Combining microarray expression data and phylogenetic profiles to learn gene functional categories using support vector machines

Paul Pavlidis
Columbia Genome Center
Columbia University
pp175@columbia.edu

William Noble Grundy
Department of Computer Science
Columbia University
bgrundy@cs.columbia.edu

Abstract

A primary goal in biology is to understand the molecular machinery of the cell. The sequencing projects currently underway provide one view of this machinery. A complementary view is provided by data from DNA microarray hybridization experiments. Synthesizing the information from these disparate types of data requires the development of improved computational techniques. We demonstrate how to apply a machine learning algorithm called support vector machines to a heterogeneous data set consisting of expression data as well as phylogenetic profiles derived from sequence similarity searches against a collection of complete genomes. The two types of data provide accurate pictures of overlapping subsets of the gene functional categories present in the cell. Combining the expression data and phylogenetic profiles within a single learning algorithm frequently yields superior classification performance compared to using either data set alone. However, the improvement is not uniform across functional classes. For the data sets investigated here, 23-element phylogenetic profiles typically provide more information than 79-element expression vectors. Often, adding expression data to the phylogenetic profiles introduces more noise than information. Thus, these two types of data should only be combined when there is evidence that the functional classification of interest is clearly reflected in both data sets.

Keywords: support vector machines, microarray expression data, functional inference

Introduction[1]

DNA microarray technology offers a new method for analyzing the molecular mechanisms of the cell on a large scale. By offering a snapshot of the messenger RNA expression levels of thousands of genes at once, microarrays allow biologists to ask formulate models of gene expression on a scale that was unimaginable several years ago. Much work remains to be done, both in perfecting the various technologies for conducting microarray hybridization experiments and in interpreting the results of those experiments. Initial analyses of this type of data focused on clustering algorithms, such as hierarchical clustering (Eisen *et al.* 1998) and self-organizing maps (Tamayo *et al.* 1999). These algorithms attempt to automatically locate clusters of genes that share similar expression patterns and hence may share similarity in function. Clustering algorithms are considered unsupervised learning algorithms, because they exploit no prior knowledge other than the gene expression data itself. As we begin to understand some of the structure in microarray gene expression data, we can begin to apply supervised learning algorithms. Such algorithms exploit our prior knowledge of gene functional categories that we expect to appear in the data.

Recently, Brown *et al.* applied a collection of supervised learning techniques to a set of microarray expression data from yeast (Brown *et al.* 2000). They showed that an algorithm known as a support vector machine (SVM) (Vapnik 1998; Burges 1998) provides excellent classification performance for several gene functional categories that were known to cluster well in the given data set.

One major goal of this paper is to extend the results from (Brown *et al.* 2000) to functional classes that have not been previously shown to cluster well from expression data. Accordingly, the experiments reported here apply the SVM technique to a large collection of gene functional categories. We show that the SVM is capable of learning to recognize many of these categories.

On the other hand, for many of the functional categories we examine, the SVM learns little or nothing about the genes. Therefore, the second major goal of this paper is to improve the SVM's ability to functionally classify genes in these hard-to-recognize categories. We do this by exploiting a completely different type of data — phylogenetic profiles (Pellegrini *et al.* 1999) — in combination with the microarray expression data. A phylogenetic profile is derived from a comparison between a given gene and a collection of complete genomes. The profile characterizes the evolutionary history of the given gene. Two genes with similar phylogenetic profiles are likely to have similar functions, under

[1]Corresponding author: William Noble Grundy, Department of Computer Science, Columbia University, 450 Computer Science Building, Mail Code 0401, 1214 Amsterdam Avenue, New York, NY 10027, Tel: (212) 939-7114, Fax: (212) 666-0140

the assumption that their similar pattern of inheritance across species is the result of a functional link. This type of functional inference is different from functional inferences from gene expression patterns. Hence, the two types of data provide complementary views of gene function.

In the experiments described below, we show that a support vector machine trained from a collection of yeast microarray expression data can learn to recognize a large number of yeast gene functional categories. We show that SVMs can also be trained from phylogenetic profiles. For the data used here, the phylogenetic profiles provide a better characterization of a larger number of functional classes. However, the most complete picture of gene function is learned by SVMs trained on a combination of these two different types of data.

The paper begins with a background section that describes DNA microarray hybridization experiments, phylogenetic profiles and the support vector machine learning algorithm. This section is followed by a detailed description of the data sets and methods employed. All of the data and software used for these experiments is available on the web at http://www.cs.columbia.edu/compbio. We conclude with a description of the experimental results and some discussion.

Background

DNA microarray expression data

Each data point produced by a DNA microarray hybridization experiment represents the ratio of expression levels of a particular gene under two different experimental conditions (Lashkari *et al.* 1997; DeRisi, Iyer, & Brown 1997). An experiment starts with microarray construction, in which several thousand DNA samples are fixed to a glass slide, each at a known position in the array. Each sequence corresponds to a single gene within the organism under investigation. Messenger RNA samples are then collected from a population of cells subjected to various experimental conditions. These samples are converted to cDNA via reverse transcription and are labeled with one of two different fluorescent dyes in the process. A single experiment consists of hybridizing the microarray with two differently labeled cDNA samples collected at different times. Generally, one of the samples is from the reference or background state of the cell, while the other sample represents a special condition set up by the experimenter. The level of expression of a particular gene is roughly proportional to the amount of cDNA that hybridizes with the DNA affixed to the slide. By measuring the ratio of each of the two dyes present at the position of each DNA sequence on the slide using laser scanning technology, the relative levels of gene expression for any pair of conditions can be measured. The result, from an experiment with n DNA samples on a single chip, is a series of n expression-level ratios. Typically, the numerator of each ratio is the expression level of the gene in the condition of interest to the experimenter, while the denominator is the expression level of the gene in the reference state of the cell.

