Assessing the limits of genomic data integration for predicting protein networks

Long J. Lu,[1] Yu Xia,[1] Alberto Paccanaro,[1] Haiyuan Yu,[1] and Mark Gerstein[1,2,3,4]

[1]*Department of Molecular Biophysics and Biochemistry,* [2]*Department of Computer Science, and* [3]*Program of Computation Biology and Bioinformatics, Yale University, New Haven, Connecticut 06520, USA*

Genomic data integration—the process of statistically combining diverse sources of information from functional genomics experiments to make large-scale predictions—is becoming increasingly prevalent. One might expect that this process should become progressively more powerful with the integration of more evidence. Here, we explore the limits of genomic data integration, assessing the degree to which predictive power increases with the addition of more features. We focus on a predictive context that has been extensively investigated and benchmarked in the past—the prediction of protein–protein interactions in yeast. We start by using a simple Naive Bayes classifier for integrating diverse sources of genomic evidence, ranging from coexpression relationships to similar phylogenetic profiles. We expand the number of features considered for prediction to 16, significantly more than previous studies. Overall, we observe a small, but measurable improvement in prediction performance over previous benchmarks, based on four strong features. This allows us to identify new yeast interactions with high confidence. It also allows us to quantitatively assess the inter-relations amongst different genomic features. It is known that subtle correlations and dependencies between features can confound the strength of interaction predictions. We investigate this issue in detail through calculating mutual information. To our surprise, we find no appreciable statistical dependence between the many possible pairs of features. We further explore feature dependencies by comparing the performance of our simple Naive Bayes classifier with a boosted version of the same classifier, which is fairly resistant to feature dependence. We find that boosting does not improve performance, indicating that, at least for prediction purposes, our genomic features are essentially independent. In summary, by integrating a few (i.e., four) good features, we approach the maximal predictive power of current genomic data integration; moreover, this limitation does not reflect (potentially removable) inter-relationships between the features.

[All genomic feature data used in this study can be downloaded at http://networks.gersteinlab.org/intint/.]

A major challenge in post-genomic biology is systematically mapping the interactome, the set of all protein–protein interactions within an organism. Since proteins carry out their functions by interacting with one another and with other biomolecules, reconstructing the interactome of a cell is the important first step toward understanding protein function and cell behavior (Hartwell et al. 1999; Eisenberg et al. 2000). Recently, several large-scale protein-interaction maps have been experimentally determined in the model organism *Saccharomyces cerevisiae* (Uetz et al. 2000; Ito et al. 2001; Gavin et al. 2002; Ho et al. 2002). These studies have drastically improved our knowledge of protein interactions. Unfortunately, the data sets generated from these studies are often noisy and incomplete (von Mering et al. 2002). In addition to experimentally determined interaction data sets, there exists a large amount of biological information in the expanding functional genomic data sets, such as sequence, structure, functional annotation, and expression-level databases. It is thus desirable to computationally predict protein–protein interactions by exploiting the interaction evidence contained in these data sets. Such predictions can serve as a valuable complement to the current experimental efforts. Several studies have been carried out to search for individual features contained in the genomic data sets that are useful for interaction prediction. For example, two proteins are likely to interact if they have homologs in another genome that are fused into a single protein, or if their mRNA expression patterns are correlated (Marcotte et al. 1999a,b; Ideker et al. 2001; Jansen et al. 2002a). Detailed reviews of these individual methods can be found elsewhere (Valencia and Pazos 2002; Xia et al. 2004).

Each genomic feature, by itself, is only a weak predictor of protein interactions. However, predictions can be improved by integrating different genomic features (Marcotte et al. 1999b). There are two main reasons for this. First, predicting a protein–protein interaction with confidence depends on how much evidence supports it. When multiple distinct features all support a predicted interaction, our confidence in the prediction increases. Second, different features may cover different subsets of the interactome, and feature integration can increase the coverage. Feature integration can be accomplished via simple rules, such as intersection, union, or majority vote. To achieve optimal predictive power, however, different genomic features need to be properly integrated into a single probabilistic framework (Gerstein et al. 2002). Many machine learning methods can be used for feature integration, such as Bayesian approaches (Troyanskaya et al. 2001; Jansen et al. 2003; Friedman 2004), decision trees (Lin et al. 2004; Zhang et al. 2004), and support vector machines (Brown et al. 2000). In particular, Bayesian approaches can be roughly divided into two broad groups as follows: (1) learning to infer the causal structure of cellular networks from quantitative measurements (Friedman 2004); (2) classification based on a set of probabilistic rules. Here, we focus on the second classification aspect of Bayesian approaches. In addition to protein–protein interaction prediction, feature integration is also essential for other prediction problems in genomics as well, such as localization prediction (Drawid et al. 2000), function prediction (Troyanskaya et al.

[4]Corresponding author.
E-mail Mark.Gerstein@yale.edu; fax (360) 838-7861.
Article and publication are at http://www.genome.org/cgi/doi/10.1101/gr.3610305.

2001; Lee et al. 2004), and genetic interaction prediction (Wong et al. 2004).

One might expect genomic data integration to become increasingly powerful with the integration of more evidence. Here, we explore the limits of genomic data integration, assessing the degree to which predictive power increases with addition of more features. We focus on a predictive context that has been extensively investigated and benchmarked in the past; the prediction of protein–protein interactions in yeast. Previously, we developed a Naive Bayesian classification approach to predict protein–protein interactions in yeast by integrating four genomic features (functional similarity based on MIPS and GO annotations, mRNA expression correlation, and coessentiality) (Jansen et al. 2003). By definition, two proteins interact if they belong to the same complex. The parameters in the Naive Bayes classifier were trained using a collection of protein pairs known to be interacting or noninteracting. The advantages of Naive Bayes classifiers are twofold. First, the models constructed by Naive Bayes classifiers are readily interpretable; they represent conditional probabilities among features and class labels (interaction vs. noninteraction). Second, Naive Bayes classifiers are very flexible for the highly heterogeneous genomic features. Numerical features and categorical features can be easily combined, and missing data can be readily handled.

In this study, we expand the list of genomic features to include 16 diverse features that are plausible indicators for protein interactions. These 16 features are assembled based on both protein pair features and single protein features, and they are derived from a wide range of physical, genetic, contextual, and evolutionary properties of yeast genes. We believe that such "feature-richness" is an essential property of genomic data sets; therefore, we would like to test whether protein-interaction predictions can be further improved by exploiting the diversity of the features, and if so, by how much.

Naive Bayes classifiers assume conditional independence between features (see Methods). In the following text, when we say (in)dependent, we mean conditionally (in)dependent. We would expect that there exists a high dependence between a number of genomic features, and that this would become increasingly likely as we try to integrate more features. In this case, Naive Bayes may no longer be the optimal approach, as the dependence among features needs to be taken into account.

In this study, we apply boosting to Naive Bayes classifiers as an automated and efficient way for handling dependent features. Boosting (Schapire 1990)—in particular, AdaBoost (Freund and Schapire 1996)—is a recent development in the field of machine learning. The process combines the performances of several weak classifiers to form strong predictions via a weighted majority vote. In our case, the weak classifiers can be either individual features or simple Naive Bayes classifiers. Boosting approximately finds the best linear combination of all possible weak classifiers via maximum likelihood on a logistic scale (Friedman et al.

