New paradigm in protein function prediction for large scale omics analysis

Troy Hawkins,[a] Meghana Chitale[b] and Daisuke Kihara*[abc]

Biological interpretation of large scale omics data, such as protein–protein interaction data and microarray gene expression data, requires that the function of many genes in a data set is annotated or predicted. Here the predicted function for a gene does not necessarily have to be a detailed biochemical function; a broad class of function, or low-resolution function, may be sufficient to understand why a set of genes shows the observed expression pattern or interaction pattern. In this *Highlight*, we focus on two recent approaches for function prediction which aim to provide large coverage in function prediction, namely omics data driven approaches and a thorough data mining approach on homology search results.

Introduction

Characterization of the function of genes is a central question in biological study, especially in molecular biology, genetics, and biochemistry. Computationally, the function of a gene can be inferred from similarity to genes of known function typically in terms of global/local sequence and the tertiary structure. The most widely used method to predict the function of a target gene is to use a homology search tool, such as BLAST,[1] PSI-BLAST[2] or FASTA,[3,4] where homologous sequences are searched and their annotated function is transferred to the target gene.

A typical usage of BLAST is to employ a constant threshold value for the statistical significance score, the *E*-value, and only consider search hits with an *E*-value of the threshold significance or more. This strategy is effective in reducing false positive function assignments to genes, which is very important for function annotation for entries in a public database, *e.g.* UniProt,[5] which will be referred to by many people for a variety of different purposes. Therefore, this constant threshold approach is usually used in genome sequence annotation in genome projects. A drawback of this approach is the small coverage in function annotation; typically only less than half of the genes in a genome can be annotated, and the rest remain unknown.[6]

However, large-scale omics data which have appeared around the beginning of this century, such as protein–protein interaction data and microarray gene expression data, have raised a different need in gene function prediction, that is, large coverage in function annotation. For biological interpretation of large-scale omics data, detailed biochemical function assignment to genes is not always necessary. Low-resolution functions, such as pathway information or a broad class of biological function, are still very helpful if they are assigned to a large number of genes in the data set in order to speculate, for example, why a given set of genes is up- or down-regulated in the same fashion as a group in a microarray data.

Numerous methods have been proposed which aim to predict function beyond a conventional BLAST approach in terms of function assignment coverage. Those methods include protein tertiary structure-based methods,[7,8] comparative genomics-based methods,[9–14] and pathway-based methods.[15,16] However because of the page limitation, here we

[a] *Department of Biological Sciences, College of Science, Purdue University, West Lafayette, IN, 47907, USA. E-mail: dkihara@purdue.edu; Fax: +1-765-496-1189; Tel: +1-765-496-2284*

[b] *Department of Computer Science, College of Science, Purdue University, West Lafayette, IN, 47907, USA*

[c] *Markey Center for Structural Biology, College of Science, Purdue University, West Lafayette, IN, 47907, USA*

Daisuke Kihara is an Assistant Professor in the Departments of Biological Sciences and Computer Science at Purdue University. He received his PhD degree from Kyoto University in 1999. His research projects include protein function and structure prediction and protein surface shape searching for function prediction and docking.
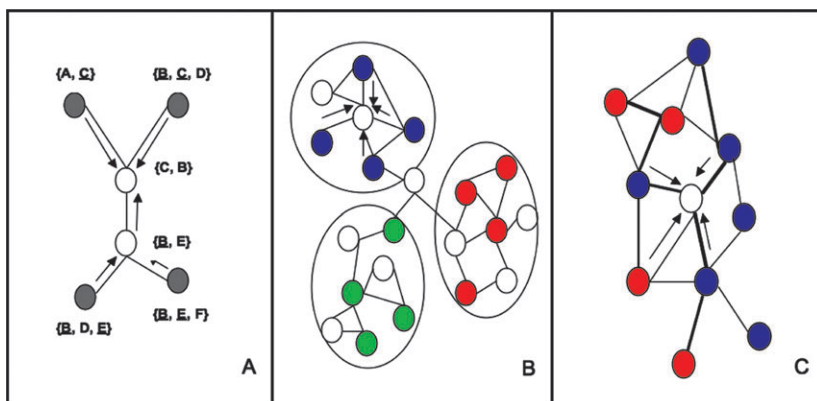
Fig. 1 Function prediction using protein–protein interaction network. A: Local network-based majority vote method:[18] each protein in the functional linkage network is represented by a node and an edge between two proteins indicates functional similarity based on evidence from some data source. Grey colored proteins have known functions indicated in brackets against them and white nodes represent un-annotated proteins. The arrows indicate the direction of transfer of the highest frequency annotation of neighbors to the un-annotated protein. The annotations transferred have been underlined. B: Clustering method:[22] proteins in the functional linkage network with each cellular function are indicated in different colors. The identified clusters of proteins in the network show dominance of a single cellular function that is achieved by the interactions between the proteins in that cluster. Arrows indicate the influence of cluster membership to determine the annotation for an un-annotated protein. C: Global network topology based method using functional flow:[27] each edge width connecting proteins represents the reliability associated with the evidence source. Nodes (proteins) colored in blue and red have annotation labels and act as a source of infinite functional flow for the annotation label represented by their color. A white node representing an un-annotated protein receives functional flow for each annotation over the course of the simulation by using each edge in the network as a conduit. In the example the white node which is close to three blue nodes connected with thick edges will receive maximum flow for the blue annotation selecting it over the red annotation.

focus on two recent new approaches: omics data driven approaches and thorough mining of homology search approaches, and leave the rest to recent review articles.[6,17]

Omics data driven approaches

First, as typical omics data driven approaches, we overview ideas of protein–protein interaction (PPI) data-based function prediction methods. PPI data can be represented as a network (graph) where proteins are represented as nodes and interacting proteins are connected by edges. PPI data are useful for function prediction because it has been shown that proteins of known function and cellular location tend to cluster together in the PPI network.[18] The methods introduced below range from simple approaches which consider immediate neighboring nodes to more advanced approaches which take global network topology into account. Actually the ideas of function prediction applied to PPI networks can also be applied to a more

generalized functional linkage network, where an edge between two nodes (proteins) is established if there is experimental (e.g. microarray gene expression data) or computational evidence that suggests that the two linked proteins are likely to share the same function.