Microarray data from a set of n separate experiments may be represented as a gene expression matrix, in which each row consists of an n-element expression vector for a single gene. Following Eisen *et al.*, we do not work directly with the ratio as discussed above but rather with its logarithm (Eisen *et al.* 1998). We define X_i to be the logarithm of the ratio of gene X's expression level in experiment i to X's expression level in the reference state. This log ratio is positive if the gene is induced (turned up) with respect to the background and negative if it is repressed (turned down). The log ratios are stored in the gene expression matrix.

The task of classifying genes into functional categories using microarray expression data rests upon the assumption that genes of similar function share similar expression profiles across a large number of experimental conditions. Clearly, whether or not this assumption holds depends upon the functional category being learned. However, previous analyses suggest that the expression patterns for some important categories, such as the tricarboxylic acid (TCA) cycle, respiration, cytoplasmic ribosomal genes, proteasome and histones, exhibit significant similarities.

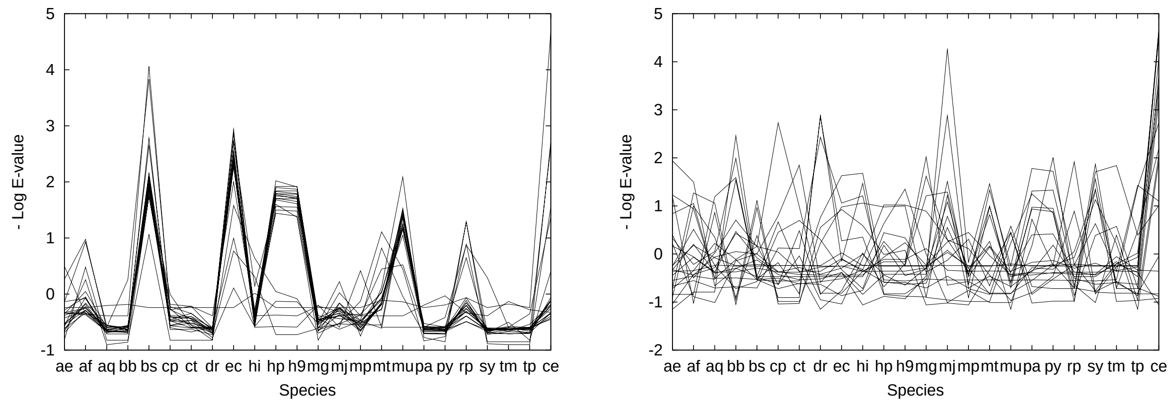
Phylogenetic profiles

In its simplest form, a phylogenetic profile is a bit string, in which each bit corresponds to a completely sequenced genome. The Boolean value of the bit indicates whether the gene of interest has a close homolog in the corresponding genome. Two genes with similar phylogenetic profiles have a similar pattern of inheritance across the organisms in the genomic database. This coupled inheritance may indicate a functional link between the genes. The inferred functional link is based upon the hypothesis that the genes are always present together or always both absent because they cannot function independently of one another.

Phylogenetic profiles need not consist entirely of Boolean values. Marcotte *et al.* mention real-valued phylogenetic profiles (Marcotte *et al.* 1999) without providing a detailed description. The profiles employed in this paper contain, at each position, the negative logarithm of the lowest E-value reported by BLAST version 2.0 (Altschul *et al.* 1997) in a search against a complete genome, with negative values (corresponding to E-values greater than 1) truncated to 0. Figure 1 shows the phylogenetic profiles of the genes in class X. The profiles show clear similarities, thus supporting the hypothesis that the profiles indicate function.

Support vector machines

A support vector machine is a supervised learning algorithm developed over the past decade by Vapnik and others (Vapnik 1998). In the form employed here, SVMs learn binary classifications; i.e., the SVM learns to answer the question, "Does the given gene belong to functional class X," where X is some category such as



(a) (b)

Figure 1: **Similarity of phylogenetic profiles among genes in the amino acid transporters class.** Figure (a) shows the phylogenetic profiles of the 22 genes in the MYGD amino acid transporter class. For comparison, Figure (b) shows profiles from a set of 22 randomly-selected genes. Both figures were generated from data with mean 0 and variance 1.

"ribosomal genes" or "sugar and carbohydrate transporters." It is straightforward to generalize SVMs to learn multiple classes at once.

Support vector machines classify points by locating them with respect to a hyperplane, known as the decision boundary, that separates class members from nonclass members. Each row in the gene expression matrix may be thought of as a point in an m-dimensional space. A simple way to build a binary classifier is to construct a decision boundary in this space. Unfortunately, most real-world problems involve non-separable data for which there does not exist a hyperplane that successfully separates the class members from non-class members in the training set. The SVM solves the inseparability problem by mapping the data into a higher-dimensional space and defining a separating hyperplane there. This higher-dimensional space is called the feature space, as opposed to the input space occupied by the training examples. With an appropriately chosen feature space of sufficient dimensionality, any consistent training set can be made separable. In order to avoid overfitting, the SVM chooses a decision boundary with maximum distance to any point in the training set. In order to allow for noise in the training set labels, most SVMs, including the ones used here, employ a soft margin that allows some members of the training set to be misclassified.

