BIOINFORMATICS

Vol. 27 ISMB 2011, pages i374–i382
doi:10.1093/bioinformatics/btr236

Small sets of interacting proteins suggest functional linkage mechanisms via Bayesian analogical reasoning

Edoardo M. Airoldi[1,*], Katherine A. Heller[2,*] and Ricardo Silva[3,*]

[1]Department of Statistics and FAS Center for Systems Biology, Harvard University, Cambridge, MA 02138,
[2]Brain and Cognitive Sciences, Massachussetts Institute of Technology, Cambridge, MA 02114, USA and
[3]Department of Statistical Science, University College London, 1-19 Torrington Place, London WC1E 6BT, UK

ABSTRACT

Motivation: Proteins and protein complexes coordinate their activity to execute cellular functions. In a number of experimental settings, including synthetic genetic arrays, genetic perturbations and RNAi screens, scientists identify a small set of protein interactions of interest. A working hypothesis is often that these interactions are the observable phenotypes of some functional process, which is not directly observable. Confirmatory analysis requires finding other pairs of proteins whose interaction may be additional phenotypical evidence about the same functional process. Extant methods for finding additional protein interactions rely heavily on the information in the newly identified set of interactions. For instance, these methods leverage the attributes of the individual proteins directly, in a supervised setting, in order to find relevant protein pairs. A small set of protein interactions provides a small sample to train parameters of prediction methods, thus leading to low confidence.

Results: We develop RBSets, a computational approach to ranking protein interactions rooted in analogical reasoning; that is, the ability to learn and generalize relations between objects. Our approach is tailored to situations where the training set of protein interactions is small, and leverages the attributes of the individual proteins indirectly, in a Bayesian ranking setting that is perhaps closest to propensity scoring in mathematical psychology. We find that RBSets leads to good performance in identifying additional interactions starting from a small evidence set of interacting proteins, for which an underlying biological logic in terms of functional processes and signaling pathways can be established with some confidence. Our approach is scalable and can be applied to large databases with minimal computational overhead. Our results suggest that analogical reasoning within a Bayesian ranking problem is a promising new approach for real-time biological discovery.

Availability: Java code is available at: http://www.gatsby.ucl.ac.uk/∼rbas/.

Contact: airoldi@fas.harvard.edu; kheller@mit.edu; ricardo@stats.ucl.ac.uk

1 INTRODUCTION

Functional mechanisms in the cell involve cascades of interactions among gene products, mostly proteins and small molecules. In recent years, a number of large-scale efforts have collected and organized data produced by the community in publicly available, online databases. Human curators summarize the current state of our understanding of the cell's functional landscape, for instance, by categorizing genes' and proteins' functional roles, distinguishing those that have been experimentally verified from those that are most probable using well established computational methods (Ashburner *et al*., 2000; Finn *et al*., 2008; Kanehisa and Goto, 2000; Letunic *et al*., 2006; Mewes *et al*., 2004; Mulder *et al*., 2007; SGD). However, the current understanding of signaling and regulation dynamics that instantiate functions in the cell is far from complete. Functional discovery is key. The main challenges are the scale of the problem and the costs of the necessary experimental validation (Evanko, 2009). The space of possible interactions and their multiple functional roles is very large. Exploring it at random is inefficient—expensive and time consuming. A sensible approach is that of prioritizing the experiments, testing the most probable functions and interactions first (Schwartz *et al*., 2008).

Methods routinely used by biologists today to assist biological discovery boil down to assigning priority scores to protein–function pairs and to protein–protein–function triples, individually or in small sets. Here we consider a slightly different scenario, where we want to assign priority scores to protein–protein pairs in the absence of a clear functional implication. We propose a methodology to rank protein–protein pairs that are most similar to a given *small set* of protein–protein pairs of interest. We show that our approach, based on analogical reasoning, is able to identify additional protein–protein pairs that share functional implications with the set of pairs of interest. These results suggest that our method may be useful in establishing new functional implications, which are not necessarily well annotated in existing databases.

Bayesian analogical reasoning, the ability to learn and generalize relations between objects, seems particularly appropriate for biological discovery because it does not estimate feature weights and relations (during training) that are then fixed in stone to produce predictions. Rather, it calibrates a prior distribution on feature weights that takes into account how features relate to one another, but then updates these weights depending on the query, on the fly.

To illustrate the practical benefits of the proposed methodology, consider the following scenario. A certain cellular process is poorly understood and the corresponding gene ontology function is poorly annotated, say, protein kinases. Consider extant protein interaction prediction methods; they typically estimate the correlation structure among features of all known proteins that is useful to predict existing interactions, e.g. with a Bayesian network (Myers *et al*., 2005), store it on a web-server, and use it to rank interactions every time users submit a set of protein interactions. However, if we were to discover a few interactions with protein kinases and submit them as a query, extant systems would not be able to rank other interactions between kinases well, mainly for two reasons: (i) we

*To whom correspondence should be addressed.

are submitting a small training set, and (ii) the correlations that were estimated during the training phase did not include kinase interactions, which were poorly annotated, and the corresponding ranking mechanism (feature weights in a classifier or conditional probability tables in a Bayesian network) will have to rely on poor estimates of the correlations. The proposed methodology on the other hand, addresses the issue of small training sets (i) by regularizing the estimates through the use of a prior distribution on the weights in a Bayesian estimation setting. It gets around the issue of poor annotations and poor parameters estimates during training (ii) by updating the feature weights on the fly using information in the query set of interactions. In this example, our method would be able to attribute more importance to features such as specific protein domains, other features of the protein amino-acid sequences and cellular localization in order to retrieve other kinase interactions with high probability. The proposed method is also computationally scalable to large databases.

In this article, we develop a statistical approach that leverages relational learning: given a set of protein pairs $\mathbf{S} = \{A^{(1)}{:}B^{(1)}, A^{(2)}{:}B^{(2)}, \ldots, A^{(N)}{:}B^{(N)}\}$, it measures how well other pairs $A{:}B$ fit in with the set $\mathbf{S}$. Our work addresses the question of whether the relation between proteins A and B is analogous to those relations found in $\mathbf{S}$. Such questions are particularly relevant in exploratory data analysis, where an investigator might want to search for protein pairs that are analogous to pairs in the query set of interest.

Our approach combines a similarity measure on function spaces with Bayesian analysis to produce a ranking of pairs. It requires data containing attributes of the proteins of interest and a link matrix specifying existing relationships; no further attributes of such relationships are necessary. We illustrate the potential of our method on a collection of protein binding events and on metabolic pathways. We show that our approach can work in practice even if a small set of protein pairs is provided, when evaluated on two functional categorization systems: GO and MIPS. Furthermore, we develop a variational inference algorithm that scales to very-large databases.