A basic idea is to consider a local network and perform a majority vote that involves assignment of the most common functions present among the neighboring proteins in the network (Fig. 1A).[18] Methods assigning functions to proteins based on frequently occurring annotations in the n-neighborhood of the protein in the PPI network are known as neighborhood counting methods. Hishigaki et al.[19] have developed an approach based on chi-square statistics that uses the frequency of the function of interest among the interacting neighbors of an unknown protein. Application of chi-square statistics has issues because of the small number of interacting neighbors present for a number of un-annotated proteins in the network.

Gao et al.[20] assign the most specific possible functional class to the unknown protein using annotation information of k neighbors that are within a specified distance in the PPI network. The annotation classes of k neighbors and their ancestors in the Gene Ontology (GO) hierarchy[21] are used as candidate annotation classes for the unknown protein. For each candidate class they sum the taxonomy similarity score with each of the annotations from the k neighbors. The larger the score for a particular candidate class the higher is its possibility of being the annotation of the unknown protein.

Brun et al. partition interaction networks into clusters (Fig. 1B) of proteins that are part of the same protein complex or take part in the same cellular function.[22] Their method is based on the principle that two proteins are more likely to be functionally related if they share a number of common interacting proteins. The results showed that clustering done based on functional distance, which is based on the number of shared interacting proteins, separates proteins involved in different biological process. The majority of proteins in a cluster do not share sequence similarity or molecular function. Unlike sequence-based methods that point towards proteins having the same molecular function, the clustering methods establish groups of proteins belonging to the same functional class representing some biological process.

Many approaches concentrate on using just the annotation information of level 1 neighbors of un-annotated proteins in the network for predicting their function; this is known as direct functional association. Chua et al.[23] make use of the fact that in many cases proteins share functional similarity with level 2 neighbors, which is indirect functional association. They show that level 2 and 3 neighbors have an above average likelihood of sharing functional similarity. A weighted averaging method based on functional similarity weight between the proteins is defined to predict the function using level 1 and level 2 neighbors. It has been shown to outperform some of the existing methods that use interconnection network information in the three main categories of GO:

molecular function, biological process, and cellular component.

Some current approaches for predicting functional assignment are based on deriving the marginal probability of a protein taking a particular function given the functional label of other proteins in the functional linkage network. Letovsky and Kasif[24] use a Markov Random Field (MRF) framework subject to a conditional independence assumption that the probability distribution for any node is conditionally independent of all other nodes given its neighbors. The algorithm starts with initializing the label (function) probabilities for unlabeled nodes to the frequency of the label and propagates frequencies iteratively. In the second iteration unlabeled nodes adjust the probabilities based on their neighbors using Bayes' rule. The algorithm stops after the second iteration to avoid self-reinforcement, and classifies the unlabeled nodes whose probability exceeds a threshold by assigning them the current label. Deng et al.[25] further explore this idea by considering the frequency of proteins having the function of interest with less weight placed on far away neighbors in the network than on the close neighbors. Their approach considers each function separately and globally takes into account the annotations of all proteins in the interaction network. Conditioned on the functional annotation of each annotated protein in the network, the posterior probability for each protein to have a particular functional annotation is computed using the MRF framework.

Vazquez et al.[26] use a global approach where instead of taking just the locally optimum function based on annotations of interacting neighbors, they determine a globally optimal functional assignment to all un-annotated proteins in the network so as to minimize the number of protein–protein interactions among different functional categories in the network. The functional flow algorithm described by Nabieva et al.[27] applies the concept of functional flow (Fig. 1C). Each protein of known functional annotation is treated as a source of functional flow, which is then propagated to annotated nodes using the edges in the interaction graph as a conduit. The effect of each annotated protein on any other protein decreases with an increase in distance between them. The scores correspond to the amount of flow for the function the protein has received over the course of the simulation. In contrast to the majority vote algorithm,[18] functional flow considers functional annotations from proteins that are not immediate neighbors, and thus can annotate proteins that have no neighbors with known annotations. To take into account the reliability of different data sources the functional linkages between proteins can be weighed independently for different data sources.

Combining heterogeneous data sources with functional linkage networks

It has become increasingly popular to explore the idea of integrative functional genomics, which combines information from multiple sources to assist the process of functional annotation of unknown proteins. Combining heterogeneous data is effective for dealing with errors and the uncertainty of each data point by weighing the evidence based on the reliability of the sources used for prediction. Most commonly, PPI data and gene expression data are combined.[28,29] Transcription pattern similarity measured by microarray techniques may indicate the presence of a functional relationship between two proteins within the context of some biological process. These two data can be unified in the form of a functional linkage network. In a gene expression data set, if the Pearson correlation coefficient of a co-expressed gene pair is above a threshold, then that gene pair is linked in the functional linkage network, which is integrated as part of the network data along with the PPI network.[28,30]

Probabilistic models, especially Bayesian frameworks, are frequently used to compute the posterior probability that a protein has a particular function given the heterogeneous network data from different sources. Nariai et al.[28] combine knowledge obtained from functional linkage graphs constructed from PPI data and microarray gene expression data as well as protein motif information, mutant phenotype data, and protein localization data by Bayesian networks. They use "biological process" terms from the Gene Ontology as the basis for functional annotation. The method shows an 18% increase in correctly recovered annotations when using heterogeneous data as compared to using PPI data alone. The Eisenberg group has developed the ProKnow server, which combines PPI data and protein global/local sequence structure-based function predictions using Bayesian networks.[31]

Troyanskaya et al.[32] have developed a system based on Bayesian networks that incorporates knowledge from different data sources taking into account their relative accuracies to predict if two proteins are functionally related. Each predicted functional relationship between proteins indicates the posterior belief that the two proteins can be involved in the same biological function. The method takes input from various data sources indicating the strength of each method's belief in the existence of a relationship between two genes. The Bayesian network has been constructed based on probabilities provided by experts in the field of yeast molecular biology. They have shown that the accuracy of grouping has been improved using the evidence from heterogeneous data sources as compared to using the microarray analysis alone.