Explicitly translating the gene expression vectors into a higher-dimensional space can be computationally expensive. SVMs avoid this overhead by using a mathematical trick. The SVM employs a function, called a kernel function, that plays the role of the dot product operator in the feature space. The mathematical trick involves writing out the SVM learning algorithm (for finding the separating hyperplane from the training set) and the classification algorithm (for deciding upon the classification of genes in the test set) entirely in terms of vectors in the input space and dot products in the feature space. By using the kernel function in place of the dot product operation, both algorithms can be carried out without ever explicitly translating the gene expression vectors into the feature space. Different feature spaces can be explored simply by substituting different kernel functions. The class of allowable kernel functions is large and consists of all continuous positive semi-definite functions. Indeed, much of the art of applying a support vector machine lies in selecting an appropriate kernel function, since the kernel function expresses prior knowledge about the phenomenon being modeled, encoded as a similarity measure between two vectors.

SVMs are similar in many ways to neural networks. Both algorithms translate input vectors into a higher-dimensional feature space and find a separating hyperplane there. When the kernel function is a simple dot product, the SVM is equivalent to a single-layer neural network. A certain class of sigmoid kernel functions allows for SVMs that mimic the behavior of two-layer neural networks. In general, however, it is not possible to characterize the feature space implied by an arbitrary neural network connection topology. In contrast, the SVM mapping depends in a straightforward way upon the selected kernel function. Furthermore, unlike the backpropagation algorithm for training neural networks, the SVM training algorithm solves a simple convex optimization problem with a single global maximum. Finally, the SVM training algorithm has strong learning-theoretic underpinnings which suggest that the algorithm offers near-optimal generalization performance. Support vector machines have been successfully applied in many domains, including handwriting recognition, object recognition, speaker identification, face detection and text categorization (Burges 1998).

A detailed description of support vector machines is beyond the scope of this paper. However, a useful SVM review is available (Burges 1998), and a book on SVMs was recently published (Cristianini & Shawe-Taylor 2000).

Methods

The analyses described here are carried out using a set of 79-element gene expression vectors for 2467 yeast genes (Eisen *et al.* 1998). These genes were selected by Eisen *et al.* (Eisen *et al.* 1998) based on the availability of accurate functional annotations. The data were generated from spotted arrays using samples collected at various time points during the diauxic shift (DeRisi, Iyer, & Brown 1997), the mitotic cell division cycle (Spellman *et al.* 1998), sporulation (Chu *et al.* 1998), and temperature and reducing shocks, and are available on the Stanford web site (http://www-genome.stanford.edu).

In addition to the microarray expression data, each of the 2467 yeast genes is characterized by a phylogenetic profile (Pellegrini *et al.* 1999). The profiles are constructed using 23 complete genomes, collected from The Institute for Genomic Research website (http://www.tigr.org/tdb). The 2467 23-element profiles are available at http://www.cs.columbia.edu/compbio.

Classification experiments are carried out using gene functional categories from the Munich Information Center for Protein Sequences Yeast Genome Database (MYGD) (http://www.mips.biochem.mpg.de/proj/yeast). The database contains several hundred functional classes, whose definitions come from biochemical and genetic studies of gene function. The experiments reported here use all classes containing ten or more genes, which yields a total of 126 classes.

For each class, a support vector machine is trained to discriminate between class members and non-members using the 79-element vectors of microarray expression data. Prior to learning, the SVM adjusts the input vectors to have a mean of 0 and a variance of 1. Then the SVM finds a separating hyperplane, using a dot-product kernel function raised to the third power:

$$K(X,Y) = \left(\frac{\vec{\mathbf{X}} \cdot \vec{\mathbf{Y}}}{\sqrt{\vec{\mathbf{X}} \cdot \vec{\mathbf{X}}}\sqrt{\vec{\mathbf{Y}} \cdot \vec{\mathbf{Y}}}} + 1 \right)^3 .$$

This kernel function takes into account pairwise and tertiary correlations between gene expression measurements. The normalization term in the denominator projects the data onto the unit sphere. This kernel has been shown to produce good classification performance for some MYGD classes using this data set (Brown *et al.* 2000). As in previous work, the SVM uses a soft margin that accounts for the disparity in the number of positive and negative examples for each class. For details about this adjustment, see (Brown *et al.* 2000). The SVM software used to perform these experiments is available at http://www.cs.columbia.edu/compbio.

Each classification experiment is performed using three-fold cross-validation. For a given class, the positively labeled and negatively labeled genes are split randomly into three groups. An SVM is trained on two-thirds of the data and is tested on the remaining third. This procedure is repeated twice more, each time using a different third of the genes as test genes. This three-fold cross-validation experiment is repeated five times with different random splits. Similar cross-validated experiments are performed using the 23-element phylogenetic profiles on all 126 classes. Finally, for all of the learnable classes, SVMs are trained using the combined vectors of 102 features.

The performance of each SVM is measured by examining how well the classifier identifies the positive and negative examples in the test sets. Each gene in the test set can be categorized in one of four ways: true positives are class members according to both the classifier and MYGD; true negatives are non-members according to both; false positives are genes that the classifier places within the given class, but MYGD classifies as non-members; false negatives are genes that the classifier places outside the class, but MYGD classifies as members. To judge overall performance, we define the cost of using the method M as $C(M) = (fp(M) + 2 \cdot fn(M))/n$, where $fp(M)$ is the number of false positives for method M, $fn(M)$ is the number of false negatives for method M, and n is the number of members in the class. The false negatives are weighted more heavily than the false positives because, for these data, the number of positive examples is small compared to the number of negatives. The cost for each method is compared to the cost $C(N)$ for using the null learning procedure, which classifies all test examples as negative. We define the cost savings of using the learning procedure M as $S(M) = C(N) - C(M)$.