Related problems and approaches: similarity-based methods compute the score of a given protein pair as a function of observable attributes (e.g. corresponding genes' expression levels) of the two proteins under examination. Possibly multiple functions are then assigned to each protein P_i using a guilt-by-association principle, e.g. by looking up in a curated database the most frequent functions of the proteins P_j's that interact with P_i with high scores (Butte and Kohane, 2006; Clare and King, 2003; Margolin *et al*., 2006; Markowetz *et al*., 2007). The main obstacle to the success of these methods is that the similarity among proteins' attributes is informative about a shared function only to a minor extent (Margoln and Califano, 2007). Clustering-based methods compute the score of a given protein pair as a function of topological properties of the interactions. The interactome is divided into clusters, or modules, and the inferred memberships of proteins-to-modules are used as attributes to predict protein functions in a number of ways (Adamcsek *et al*., 2006; Altaf-Ul-Amin *et al*., 2006; Bader and Hougue, 2003; Enright *et al*., 2002; Sharan *et al*., 2005). Recent results, however, suggest that a simple non-clustering method that relies on the guilt-by-association principle is more accurate in predicting proteins' functional roles (Song and Singh, 2009). The curation of the interactome is noisy and its degree of inaccuracy is higher than expected (Cusick *et al*., 2009). Proteins participate in the execution of multiple functional processes (Airoldi *et al*., 2008). These are perhaps two of the factors that can explain this surprising result. Data integration methods compute the score of a given protein pair as a function that combines observed attributes, both of individual proteins and of protein pairs, from multiple data sources. These integrative methods have been the most successful to date in predicting protein function (Fraser and Marcotte, 2004; Guan *et al*., 2008; Llewellyn and Eisenberg, 2008) and networks of functional relationships between proteins (Hess *et al*., 2009; Huttenhower *et al*., 2009; Ideker *et al*., 2002; Jansen *et al*., 2003; Jensen *et al*., 2009; Lee *et al*., 2004; Myers *et al*., 2005; Troyanskaya *et al*., 2003; von Mering *et al*., 2005) from large collections of data. The main drawback of these approaches is that they often involve a hodgepodge of different scores that only resemble P-values (Schervish, 1996; Sterne and Smith, 2001). Multiple scores are usually combined using ad hoc considerations that are specific to the data under examination. The computational burden is substantial.

Our contribution: we introduce a new approach that determines similarity between protein pairs by essentially computing similarity between predictive functions that relate proteins pairs to functions. Our methodology is based on the idea of analogical reasoning.

The proposed methodology departs from the existing approaches in a few aspects. First, we explicitly address the ranking problem of which pairs of proteins are most similar to an input set of protein pairs. This is different from the typical protein pair prediction problem addressed in the literature. While we rely on similarity to rank protein pairs, we compute similarity between *predictive functions* that map pairs of proteins to functional links, rather than between *attributes* of functionally related proteins. This will require a description of the space where such functions live. Second, our methodology focuses on the case where little evidence is available, i.e. a small set of input pairs. This will require using prior functional knowledge to calibrate a prior on the space of functions that places mass on a most likely subspace of functions. Third, our methodology is rooted in Bayesian statistical methodology. It captures and updates prior knowledge about proteins' functions stored in online databases within a hierarchical Bayesian model. It can easily be integrated in a computational pipeline for general use. Fourth, our variational inference algorithm scales to very-large databases.

We consider two case studies where the underlying functional implications of the interactions we consider correspond to physical protein binding events and to signaling events within a metabolic pathway. We quantify the extent to which small sets of interacting proteins are suggestive of functional linkage mechanisms on the collection of pathways in KEGG and of functions in MIPS.

2 METHODS

To define an analogy is to define a measure of similarity between structures of related objects. In our setting, we need to measure the similarity between pairs of objects. The key aspect that distinguishes our approach from others is that we focus on the similarity between *predictive functions* that map pairs to links, rather than focusing on the similarity between the *attributes* of objects in a candidate pair and the features of objects in the query pairs.

As an illustration, consider an analogical reasoning question from a SAT-like exam where for a given pair (say, *water:river*) we have to choose, out of 5 pairs, the one that best matches the type of relation implicit in such a 'query'. In this case, it is reasonable to say *car:highway* would be a better

DOCTOR:HOSPITAL ::
A) sports fan : stadium
B) cow : farm
C) professor : college
D) criminal : jail
E) food : grocery store

Fig. 1. Example with words.

YDL061C : YLR167W ::
A) YBR084CA : YJL189W
B) YBL092W : YKR094C
C) YDL083C : YGL189C
D) YBL027W : YJL189W
E) YDR178W : YKL148C

Fig. 2. Example with proteins.

match than (the somewhat nonsensical) *soda:ocean*, since cars flow on a highway, and so does water in a river. Notice that if we were to measure the similarity between *objects* instead of *relations*, *soda:ocean* would be a much closer pair, since *soda* is similar to *water*, and *ocean* is similar to *river*.

It is reasonable to infer relational similarity from individual object features (Gentner and Medina, 1998). What is needed is a mechanism by which object features should be weighted in a particular relational similarity problem. We postulate that, in analogical reasoning, similarity between features of objects is only meaningful to the extent to which such features are useful to predict the existence of the relationships.

2.1 An illustrative example

As an intuitive illustration, consider university admission exams, like the American Scholastic Assessment Test (SAT) and Graduate Record Exam (GRE). These exams used to include a section on analogical reasoning. A prototypical analogical reasoning question is shown in Figure 1. The examinee has to answer which of the 5 pairs best matches the relation implicit in `DOCTOR:HOSPITAL`. Although all candidate pairs interact in some way, pair `professor:college` seems to best capture the notion of (*object*, *place of work*) implicit in the relation between doctor and hospital.

Performing this type of analogical reasoning in an automated way may be useful in less mundane domains, where it is hard to reason about relations that have not yet been discovered. Consider the question shown in Figure 2, composed solely of pairs of interacting proteins according to the MIPS classification system (Mewes *et al*., 2004). In the given question, `YDL061C` is a protein of type *cytoplasm* (category 40.03), while `YLR167W` is of type *cytoplasmic and nuclear degradation* (category 6.13.01). Such is also the case of B) `YBL092W : YKR094C`. Other pairs contain members which are both in the 40.03 category but none in 6.13.01, and are in this sense not as close to the question pair as option B. In a realistic situation where the categorization system at hand contains a few annotation errors and a large number of omissions, automated strategies for analogical reasoning that leverage the current state of our knowledge about the proteins would be useful.

Dividing the population of protein interactions into *subpopulations with similar mechanisms of linkage* is therefore a way of categorizing proteins and their functional roles. The population of interacting pairs of proteins is not uniform. The biological mechanism by which protein pair P1:P2 is linked might not be the same mechanism behind the linkage of P3:P4, as illustrated above. The result of this effort is that taxonomies such as the Gene Ontology (Ashburner *et al*., 2000), or the Munich Institute for Protein Sequencing (MIPS) database (Mewes *et al*., 2004) can be enriched by suggesting analogical similarities of protein interactions.