	Features	Description		Biological Meaning and Rationale for Using this Feature
1.1 Four Original Features (F1-F4) Used in (Jansen et al., 2003)	**COE** F-1. mRNA Co-expression	Source #O/#P OvIp+/-	Cho et al, (1998); Ho et al, (2002) 6,128 / 18,773,128 7,614 / 2,675,273	These data can be used for the prediction of protein-protein interaction, because proteins in the same complex are often co-expressed (Ge et al., 2001; Jansen et al., 2002b; Kemmeren et al., 2002). This feature is obtained in both the Rosetta and cell cycle datasets by computing the Pearson correlations for each protein pair.
	MIP F-2. MIPS Functional Similarity	Source #O/#P OvIp+/-	Mewes et al, (2002) 3,511 / 6,161,805 8,051 / 1,313,579	Interacting proteins often function in the same biological process (Letovsky and Kasif, 2003; Schwikowski et al., 2000; Vazquez et al., 2003). This means two proteins that interact are more likely to belong to the same biological process than to different processes. We collected information from two catalogs of functional information about proteins: the MIPS functional catalog (Mewes et al., 2002), which is separate from the MIPS complexes catalog (Mewes et al., 2002), and the data on biological processes from Gene Ontology (GO) (Ashburner et al., 2000).
	GOF F-3. GO Functional Similarity	Source #O/#P OvIp+/-	Ashburner et al, (2000) 2,399 / 2,878,800 7,520 / 647,060	The rational is the same as F-2. The MIPS and GO functional similarity scores are calculated as follows: First, two proteins of interest are assigned to a set of functional classes that two proteins share, given one of the functional classification systems. Then, the ~18 million protein pairs in yeast that share the exact same functional classes as the protein pairs in question are counted (yielding a count between 1 and ~18 million). In general, a small count entails higher similarity and specificity for the functional description of the two proteins.
	ESS F-4. Co-essentiality	Source #O/#P OvIp+/-	Mewes et al., (2002) 4,040 / 8,130,528 2,150/573,724	Yeast proteins can be experimentally characterized as either essential or non-essential (Mewes et al., 2002). If two proteins exist in a complex, they are likely to both be either essential or non-essential, but not a mixture thereof. This is because a deletion mutant of either one protein should produce the same phenotype: both would impair the function of the same complex.
1.2. New Features from Functional and Comparative Genomics	**EXP** F-5. Absolute mRNA Expression	Source #O/#P OvIp+/-	Greenbaum et al, (2002) 6,214 / 19,303,791 7,786 / 2,696,002	We will discuss this feature together with F-7. APA – Absolute Protein Abundance (see below).
	MES F-6. Marginal Essentiality	Source #O/#P OvIp+/-	Yu et al., (2004a) 5,963 / 17,775,703 7,738 / 2,588,199	Marginal essentiality is a quantitative measure of the importance of a non-essential gene to a cell (Yu et al., 2004a); it is based on the "marginal benefit" hypothesis that many non-essential genes make significant but small contributions to the fitness of the cell, even though the effects might not be large enough for detection by conventional methods (Thatcher et al., 1998). Yu et al. (2004a) found that this quantity relates to many of the topological characteristics of protein interaction networks. In particular, proteins with a greater degree of MES tend to be network hubs (i.e. they have many interactions) and tend to have a shorter characteristic path length than others. Based on this observation, we hypothesize that two proteins are more likely to interact with a higher combined marginal essentiality.*
	APA F-7. Absolute Protein Abundance	Source #O/#P OvIp+/-	Greenbaum et al, (2002) 3,867 / 7,474,911 5,192 / 1,514,555	mRNA expression level/protein abundance level can be used to predict protein interactions because two proteins that interact should be present in stoichiometrically similar amounts. Protein abundance (number of proteins per cell) can be determined by gel electrophoresis and several mass spectrometric approaches with varying accuracy. However, as tools for analyzing mRNA expression level become more mainstream, mRNA expression level has often been used as a surrogate for protein abundance, and substantial agreement between these two kinds of datasets have been found (Greenbaum et al., 2003). In this study, we will use the scaled merged protein abundance and absolute expression level sets that we have developed for yeast.
	REG F-8. Co-regulation	Source #O/#P OvIp+/-	Yu et al., (2003) 3,268 / 449,091 3,948 / 59,767	Gene regulatory proteins regulate the transcription of specific sets of target genes to respond to changes in condition. Many co-regulated target genes function together through protein interactions. Thus, co-regulation between genes – determined, for instance, through chip-chip experiments (Horak and Snyder, 2002; Lee et al., 2002; Martone et al., 2003) – can help predict protein interactions.
	PGP F-9 Phylogenetic Profiles	Source #O/#P OvIp+/-	Pellegrini et al, (1999) 1,722 / 152,506 914/26,095	Pairs of non-homologous proteins that are present or absent together in different organisms are likely to have co-evolved (Pellegrini et al., 1999). Co-evolution has been observed between interacting proteins, such as chemokine and its receptors (Goh et al., 2000). Pellegrini et al. (1999) have examined the co-occurrence or absence of genes across multiple genomes, thereby inferring functional relatedness.
	GNN F-10 Gene Neighborhood	Source #O/#P OvIp+/-	Bowers et al., (2004) 1,333 / 8,797 312 / 1,161	It has been suggested that genes located near each other on the chromosome are more likely to interact (Tamames et al., 1997). Such chromosomal proximity between functionally related genes may be conserved across different organisms. By comparing multiple genomes, these neighboring pairs of genes can be identified and used to establish functional linkages.

Figure 1. (Continued on next page)

2000), thereby solving potential feature redundancy and statistical dependence problems. By comparing the performance of a simple Naive Bayes classifier with a boosted Naive Bayes classifier on our collection of features, we will be able to address whether or not the dependence among our collection of features—if any—decreases the Naive Bayes classifier's predictive power. In other words, does the Naive Bayes approach perform sufficiently well at the current level of feature dependence? This comparison will also be done on a set of highly dependent features as a control.

Results and Discussion

A list of features useful for predicting protein interactions

In addition to the four features in Jansen et al. (2003), we consider 12 more features as listed in Figure 1. These features are divided into four categories; each of them is assigned a three-character identification code for convenient reference. Also included in Figure 1 are two gold-standard data sets (GSTDs, positive and negative sets) that will be used to evaluate features in subsequent sections. These GSTDs have various degrees of overlap with the 16 features. In Figure 1, we present the four categories of features in the descending order according to the degree of overlaps with the GSTDs (Fig. 2). For each of them, we shall describe its biological meanings and the rationale to use it. The reference to the data source is in the parenthesis that follows the feature's name.

Predictive power of individual features

We use ROC curves (see Methods) to illustrate the predictive power of each individual feature. Figure 2 shows that there is a distinct difference between the features to the left and right of the divider in terms of overlapping with the GSTDs (note, Fig. 2 is in log-scale). For this reason, and in the interest of a clear presentation, we plot the ROC curves in two panels, with the seven most populous features in one group and the remaining features in the other (Fig. 3).

