In Silico Biology 9 (2009) 23–34
DOI 10.3233/ISB-2009-0382
IOS Press

A Naive Bayes Classifier for Protein Function Prediction

Jukka Kohonen[a,*], Sarish Talikota[a], Jukka Corander[a], Petri Auvinen[b] and Elja Arjas[a]

[a]*Department of Mathematics and Statistics, University of Helsinki, Helsinki, FI-00014, Finland*
[b]*Institute of Biotechnology, University of Helsinki, Helsinki, FI-00014, Finland*

Edited by H. Michael; received 9 November 2007; revised 20 November 2008; accepted 24 November 2008; published 13 December 2008

ABSTRACT: A Naive Bayes classifier tool is presented for annotating proteins on the basis of amino acid motifs, cellular localization and protein-protein interactions. Annotations take the form of posterior probabilities within the Molecular Function hierarchy of the Gene Ontology (GO). Experiments with the data available for yeast, *Saccharomyces cerevisiae*, show that our prediction method can yield a relatively high level of accuracy. Several apparent challenges and possibilities for future developments are also discussed.

A common approach to functional characterization is to use sequence similarities at varying levels, by utilizing several existing databases and local alignment/identification algorithms. Such an approach is typically quite labor-intensive when performed by an expert in a manual fashion. Integration of several sources of information is in this context generally considered as the only possibility to obtain valuable predictions with practical implications. However, some improvements in the prediction accuracy of the molecular functions, and thereby also savings in the computational effort, can be achieved by restricting attention to only those data sources that involve a higher degree of specificity. We employ here a Naive Bayes model in order to provide probabilistic predictions, and to enable a computationally efficient approach to data integration.

KEYWORDS: Protein function prediction, Naive Bayes, data integration, Gene Ontology

INTRODUCTION

Prediction of the protein functions has become rapidly a high priority research area in the post-genomic era. By April 2007, more than 757 bacterial, 41 archaeal and 134 eukaryotic genomes had been sequenced, and approximately 1500 genomes are currently in the process of being sequenced (http://www.ncbi.nlm.nih.gov/sutils/genom_table.cgi). This vast amount of sequence information poses a major challenge for the decoding of the functionalities hidden inside it [Eisenberg *et al.*, 2000; Gabaldón and Huynen, 2004]. The genome sequence of *Saccharomyces cerevisiae*, the budding yeast, provides a classic example of the annotation deficiencies. In spite of this species being generally considered as the model organism of choice, and having been extensively studied, approximately 30% of the identified Open Reading Frames (ORFs) are still labeled as only hypothetical in the SGD (*S. cerevisiae* Genome Database, www.yeastgenome.org) annotation database. Relatively few hypothetical proteins have been characterized in detail during recent years, and thus the pace of this progress has become a major

*Corresponding author. E-mail: jukka.kohonen@helsinki.fi.

Electronic publication can be found in *In Silico Biol.* **9**, 0003 <http://www.bioinfo.de/isb/2008/09/0003/>, 13 December 2008 .

1386-6338/09/$17.00 © 2009 – IOS Press and Bioinformation Systems e.V. and the authors. All rights reserved

concern. To overcome such problems, efficient automated methods have a considerable potential in terms of generating reasonable hypotheses about the functionalities.

Numerous algorithmic and statistical methods have been applied to the putative characterization of proteins in the setting described above. Homology-based [Altschul *et al.*, 1997; Pearson and Lipman, 1998], neighborhood-based [Schwikowski *et al.*, 2000], probabilistic [Deng *et al.*, 2003; Letovsky and Kasif, 2003], and graph theoretic approaches [Vazquez *et al.*, 2003] are examples of direct methods that rely on the protein proximity. Machine learning with support vector machines [Lanckriet *et al.*, 2004; Barutcuoglu *et al.*, 2006] and Bayesian networks [Troyanskaya *et al.*, 2003] have recently been popularized, not only in the current context, but also as more general data mining and prediction tools. In the applications of these methods, several information sources and technologies have been utilized, such as the yeast-2-hybrid technology, DNA microarrays, synthetic lethality and protein complexes.