Results

The first goal of this work is to determine whether the SVM classification results described in (Brown *et al.* 2000) generalize to other gene functional classifications. We consider a class to be learnable by the SVM using a given data set if the average cost savings for that class across the five cross-validated tests is more than one standard deviation above zero. Of the 126 MYGD classifications that we investigated, 44 are learnable using the gene expression data described above. The twenty most easily learned classes are listed in Table 1. Not surprisingly, the classes identified by Eisen *et al.* (Eisen *et al.* 1998) via clustering are among the most learnable. Indeed, the class of ribosomal proteins is the most learnable of all. Two additional classes, respiration and the TCA pathway, are listed among the top fifteen. The remaining two classes identified by Eisen *et al.* (histones and proteasome) no longer exist in the MYGD. However, analyses performed using the old class definitions suggest that the histones and proteasomes would also appear in the list of highly learnable classes (data not shown).

Class	Expression	Class	Phylogenetic
ribosomal proteins	1.446	amino acid transporters	1.573
CELLULAR ORGANIZATION*	1.292	CELLULAR ORGANIZATION*	1.504
PROTEIN SYNTHESIS*	1.018	amino acid transport	1.390
mitochondrial organization	0.784	C compound carbohydrate transport*	1.387
organization of cytoplasm	0.691	sugar and carbohydrate transporters	1.350
cytoplasmic degradation	0.686	glycolysis and gluconeogenesis*	0.800
sugar and carbohydrate transporters	0.644	pyrimidine ribonucleotide metabolism	0.758
organization of chromosome structure	0.607	METABOLISM*	0.753
respiration	0.607	transport ATPases	0.661
C compound carbohydrate transport*	0.587	TRANSPORT FACILITATION	0.657
proteolysis	0.497	amino acid metabolism	0.619
ENERGY	0.492	amino acid biosynthesis*	0.571
METABOLISM*	0.459	organization of plasma membrane	0.554
tricarboxylic acid pathway	0.423	protein folding and stabilization	0.520
organization of endoplasmatic reticulum	0.409	organization of cell wall	0.512
nuclear organization	0.389	PROTEIN SYNTHESIS*	0.493
glycolysis and gluconeogenesis*	0.379	cellular import	0.491
pheromone response generation	0.363	purine ribonucleotide metabolism	0.440
TRANSCRIPTION	0.318	homeostasis of other ions	0.388
amino acid biosynthesis*	0.306	metal ion transporters	0.358

Table 1: **MYGD classes most learnable from expression vectors alone and from phylogenetic profiles alone.** Each side of the table lists the twenty classes that are most learnable from a given type of data. Classes are ranked according to the average cost savings, as defined in the text, over five cross-validation experiments. Classes that appear on both sides of the table are marked with an asterisk. Classes in SMALL CAPS are large superclasses at the top of the MYGD functional hierarchy.

The ability of the SVM to learn to recognize 44 functional classes is impressive; however, this result still leaves 80 functional classes that are defined by MYGD but are not evident to the SVM in the expression data. Although some of these remaining classes might be recognizable using an alternate machine learning algorithm, it is likely that many of the classes are not recognizable at all from this data. As mentioned previously, microarray hybridization experiments provide a large-scale but necessarily limited view of the state of a cell. Apparently, many of the 80 unlearnable classes are not visible from mRNA expression levels. The introduction of alternate types of data, such as phylogenetic profiles, promises to offer a complementary view.

Our experiments show that the SVM methodology generalizes well to phylogenetic profile data, and that this new type of data allows for the characterization of new functional classes. Of the 126 MYGD functional classes investigated, 49 are learnable from phylogenetic profiles. Table 1 lists the twenty most learnable classes. Only six classes are common to the two lists shown in Table 1. In total, twenty classes that were not learnable using expression data are learnable from phylogenetic profiles. The situation is summarized in Figure 2. While the expression data alone provides insight into 44 functional classes, adding the phylogenetic profile data increases the coverage of classes to 64, or more than half the classes we investigated. The large number of classes that are learnable from only one of the two types of data indicates that expression vectors and phylogenetic

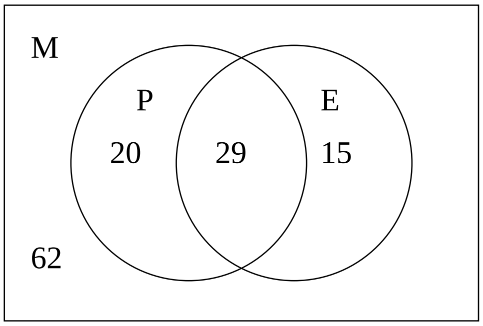


Figure 2: **Complementary views of gene functional categories.** The set M contains all 126 MYGD functional classes investigated in this work. The set P contains the 49 classes that are learnable from phylogenetic profiles alone. The set E contains the 44 classes that are learnable from expression vectors alone. Together, the two types of data provide insight into more than half of the functional categories defined in MYGD. The classes in P intersect E, not-P intersect E, and P intersect not-E are listed in Tables 2, 3 and 4, respectively.

profiles provide complementary, rather than redundant, information.

Furthermore, the results also indicate that, for these particular data and this classification algorithm, phylogenetic profiles provide better classification performance than expression vectors do. In addition to learning to recognize four more functional classes, the SVMs trained using phylogenetic profiles perform better overall. Among the 64 classes learnable by either method, phylogenetic profiles yield a higher cost savings 41 times and have a mean cost savings of 0.346, whereas the expression data yield a higher cost savings 23 times and have a mean cost savings of 0.245. Assigning statistical significance to these results is difficult because many of the MYGD classifications overlap one another, thus making some of the performance measurements dependent upon one another. However, for the 64 learnable classes, a two-tailed signed rank test (Henikoff & Henikoff 1997; Snedecor & Cochran 1980; Salzberg 1997) assigns a p-value of 0.0283 to the rejection of the null hypothesis that the two methods are eqivalent. This result is suggestive, if not conclusive, that the phylogenetic profiles provide better classification performance.