A good feature, i.e., one with high predictive power, simultaneously has a large number of true positives and a small number of false positives. In this case, the ROC curve climbs rapidly away from the origin (lower left hand corner of the graph). How quickly the ROC curve arises away from the origin can be quantified by measuring the area under the curve. The larger the area, the better the feature. Ranking the features by the area they cover in the ROC curves (easily seen in Fig. 3A), the best feature in the first group is MIP, followed by GOF, COE, EXP, ESS, MES, and APA. All of these features show strong predictive power (i.e., well above the diagonal). The best feature in the second group is INT, followed by PGP, GNN, REG, ROS, and THR, while SYL shows very little predictive power. EVL and GNC are not shown here

1.2 (Continued)	**ROS** F-11. Rosetta Stone	Source #O/#P Ovlp+/-	Marcotte et al. (1999) 1,112 / 8,197 113 / 1,303	Proteins that are involved in the same pathway or molecular complex in one organism are sometimes fused into a single polypeptide chain in another organism to facilitate reaction efficiency (Berger et al., 1996). This gene-fusion event can be useful in detecting interacting proteins (Marcotte et al., 1999a). This method also called Domain Fusion Method.
	SYL F-12. Synthetic Lethality	Source #O/#P Ovlp+/-	Tong et al., (2004) 1,468 / 4,917 95 / 792	This information is associated with the observation that jointly knocking out two genes, individually not essential, is lethal to a cell (Tong et al., 2001). Synthetic lethal relationships may occur for a pair of genes involved in a single biochemical pathway or complex, or for genes within two distinct pathways. In the latter case, one process functionally compensates for or buffers the defects in the other. Synthetic genetic array analysis, an approach that allows systematic construction of double mutants, enables large-scale mapping of genetic interactions.
	GNC F-13. Gene Cluster (or Operon Method)	Source #O/#P Ovlp+/-	Bowers et al, (2004) 4,492 / 2,968 2 / 407	A cluster of genes transcribed as a single mRNA molecule is called an operon, commonly found in bacteria. Operons contain two or more closely spaced genes located on the same DNA strand. The encoded proteins of a common operon often function together (Alberts, 2002). The GNC method utilizes physical gene proximity to reconstruct plausible operon structures and predict functional relatedness between pairs of genes (Bowers et al., 2004). In other words, a pair of genes is "linked" by GNC if the intergenic nucleotide distance between them is less than a specified threshold.**
1.3 New Sequence/ Structure Features	**THR** F-14. Threading Scores	Source #O/#P Ovlp+/-	Lu et al., (2003) 1,241 / 7,300 103 / 1,155	Threading has been widely used in the predictions of protein tertiary structures (Baker and Sali, 2001; Skolnick and Kolinski, 2002). Lu et al. (2002) extended the traditional threading to predict protein quaternary structures (i.e., protein complexes) by incorporating the interfacial energy between two protein chains. Although this multimeric threading algorithm uses structural information, it does not require the structures of the query proteins be solved experimentally, making it more widely applicable than a docking approach. This algorithm has predicted yeast interactome with an above-average accuracy among high-throughput methods.
	EVL F-15. Co-evolution Scores	Source #O/#P Ovlp+/-	Goh et al., (2000) 1,304 / 1,303 2 / 299	Co-evolutionary analysis on protein families has also been useful to identify protein interaction partners. Protein-protein interfaces can adapt to mutations as they co-evolve. Based on this hypothesis, Goh et al. (2000) quantified the co-evolution between soluble protein families that were known to interact. They were able to identify binding partners for proteins with previously unknown interaction partners (Goh and Cohen, 2002). Pazos and Valencia (2002) extended this idea by applying it to large sets of proteins and protein domains, thereby identifying pairs of interacting proteins.
1.4 Other Features	**INT** F-16. Interologs in Another Organism	Source #O/#P Ovlp+/-	Yu et al., (2004b) 787 / 21,290 3,741 / 3,996	Interolog mapping is the transfer of interaction annotation from one organism to another using comparative genomics. Yu et al. (2004b) quantitatively assess the degree to which interologs can be reliably transferred between species as a function of the sequence similarity between the corresponding interacting proteins. Using interaction information generated by yeast two-hybrid experiments, they find that protein–protein interactions can be transferred when a pair of proteins has a joint sequence identity >80% or a joint E-value $<10^{-70}$. (These "joint" quantities are the geometric means of the identities or E-values for the two pairs of interacting proteins.)

	Data Sets	Description		Construction of Training/Testing Datasets from GSTDs
1.5 GSTDs	**GSTD+** Gold Standard Positive Set	Source #O/#P	Jansen et al., (2003) 871 / 8,250	We continue to use the two GSTDs (positive and negative sets) constructed in our original study (Jansen et al., 2003). The GSTD+ is extracted from the MIPS complexes catalog, which consists of a filtered set of 8,250 protein pairs within the same complex. The GSTD- of ~2.7 million protein pairs is compiled by pairing proteins from different subcellular compartments. In order for a (boosted) Naïve Bayes classifier to integrate multiple features and evaluate their integrated predictive power, we construct training and testing sets from a subset of the GSTDs, in which every protein pair has at least one feature value. We then randomly select a quarter of these protein pairs from this subset as a testing set, and the remaining three quarters as a training set. To evaluate the predictive power of a single feature, we apply a Naïve Bayes classifier to the single feature. The training and testing sets are constructed using the same procedure as described above, except that the subset of the GSTDs is now the intersection of this single feature and the GSTDs.
	GSTD- Gold Standard Negative Set	Source #O/#P	Jansen et al., (2003) 2,903 / 2,708,622	

#O / #P — Number of ORFs / Number of ORF Pairs; Ovlp+/- — Number of Overlaps with GSTD+/GSTD-.
*It is also reasonable to hypothesize that proteins in one protein complex have a similar level of marginal essentiality, because a deletion mutant of any one protein should normally produce the same phenotype: both impair the function of the same complex. However, we observe a stronger predictive power by assuming the former hypothesis (results not shown). **This GNC method can be distinguished from the GNN method (F-10): the former relies only on a single genome to establish functional linkages and the latter compares multiple genomes to identify genes of close chromosomal proximity (Strong et al., 2003).

Figure 1. Useful genomic features in prediction of protein interactions.

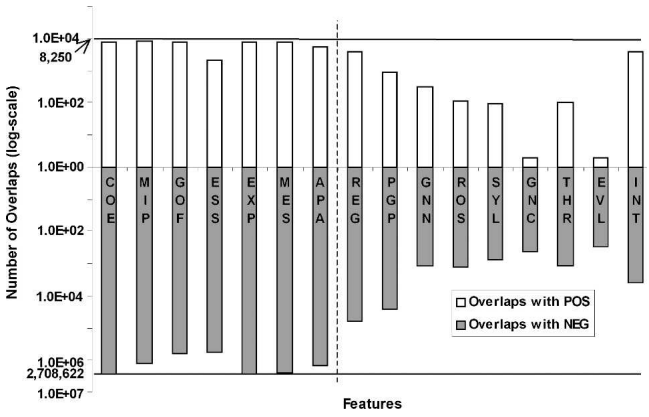

Figure 2. Overlaps between features and GSTDs. The blank and shaded columns represent the size of overlaps between the 16 features and the GSTD+ and GSTD−, respectively. The total numbers of protein pairs in the GSTD+ (8250) and GSTD− (2,708,622) are marked by two horizontal lines. Each of the seven features to the *left* of the dashed divider has at least 20% coverage of the GSTDs (positive and negative combined). Note that the plot is in log-scale; therefore, the APA column actually represents 23 times more protein pairs than REG column.

because they each have only two overlaps with the positive GSTD, and are thus unsuitable for this test. Because of the low coverage of these group-two features, the results in Figure 3B may be misleading without a careful interpretation. For example, SYL covers only 887 protein pairs in the GSTDs, it is thus unreliable to estimate its overall predictive power based on this 0.04% of the GSTDs when its coverage is likely to increase in the future (Fig. 3B).

Another point we need to pay attention to is that we should not take the performance of a feature against the GSTDs as indicative of the accuracy or usefulness of the feature in its original context. This is because the performance of a feature against the GSTDs only measures its usefulness in relation to a specific task—i.e., predicting complex membership—which is probably not what the feature was originally designed to do. For example, multimeric threading method is designed for predicting physical interactions between two proteins. However, because of the way the GSTDs are constructed, the majority of protein pairs in the GSTDs are simply in the same molecular complex without direct contacts. Therefore, when predicting physical interactions, these GSTDs are not a good means of judging the accuracy or usefulness of the multimeric threading method.

Quite often, only the TPR for a specific FPR is valued. For example, COE outperforms MIP until the FPR reaches 5%, even though MIP covers more area in the whole range of FPR. Thus, the features can also be ranked and selected according to the acceptable FPR in prediction.