2.2 Statistical background on analogical reasoning

Probabilities can be exploited as a measure of similarity. Let R be a binary random variable representing whether an arbitrary data point X is 'relevant' for a given query set $\mathbf{S}$ ($R=1$) or not ($R=0$). Let $P(\cdot \mid \cdot)$ be a generic probability mass function or density function, with its meaning given by the context. Points are ranked in decreasing order by the following criterion

$$\frac{P(R=1 \mid X, \mathbf{S})}{P(R=0 \mid X, \mathbf{S})} = \frac{P(R=1 \mid \mathbf{S})}{P(R=0 \mid \mathbf{S})} \frac{P(X \mid R=1, \mathbf{S})}{P(X \mid R=0, \mathbf{S})},$$

which is equivalent to ranking points by the expression

$$\log P(X \mid R=1, \mathbf{S}) - \log P(X \mid R=0, \mathbf{S}) \quad (1)$$

The challenge is to define what form $P(X \mid R=r, \mathbf{S})$ should assume. It is not practical to collect labeled data in advance which, for every possible class of queries, will give an estimate for $P(R=1 \mid X, \mathbf{S})$: in general, one cannot anticipate which classes of queries will exist. Methods based on Bayesian networks with known labels (e.g. Myers *et al*., 2005), for instance, would suffer in terms of recall. Instead, a variety of approaches has been developed in the literature in order to define a suitable instantiation of (1). These include a method that builds a classifier on-the-fly using $\mathbf{S}$ as elements of the positive class $R=1$, and a random subset of data points as the negative class $R=0$ (e.g. Turney, 2008).

In this setup, the event '$R=1$' is equated with the event that X and the elements of $\mathbf{S}$ are i.i.d points generated by the same model. The event '$R=0$' is the event by which X and $\mathbf{S}$ are generated by two independent models: one for X and another for $\mathbf{S}$. The parameters of all models are random variables that have been integrated out, with fixed (and common) hyperparameters. The result is the instantiation of (1) as

$$\log P(X \mid \mathbf{S}) - \log P(X) = \log \frac{P(X, \mathbf{S})}{P(X)P(\mathbf{S})}, \quad (2)$$

the Bayesian *score function* by which we rank points X given a query $\mathbf{S}$. The right-hand side was rearranged to provide a more intuitive graphical model. From this graphical model interpretation we can see that the score function is a Bayes factor comparing two models (Kass and Raftery, 1995).

Next, we describe how the Bayesian sets method can be adapted to define analogical similarity in biological networks settings.

2.3 Bayesian analogical similarity for protein pairs

Let $\mathcal{A}$ and $\mathcal{B}$ represent object spaces. To say that an interaction $A{:}B$ is analogous to $\mathbf{S}=\{A^{(1)}{:}B^{(1)}, A^{(2)}{:}B^{(2)}, \ldots, A^{(N)}{:}B^{(N)}\}$ amounts to implicitly defining a measure of similarity between the pair $A{:}B$ and the set of pairs $\mathbf{S}$, where each query item $A^{(k)}{:}B^{(k)}$ corresponds to some pair $A^i{:}B^j$. However, this similarity is not directly derived from the similarity of the information contained in the distribution of objects themselves, $\{A^i\} \subset \mathcal{A}$, $\{B^i\} \subset \mathcal{B}$. Rather, the similarity between $A{:}B$ and the set $\mathbf{S}$ is defined in terms of the similarity of the *functions* mapping the pairs as being linked. Each possible function captures a different possible relationship between the objects in the pair.

Bayesian analogical reasoning: Consider a space of latent functions in $\mathcal{A} \times \mathcal{B} \to \{0,1\}$. Assume that A and B are two objects classified as linked by some unknown function $f(A,B)$, i.e. $f(A,B)=1$. We want to quantify how similar the function $f(A,B)$ is to the function $g(\cdot,\cdot)$, which classifies all pairs $(A^i,B^j) \in \mathbf{S}$ as being linked, i.e. where $g(A^i,B^j)=1$. The similarity depends on the observations $\{\mathbf{S},A,B\}$ and our prior distribution over $f(\cdot,\cdot)$ and $g(\cdot,\cdot)$.

Functions $f(\cdot)$ and $g(\cdot)$ are unobserved, hence the need for a prior that will be used to integrate over the function space. Our similarity metric will be defined using Bayes factors, as explained next.

2.3.1 Scoring analogy of linkage functions using logistic regression For simplicity, we will consider a family of latent functions that is parameterized by a finite-dimensional vector: the logistic regression function with multivariate Gaussian priors for its parameters.

For a particular pair ($A^i \in \mathcal{A}$, $B^j \in \mathcal{B}$), let $X^{ij} = [\Phi_1(A^i, B^j)\ \Phi_2(A^i, B^j)\ \ldots\ \Phi_K(A^i, B^j)]^\mathsf{T}$ be a point on a feature space defined by the mapping $\Phi : \mathcal{A} \times \mathcal{B} \to \Re^K$. This feature space mapping computes a K-dimensional vector of attributes of the pair that may potentially be relevant to predicting the relation between the objects in the pair. Let $L^{ij} \in \{0, 1\}$ be an indicator of the existence of a link or relation between A^i and B^j in the database. Let $\Theta = [\theta_1, \ldots, \theta_K]^\mathsf{T}$ be the parameter vector for our logistic regression model such that

$$P(L^{ij} = 1 | X^{ij}, \Theta) = \text{logistic}(\Theta^\mathsf{T} X^{ij}), \tag{3}$$

and $\text{logistic}(x) = (1 + e^{-x})^{-1}$ is the standard mapping from $\mathbb{R} \to [0, 1]$.

We now apply the same score function underlying the Bayesian methodology explained in Section 2.2. However, instead of comparing objects by marginalizing over the parameters of their feature distributions, we compare *functions* for link indicators by marginalizing over the parameters of the functions.

Let $\mathbf{L}^\mathbf{S}$ be the vector of link indicators for $\mathbf{S}$: in fact each $L \in \mathbf{L}^\mathbf{S}$ has the value $L = 1$, indicating that every pair of objects in $\mathbf{S}$ is linked. Consider the following Bayes factor:

$$\frac{P(L^{ij} = 1,\ \mathbf{L}^\mathbf{S} = 1 \mid X^{ij}, \mathbf{S})}{P(L^{ij} = 1 \mid X^{ij}) P(\mathbf{L}^\mathbf{S} = 1 \mid \mathbf{S})} \tag{4}$$

This is an adaptation of Equation (2) where relevance is defined now by whether L^{ij} and $\mathbf{L}^\mathbf{S}$ were generated by the same model, for fixed $\{X^{ij}, \mathbf{S}\}$. In one sense, this is a discriminative Bayesian sets model, where we predict links instead of modeling joint object features. Since we are integrating out Θ, a prior for this parameter vector is needed.

Thus, each pair (A^i, B^j) is evaluated with respect to a query set $\mathbf{S}$ by the score function given in (4), rewritten after taking a logarithm and dropping constants as:

$$\begin{aligned} score(A^i, B^j) \ = \ & \log P(L^{ij} = 1 \mid X^{ij}, \mathbf{S}, \mathbf{L}^\mathbf{S} = 1) \\ & - \log P(L^{ij} = 1 \mid X^{ij}) \end{aligned} \tag{5}$$

The exact details of our procedure are as follows. We are given a relational database $(\mathcal{D}_A, \mathcal{D}_B, \mathcal{L}_{AB})$. Dataset $\mathcal{D}_A$ ($\mathcal{D}_B$) is a sample of objects of type $\mathcal{A}$ ($\mathcal{B}$). Relationship table $\mathcal{L}_{AB}$ is a binary matrix modeled as generated from a logistic regression model of link existence. A query proceeds according to the following steps:

(i) the user selects a set of pairs $\mathbf{S}$ that are linked in the database, where the pairs in $\mathbf{S}$ are assumed to have some relation of interest;

(ii) perform Bayesian inference to obtain the corresponding posterior distribution for Θ, $P(\Theta | \mathbf{S}, \mathbf{L}^\mathbf{S})$, given a Gaussian prior $P(\Theta)$; and

(iii) iterate through all linked pairs, computing the following for each pair:

$$P(L^{ij} = 1 | X^{ij}, \mathbf{S}, \mathbf{L}^\mathbf{S} = 1) = \int P(L^{ij} = 1 | X^{ij}, \Theta) P(\Theta | \mathbf{S}, \mathbf{L}^\mathbf{S} = 1) d\Theta,$$

where $P(L^{ij} = 1 | X^{ij})$ is similarly computed by integrating over $P(\Theta)$. All pairs are presented in decreasing order according to the score in Equation (5).