The previous analyses suggest that the expression vectors and phylogenetic profiles provide complementary information. We could predict, therefore, that an SVM trained from both types of data should be capable of better classification accuracy than SVMs trained from either data set alone. Our experiments bear out this prediction, although the improvement is not consistent across all classes. Results of experiments using the combined data set of expression vectors and phylogenetic profiles are shown in Tables 2-4. Table 2 lists the functional classes that are learnable from both types of training data alone (set P intersect E in Figure 2). Among these 29 classes, combining the phylogenetic profile and expression data improves classification performance in 14 cases. For the other 15 classes, the phylogenetic profiles alone provides better performance than the combined data set. This is a surprising result: for approximately half of these classes, adding the microarray expression data to the phylogenetic profiles hurts the performance of the SVM, even though we have already shown that these are classes for which the expression data provides some useful information to the SVM.

These conclusions are further supported by the data shown in Figure 3. For each gene in each class, we scored whether the SVM correctly classified the gene in at least three of the five trials using the phylogenetic data, the expression data, or the combined data. Overall, the SVM consistently misclassifies 7138 of 13321 genes, or 54=%.[2] The combined data leads to correct classification of 4759 genes, which amounts to 36% of the total and 77% of the correctly classified genes. In contrast, the phylogenetic data correctly classifies 3730 genes (28%), and the expression data correctly classifies 4308 genes (32%). In 422 cases, even though neither type of data is sufficient by itself, the combined data is. On the other hand, in 1424 cases, despite one data type being sufficient by itself, the combined data fails to correctly classify the genes, indicating that the extra data only adds noise in these cases. In only 117 cases (0.9%) does the SVM fail to classify correctly a gene using the combined data despite both data types being sufficient when used alone.

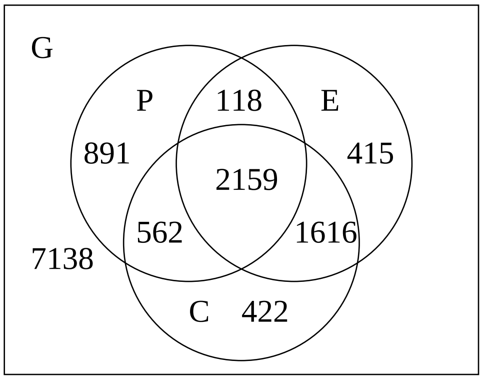


Figure 3: **Summary of gene functional classifications**. Each of the three sets, E, P and C, contain all genes that correctly are classified at least three times by the SVM, using expression vectors, phylogenetic profiles and the combined data sets, respectively. The universe G of all genes is larger than the number of genes in yeast because of the overlap in MYGD functional categories.

The general picture that emerges from this analysis is that many of the genes (43%) are classifiable on the basis of at least one of the two data types. Putting the two data types together typically allows the SVM to take advantage of the informative data type without interference. However, cases where such interference does occur greatly outnumber the cases in which only the combined data is successful. At the level of the functional classes, this situation is reflected in the fact that if a class is not learnable by at least one of the data types, then it is not learnable at all. There are no classes in the MYGD scheme that are only learnable using the combined data.

While errors made by the SVM may often be due to limitations in the data, in some cases errors reflect how the underlying biology relates to the classification scheme. Brown *et al.* point out several genes that, for biological reasons, lie on the border of their respective classes or may be mislabeled in MYGD. Similar cases emerged in our study when using phylogenetic profiles. For example, the top-scoring class, amino acid transporters, yielded three consistent false nega-

[2]The total number of genes classified is larger than the number of genes in yeast because many of the MYGD classifications overlap one another, even for classes from the bottom of the functional class hierarchy.

Class	Size	Expression	Phylogenetic	Combined	Win
ribosomal proteins	173	1.446 ± 0.04	0.152 ± 0.04	1.496 ± 0.04	C
C compound carbohydrate transport	31	0.587 ± 0.14	1.387 ± 0.03	1.355 ± 0.06	C
CELLULAR ORGANIZATION	1799	1.292 ± 0.03	1.504 ± 0.01	1.317 ± 0.01	P
sugar and carbohydrate transporters	32	0.644 ± 0.12	1.350 ± 0.03	1.250 ± 0.12	P
PROTEIN SYNTHESIS	296	1.018 ± 0.06	0.493 ± 0.08	1.149 ± 0.05	C
amino acid transporters	22	0.182 ± 0.11	1.573 ± 0.12	1.045 ± 0.09	P
amino acid transport	20	0.270 ± 0.14	1.390 ± 0.02	1.000 ± 0.14	P
mitochondrial organization	296	0.784 ± 0.04	0.309 ± 0.05	0.816 ± 0.01	C
organization of cytoplasm	464	0.691 ± 0.03	0.349 ± 0.02	0.749 ± 0.03	C
METABOLISM	701	0.459 ± 0.02	0.753 ± 0.01	0.723 ± 0.03	P
amino acid metabolism	158	0.223 ± 0.07	0.619 ± 0.07	0.682 ± 0.09	C
glycolysis and gluconeogenesis	29	0.379 ± 0.16	0.800 ± 0.13	0.607 ± 0.10	P
amino acid biosynthesis	89	0.306 ± 0.05	0.571 ± 0.07	0.573 ± 0.06	C
ENERGY	157	0.492 ± 0.04	0.283 ± 0.04	0.558 ± 0.03	C
TRANSPORT FACILITATION	182	0.181 ± 0.03	0.657 ± 0.04	0.549 ± 0.09	P
nuclear organization	641	0.389 ± 0.03	0.131 ± 0.04	0.482 ± 0.03	C
organization of plasma membrane	109	0.125 ± 0.05	0.554 ± 0.02	0.455 ± 0.06	P
TRANSCRIPTION	584	0.318 ± 0.04	0.195 ± 0.01	0.435 ± 0.05	C
organization of cell wall	16	0.287 ± 0.07	0.512 ± 0.10	0.413 ± 0.06	P
transport ATPases	33	0.170 ± 0.12	0.661 ± 0.13	0.370 ± 0.14	P
amino acid degradation	23	0.174 ± 0.09	0.278 ± 0.13	0.356 ± 0.21	C
nitrogen and sulphur metabolism	50	0.216 ± 0.12	0.136 ± 0.10	0.312 ± 0.08	C
purine ribonucleotide metabolism	30	0.233 ± 0.18	0.440 ± 0.11	0.307 ± 0.10	P
CELL GROWTH CELL DIVISION AND DNA SYNTH.	505	0.249 ± 0.05	0.066 ± 0.03	0.296 ± 0.05	C
homeostasis of other ions	48	0.142 ± 0.13	0.388 ± 0.09	0.279 ± 0.07	P
C compound and carbohydrate metabolism	259	0.080 ± 0.05	0.252 ± 0.05	0.248 ± 0.06	P
C compound and carbohydrate utilization	154	0.076 ± 0.02	0.186 ± 0.06	0.238 ± 0.08	C
stress response	89	0.101 ± 0.08	0.119 ± 0.05	0.130 ± 0.08	C
detoxificaton	50	0.108 ± 0.07	0.196 ± 0.06	0.112 ± 0.05	P