Feature selection and improvement of performance

Because of the varying quality and predictive powers of genomic features, incorporating all features without selection will likely decrease the predictive power by introducing noise rather than improving the results. Therefore, we select only those new features with high predictive power based on the performance of individual features. Another factor we need to take into account is the coverage of features. It is obvious that there is a distinct difference between the features to the left and right of the divider in Figure 2; each of the first seven features covers at least a half million (~20%) ORF pairs in the GSTDs, while the next most populous feature (REG) covers only 2%. Even though some of the features with very low coverage show strong predictive power, whether or not that predictive power will remain is in question once the coverage increases in the future. Therefore, at the current stage, only the first seven features (i.e., F1–F7) are considered in the following calculation. The new features are EXP, MES, and APA.

The performance of combining new features is presented in Figure 4A by a ROC curve. By integrating the three additional features in the range of all FPR values, we obtain a better performance in the predictive power (higher TPR at a certain FPR value) than by integrating the four original features. However, such improvement is marginal; although each of the three new features shows a fairly strong predictive power, the increase of TPR at any value of FPR is no more than 3%.

Because of the dominant performance of the two functional similarity features (MIP and GOF), the improvement accomplished by incorporating new features may not seem obvious. We thus exclude these two functional features, showing the improvement by incorporating three additional features over the

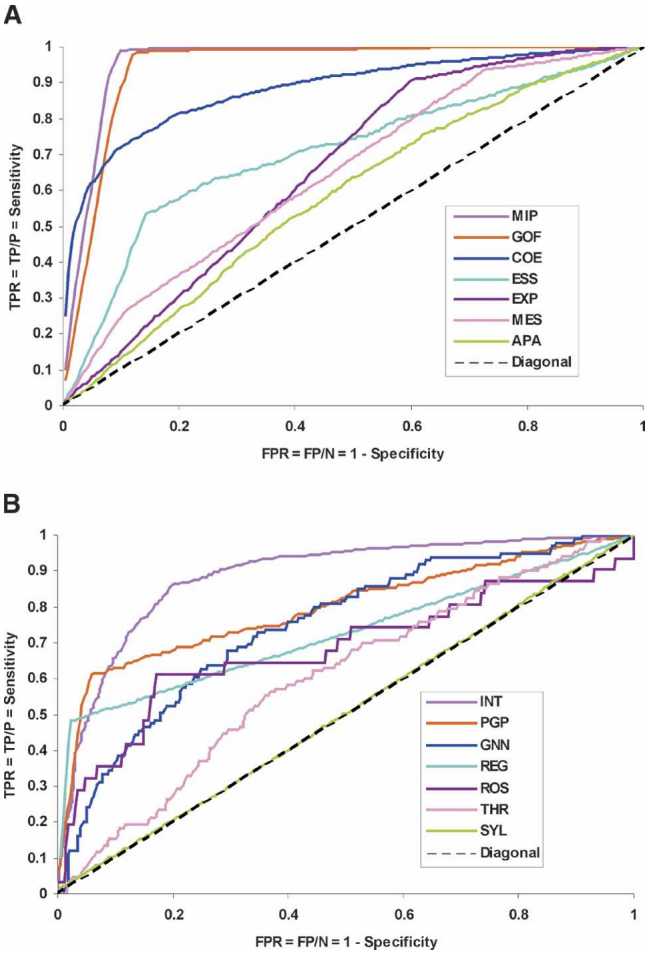

Figure 3. Predictive power of individual features illustrated by ROC curves. We plot ROC curves for individual features in two panels; the seven most populous features in *A*, and the remaining nine features in *B*. The acronyms signify the following: (TPR) True positive rate; (FPR) false positive rate; (TP) true positives; (FP) false positives; (P) total number of positives; (N) total number of negatives (see Methods).

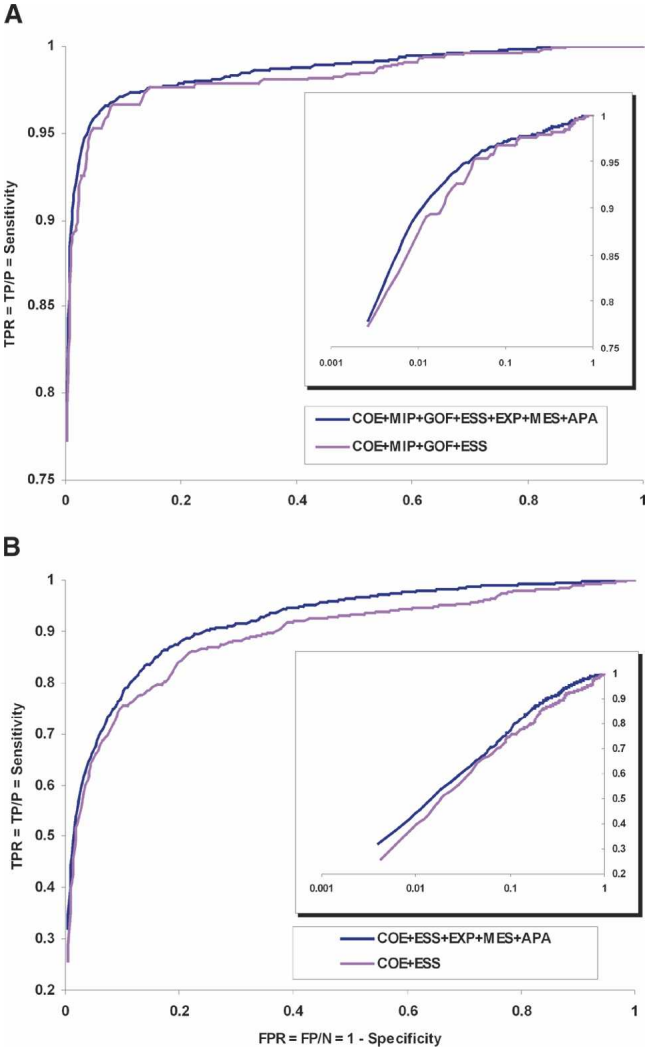

Figure 4. Integration of three additional features versus: (*A*) Four original features. Integration of three additional features (EXP, MES, APA) shows an improvement over the original four features at all range of FPRs. (*B*) Two original features. By excluding the two strongest features (MIP, GOF), it becomes more obvious that integrating three additional features outperforms the original two features. The *insets* are a closer look at the small FPR region by taking a log-scale of the *x*-axis. TPR, FPR, TP, FP, P, N are the same as in Figure 3.

remaining two original features (i.e., COE and ESS). Including three additional features shows a significant improvement over the original two features (Fig. 4B).

Another benefit of genomic data integration is the improvement in coverage; by incorporating more features, two predictors with similar ROC curve performance may cover different parts of the system to varying degrees. Note, it is the coverage of not only the labeled pairs (GSTDs), but also unlabeled pairs (unseen pairs). So far, our assessments have been done for labeled pairs only; however, if additional features allow the predictor to have a more extensive view of the system despite no significant improvement in ROC curve, they probably should be considered as beneficial, because in this case, the coverage of unlabeled pairs is improved. Here, we find the coverage is slightly improved by integrating more features. For all possible 21,658,071 protein pairs (6582 ORFs from MIPS), the four original features cover 18,527,741 pairs (85.5%), whereas the seven most populous features cover 18,880,102 (87.2%).

Correlations and statistical dependence between features

In this section, we investigate whether or not the marginality of improvement is confounded by the correlation and dependencies between features.

We first calculate the Pearson correlation coefficients (CCs) between each pair of features. Such correlations between features can often generate useful biological insights. The five highest absolute values are highlighted in bold in Table 1A. None of the feature pairs exhibit significant correlation.

