it has learned. In this graph, the margin for each tissue sample is shown. A positive value indicates a correct classification, and a negative value indicates an incorrect classification. The incorrect classifications correspond to (from left to right) N8, T30, T33, N34, N36, and T36.

7 Comparison to perceptron-like classification algorithms

As discussed in the introduction, we do not claim that we can prove that the SVM method is better than other classification techniques on this type of dataset. The second family of algorithms we tested are generalizations of the Perceptron algorithm [23]. This simple algorithm works in an on-line way, running through the data and updating a weight vector each time it makes a mistake. The new weight vector is

$$\mathbf{w}^{t+1} = \mathbf{w}^t + y^t \mathbf{x}^t \tag{11}$$

and again the resulting decision rule is linear[6], the classification is given by $\text{sign}(\langle \mathbf{w^t}, \mathbf{x} \rangle)$. However, this algorithm requires modification when there is no perfect linear decision rule. Helmbold and Warmuth [16] provided such a modification, for which they derived performance guarantees. The modification simply amounts to taking a linear combination of the decision rules used at each iteration of the algorithm. The final decision rule is $\text{sign}(\sum_t \langle \mathbf{w^t}, \mathbf{x} \rangle)$. Freund and Schapire [12] demonstrated that kernels other than the simple inner product can also 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, the use of a more complex kernel did not improve performance for these problems and so we report only results for an inner product kernel. We also tested an algorithm known as the p-norm perceptron [14], using the same averaging procedure[7]. Theoretical results suggest that these algorithms perform well when good sparse hypotheses are available.

Our results for the modified perceptron are comparable to those for the SVM and the scores achieved for each dataset are given in Table 5.

Although it is not suggested by the theory, we observed that this algorithm achieves improved performance by running through the data several times; this was also observed by Freund and

[6] We did not use a bias in these experiments.

[7] It is an open question whether kernels can be applied to such algorithms.

Dataset	Features	FP	FN	SVM FP	SVM FN
Ovarian(original)	97802	4.6	4.8	5	3
Ovarian(modified)	97802	4.4	3.4	0	0
AML/ALL train	7129	0.6	2.8	0	0
AML treatment	7129	4.8	3.5	3	2
Colon	2000	3.8	3.7	3	3

Table 5: **Results for the perceptron on all data sets.** The results are averaged over 5 shufflings of the data as this algorithm is sensitive to the order in which it receives the data points. The first column is the dataset used and the second is number of features in the dataset. For the ovarian and colon datasets, the number of normal tissues misclassified (FP) and the number of tumor tissues misclassified (FN) is reported. For the AML/ALL training dataset, the number of AML samples misclassified (FP) and the number of ALL patients misclassified (FN) is reported. For the AML treatment dataset, the number of unsuccessfully treated patients misclassified (FP) and the number of successfully treated patients misclassified (FN) is reported. The last two columns report the best score obtained by the SVM on that dataset.

Schapire[12]. The p-norm perceptron did not perform as well as the theory might suggest and we only report results for the standard perceptron.

8 Conclusion

We have presented a method to analyze microarray expression data for genes from several tissue or cell types using support vector machines. 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 more complex datasets become available, the use of more complex kernels will 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.

9 Acknowledgments

We have used the SVM software written by Bill Grundy and thank him for his assistance with the software and also for his comments on a earlier draft. We are particularly grateful to Tomaso Poggio for pointing out a flaw in our 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

[1] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra D. Mack, and J. Levine. 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, 1999.

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

[3] Chris Bishop. *Neural Networks for Pattern Recognition.* Oxford UP, Oxford, UK, 1995.

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

[5] M.P.S. Brown, W.N. Grundy, D. Lin, N. Cristianini, C.W. Sugnet, T.S. Furey, Jr M. Ares, and D. Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc. Natl Acad. Sci. USA*, 97(1):262–267, 2000.

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

[7] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[8] N. Cristianini and J. Shawe-Taylor. *An introduction to Support Vector Machines.* Cambridge University Press, Cambridge, UK, 2000. www.support-vector.net.

[9] J.L. DeRisi, V.R. Iyer, and P.O. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 278:680–686, 1997.

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

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

[12] Yoav Freund and Robert E. Schapire. Large margin classification using the perceptron algorithm. In *Proceedings of the 11th Annual Conference on Computational Learning Theory*, pages 209–217, New York, NY, 1998. ACM Press.

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

[14] Adam J. Grove, Nick Littlestone, and Dale Schuurmans. General convergence results for linear discriminant updates. In *Proceedings of the 10th Annual Conference on Computational Learning Theory*, pages 171–183, New York, NY, 1997. ACM Press.

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

[16] D. Helmbold and M. K. Warmuth. On weak learning. *Journal of Computer and System Sciences*, 50(3):551–573, June 1995.

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

[18] D.J. Lockhart, H. Dong, M.C. Byrne, M.T. Follettie, M.V. Gallo, M.S. Chee, M. Mittmann, C. Wang, M. Kobayashi, H. Horton, and E.L. Brown. Expression monitoring by hybridization to high–density oligonucleotide arrays. *Nature Biotechnology*, 14:1675–1680, 1996.

[19] O.L. Mangasarian, W.N. Street, and W.H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. *Operations Research*, 43(4):570–577, 1995.

[20] S. Mukherjee, P. Tamayo, J.P. Mesirov, D. Slonim, A. Verri, and T. Poggio. Support vector machine classification of microarray data. Technical Report CBCL Paper 182/AI Memo 1676, M.I.T., December 1999.

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

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

[23] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psych. Rev.*, 65:386–407, 1958.

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

[25] J. Shawe-Taylor and N. Cristianini. Further results on the margin distribution. In *Proc. 12th Annual Conf. on Computational Learning Theory*, New York, NY, 1999. ACM Press.

[26] P. Spellman, G. Sherlock, M. Zhang, V. Iyer, K. Anders, M. Eisen, P. Brown, D. Botstein, and B. Futcher. Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell*, 9:3273–3297, 1998.

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

[28] P.K. Tripathi and S.K. Chatterjee. Elevated expression of ferritin H-chain mRNA in metastatic ovarian tumor. *Cancer Invest*, 14:518–526, 1996.

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

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

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

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

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

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