Gene Ontology (GO) [Ashburner *et al.*, 2000] provides a controlled vocabulary in a hierarchical fashion for the annotation of proteins with three ontologies, *Molecular Function* (MF), *Biological Process* (BP) and *Cellular Component* (CC). Since many proteins have multiple functions, the use of individual functional labels can be too restrictive; instead, utilization of the hierarchical structure of GO generally provides improved accuracy in prediction and reduces inconsistencies [Barutcuoglu *et al.*, 2006]. For unannotated proteins, sequence features are a primary source of information for predicting the molecular functions. However, a systematic integration of different sources of data is also without doubt vital for the quality of the predictions. Even so, it is not necessarily true that an increase in the number of protein features upon which the predictions are based will ultimately improve the accuracy of the predictions. Under many circumstances, the additional information may be confounding or simply represent some noise, and may thus affect adversely the prediction procedure. Depending on the category of GO (MF, BP, CC) that is the primary target of the predictions, it is highly beneficial to try to access the most relevant sources of information related to the protein characteristics. In the present illustration concerning *S. cerevisiae* we have used the following databases: InterPro [Apweiler *et al.*, 2001], Interactions [Stark *et al.*, 2006], and CC (GO).

To derive a computationally feasible approach to data integration for making probabilistic predictions, we exploit the generic principle of the so-called supervised Naive Bayesian classifier for discrete-valued data [Hand and Yu, 2001]. Such classifiers treat the data features as conditionally independent given the classification variable, which takes values in the finite set of the labels of the *a priori* determined classes of interest (here functional categories of proteins). In spite of the apparently restrictive nature of this assumption, such classifiers have been demonstrated to be able to achieve a surprisingly high degree of prediction accuracy in a wide variety of applied contexts. To demonstrate the general prediction ability of our classifier tool for the *S. cerevisiae* data, we use the commonly applied strategy of splitting the currently available annotation data into separate sets for training and testing. In addition, to illustrate the deeper challenges generally related to prediction problems in functional genomics, we also use GO annotations from October 2004 for making the predictions, and then utilize the currently available annotations for validatory purposes.

MATERIALS AND METHODS

Functional classes

The classes to be predicted correspond to the terms on the GO molecular function hierarchy. In contrast to, for example, Barutcuoglu *et al.*, 2006, we do not prune the class hierarchy according to any *a priori*

criteria such as number of annotations or biological relevance. Only empty classes and the root term were eliminated, leaving 934 functional (MF) classes (751 leaf classes and 183 intermediate, non-leaf classes).

Proteins that had more than one direct MF annotation were removed from the data, in order to facilitate our prediction model, which assumes mutually exclusive classes. Inclusion of multiply-annotated proteins is left for further research.

About one third of the annotated proteins had a non-leaf annotation. We note that such annotations may result from two conceptually different situations. First, a protein may be annotated to a generic term because it is not yet known which child term it belongs to. Second, the ontology itself may be incomplete; none of the existing child terms are applicable, but a new child term may be added later. For example, while the protein YIA6 was annotated to *transporter activity* in 2004, a new more specific term *NAD transporter activity* has since been created, and YIA6 is now annotated there. Since these two situations are not distinguishable from the annotation data itself, we treat them identically. If a protein is annotated to a non-leaf term X, which currently has children A and B, we treat the protein as annotated to an imaginary leaf term X, defined as "X excluding A and B".

Probability model

Our functional predictions are based on a Naive Bayes model (see, e.g., Hand and Yu, 2001). Each protein is assumed to belong to one of K mutually exclusive functional classes, as described above. Assuming that the classifications are known for a set of annotated proteins (*training data*), we wish to predict the classification for other proteins (*test data*).

Suppose that for proteins $i = 1,\ldots, n-1$ the classification is known, and we wish to classify protein n. For each protein considered ($i = 1,\ldots, n$), we introduce a class indicator $C_i \in \{1, \ldots, K\}$ and binary feature indicators X_{ij} for each feature ($j = 1,\ldots, m$). The characteristics of the utilized features will be explained in detail later. We assume that $C_i \sim$ Discrete ($\Psi_1, \ldots, \Psi_K$), and $(X_{ij}|C_i = k) \sim$ Bernoulli (Θ_{jk}), where Ψ_k and Θ_{jk} are latent parameters ($k = 0, 1$). In other words, Ψ_k is the prior probability of belonging to class k, and Θ_{jk} is the probability of the jth feature being present among the proteins in class k. The latent parameters are further assumed to have prior distributions $\Psi \sim$ Dirichlet($\alpha_1, \ldots, \alpha_K$) and $\Theta_{jk} \sim$ Beta(β_1, β_0). The structure of the model is illustrated in Fig. 1.