In addition, we calculate mutual information between genomic features as an alternative to CCs. Whereas CC only measures linear relationships, mutual information is a more general measure of correlation. The results show an agreement with Ccs. The five pairs containing the most mutual information are exactly the same as those of the CCs. These correlations between some of the features, albeit not strong, are expected. For example, the correlations between the two functional features (MIP and GOF) are the highest among feature pairs. It is also expected that absolute mRNA expression (EXP) and absolute protein abundance (APA) are somewhat correlated.

We next investigate the conditional dependence between features given the positive or negative GSTD by calculating mutual information. In other words, we calculate the mutual information between pairs of features by taking into account only protein pairs that occur in both features and in either set of GSTDs. The small amount of mutual information, given either set of GSTDs, indicates that the features we integrated by Naive Bayes classifier are largely conditionally independent (Table 1B).

Simple Naive Bayes classifier vs. boosted Naive Bayes classifier on data sets with or without high dependence

Even though the conditional dependence between our features is not strong, it is possible that the combined weak dependence can still significantly decrease the predictive power of a Naive Bayes classifier. In this section, we address this question by comparing the performance of a simple Naive Bayes classifier (SNB) with that of a boosted Naive Bayes classifier (BNB). Since a BNB is fairly resistant to feature dependence, a significantly worse performance by a SNB on the same data set means that the feature dependence does affect the predictive power of the SNB.

We first conduct a control experiment with highly depen-

Table 1A. Absolute values of Pearson Correlation coefficients and mutual information between genomic features

MI × 100 \ CCs	COE	MIP	GOF	ESS	EXP	MES	APA	GSTDs
COE		0.08	0.08	0.05	0.04	0.00	0.03	0.11
MIP	0.45		**0.37**	**0.08**	0.04	0.05	0.02	0.21
GOF	0.69	**10.97**		**0.13**	0.05	0.04	0.04	0.18
ESS	0.63	**1.58**	**2.05**		0.01	**0.13**	0.00	0.05
EXP	0.17	0.26	0.30	0.05		0.03	**0.37**	0.03
MES	0.03	0.51	0.58	**7.31**	0.12		0.01	0.03
APA	0.12	0.06	0.19	0.04	**8.81**	0.06		0.02
GSTDs	0.71	2.01	3.30	0.21	0.09	0.08	0.02	

(CCs) Pearson Correlation coefficients; (MI) mutual information; (GSTDs) gold-standard data set. The five highest absolute values in each category are highlighted in bold.

Table 1B. Conditional mutual information[a] between genomic features

NEG × 100 \ POS × 100	COE	MIP	GOF	ESS	EXP	MES	APA
COE		**22.64**	**29.88**	7.11	15.29	12.09	14.70
MIP	0.17		**59.01**	16.26	6.31	9.40	6.26
GOF	0.34	**8.24**		**28.16**	5.73	11.18	5.81
ESS	0.78	**0.90**	**0.78**		2.09	**20.67**	2.81
EXP	0.14	0.38	0.58	0.05		8.86	12.75
MES	0.07	0.55	0.73	**6.74**	0.20		9.65
APA	0.10	0.05	0.22	0.05	**10.62**	0.09	

(CCs) Pearson Correlation coefficients; (MI) mutual information; (GSTDs) gold-standard data sets; (POS) GSTD+; (NEG) GSTD−.
[a]For a given feature pair, conditional mutual information for the GSTD+ (GSTD−) is computed by considering only protein pairs in the GSTD+ (GSTD−). The five highest absolute values in each category are highlighted in bold.

dent features to verify the resistance of BNB to feature dependence. To obtain a highly dependent set of features, we used mRNA expression data from microarray experiments conducted by Cho et al. (1998) under eight different conditions. Such expression data are highly dependent with regard to high CCs—the minimum CC between each pair of conditions is 0.904, the maximum CC is 0.970. Treating these eight sets of expression data as if they were eight features, we integrate them with the original four features. When evaluated on this highly dependent data set, the BNB significantly outperforms the SNB. Figure 5 shows the robustness of the BNB on this highly dependent data set.

We then compare a SNB with a BNB on our data set, with only weak conditional dependence; the original four features plus only one instead of eight sets of expression data. If the BNB significantly outperforms the SNB, it indicates that the SNB is affected by feature dependence, even though it is not strong. The results show that the SNB performs as well as the BNB on this weakly dependent data set (Fig. 5). Clearly, the SNB is hardly affected by this weak feature dependence.

The results in Figure 5 also suggest that the SNB performs sufficiently well on our collection of genomic features, while the BNB may be useful to analyze the potential problem of highly dependent features as more features are considered in the future.

Conclusions

In this study, we quantitatively address the question of how far genomic data integration can be improved by integrating more and more features. We use a SNB for integrating diverse sources of genomic evidence, ranging from coexpression relationships to similar phylogenetic profiles. By integrating three more strong features, marginal improvement on both accuracy and coverage can be achieved.

The calculations of correlation coefficients, mutual information, and boosting all suggest that the marginality of the improvement on prediction by incorporating more features is unlikely to result from the weak feature dependencies. It is also unlikely to result from an excess of parameters, relative to data points (resulting in overfitting), because our Naive Bayes approach involves simple models with only small numbers of free parameters that are fitted against a large number of data points. Rather, this suggests that by integrating a few good features, we approach the maximal predictive power, or limit, of current genomic data integration. Furthermore, this limitation does not reflect (potentially removable) inter-relationships between the features. Unless we obtain features that are stronger in predictive power than MIP and GOF and simultaneously possess a reasonable coverage, it is unlikely that the prediction will be significantly improved by integrating a few more features. It is also possible that a higher coverage of our examined 16 features may allow better predictive power in the future.

Our discovery that no strong dependence exists between features is an interesting finding in and of itself. Among as many as seven populous features, one might expect some dependence high enough to significantly decrease SNB's predictive power. However, our calculation on correlation coefficients and mutual information, as well as our boosting results, suggest otherwise. One possibility is that the observed lack of dependence among different features may result from differences in coverage, since all of these data sets are essentially incomplete. Specifically, the overlap of proteins or protein pairs represented among the different features is likely to increase with extended coverage and possibly results in higher feature dependence. In this case, the BNB can be used as an alternative solution.

Finally, SNB is chosen in this study because of its simplicity, as well as the ability to compare with an existing benchmark study using the same technique (Jansen et al. 2003). Furthermore, we use BNB to specifically address SNB's well-known limitation relating to high feature dependency.

Other machine-learning techniques could have been potentially used in this study. However, most alternative techniques have issues in their own right, such as suffering from the missing value problems or being prohibitively time-consuming. Such problems prevent them from being applied to this problem as readily as a SNB. In addition, since BNB does not improve SNB on our collection of features, it is probably not the case that the conclusions made here will be significantly different if other machine-learning techniques are used—though, of course, we cannot definitely say this without a comprehensive test.

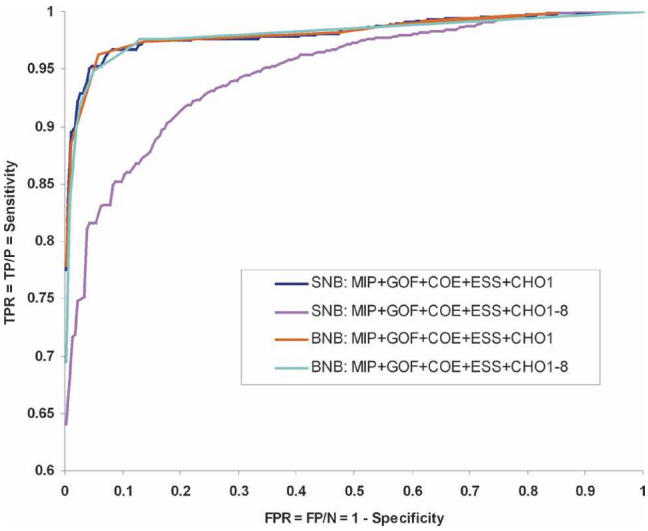

Figure 5. A SNB versus a BNB over sets of genomic features with or without high dependence. TPR, FPR, TP, FP, P, and N are the same as in Figure 3.