Table 2: **Classes learnable from both types of data.** The table lists all classes for which the mean cost savings is more than one standard deviation above zero for SVMs trained using expression data alone and for SVMs trained using phylogenetic profiles alone. The first two columns contain the class name and the number of genes in the class. The next three columns contain the average and standard deviation of the cost savings across five cross-validated experiments. These values are reported for experiments using expression vectors, phylogentic profiles, and the combined data set. The final column indicates which of the three experiments achieved the highest cost savings. Classes are ranked according to the mean cost savings using the combined data. Classes in SMALL CAPS are superclasses at the top of the MYGD functional hierarchy.

Class	Size	Expression	Phylogenetic	Combined	Win
cytoplasmic degradation	82	0.686 ± 0.03	-0.027 ± 0.01	0.688 ± 0.04	E
organization of chromosome structure	28	0.607 ± 0.04	0.000 ± 0.00	0.607 ± 0.03	E/C
respiration	57	0.607 ± 0.10	-0.098 ± 0.07	0.505 ± 0.10	E
proteolysis	115	0.497 ± 0.05	0.010 ± 0.03	0.489 ± 0.05	C
tricarboxylic acid pathway	17	0.423 ± 0.23	0.353 ± 0.36	0.776 ± 0.15	C
organization of endoplasmatic reticulum	134	0.409 ± 0.04	-0.028 ± 0.02	0.451 ± 0.03	C
pheromone response generation	16	0.363 ± 0.19	0.000 ± 0.00	0.237 ± 0.11	E
phosphate utilization	10	0.280 ± 0.27	0.080 ± 0.13	0.360 ± 0.26	E
fermentation	12	0.217 ± 0.16	0.250 ± 0.26	0.600 ± 0.18	C
phosphate metabolism	23	0.217 ± 0.04	0.043 ± 0.14	0.252 ± 0.02	C
rRNA transcription	94	0.185 ± 0.10	-0.013 ± 0.02	0.272 ± 0.10	C
PROTEIN DESTINATION	440	0.139 ± 0.03	0.013 ± 0.02	0.205 ± 0.02	C
mitochondrial transport	52	0.119 ± 0.05	0.008 ± 0.02	0.173 ± 0.07	C
DNA synthesis and replication	74	0.095 ± 0.09	-0.021 ± 0.07	0.081 ± 0.08	E
intracellular communication	45	0.062 ± 0.05	-0.013 ± 0.01	-0.014 ± 0.06	E

Table 3: **Classes learnable only from expression data.** The table lists all classes for which the mean cost savings is more than one standard deviation above zero for SVMs trained using expression data alone, and less than zero for SVMs trained using phylogenetic profiles alone. For further details, see the caption for Table 2.