For any given $\Psi = \psi$ and $\Theta = \theta$, the likelihood of data (C, X) can now be written as

$$P(C = c, X = x|\Psi = \psi, \Theta = \theta) = \prod_{i=1}^{n} P(C_i = c_i|\Psi = \psi) \prod_{j=1}^{m} P(X_{ij} = x_{ij}|C_i = c_i, \Theta_j = \theta_j).$$

Conditioning on observed feature indicators X_{ij} (for all i, j) and class indicators $C_{\text{obs}} = (C_1, \ldots, C_{n-1})$, and integrating out Ψ and Θ, the posterior class probabilities are

$$P(C_n = k|X = x, C_{\text{obs}} = c_{\text{obs}}) \propto (\alpha_k - n'_k) \prod_{j=1}^{m} (\beta_l - n'_{jkl})$$

Here n'_k is the number of proteins observed in class k, and n'_{jkl} counts how many times the jth feature had the value $l = x_{nj}$ in class k in the observed data. The proof of Eq. (2) is relegated to the Appendix. Since the posterior class probability has such a simple analytic form, it can be computed very efficiently without resorting to sampling or approximation methods.

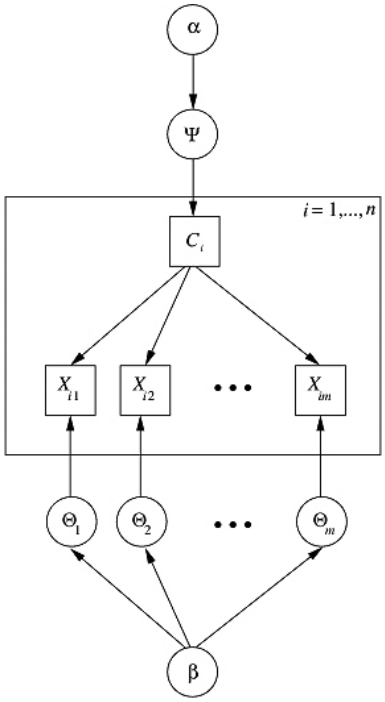

Fig. 1. Structure of the Naive Bayes model. Variables inside the box are replicated for $i = 1, \ldots, n$ ("plate notation"). Square nodes represent discrete variables: class indicators C_i and feature indicators X_{ij}. Circular nodes represent continuous variables: latent parameters (Ψ and Θ) and their hyperparameters (α and β).

The Naive Bayes feature model assumes that, within each class, the features are independently distributed. While this assumption is obviously not literally true, it can be a reasonable approximation especially if the classes are relatively small. This is one of the reasons for formulating the classification task at the leaf level of GO.

Noninformative class hyperparameters $\alpha_k = 1$ are used ($k = 1, \ldots, K$). In fact their exact values do not have great significance; the class posteriors are mostly determined by feature likelihood.

Feature hyperparameters β_1 and β_0, however, cannot be chosen to be noninformative, since the feature data are in this context extremely sparse. A practical choice that reasonably well fits to the observed features is to set the ratio of β_1 and β_0 to be equal to the observed overall feature density in the training data (separately for the three feature sets: motifs, interactions and localization). The flexibility of the model is then determined by the sum $\beta_1 + \beta_0$. In our experiments $\beta_1 + \beta_0$ was given the value 10.

We performed some sensitivity analysis by setting $\beta_1 + \beta_0$ to values 1, 10, 100, 1000, and 10000; values between 1 and 10 generally resulted in reasonable performance both in terms of sensitivity and precision; for higher values the model is too stiff to learn from the features in the training data, and predictive performance deteriorates considerably.

Data sources

Gene Ontology (*GO*)

Gene Ontology (GO) provides a detailed basis for protein annotations via a controlled vocabulary. Apart from MF, current GO annotations classify 2238 proteins as hypothetical in the sense of lacking

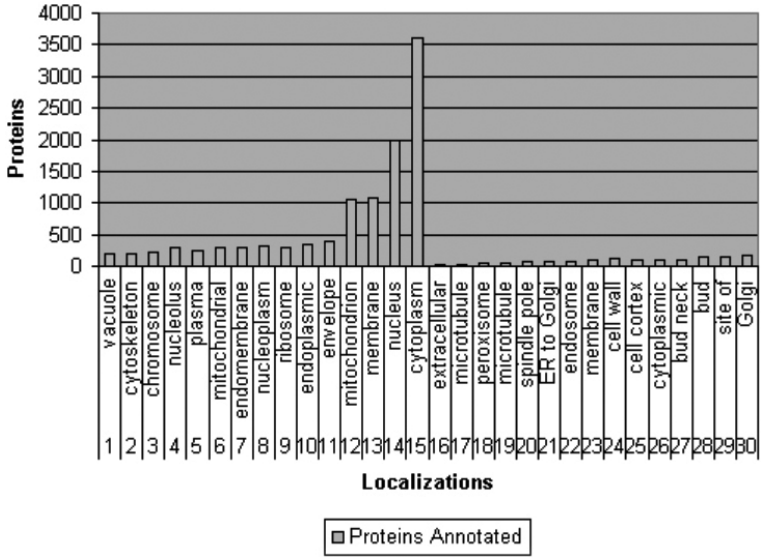

Fig. 2. Localization terms. Most of these terms were picked manually, based on their assumed relevance, and some were taken from SGD GO Slim Mapper (function ontology). 30 localization terms are included in the graph. (http://db.yeastgenome.org/cgi-bin/GO/goTermMapper).