Alternative choices of computational techniques to integrate heterogeneous data include Boltzmann machines, neural networks and Support Vector Machines (SVMs). Chen and Xu[30] use PPI data, protein complex data and microarray gene expression data together to predict protein function. In the global prediction of protein function they do not rely only on the functional annotations of the interacting neighbors, but also base the prediction on the global structure of the interaction network using a Boltzmann machine to characterize the global stochastic behaviors of the network. The Global Mapping of Unknown Proteins (GMUP) method constructs an abstract layer of evidence by combining heterogeneous sources like PPI data, gene expression data and protein complex data by extracting common terminology from different datasets.[33] Each piece of evidence is converted into a pair of proteins with corresponding associated GO terms and a source of information in the form of an input vector and it is used for training the

neural network for prediction of un-known function. Here information from heterogeneous data sources is unified in the form of linkage between pairs of proteins.

Methods using functional linkage net-works provide computational determina-tion of the biological process in which the unknown proteins are likely to be involved whereas the sequence based methods provide important clues about the molecular function of the protein. Thus these two complimentary ap-proaches can be combined to computa-tionally predict different Gene Ontology categories with increased accuracy.

Sequence-based approaches

Although there is a significant body of work utilizing PPI and expression data to predict function, it remains true that sequence data are far more abundant and easy to retrieve. Genome sequencing projects have been completed for over 600 organisms with at least 3000 more underway,[34,35] and large metagenomics projects[36–45] are producing new se-quences at an extraordinary pace. It should be also noted that most of the gene function information is available through sequence databases, e.g. Uni-Prot and PROSITE.[46]

Consensus functional similarity from database searches

As stated at the beginning of this review, the default method characterizing a new gene sequence, using BLAST with a con-stant E-value threshold value, has small coverage in function annotation in a genome. Recently, there have been sig-nificant efforts dedicated to extending the capabilities of the still powerful da-tabase searching tools BLAST and PSI-BLAST. The common feature between all of these approaches is the combina-tion of functional information from mul-tiple sequences retrieved by BLAST. The consensus approach of identifying fre-quently occurring functional characteris-tics among similar sequences to a query mimics the natural human approach to interpretation of BLAST results. Auto-mation of this process allows it to be used both on a large scale, i.e. for ana-
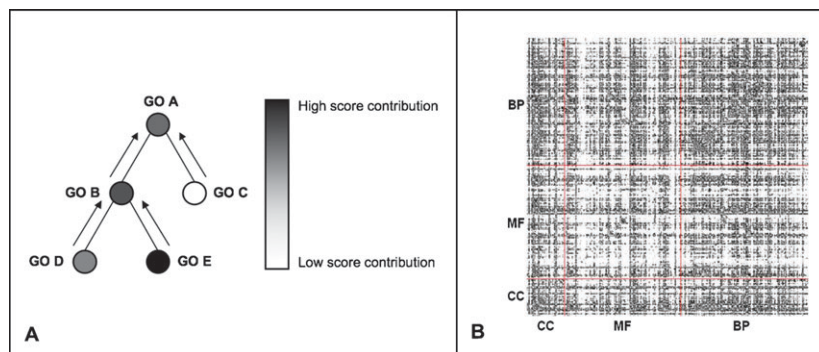
Fig. 2 Use of Gene Ontology hierarchical structure in function prediction. A: Propagation of scores through GO. Several methods apply scores for GO terms to ancestors according to the structure of the GO. Here, scores for GO terms D and E combine to give their common parent term B a medium-high score, which combines with the low score for GO term C to give the root term A a medium-low score. B: Function Association Matrix: visual representation of FAM used in the PFP method. This represents association between 475 GO terms within UniProt sequences, or p(GO term Y|GO term X). Both axes are ordered by Gene Ontology ID number and category (CC = cellular component; MF = molecular function; BP = biological process). Darker spots indicate a higher degree of association between two terms.

lysis of sequences on a genome scale or larger, and also by individuals who do not otherwise have expert intuitive knowledge sufficient to interpret data-base search results effectively.

The GOblet method[47,48] identifies GO functional terms associated to sequence hits in a BLAST search and maps their occurrences onto the GO tree. This map-ping effectually identifies enriched groups of common functions among BLAST hits. OntoBlast[49] takes this a step further by scoring each GO term using E-values of each sequence hit. The E-values for all sequence hits associated with a single term are multiplied together to produce a weighted list of predictions. Enzyme Commission (E.C.) numbers for enzyme function[78] and GO have the advantage of being structured hierarchi-cally, so consensus functions among BLAST results can be described in broader terms. For example, if a query sequence hits both tyrosine kinases and serine kinases, the consensus function can be predicted as a general protein kinase. GOFigure[50] utilizes the hierarch-ical nature of the GO to apply scores for any retrieved functional terms to their parents, providing an output graph with probability scores for each term leading back to the root of the GO tree (Fig. 2A). Gotcha[51] applies an E-value based weighting scheme to the GO structure in a similar way. The score for each GO term found in BLAST hits is propagated onto all ancestor terms in the tree and

then normalized to the total score of the root term to provide a measure of con-fidence.

An extension of the direction of data mining on database search results can be naturally achieved by applying machine learning techniques. GOPET[52] applies a SVM to a list of BLAST results with inputs including alignment length and E-value of sequence hits, GO term fre-quency among hits, GO term relation-ships between homologs, annotation quality of homologs, and the level of annotation within the GO hierarchy to predict GO terms for a query sequence. ProtFun[53] uses a series of sequence-re-lated features as input into an SVM. Rather than using BLAST as a base, this method uses the amino acid composition of the query sequence, predicted post-translational modifications, presence of signal peptides, sequence length, isoelec-tric point, and disordered region predic-tion. Finally the query sequence is scored with probabilities as enzyme/non-en-zyme and also into both GO and E.C. classes.

PFP: considering functional association

We have recently developed the PFP algorithm[54] which uses PSI-BLAST in a similar consensus approach to those de-scribed above. GO terms associated to sequence hits are combined using an

E-value-based scoring scheme that takes into account even high *E*-value sequence hits. Additionally, PFP also uses a novel data mining tool to predict additional GO terms which are highly associated to those terms associated to sequence hits from PSI-BLAST. This tool, the Function Association Matrix (FAM), describes the probability that two GO terms are associated to the same sequence based on the frequency at which they co-occur in UniProt sequences (Fig. 2B). This allows the FAM to associate function annotations from different GO categories, *e.g.* the biological process "positive regulation of transcription, DNA-dependent" is strongly associated with the molecular function "DNA binding activity" and the cellular component "nucleus". Associations can describe parallel functions that may be defined in multiple categories or complementary functions that are defined in one or more categories.