Class	Size	Expression	Phylogenetic	Combined	Win
pyrimidine ribonucleotide metabolism	24	0.033 ± 0.06	0.758 ± 0.07	0.233 ± 0.16	P
protein folding and stabilization	45	-0.062 ± 0.05	0.520 ± 0.07	0.360 ± 0.10	P
cellular import	96	0.029 ± 0.04	0.491 ± 0.07	0.323 ± 0.07	P
metal ion transporters	19	0.105 ± 0.13	0.358 ± 0.02	0.010 ± 0.06	P
ABC transporters	16	-0.075 ± 0.03	0.350 ± 0.14	-0.037 ± 0.03	P
lipid fatty acid and isoprenoid utilizat	24	-0.058 ± 0.05	0.300 ± 0.02	-0.008 ± 0.08	P
metabolism of vitamins cofactors	54	-0.067 ± 0.03	0.278 ± 0.07	-0.037 ± 0.10	P
other cation transporters	30	-0.033 ± 0.09	0.273 ± 0.13	0.187 ± 0.12	P
IONIC HOMEOSTASIS	88	0.029 ± 0.07	0.246 ± 0.02	0.116 ± 0.04	P
biosynthesis of vitamins cofactors	42	-0.081 ± 0.04	0.238 ± 0.14	-0.033 ± 0.05	P
nucleotide metabolism	99	-0.063 ± 0.07	0.228 ± 0.09	0.085 ± 0.04	P
metabolism of energy reserves	21	-0.086 ± 0.06	0.209 ± 0.05	0.181 ± 0.10	P
nuclear transport	44	-0.155 ± 0.06	0.209 ± 0.08	-0.027 ± 0.02	P
rRNA processing	54	-0.082 ± 0.07	0.204 ± 0.03	0.263 ± 0.09	C
ion transporters	58	0.031 ± 0.09	0.193 ± 0.09	0.069 ± 0.09	P
homeostasis of metal ions	41	-0.020 ± 0.06	0.127 ± 0.10	0.025 ± 0.05	P
INTRACELLULAR TRANSPORT	361	-0.087 ± 0.06	0.122 ± 0.03	0.064 ± 0.03	P
recombination and DNA repair	63	-0.108 ± 0.05	0.079 ± 0.01	-0.013 ± 0.06	P
mRNA processing	66	-0.148 ± 0.10	0.070 ± 0.04	-0.094 ± 0.07	P
mRNA transcription	426	0.038 ± 0.07	0.040 ± 0.03	0.130 ± 0.03	C

Table 4: **Classes learnable only from phylogenetic profiles.** The table lists all classes for which the mean cost savings is more than one standard deviation above zero for SVMs trained using phylogenetic profiles alone, and less than zero for SVMs trained using expression data alone. For further details, see the caption for Table 2.

tives and one consistent false positive. The false positive, (YDR160W, SSY1) is a regulator of amino acid transporters (Klasson, Fink, & Ljungdahl 1999) and so is apparently phylogenetically conserved along with the transporters themselves. The false negatives are also instructive. Two of them, YNR056C (BIO5) and YOR071C are not known to be amino acid transporters, but rather transport biotin (Phalip *et al.* 1999) and possibly thiamine (based on homology to THI10), respectively. The third, YOR160W (ARG11) is thought to be required primarily for mitochondrial ornithine transport, though it can act as an arginine transporter (Palmieri *et al.* 1997). All three of these genes are likely to play a role in amino acid metabolism, but based on the phylogenetic profile, the SVM makes a distinction between them and the rest of this MYGD class, which primarily consists of plasma membrane amino acid transporters.

Discussion

As the quantity and variety of genomic data increases, molecular biology shifts from a hypothesis-driven model to a data-driven one. Whereas previously a single laboratory could collect data and test hypotheses regarding a single system or pathway, this new paradigm requires combining genome-wide experimental results, typically gathered and shared across multiple laboratories. For example, constructing a single, n-element phylogenetic profile requires the availability of n complete genomic sequences, which clearly could not yet be generated by a single laboratory. The data-driven model requires sophisticated computational techniques that handle very large, heterogeneous data sets.

The support vector machine learning algorithm is such a technique. SVMs scale well and have been used successfully with large training sets in the domains of text categorization and image recognition (Cristianini & Shawe-Taylor 2000). Furthermore, in this paper, we demonstrate that SVMs can be used with a heterogeneous data set. With appropriate normalization, the SVM learns from a concatenation of two different types of feature vectors. In many cases, the resulting trained SVM provides better gene functional classification performance than an SVM trained on either data set alone.

The idea of combining heterogeneous data sets to infer gene function is not new. Marcotte *et al.* describe an algorithm for functional annotation that uses expression vectors and phylogenetic profiles, as well as evolutionary evidence of domain fusion (Marcotte *et al.* 1999). However, the algorithm consists of predicting functional links between pairs of genes using each type of data separately, and then cataloging the complete list of links. In contrast, the SVM method described here considers the various types of data at once, making a single prediction for each gene with respect to each functional category.

Clearly, support vector machines are not unique in their ability to learn from heterogenous data sets. We investigated this technique because it has previously been shown to provide good classification performance on a small set of gene functional categories, when compared with a collection of standard machine learning techniques (Brown *et al.* 1999). In future, however, the comparison of classification algorithms should be carried out on a larger scale, using the additional gene functional categories and phylogenetic profiles described here.

In addition to testing the ability of SVMs to learn from a combination of gene expression vectors and phylogenetic profiles, this paper explores the two views of gene function provided by these two types of data. Two conclusions are apparent. First, the two types of data provide complementary pictures of gene function. Many gene functional categories are recognizable using only one of these two types of data. This difference is not surprising, since DNA microarray expression experiments provide a snapshot of mRNA expression levels in the cell at a particular moment, whereas phylogenetic profiles describe the inheritance pattern of the gene during various speciation events.

Second, the results reported here suggest that, for these data, phylogenetic profiles allow the SVM to predict gene function more accurately than expression vectors do. In part, this result may be due to uncharacterized noise in the microarray expression measurements. In the rush to apply this new technology to biologically important questions, little research has been performed to quantify and characterize the experimental error in data produced using these techniques. An accurate noise model might allow, for example, for the removal of some of the noise and for the inclusion of appropriate weighting on the two types of data.

Although the method described here provides good classification performance for many functional categories, the task is far from complete. Our analysis reveals that for many functional classes, the phylogenetic profile is not informative. This situation may change as more complete genomes become available. Among the genomes from which we derived the phylogenetic profiles, all but one are bacterial. Thus, in the current state of affairs, it is difficult to generate useful phylogenetic profiles for genes that are specific to eucaryotes. All such genes would be expected to have homologs in at most one of the complete genomes (that of *C. elegans*), and thus their phylogenetic profiles will not be easily distinguishable. Similiarly, for genes that are common to all organisms, the phylogenetic profile is not likely to be informative. Because of the over-representation of bacterial genomes, we would predict that phylogenetic profiles would be more successful for classifiying bacterial genes, until more genomes become available. Ideally, the data set would include both closely and distantly related genomes. An analogous situation exists for the expression data. Far more data will be required to permit effective classification of the genes in many of the functional classes. This is because, in the limited data set, many genes in different classes have common expression profiles.