BP, and 1779 in the sense of lacking CC. To build a testing framework for functional classification, the complete yeast protein set (5862 proteins) was downloaded from the SGD, and the GO hierarchy available on April 1st, 2007 was chosen.

Motifs

Motif information was taken from InterPro, which integrates various sequence features from different databases based on similar functionality. Note that not all proteins included in SGD have motifs in this database. Approximately 3000 InterPro yeast-specific identifiers (IPRs) have been identified, and each of these identifiers is considered as a feature (present in some proteins, and absent in others).

Interactions

The data were downloaded from GRID, which comprises both protein-protein interactions (PPI) and genetic interactions (GI). The interactions are pairwise and the evidence comes from 8 different kinds of experiments (yeast-2-hybrid, synthetic rescue, synthetic lethality, affinity chromatography, dosage lethality, purified complex, biological assay, and affinity precipitation). We downloaded the latest version of BIOGRID 2.0.25 (www.thebiogrid.org), which has $\sim$ 65000 interactions, of which $\sim$ 31000 are genetic, and the remaining $\sim$ 34500 are protein interactions.

Localization

The data represent sub-cellular localization of proteins. We made use of localization information from the GO cellular component hierarchy. Initially we considered using all CC terms for which there are at least 100 proteins annotated. However, many such terms are very generic (such as cell, cell part etc.) and contain duplicate information. As a result, we selected manually the most relevant terms, and then appended all the GO slim terms from SGD Slim Term Mapper, resulting in 30 localization terms (Fig. 2) which span a total of 5339 proteins. Each of these 30 terms was then considered as a particular feature of a protein.

The URLs concerning all the datasets are presented elsewhere, and for those not available, the data can be obtained from supporting websites of the authors. Our naming conventions are based on the SGD, and we have used genes and proteins interchangeably in our notation.

Data representation

All three sources of feature data were combined into a common representation in the form of a feature matrix $X = (X_{ij})$ such that $X_{ij} = 1$ if protein i has feature j, and $X_{ij} = 0$ otherwise. As a preprocessing step, null features (all-zero columns) were removed, and so were featureless proteins (all-zero rows). The resulting feature matrix has 8256 columns, consisting of 3075 motif features, 5151 interaction features, and 30 localization features. In certain experiments we only used a subset of the columns. The matrix is quite sparse since a typical protein has only a few motifs and a few interactions (if any).

The representation of interactions chosen here is similar to that used in other functional prediction methods [e.g. Barutcuoglu *et al.*, 2006], such that an interaction (either PPI or GI) between the proteins i and j was represented as protein i having the "interaction feature" j and *vice versa*.

Experimental setting

Our classification model was evaluated for the prediction of functions at molecular level for yeast *S. cerevisiae*. The performance of the classifiers was evaluated in two different experimental settings. In the *held-out* data setting, we randomly split currently available annotation data into separate equal-sized training and test sets (1661 proteins each). Predictions were generated based on the training data and then evaluated against the test data. Each test protein was assigned to the maximum a posteriori (MAP) class among leaf and imaginary leaf terms.

Leaf predictions were then aggregated into higher level terms according to the GO hierarchy; for example, a test protein was classified into *ligase activity* whenever its MAP prediction was any of the 47 descendant terms of ligase activity. Note that while the leaf terms are mutually exclusive, the aggregate terms are not. For each aggregate term, prediction quality was assessed by means of sensitivity (true positive rate), specificity (true negative rate), and precision (positive predictive value).

In the *delayed data* setting, we used the GO annotations from September 2004 as the training data, and generated functional predictions for those proteins that were still unannotated in that data set. These predictions were then evaluated against more recent annotations from April 2007.

RESULTS

Prediction of molecular function for held-out data

We assessed the predictive performance of the model using GO annotation data from April 2007. Prediction quality was assessed for fifty aggregate terms, which were chosen from the first three levels of the MF hierarchy, and had at least 30 proteins annotated. There were 7 aggregate terms on the first level: *catalytic activity, binding, transporter activity, structural molecule activity, transcription regulator activity, enzyme regulator activity*, and *translation regulator activity*. There were 20 aggregate terms on the second level and 23 on the third level. For each aggregate term, we compared the predictive performance against direct term-wise binary prediction, i.e., a two-class Naive Bayes model that separates proteins annotated to that term (positive class) from all others (negative class).