Significant associations, those with high confidence scores, are used to contribute additional functional term predictions to those that can be directly mined from PSI-BLAST results (eqn (1) and (2)):

$$s(f_a) = \sum_{i=1}^{N} \sum_{j=1}^{N\text{func}(i)}$$
$$((-\log(E\_value(i)) + b)P(f_a|f_j)),$$
$$(1)$$

$$P(f_a|f_j) = \frac{c(f_a, f_j) + \varepsilon}{c(f_j) + \mu \cdot \varepsilon},\qquad(2)$$

where where $s(f_a)$ is the final score assigned to the GO term, f_a, N is the number of similar sequences retrieved by PSI-BLAST, $N\text{func}(i)$ is the number of GO terms assigned to sequence j, $E\_value(i)$ is the *E*-value given to the sequence i, f_j is a GO term assigned to the sequence i, $P(f_a|f_j)$ is the conditional probability that f_a is associated with f_j, $c(f_a, f_j)$ is the number of times f_a and f_j are assigned simultaneously to each sequence in UniProt, $c(f_j)$ is the total number of times f_j appears in UniProt, μ is the size of one dimension of the FAM (*i.e.* the total number of unique GO terms), and ε is the pseudo-count.

PFP also uses the structured nature of the GO to predict broad function in the case where no specific functional terms

can be found from the PSI-BLAST results or the additional data mining component. Scores from each GO term in the results are propagated to ancestor or parent terms in the ontology according to the number of genes associated to the predicted term relative to the ancestor term (eqn (3)):

$$s(f_p) = \sum_{i=1}^{N_c} \left(s(f_{ci}) \left(\frac{c(f_{ci})}{c(f_p)} \right) \right).\qquad(3)$$

The advantage of considering consensus GO terms among PSI-BLAST hits (eqn (1)), FAM (eqn (2)) and GO hierarchical structure (eqn (3)) is that a low-resolution function will often show up with a high score when there is no strong hit in a PSI-BLAST result. As a result, annotation coverage of a genome increases dramatically: for example, PFP assigns function to approximately 49.1% more genes than the original annotation in *Plasmodium falciparum* (malaria) genome and 34.3% more in *C. elegans* genome (manuscript in preparation). PFP has already been used in some genome sequencing projects.[55,56] The PFP server is available at http://dragon.bio.purdue.edu/pfp/.

## Function prediction competitions

The biennial Critical Assessment of Techniques for Protein Structure Prediction has included a function prediction category in CASP6 and CASP7 (the previous two rounds). For each target protein, sequence data were provided to groups predicting both E.C. number and GO terms and binding site residues. In CASP7, 22 groups (and two consensus methods) submitted function predictions for 66 target protein sequences. Remarkably, PFP had the highest overall average score and the most wins in a head-to-head comparison between methods predicting terms for the same targets.[57] PFP was also ranked at the top in the Automated Function Prediction meeting (AFP-SIG 05) held at the 13th Annual International Conference on Intelligent Systems for Molecular Biology (ISMB) in summer 2005. These competitions are viewed as an important fixture to maintain a vibrant community of predictors who are consistently working to improve

prediction methods and results. Interest is rapidly growing in the field of automated function prediction, so we can expect that in the future many more groups will be participating in such activities, with a much greater diversity in the information used to make predictions.

Sequence *vs.* interaction *vs.* correlated expression for yeast sequences

To illustrate the effectiveness of utilizing different data sources for the prediction of protein function, we compared GO function prediction by PFP (a sequence-based method) to omics data driven approaches, which use PPI, microarray, or synthetic lethality data for sequences in the yeast genome (Fig. 3). The results and the evaluation method used for the omics data driven approaches are described by Myers *et al.*[58] Note that because of the nature of the prediction methods used, the evaluation methods used for predictions by PFP (lower half of Fig. 3) and omics data driven approaches (upper half of Fig. 3) are different: protein interactions are considered to be correct for a particular GO term when associating partners as identified by two hybrid analysis, affinity precipitation, synthetic lethality, or microarray have the function in common. PFP predictions are considered correct for a particular GO term when predictions are made for that function above a set confidence threshold. In both cases we measured precision (True Positives/[True Positives + False Positives]) and recall (True Positives/[True Positives + False Negatives]) at several thresholds to determine an overall relevance of the method for that term and its children in the ontology. (These data for the experimental datasets were obtained using the GRIFn server at Princeton, http://avis.princeton.edu/GRIFn/.) The intensity of each square in Fig. 3 corresponds to the overall accuracy of each method for each term as measured by the area under the precision-recall curve (AUPRC) (the higher the AUPRC, the brighter the square).