The experiments reported here suggest many avenues for future research. The two types of data could be combined in a more complex fashion. For example, introducing relative weights on the two types of data could improve performance. The weights could reflect prior knowledge of the informativeness of the data or could be determined experimentally using a hold-out set. More complex methods of combining data could be accomplished using data-specific kernel functions. Another obvious research direction involves including additional types of data. Having shown that two types of data can be fruitfully combined, we plan to extend the techniques described here to feature vectors derived from the upstream promoter regions of genes and from the protein sequences themselves, as described previously (Jaakkola, Diekhans, & Haussler 1999).

Support vector machines are part of a larger class of algorithms known as kernel methods, which have recently been gaining in popularity (Schölkopf, Burges, & Smola 1999). A kernel method is any algorithm that employs a kernel function to implicitly operate in a higher-dimensional space. In addition to SVM classifiers, kernel methods have been developed for regression (Schölkopf, Smola, & Müller 1996) and principal components analysis (Schölkopf, Smola, & Müller 1997). More members of this promising class of algorithms should be applied to problems in computational biology.

References

Altschul, S. F.; Madden, T. L.; Schaffer, A. A.; Zhang, J.; Zhang, Z.; Miller, W.; and Lipman, D. J. 1997. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Research* 25:3389–3402.

Brown, M. P. S.; Grundy, W. N.; Lin, D.; Cristianini, N.; Sugnet, C.; M. Ares, J.; and Haussler, D. 1999. Support vector machine classification of microarray gene expression data. Technical Report UCSC-CRL-99-09, University of California, Santa Cruz.

Brown, M. P. S.; Grundy, W. N.; Lin, D.; Cristianini, N.; Sugnet, C.; Furey, T.; M. Ares, J.; and Haussler, D. 2000. Knowledge-based analysis of microarray gene expression data using support vector machines. *Proceedings of the National Academy of Sciences of the United States of America* 97(1):262–267.

Burges, C. J. C. 1998. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery* 2(2):121–167.

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.

Cristianini, N., and Shawe-Taylor, J. 2000. *An Introduction to Support Vector Machines.* Cambridge UP.

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.

Eisen, M.; Spellman, P.; Brown, P.; and Botstein, D. 1998. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences of the United States of America* 95:14863–14868.

Henikoff, S., and Henikoff, J. G. 1997. Embedding strategies for effective use of information from multiple sequence alignments. *Protein Science* 6(3):698–705.

Jaakkola, T.; Diekhans, M.; and Haussler, D. 1999. A discriminative framework for detecting remote protein homologies. *Journal of Computational Biology.* To appear.

Klasson, H.; Fink, G.; and Ljungdahl, P. 1999. Ssy1p and Ptr3p are plasma membrane components of a yeast system that senses extracellular amino acids. *Molecular and Cellular Biology* 19:5405–5416.

Lashkari, D. A.; L., J.; DeRisi; McCusker, J. H.; Namath, A. F.; Gentile, C.; Hwang, S. Y.; Brown, P. O.; and Davis, R. W. 1997. Yeast microarrays for genome wide parallel genetic and gene expression analysis. *Proceedings of the National Academy of Sciences of the United States of America* 94:13057–13062.

Marcotte, E. M.; Pellegrini, M.; Thompson, M. J.; Yeates, T. O.; and Eisenberg, D. 1999. A combined algorithm for genome-wide prediction of protein function. *Nature* 402(6757):83–86.

Palmieri, L.; Marco, V. D.; Iacobazzi, V.; Palmieri, F.; Runswick, M.; and Walker, J. 1997. Identification of the yeast arg-11 gene as a mitochondrial ornithine carrier involved in arginine biosynthesis. *FEBS Letters* 410:447–451.

Pellegrini, M.; Marcotte, E. M.; Thompson, M. J.; Eisenberg, D.; and Yeates, T. O. 1999. Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. *Proceedings of the National Academy of Sciences of the United States of America* 96(8):4285–4288.

Phalip, V.; I.Kuhn; Lemoine, Y.; and Jeltsch, J. 1999. Characterization of the biotin biosynthesis pathway in *Saccharomyces cerevisiae* and evidence for a cluster containing bio5, a novel gene involved in vitamer uptake. *Gene* 232(43):43–51.

Salzberg, S. L. 1997. On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Mining and Knowledge Discovery* 1:371–328.

Schölkopf, B.; Burges, C. J. C.; and Smola, A. J., eds. 1999. *Advances in Kernel Methods: Support Vector Learning.* Cambridge, MA: MIT Press.

Schölkopf, B.; Smola, A.; and Müller, K.-R. 1996. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation* 10(5):1299–1319.

Schölkopf, B.; Smola, A.; and Müller, K.-R. 1997. Kernel principal component analysis. In *Proceedings*

ICANN97, Springer Lecture Notes in Computer Science, 583.

Snedecor, G. W., and Cochran, W. G. 1980. *Statistical Methods*. Iowa: Iowa State University Press.

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

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. *Proceedings of the National Academy of Sciences of the United States of America* 96:2907–2912.

Vapnik, V. N. 1998. *Statistical Learning Theory*. Adaptive and learning systems for signal processing, communications, and control. New York: Wiley.