The integral presented above does not have a closed formula. Because computing the integrals by a Monte Carlo method for a large number of pairs would be unreasonable, we use a variational approximation (Airoldi, 2007; Jordan *et al.*, 1999). Figure 3 presents a summary of the approach.

The suggested setup scales as $O(K^3)$ with the feature space dimension, due to the matrix inversions necessary for (variational) Bayesian logistic regression (Jaakkola and Jordan, 2000). A less precise approximation to $P(\Theta | \mathbf{S}, \mathbf{L}^\mathbf{S})$ can be imposed if the dimensionality of Θ is too high. However, it is important to point out that once the initial integral $P(\Theta \mid \mathbf{S}, \mathbf{L}^\mathbf{S})$ is approximated, each score function can be computed at a cost of $O(K^2)$.

Our analogical reasoning formulation is a relational model in that it models the presence and absence of interactions between objects. By conditioning on the link indicators, the similarity score between A:B and C:D is always a function of pairs (A, B) and (C, D) that is not in general decomposable as similarities between A and C, and B and D.

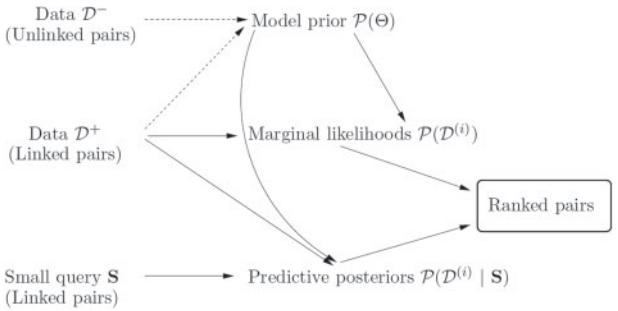

Fig. 3. General framework of the procedure: first, a 'prior' over parameters Θ for a link classifier is defined empirically using linked and unlinked pairs of points (the dashed edges indicate that creating a prior empirically is optional, but in practice we rely on this method). Given a query set $\mathbf{S}$ of linked pairs of interest, the system computes the predictive likelihood of each linked pair $\mathcal{D}^{(i)} \in \mathcal{D}^+$ and compares it to the conditional predictive likelihood, given the query. This defines a measure of similarity with respect to $\mathbf{S}$ by which all pairs in $\mathcal{D}^+$ are sorted.

2.3.2 Empirical priors and calibration using biological databases The choice of prior is based on the observed data, in a way that is equivalent to the choice of priors used in the original formulation of Bayesian sets (Ghahramani and Heller, 2005). Let $\widehat{\Theta}$ be the maximum-likelihood estimator (MLE) of Θ using the relational database $(\mathcal{D}_A, \mathcal{D}_B, \mathcal{L}_{AB})$. Since the number of possible pairs grows at a quadratic rate with the number of objects, we do not use the whole database for MLE. Instead, to get $\widehat{\Theta}$ we use all linked pairs as members of the 'positive' class ($L = 1$), and subsample unlinked pairs as members of the 'negative' class ($L = 0$). We subsample by sampling each object uniformly at random from the respective datasets $\mathcal{D}_A$ and $\mathcal{D}_B$ to get a new pair. Since link matrices $\mathcal{L}_{AB}$ are usually very sparse, in practice this will almost always provide an unlinked pair. Section 3 provides more details.

We use the prior $P(\Theta) = \mathcal{N}(\widehat{\Theta}, (c\widehat{\mathbf{T}})^{-1})$, where $\mathcal{N}(\mathbf{m}, \mathbf{V})$ is a normal of mean $\mathbf{m}$ and variance $\mathbf{V}$. Matrix $\widehat{\mathbf{T}}$ is the empirical second moments matrix of the linked object features, although a different choice might be adequate for different applications. Constant c is a smoothing parameter set by the user. In all of our experiments, we set c to be equal to the number of positive pairs. A good choice of c might be important to obtain maximum performance, but we leave this issue as future work. (Wang *et al.*, 2009) present some sensitivity analysis results.

Empirical priors are a sensible choice, since this is a retrieval, not a predictive, task. Basically, the entire dataset is the population, from which prior information is obtained on possible query sets. A data-dependent prior based on the population is important for an approach such as Bayesian sets, since deviances from the 'average' behavior in the data are useful to discriminate between subpopulations.

2.3.3 Extensions to continuous/multivariate relations Although here we focused on measuring similarity of qualitative relationships, the same idea could be extended to *continuous* (or ordinal) measures of relationship, or relationships where each L^{ij} is a vector. Several similarity metrics can be defined on this vector of continuous relationships. Given data on protein attributes, one can easily modify our approach by substituting the logistic regression component with some multiple regression model.

3 RESULTS

The budding yeast is a unicellular organism that has become a de-facto model organism for the study of molecular and cellular biology (Botstein *et al.*, 1997). There are about 6000 proteins in

the budding yeast, which interact in a number of ways (Cherry *et al*., 1997). For instance, proteins bind together to form protein complexes, the physical units that carry out most functions in the cell (Krogan *et al*., 2006). In recent years, significant resources have been directed to collect experimental evidence of physical proteins binding, in an effort to infer and catalogue protein complexes and their multifaceted functional roles (Fields and Song, 1989; Ho *et al*., 2002; Ito *et al*., 2000; Uetz *et al*., 2000). Currently, there are four main sources of interactions between pairs of proteins that target proteins localized in different cellular compartments with variable degrees of success: (i) literature curated interactions (Reguly *et al*., 2006); (ii) yeast two-hybrid (Y2H) interaction assays (Yu *et al*., 2008); (iii) protein fragment complementation (PCA) interaction assays (Tarassov *et al*., 2008); and (iv) tandem affinity purification (TAP) interaction assays (Gavin *et al*., 2006; Krogan *et al*., 2006). These collections include a total of about 12 292 protein interactions (Jensen and Bork, 2008), although the number of such interactions is estimated to be between 18 000 (Yu *et al*., 2008) and 30 000 (von Mering *et al*., 2002).

3.1 Design of experiments

We consider multiple functional categorization systems for the proteins in budding yeast. For evaluation purposes, we use individual proteins' functional annotations curated by the Munich Institute for Protein Sequencing (MIPS) (Mewes *et al*., 2004), those by the Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa and Goto, 2000), and those by the Gene Ontology consortium (GO) (Ashburner *et al*., 2000). We consider multiple collections of physical protein interactions that encode alternative semantics. Physical protein interactions in the MIPS curated collection measure physical binding events observed experimentally in Y2H and TAP experiments, whereas physical protein-to-protein interactions in the KEGG curated collection measure a number of different modes of interactions, including phosporelation, methylation and physical binding, all taking place in the context of a specific signaling pathway. So we have three possible functional annotation databases (MIPS, KEGG and GO) and two possible link matrices (MIPS and KEGG), which can be combined.