Table 1
Prediction from held-out data (results averaged over the three levels nearest to the root)

Level	# Terms	Sensitivity		Specificity		Precision	
		leaf	binary	leaf	binary	leaf	binary
1	7	0.653	0.642	0.930	0.892	0.643	0.498
2	20	0.440	0.620	0.987	0.916	0.458	0.195
3	23	0.425	0.633	0.986	0.917	0.413	0.179

Leaf = prediction performed at leaf terms, then aggregated to more generic terms. Binary = direct binary prediction (members of a functional term versus nonmembers).

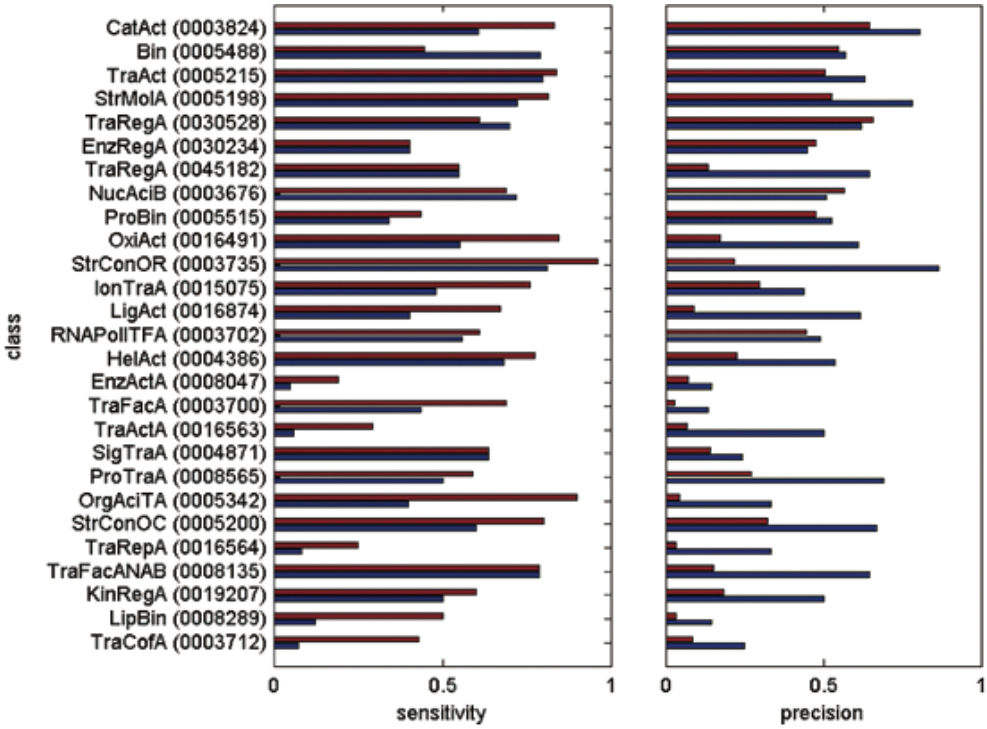

Fig. 3. Comparison between leaf-level and binary prediction. Sensitivity and precision for held-out data (first and second level functional terms). Blue bars indicate aggregated leaf-level prediction; red bars indicate direct binary prediction. (Please see the online version of the article for the colour figure).

Aggregate sensitivity and precision values for the first and second level terms are shown Fig. 3. In terms of specificity and precision, aggregated leaf-level prediction clearly outperformed direct binary prediction. A similar tendency is seen in average results over the first, second and third hierarchy level (Table 1).

In terms of sensitivity, binary predictors appeared slightly better, but this was mostly caused by the binary predictors producing more predictions overall (totaling 8997 positive predictions for the 50 terms and 1661 test proteins, compared to only 3598 from the leaf-level prediction model). Note that the binary predictors could predict a single protein to have arbitrarily many molecular functions, which at least partly explains the greater overall rate of positive predictions.