Two conclusions can be drawn from the comparison here. First, it is
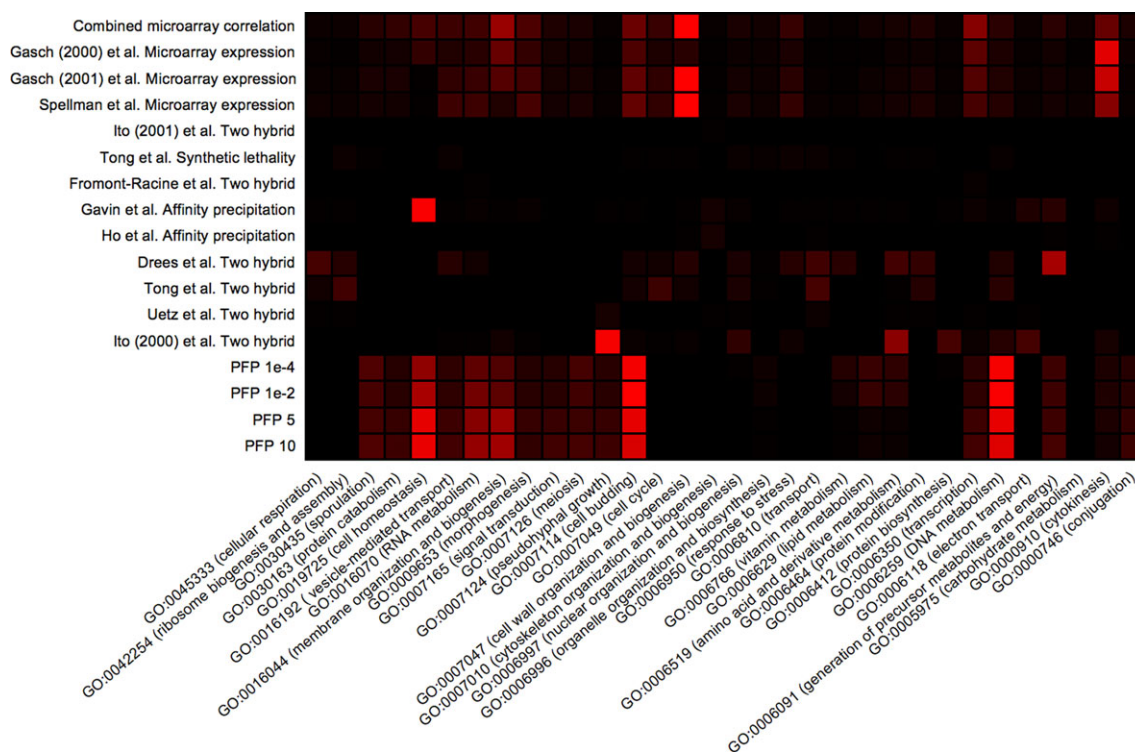
Fig. 3 Comparison of predictions for yeast proteins: rows 1–4 are data from correlated expression as identified by microarray. Rows 5–13 are data from protein–protein interaction as identified by two hybrid and affinity precipitation methods. Rows 14–17 are predictions from PFP using different *E*-value thresholds for inclusion of BLAST results (*i.e.*, "10" means only sequences with *E*-values above 10.0 (*e.g.*, 15, 100) were used to make predictions). Columns are Gene Ontology terms. The intensity of red at each square represents the area under a precision-recall curve (AUPRC) or overall accuracy of the dataset for a GO term and its children.

somewhat obvious that sequence data have more universal relevance to a broader number of functional categories than correlated expression, and both are more appropriate for more functional categories than physical protein–protein interaction or synthetic lethality. PFP shows strong performance on six GO terms and moderate performance for 14 additional terms (of 32), while the inter-action data only give a strong indication of functional similarity for two GO terms and a moderate indication for ten additional terms, although there is little consistency between the physical interaction datasets. Second, and more important, is the apparent complementary nature of the different data types in terms of their relevance for different subsets of function. Most of the GO categories in which our sequence-based approach gave moderate to strong prediction performance are covered poorly by physical protein interactions and microarrays, and all of the categories in which protein–protein interaction or correlated expression gave a strong indication of function are covered poorly by our se-

quence-based approach. These results are somewhat intuitive. The functional categories for which protein–protein interaction strongly indicates functional similarity are known to involve significant protein interactions, *e.g.* protein complexes like the ribosome in the case of "ribosome biogenesis" (GO:0007046) and "protein synthesis" (GO:0006412) and extensive and prolonged functional dimerization in the case of "cell cycle" (GO:0007049). A general conclusion could be that although sequence similarity approaches work from a larger pool of functional knowledge and can thus provide high coverage and high accuracy, other high throughput omics data can provide additional clues for protein function that sequence fails to uncover.

Summary

The recent availability of both tremendous amounts of genomic sequence data and increasing numbers of high through-put experimental datasets has made possible the creation of several techniques

that utilize these data for the prediction of protein function. As we have discussed here, these approaches go well beyond the traditional use of BLAST or similar tools for blind function annotation of new proteins. The large volume of experimental omics data allows us to investigate relationships between genes, a new perspective from which we can draw clues toward function. And although sequences have been investigated for almost 20 years by BLAST and FASTA, new consensus and data mining approaches show that there is still rich functional information that can be drawn beyond what traditional tools can find. For users' convenience, software resources of the methods mentioned in the text are listed in Table 1. In the future we will see combinations of these methods that will take advantage of the complementarity of heterogeneous data sources. These approaches to function prediction are becoming increasingly important in the omics age of biology and systems biology, where even broad functional clues can assist in the interpretation of new datasets.

Table 1 Software resources for omics- and sequence-based function prediction

Software	Description	URL
CYGD	The MIPS comprehensive Genome Database of Yeast functional network	http://mips.gsf.de/genre/proj/yeast/
BioGRID	Freely accessible database of protein and genetic interactions	http://www.thebiogrid.org
DIP	Database of interacting proteins	http://dip.doe-mbi.ucla.edu/
BOND	Biomolecular interaction network database	http://bond.unleashedinformatics.com/
PROTEOME[59]	Protein interaction database	http://www.proteome.com
MCODE[60]	Molecular complex detection by clustering in PPI network	http://www.baderlab.org/Software/MCODE
PRODISTIN[61]	Classification tool for genes/proteins using interaction networks	http://gin.univ-mrs.fr/webdistin
PathBLAST[62]	Tool for aligning two PPI networks	http://www.pathblast.org/
Cfinder[63]	Tool for finding densely interconnected nodes in PPI networks	http://www.cfinder.org/D.html
GAIN[64]	Tool for gene annotation using integrated networks	http://genomics10.bu.edu/gain/index.html
GOMiner[65]	Microarray clustering tool	http://discover.nci.nih.gov/gominer/
GOSurfer[66]	Microarray clustering tool	http://bioinformatics.bioen.uiuc.edu/gosurfer/
GenMAPP[67]	Microarray clustering tool	http://www.genmapp.org/
ArrayTrack[68]	Microarray clustering tool	http://www.fda.gov/nctr/science/centers/toxicoinformatics/ArrayTrack/index.htm
CaGEDA[69]	Microarray analysis software	http://bioinformatics.upmc.edu/GE2/GEDA.html
SAM[70]	Microarray analysis software	http://www-stat.stanford.edu/~tibs/SAM/
NUDGE[71]	Microarray analysis software	http://www.bioconductor.org/
GRIFn[58]	Proteomics data evaluation	http://avis.princeton.edu/GRIFn/
BLAST;[1] PSI-BLAST[2]	Database homology search	http://www.ncbi.nlm.nih.gov/blast/
FASTA[4]	Database homology search	http://www.ebi.ac.uk/fasta33/
PFP[54]	BLAST-based GO term prediction + data mining	http://dragon.bio.purdue.edu/pfp/
GOtcha[51]	BLAST-based GO term prediction	http://www.compbio.dundee.ac.uk/gotcha/gotcha.php
GOFigure[50]	BLAST-based GO term prediction	http://udgenome.ags.udel.edu/gofigure/
OntoBlast[49]	BLAST-based GO term prediction	http://functionalgenomics.de/ontogate/
Goblet[47,48]	BLAST-based GO term prediction	http://goblet.molgen.mpg.de/
ProKnow[31]	Sequence + structure GO term prediction	http://proknow.mbi.ucla.edu/
ELM[72]	Functional motif scanning	http://elm.eu.org/
InterProScan[73]	Functional motif scanning	http://www.ebi.ac.uk/InterProScan/
ScanProsite[74]	Functional motif scanning	http://www.expasy.ch/prosite/
Pfam[75]	Protein family classification	http://pfam.sanger.ac.uk/
SMART[76]	Sequence fingerprint scanning	http://smart.embl-heidelberg.de/
JAFA[77]	GO term prediction metaserver	http://jafa.burnham.org/
GOPET[52]	BLAST-based GO term prediction by SVM	http://genius.embnet.dkfz-heidelberg.de/menu/biounit/open-husar
ProtFun[53]	Sequence feature based function classification	http://www.cbs.dtu.dk/services/ProtFun/