Our experimental pipeline is as follows. (i) Pick a database of functional annotations, say MIPS, and a collection of interactions, say MIPS (again). (ii) Pick a pair of categories, M_1 and M_2. For instance, take M_1 to be *cytoplasm* (MIPS 40.03) and M_2 to be *cytoplasmic and nuclear degradation* (MIPS 06.13.01). (iii) Sample, uniformly at random and without replacement, a set $\mathbf{S}$ of 15 interactions in the chosen collection. (iv) Rank other interacting pairs[1] according to the score in Equation (5) and, for comparison purposes, according to three other approaches to be described in Section 3.1.4. (v) The process is repeated for a large number of pairs $M_1 \times M_2$, and 5 different query sets $\mathbf{S}$ are generated for each pair of categories. (vi) Calculate an evaluation metric for each query and each of the four scores. Report a comparative summary of results.

3.1.1 Protein specific features The protein specific features were generated using the datasets summarized in Table 1 and an additional dataset (Qi *et al*., 2006), from which 20 gene expression attributes were obtained. Each gene expression attribute for a protein pair P_i:P_j corresponds the correlation coefficient between the expression levels of corresponding genes. The 20 different attributes are obtained from 20 different experimental conditions as measured by microarrays. We did not use pairs of proteins from Qi *et al*. (2006) for which we did not have data in the datasets listed in Table 1. This resulted in approximately 6000 positively linked data points for the MIPS network and 39 000 for KEGG. We generated another 25 protein–protein gene expression features from the data in Table 1 using the same procedure based on correlation coefficients. This gives a total of 45 attributes, corresponding to the main dataset used in our relational Bayesian sets runs. Another dataset was generated using the remaining (i.e. non-microarray) features of Table 1. Such features are binary and highly sparse, with most entries being 0 for the majority of linked pairs. We removed attributes for which we had fewer than 20 linked pairs with positive values according to the MIPS network. The total number of extra binary attributes was 16. Several measurements were missing. We imputed missing values for each variable in a particular datapoint by using its empirical average among the observed values.

Given the 45 or 61 attributes of a given pair $\{P_i, P_j\}$, we applied a non-linear transformation where we normalize the vector by its Euclidean norm in order to obtain our feature table $\mathbf{X}$.

Table 1. Collection of datasets used to generate protein-specific features

Type of data	Data sources for our study
Gene expression	Brem *et al*. (2005); Gasch *et al*. (2000) Primig *et al*. (2000); Yvert *et al*. (2003)
Synthetic genetic int.	Breitkreutz *et al*. (2003); SGD
Cellular localization	Huh *et al*. (2003)
TF binding sites	Harbison *et al*. (2004); TRANSFAC
Sequence data	Altschul *et al*. (1990); Zhu and Zhang (1999)

3.1.2 Calibrating the prior for Θ We fit a logistic regression classifier using a MLE and our data, obtaining the estimate $\widehat{\Theta}$. Our choice of covariance matrix $\widehat{\Sigma}$ for Θ is defined to be a rescaling of a squared norm of the data:

$$(\widehat{\Sigma})^{-1} = \mathbf{X}_{POS}^{\mathsf{T}} \mathbf{X}_{POS}, \tag{6}$$

where $\mathbf{X}_{POS}$ is the matrix containing the protein–protein features only of the linked pairs used in the MLE computation.

3.1.3 Evaluation metrics We propose an objective measure of evaluation that is used to compare different algorithms. Consider a query set $\mathbf{S}$, and a ranked response list $\mathbf{R} = \{R^1, R^2, R^3, \ldots, R^N\}$ of protein–protein pairs. Every element of $\mathbf{S}$ is a pair of proteins P_i:P_j such that P_i is of class M_i and P_j is of class M_j, where M_i and M_j are classes from either MIPS, KEGG or Gene Ontology. In general, proteins belong to multiple classes. The retrieval algorithm that generates $\mathbf{R}$ does not receive any information concerning the MIPS, KEGG or GO taxonomy. $\mathbf{R}$ starts with the linked protein pair that is judged most similar to $\mathbf{S}$, followed by the other protein pairs in the population, in decreasing order of similarity. Each algorithm has its own measure of similarity.

The evaluation criterion for each algorithm is as follows: as before, we generate a precision-recall curve and calculate the area under the curve (AUC). We also calculate the proportion (TOP10),

[1]The portion of ranked list that is relevant for evaluation purposes is limited to a subset of the protein–protein interactions—see Section 3.1.3.

among the top 10 elements in each ranking, of pairs that match the original $\{M_1, M_2\}$ selection (i.e. a 'correct' P_i:P_j is one where P_i is of class M_1 and P_j of class M_2, or vice-versa. Notice that each protein belongs to multiple classes, so both conditions might be satisfied.) Since a researcher is only likely to look at the top ranked pairs, it makes sense to define a measure that uses only a subset of the ranking. AUC and TOP10 are our two evaluation measures.

The original classes $\{M_1, M_2\}$ are known to the experimenter but not known to the algorithms. Our criterion is rather stringent, in the sense it requires a perfect match of each R^I with the MIPS, KEGG or GO categorization. There are several ways by which a pair R^I might be analogous to the relation implicit in **S**, and they do not need to agree with MIPS, GO or KEGG. Still, if we are willing to believe that these standard categorization systems capture functional organization of proteins at some level, this must lead to association between categories given to **S** and relevant subpopulations of protein–protein interactions similar to **S**. Therefore, the corresponding AUC and TOP10 are useful tools for comparing different algorithms even if the actual measures are likely to be pessimistic for a fixed algorithm.

3.1.4 Competing algorithms We compare our method against a variant of it and two similarity metrics widely used for information retrieval: (i) the cosine score (Manning *et al*., 2008), denoted by COS; (ii) the nearest neighbor score, denoted by NNS; (iii) the relational maximum-likelihood sets score, denoted by MLS. The nearest neighbor score measures the minimum Euclidean distance between R^I and any individual point in **S**, for a given query set **S** and a given candidate point R^I. The relational MLS is a variation of RBSETS where we initially sample a subset of the unlinked pairs (10 000 points in our setup) and, for each query **S**, we fit a logistic regression model to obtain the parameter estimate $\Theta_{\mathbf{S}}^{\mathrm{MLE}}$. We also use a logistic regression model fit to the *whole* dataset (the same one used to generate the prior for RBSETS), giving the estimate Θ^{MLE}. A new score, analogous to (5), is given by $\log P(L^{ij}=1|X^{ij}, \Theta_{\mathbf{S}}^{\mathrm{MLE}}) - \log P(L^{ij}=1|X^{ij}, \Theta^{\mathrm{MLE}})$, i.e. we do not integrate out the parameters or use a prior, but instead the models are fixed at their respective estimates.

Neither COS or NNS can be interpreted as measures of analogical similarity, in the sense that they do not take into account how the protein pair features **X** contribute to their interaction[2].