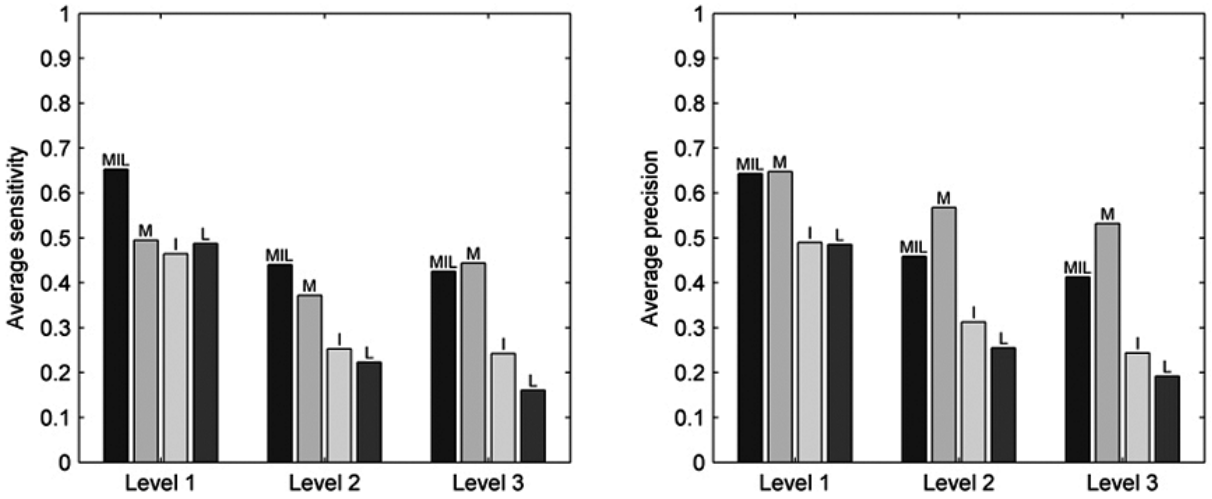

Fig. 4. Effect of feature selection. Average sensitivity and precision for MF prediction at three levels of the GO hierarchy. Bars from left to right: all features (**MIL**), motifs (**M**), interactions (**I**), cellular localization (**L**).

Effect of feature selection

We further studied how the choice of the features affected the predictive performance of the model. An obvious idea, supported, e.g., by Nariai *et al.*, 2007, is that joint consideration of a number of different data sources leads to a systematic improvement in the predictions. This is not true in general, however. Moreover, it is not always clear in advance which features would be most relevant for predicting a particular aspect of GO annotation.

In order to see which features are relevant for MF prediction in our model, we performed experiments with different feature sets: all features (MIL), motifs (M), interactions (I), and cellular localization (L). Apart from such feature selection, the experimental setting was the same as in the previous section. Prediction was performed at leaf level and then aggregated. Average sensitivity and precision values over the three levels of the GO hierarchy are depicted in Fig. 4. In terms of sensitivity, combined feature data generally outperformed any single feature set. However, in terms of precision, it was more efficient to use motifs only.

The results varied along the GO hierarchy. For example, when predicting *oxidoreductase activity* (and other kinds of catalytic activity), *motifs* were clearly the best predictor (sensitivity 0.594; precision 0.891) and in fact better than using all features (sensitivity 0.552; precision 0.609), and other features provided little information (interactions: sensitivity 0.115; localization: sensitivity 0.000). Motifs were generally good predictors for all kinds of catalytic activity, which could perhaps have been anticipated, as catalytic enzymes are among the best-studied proteins and they are highly conserved in evolution. Many motifs from this category of proteins have already been described, and they form a large proportion of the data in PROSITE [Hulo *et al.*, 2006] and in InterPro [Apweiler *et al.*, 2001].

For *nucleic acid binding*, localization was the most important feature set (sensitivity 0.700; precision 0.324), compared to other features (motifs: sensitivity 0.438, precision 0.317; interactions: sensitivity 0.381, precision 0.302). The best results were achieved by combining all features (sensitivity 0.719; precision 0.507). These proteins are often localized in the nucleus for their activity, explaining the relative importance of localization.

Table 2
Prediction from delayed data. Predictions were based on data from September 2004 and evaluated against data from April 2007

Level	GO id	GO term	**#new**	**#pred**	**#correct**	**Sensitivity**	**Precision**
1	3824	catalytic	137	137	68	0.496	0.496
1	5488	binding	181	136	84	0.464	0.618
1	5215	transporter	31	56	26	0.839	0.464
1	30528	transcription regulator	16	26	4	0.250	0.154
1	30234	enzyme regulator	16	15	6	0.375	0.400
2	16787	hydrolase	61	39	24	0.393	0.615
2	16740	transferase	46	55	15	0.326	0.273
2	3676	nucleic acid binding	63	69	41	0.651	0.594
2	5515	protein binding	94	22	14	0.149	0.636
2	16491	oxidoreductase	16	26	6	0.375	0.231
3	3723	RNA binding	50	52	32	0.640	0.615
3	16788	hydrolase (ester bonds)	13	8	3	0.231	0.375
3	3677	DNA binding	13	10	0	0.000	0.000

Columns: #**new** = number of new annotations to each class; #**pred** = number of proteins predicted to each class; #**correct** = number of proteins *correctly* predicted to each class; **Sensitivity** = (#correct/#new); **Precision** = (#correct/#pred).