References

1 S. F. Altschul, W. Gish, W. Miller, E. W. Myers and D. J. Lipman, *J. Mol. Biol.*, 1990, **215**, 403–410.

2 S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller and D. J. Lipman, *Nucleic Acids Res.*, 1997, **25**, 3389–3402.

3 W. R. Pearson, *Methods Enzymol.*, 1990, **183**, 63–98.

4 W. R. Pearson and D. J. Lipman, *Proc. Natl. Acad. Sci. U. S. A.*, 1988, **85**, 2444–2448.

5 C. H. Wu, R. Apweiler, A. Bairoch, D. A. Natale, W. C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, M. J. Martin, R. Mazumder, C. O'Donovan, N. Redaschi and B. Suzek, *Nucleic Acids Res.*, 2006, **34**, D187–191.

6 T. Hawkins and D. Kihara, *J. Bioinf. Comput. Biol.*, 2007, **5**, 1–30.

7 D. Kihara and M. Kanehisa, *Genome Res.*, 2000, **10**, 731–743.

8 B. Li, S. Turuvekere, M. Agrawal, D. La, K. Ramani and D. Kihara, *Proteins*, 2007.

9 I. Yanai, A. Derti and C. DeLisi, *Proc. Natl. Acad. Sci. U. S. A.*, 2001, **98**, 7940–7945.

10 E. M. Marcotte, M. Pellegrini, H. L. Ng, D. W. Rice, T. O. Yeates and D. Eisenberg, *Science*, 1999, **285**, 751–753.

11 M. Pellegrini, E. M. Marcotte, M. J. Thompson, D. Eisenberg and T. O. Yeates, *Proc. Natl. Acad. Sci. U. S. A.*, 1999, **96**, 4285–4288.

12 J. O. Korbel, T. Doerks, L. J. Jensen, C. Perez-Iratxeta, S. Kaczanowski, S. D. Hooper, M. A. Andrade and P. Bork, *PLoS Biol.*, 2005, **3**, e134.

13 J. O. Korbel, L. J. Jensen, C. von Mering and P. Bork, *Nat. Biotechnol.*, 2004, **22**, 911–917.

14 C. von Mering, L. J. Jensen, B. Snel, S. D. Hooper, M. Krupp, M. Foglierini, N. Jouffre, M. A. Huynen and P. Bork, *Nucleic Acids Res.*, 2005, **33**, D433–437.

15 M. L. Green and P. D. Karp, *Nucleic Acids Res.*, 2006, **34**, 3687–3697.

16 M. L. Green and P. D. Karp, *BMC Bioinf.*, 2004, **5**, 76.

17 J. D. Watson, R. A. Laskowski and J. M. Thornton, *Curr. Opin. Struct. Biol.*, 2005, **15**, 275–284.

18 B. Schwikowski, P. Uetz and S. Fields, *Nat. Biotechnol.*, 2000, **18**, 1257–1261.

19 H. Hishigaki, K. Nakai, T. Ono, A. Tanigami and T. Takagi, *Yeast*, 2001, **18**, 523–531.

20 L. Gao, X. Li, Z. Guo, M. Zhu, Y. Li and S. Rao, *Sci. China, Ser. C: Life Sci.*, 2007, **50**, 125–134.

21 M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight,

J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin and G. Sherlock, *Nat. Genet.*, 2000, **25**, 25–29.

22 C. Brun, F. Chevenet, D. Martin, J. Wojcik, A. Guenoche and B. Jacq, *Genome Biol.*, 2003, **5**, R6.

23 H. N. Chua, W. K. Sung and L. Wong, *Bioinformatics*, 2006, **22**, 1623–1630.

24 S. Letovsky and S. Kasif, *Bioinformatics*, 2003, **19**(Suppl. 1), i197–204.

25 M. Deng, K. Zhang, S. Mehta, T. Chen and F. Sun, *J. Comput. Biol.*, 2003, **10**, 947–960.

26 A. Vazquez, A. Flammini, A. Maritan and A. Vespignani, *Nat. Biotechnol.*, 2003, **21**, 697–700.

27 E. Nabieva, K. Jim, A. Agarwal, B. Chazelle and M. Singh, *Bioinformatics*, 2005, **21**(Suppl. 1), i302–310.

28 N. Nariai, E. D. Kolaczyk and S. Kasif, *PLoS One*, 2007, **2**, e337.

29 F. Markowetz and O. G. Troyanskaya, *Mol. Biosyst.*, 2007, **3**, 478–482.

30 Y. Chen and D. Xu, *Pac. Symp. Biocomput. 2005*, 2005, 471–482.

31 D. Pal and D. Eisenberg, *Structure*, 2005, **13**, 121–130.

32 O. G. Troyanskaya, K. Dolinski, A. B. Owen, R. B. Altman and D. Botstein, *Proc. Natl. Acad. Sci. U. S. A.*, 2003, **100**, 8348–8353.