3.2 Analysis of physical interactions (MIPS)

For this batch of experiments, we use the MIPS network of protein–protein interactions to define the relationships. In the initial experiment, we selected queries from all combinations of MIPS classes for which there were at least 50 linked pairs P_i:P_j in the network that satisfied the choice of classes. Each query set contained 15 pairs. After removing the MIPS-categorized proteins for which we had no feature data, we ended up with a total of 6125 proteins and 7788 positive interactions. We set the prior for RBSETS using a sample of 225 842 pairs labeled as having no interaction, as selected by (Qi *et al*., 2006).

[2]As a consequence, none uses negative data. Another consequence is the necessity of modeling the input space that generates **X**, a difficult task given the dimensionality and the continuous nature of the features.

Table 2. Number of times each method wins when querying pairs of MIPS classes using the MIPS protein–protein interaction network

Method	#AUC	#TOP10	#AUC.S	#TOP10.S
COS	240	294	219	277
NNS	42	122	28	75
MLS	105	270	52	198
RBSETS	542	556	578	587

Method	#AUC	#TOP10	#AUC.S	#TOP10.S
COS	314	356	306	340
NNS	75	146	62	111
MLS	273	329	246	272
RBSETS	267	402	245	387

The first two columns, #AUC and #TOP10, count the number of times the respective method obtains the best score according to the AUC and TOP10 measures, respectively, among the 4 approaches. This is divided by the number of replications of each query type (5). The last two columns, #AUC.S and #TOP10.S are 'smoothed' versions of this statistic: a method is declared the winner of a round of 5 replications if it obtains the best score in at least 3 out of the 5 replications. The top table shows the results when only the continuous variables are used by RBSETS, and in the bottom table when the discrete variables are also given to RBSETS.

For each tentative query set **S** of categories $\{M_1, M_2\}$, we scored and ranked pairs P_i':P_j' such that both P_i' and P_j' were connected to some protein appearing in **S** by a path of no more than two steps, according to the MIPS network. The reasons for the filtering are 2-fold: to increase the computational performance of the ranking since fewer pairs are scored; and to minimize the chance that undesirable pairs would appear in the top 10 ranked pairs. Tentative queries would not be performed if after filtering we obtained fewer than 50 possible correct matches. Trivial queries, where filtering resulted only in pairs in the same class as the query, were also discarded. The resulting number of unique pairs of categories $\{M_1, M_2\}$ was 931 classes of interactions. For each pair of categories, we sampled our query set **S** 5 times, generating a total of 4655 rankings per algorithm.

We run two types of experiments. In one version, we give to RBSETS the data containing only the 45 (continuous) microarray measurements. In the second variation, we provide to RBSETS all 61 variables, including the 16 sparse binary indicators. However, we noticed that the addition of the 16 binary variables hurts RBSETS considerably. We conjecture that one reason might be the degradation of the variational approximation. Including the binary variables hardly changed the other three methods, so we choose to use the 61 variable dataset for the other methods.

Table 2 summarizes the results of this experiment. We show the number of times each method wins according to both the AUC and TOP10 criteria. The number of wins is presented as divided by 5, the number of random sets generated for each query type $\{M_1, M_2\}$ (notice these numbers do not need to add up to 931, since ties are possible). Moreover, we also presented 'smoothed' versions of this statistic, where we count a method as the winner for any given $\{M_1, M_2\}$ category if, for the group of 5 queries, the method obtains the best result in at least 3 of the sets. The motivation is to smooth out the extra variability added by the particular set of 15 protein pairs for a fixed $\{M_1, M_2\}$. The proposed relational Bayesian sets method is the clear winner according to all measures when we select only

the continuous variables. For this reason, for the rest of this section all analysis and experiments will consider only this case. Table 3 displays a pairwise comparison of the methods. In this Table, we show how often the row method performs better than the column method, among those trials where there was no tie. Again, RBSets dominates.

Another useful summary is the distribution of correct hits in the top 10 ranked elements across queries. This provides a measure of the difficulty of the problem, besides the relative performance of each algorithm. In Table 4, we show the proportion of correct hits among the top 10 for each algorithm for our queries using MIPS categorization and also GO categorization, as explained in the next section. About 14% of the time, all pairs in the top 10 pairs ranked by RBSets were of the intended type, compared to 8% of the second best approach.

3.2.1 Changing the categorization system A variation of this experiment was performed where the protein categorizations do *not* come from the same family as the link network, i.e. where we used the MIPS network but not the MIPS categorization. Instead we performed queries according to Gene Ontology categories. Starting from 150 pre-selected GO categories (Myers *et al*., 2006), we once again generated unordered category pairs $\{M_1, M_2\}$. A total of 179 queries, with 5 replications each (a total of 895 rankings), were generated and the results summarized in Table 5.

This is a more challenging scenario for our approach, which is optimized with respect to MIPS. Still, we are able to outperform other approaches. Differences are smaller, but consistent. In the pairwise comparison of RBSets against the second best method, cos, our method wins 62% of the time by the TOP10 criterion.

3.2.2 The role of filtering In both experiments with the MIPS network, we filtered candidates by examining only a subset of the proteins linked to the elements in the query set by a path of no more than two proteins. It is relevant to evaluate how much coverage of each category pair $\{M_1, M_2\}$ we obtain by this neighborhood selection.

For each query **S**, we calculate the proportion of pairs P_i:P_j of the same categorization $\{M_1, M_2\}$ such that both P_i and P_j are included in the neighborhood. For the MIPS categorization, 93% of the queries resulted in a coverage of at least 75% (with 24% of the queries resulting in perfect coverage). Although filtering implies that some valid pairs will never be ranked, the gain obtained by reducing false positives in the top 10 ranked pairs is considerable (results not shown) across all methods, and the computational gain of reducing the search space is particularly relevant in exploratory data analysis.

Table 3. Pairwise comparison of methods according to the AUC and TOP10 criterion

	COS	NNS	MLS	RBSets
AUC				
COS	–	0.67	0.43	0.30
NNS	0.32	–	0.18	0.06
MLS	0.56	0.81	–	0.25
RBSets	0.69	0.93	0.74	–
TOP10				
COS	–	0.70	0.46	0.30
NNS	0.29	–	0.25	0.11
MLS	0.53	0.74	–	0.28
RBSets	0.69	0.88	0.71	–

Each cell shows the proportion of the trials where the method in the respective row wins over the method in the column, according to both criteria. In each cell, the proportion is calculated with respect to the 4655 rankings where no tie happened.

3.3 Analysis of signaling pathways (KEGG)

We repeated the same experimental setup, now using the KEGG network to define the protein interactions. We selected proteins from

Table 5. Number of times each method wins when querying pairs of GO classes using the MIPS protein–protein interaction network

Method	#AUC	#TOP10	#AUC.S	#TOP10.S
COS	58	73	58	72
NNS	1	10	0	4
MLS	26	55	13	38
RBSets	93	105	101	110

Columns #AUC, #TOP10, #AUC.S and #TOP10.S are defined as in Table 2.