Generally speaking, our findings support the view that combining a number of different sources of information improves the level of predictions throughout the GO hierarchy. However, the Naive Bayes model employed here is not always able to combine such additional information in an optimal way, and occasionally the predictive performance may decrease when more features are included.

Prediction of molecular function from delayed data

In the *delayed data* setting we used 2904 protein annotations from September 2004 as training data, and using the full feature set, generated functional predictions for 369 proteins that lacked MF annotation at that time, but had been annotated in the newer (April 2007) data. Our predictions were then compared against the new annotations.

The aim of this experiment was to simulate the task of actually annotating previously unannotated proteins ("predicting the future"), as opposed to the held-out setting, where a randomly held-out part of the present annotations are being predicted ("predicting the present"). It is not obvious that these two tasks would be equally difficult; in fact, one could well presume that more difficult annotation tasks generally take longer to be resolved, in which case genuine prediction of future annotations would be harder than guessing values of held-out data.

Prediction results, for those terms on the first three levels that had at least ten new proteins annotated, are shown in Table 2. Compared to the held-out setting, sensitivity and precision were considerably lower: "predicting the future" was thus indeed more difficult than "predicting the present". This implies that one should be cautious concerning performance estimates for molecular function prediction, when the estimates are based on a held-out (cross-validation) experiment. Concretely, the result implies that the currently unannotated proteins are qualitatively different from the currently annotated ones, and consequently, a well-designed predictive model for molecular functions should not assume that the currently known annotations are a random sample from the conceivable set of all "true" annotations. If the lack of annotation is viewed as an instance of missing data, the data are not "missing at random".

A further complication is that annotations are not only added over time, but also occasionally removed. In this work we chose to ignore this complication and only address the task of predicting the addition of annotations.

DISCUSSION

Despite systematic efforts extending over almost a decade now, the progress in associating genes and gene products to GO terms has been relatively slow. For instance, when comparing SGD GO data from year 2004 to those from April 2007, only ~ 350 molecular function annotations were added during these three years. This leaves more than 2500 proteins yet to be characterized.

We applied a simple Naive Bayes model for an automatic assignment of molecular functions to proteins. Employment of different sources of data, both separately and jointly, showed a reasonable degree of consistency in the predictions that were made along different levels of the GO hierarchy, but their accuracy varied considerably. Motifs were seen to be the most important single feature for the prediction task, as their removal led to a considerable deterioration in predictive performance (Fig. 4). The contributions of localization and protein-protein interactions were smaller. Obviously, the conditional independence assumption for the features upon which the Naive Bayes classifier is built, is not realistic for many of the features considered here. A possibility to account for *a priori* unknown dependencies among the features in a functional class would be to use graphical modeling tools to provide computationally feasible decompositions of the joint feature distributions. Such tools have been quite widely exploited, e.g., in the form of the Bayesian network classifier [Friedman *et al.*, 1997] and tree augmented network classifiers [Cerquides and de Mántaras, 2005]. In the present context, learning the structure and the parameter values of such classifiers still poses a considerable computational challenge due to the high dimensionality of the models that are involved. Potential approaches to solving such issues could be developed by using parallel computing combined with some fast heuristics suggested in the machine learning literature.

APPENDIX: PROOF OF Eq. (2)

We include here a proof of Eq. (2) for the reader's convenience. According to Eq. (1), the likelihood function is

$$\begin{aligned} P(c, x|\psi, \theta) &= \prod_{i=1}^{n} P(c_i|\psi) \prod_{j=1}^{m} P(x_{ij}|c_i, \theta_j) \\ &= \prod_{k=1}^{K} \psi_k^{n_k} \prod_{j=1}^{m} \theta_{jk}^{n_{jkl}} (1-\theta_{jk})^{n_{jko}}, \end{aligned}$$

where n_k is the number of proteins in class k, and n_{jkl} counts how many times the jth feature has the value l in class k. Since Ψ and Θ_{jk} have Dirichlet and beta distributions, respectively, we can write the joint density as

$$P(c, x, \psi, \theta) = Z \prod_{k=1}^{K} \psi^{\alpha_k + n_k - 1} \prod_{j=1}^{m} \theta_{jk}^{\beta_1 + n_{jkl} - 1} (1-\theta_{jk})^{\beta_o + n_{jko} - 1},$$

where Z is a normalizing constant. Integrating with respect to ψ_k and θ_{jk} (for $j = 1, \ldots, m$ and $k = 1, \ldots, K$) we obtain the marginal likelihood of the data

$$P(c, x) = ZB(\alpha_1 + n_1, \ldots, \alpha_K + n_K) \prod_{j=1}^{m} \prod_{k=1}^{K} B(\beta_1 + n_{jkl}, \beta_o + n_{jko}),$$