Methods

Naive Bayesian formalism

Inferring protein–protein interactions from genomic features can be formulated as a classification problem, in which we classify a pair of proteins into two classes (C_1 = interact, C_0 = not interact), given an n-dimensional vector of genomic features $\mathbf{x} = (x_1,x_2,\ldots,x_n)$.[5]

The Bayesian Decision Rule states that in order to minimize the average probability of a classification error, one must choose the class with the highest posterior probability, i.e., assign a feature vector $\mathbf{x}$ to the class C_k, such that: $C_k = \arg_{C_i}\max P(C_i \mid \mathbf{x})$, where C_i ranges over the set of classes (see for example, Bishop 1995; Duda et al. 2001). C_k is known as the maximum a posteriori (MAP) estimate.

Using Bayes theorem, the posterior probability can be rewritten, as

$$P(C_k \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid C_k) \cdot P(C_k)}{p(\mathbf{x})}.$$

Notice that the unconditional density $p(\mathbf{x})$ in the denominator does not depend on the class label; therefore, it does not affect the classification decision and can be omitted when computing $C_k = \arg_{C_i}\max P(C_i \mid \mathbf{x})$. Each of the priors, $P(C_i)$, can be easily estimated by computing the frequency with which each class occurs in the data. However, the evaluation of $p(\mathbf{x} \mid C_i)$ cannot generally be accomplished in the same way, especially if the number of features is high; it would require a set of data large enough to contain many instances for each possible combination of feature values, in order to obtain reliable estimates.

The idea behind Naive Bayes is to make the simplifying assumption that the attribute values are conditionally independent, given the target values. The computation of each is thus made efficient by approximating it as a product of conditional probabilities

$$\begin{aligned} p(\mathbf{x} \mid C_i) &= p(x_1, x_2, \ldots, x_n \mid C_i) \\ &\approx p(x_1 \mid C_i)p(x_2 \mid C_i) \ldots p(x_n \mid C_i) \\ &= \prod_j p(x_j \mid C_i). \end{aligned} \quad (1)$$

Learning in Naive Bayes consists of estimating the various $P(C_i)$ and various $p(x_j \mid C_i)$ using equation 1, based on their frequencies over the training data. Clearly, the approximation in equation 1 becomes exact only in the event of stochastic independence between the various features, given the class. In spite of its simple way of approximating the posterior distributions, Naive Bayes has, in practice, yielded quite good results for several types of problems; for example, it is among the best methods for text classification (Joachims 1997; McCallum and Nigam 1998).

In the case of stochastic independence, the covariance between two features is zero. Thus, the covariance between features is a measure of the deviation from the condition of stochastic independence and is indicative of the amount of approximation introduced by the Naive Bayes assumption. For this reason, the next section shall present an analysis of the covariance between the various features, given the class.

Alternatively, the Bayesian Decision rule for two classes can be stated thusly:

- If $\dfrac{p(\mathbf{x} \mid C_1) \cdot P(C_1)}{p(\mathbf{x} \mid C_0) \cdot P(C_0)} > 1$ then choose class C_1 (2)
- Otherwise, choose class C_0.

If we then introduce the Naive Bayes approximation, we can rewrite equation 2 as:

$$\frac{p(x_1 \mid C_1) \cdot p(x_2 \mid C_1) \ldots p(x_n \mid C_1) \cdot p(C_1)}{p(x_1 \mid C_1) \cdot p(x_2 \mid C_0) \ldots p(x_n \mid C_0) \cdot p(C_0)} > 1; \quad L_1 \cdot L_2 \cdot \ldots \cdot L_n > \frac{P(C_0)}{P(C_1)} \quad (3)$$

where

$$L_i \equiv \frac{p(x_i \mid C_1)}{p(x_i \mid C_0)}$$

and are called Likelihood Ratio for feature *i*. Notice that for a given feature, a likelihood ratio different than 1 indicates that the feature conveys information about the class. In other words, there is a correlation between the feature and the target. For this reason, in the next section we shall look at the likelihood ratios of the various features and the correlation between such features and the class labels.

ROC (receiver operating characteristic) curve

In a two-class classification problem, with classes C_1 (or positive) and C_0 (or negative), for each prediction there are four possible outcomes. The true positives (TP) and the true negatives (TN) are correct classifications. Wrong classifications can be of two types. For a false positive (FP), the outcome is incorrectly predicted as belonging to C_1, when in fact it belongs to C_0; for a false negative (FN), the outcome is incorrectly predicted as belonging to C_0, when it belongs to C_1.

Our earlier discussion on Naive Bayes was motivated by the goal of minimizing the average probability of a classification error; it was aimed at reducing the total number of wrong predictions, regardless of the type of error that was made. This amounts to saying that we were maximizing the number of

$$\frac{TP + TN}{TP + TN + FP + FN}.$$

In general, however, the two different types of errors will have different costs, just as the two different types of correct classification will have different benefits. Taking such costs into account amounts to multiplying the right hand side of equation 3 by a cost factor. In practice, these costs are rarely known with accuracy. Thus, to evaluate a classification method, it is useful to look at its ROC curve.

A ROC curve graphically depicts the performance of a classification method for different costs. It consists of a set of points, each computed for a different setting of the cost, connected by lines. For each point, the vertical coordinate is a true positive rate (TPR) given by the ratio of the number of true positives to the total number of positives (i.e., $TP/[TP+FN]$), while the horizontal coordinate is a false positive rate (FPR) given by the ratio of the number of false positives to the total number of negatives (i.e., $FP/[FP+TN]$). Note that the TPR is equivalent to the commonly used term sensitivity, while FPR is equivalent to 1—specificity. Clearly, the ROC curve for a good classifier will be as close as possible to the upper-left corner of the chart; that is where we have the highest number of true positives and at the same time the smallest number of false positives.

[5]Bold letters denote vectors; $P(\cdot)$ denote probabilities; $p(\cdot)$ denote probability density functions.

Mutual information

Given two random variables, X and Y (in this study, X and Y are either feature values or class labels), the Mutual Information $I(X;Y)$ between X and Y measures how much information one variable conveys about the other one. It is defined as the relative entropy (or Kullback-Leibler distance) between the joint distribution and the product distribution of X and Y, that is

$$I(X;Y) = \sum_x \sum_y P(x, y) \log \frac{P(x, y)}{P(x)P(y)},$$

where $P(x, y)$ indicates the joint distribution of X and Y and $P(x)$ and $P(y)$ their marginal distributions. It is easy to prove that $I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) = I(Y;X)$, where $H(X)$ and $H(Y)$ are the entropies of X and Y, and $H(X|Y)$ and $H(Y|X)$ are the conditional entropies of X given Y and Y given X, respectively. This states that the information Y conveys about X is the reduction in uncertainty about X, due to knowledge of Y (and vice-versa).

Boosting

Boosting is a general method that can be used for improving the performance of any classifier. The idea behind boosting is to combine the outputs of many different "weak" classifiers to produce a powerful "committee." We have used one of the most popular boosting algorithms, AdaBoost (Freund and Schapire 1999), which we shall briefly describe here. For more information on this and other boosting algorithms refer to Friedman et al. 2000.