Table 4. Distribution across all queries of the number hits in the top 10 pairs, as ranked by each algorithm

	0	1	2	3	4	5	6	7	8	9	10
	Proportion of top hits using MIPS categories and links specified by the MIPS database										
COS	0.12	0.15	0.12	0.10	0.08	0.07	0.06	0.05	0.04	0.07	0.08
NNS	0.29	0.16	0.14	0.10	0.06	0.05	0.03	0.03	0.03	0.03	0.02
MLS	0.12	0.12	0.12	0.10	0.09	0.08	0.07	0.06	0.07	0.06	0.07
RBSets	0.04	0.08	0.09	0.09	0.09	0.08	0.09	0.07	0.09	0.08	0.14
	Proportion of top hits using GO categories and links specified by the MIPS database										
COS	0.12	0.13	0.11	0.10	0.11	0.09	0.06	0.06	0.04	0.06	0.06
NNS	0.53	0.23	0.07	0.02	0.02	0.02	0.04	0.01	0.00	0.00	0.01
MLS	0.16	0.11	0.12	0.10	0.08	0.08	0.08	0.06	0.05	0.06	0.05
RBSets	0.09	0.09	0.10	0.10	0.08	0.08	0.06	0.08	0.08	0.07	0.12

The more skewed to the right, the better. Notice that using GO categories doubles the number of zero hits for RBSets.

Table 6. Number of times each method wins when querying pairs of KEGG classes using the KEGG protein–protein interaction network

Method	#AUC	#TOP10	#AUC.S	#TOP10.S
COS	159	575	134	507
NNS	30	305	17	227
MLS	290	506	199	431
RBSets	1042	1091	1107	1212

Columns #AUC, #TOP10, #AUC.S and #TOP10.S are defined as in Table 2.

Table 7. Distribution across all queries of the number hits in the top 10 pairs, as ranked by each algorithm

	0	1	2	3	4	5	6	7	8	9	10
	Proportion of top hits using KEGG categories and links specified by the KEGG database										
COS	0.56	0.21	0.08	0.03	0.02	0.01	0.01	0.01	0.01	0.01	0.01
NNS	0.89	0.03	0.04	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
MLS	0.57	0.21	0.08	0.04	0.02	0.01	0.01	0.00	0.00	0.00	0.00
RBSets	0.29	0.24	0.16	0.09	0.06	0.03	0.02	0.01	0.03	0.02	0.01

The more skewed to the right, the better.

the KEGG categorization system for which we had data available. A total of 6125 proteins was selected. The KEGG network is much more dense than MIPS. A total of 38 961 positive pairs and 226 188 negative links were used to generate our empirical prior.

Since the KEGG network is much more dense than MIPS, we filtered our candidate pairs by allowing only proteins that are directly linked to the proteins in the query set **S**. Even under this restriction, we are able to obtain high coverage: the neighborhood of 90% of the queries included all valid pairs of the same category, and essentially all queries included at least 75% of the pairs falling in the same category as the query set. A total of 1523 possible category pairs (7615 queries, considering 5 replications) were generated.

Results are summarized in Table 6. Again, it is evident that RBSets dominates other methods. In the pairwise comparison against COS, RBSets wins 76% of the times according to the TOP10 criterion. However, the ranking problem in the KEGG network was much harder than in the MIPS network (according to our automated non-analogical criterion). We believe that the reason is that, in KEGG, the simple filtering scheme has much less influence, as reflected by the high coverage. The distribution of the number of hits in the top 10 ranked items is shown in Table 7. Despite the success of RBSets relative to the other algorithms, there is room for improvement.

4 CONCLUDING REMARKS

We presented a novel measure of similarity between biological structures based on the principle of analogical comparison. It provides a way of clustering biological data that is considerably different from other methods, due to its focus on analysing the space of functions that map object features to their relations, instead of the feature space itself. For small size queries, our method finds analogies that are functionally relevant among the top matches. We evaluated our approach with thousands of experiments.

This work can be expanded in many ways, including but not limited to: allowing for extra dependencies between interactions that are not due to input features X, scaling up the algorithm to allow for higher-dimensional data, and applying it to other domains such as evaluating analogies between cells from different species. Filtering could also be improved by using criteria other than path lengths in a interaction network. It is also of interest to apply substantive background knowledge in evaluating rankings that take into account the actual analogical similarity of different pairs. We believe several useful variations of our approach can be designed in the future.

Funding: National Science Foundation (under grants no. DMS-0907009 and no. IIS-1017967); National Institute of Health (under grant no. R01 GM096193); Army Research Office Multidisciplinary University Research Initiative (under grant no. 58153-MA-MUR) all to Harvard University; additional funding was provided by the Harvard Medical School's Milton Fund.

Conflict of Interest: none declared.

REFERENCES

Adamcsek,B. *et al.* (2006) CFinder: locating cliques and overlapping modules in biological networks. *Bioinformatics*, **22**, 1021–1023.

Airoldi,E.M. (2007) Getting started in probabilistic graphical models. *PLoS Comput. Biol.*, **3**, e252.

Airoldi,E.M. *et al.* (2008) Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.*, **9**,1981–2014.

Altaf-Ul-Amin,M. *et al.* (2006) Development and implementation of an algorithm for detection of protein complexes in large interaction networks. *BMC Bioinformatics*, **7**.

Altschul,S.F. *et al.* (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Ashburner,M. *et al.* (2000) Gene ontology: tool for the unification of biology. *Nat. Genet.*, **25**,155–170.

Bader,G.D. and Hougue,C.W. (2003) An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, **4**, 2.

Botstein,D. *et al.* (1997) Yeast as a model organism. *Science*, **277**, 1259–1260.

Breitkreutz,B.J. *et al.* (2003) The GRID: the general repository for interaction datasets. *Genome Biol.*, **4**, R23.

Brem,R.B. *et al.* (2005) Genetic interactions between polymorphisms that affect gene expression in yeast. *Nature*, **436**, 701–703.

Butte,A.J. and Kohane,I.S. (2006) Creation and implications of a phenome-genome network. *Nature Biotechnol.*, **24**, 55–62.

Cherry,J.M. *et al.* (1997) Genetic and physical maps of saccharomyces cerevisiae. *Nature*, **387** (Suppl. 6632), 67–73.

Clare,A. and King,R.D. (2003) Predicting gene function in *saccharomyses cerevisiae*. *Bioinformatics*, **19**, 1142–1149.

Cusick,M.E. *et al.* (2009) Literature-curated protein interaction datasets. *Nat. Met.*, **6**, 39–46.

Enright,A.J. *et al.* (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res.*, **30**, 1575–1584.

Evanko,D. (2009) Maturing interactions. *Nat. Met.*, **6**, 2. (Editorial).

Fields,S. and Song,O. (1989) A novel genetic system to detect protein–protein interactions. *Nature*, **340**, 245–246.

Finn,R.D. *et al.* (2008) The pfam protein families database. *Nucleic Acids Res.*, **36**, D281–D288.

Fraser,A.G. and Marcotte,E.M. (2004) A probabilistic view of gene function. *Nat. Genet.*, **36**, 559–564.

Gasch,A.P. *et al.* (2000) Genomic expression programs in the response of yeast cells to environmental changes. *Mol. Biol. Cell*, **11**, 4241–4257.

Gavin,A.-C. *et al.* (2006) Proteome survey reveals modularity of the yeast cell machinery. *Nature*, **440**, 631–636.