where

$$B(\alpha_1, \ldots, \alpha_K) = \left(\prod_{k=1}^{K} \Gamma(\alpha_k) \right) \Big/ \Gamma \left(\sum_{k=1}^{K} \alpha_k \right)$$

is the multinomial beta function. Now conditioning on the observed feature indicators x_{ij} (for all i, j) and observed class indicators c_1, ..., c_{n-1}, we can compute the posterior class probabilities of the nth protein, for each class k, as

$$P(C_n = k | x, c_1, \ldots, c_{n-1}) \propto (\alpha_k + n_k) \prod_{j=1}^{m} (\beta_1 + n'_{jkl})$$

where the counts n'_k and n'_{jkl} are taken over the observed data ($i = 1, \ldots, n-1$).

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 Res. **25**, 3389-3402.
- Apweiler, R., Attwood, T. K., Bairoch, A., Bateman, A., Birney, E., Biswas, M., Bucher, P., Cerutti, L., Corpet, F., Croning, M. D., Durbin, R., Falquet, L., Fleischmann, W., Gouzy, J., Hermjakob, H., Hulo, N., Jonassen, I., Kahn, D., Kanapin, A., Karavidopoulou, Y., Lopez, R., Marx, B., Mulder, N. J., Oinn, T. M., Pagni, M., Servant, F., Sigrist, C. J. and Zdobnov, E. M. (2001). The InterPro database, an integrated documentation resource for protein families, domains and functional sites. Nucleic Acids Res. **29**, 37-40.
- 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., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M. and Sherlock, G. (2000). Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat. Genet. **25**, 25-29.
- Barutcuoglu, Z., Schapire, R. E. and Troyanskaya, O. G. (2006). Hierarchical multi-label prediction of gene function. Bioinformatics **22**, 830-836.
- Cerquides, J. and de Mántaras, R. L. (2005). TAN classifiers based on decomposable distributions. Mach. Learning **59**, 323-354.
- Deng, M., Zhang, K., Mehta, S., Chen, T. and Sun, F. (2003). Prediction of protein function using protein-protein interaction data. J. Comput. Biol. **10**, 947-960.
- Eisenberg, D., Marcotte, E. M., Xenarios, I. and Yeates, T. O. (2000). Protein function in the post-genomic era. Nature **405**, 823-826.
- Friedman, N., Geiger, D. and Goldszmidt, M. (1997). Bayesian Network Classifiers. Mach. Learning **29**, 131-163.
- Gabaldón, T. and Huynen, M. A. (2004). Prediction of protein function and pathways in the genome era. Cell. Mol. Life Sci. **61**, 930-944.
- Hand, D. J. and Yu, K. (2001). Idiots Bayes: not so stupid after all. Int. Stat. Review **69**, 385-398.
- Hulo, N., Bairoch, A., Bulliard, V., Cerutti, L., De Castro, E., Langendijk-Genevaux, P. S., Pagni M. and Sigrist, C. J. A. (2006). The PROSITE database. Nucleic Acids Res. **34**, D227-D230.
- Lanckriet, G. R., Deng, M., Cristianini, N., Jordan, M. I. and Noble, W. S. (2004). Kernel-based data fusion and its application to protein function prediction in yeast. Pac. Symp. Biocomput., 300-311.
- Letovsky, S. and Kasif S. (2003). Predicting protein function from protein/protein interaction data: a probabilistic approach. Bioinformatics **19**, i197-i204.
- Nariai N., Kolaczyk, E. D. and Kasif, S. (2007). Probabilistic protein function prediction from heterogeneous genome-wide data. PLoS ONE **2**, e337.
- Pearson, W. and Lipman, D. (1988). Improved tools for biological sequence comparison. Proc. Natl. Acad. Sci. USA **85**, 2444-2448.
- Schwikowski, B., Uetz, P. and Fields, S. (2000). A network of protein-protein interactions in yeast. Nat. Biotech. **18**, 1257-1261.
- Stark, C., Breitkreutz, B. J., Reguly, T., Boucher, L., Breitkreutz, A. and Tyers, M. (2006). BioGRID: a general repository for interaction datasets. Nucleic Acids Res. **34**, D535-D539.

- Troyanskaya, O. G., Dolinski, K., Owen, A. B., Altman, R. B. and Botstein, D. (2003). A Bayesian framework for combining heterogeneous data sources for gene function prediction (in Saccharomyces cerevisiae). Proc. Natl. Acad. Sci. USA **100**, 8348-8353.
- Vazquez, A., Flammini, A., Maritan, A. and Vespignani, A. (2003). Global protein function prediction from protein-protein interaction networks. Nat. Biotech. **21**, 697-700.