AdaBoost consists of sequentially applying a weak classification algorithm to modified versions of the data, producing a sequence of weak classifiers. Then, the prediction from each classifier is combined through a weighted majority vote. The data is modified by applying weights to each of the training observations. At each iteration, a weak learner is trained on the weighted set of data and the weights are updated. This operation is repeated until the desired performance for the training data is achieved. The updating rule for these weights is such that training pairs that had been misclassified in the previous step will have their weights increased, while those that were correctly classified will have their weights decreased. At each iteration, then, training pairs that are more difficult to classify have more influence, and classifiers are forced to focus on pairs overlooked by previous classifiers.

Given a data set of N training pairs $(\mathbf{x}_i, y_i)$, $i = 1 \ldots N$, where $\mathbf{x}_i$ is an input vector of features and $y_i \in \{-1, 1\}$ is the target value representing classes C_0 and C_1, respectively, let us denote the weight associated with training pair i at time t as $D_t(i)$, and the weak classification algorithm used at time t as h_t. The AdaBoost algorithm to iterate T times is as follows:

- Initialize the observation weights for each pair

$$D_1(i) = \frac{1}{N}$$

- For $t = 1 \ldots T$ do:

  1. Train h_t using the training pairs weighted by D_t
  2. Compute E_t, the global error of h_t as:

$$E_t = \sum_{i:h_t(\mathbf{x}_i) \neq y_i} D_t(i)$$

  3. Set $\alpha_t = \frac{1}{2} \ln\left(\frac{1 - E_t}{E_t}\right)$

  4. $D_{t+1}(i) = \frac{D_t(i) \cdot e^{-\alpha_t y_i h_t(\mathbf{x}_i)}}{Z_t}$

  where Z_t is a normalization factor such that

$$\sum_i D_{t+1}(i) = 1$$

- The output of the final classifier is:

$$H(\mathbf{x}) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(\mathbf{x})\right)$$

Training and testing data sets

The details of construction of the training and testing data sets are described in Figure 1.

Acknowledgments

We thank Drs. Ronald Jansen, Valery Trifonov, and Haoxin Lu for stimulating discussions and proofreading of this manuscript. Y.X. is a Fellow of the Jane Coffin Childs Memorial Fund for Medical Research. This work is supported by a grant from NIH/NIGMS for work in the PSI.

References

Alberts, B. 2002. *Molecular biology of the cell*. Garland Science, New York.
Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., et al. 2000. Gene ontology: Tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.* **25:** 25–29.
Baker, D. and Sali, A. 2001. Protein structure prediction and structural genomics. *Science* **294:** 93–96.
Berger, J.M., Gamblin, S.J., Harrison, S.C., and Wang, J.C. 1996. Structure and mechanism of DNA topoisomerase II. *Nature* **379:** 225–232.
Bishop, C.M. 1995. *Neural networks for pattern recognition*. Clarendon Press, Oxford University Press, Oxford, UK.
Bowers, P.M., Pellegrini, M., Thompson, M.J., Fierro, J., Yeates, T.O., and Eisenberg, D. 2004. Prolinks: A database of protein functional linkages derived from coevolution. *Genome Biol.* **5:** R35.
Brown, M.P., Grundy, W.N., Lin, D., Cristianini, N., Sugnet, C.W., Furey, T.S., Ares Jr., M., and Haussler, D. 2000. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc. Natl. Acad. Sci.* **97:** 262–267.
Cho, R.J., Campbell, M.J., Winzeler, E.A., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T.G., Gabrielian, A.E., Landsman, D., Lockhart, D.J., et al. 1998. A genome-wide transcriptional analysis of the mitotic cell cycle. *Mol. Cell* **2:** 65–73.
Drawid, A., Jansen, R., and Gerstein, M. 2000. Genome-wide analysis relating expression level with protein subcellular localization. *Trends Genet.* **16:** 426–430.
Duda, R.O., Hart, P.E., and Stork, D.G. 2001. *Pattern classification* Wiley, New York; Chichester, UK.
Eisenberg, D., Marcotte, E.M., Xenarios, I., and Yeates, T.O. 2000. Protein function in the post-genomic era. *Nature* **405:** 823–826.
Freund, Y. and Schapire, R.E. 1996. Experiments with a new boosting algorithm. In *Proceedings of the thirteenth conference on machine learning*, pp. 148–156.
———. 1999. A short introduction to boosting. *J. Japanese Soc. Artificial Intell.* **14:** 771–780.
Friedman, N. 2004. Inferring cellular networks using probabilistic graphical models. *Science* **303:** 799–805.
Friedman, J., Hastie, T., and Tibshirani, R. 2000. Additive logistic regression: A statistical view of boosting, *Ann. Stat.* **28:** 337–374.
Gavin, A.C., Bosche, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A., Schultz, J., Rick, J.M., Michon, A.M., Cruciat, C.M., et al. 2002. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature* **415:** 141–147.
Ge, H., Liu, Z., Church, G.M., and Vidal, M. 2001. Correlation between transcriptome and interactome mapping data from *Saccharomyces cerevisiae*. *Nat. Genet.* **29:** 482–486.