Gentner,D. and Medina,J. (1998) Similarity and the development of rules. *Cognition*, **65**, 263–297.

Ghahramani,Z. and Heller,K.A. (2005) Bayesian sets. *Adv. Neural Inform. Proc. Syst.*, **18**,435–442.

Guan,Y. *et al.* (2008) Predicting gene function in a hierarchical context with an ensemble of classifiers. *Genome Biol.*, **9** (Suppl. 1), S3.

Harbison,C.T. *et al.* (2004) Transcriptional regulatory code of a eukaryotic genome. *Nature*, **431**, 99–104.

Hess,D.C. *et al.* (2009) Computationally driven, quantitative experiments discover genes required for mitochondrial biogenesis. *PLoS Genetics*, **5**, e1000407.

Ho,Y. *et al.* (2002) Systematic identification of protein complexes in saccharomyces cerevisiae by mass spectrometry. *Nature*, **415**,180–183.

Huh,W.K. *et al.* (2003) Global analysis of protein localization in budding yeast. *Nature*, **425**, 686–691.

Huttenhower,C. *et al.* (2009) Detailing regulatory networks through large scale data integration. *Bioinformatics*, **25**, 3267–3274.

Ideker,T. *et al.* (2002) Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics*, **18** (Suppl. 1), S233–S240.

Ito,T. *et al.* (2000) Toward a protein–protein interaction map of the budding yeast: a comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins. *Proc. Natl Acad. Sci. USA*, **97**, 1143–1147.

Jaakkola,T. and Jordan,M. (2000) Bayesian parameter estimation via variational methods. *Stat. Comput.*, **10**, 25–37.

Jansen,R. *et al.* (2003) A bayesian networks approach for predicting protein–protein interactions from genomic data. *Science*, **302**, 449–453.

Jensen,L.J. *et al.* (2009) STRING 8–a global view on proteins and their functional interactions in 630 organisms. *Nucleic Acids Res.*, **37**, D412–D416.

Jensen,L.J. and Bork,P. (2008) Biochemistry: not comparable, but complementary. *Science*, **322**, 56–57.

Jordan,M. *et al.* (1999) Introduction to variational methods for graphical models. *Mach. Learn.*, **37**, 183–233.

Kanehisa,M. and Goto,S. (2000) KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.*, **28**, 27–30.

Kass,R. and Raftery,A. (1995) Bayes factors. *J. Am. Stat. Assoc.*, **90**, 773–795.

Krogan,N.J. *et al.* (2006) Global landscape of protein complexes in the yeast Saccharomyces Cerevisiae. *Nature*, **440**, 637–643.

Lee,I. *et al.* (2004) A probabilistic functional network of yeast genes. *Science*, **306**, 1555–1558.

Letunic,I. *et al.* (2006) Smart 5: domains in the context of genomes and networks. *Nucleic Acids Res.*, **34**, D257–D260.

Llewellyn,R. and Eisenberg,D.S. (2008) Annotating proteins with generalized functional linkages. *Proc. Natl Acad. Sci. USA*.

Manning,C. *et al*. (2008) *Introduction to Information Retrieval*. Cambridge University Press.

Margolin,A.A. *et al.* (2006) ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, **7** (Suppl. 1), S7.

Margoln,A. and Califano,A. (2007) Theory and limitations of genetic network inference from microarray data. *Ann. New York Acad. Sci.*

Markowetz,F. *et al.* (2007) Nested effects models for high-dimensional phenotyping screens. *Bioinformatics*, **23**, i305–i312.

Mewes,H.W. *et al.* (2004) MIPS: Analysis and annotation of proteins from whole genomes: analysis and annotation of proteins from whole genomes. *Nucleic Acids Res.*, **32**, D41–D44.

Mulder,N.J. *et al.* (2007) New developments in the interpro database. *Nucleic Acids Res.*, **35**, D224–D228.

Myers,C.L. *et al.* (2005) Discovery of biological networks from diverse functional genomic data. *Genome Biol.*, **6**, R114.

Myers,C.L. *et al.* (2006) Finding function: an evaluation framework for functional genomics. *BMC Genomics*, **7**, 187.

Primig,M. *et al.* (2000) The core meiotic transcriptome in budding yeasts. *Nat. Genet.*, **26**, 415–423.

Qi,Y. *et al.* (2006) Evaluation of different biological data and computational classification methods for use in protein interaction prediction. *Proteins: Struct. Funct. Bioinf.*, **63**, 490–500.

Reguly,T. *et al.* (2006) Comprehensive curation and analysis of global interaction networks in saccharomyces cerevisiae. *J. Biol.*, **5**, 11.

Schervish,M.J. (1996) P values: what they are and what they are not. *Amer. Statistician*, **50**, 203–206.

Schwartz,A.S. *et al.* (2008) Cost-effective strategies for completing the interactome. *Nat. Meth.*, **6**, 55–61.

SGD. Saccharomyces Genome Database. Available at http://www.yeastgenome.org (last accessed date January 3, 2011).

Sharan,R. *et al.* (2005) Conserved patterns of protein interaction in multiple species. *Proc. Natl Acad. Sci. USA*, **102**, 1974–1979.

Song,J. and Singh,M. (2009) How and when should interactome-derived clusters be used to predict functinoal modules and protein function? *Bioinformatics*, **25**, 3143–3150.

Sterne,J.A.C. and Smith,G.D. (2001) Sifting the evidence—what's wrong with significance tests. *BMJ.*, **322**, 226–231.

Tarassov,K. *et al.* (2008) An in vivo map of the yeast protein interactome. *Science*, **320**, 1465–1470.

TRANSFAC. Transcription factor database. Available at http://www.gene-regulation.com/ (last accessed on January 3, 2011).

Troyanskaya,O.G. *et al.* (2003) A bayesian framework for combining heterogeneous data sources for gene function prediction (in saccharomyces cerevisiae). *Proc. Natl Acad. Sci. USA*, **100**, 8348–8353.

Turney,P. (2008) A uniform approach to analogies, synonyms, antonyms, and associations. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING-08)*, Manchester, UK, pp. 905–912.

Uetz,P. *et al.* (2000) A comprehensive analysis of protein–protein interactions in saccharomyces cerevisiae. *Nature*, **403**, 623–627.

von Mering,C. *et al.* (2002) Comparative assessment of large-scale data sets of protein–protein interactions. *Nature*, **417**, 399–403.

von Mering,C. *et al.* (2005) STRING: known and predicted protein–protein associations, integrated and transferred across organisms. *Nucleic Acids Res.*, **33**, D433–D437.

Wang,X.-J. *et al.* (2009) Ranking community answers by modeling question-answer relationships via analogical reasoning. In *Proceedings of the 32nd Annual ACM SIGIR Conference on Research & Development on Information Retrieval*. Boston, MA.

Yu,H. *et al.* (2008) High-quality binary protein interaction map of the yeast interactome network. *Science*, **322**, 104–110.

Yvert,G. *et al.* (2003) Trans-acting regulatory variation in saccharomyces cerevisiae and the role of transcription factors. *Nat. Genet.*, **35**, 57–64.

Zhu,J. and Zhang,M.Q. (1999) SCPD: a promoter database of the yeast Saccharomyces cerevisiae. *Bioinformatics*, **15**, 607–611.