33 J. Xiong, S. Rayner, K. Luo, Y. Li and S. Chen, *BMC Bioinf.*, 2006, **7**, 268.

34 K. Liolios, K. Mavromatis, N. Tavernarakis and N. C. Kyrpides, *Nucleic Acids Res.*, 2008, **36**, D475–479.

35 K. Liolios, N. Tavernarakis, P. Hugenholtz and N. C. Kyrpides, *Nucleic Acids Res.*, 2006, **34**, D332–334.

36 G. W. Tyson, J. Chapman, P. Hugenholtz, E. E. Allen, R. J. Ram, P. M. Richardson, V. V. Solovyev, E. M. Rubin, D. S. Rokhsar and J. F. Banfield, *Nature*, 2004, **428**, 37–43.

37 D. B. Rusch, A. L. Halpern, G. Sutton, K. B. Heidelberg, S. Williamson, S. Yooseph, D. Wu, J. A. Eisen, J. M. Hoffman, K. Remington, K. Beeson, B. Tran, H. Smith, H. Baden-Tillson, C. Stewart, J. Thorpe, J. Freeman, C. Andrews-Pfannkoch, J. E. Venter, K. Li, S. Kravitz, J. F. Heidelberg, T. Utterback, Y. H. Rogers, L. I. Falcon, V. Souza, G. Bonilla-Rosso, L. E. Eguiarte, D. M. Karl, S. Sathyendranath, T. Platt, E. Bermingham, V. Gallardo, G. Tamayo-Castillo, M. R. Ferrari, R. L. Strausberg, K. Nealson, R. Friedman, M. Frazier and J. C. Venter, *PLoS Biol.*, 2007, **5**, e77.

38 J. C. Venter, K. Remington, J. F. Heidelberg, A. L. Halpern, D. Rusch, J. A. Eisen, D. Wu, I. Paulsen, K. E. Nelson, W. Nelson, D. E. Fouts, S. Levy, A. H. Knap, M. W. Lomas, K. Nealson, O. White, J. Peterson, J. Hoffman, R. Parsons, H. Baden-Tillson, C. Pfannkoch, Y. H. Rogers and H. O. Smith, *Science*, 2004, **304**, 66–74.

39 S. Yooseph, G. Sutton, D. B. Rusch, A. L. Halpern, S. J. Williamson, K. Remington, J. A. Eisen, K. B. Heidelberg, G. Manning, W. Li, L. Jaroszewski, P. Cieplak, C. S. Miller, H. Li, S. T. Mashiyama, M. P. Joachimiak, C. van Belle, J. M. Chandonia, D. A. Soergel, Y. Zhai, K. Natarajan, S. Lee, B. J. Raphael, V. Bafna, R. Friedman, S. E. Brenner, A. Godzik, D. Eisenberg, J. E. Dixon, S. S. Taylor, R. L. Strausberg, M. Frazier and J. C. Venter, *PLoS Biol.*, 2007, **5**, e16.

40 E. F. DeLong, C. M. Preston, T. Mincer, V. Rich, S. J. Hallam, N. U. Frigaard, A. Martinez, M. B. Sullivan, R. Edwards, B. R. Brito, S. W. Chisholm and D. M. Karl, *Science*, 2006, **311**, 496–503.

41 S. J. Hallam, N. Putnam, C. M. Preston, J. C. Detter, D. Rokhsar, P. M. Richardson and E. F. DeLong, *Science*, 2004, **305**, 1457–1462.

42 S. G. Tringe, C. von Mering, A. Kobayashi, A. A. Salamov, K. Chen, H. W. Chang, M. Podar, J. M. Short, E. J. Mathur, J. C. Detter, P. Bork, P. Hugenholtz and E. M. Rubin, *Science*, 2005, **308**, 554–557.

43 S. R. Gill, M. Pop, R. T. Deboy, P. B. Eckburg, P. J. Turnbaugh, B. S. Samuel, J. I. Gordon, D. A. Relman, C. M. Fraser-Liggett and K. E. Nelson, *Science*, 2006, **312**, 1355–1359.

44 P. J. Turnbaugh, R. E. Ley, M. A. Mahowald, V. Magrini, E. R. Mardis and J. I. Gordon, *Nature*, 2006, **444**, 1027–1031.

45 H. Garcia Martin, N. Ivanova, V. Kunin, F. Warnecke, K. W. Barry, A. C. McHardy, C. Yeates, S. He, A. A. Salamov, E. Szeto, E. Dalin, N. H. Putnam, H. J. Shapiro, J. L. Pangilinan, I. Rigoutsos, N. C. Kyrpides, L. L. Blackall, K. D. McMahon and P. Hugenholtz, *Nat. Biotechnol.*, 2006, **24**, 1263–1269.

46 C. J. Sigrist, L. Cerutti, N. Hulo, A. Gattiker, L. Falquet, M. Pagni, A. Bairoch and P. Bucher, *Briefings Bioinf.*, 2002, **3**, 265–274.

47 D. Groth, H. Lehrach and S. Hennig, *Nucleic Acids Res.*, 2004, **32**, W313–317.

48 S. Hennig, D. Groth and H. Lehrach, *Nucleic Acids Res.*, 2003, **31**, 3712–3715.

49 G. Zehetner, *Nucleic Acids Res.*, 2003, **31**, 3799–3803.

50 S. Khan, G. Situ, K. Decker and C. J. Schmidt, *Bioinformatics*, 2003, **19**, 2484–2485.

51 D. M. Martin, M. Berriman and G. J. Barton, *BMC Bioinf.*, 2004, **5**, 178.

52 A. Vinayagam, C. del Val, F. Schubert, R. Eils, K. H. Glatting, S. Suhai and R. Konig, *BMC Bioinf.*, 2006, **7**, 161.