Gerstein, M., Lan, N., and Jansen, R. 2002. Proteomics. Integrating interactomes. *Science* **295:** 284–287.
Goh, C.S. and Cohen, F.E. 2002. Co-evolutionary analysis reveals insights into protein–protein interactions. *J. Mol. Biol.* **324:** 177–192.
Goh, C.S., Bogan, A.A., Joachimiak, M., Walther, D., and Cohen, F.E. 2000. Co-evolution of proteins with their interaction partners. *J. Mol. Biol.* **299:** 283–293.
Greenbaum, D., Jansen, R., and Gerstein, M. 2002. Analysis of mRNA expression and protein abundance data: An approach for the comparison of the enrichment of features in the cellular population of proteins and transcripts. *Bioinformatics* **18:** 585–596.
Greenbaum, D., Colangelo, C., Williams, K., and Gerstein, M. 2003. Comparing protein abundance and mRNA expression levels on a genomic scale. *Genome Biol.* **4:** 117.
Hartwell, L.H., Hopfield, J.J., Leibler, S., and Murray, A.W. 1999. From molecular to modular cell biology. *Nature* **402:** C47–C52.
Ho, Y., Gruhler, A., Heilbut, A., Bader, G.D., Moore, L., Adams, S.L., Millar, A., Taylor, P., Bennett, K., Boutilier, K., et al. 2002. Systematic identification of protein complexes in *Saccharomyces cerevisiae* by mass spectrometry. *Nature* **415:** 180–183.
Horak, C.E. and Snyder, M. 2002. ChIP-chip: A genomic approach for identifying transcription factor binding sites. *Methods Enzymol.* **350:** 469–483.
Ideker, T., Thorsson, V., Ranish, J.A., Christmas, R., Buhler, J., Eng, J.K., Bumgarner, R., Goodlett, D.R., Aerbersold, R., and Hood, L. 2001. Integrated genomic and proteomic analyses of a systematically perturbed metabolic network. *Science* **292:** 929–934.
Ito, T., Chiba, T., Ozawa, R., Yoshida, M., Hattori, M., and Sakaki, Y. 2001. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proc. Natl. Acad. Sci.* **98:** 4569–4574.
Jansen, R., Greenbaum, D., and Gerstein, M. 2002a. Relating whole-genome expression data with protein–protein interactions. *Genome Res.* **12:** 37–46.
Jansen, R., Lan, N., Qian, J., and Gerstein, M. 2002b. Integration of genomic datasets to predict protein complexes in yeast. *J. Struct. Funct. Genomics* **2:** 71–81.
Jansen, R., Yu, H., Greenbaum, D., Kluger, Y., Krogan, N.J., Chung, S., Emili, A., Snyder, M., Greenblatt, J.F., and Gerstein, M. 2003. A Bayesian networks approach for predicting protein–protein interactions from genomic data. *Science* **302:** 449–453.
Joachims, T. 1997. A probabilistic analysis of the Rocchio Algorithm with TFIDF for text categorization. *14th International Conference on Machine Learning*.
Kemmeren, P., van Berkum, N.L., Vilo, J., Bijma, T., Donders, R., Brazma, A., and Holstege, F.C. 2002. Protein interaction verification and functional annotation by integrated analysis of genome-scale data. *Mol. Cell.* **9:** 1133–1143.
Lee, T.I., Rinaldi, N.J., Robert, F., Odom, D.T., Bar-Joseph, Z., Gerber, G.K., Hannett, N.M., Harbison, C.T., Thompson, C.M., Simon, T., et al. 2002. Transcriptional regulatory networks in *Saccharomyces cerevisiae*. *Science* **298:** 799–804.
Lee, I., Date, S.V., Adai, A.T., and Marcotte, E.M. 2004. A probabilistic functional network of yeast genes. *Science* **306:** 1555–1558.
Letovsky, S. and Kasif, S. 2003. Predicting protein function from protein/protein interaction data: A probabilistic approach. *Bioinformatics* **19:** 197–204.
Lin, N., Wu, B., Jansen, R., Gerstein, M., and Zhao, H. 2004. Information assessment on predicting protein–protein interactions. *BMC Bioinformatics* **5:** 154.
Lu, L., Lu, H., and Skolnick, J. 2002. MULTIPROSPECTOR: An algorithm for the prediction of protein–protein interactions by multimeric threading. *Proteins* **49:** 350–364.
Lu, L., Arakaki, A.K., Lu, H., and Skolnick, J. 2003. Multimeric threading-based prediction of protein–protein interactions on a genomic scale: Application to the *Saccharomyces cerevisiae* proteome. *Genome Res.* **13:** 1146–1154.
Marcotte, E.M., Pellegrini, M., Ng, H.L., Rice, D.W., Yeates, T.O., and Eisenberg, D. 1999a. Detecting protein function and protein–protein interactions from genome sequences. *Science* **285:** 751–753.
Marcotte, E.M., Pellegrini, M., Thompson, M.J., Yeates, T.O., and Eisenberg, D. 1999b. A combined algorithm for genome-wide prediction of protein function. *Nature* **402:** 83–86.
Martone, R., Euskirchen, G., Bertone, P., Hartman, S., Royce, T.E., Luscombe, N.M., Rinn, J.L., Nelson, F.K., Miller, P., Gerstein, M., et al. 2003. Distribution of NF-κB-binding sites across human chromosome 22. *Proc. Natl. Acad. Sci.* **100:** 12247–12252.
McCallum, A. and Nigam, K. 1998. A comparison of event models for Naive Bayes text classification. *AAAI/ICML-98 Workshop on Learning for Text Categorization*, pp. 41–48.
Mewes, H.W., Frishman, D., Guldener, U., Mannhaupt, G., Mayer, K., Mokrejs, M., Morganstern, B., Munsterkotter, M., Rudd, S., and Weil, B. 2002. MIPS: A database for genomes and protein sequences. *Nucleic Acids Res.* **30:** 31–34.
Pazos, F. and Valencia, A. 2002. In silico two-hybrid system for the selection of physically interacting protein pairs. *Proteins* **47:** 219–227.
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. *Proc. Natl. Acad. Sci.* **96:** 4285–4288.
Schapire, R.E. 1990. The strength of weak learnability. *Machine Learning* **5:** 197–227.
Schwikowski, B., Uetz, P., and Fields, S. 2000. A network of protein–protein interactions in yeast. *Nat. Biotechnol.* **18:** 1257–1261.
Skolnick, J. and Kolinski, A. 2002. In *Computational methods for protein folding*. Vol. 120 (ed. R.A. Friesner), pp. 131–192. John Wiley & Sons, New York.
Strong, M., Mallick, P., Pellegrini, M., Thompson, M.J., and Eisenberg, D. 2003. Inference of protein function and protein linkages in *Mycobacterium tuberculosis* based on prokaryotic genome organization: A combined computational approach. *Genome Biol.* **4:** R59.
Tamames, J., Casari, G., Ouzounis, C., and Valencia, A. 1997. Conserved clusters of functionally related genes in two bacterial genomes. *J. Mol. Evol.* **44:** 66–73.
Thatcher, J.W., Shaw, J.M., and Dickinson, W.J. 1998. Marginal fitness contributions of nonessential genes in yeast. *Proc. Natl. Acad. Sci.* **95:** 253–257.
Tong, A.H., Evangelista, M., Parsons, A.B., Xu, H., Bader, G.D., Page, N., Robinson, M., Raghibizadeh, S., Hogue, C.W., Bussey, H., et al. 2001. Systematic genetic analysis with ordered arrays of yeast deletion mutants. *Science* **294:** 2364–2368.
Tong, A.H., Lesage, G., Bader, G.D., Ding, H., Xu, H., Xin, X., Young, J., Berriz, G.F., Brost, R.L., Chang, M., et al. 2004. Global mapping of the yeast genetic interaction network. *Science* **303:** 808–813.
Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., and Altman, R.B. 2001. Missing value estimation methods for DNA microarrays. *Bioinformatics.* **17:** 520–525.
Uetz, P., Giot, L., Cagney, G., Mansfield, T.A., Judson, R.S., Knight, J.R., Lockshon, D., Narayan, V., Srinivasan, M., Pochart, P., et al. 2000. A comprehensive analysis of protein–protein interactions in *Saccharomyces cerevisiae*. *Nature* **403:** 623–627.
Valencia, A. and Pazos, F. 2002. Computational methods for the prediction of protein interactions. *Curr. Opin. Struct. Biol.* **12:** 368–373.
Vazquez, A., Flammini, A., Maritan, A., and Vespignani, A. 2003. Global protein function prediction from protein–protein interaction networks. *Nat. Biotechnol.* **21:** 697–700.
von Mering, C., Krause, R., Snel, B., Cornell, M., Oliver, S.G., Fields, S., and Bork, P. 2002. Comparative assessment of large-scale data sets of protein–protein interactions. *Nature* **417:** 399–403.
Wong, S.L., Zhang, L.V., Tong, A.H., Li, Z., Goldberg, D.S., King, O.D., Lesage, G., Vidal, M., Andrews, B., Bussey, H., et al. 2004. Combining biological networks to predict genetic interactions. *Proc. Natl. Acad. Sci.* **101:** 15682–15687.
Xia, Y., Yu, H., Jansen, R., Seringhaus, M., Baxter, S., Greenbaum, D., Zhao, H., and Gerstein, M. 2004. Analyzing cellular biochemistry in terms of molecular networks. *Annu. Rev. Biochem.* **73:** 1051–1087.
Yu, H., Luscombe, N.M., Qian, J., and Gerstein, M. 2003. Genomic analysis of gene expression relationships in transcriptional regulatory networks. *Trends Genet.* **19:** 422–427.
Yu, H., Greenbaum, D., Xin Lu, H., Zhu, X., and Gerstein, M. 2004a. Genomic analysis of essentiality within protein networks. *Trends Genet.* **20:** 227–231.
Yu, H., Luscombe, N.M., Lu, H.X., Zhu, X., Xia, Y., Han, J.D., Bertin, N., Chung, S., Vidal, M., and Gerstein, M. 2004b. Annotation transfer between genomes: Protein–protein interologs and protein–DNA regulogs. *Genome Res.* **14:** 1107–1118.
Zhang, L.V., Wong, S.L., King, O.D., and Roth, F.P. 2004. Predicting co-complexed protein pairs using genomic and proteomic data integration, *BMC Bioinformatics* **5:** 38.

Received December 22, 2004; accepted in revised form May 2, 2005.