53 L. J. Jensen, D. W. Ussery and S. Brunak, *Genome Res.*, 2003, **13**, 2444–2449.

54 T. Hawkins, S. Luban and D. Kihara, *Protein Sci.*, 2006, **15**, 1550–1556.

55 J. Gioia, S. Yerrapragada, X. Qin, H. Jiang, O. C. Igboeli, D. Muzny, S. Dugan-Rocha, Y. Ding, A. Hawes, W. Liu, L. Perez, C. Kovar, H. Dinh, S. Lee, L. Nazareth, P. Blyth, M. Holder, C. Buhay, M. R. Tirumalai, Y. Liu, I. Dasgupta, L. Bokhetache, M. Fujita, F. Karouia, P. Eswara Moorthy, J. Siefert, A. Uzman, P. Buzumbo, A. Verma, H. Zwiya, B. D. McWilliams, A. Olowu, K. D. Clinkenbeard, D. Newcombe, L. Golebiewski, J. F. Petrosino, W. L. Nicholson, G. E. Fox, K. Venkateswaran, S. K. Highlander and G. M. Weinstock, *PLoS One*, 2007, **2**, e928.

56 S. K. Highlander, K. G. Hulten, X. Qin, H. Jiang, S. Yerrapragada, E. O. Mason, Y. Shang, T. M. Williams, R. M. Fortunov, Y. Liu, O. Igboeli, J. Petrosino, M. Tirumalai, A. Uzman, G. E. Fox, A. M. Cardenas, D. M. Muzny, L. Hemphill, Y. Ding, S. Dugan, P. R. Blyth, C. J. Buhay, H. H. Dinh, A. C. Hawes, M. Holder, C. L. Kovar, S. L. Lee, W. Liu, L. V. Nazareth, Q. Wang, J. Zhou, S. L. Kaplan and G. M. Weinstock, *BMC Microbiol.*, 2007, **7**, 99.

57 G. Lopez, A. Rojas, M. Tress and A. Valencia, *Proteins*, 2007, **69**, 165–174.

58 C. L. Myers, D. R. Barrett, M. A. Hibbs, C. Huttenhower and O. G. Troyanskaya, *BMC Genomics*, 2006, **7**, 187.

59 M. C. Costanzo, M. E. Crawford, J. E. Hirschman, J. E. Kranz, P. Olsen, L. S. Robertson, M. S. Skrzypek, B. R. Braun, K. L. Hopkins, P. Kondu, C. Lengieza, J. E. Lew-Smith, M. Tillberg and J. I. Garrels, *Nucleic Acids Res.*, 2001, **29**, 75–79.

60 G. D. Bader and C. W. Hogue, *BMC Bioinf.*, 2003, **4**, 2.

61 A. Baudot, D. Martin, P. Mouren, F. Chevenet, A. Guenoche, B. Jacq and C. Brun, *Bioinformatics*, 2006, **22**, 248–250.

62 B. P. Kelley, B. Yuan, F. Lewitter, R. Sharan, B. R. Stockwell and T. Ideker, *Nucleic Acids Res.*, 2004, **32**, W83–88.

63 B. Adamcsek, G. Palla, I. J. Farkas, I. Derenyi and T. Vicsek, *Bioinformatics*, 2006, **22**, 1021–1023.

64 U. Karaoz, T. M. Murali, S. Letovsky, Y. Zheng, C. Ding, C. R. Cantor and S. Kasif, *Proc. Natl. Acad. Sci. U. S. A.*, 2004, **101**, 2888–2893.

65 B. R. Zeeberg, W. Feng, G. Wang, M. D. Wang, A. T. Fojo, M. Sunshine, S. Narasimhan, D. W. Kane, W. C. Reinhold, S. Lababidi, K. J. Bussey, J. Riss, J. C. Barrett and J. N. Weinstein, *Genome Biol.*, 2003, **4**, R28.

66 S. Zhong, K. F. Storch, O. Lipan, M. C. Kao, C. J. Weitz and W. H. Wong, *Appl. Bioinf.*, 2004, **3**, 261–264.

67 K. D. Dahlquist, N. Salomonis, K. Vranizan, S. C. Lawlor and B. R. Conklin, *Nat. Genet.*, 2002, **31**, 19–20.

68 W. Tong, X. Cao, S. Harris, H. Sun, H. Fang, J. Fuscoe, A. Harris, H. Hong, Q. Xie, R. Perkins, L. Shi and D. Casciano, *Environ. Health Perspect.*, 2003, **111**, 1819–1826.

69 S. Patel and J. Lyons-Weiler, *Appl. Bioinf.*, 2004, **3**, 49–62.

70 V. G. Tusher, R. Tibshirani and G. Chu, *Proc. Natl. Acad. Sci. U. S. A.*, 2001, **98**, 5116–5121.

71 N. Dean and A. E. Raftery, *BMC Bioinf.*, 2005, **6**, 173.

72 P. Puntervoll, R. Linding, C. Gemund, S. Chabanis-Davidson, M. Mattingsdal, S. Cameron, D. M. Martin, G. Ausiello, B. Brannetti, A. Costantini, F. Ferre, F. Maselli, A. Via, G. Cesareni, F. Diella, G. Superti-Furga, L. Wyrwicz, C. Ramu, C. McGuigan, R. Gudavalli, I. Letunic, P. Bork, L. Rychlewski, B. Kuster, M. Helmer-Citterich, W. N. Hunter, R. Aasland and T. J. Gibson, *Nucleic Acids Res.*, 2003, **31**, 3625–3630.

73 E. M. Zdobnov and R. Apweiler, *Bioinformatics*, 2001, **17**, 847–848.

74 E. de Castro, C. J. Sigrist, A. Gattiker, V. Bulliard, P. S. Langendijk-Genevaux, E. Gasteiger, A. Bairoch and N. Hulo, *Nucleic Acids Res.*, 2006, **34**, W362–365.

75 R. D. Finn, J. Mistry, B. Schuster-Bockler, S. Griffiths-Jones, V. Hollich, T. Lassmann, S. Moxon, M. Marshall, A. Khanna, R. Durbin, S. R. Eddy, E. L. Sonnhammer and A. Bateman, *Nucleic Acids Res.*, 2006, **34**, D247–251.

76 I. Letunic, R. R. Copley, B. Pils, S. Pinkert, J. Schultz and P. Bork, *Nucleic Acids Res.*, 2006, **34**, D257–260.

77 I. Friedberg, T. Harder and A. Godzik, *Nucleic Acids Res.*, 2006, **34**, W379–381.

78 Nomenclature Committee of the International Union of Biochemistry and Molecular Biology (Nc-Iubmb), Enzyme Supplement 5 (1999), *Eur. J. Biochem.*, 1999, **264**, 610–650.